

GETTING STARTED WITH
INIZIARE A USARE

DB2 Express-C

A book for the community by the community
Un libro dalla comunità per la comunità

RAUL CHONG, IAN HAKES, RAV AHUJA

FOREWORD BY DR. ARVIND KRISHNA



SECOND EDITION

Getting Started with DB2 Express-C

Iniziare a usare DB2 Express-C

A book for the community by the community

Un libro dalla comunità per la comunità

Second Edition (April 2008)

This edition has been updated for IBM® DB2® Express-C Version 9.5 for Linux®, UNIX® and Windows®.

© Copyright IBM Corporation, 2007, 2008. All rights reserved.

Seconda Edizione (aprile 2008)

Questa edizione è stata aggiornata per IBM® DB2® Express-C Version 9.5 per Linux®, UNIX® e Windows®.

© Copyright IBM Corporation, 2007, 2008. Tutti i diritti riservati.

Contents / Indice

| | |
|--|-----------|
| About this book / A proposito di questo libro | 11 |
| Notices and Trademarks / Avvisi e Marchi | 11 |
| Who should read this book? / A chi è rivolto questo libro? | 13 |
| How is this book structured? / Come è strutturato questo libro? | 13 |
| A book for the community / Un libro per la comunità | 14 |
| Authors and Contributors / Autori e collaboratori | 15 |
| Acknowledgements / Ringraziamenti | 17 |
| Foreword / Prefazione | 18 |
| | |
| PART I – OVERVIEW AND SETUP / PARTE I – VISIONE GENERALE E SETUP | 21 |
| | |
| Chapter 1 – What is DB2 Express-C? / Capitolo 1 - Che cos'è DB2 Express-C? | 23 |
| 1.1 Free to develop, deploy, and distribute...no limits! / Liberi di sviluppare, utilizzare, e distribuire...senza limiti! | 24 |
| 1.2 User assistance and technical support / Assistenza per l'utente e supporto tecnico | 26 |
| 1.3 DB2 servers / Server DB2 | 27 |
| 1.4 DB2 clients and drivers / Client e driver DB2 | 28 |
| 1.5 Application development freedom / Libertà di sviluppare applicazioni | 30 |
| 1.6 DB2 versions versus DB2 editions / Versioni di DB2 e release di DB2 | 30 |
| 1.7 Moving up to another DB2 edition / Verso la prossima edizione di DB2 | 32 |
| 1.8 Maintenance of DB2 Express-C / Manutenzione di DB2 Express-C | 33 |
| 1.9 Related free software / Software correlato free | 33 |
| 1.9.1 IBM Data Studio | 34 |
| 1.9.2 DB2 Net Search Extender | 35 |
| 1.9.3 Starter Toolkit for DB2 on Rails / Starter Toolkit per DB2 on Rails | 35 |
| 1.9.4 Web 2.0 Starter Toolkit for DB2 / Web 2.0 Starter Toolkit per DB2 | 35 |
| 1.9.5 WebSphere Application Server – Community Edition | 36 |
| | |
| Chapter 2 – Related features and products / Capitolo 2 – Prodotti e feature correlati | 37 |
| | |
| 2.1 Features included with DB2 Express-C subscription / Caratteristiche incluse con l'abbonamento di DB2 Express-C | 41 |
| 2.1.1 Fix packs / Fix pack | 41 |
| 2.1.2 High Availability Disaster Recovery (HADR) | 41 |
| 2.1.3 Data Replication / Replicazione dei Dati | 42 |
| 2.2 Features not available with DB2 Express-C / Caratteristiche non disponibili con DB2 Express-C | 44 |
| 2.2.1 Database Partitioning | 44 |
| 2.2.2 Connection Concentrator | 44 |
| 2.2.3 Geodetic Extender | 45 |
| 2.2.4 Workload Management (WLM) | 45 |
| 2.3 Fee-based products that are related to DB2 / Prodotti a pagamento collegati con DB2 | 46 |
| 2.3.1 DB2 Connect | 46 |
| 2.3.2 WebSphere Federation Server | 47 |
| 2.3.3 WebSphere Replication Server | 48 |
| | |
| Chapter 3 – DB2 installation / Capitolo 3 – Installazione di DB2 | 49 |

| | |
|---|----|
| 3.1 Installation prerequisites / Prerequisiti di installazione | 49 |
| 3.2 Operating system installation authority / L'autorizzazione di installazione del sistema operativo | 50 |
| 3.3 Installation wizard / Wizard di installazione | 51 |
| 3.4 Silent Install..... | 56 |
| QuickLab #1: Install DB2 Express-C and create SAMPLE database / Installare DB2 Express-C e creare il database SAMPLE..... | 59 |

Chapter 4 – DB2 Environment / Capitolo 4 – Ambiente DB2..... 65

| | |
|--|----|
| Quicklab #2 – Create a new database / Creare un nuovo database.... | 79 |
| 4.1 DB2 configuration / Configurazione di DB2 | 81 |
| 4.1.1 Environment variables / Variabili d'ambiente..... | 82 |
| 4.1.2 Database manager configuration file (dbm cfg) / Il file di configurazione del database manager (dbm cfg)..... | 82 |
| 4.1.3 Database configuration file (db cfg) / Il file di configurazione del database (db cfg)..... | 85 |
| 4.1.4 DB2 profile registry / Il registro del profilo di DB2..... | 87 |
| 4.2 The DB2 Administration Server / Il Server di Amministrazione di DB2 | 89 |
| Quicklab #3 – Working with instances, databases and configuration / Lavorare con instance, database e configurazione | 90 |

Chapter 5 – DB2 Tools / Capitolo 5 – Tool DB2 93

| | |
|--|-----|
| 5.1 Control Center..... | 96 |
| 5.2 Command Editor | 100 |
| 5.3 SQL Assist Wizard..... | 104 |
| 5.4 Show SQL Button / Il tasto Show SQL | 105 |
| Quicklab #4: Populating the EXPRESS Database using scripts / Popolare il database EXPRESS usando gli script..... | 107 |
| 5.5 Scripting | 109 |
| 5.5.1 SQL scripts / Script SQL..... | 109 |
| 5.5.2 Operating system (shell) scripts / Script del sistema operative (shell)..... | 112 |
| Quicklab #5: Create an installation script for EXPRESS Database / Creare uno script di installazione per il database EXPRESS | 114 |
| 5.6 Task Center | 119 |
| 5.6.1 The Tools Catalog database / Il database Tools Catalog | 120 |
| 5.7 Journal | 122 |
| 5.8 Health Monitor..... | 125 |
| 5.8.1 Health Center..... | 125 |

PART II – LEARNING DB2: DATABASE ADMINISTRATION / PARTE II – IMPARARE DB2: AMMINISTRAZIONE DEL DATABASE 129

Chapter 6 – DB2 Architecture / Capitolo 6 – L'architettura DB2 131

| | |
|--|-----|
| 6.1 DB2 process model / Modello di processo in DB2 | 131 |
| 6.2 DB2 memory model / Modello di memoria di DB2..... | 135 |
| 6.3 DB2 storage model / Modello di storage di DB2..... | 136 |
| 6.3.1 Pages and extents / Pagine e extent..... | 136 |
| 6.3.2 Buffer pools / I buffer pool | 137 |
| 6.3.3 Table spaces / Spazi di tabella..... | 140 |

| | |
|--|------------|
| Chapter 7 – DB2 Client Connectivity / Capitolo 7 - Connettività del client DB2 | |
| | 149 |
| 7.1 DB2 Directories / Directory di DB2 | 149 |
| 7.2 Configuration Assistant | 151 |
| 7.2.1 Setup required at the server / Setup richiesto al server | 152 |
| 7.2.2 Setup required at the client / Setup richiesto al client | 155 |
| 7.2.3 Creating Client and Server Profiles / Creare i profili del client e del server | 161 |
| Quicklab #6: Using the Configuration Assistant / Usare il Configuration Assistant | 167 |
| | |
| Chapter 8 – Working with Database Objects / Capitolo 8 – Lavorare con gli oggetti di database | 173 |
| 8.1 Schema | 173 |
| 8.2 Tables / Tabelle | 174 |
| 8.2.1 Data types / Tipi di data | 175 |
| 8.2.2 Identity columns / Colonne identity | 179 |
| 8.2.3 SEQUENCE objects / Oggetti di SEQUENCE | 180 |
| 8.2.4 System catalog tables / Tabelle del catalogo di sistema | 181 |
| 8.2.5 Declared temporary tables / Tabelle provvisorie dichiarate .. | 182 |
| Quicklab #7: Creating a new table / Creare un nuova tabella | 184 |
| 8.3 Views / View | 188 |
| 8.4 Indexes / Indici | 188 |
| 8.4.1 Design Advisor | 189 |
| 8.5 Referential integrity / Integrità referenziale | 191 |
| | |
| Chapter 9 – Data Movement Utilities / Capitolo 9 – Utility di Data Movement | 193 |
| 9.1 EXPORT utility / Utility EXPORT | 195 |
| 9.2 IMPORT utility / Utility IMPORT | 196 |
| 9.3 LOAD | 197 |
| 9.4 The db2move utility / Utility db2move | 199 |
| 9.5 The db2look utility / Utility db2look | 200 |
| Quicklab #8 – Extracting DDL for the EXPRESS database / Estrazione del DDL per il database EXPRESS | 203 |
| | |
| Chapter 10 – Database Security / Capitolo 10 – Sicurezza del Database | 209 |
| 10.1 Authentication / Autenticazione | 211 |
| 10.2 Authorization / Autorizzazione | 212 |
| 10.3 DBADM authority / Autorità di DBADM | 215 |
| 10.4 The PUBLIC group / Il gruppo PUBLIC | 216 |
| 10.5 The GRANT and REVOKE statements / Gli statement GRANT e REVOKE | 216 |
| 10.6 Authorization and privilege checking / Controllo di privilegi e autorizzazione | 217 |
| 10.7 Group privilege considerations / Considerazioni sui privilegio dei gruppi | 220 |
| Quicklab #9 – Granting and revoking user permissions / Assegnare e revocare i permessi ad un user | 221 |

| | |
|--|------------|
| Chapter 11 – Backup and Recovery / Capitolo 11 - Backup e Recovery | 227 |
| 11.1 Database Logging / Logging del database | 227 |
| 11.2 Types of logs / Tipi di log | 229 |
| 11.3 Types of logging / Tipi di logging | 230 |
| 11.3.1 Circular logging..... | 230 |
| 11.3.2 Archival logging or log retain / Archival logging o log retain | 231 |
| 11.4 Database logging from the Control Center / Database logging dal Control Center | 233 |
| 11.5 Logging parameters / Parametri di logging..... | 234 |
| 11.6 Database backup / Backup del database | 236 |
| Quicklab #10 – Scheduling a backup / Schedulare un backup | 239 |
| 11.7 Database recovery / Recovery del database..... | 243 |
| 11.7.1 Recovery types / Tipi di recupero | 243 |
| 11.7.2 Database restore / Restore del database..... | 244 |
| 11.8 Other operations with BACKUP and RESTORE / Altri funzionamenti con BACKUP e RESTORE..... | 245 |
| | |
| Chapter 12 – Maintenance Tasks / Capitolo 12 - Attività di Manutenzione | 247 |
| 12.1 REORG, RUNSTATS, REBIND..... | 247 |
| 12.1.1 The REORG command / Il comando REORG | 248 |
| 12.1.2 The RUNSTATS command / Il comando RUNSTATS | 249 |
| 12.1.3 BIND & REBIND | 250 |
| 12.1.4 Maintenance tasks from the Control Center / Attività di manutenzione dal Control Center | 252 |
| 12.2 Maintenance Choices / Scelte di manutenzione..... | 254 |
| Quicklab #11 – Configuring automated maintenance / Configurare la manutenzione automatizzata..... | 257 |
| | |
| Chapter 13 – Concurrency and Locking / Capitolo 13 - Concorrenza e Locking | 261 |
| 13.1 Transactions / Transazioni..... | 261 |
| 13.2 Concurrency / Concorrenza..... | 262 |
| 13.3 Problems without concurrency control / Problemi senza controllo di concorrenza | 265 |
| 13.3.1 Lost update / Update perso | 265 |
| 13.3.2 Uncommitted read / Read non committata | 266 |
| 13.3.3 Non-repeatable read / Read non-ripetibile | 267 |
| 13.3.4 Phantom read / Read fantasma..... | 267 |
| 13.4 Isolation Levels / Livelli di isolamento..... | 268 |
| 13.4.1 Uncommitted read / Read non committato | 269 |
| 13.4.2 Cursor stability / Stabilità del cursore | 269 |
| 13.4.3 Read stability / Stabilità della read | 270 |
| 13.4.4 Repeatable read / Read ripetibile | 270 |
| 13.4.5 Comparing isolation levels / Confronto tra i livelli di isolamento | 272 |
| 13.4.6 Setting the isolation level / Settare il livello di isolamento ... | 273 |
| 13.5 Lock escalation | 274 |
| 13.6 Lock monitoring / Monitoraggio del locking | 275 |
| 13.7 Lock wait..... | 277 |
| 13.8 Deadlock causes and detection / Cause e rilevazione del deadlock | 277 |

| | |
|---|------------|
| 13.9 Concurrency and locking best practices / Scelte ottimali per concorrenza e locking | 281 |
| PART III – LEARNING DB2: APPLICATION DEVELOPMENT / PARTE III – IMPARARE DB2: SVILUPPO DI APPLICAZIONI283 | |
| Chapter 14 – SQL PL Stored Procedures / Capitolo 14 - Stored Procedure di SQL PL | |
| | 285 |
| 14.1 The IBM Data Studio / L'IBM Data Studio | 287 |
| 14.1.2 Create a stored procedure in Data Studio / Creare una stored procedure nel Data Studio | 289 |
| 14.2 SQL PL stored procedures basics / Principi fondamentali delle stored procedure di SQL PL | 293 |
| 14.2.1 Stored procedure structure / Struttura delle stored procedure | 293 |
| 14.2.2 Optional stored procedure attributes / Attributi facoltativi delle stored procedure | 293 |
| 14.2.3 Parameters / Parametri | 294 |
| 14.2.4 Comments in an SQL PL stored procedure / Commenti in una stored procedure di SQL PL | 295 |
| 14.2.5 Compound statements / Statement composti | 295 |
| 14.2.6 Variable declaration / Dichiarazione dei variabile | 296 |
| 14.2.7 Assignment statements / Statement di assegnazione | 297 |
| 14.3 Cursors / Cursori | 297 |
| 14.4 Flow control / Controllo di flusso | 298 |
| 14.5 Calling stored procedures / Chiamare le stored procedure | 299 |
| 14.6 Errors and condition handlers / Handler per errori e condizioni | 301 |
| 14.7 Dynamic SQL / SQL dinamico | 303 |
| Chapter 15 – Inline SQL PL, Triggers, and UDFs / Capitolo 15 – SQL PL inline, Trigger, e UDF | |
| | 305 |
| 15.1 Inline SQL PL / SQL PL inline | 305 |
| 15.2 Triggers / I trigger | 307 |
| 15.2.1 Types of triggers / Tipi di trigger | 307 |
| Quicklab #12 – Creating a trigger in the Control Center / Creare un trigger nel Control Center | 312 |
| 15.3 User-defined functions (UDFs) / Funzioni definite dall'utente (UDFs) | 316 |
| 15.3.1 Scalar functions / Funzioni scalari | 316 |
| 15.3.2 Table functions / Funzioni di tabella | 318 |
| Quicklab #13 – Creating a UDF using the IBM Data Studio / Creare un UDF usando l'IBM Data Studio | 320 |
| Chapter 16 – DB2 pureXML / Capitolo 16 – pureXML DB2 | |
| | 323 |
| 16.1 Using XML with databases / Usare XML con i database | 324 |
| 16.2 XML databases / Database XML | 325 |
| 16.2.1 XML-enabled databases / Database XML-enabled | 325 |
| 16.2.2 Native XML Databases / Database Native XML | 327 |
| 16.3 XML in DB2 | 327 |
| 16.3.1 pureXML technology advantages / I vantaggi della tecnologia pureXML | 330 |

| | |
|---|-----|
| 16.3.2 XPath basics / Principi di XPath | 333 |
| 16.3.3 XQuery defined / XQuery definito | 337 |
| 16.3.4 Inserting XML documents / Inserimento di documenti XML | 339 |
| 16.3.5 Querying XML data / Query su dati XML | 343 |
| 16.3.6 Joins with SQL/XML / Join con SQL/XML | 352 |
| 16.3.7 Joins with Xquery / Join con XQuery | 353 |
| 16.3.8 Update and delete operations / Operazioni di update e delete | 354 |
| 16.3.9 XML indexing / L'indicizzazione XML | 357 |
| QuickLab #14 - SQL/XML and Xquery / SQL/XML e XQuery | 359 |

Chapter 17 –Developing with Java, PHP, and Ruby / Capitolo 17 - Sviluppare con Java, PHP, e Ruby..... 361

| | |
|---|-----|
| 17.1 Application development in Java / Sviluppo di applicazioni in Java | 361 |
| 17.1.1 JDBC Type 2 driver / Il driver JDBC type 2 | 362 |
| 17.1.2 JDBC Type 4 driver / Il driver JDBC type 4 | 363 |
| 17.2 Application development in PHP / Sviluppo di applicazioni in PHP | 364 |
| 17.2.1 DB2 connection options for PHP / Opzioni di connessione di DB2 per PHP | 364 |
| 17.2.2 Zend Core for IBM / Zend Core per IBM | 367 |
| 17.3 Application development in Ruby on Rails / Sviluppo di applicazioni in Ruby on Rails | 370 |
| 17.3.1 Startup Toolkit for DB2 on Rails / Startup Toolkit per DB2 on Rails | 371 |

Appendix A – Troubleshooting / Appendice A – Risoluzione dei problemi 373

| | |
|---|-----|
| A.1 Finding more information about error codes / Trovare maggiori informazioni sui codici di errore | 374 |
| A.2 SQLCODE and SQLSTATE / SQLCODE e SQLSTATE | 375 |
| A.3 DB2 Administration Notification Log / L'Administration Notification Log di DB2 | 376 |
| A.4 db2diag.log | 376 |
| A.5 CLI traces / Trace di CLI | 377 |
| A.6 DB2 Defects and Fixes / Difetti e Fix di DB2..... | 378 |

Resources / Risorse 379

| | |
|---------------------------|-----|
| Web sites / Siti web..... | 379 |
| Books / Libri | 381 |

About this book / A proposito di questo libro

Notices and Trademarks / Avvisi e Marchi

© Copyright IBM Corporation 2007, 2008
All Rights Reserved. / Tutti i diritti riservati.
IBM Canada
8200 Warden Avenue
Markham, ON
L6G 1C7
Canada

Neither this documentation nor any part of it may be copied or reproduced in any form or by any means or translated into another language, without the prior consent of all of the above mentioned copyright owners.

IBM makes no warranties or representations with respect to the content hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. IBM assumes no responsibility for any errors that may appear in this document. The information contained in this document is subject to change without any notice. IBM reserves the right to make any such changes without obligation to notify any person of such revision or changes. IBM makes no commitment to keep the information contained herein up to date.

The information in this document concerning non-IBM products was obtained from the supplier(s) of those products. IBM has not tested such products and cannot confirm the accuracy of the performance, compatibility or any other claims related to non-IBM products. Questions about the capabilities of non-IBM products should be addressed to the supplier(s) of those products.

IBM, the IBM logo, AIX, DB2, DB2

Né questa documentazione né qualsiasi parte di essa può essere copiata o riprodotta in qualsiasi forma o con qualsiasi mezzo o può essere tradotta in un'altra lingua, senza l'autorizzazione di tutti i suddetti proprietari di copyright.

IBM non offre alcuna garanzia o dichiarazione rispetto al contenuto del presente documento e, in particolare, nega qualsiasi garanzia implicita di commerciabilità o idoneità per qualsiasi scopo particolare. IBM non si assume e declina ogni responsabilità per eventuali errori che possono apparire in questo documento. Le informazioni contenute in questo documento sono soggette a modifiche senza preavviso. IBM si riserva il diritto di apportare tali modifiche senza obbligo di notificare qualsiasi persona di tali revisioni o modifiche. IBM non si impegna a mantenere aggiornati le informazioni contenute in questo documento.

Le informazioni contenute in questo documento riguardo a prodotti non-IBM sono state rilasciate dai fornitori di tali prodotti. IBM non ha verificato tali prodotti e non può confermare l'attendibilità delle performance, la compatibilità o qualsiasi altra caratteristica collegata a prodotti non-IBM. Quesiti circa le capacità dei prodotti non IBM devono essere rivolte direttamente ai fornitori di tali prodotti.

IBM, il logo IBM, AIX, DB2, DB2

Connect, DB2 Universal Database, i5/OS, Informix, pureXML, Tivoli, WebSphere, and z/OS are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Connect, DB2 Universal Database, i5/OS, Informix, pureXML, Tivoli, WebSphere, e z/OS sono marchi di fabbrica o marchi registrati dell'International Business Machines Corporation negli Stati Uniti, o in altri paesi, o in entrambi.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Java e tutti i marchi Java-based sono marchi di fabbrica di Sun Microsystems, Inc negli Stati Uniti, o in altri paesi, o in entrambi.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Microsoft e Windows sono marchi di fabbrica di Microsoft Corporation negli Stati Uniti, in altri paesi, o in entrambi.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Linux è un marchio registrato da Linus Torvalds negli Stati Uniti, o in altri paesi, o in entrambi.

UNIX is a registered trademark of The Open Group in the United States, other countries, or both.

UNIX è un marchio registrato da The Open Group negli Stati Uniti, o in altri paesi, o in entrambi.

Other company, product, or service names may be trademarks or service marks of others.

Altri azienda, prodotti, o nomi di servizi possono essere marchi di fabbrica o marchi di servizio di altri.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.

Riferimenti in questa pubblicazione a prodotti o servizi di IBM non implicano che IBM le intenda mettere a disposizione in tutti i paesi in cui IBM opera.

Who should read this book? / A chi è rivolto questo libro?

This book is intended for anyone who works with or intends to work with databases, such as database administrators (DBAs), application developers, consultants, software architects, product managers, instructors, and students.

Questo libro è destinato a chiunque lavori o intenda lavorare con i database, ad esempio gli amministratori di database (DBAs), gli sviluppatori di applicazioni, i consulenti, gli ingegneri del software, i manager dei prodotti, gli insegnanti, e gli studenti.

How is this book structured? / Come è strutturato questo libro?

Part I, Overview and Setup, explains what DB2 Express-C edition is all about, introduces the DB2 family of products and features, assists with installation and creation of databases, and explores the tools available with DB2.

Parte I, Visone generale e Setup, spiega di cosa tratta l'edizione di DB2 Express-C, introduce alla famiglia dei prodotti di DB2 e alle sue caratteristiche, assiste nell'installazione e nella creazione dei database, ed esamina gli strumenti a disposizione di DB2.

Part II, Learning DB2: Database Administration, is designed to familiarize you with the DB2 environment, architecture, remote connectivity, database objects, data movement (import/export/load), security, backup and recovery, concurrency and locking, and other common maintenance tasks.

Parte II, Imparare DB2: Amministrazione di database. È stato creato per farvi familiarizzare con l'ambiente di DB2, l'architettura, la connettività remota, i database object, i dati in movimento (import / export / load), la sicurezza, il backup e il ripristino, la concorrenza e le comuni operazioni di manutenzione.

Part III - Learning DB2: Application Development, covers stored procedures, user defined functions, triggers, SQL/XML, XQuery, development in Java™, PHP and Ruby.

Parte III - Imparare DB2: Sviluppo di applicazioni, riguarda le stored procedure, le funzioni definite dagli utenti, i trigger, SQL/XML, XQuery, lo sviluppo in Java™, PHP e Ruby.

The Appendix contains useful information about troubleshooting.

L'appendice contiene informazioni utili sulla risoluzione dei problemi.

Exercises called "Quicklabs" are provided with most chapters; any input files required for these labs are provided in the zip file **expressc_book_quicklabs_9.5.zip** accompanying this book, or provided in the IBM® DB2 Express-C Web site: www.ibm.com/db2/express.

Esercizi chiamati "Quicklab" sono forniti con la maggior parte dei capitoli; qualsiasi file di input necessario per questi laboratori è fornito nel file zip **expressc_book_quicklabs_9.5.zip** incluso con questo libro, o fornito sul sito web IBM®, DB2 Express-C: www.ibm.com/db2/express.

The materials in this book are also used in courses offered as part of the "DB2 on

I materiali in questo libro sono anche utilizzati in corsi offerti come parte del

Campus" Program, and closely match the e-learning video presentations available at www.channelDB2.com/oncampus.

This book can also help prepare you for the DB2 on Campus exam. This exam provides you with a program completion confirmation that acknowledges receiving 16 hours worth of DB2 training. You can read more about this program at the DB2 Express-C website www.ibm.com/db2/express/students.html

programma "DB2 on Campus", e sono accompagnati dalle presentazioni video di e-learning disponibili su www.channelDB2.com/oncampus.

Questo libro può anche aiutare a preparare l'esame DB2 on Campus. Questo esame fornisce la conferma di aver completato il programma ricevendo un totale di 16 ore di formazione su DB2. Potete leggere di più su questo programma sul sito web di DB2 Express-C www.ibm.com/db2/express/students.html

Note:

For more information about the DB2 on Campus program, watch the video at: <http://www.channeldb2.com/video/video/show?id=807741:Video:3902>

Nota:

Per ulteriori informazioni sul programma DB2 on Campus, guardi il video su: <http://www.channeldb2.com/video/video/show?id=807741:Video:3902>

A book for the community / Un libro per la comunità

This book was created by the DB2 Express-C team and released to the DB2 Express-C community at no-charge. Numerous members of the community from around the world have translated this book into different languages. If you would like to provide feedback, contribute new material, improve existing material, or help with translating this book to another language, please send an email of your planned contribution to db2x@ca.ibm.com with the subject "DB2 Express-C book changes."

Questo libro è stato creato dal team di DB2 Express-C e rilasciato per la community di DB2 Express-C gratuitamente. Numerosi membri della comunità da tutto il mondo hanno tradotto questo libro in diverse lingue. Se volete lasciare un feedback, contribuire alla creazione di nuovi materiali, migliorare il materiale esistente, o aiutare a tradurre questo libro in un'altra lingua, siete pregati di inviare una e-mail con il contributo previsto a db2x@ca.ibm.com con soggetto "DB2 Express-C book changes."

Authors and Contributors / Autori e collaboratori

The following people have provided content and other significant contributions to this book.

Raul F. Chong – Lead Author
Raul is the DB2 on Campus Program Manager at IBM Toronto Lab.

Ian Hakes – Editor
Ian is the DB2 Express-C Community Facilitator at IBM Toronto Lab.

Rav S. Ahuja – Co-author and Publishing
Rav is a senior DB2 Product Manager at IBM Toronto Lab.

We would like to thank the incredible work done by volunteer translators community of DB2! They spent many hours and weekends to ensure that the translation of this book became a reality. The following table lists those volunteers:

Le seguenti persone hanno fornito il contenuto e altri significativi contributi a questo libro.

Raul F. Chong - Autore Principale
Raul è il DB2 on Campus Program Manager agli IBM Toronto Lab.

Ian Hakes - Editor
Ian è il DB2 Express-C Community Facilitator agli IBM Toronto Lab.

Rav S. Ahuja – Co-autore ed editoria
Rav è un senior DB2 Product Manager agli IBM Toronto Lab.

Vorremmo ringraziare l'incredibile lavoro svolto dai traduttori volontari della community di DB2! Hanno trascorso molte ore e week-end per far sì che la traduzione di questo libro diventasse una realtà. La seguente tabella elenca i diversi volontari:

| Name of employee/ Student Nome del dipendente / Studente | Company/ University Società / Università | Title / Titolo | Contribution | Contributo |
|---|---|-----------------------------|--|--|
| Agatha Colangelo | IBM Toronto Lab | DB2 Information Development | Leader of the team of translators/reviewers in Italy. Translated the first draft of all chapters, and reviewed and approved the final version. Also verified the accuracy of the translation, and reviewed all work done by translators and reviewers. | Il leader del team di traduttori in Italia. Ha tradotto la prima bozza di tutti i capitoli, e ha effettuato la revisione finale del lavoro svolto da altri volontari. Inoltre, ha verificato l'esattezza delle traduzioni ed ha esaminato tutto il lavoro fatto dai traduttori e dai revisori. |

| | | | | |
|--------------------|-----------------------|----------|--|--|
| Sergio Matone | Politecnico di Torino | Studente | Corrected and reviewed all the chapters and helped with the translation of all chapters. Coordinated the review of all chapters. | Ha rettificato e riesaminato tutti i capitoli e ha contribuito con la traduzione di tutti i capitoli tradotti. Ha anche coordinato un team di studenti per il lavoro di revisione e riesaminato e diretto la recensione di tutto il libro. |
| Andrea Pignatiello | Politecnico di Torino | Studente | Translated chapters 8 - 9 - 10 - 11 (2nd draft) | Ha tradotto la seconda bozza dei capitoli 8 - 9 - 10 - 11. |
| Edoardo Radica | Politecnico di Torino | Studente | Reviewed chapters 3 - 4 - 5 - 6. | Ha riesaminato i capitoli 3 - 4 - 5 - 6. |
| Paolo Doz | Politecnico di Torino | Studente | Reviewed chapters 12 - 13 - 16; also translated chapter 6 (2nd draft). | Ha riesaminato capitoli 12 - 13 - 16; ha anche tradotto la seconda bozza del capitolo 6. |
| Fabrizio Garetto | Politecnico di Torino | Studente | Reviewed chapters 1 - 2 - 11 - 14; also translated chapter 7 (2nd draft). | Ha riesaminato i capitoli 1 - 2 - 11 - 14; ha anche tradotto la seconda bozza del capitolo 7. |
| Antonio Vetrò | Politecnico di Torino | Studente | Reviewed chapters 7 - 8 - 9 - 10 - 15. | Ha riesaminato i capitoli 7 - 8 - 9 - 10 - 15. |
| Rocco Ruberti | Politecnico di Torino | Studente | Reviewed chapters/sections 17 - Appendix A - Resources. | Ha riesaminato i capitoli 17 - Appendice A - Risorse. |
| Felix Mahimai John | Politecnico di Milano | Studente | Produced the first draft translation of the introduction and chapter 1. | Ha prodotto una prima bozza per l'introduzione e per il capitolo 1. |

Acknowledgements / Ringraziamenti

We greatly thank the following individuals for their assistance and developing materials referenced in this book:

- Ted Wasserman, Clara Liu and Paul Yip from the IBM Toronto Lab who developed materials that served as the framework for this book.
- Don Chamberlin and Cindy Saracco for their IBM developerWorks articles on XQuery, and Matthias Nicola for his presentations on pureXML™.
- Kevin Czap and Grant Hutchison for developing DB2 technical briefing materials.
- Katherine Boyachok for designing the cover of this book.
- Susan Visser for assistance with publishing this book.

Siamo molto lieti di ringraziare le seguenti persone per la loro assistenza e per lo sviluppo di materiali citati in questo libro:

- Ted Wasserman, Clara Liu e Paolo Yip degli IBM Toronto Lab per lo sviluppo di materiali che sono serviti come riferimenti per questo libro.
- Don Chamberlin e Cindy Saracco, sviluppatori IBM, per i loro articoli su XQuery, e Matthias Nicola per le sue presentazioni su pureXML™.
- Kevin Czap e Grant Hutchison per lo sviluppo di insegnamenti di DB2 su materiali tecnici.
- Katherine Boyachok per la progettazione della copertina di questo libro.
- Susan Visser per l'assistenza con la pubblicazione di questo libro.

Foreword / Prefazione

Innovation is the cornerstone of progress in technology. At IBM, innovation has been an integral part of the evolution of our data servers. Having pioneered data management techniques in the 1960s and 1970s, we have continually delivered innovative information management technologies, reflected in the thousands of data management patents authored by IBM's technologists. As a result, some of the largest organizations in the world today rely on IBM products such as DB2 to power their most demanding and mission-critical data management solutions.

However DB2 is not just for large enterprises anymore. With the release of DB2 Express-C, the award-winning DB2 technology is now available to meet the needs of small and mid-size companies – and that with no mandatory cost! Although there are other free or open-source data servers out there, DB2 Express-C offers unique advantages over these alternatives.

There are many technological innovations present in DB2 Express-C. Some of these innovations are aimed at advanced new capability, some at reducing administrative burdens, some at improving performance, and some at reducing infrastructure cost. We will not discuss most of these here, hoping you will be tempted to read the book – but we'll briefly describe one as a teaser.

DB2 Express-C is built on 'Viper' technology, making it the first hybrid data server for managing both relational and XML data in their native formats. This makes DB2 ideal for powering a new breed of SOA and Web 2.0 applications where XML data flows in

L'innovazione è uno dei capisaldi dei progressi in campo tecnologico. All'IBM, l'innovazione è stata una parte integrale dell'evoluzione dei nostri data server. Avendo sperimentato varie tecniche di amministrazione dei dati tra gli anni 60 e 70, abbiamo continuamente apportato tecniche innovative per il management dei dati, che si riflette nelle migliaia di brevetti di gestione di dati appartenenti a sviluppatori di IBM. Come risultato, alcuni delle più grandi organizzazioni nel mondo di oggi contano su prodotti di IBM come DB2 per potenziare le loro soluzioni di gestione dati più esigenti e mission-critical.

DB2 tuttavia non è più solo per le grandi imprese. Con il rilascio di DB2 Express-C, la pluripremiata tecnologia DB2 è ora disponibile per soddisfare le esigenze delle piccole e medie aziende - e questo senza alcun costo obbligatorio! Anche se ci sono altri data server free o open-source in circolazione, DB2 Express-C offre vantaggi unici rispetto a queste alternative.

Ci sono molte innovazioni tecnologiche presenti in DB2 Express-C. Alcune di queste innovazioni si orientano a nuove funzionalità avanzate, altre a ridurre gli oneri amministrativi, altre a migliorare le prestazioni, ed altre ancora a ridurre il costo delle architetture. Non discuteremo la maggior parte di questi aspetti qui, sperando che vi verrà la tentazione di leggere il libro - ma ne descriviamo brevemente uno come anteprima.

DB2 Express-C è costruito sulla tecnologia 'Viper', ottenendo il primo data server ibrido per gestire dati sia relazionali che XML nel loro formato nativo. Ciò rende ideale DB2 per le nuove generazioni di applicazioni basate su SOA e Web 2.0 in cui ci sono

abundance. Unlike data servers from other commercial vendors, DB2 Express-C does not limit the amount of data you can store in a database or the number of databases you can create on a system. And of course, if you require support or assistance from IBM, help is just a click away.

This book serves as a guide to getting started with and using DB2 Express-C. It will assist you with understanding DB2 concepts and enable you to develop skills for DB2 administration and application development. The skills and knowledge gained with the help of this book are very relevant to other advanced editions of DB2 on Linux, UNIX, and Windows.

While DB2 Express-C is not an open-source product, at IBM we very much believe in supporting and fostering community initiatives. I am delighted with this book being developed by DB2 Express-C community members and becoming freely available to anyone in the community. I very much encourage you to enrich and update this book with your know-how, experiences, and assist with translating this book into other languages so others can benefit from your knowledge.



Arvind Krishna
Vice President, Data Servers
Information Management, IBM Software Group

flussi di dati XML in abbondanza. Diversamente dai data server provenienti da altri venditori commerciali, DB2 Express-C non limita la quantità di dati che è possibile memorizzare in una database o il numero di database che è possibile creare in un sistema. E, naturalmente, se avete bisogno di supporto o assistenza da IBM, l'aiuto è a portata di clic.

Questo libro serve come guida per iniziare a utilizzare DB2 Express-C. Vi assisterà per comprendere i concetti di DB2 e vi permetterà di sviluppare competenze per l'amministrazione di DB2 e lo sviluppo di applicazioni. Le competenze e le conoscenze acquisite con l'aiuto di questo libro sono molto importanti in altre edizioni avanzate di DB2 su Linux, UNIX e Windows.

Mentre DB2 Express-C non è un prodotto open-source, all'IBM noi crediamo molto nel supportare e promuovere iniziative di community. Sono lieta che questo libro sia sviluppato dai membri della community di DB2 Express-C e diventi disponibile gratuitamente a chiunque nella community. Vi incoraggio ad arricchire e aggiornare questo libro con il vostro "know-how" e assistere per tradurre questo libro in altre lingue in modo che altri possano beneficiare dalla vostra conoscenza.

**PART I – OVERVIEW AND SETUP /
PARTE I – VISIONE GENERALE E
SETUP**

1

Chapter 1 – What is DB2 Express-C? / Capitolo 1 - Che cos'è DB2 Express-C?

DB2 Express-C is a member of the IBM DB2 family of powerful data server software for managing both relational and XML data. DB2 Express-C is a free, no-limits, and easy to use edition of DB2. The 'C' in DB2 Express-C stands for the Community. A community of DB2 Express-C users that bands together to assist each other, both online and offline. The DB2 Express-C community consists of all sorts of people and companies who design, develop, deploy, or utilize database solutions. Community members include:

- Application developers who require an open standards database software for building standalone, client-server, web-based, and enterprise applications
- ISVs, hardware vendors, infrastructure stack vendors, and other types of solution providers who want to bundle or embed a full-featured data server as part of their solutions
- Consultants, database administrators, and IT architects who need a robust data server for training, skills development, evaluation and prototyping

DB2 Express-C fa parte della famiglia IBM DB2 del potente software di data server per amministrare dati sia relazionali che XML. DB2 Express-C è un'edizione di DB2 gratuita, senza limiti e facile da usare. La 'C' in DB2 Express-C sta per Community. Una community di utenti di DB2 Express-C che collaborano per assistersi l'un l'altro, sia online che offline. La community di DB2 Express-C è formata da tutte le persone e le aziende che progettano, sviluppano, distribuiscono o utilizzano soluzioni database. I membri della community includono:

- Sviluppatori di applicazioni che richiedono software di database basato su standard aperti per la costruzione di applicazioni standalone, client-server, web-based, e applicazioni per le imprese
- ISV, produttori di hardware, venditori di stack di infrastrutture, e altri tipi di fornitori di soluzioni che desiderano includere o incorporare le piene funzionalità dei data server come parte delle loro soluzioni
- Utenti, amministratori di database, creatori di IT ai quali serve un robusto data server per la formazione, lo sviluppo delle competenze, e la valutazione e il prototyping delle loro soluzioni

-
- Startups, small and medium-sized companies who need a reliable data server for their applications and operations
 - Database hobbyists and cutting-edge technology enthusiasts who want an easy to use data server for building Web 2.0 and next generation applications
 - Students, teachers, and other academic users who want a highly versatile data server for teaching, courseware, projects and research
 - Startup, piccole e medie imprese che hanno bisogno di un data server affidabile per le loro applicazioni e operazioni
 - Persone con l'hobby per i database e appassionati d'avanguardia della tecnologia che desiderano un data server facile da utilizzare per la costruzione di applicazioni Web 2,0 per la prossima generazione di applicazioni
 - Studenti, insegnanti, e altri utenti accademici che desiderano un data server versatile per l'insegnamento, i corsi, i progetti e la ricerca

DB2 Express-C shares the same core functionality and code-base as the other priced editions of DB2 on Linux, UNIX, and Windows. DB2 Express-C can be run on either 32-bit or 64-bit systems with Linux or Windows operating systems. It can be run on systems with any amount of processors and memory and does not have any specialized storage or system setup requirements. DB2 Express-C also includes pureXML at no charge. pureXML is DB2's unique technology to store and process XML documents natively.

DB2 Express-C condivide le stesse funzionalità fondamentali e il codice base delle altre edizioni a pagamento di DB2 su Linux, UNIX e Windows. DB2 Express-C può essere eseguito con sistemi sia a 32-bit che 64-bit, con sistemi operativi Linux o Windows. Può essere eseguito su sistemi con un qualsiasi numero di processori e memoria e non ha alcun limite di storage o di requisiti di sistema di installazione. DB2 Express-C include anche pureXML senza alcun costo. pureXML è la tecnologia unica di DB2 per memorizzare ed elaborare documenti XML in modo nativo.

1.1 Free to develop, deploy, and distribute...no limits! / Liberi di sviluppare, utilizzare, e distribuire...senza limiti!

This sentence summarizes the key ideals behind DB2 Express-C:

Free to develop: If you are an application developer and need a database for your application, you can use DB2 Express-C.

Questa frase riassume l'idea chiave che sta dietro a DB2 Express-C:

Liberi di sviluppare: Se sei uno sviluppatore di applicazioni e hai bisogno di un database per la tua applicazione, puoi utilizzare DB2 Express-C.

Free to deploy: If you are working in a production environment, and need a data management system to store your vital records, you can use DB2 Express-C.

Free to distribute: If you are developing an application or a tool that requires an embedded data server, you can include DB2 Express-C. Even though DB2 Express-C is embedded in your application, and distributed every time you sell your application, it is still free. You are required to register with IBM in order to re-distribute DB2 Express-C; however this registration is also free.

No limits: While other competitor database offerings set limits on database sizes, number of databases, and number of users, with DB2 Express-C there are NO data limits. Your database can grow and grow without violating the licensing agreement. There are also no limits in terms of the number of connections or users per server.

Liberi di utilizzare: Se lavori in un ambiente produttivo, ed hai bisogno di un sistema di gestione dati per memorizzare record di vitale importanza, puoi utilizzare DB2 Express-C.

Liberi di distribuire: se stai sviluppando un'applicazione o uno strumento che richiede un data server embedded, puoi includere DB2 Express-C. Anche se DB2 Express-C è incorporato nella tua applicazione, ed è distribuito ogni volta che vendi la tua applicazione, è ancora free. È richiesta la registrazione presso IBM per ridistribuire DB2 Express-C, tuttavia anche questa registrazione è gratuita.

Senza limiti: Mentre le altre offerte concorrenti di database fissano limiti su le dimensioni dei database, sui numeri di database, e sui numeri di utenti, con DB2 Express-C non ci sono alcuni limiti sui dati. Il vostro database può crescere e crescere senza violare gli accordi di licenza. Inoltre, non esistono limiti in termini di numero di connessioni o numero di utenti per server.

Note:

To learn more about DB2 Express-C and its role in the information on-demand world and Web 2.0, take a look at these video presentations:

<http://www.channeldb2.com/video/video/show?id=807741:Video:3922>

<http://www.channeldb2.com/video/video/show?id=807741:Video:3942>

Nota:

Per ulteriori informazioni su DB2 Express-C e il suo ruolo nel mondo dell'informazione on-demand e nel Web 2,0, dai un'occhiata a queste video di presentazione:

<http://www.channeldb2.com/video/video/show?id=807741:Video:3922>

<http://www.channeldb2.com/video/video/show?id=807741:Video:3942>

1.2 User assistance and technical support / Assistenza per l'utente e supporto tecnico

If you have technical questions about DB2 Express-C, you can post your questions in the DB2 Express-C forum. This free forum is monitored by a dedicated DB2 Express-C team, though it is the community who provides most of the answers on a voluntary basis.

IBM also gives users the choice to purchase a low cost yearly subscription (also known as 12 Months License and Subscription or Fixed Term License or FTL). This subscription for DB2 Express-C comes with the backing of IBM for 24 x 7 technical support and software updates.

In addition to support and software maintenance, with the yearly low cost subscription fee (\$2,995 per Server per Year in the United States – may vary in other countries) you also get to use two key additional features: HADR (clustering for High Availability and Disaster Recovery), and SQL replication (for replicating data with other DB2 servers). Further information about DB2 Express-C subscription can be found at:

www.ibm.com/db2/express/support.html

Se avete domande tecniche su DB2 Express-C, potete postarle nel forum DB2 Express-C. Questo forum gratuito è monitorato da un team dedicato a DB2 Express-C, anche se è la community che fornisce la maggior parte delle risposte volontariamente.

IBM inoltre fornisce agli utenti la possibilità di acquistare un abbonamento annuale a basso costo (noto anche come 12 Months License and Subscription, Fixed Term License oppure FTL). Questo abbonamento per DB2 Express-C viene fornito con il supporto di IBM per 24 x 7 (24 su 24) di supporto tecnico e software aggiornamenti.

In aggiunta al supporto e alla manutenzione del software, con il pagamento della sottoscrizione annuale (\$ 2.995 all'anno per server negli Stati Uniti – può variare in altri paesi), potete anche usare due principali caratteristiche aggiuntive: HADR (clustering per High Availability e Disaster Recovery), SQL replication (per la replica dei dati con altri server di DB2). Ulteriori informazioni sull'abbonamento a DB2 Express-C possono essere reperite su:

1.3 DB2 servers / Server DB2

All DB2 server editions contain the same core components; they are packaged in such a way that users can choose the functions they need at the right price. Figure 1.1 illustrates the different DB2 product editions.

Tutte le edizioni del server DB2 contengono gli stessi componenti di base, sono fornite in un modo tale che gli utenti possano scegliere le funzioni di cui hanno bisogno al prezzo giusto. La figura 1.1 illustra le diverse edizioni del prodotto DB2.

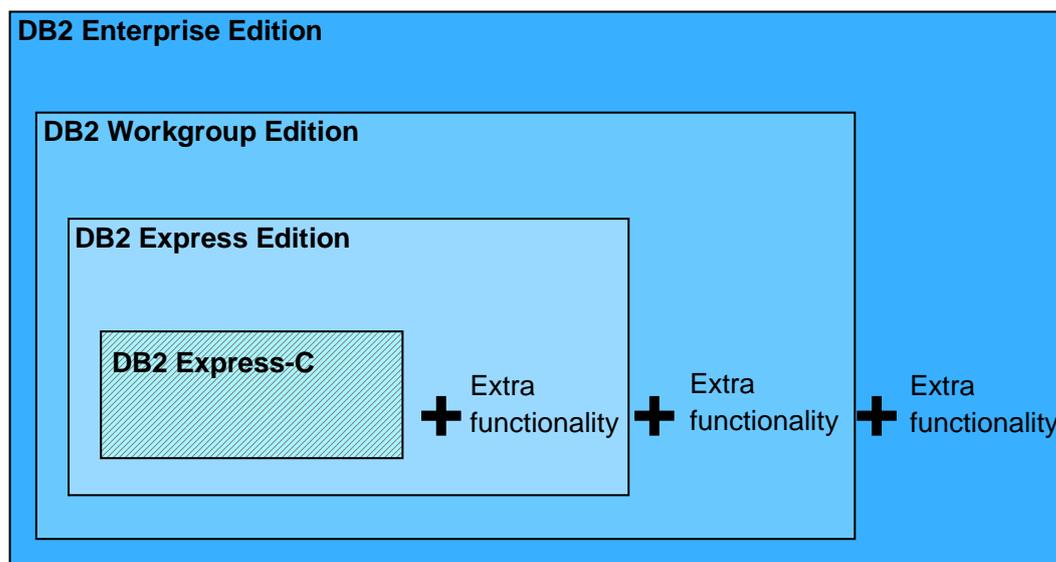


Figure 1.1 – DB2 Servers / Figura 1.1 – Server DB2

As shown by Figure 1.1, DB2 Express-C is the same as DB2 Express without a few components. DB2 Express-C is free to the community. As mentioned earlier, technical assistance is available through a free online forum, or you can receive official 24 x 7 IBM DB2 technical support if you purchase the 12 month subscription license.

Come mostrato in figura 1.1, DB2 Express-C è analogo a DB2 Express eccetto che per alcuni componenti. DB2 Express-C è gratis per la community. Come già accennato, l'assistenza tecnica è disponibile gratuitamente attraverso un forum on-line, oppure si può ricevere ufficialmente il supporto tecnico DB2 24 x 7 (24 su 24) se si acquista la licenza di sottoscrizione di 12 mesi.

Figure 1.1 also explains why it is so easy to upgrade from DB2 Express-C. If you wish to upgrade to any of the other DB2 servers in the future, all DB2 servers have the same core components. This also means that any application developed for one edition

La figura 1.1 spiega anche il motivo per cui è così facile effettuare un upgrade da DB2 Express-C. Se desiderate effettuare l'aggiornamento a uno qualsiasi degli altri server di DB2 in futuro, tutti i server di DB2 hanno gli stessi componenti come nucleo. Questo

will work, without modification, in other edition. And any skills you learn in one edition will apply to other editions.

significa anche che qualsiasi applicazione sviluppata per una edizione funziona, senza modifiche, in altre edizioni. E le eventuali competenze imparate con una edizione, si applicheranno alle altre edizioni.

1.4 DB2 clients and drivers / Client e driver DB2

A DB2 client includes the necessary functionality to connect to a DB2 server; however, a DB2 client does not always need to be installed. For example, a JDBC Type 4 application can connect directly to a DB2 server. DB2 Clients and drivers come in several different flavors:

- IBM Data Server Client: most complete, includes GUI Tools, drivers
- IBM Data Server Runtime Client: a lightweight client with basic functionality, and includes drivers
- DB2 Runtime Client Merge Modules for Windows: mainly used to embed a DB2 runtime client as part of a Windows application installation
- IBM Data Server Driver for JDBC and SQLJ: allows Java applications to connect to DB2 servers without having to install a full client
- IBM Data Server Driver for ODBC and CLI: allows ODBC and CLI applications to connect to a DB2 server without the large footprint of having to install a Client
- IBM Data Server Driver for ODBC, CLI and .NET: a Windows-specific driver with support for .NET environments in addition to ODBC and CLI

Figure 1.2 shows the different DB2 clients and drivers available.

Un client DB2 include le funzionalità necessarie per connettersi ad un server DB2; tuttavia, un client DB2 non sempre necessita di essere installato. Ad esempio, un'applicazione JDBC Type 4 può connettersi direttamente a un server DB2. I client e i drivers DB2 sono disponibili in diverse modalità:

- IBM Data Server Client: il più completo, comprende strumenti GUI, driver
- IBM Data Runtime Server Client: un client "lightweight" con funzionalità di base, che include i driver
- DB2 Runtime Client Merge Modules per Windows: principalmente utilizzato per incorporare un client DB2 runtime come parte di una installazione di Windows
- IBM Data Server Driver per JDBC e SQLJ: permette alle applicazioni Java di connettersi a server DB2 senza dover installare un client completo.
- IBM Data Server Driver per ODBC e per CLI: permette ad applicazioni per ODBC e per CLI di connettersi a un server DB2 senza l'aggravio di dover installare un Client
- IBM Data Server Driver per ODBC, CLI, e NET: uno driver specifico per Windows con il supporto per ambienti .NET in aggiunta a ODBC e CLI

La figura 1.2 mostra i differenti client e driver DB2 disponibili.

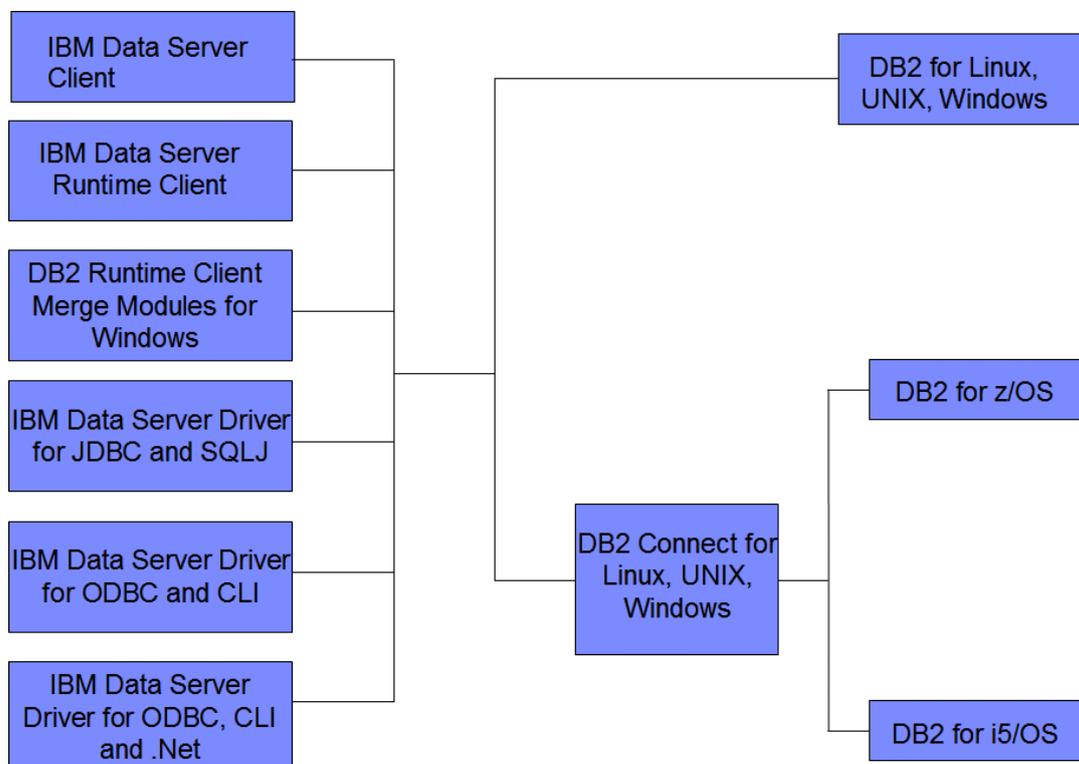


Figure 1.2 – DB2 clients and drivers / Figura 1.2 Client e driver DB2

On the left side of Figure 1.2, all the DB2 clients and drivers are shown. Although all DB2 clients include the required drivers, starting with DB2 9 we provide the individual drivers as well. DB2 clients and drivers are all free and available for download from the DB2 Express-C web site. The clients and drivers can be used to connect to a DB2 server on Linux, UNIX or Windows. To connect to a DB2 for z/OS® or DB2 for i5/OS® server, you will need to go through a DB2 Connect™ server (shown in the middle of Figure 1.2). We will discuss the DB2 Connect software in Chapter 2.

Sul lato sinistro della figura 1.2, vengono mostrati tutti i client e i driver di DB2. Nonostante tutti i client DB2 includano i driver necessari, a partire da DB2 9 mettiamo a disposizione anche singoli driver. I client e i driver sono tutti gratuiti e disponibili per il download al sito web di DB2 Express C. I client e i driver possono essere utilizzati per connettersi a un server DB2 su Linux, Unix o Windows. Per connettersi a DB2 per z/OS® o DB2 per i5/OS® server, sarà necessario passare attraverso un server DB2 Connect™ (mostrato al centro della figura 1.2). Discuteremo il software DB2 Connect nel capitolo 2.

1.5 Application development freedom / Libertà di sviluppare applicazioni

DB2 offers an application development environment that is standards-based and is transparent across the DB2 family. SQL standardization across the DB2 product line provides a common set of application programming interfaces for database access. In addition, each DB2 product provides SQL pre-compilers which allow developers to embed static and dynamic SQL in portable application programs. DB2 even has a native .NET managed provider and integration with Microsoft® Visual Studio tools.

DB2 offre un ambiente di sviluppo di applicazioni che è basato su standard ed è trasparente all'interno della famiglia DB2. Tra tutta la linea di prodotti DB2 lo standard SQL fornisce un insieme comune di interfacce di programmazione di applicazioni per l'accesso a database. Inoltre, ogni prodotto DB2 fornisce pre-compilatori SQL che consentono agli sviluppatori di includere SQL statico e dinamico in applicazioni portabili. DB2 dispone anche di un gestore .NET nativo e dell'integrazione con i tool di Microsoft® Visual Studio.

Languages and standards you can use with DB2 include:

- SQL, XQuery, XPath
- C/C++ (CLI, ODBC and embedded SQL)
- Java (JDBC and SQLJ)
- COBOL
- PHP
- Perl
- Python
- Ruby on Rails
- .NET languages
- OLE-DB
- ADO
- MS Office: Excel, Access, Word
- Web services

I linguaggi e gli standard che potete utilizzare con DB2 comprendono:

- SQL, XQuery, Xpath
- C / C + + (CLI, ODBC e embedded SQL)
- Java (JDBC e SQLJ)
- COBOL
- PHP
- Perl
- Python
- Ruby on Rails
- .NET lingue
- OLE-DB
- ADO
- MS Office: Excel, Access, Word
- Web services

1.6 DB2 versions versus DB2 editions / Versioni di DB2 e release di DB2

If you are new to DB2, you may be a bit confused as to the distinction between a DB2 version, and a DB2 edition.

Se siete nuovi di DB2, potreste essere un po' confusi dalla distinzione tra versione DB2 e edizione DB2.

Every few years, IBM publicly releases a new DB2 Version. A Version includes new features and significant improvements to the product. Currently,

Saltuariamente, IBM rilascia al pubblico una nuova versione di DB2. Una nuova versione include nuove funzionalità e significativi miglioramenti del prodotto.

DB2 Version 8 and Version 9 are officially supported by IBM. A Version may also have a few Releases which can include some new functionality but usually not significant enough to warrant a new Version. For example 8.1 and 8.2 are Release levels for DB2 Version 8. Going by the past history, IBM seems to come out with a new Release of DB2 almost every year, however new Versions are typically spaced 2-3 years apart. The most current release is V9.5 (previously code-named as DB2 'Viper 2') and became Generally Available (GA) in October of 2007.

Each release may also have several Modification levels, which typically contain fixes or correspond to fix pack levels, and seldom contain new functionality. At the time of writing the most current Version, Release, Modification (V,R,M) level of DB2 Express-C is 9.5.0 which corresponds to a code-level of 9.5 with Fix pack 0, which means it is at GA level.

On the other hand, editions are select offerings or package groupings within each version. As discussed earlier, an edition is a packaging of different functions for a given price and license. DB2 Version 9.5 (also known as DB2 9.5) has several editions; for example, DB2 Express-C 9.5, DB2 Express 9.5, DB2 Workgroup 9.5, and DB2 Enterprise 9.5 (see Figure 1.1).

Attualmente, DB2 versione 8 e versione 9 sono ufficialmente supportati da IBM. Una versione può anche avere delle release che possono includere alcune nuove funzionalità, ma di solito non abbastanza significative per giustificare una nuova versione. Per esempio 8.1 e 8.2 sono i livelli di rilascio per DB2 Versione 8. Andando indietro nel tempo, IBM sembra rilasciare nuove release di DB2 quasi ogni anno, tuttavia nuove versioni escono generalmente a distanza di 2-3 anni di distanza. La versione più recente è la V9.5 (precedentemente denominata con il nome in codice di DB2 'Viper 2') e divenne disponibile al pubblico (Generally Available - GA) nel mese di ottobre del 2007.

Ogni release può anche avere diversi livelli di Modification, che di solito contengono correzioni o corrispondono ai livelli di fix pack, e raramente contengono nuove funzionalità. In questo momento il più recente livello di versione, release, modification (V,R,M) di DB2 Express-C è 9.5.0 che corrisponde a un code-level di 9.5 con Fix pack 0, il che significa che è a un livello GA.

D'altra parte, le release sono scelte o raggruppate in package all'interno di ogni versione. Come già discusso, una release è un package di diverse funzioni ad un determinato prezzo e licenza. DB2 Versione 9.5 (nota anche come DB2 9.5) ha diverse edizioni, ad esempio, DB2 Express-C 9,5, DB2 Express 9,5, DB2 Workgroup 9.5, e DB2 Enterprise 9.5 (vedete figura 1.1).

1.7 Moving up to another DB2 edition / Verso la prossima edizione di DB2

As your database needs grow, you may need to upgrade to a DB2 edition that supports a larger hardware configuration. If this situation arises, it is easy to upgrade to another DB2 edition:

Poiché il vostro database ha bisogno di crescere, potrebbe essere necessario effettuare l'upgrade a un'edizione di DB2 che supporta una configurazione hardware più grande. Se questa situazione si verifica, sarà facile effettuare l'aggiornamento a un'altra edizione di DB2:

- If you are upgrading to another DB2 edition on the same computer system, install the new DB2 edition on top of DB2 Express-C, and the corresponding new license. Your databases will not be deleted (but a backup is always recommended)
 - If you are upgrading DB2 where the new edition will be installed on a different, larger computer using the same operating system, install the new DB2 edition on the larger computer, backup your databases from the smaller computer, move the backup images to the larger computer, and restore from the backup images the databases on the larger computer. You may also need to save the instance configuration settings (`dbm cfg`) from your smaller computer, and apply this configuration to the larger computer. The `backup` and `restore` commands are discussed in more details in Chapter 11, Backup and Recovery. The `dbm cfg` is discussed in more detail in Chapter 4, DB2 Environment.
 - In either case your application will not need modification.
- Se state aggiornando ad un'altra edizione di DB2 sullo stesso computer, installate la nuova edizione di DB2 sopra DB2 Express-C, con la corrispondente nuova licenza. I vostri database non verranno eliminati (ma un backup è sempre consigliato)
 - Se state aggiornando DB2 dove la nuova edizione verrà installata su un altro computer più performante utilizzando lo stesso sistema operativo, installate la nuova edizione DB2 su nuovo computer, fate un backup del database dal vecchio computer, spostate le immagini di backup al nuovo computer, e ripristinate dalle immagini di backup i database su questo computer. Potrebbe anche essere necessario salvare la configurazione della "instance" (`dbm cfg`) dal precedente computer, e applicare questa configurazione al computer corrente. I comandi di `backup` e `restore` sono discussi in dettaglio nel capitolo 11, Backup e Recovery. Il `dbm cfg` è discusso in dettaglio nel capitolo 4, Ambiente di DB2.
 - In entrambi i casi le vostre applicazioni non richiederanno modifiche.

1.8 Maintenance of DB2 Express-C / Manutenzione di DB2 Express-C

As discussed earlier, there are two support options for DB2 Express-C:

1. Buy the 12 month subscription license. This provides you with full time coverage from IBM DB2 Technical support, gives you the ability to install DB2 software updates (also called fixpacks).
2. Use the online DB2 Express-C community forum. This is totally free, but comes with no official support from IBM. Also, under this option, IBM does not commit to provide new features and bug fixes at scheduled dates. The concept of a fixpack, which is discussed in Chapter 2, does not apply either; instead, refreshes of the entire DB2 Express-C image are made available from time to time. As new releases come out, you can typically expect refreshed DB2 Express-C images to be available for the new releases rather than the older ones.

Come discusso precedentemente, ci sono due scelte di supporto per DB2 Express-C:

1. Acquistare la licenza di sottoscrizione di 12 mesi. Questa vi fornisce la copertura a tempo pieno dell'IBM DB2 Technical Support, e vi dà la possibilità di installare aggiornamenti del software DB2 (chiamati anche fix packs).
2. Utilizzare il forum on-line della community di DB2 Express-C. Questo è totalmente gratuito, ma è privo del supporto ufficiale di IBM. Inoltre, in virtù di questa scelta, IBM non si impegna a fornire nuove funzionalità e bug-fix in date precise. Nemmeno il concetto di fix pack, discusso nel capitolo 2, è valido; invece, aggiornamenti dell'intera immagine di DB2 Express-C sono messi a disposizione di volta in volta. Appena nuove release sono distribuite, ci si può aspettare tipicamente immagini di DB2 Express-C aggiornate messe a disposizione per nuove release piuttosto che per le precedenti.

1.9 Related free software / Software correlato free

All the software that is available for download from the DB2 Express-C download page:

(www.ibm.com/db2/express/download.html)

is free of charge. Besides the images for DB2 Express-C (for Linux and Windows, both 32 and 64-bit architectures), there is other useful software that can be downloaded and used for free:

- IBM Data Studio
- DB2 Net Search Extender
- DB2 Spatial Extender

Tutto il software disponibile per il download dalla pagina di download DB2 Express-C:

è gratuito. Oltre le immagini per DB2 Express-C (per Linux e Windows, sia a 32 che a 64 bit), c'è altro software utile che può essere scaricato e utilizzato gratuitamente:

- IBM Data Studio
- DB2 Net Search Extender
- DB2 Spatial Extender

There are also additional starter toolkits based on DB2 Express-C and available for download from IBM Alphaworks web site:

www.alphaworks.ibm.com/datamgmt)

that you may find useful:

- Starter Toolkit for DB2 on Rails
www.alphaworks.ibm.com/tech/db2onrails/)
- Web 2.0 Starter Toolkit for DB2
www.alphaworks.ibm.com/tech/web2db2)
- Starter Toolkit per DB2 on Rails
- Web 2.0 Starter Toolkit per DB2

If you are looking for a lightweight application server that is free, IBM offers:

- WebSphere® Application Server – Community Edition (WAS CE)

Ci sono anche starter toolkit aggiuntivi basati su DB2 Express-C disponibili per il download dal sito web IBM

Alphaworks:

che possono essere utili:

Se cercate un application server lightweight che sia gratuito, IBM offre:

1.9.1 IBM Data Studio

IBM Data Studio is a tool based on Eclipse that allows you to design, develop, deploy and manage your data, databases, and database applications throughout the entire data management life cycle. Data Studio replaces the previously available DB2 Developer Workbench 9.1 solution. IBM Data Studio helps you develop user-defined functions, stored procedures, XQuery, SQL statements, and includes an integrated debugger. In addition, Data Studio allows you to work with Physical Data Modeling diagrams to understand entity relationships between tables. It can also help you to develop and publish data as a Web service without programming. We will discuss Data Studio in Chapter 14, SQL PL stored procedures.

IBM Data Studio è un tool basato su Eclipse che permette il design, lo sviluppo, il dispiegamento e la gestione dei dati, dei database e delle applicazioni di database attraverso l'intero ciclo vitale della gestione dei dati. Data Studio rimpiazza la soluzione disponibile precedentemente, DB2 Developer Workbench 9.1. IBM Data Studio vi aiuta nello sviluppo di funzioni user-defined, stored procedure, XQuery, statement SQL, e include un debugger integrato. In aggiunta, Data Studio vi permette di lavorare con diagrammi di Physical Data Modeling per comprendere a pieno le relazioni tra le tabelle. Vi aiuterà anche nello sviluppo e nella pubblicazione di dati come un web service senza programmarlo. Discuteremo di Data Studio nel capitolo 14, Stored Procedure di SQL PL

1.9.2 DB2 Net Search Extender

With DB2 Net Search Extender, you can execute fast and detailed full-text searches in text documents, including any XML documents stored natively in DB2.

Con DB2 Net Search Extender, potrete eseguire ricerche veloci e dettagliate di parole e testi all'interno di documenti di testo, incluso qualsiasi documento XML salvato nativamente in DB2.

1.9.3 Starter Toolkit for DB2 on Rails / Starter Toolkit per DB2 on Rails

Starter Toolkit for DB2 on Rails is a conveniently-packaged set of products and technologies that enables the quick creation of an environment to build DB2 Web applications using Ruby on Rails technology. All required software is included: DB2 Express-C; DB2 driver for Ruby; DB2 adapter for Rails; along with tutorials, examples, and other learning materials. We will discuss Ruby on Rails further in Chapter 17, Development in Java, PHP and Ruby.

Lo Starter Toolkit per DB2 on Rails è un pacchetto opportunamente confezionato che racchiude un insieme di prodotti e tecnologie che permettono la creazione rapida di un ambiente per creare applicazioni web DB2 usando la tecnologia Ruby on Rails. Tutto il software necessario è incluso: DB2 Express-C; il driver DB2 per Ruby; l'adapter DB2 per Rails; attraverso tutorial, esempi ed altro materiale. Discuteremo Ruby on Rails in seguito nel Capitolo 17, Sviluppo in Java, PHP e Ruby.

1.9.4 Web 2.0 Starter Toolkit for DB2 / Web 2.0 Starter Toolkit per DB2

Web 2.0 Starter Toolkit for DB2 is an easy way to get started with DB2, PHP, and Dojo. It helps you deploy the required software, links you to tutorials, and includes demo applications. Two of the demo applications are the Atom Feed Control Panel, which generates Atom feeds from DB2 tables, and the Web Services Control Panel, which creates REST web service wrappers around your DB2 tables. Both rely on Dojo for its significant Ajax and widget capabilities.

Web 2.0 Starter Toolkit per DB2 è un modo semplice per iniziare con DB2, PHP, e Dojo. Aiuta nello sviluppo del software richiesto, rimanda a collegamenti a tutorial e include applicazioni demo. Due delle applicazioni demo sono il pannello di controllo per Atom Feed, che genera feed Atom da tabelle DB2, e il pannello di controllo per Web Services, che crea dei wrapper del web service REST partendo dalle tabelle DB2. Entrambe si basano su Dojo per le significative capacità di Ajax e dei widget.

1.9.5 WebSphere Application Server – Community Edition

IBM WebSphere Application Server Community Edition is a lightweight Java EE 5 application server available free of charge. Built on Apache Geronimo technology, it harnesses the latest innovations from the open-source community to deliver an integrated, readily accessible and flexible foundation for developing and deploying Java applications. Optional technical support for WAS CE is available through annual subscription.

IBM WebSphere Application Server Community Edition è un application server basato su JAVA EE 5 leggero e gratuito. Costruito sulla tecnologia Apache Geronimo, ingloba le ultime innovazioni dalla community open-source per fornire una base integrata, facilmente accessibile e flessibile per lo sviluppo e la distribuzione di applicazioni Java. Supporto tecnico facoltativo per WAS CE è disponibile attraverso un abbonamento annuale.

2

Chapter 2 – Related features and products / Capitolo 2 – Prodotti e feature correlati

This chapter describes DB2 features included with the purchase of a DB2 Express-C 12 month subscription license. It also describes features included with other DB2 editions, in some cases, for an additional fee.

Questo capitolo descrive le caratteristiche di DB2 incluso l'acquisto di una DB2 Express-C 12 month subscription license. Descrive anche le caratteristiche incluse in altre edizioni DB2, in alcuni casi, con un pagamento aggiuntivo.

Capabilities included in the free edition of DB2 Express-C:

- Core DB2 functionality
- Control Center, Data Studio and other management tools
- pureXML
- Resource utilization of up to 2GB and 2 processor cores
- Availability on Linux, Windows, and Solaris (x86)

Le capacità incluse nell'edizione gratuita di DB2 Express-C sono:

- Funzionalità di base di DB2
- Control Center, Data Studio ed altri strumenti di gestione
- pureXML
- Utilizzazione delle risorse fino a 2GB e processori a 2 core
- Disponibilità su Linux, Windows e Solaris (x86)

Features not available in the free edition of DB2 Express-C but included with the DB2 Express-C 12 month subscription license are:

- Fix packs
- High Availability
- Data Replication (Homogenous SQL)
- Resource utilization of up to 4GB and 4 processor cores (on 2 sockets)

Le caratteristiche non disponibili nell'edizione free di DB2 Express-C ma incluse con DB2 Express-C con 12 month subscription license sono:

- Fix packs
- High Availability
- Data Replication (Homogenous SQL)
- Utilizzo delle risorse fino a 4GB e con processori a 4 core (su 2 socket)

The following table lists product features and whether they are included with the different editions of DB2. Features that you can purchase separately are listed by name for the corresponding DB2 edition.

La seguente tabella elenca le caratteristiche del prodotto e indica se sono incluse con le diverse edizioni di DB2. Le caratteristiche che potete acquistare separatamente sono elencate per edizione DB2 corrispondente.

| Function/ Funzione | Express-C fixed term license / Express-C licenza a scadenza fissa | DB2 Express Edition | DB2 Workgroup Server Edition | DB2 Enterprise Server Edition |
|--|---|--------------------------------------|--------------------------------------|--------------------------------------|
| Homogenous SQL Replication | Yes / Sì | Yes / Sì | Yes / Sì | Yes / Sì |
| Net Search Extender | Yes / Sì | Yes / Sì | Yes / Sì | Yes / Sì |
| Spatial Extender | Yes / Sì | Yes / Sì | Yes / Sì | Yes / Sì |
| pureXML™ technology | Yes / Sì | pureXML Feature | pureXML Feature | pureXML Feature |
| High availability disaster recovery | Yes / Sì | High Availability Feature | Yes / Sì | Yes / Sì |
| Tivoli® System Automation | Yes / Sì | High Availability Feature | Yes / Sì | Yes / Sì |
| Advanced Copy Services | No | High Availability Feature | Yes / Sì | Yes / Sì |
| Online reorganization | No | High Availability Feature | Yes / Sì | Yes / Sì |
| Homogenous Federation | No | Homogeneous Federation Feature | Homogeneous Federation Feature | Homogeneous Federation Feature |
| MQT | No | No | Query Optimization Feature | Yes / Sì |
| MDC | No | No | Query Optimization Feature | Yes / Sì |
| Query parallelism | No | No | Query Optimization Feature | Yes / Sì |
| Connection concentrator | No | No | No | Yes / Sì |
| Table partitioning | No | No | No | Yes / Sì |

| Function/ Funzione | Express-C fixed term license / Express-C licenza a scadenza fissa | DB2 Express Edition | DB2 Workgroup Server Edition | DB2 Enterprise Server Edition |
|---|---|------------------------|---------------------------------|--|
| DB2 Governor | No | No | No | Yes / Sì |
| Compression: row level | No | No | No | Storage Optimization Feature |
| Compression: backup | No | No | No | Storage Optimization Feature |
| Label-based access control (LBAC) | No | No | No | Advanced Access Control Feature |
| Geodetic Extender | No | No | No | Geodetic Data Management Feature |
| Query Patroller | No | No | No | Performance Optimization Feature |
| DB2 workload management | No | No | No | Performance Optimization Feature |
| Performance Expert | No | No | No | Performance Optimization Feature |
| Homogenous Q Replication | No | No | No | Homogeneous Replication Feature |
| Database partitioning | No | No | No | No |

Table 2.1: DB2 Version 9.5 editions: feature and function support / Tabella 2.1: Edizioni di DB2 Versione 9.5: caratteristiche e funzioni

Features available with other DB2 editions are:

Le caratteristiche disponibili con altri edizioni di DB2 sono:

Chargeable DB2 Express Edition

Caratteristiche a pagamento di DB2

Features:

Express Edition:

- pureXML
- High Availability
- Homogenous Federation Feature

Features included at no-charge in DB2 Workgroup Edition:

Caratteristiche incluse gratuitamente nella DB2 Workgroup Edition:

- High Availability
- Availability on AIX®, Solaris, and HP-UX in addition to Linux and Windows / Disponibilità su AIX®, su Solaris e sul HP-UX oltre a Linux e Windows

Chargeable DB2 Workgroup Edition

Caratteristiche a pagamento nella DB2

Features:

Workgroup Edition:

- pureXML
- Query Optimization Feature (MQT, MDC, query parallelism)
- Homogenous Federation Feature

Features included at no-charge in DB2 Enterprise Edition:

Caratteristiche incluse gratuitamente nella DB2 Enterprise Edition:

- Table (Range) Partitioning
- Materialized Query Tables (MQT)
- Multi-dimensional Clustering (MDC)
- High Availability and Disaster Recovery (HADR), Tivoli® System Automation
- Connection Concentrator

Chargeable DB2 Enterprise Edition

Caratteristiche a pagamento nella DB2

Features:

Enterprise Edition:

- pureXML
- Storage Optimization Feature (includes compression / comprende la compressione)
- Advanced Access Control (fine grained and advanced security / sicurezza precisa e avanzata)
- Performance Optimization (Workload Management, Performance Expert, Query Patroller)
- Geodetic Data Management (geographical location analysis / analisi di posizione geografica)
- Homogenous Federation Feature (access remote DB2 and Informix® data as local tables / accesso a distanza dei dati di DB2 e di Informix® come tabelle locali)

Fee-based products related to DB2:

Prodotti a pagamento relativi a DB2:

- DB2 Connect
- DB2 Warehouse Editions
- WebSphere® Federation Server
- WebSphere Replication Server

2.1 Features included with DB2 Express-C subscription / Caratteristiche incluse con l'abbonamento di DB2 Express-C

This section outlines DB2 Fix packs, HADR and SQL replication.

Questa sezione descrive i Fix pack di DB2, HADR e della SQL replication.

2.1.1 Fix packs / Fix pack

A DB2 fix pack is a set of code fixes applied onto an installed DB2 product, in order to fix different issues reported after the product was released. With an installed subscription license, fix packs are free to download and install. They are typically available every three months.

Un DB2 fix pack è un insieme di correzioni di codice applicate su un prodotto DB2 installato, per correggere i diversi problemi riportati dopo che il rilascio del prodotto. Con una license subscription installata, i fix pack sono gratuiti per il download e l'installazione. Sono in genere disponibili ogni tre mesi.

To download the latest fix pack, review the DB2 technical support site at:

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

Per scaricare l'ultimo fix pack, visita il sito del supporto tecnico di DB2 su:

2.1.2 High Availability Disaster Recovery (HADR)

High Availability Disaster Recovery (HADR) is a database reliability feature that provides a high-availability and disaster recovery solution for complete as well as partial site failures. An HADR environment generally consists of two data servers, the primary and the secondary (which can be in geographically apart locations). The primary server is where the source database is stored and accessed by client applications. As transactions are processed on the primary database, database log records are automatically shipped to the secondary server across the network. The secondary server has a cloned copy of the primary database, usually created by backing up the primary database and restoring it on the secondary system. When the primary database logs are received they are replayed and applied to the secondary database. Through continuous replay of the log records, the secondary database

High Availability Disaster Recovery (HADR) è una feature di affidabilità del database che offre una soluzione di tipo High Availability e Disaster Recovery per fallimenti totali o parziali di siti. Un ambiente HADR consiste generalmente di due data server, il primario e il secondario (che possono essere in località geograficamente differenti). Il server primario è dove il database sorgente è memorizzato ed accessibile dalle applicazioni client. Mentre le transazioni sono processate sul database primario, i log dei record sono automaticamente spediti al server secondario attraverso la rete. Il server secondario ha una copia clonata del database primario, di solito creato dal backup del database primario e poi ristabilita sul sistema secondario. Quando i log del database primario vengono ricevuti sono ripetuti e applicati al database secondario. Con il replay continuo delle annotazioni dei log dei

keeps an in-sync replica of the primary database that can take over if the primary database fails.

A full DB2-supported HADR solution gives you:

- Lightning fast failover capability, with complete transparency for customers and client applications
- Full transaction atomicity to prevent data loss
- The ability to upgrade systems or applications without visible service interruption
- Remote system failover, providing full recovery from local disaster striking the data center
- Easy management with DB2 graphical tools
- All of this with negligible impact on overall system performance

record, il database secondario mantiene una replica sincronizzata del database primario che può subentrare nel caso in cui il database primario fallisse.

Una versione di DB2 con supporto completo a HADR offre:

- Capacità di failover rapidissima, con la completa trasparenza per i clienti e per le applicazioni client
- La piena atomicità di transazione per prevenire perdita di dati
- L'abilità di effettuare upgrade di sistemi o applicazioni senza interruzioni visibili di servizio
- Failover remoto del sistema, offre recupero completo dal disastro locale che attacca il centro dati
- Amministrazione facile con i tool grafici di DB2
- Tutto questo con impatto trascurabile sulle prestazioni generali del sistema

Note:

To view a demonstration of how HADR works, please visit this site:
<http://www-306.ibm.com/software/data/db2/express/demo.html>

Nota:

Per osservare una dimostrazione di come l'HADR funzioni, visita questo sito:
<http://www-306.ibm.com/software/data/db2/express/demo.html>

2.1.3 Data Replication / Replicazione dei Dati

This feature allows for replication of data between a source server where data changes are captured, and a target server where data changes are applied. Figure 2.1 provides an overview of how replication works.

Questa caratteristica permette la replica dei dati fra un server source in cui i cambiamenti di dati sono catturati e un target server in cui i cambiamenti di dati sono applicati. Figura 2.1 offre una veduta d'insieme di come la replica funzioni.

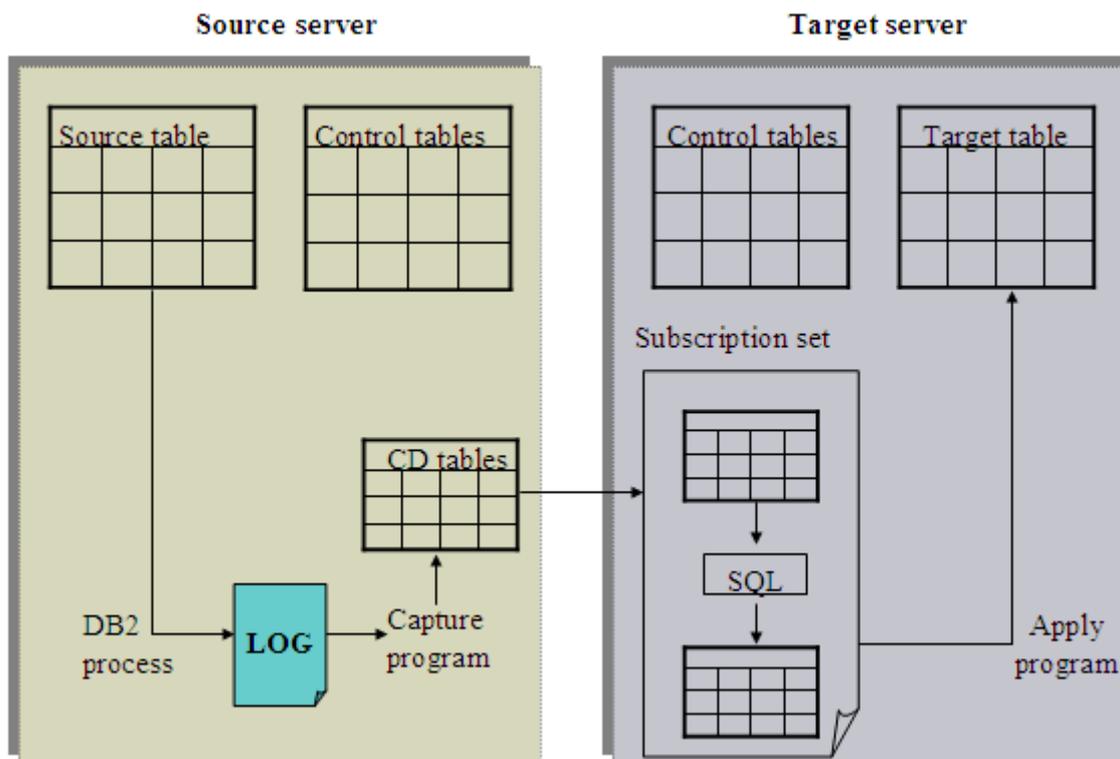


Figure 2.1 –SQL Replication / Figura 2.1 –Replicazione SQL

In Figure 2.1 there are two servers, a source server and a target server. On the source server, a Capture program captures the changes made to the database. On the target server, an Apply program applies the changes to the database replica. Replication is useful for a variety of purposes that require replicated data, including capacity relief, feeding data warehouses and data marts, and auditing change history. Using the SQL replication feature you can replicate data between DB2 Express-C and other DB2 servers, including those on other Linux, UNIX, z/OS, and i5/OS systems.

In figura 2.1 ci sono due server, un source server e un target server. Sul server di origine, un programma Capture cattura le modifiche fatte al database. Sul server target, un programma Apply applica le modifiche fatte alla replica del database. La replica è utile per vari scopi che richiedono i dati replicati, compreso la capacità di soccorso, alimentando data warehouse e data mart e il controllo dell'history dei cambiamenti. Usando la caratteristica SQL replication si può replicare i dati fra DB2 Express-C ed altri server di DB2, compresi quelli su altri sistemi Linux, UNIX, z/OS e i5/OS.

2.2 Features not available with DB2 Express-C / Caratteristiche non disponibili con DB2 Express-C

This section describes some of the features available in other editions of DB2 but not in DB2 Express-C.

Questa sezione descrive alcune delle caratteristiche disponibili in altre edizioni di DB2 ma non in DB2 Express-C.

2.2.1 Database Partitioning

The database partitioning feature (DPF) is only available with DB2 Warehouse Editions. It allows databases to be spread across multiple partitions which can reside in several computers. DPF is based on a shared-nothing architecture. Each computer, as it is added to the partition group, brings additional data processing power with its own CPUs and memory. DPF is particularly useful in large data server environments like data warehouses where decision support systems (DSS) queries are run.

La database partitioning feature (DPF) è disponibile soltanto con DB2 Warehouse Editions. Permette che i database siano sparsi attraverso partizioni multiple che possono risiedere su parecchi computer. DPF è basato su un'architettura non-condivisa. Ogni computer, non appena viene aggiunto al partition group, porta ulteriore potenza nel processare i dati con la propria CPU e memoria. DPF è particolarmente utile nei grandi data server environment come le data warehouse in cui vengono eseguite decision support systems (DSS) query.

2.2.2 Connection Concentrator

Connection concentrator is a feature that allows for support of a large number of concurrently connected users. Previously, every database connection required one database agent. The connection concentrator introduces the concept of a "logical agent", allowing one agent to handle several connections. Agents are discussed in more detail in Chapter 6, DB2 Architecture.

Il connection concentrator è una caratteristica che permette il supporto di tantissimi utenti collegati simultaneamente. Prima, ogni collegamento al database richiedeva un agente di database. Il connection concentrator introduce il concetto di "agente logico", permettendo che un agente tratti parecchie connessioni. Gli agenti sono discussi con maggiore dettaglio nel capitolo 6, L'architettura DB2.

2.2.3 Geodetic Extender

DB2 Geodetic Extender is available as priced option for DB2 Enterprise Edition. This extender makes development for business intelligence and e-government applications that require geographical location analysis much easier. DB2 Geodetic Extender can construct a virtual globe at any scale. Most location information is collected using worldwide systems, such as global positioning satellites (GPS), and can be represented in latitude/longitude coordinates (geocode). Business data, such as addresses, can be converted to a geocode by DB2 Geodetic Extender and enterprise applications work better when they keep the data in this unprojected form, leaving map projections (earth to flat map) where they belong: in the presentation layer, to display and print maps.

Il DB2 Geodetic Extender è disponibile come un'opzione a pagamento con il DB2 Enterprise Edition. Questo extender rende molto più facile lo sviluppo di business intelligence e delle applicazioni di e-government che richiedono l'analisi della posizione geografica. Il DB2 Geodetic Extender può costruire un globo virtuale in qualsiasi scala. La maggior parte delle informazioni di localizzazione sono raccolte usando i sistemi mondiali, come i satelliti di posizionamento globali (GPS), e possono essere rappresentate nelle coordinate di longitudine/latitudine (di geocode). I dati business, come gli indirizzi, possono essere convertiti in un geocode dal DB2 Geodetic Extender e le applicazioni di enterprise funzionano meglio quando mantengono i dati in questa forma unprojected, lasciando le proiezioni di mappa (terra in mappe piane) dove compete loro: nel livello di presentazione, per visualizzare e stampare le mappe.

2.2.4 Workload Management (WLM)

Manages workloads across a database based on user and application priorities combined with resource availability and workload thresholds. It allows you to regulate your database workload and queries so that important and high-priority queries can run promptly, and prevent 'rogue' queries from monopolizing your system resources, ensuring that your system runs efficiently. WLM is new in DB2 9.5 and provides more powerful capabilities than Query Patroller and DB2 Governor tools available with previous versions of DB2.

Si occupa del workload attraverso un database basato sulle priorità di utenti e applicazioni combinate con la disponibilità delle risorse e le soglie di lavoro. Vi permette di regolare la tua quota di workload di database e query così le query importanti e ad alta priorità possono essere eseguite prontamente, e previene il monopolio delle query "rogue" delle tue risorse di sistema, accertandosi che il tuo sistema funzioni efficientemente. WLM è nuovo in DB2 9.5 e offre capacità più potenti rispetto ai tool Query Patroller e DB2 Governor disponibili con le versioni precedenti di DB2.

2.3 Fee-based products that are related to DB2 / Prodotti a pagamento collegati con DB2

2.3.1 DB2 Connect

DB2 Connect is fee-based software that allows a DB2 for Linux, UNIX or Windows client to connect to a DB2 for z/OS or DB2 for i5/OS server as shown in Figure 2.2. DB2 Connect is not required when the connection occurs in the opposite direction; when you connect from DB2 for z/OS or DB2 for i5/OS to a DB2 for Linux, UNIX or Windows server. DB2 Connect comes in two main editions depending on your connection needs: A DB2 Connect Personal Edition, and a DB2 Connect Enterprise Edition.

DB2 Connect è un software a pagamento che permette a un client DB2 per Linux, UNIX o Windows di collegarsi ad un server DB2 per z/OS o per i5/OS come è illustrato in figura 2.2. DB2 Connect non è necessario quando il collegamento avviene nella direzione opposta; quando vi collegate da DB2 per z/OS o da DB2 per i5/OS ad un server di DB2 per Linux, UNIX o Windows. DB2 Connect è disponibile in due edizioni principali secondo le tue necessità di collegamento: DB2 Connect Personal Edition e DB2 Connect Enterprise Edition.

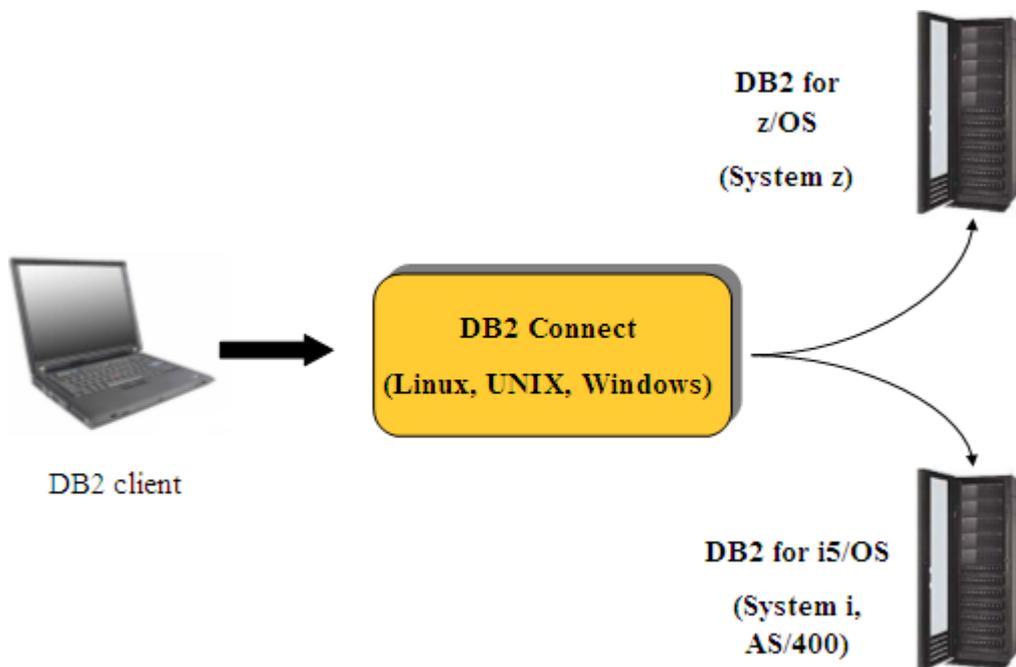


Figure 2.2 – DB2 Connect

2.3.2 WebSphere Federation Server

Formerly known as WebSphere Information Integrator (for federation support), the WebSphere Federation Server allows for federation of databases, meaning that you can run database queries that can work with objects from different relational database systems. For example, if you buy WebSphere Federation Server, you can run the following query:

```
SELECT *
FROM   Oracle.Table1  A
       DB2.Table2     B
       SQLServer.Table3 C
WHERE  A.col1 < 100
       and B.col5 = 1000
       and C.col2 = 'Test'
```

Figure 2.3 provides an illustration where WebSphere Federation Server is used.

For relational database management systems that are part of the IBM family, federation support is built into DB2 Express-C. This means that the WebSphere Federation Server product is not required when, for example, you want to run a query between two different DB2 databases, or between one DB2 database and an Informix database (Informix is part of the IBM family).

Precedentemente conosciuto come WebSphere Information Integrator (per il supporto alla federazione), il WebSphere Federation Server permette la federazione dei database, che significa che potete eseguire le query che possono funzionare con diversi sistemi di database relazionale. Per esempio, se comprate WebSphere Federation Server, potete eseguire la seguente query:

La figura 2.3 offre un'illustrazione in cui viene usato il WebSphere Federation Server.

Per la gestione del database relazionale che fa parte della famiglia di IBM, il supporto di federazione è costruito all'interno di DB2 Express-C. Ciò significa che il prodotto di WebSphere Federation Server non è necessario quando, per esempio, vuoi eseguire una query tra due diversi database DB2, o tra un database di DB2 e un database Informix (Informix è parte della famiglia IBM).

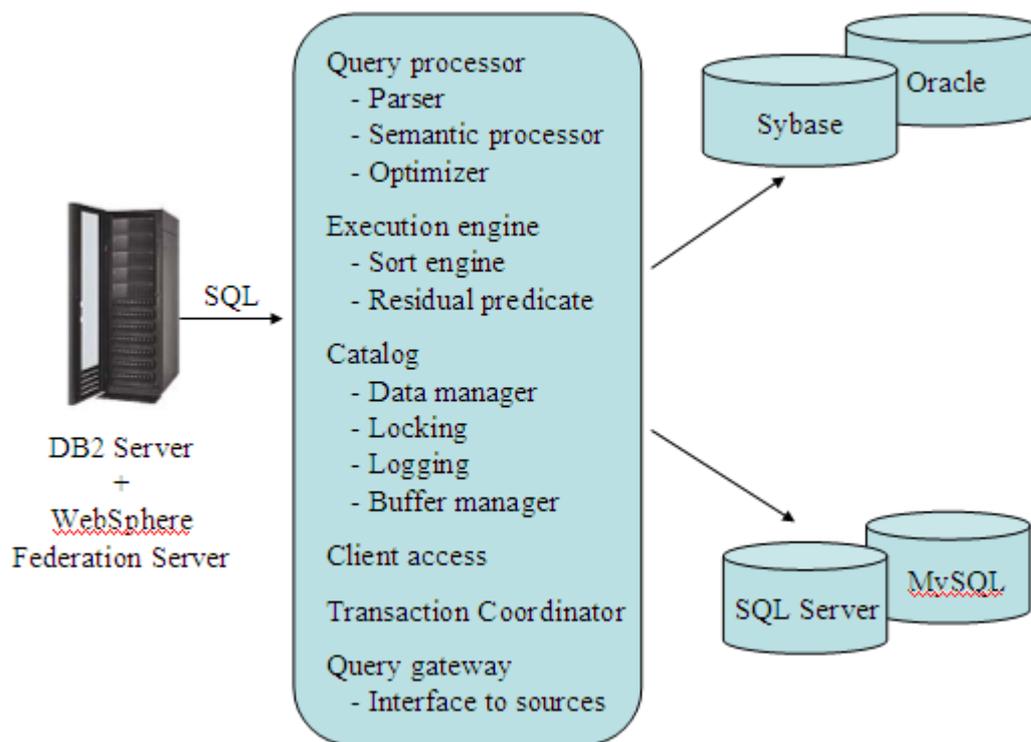


Figure 2.3 – WebSphere Federation Server

2.3.3 WebSphere Replication Server

Formerly known as WebSphere Information Integrator (for replication support), the WebSphere Replication Server allows for SQL replication of database records when non-IBM data servers are involved. It also includes a feature known as Q-Replication for replicating data using message queues.

Precedentemente conosciuto come WebSphere Information Integrator (per il supporto alla replica), il WebSphere Replication Server permette la replica SQL dei record di database quando i data server non-IBM sono coinvolti. Include anche una feature conosciuta come Q-Replication per la replica dei dati usando code di messaggi.

3

Chapter 3 – DB2 installation / Capitolo 3 – Installazione di DB2

To install the DB2 Express-C edition in either Linux or Windows, ensure that your systems satisfy the installation prerequisites.

Per installare l'edizione di DB2 Express-C sia in Linux che in Windows, assicurati che i tuoi sistemi soddisfino i prerequisiti di installazione.

3.1 Installation prerequisites / Prerequisiti di installazione

With respect to operating system version and level requirements, DB2 Express-C is available on Linux, Solaris (x64), and Windows 2000, 2003, XP, and Vista. The processor architectures available are 32-bit, 64-bit and PowerPC (Linux). If you need to run DB2 on another platform (such as UNIX), you should purchase one of the different data server editions described earlier in this book. Operating system requirements for all DB2 editions are also described in this document:

<http://www.ibm.com/software/data/db2/udb/sysreqs.html>

In terms of hardware resources, DB2 Express-C can be installed on systems with any number of CPU cores and memory, however, it will only utilize up to 2 cores and 2GB of memory for the free unwarranted license version, and up to 4 cores and 4 GB of memory for the paid subscription version. . The systems can be physical systems, or virtual systems created by partitioning or running virtual machine software. You can of course run on smaller systems if you prefer, for example a single processor system with 1GB of memory.

Riguardo ai requisiti di versione e livello del sistema operativo, DB2 Express-C è disponibile su Linux, Solaris (x64) e Windows 2000, 2003, XP e Vista. Le architetture dei microprocessori supportate sono a 32-bit, 64-bit e PowerPC (Linux). Se hai bisogno di eseguire DB2 su un'altra piattaforma (come UNIX), dovresti comprare una delle varie edizioni dei data server descritte precedentemente in questo libro. I requisiti del sistema operativo di tutte le edizioni di DB2 sono descritti anche in questo documento:

In termini di risorse di hardware, DB2 Express-C può essere installato su sistemi con un numero qualsiasi di core all'interno della CPU ed un qualunque quantitativo di memoria, tuttavia, utilizzerà soltanto fino a 2 core e 2GB di memoria per la versione gratuita con licenza unwarranted, e fino a 4 core e 4GB di memoria per la versione a pagamento. I sistemi possono essere sistemi fisici, o sistemi virtuali creati dal partitioning o dall'esecuzione di software dal computer virtuale. Può ovviamente essere eseguito anche su sistemi più piccoli, per esempio un

sistema a microprocessore singolo con 1GB di memoria.

For the latest information on DB2 Express-C hardware requirements, review the DB2 Express-C web page:

Per le ultime informazioni sui requisiti hardware di DB2 Express-C, esamina la pagina web di DB2 Express-C:

<http://www-306.ibm.com/software/data/db2/express/getstarted.html>

3.2 Operating system installation authority / L'autorizzazione di installazione del sistema operativo

To install DB2 Express-C on Linux or Windows, you must use an operating system user with sufficient authority.

Per installare DB2 Express-C su Linux o su Windows, devi usare un utente del sistema operativo con autorizzazione sufficiente.

For **Linux**, you need to be root (the superuser) to install DB2 Express-C. You can also install DB2 Express-C as a non-root user; however, you will be limited in what you can do with the product. For example, under a non-root installation, you cannot create more instances than the default one created at installation time.

Per **Linux**, devi essere root (super-user) per installare DB2 Express-C. Puoi anche installare DB2 Express-C come utente non-root; tuttavia, sarai limitato in quello che puoi fare con il prodotto. Per esempio, con un'installazione non-root, non puoi creare altre instance salvo quella di default creata al momento dell'installazione.

For **Windows**, the user account must belong to the Administrators group on the machine where you will perform the installation. Alternatively, a non-Administrator user account can be used, provided that a member of the Windows Administrators group first configures the Windows elevation privileges settings to allow a non-Administrator user account to perform an installation.

Per **Windows**, l'account dell'utente deve appartenere al gruppo degli Amministratori del computer su cui esegui l'installazione. Alternativamente, può essere usato un account di un utente non-Amministratore, a condizione che un membro del gruppo Windows Administrators prima configuri i setting dei privilegi di elevation di Windows per permettere a un account di un utente non-Amministratore di eseguire un'installazione.

For Windows domain accounts, to verify user IDs on the DB2 server, the installation user ID must belong to the Domain Administrators group on the domain where the accounts are going to be created. You may also use the built-

Per i domain account di Windows, al fine di verificare l'user ID sul server DB2, l'user ID dell'installazione deve appartenere al gruppo Domain Administrators sul dominio dove gli account saranno creati. Puoi anche

in Local System account to run the installation for all products.

usare l'account built-in del Local System per eseguire l'installazione per tutti i prodotti.

The user account must also have the user right to "Access this computer from the network".

L'user account deve avere anche il diritto di "Accedere questo computer dalla rete".

Note:

See a video presentation about DB2 Express-C installation at this link:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4442>

Nota:

Vedi una presentazione dell'installazione di DB2 Express-C a questo link:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4442>

3.3 Installation wizard / Wizard di installazione

Although there are several methods to install DB2 Express-C, the easiest method is to use the GUI-based DB2 Installation wizard. After downloading and unzipping the DB2 Express-C image, you can launch the wizard to handle the installation:

- Windows: execute the `setup.exe` file in the `EXP/image` directory
- Linux: execute the `db2setup` command in the `exp/disk1` directory

DB2 Express-C is very easy to install by following the instructions of the DB2 installation wizard. In most cases, the default settings are sufficient, so all you need to do is accept the license, click the *Next* button several times, and click the *Finish* button. After a few minutes, your installation will be complete and DB2 is up and running!

Benché ci siano diversi metodi per installare DB2 Express-C, il metodo più facile è usare il GUI-based wizard di installazione di DB2. Dopo aver scaricato e decompresso l'immagine di DB2 Express-C, puoi lanciare il wizard per gestire l'installazione:

- Windows: eseguire il file `setup.exe` nella directory `EXP/image`
- Linux: eseguire il comando `db2setup` nella directory `exp/disk1`

DB2 Express-C è molto facile da installare seguendo le istruzioni del wizard di installazione di DB2. Nella maggior parte dei casi, i setting di default sono sufficienti, devi soltanto accettare la licenza, cliccare il tasto *Next* alcune volte, e poi cliccare su *Finish*. Dopo alcuni minuti, la vostra installazione sarà completa e DB2 pronto e funzionante!

Figure 3.1 shows the DB2 Setup Launchpad. Click on *Install a Product* and then choose *Install New* to install a new copy of DB2 Express-C in your system.

La figura 3.1 mostra il DB2 Setup Launchpad. Clicca *Install a Product* e poi scegli *Install New* per installare una nuova copia di DB2 Express-C nel vostro sistema.

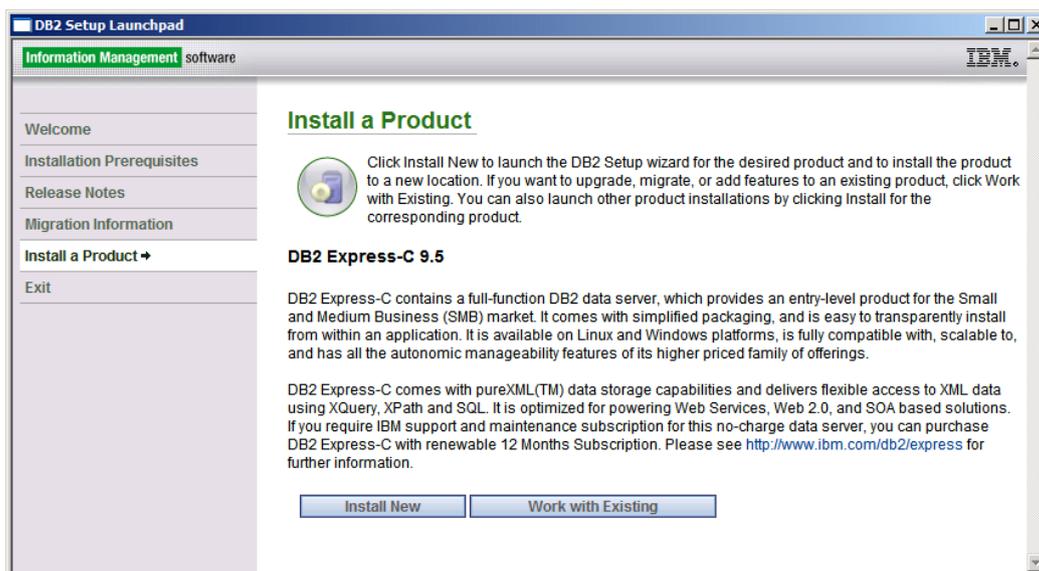


Figure 3.1 – The DB2 Setup Launchpad / Figura 3.1 - Il DB2 Setup Launchpad

After accepting the license, it is usually sufficient to choose the “Typical” installation (default) as shown in Figure 3.2.

Dopo aver accettato la licenza, è solitamente sufficiente scegliere l'installazione "Tipica" (default) come mostrato nella figura 3.2.

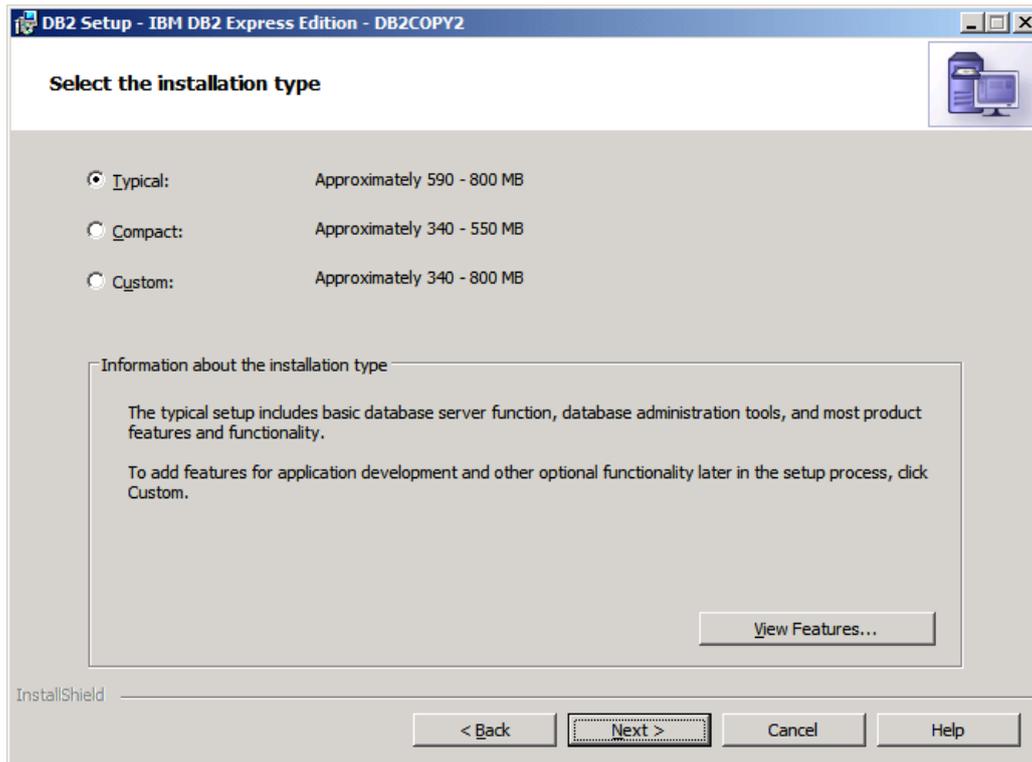


Figure 3.2 – Installation types / Figura 3.2 - Tipi di installazione

In Figure 3.3, you have the choice to install the product, create a response file, or both. Response files are discussed in section 3.4, Silent Install. Choosing the default (Install IBM DB2 Express Edition on this computer and save my settings in a response file) is good enough.

Nella figura 3.3, hai la scelta di installare il prodotto, creare un file di risposta, o entrambe. I file di risposta sono discussi nella sezione 3.4, Silent Install. La scelta di default (installare l'edizione di DB2 Express-C su questo computer e salva i miei setting in un file di risposta) è sufficiente.

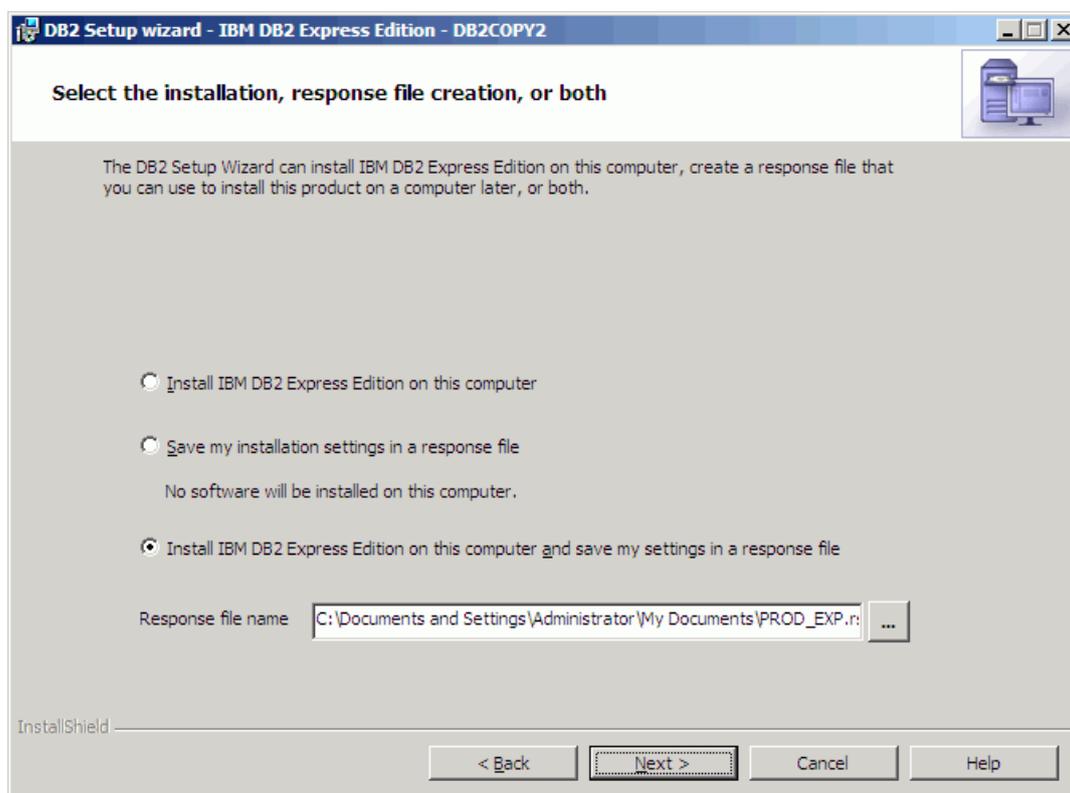


Figure 3.3 – Selecting the installation / Figura 3.3 – Selezionare l'installazione

Choose default values for the next few screens. When you get to the window shown in Figure 3.4, you can input an existing user who will be used to work with the instance and other services. This user must be part of the Local Administrator group in Windows. If the user ID you input does not belong to an existing user, it will be created as a Local Administrator. You can leave the domain field blank if the user ID does not belong to a domain. The default user ID to create in Windows is called db2admin. In the case of Linux, the default user ID created is called db2inst1.

Scegli i valori di default per le prossime schermate. Quando arrivi alla finestra che appare in figura 3.4, puoi immettere un user esistente che sarà usato per lavorare con l'instance e con gli altri servizi. Questo user ID deve far parte del gruppo Local Administrator in Windows. Se l'user ID che immetti non appartiene ad un utente esistente, sarà creato come Local Administrator. Puoi lasciare lo spazio del campo domain in bianco se l'user ID non appartiene ad un domain. L'user ID di default da creare in Windows è chiamato db2admin. Nel caso di Linux, l'user ID creato di default è chiamato db2inst1.

Figure 3.4 – Specifying user information for the default DB2 instance / Figura 3.4 - Specificare le informazioni dell'utente per la default instance di DB2

Finally, in Figure 3.5, the installation wizard displays a summary of what will be installed, and the different configuration information provided in the previous steps. When you click *Finish*, installation will start, and the program files will be laid down on your system.

Per concludere, nella figura 3.5, il wizard di installazione riporta un riassunto di che cosa sarà installato e diverse informazioni di configurazione fornite nei punti precedenti. Quando clicchi *Finish*, l'installazione comincerà e i file del programma saranno salvati sul tua sistema.

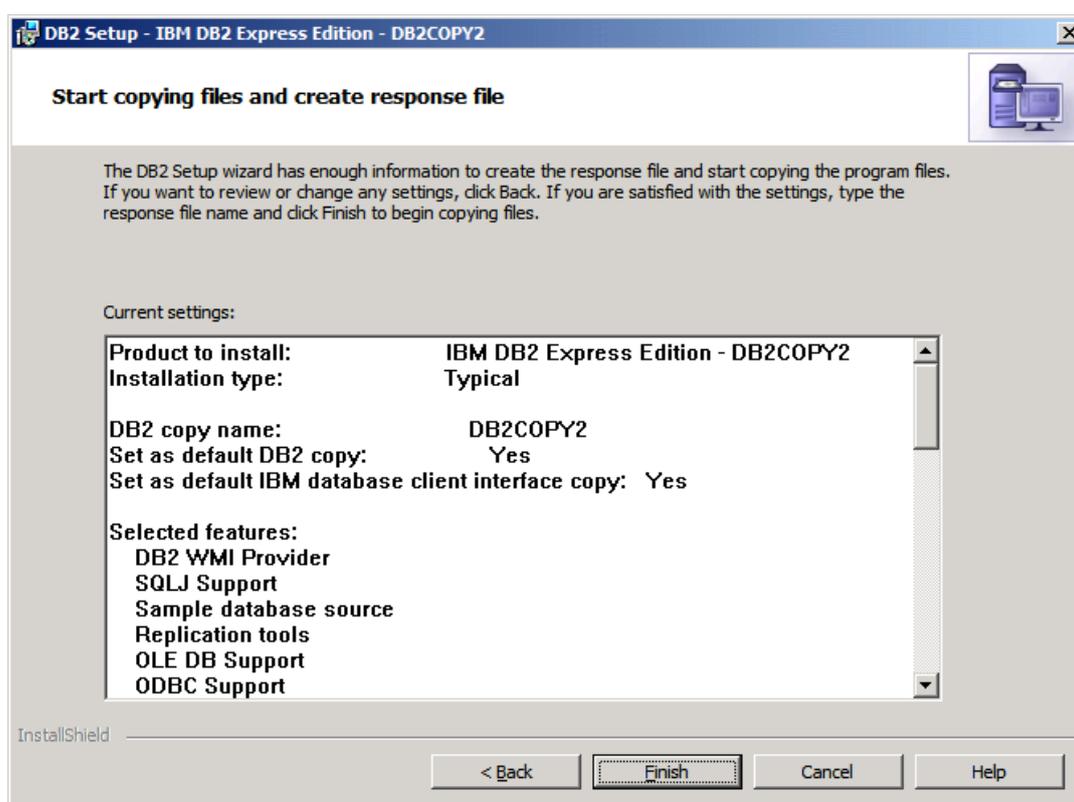


Figure 3.5 – Summary of what will be installed / Figura 3.5 – Riassunto di che cosa sarà installato

3.4 Silent Install

There may be situations where you need to install a DB2 client on multiple computers; or you need to embed a DB2 data server as part of your application, and would like to install it as part of your application installation process. In these situations, a silent

Ci possono essere situazioni in cui hai bisogno di installare un client DB2 su diversi computer; o hai bisogno di inserire un data server DB2 come componente della tua applicazione e vorresti installarlo come parte del processo di installazione della tua

install is the ideal way to install DB2.

applicazione. In queste situazioni, un silent install è il modo migliore di installare DB2.

DB2 enables silent installs through the use of text response files which store installation information. The following shows a snippet of a sample response file.

DB2 permette le silent install attraverso l'uso dei file di risposta di testo che memorizzano le informazioni dell'installazione. La seguente figura mostra un esempio di un frammento di un file di risposta.

Sample response file code snippet:

Esempio di un frammento di codice del file di risposta:

```
PROD=UDB_EXPRESS_EDITION
LIC_AGREEMENT=ACCEPT
FILE=C:\Program Files\IBM\SQLLIB\
INSTALL_TYPE=TYPICAL

LANG=EN

INSTANCE=DB2
DB2.NAME=DB2
DEFAULT_INSTANCE=DB2
DB2.SVCENAME=db2c_DB2
DB2.DB2COMM=TCPIP
...
```

There are a number of ways to generate a response file:

Ci sono vari modi di generare un file di risposta:

- Install DB2 Express-C once on a computer using the DB2 Installation wizard. One of the first wizard options allows you to select the checkbox to save your install responses to a response file. At the end of the wizard, a response file will be generated in a directory and filename that you specify. This is a text file, so you can edit it
- Installare la prima volta DB2 Express-C su un computer usando il wizard di installazione di DB2. Una delle prime opzioni del wizard permette di selezionare la casella per conservare le tue scelte di installazione in un file di risposta. Alla fine del wizard, un file di risposta sarà generato in una directory con un nome che

afterwards. This was shown in Figure 3.3.

- Edit the sample response file packaged with the DB2 image. This sample file (denoted with a .rsp file extension) is located in the *db2/platform/samples/* directory
- For Windows, you can also use the response file generator command:

```
db2rspgn -d <output directory>
```

Then to install DB2 using a response file, for Windows you issue the command:

```
setup -u <response filename>
```

For Linux you issue the command:

```
db2setup -r <response filename>
```

specificherai. Questo è un file di testo, così potrai modificarlo in seguito, come mostrato nella figura 3.3.

- Edita il file di risposta d'esempio contenuto nell'immagine di DB2. Questo file d'esempio (caratterizzato con un'estensione file .rsp) è situato nella directory *db2/platform/samples/*
- Per Windows, puoi usare anche il comando di generazione del file di risposta:

Quindi per installare DB2 usando un file di risposta, in Windows esegui il comando:

In Linux esegui il comando:

QuickLab #1: Install DB2 Express-C and create SAMPLE database / Installare DB2 Express-C e creare il database SAMPLE

Objective

Before you can begin exploring all the features and tools that come with DB2 Express-C, you must first install it on your system. In this Quicklab, you will perform a basic installation of DB2 Express-C on Windows. The same installation wizard is available on Linux; therefore the steps are very similar on that platform.

Procedure

- 1. Obtain DB2 Express-C images.** Download the appropriate DB2 Express-C image, or order the Discovery Kit DVD with the images from the DB2 Express-C Web site (ibm.com/db2/express). Unzip the files into any directory you wish.
- 2. Locate files.** Navigate to the directory (or drive) containing the unzipped DB2 product installation files.
- 3. Run Launchpad.** Launch the DB2 Launchpad by double-clicking on the `setup.exe` file. On Linux, run the `db2setup` command as root. From the Launchpad, click the *Install Product* option on the left pane of the window.
- 4. Run DB2 setup wizard.** The DB2 setup wizard checks that

Obiettivo

Prima che tu possa cominciare ad esplorare tutti le feature e i tool contenuti in DB2 Express-C, devi installarlo sul tuo sistema. In questo Quicklab, eseguirai l'installazione base di DB2 Express-C su Windows. Lo stesso wizard di installazione è disponibile su Linux; i passi da seguire sono molto simili su quella piattaforma.

Procedura

- 1. Ottieni l'immagine di DB2 Express-C.** Scarica l'immagine di DB2 Express-C appropriata, o ordina il Discovery Kit DVD con le immagini dal sito web di DB2 Express-C (ibm.com/db2/express). Decomprimi i file in qualsiasi directory desideri.
- 2. Trova i file.** Naviga fino alla directory (o al disco) che contiene i file d'installazione del prodotto DB2 in formato compresso.
- 3. Esegui il Launchpad.** Lancia il Launchpad DB2 cliccando due volte sul file `setup.exe`. Su Linux, esegui il comando `db2setup` come root. Dal Launchpad, clicca sull'opzione *Install Product* sul pannello a sinistra della finestra.
- 4. Esegui il DB2 setup wizard.** Il DB2 setup wizard controlla che

all system requirements are met and sees if there are any existing DB2 installations. Click the *Next* button to continue with the installation.

5. Review license agreement.

Read and accept the license agreement (select the "*I Accept...*" radio button) and click the *Next* button to continue.

- 6. Choose installation type.** For this exercise, select the *Typical* option (this is the default). The *Compact* option performs a basic installation, while the *Custom* option allows you to customize the specific features you want to install. Click the *Next* button to continue.

7. Select installation folder.

This screen allows you to customize the drive and directory where the DB2 code is installed on your system. Ensure sufficient space exists for the installation. Use the default drive and directory settings for this example (shown below):

Drive: C:

Directory: C:\Program Files\IBM\SQLLIB

Click the *Next* button to continue.

- 8. Set user information.** Once DB2 Express-C is installed, certain DB2 processes are run

tutti i requisiti del sistema siano soddisfatti e se ci sono delle installazioni di DB2 esistenti. Clicca il tasto *Next* per continuare con l'installazione.

5. Esamina il contratto di licenza.

Leggi e accetta il contratto di licenza (seleziona il tasto "*I Accept ...*") e clicca *Next* per continuare.

6. Scegli il tipo dell'installazione.

Per questa esercitazione, seleziona l'opzione *Typical* (questo è il default). L'opzione *Compact* compie un'installazione di base, mentre l'opzione *Custom* ti permette di personalizzare specifiche feature che vuoi installare. Clicca *Next* per continuare.

7. Seleziona cartella di installazione.

Questa schermata ti permette di scegliere il disco e la directory dove DB2 verrà installato sul sistema. Assicurati che esista spazio sufficiente per l'installazione. Usa le impostazioni di default per disco e directory per questo esempio (come il seguente):

Clicca *Next* per continuare.

- 8. Configura le informazioni dell'utente.** Una volta che DB2 Express-C è installato, alcuni

as system services. These services require an operating system account in order to run. In the Windows environment, using the default **db2admin** user account is recommended. If the user account does not yet exist, DB2 creates it in the operating system for you. You can also specify to use an existing account, but that account must have local administrator authority. We recommend using the defaults suggested. Ensure you specify a password for the account. On Linux use the default **db2inst1** user ID for the instance owner, **db2fenc1** for the fenced user and **dasusr1** for the DB2 Administration server user. Click the *Next* button to continue.

- 9. Configure DB2 instance.** A DB2 instance can be thought of as a container for databases. An instance must exist before a database can be created inside it. During a Windows installation, an instance called *DB2* is automatically be created. In a Linux environment, the default instance name is *db2inst1*. We will cover instances later in this book.

By default, the *DB2* instance is configured to listen for TCP/IP connections on port 50000. Both the default protocol and

processi di DB2 vengono eseguiti come servizi di sistema. Questi servizi per funzionare richiedono un account sul sistema operativo. Nell'ambiente Windows, si raccomanda di usare l'user account di default **db2admin**. Se questo ancora non esiste, DB2 lo crea per te. Puoi anche specificare di utilizzare un account esistente, ma quell'account deve avere l'autorizzazione di amministratore locale. Raccomandiamo di usare la configurazione di default suggerita. Assicurati di specificare una password per l'account. Su Linux usa l'user ID di default **db2inst1** per il proprietario dell'instance, **db2fenc1** per l'user fenced e **dasusr1** per l'user amministratore del server. Clicca *Next* per continuare.

- 9. Configura l'instance di DB2.** Un'instance di DB2 può essere pensata come un contenitore per i database. Un instance deve esistere prima che un database possa essere creato all'interno di esso. Durante l'installazione di Windows, un instance chiamata *DB2* è creata automaticamente. In un ambiente Linux, il nome default dell'instance è *db2inst1*. Affronteremo l'argomento delle instance più avanti in questo libro.

Di default, l'instance *DB2* è configurata per ascoltare i collegamenti di TCP/IP sulla port 50000. Sia il protocollo che la

the port can be changed by clicking the *Protocols* and *Startup* buttons, respectively. We recommend using the default settings in this example. Click the *Next* button to continue.

- 10. Start installation.** Review the installation options previously selected. Click the *Install* button to begin copying the files to the installation location. DB2 will also perform some initial configuration processes.

- 11. First Steps.** After the installation is complete, another launch utility, called First Steps, is displayed. First Steps can also be started later with the command `db2fs`.

- 12.** The SAMPLE database is a database that you can use for test purposes. It is created automatically right after the installation of DB2. Verify the database exists by opening the DB2 Control Center tool. To open this tool, from the Windows Start Menu choose: *Start -> Programs -> IBM DB2 -> DB2COPY1 (Default) -> General Administration Tools -> Control Center*

You can also start the Control Center with the command `db2cc`.

porta di default possono essere cambiati cliccando rispettivamente sui tasti *Protocols* e *Startup*. Raccomandiamo di usare i default setting in questo esempio. Clicca *Next* per continuare.

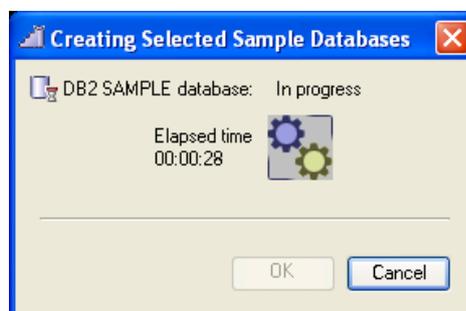
- 10. Iniziare l'installazione.** Controlla le opzioni dell'installazione precedentemente selezionate. Clicca sul tasto *Install* per cominciare a copiare i file sulla destinazione dell'installazione. DB2 eseguirà anche alcuni processi iniziali di configurazione.

- 11. Primi Passi.** Finita l'installazione, un altro programma di launch utility, chiamato First Steps, verrà visualizzato. First Steps può essere lanciato anche in seguito con il comando `db2fs`.

- 12.** Il database SAMPLE è un database che puoi usare per effettuare dei test. È generato automaticamente appena finita l'installazione di DB2. Verifica che il database esista aprendo il DB2 Control Center. Per aprire questo tool, dal Menu Start di Windows scegli: *Start -> Programs -> IBM DB2 -> DB2COPY1 (Default) -> General Administration Tools -> Control Center*

Puoi anche aprire il Control Center con il comando `db2cc`.

- 13.** If the SAMPLE database is displayed from the Control Center, you can jump to step 16. If it is not displayed, select the *Refresh* item from the Control Center *View* menu to ensure you are looking at the most up-to-date information. If the SAMPLE database still does not display, it may not have been created. You can manually create it from First Steps. Choose the tab that says “Database Creation”, and then follow the wizard to create the SAMPLE database. Ensure the *XML and SQL objects and data* option is chosen, and click *OK*. This last option will create a UNICODE database which was required in version 9 to support pureXML, but is not longer needed in DB2 9.5.
- 14.** The following progress screen is displayed while the database is being created. (This procedure may take several minutes). When database creation is complete, click the *OK* button and close the First Steps tool.
- 13.** Se il database SAMPLE è visualizzato dal Control Center, puoi passare al punto 16. Se non è visualizzato, seleziona *Refresh* dal menù View del Control Center per assicurarti di stare osservando le informazioni aggiornate. Se il database SAMPLE ancora non viene visualizzato, forse non è stato creato. Puoi crearlo manualmente dal tool First Steps. Scegli la scheda che dice "Database Creation" e poi segui il wizard per creare il database SAMPLE. Assicurati che l'opzione *XML and SQL objects and data* sia selezionata e clicca *OK*. Questa ultima opzione crea un database UNICODE che era necessario nella versione 9 per supportare il pureXML, ma non è più necessario in DB2 9.5.
- 14.** La seguente schermata indicante il progresso viene visualizzata durante la creazione del database. (Questa procedura può richiedere alcuni minuti). Quando la creazione del database è completa, clicca *OK* per chiudere il tool First Steps.



- 15.** Go back to Control Center and verify that a database called SAMPLE now appears in the Object Tree pane. You may have to refresh Control Center view to see the new changes. To do this, select the *Refresh* item from the Control Center *View* menu.
- 16. Restart the computer.** Although this step is not mentioned in the official DB2 installation documentation, we recommend rebooting the system (if possible, at least on Windows) to ensure all processes start successfully and to clean up any memory resources that might not have been cleaned up correctly. This is OPTIONAL.
- 15.** Ritorna indietro al Control Center e verifica che un database chiamato SAMPLE compaia nel pannello Object Tree. Può darsi che tu debba ricaricare la vista del Control Center per vedere i nuovi cambiamenti. Per fare questo, seleziona *Refresh* dal menu *View* dal Control Center.
- 16. Riavvia il computer.** Benché questo punto non sia menzionato nella documentazione ufficiale dell'installazione di DB2, raccomandiamo di riavviare il sistema (se possibile, almeno su Windows) per assicurare che tutti i processi comincino con successo e per pulire tutte le risorse di memoria che potrebbero non essere pulite correttamente. Ma questo è OPZIONALE.

4

Chapter 4 – DB2 Environment / Capitolo 4 – Ambiente DB2

In this chapter we will discuss the DB2 environment. Figure 4.1 provides a summary about DB2, and the red ellipse highlights the area of focus for this chapter. The left side of the figure shows the different DB2 commands, SQL, SQL/XML, and XQuery statements that can be created to interact with a DB2 data server. The middle of the figure shows the names of the different tools used to interact with a DB2 data server. The right side of the figure shows the basic DB2 environment consisting of an instance, a database, and the associated configuration files.

In questo capitolo discuteremo l'ambiente di DB2. La figura 4.1 fornisce una sintesi riguardo DB2 e l'ellisse rosso evidenzia la zona su cui ci si focalizzerà per questo capitolo. La parte di sinistra della figura mostra i diversi comandi di DB2, SQL, SQL/XML e gli statement XQuery che possono essere creati per interagire con un data server DB2. Al centro sono mostrati i nomi dei diversi tool usati per interagire con un data server DB2. La parte di destra invece mostra l'ambiente di base di DB2 che consiste in un instance, un database, e dei file di configurazione associati.

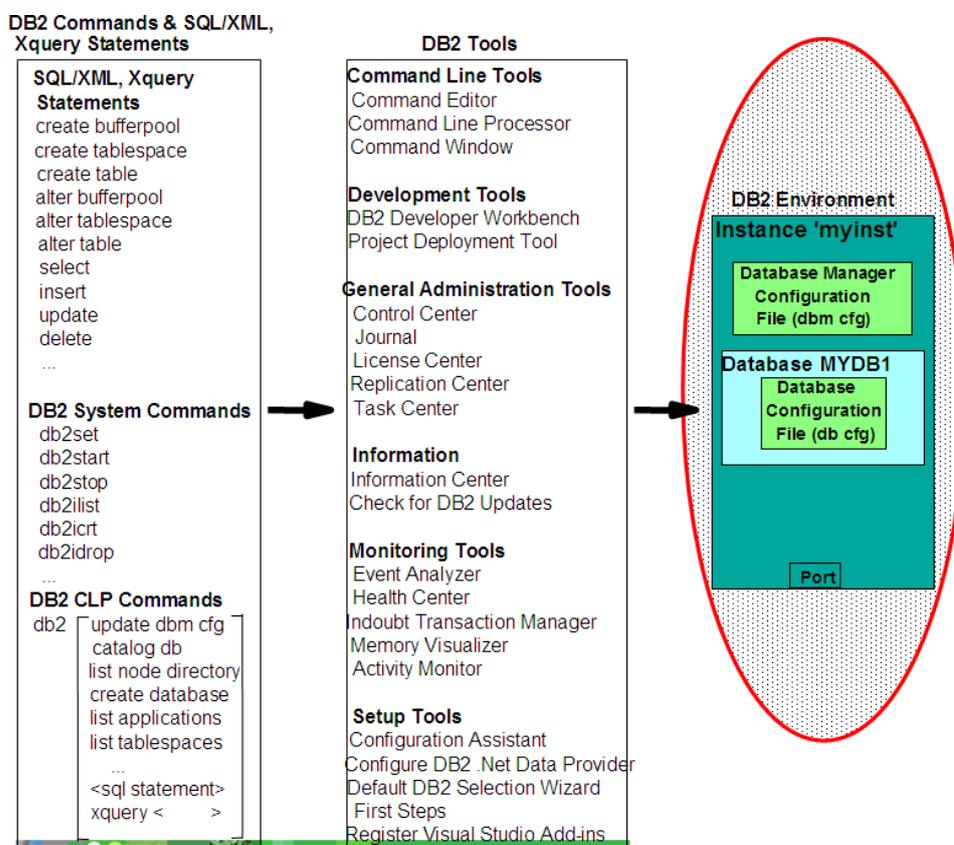


Figure 4.1 – The DB2 big picture: DB2 environment / Figura 4.1 DB2 big picture: Ambiente DB2

Note:
 See video presentations about the DB2 Environment at these links:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4029>
<http://www.channeldb2.com/video/video/show?id=807741:Video:4042>

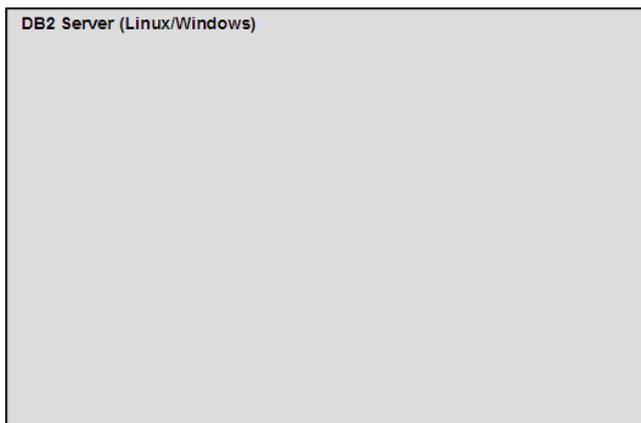
Nota:
 Vedi le presentazioni video sull'ambiente DB2 a questi link:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4029>
<http://www.channeldb2.com/video/video/show?id=807741:Video:4042>

To describe the DB2 environment, let's describe each component element step by step.

Per descrivere l'ambiente DB2, descriviamo ogni elemento che lo compone punto per punto.

Figure 4.2 shows a representation of a DB2 data server after installing DB2 Express-C 9.5.

La figura 4.2 mostra una rappresentazione di un data server DB2 dopo l'installazione di DB2 Express-C 9.5.



**Figure 4.2 – Representation of a DB2 Server after installing DB2 Express-C 9.5 /
Figura 4.2 – Rappresentazione di un server DB2 dopo l'installazione di DB2 Express-C 9.5**

As part of the installation in Windows, a default instance called "DB2" ("db2inst1" on Linux) is created. This is represented by the green box in Figure 4.3. An instance is simply an independent environment where applications can run, and databases can be created. You can create multiple instances on a data server, and use them for different purposes. For example, one instance can be used to hold databases for production use, another instance can be used for test environment databases, and another one for a development environment. All of these instances are independent; that is, performing operations on one instance will not affect the other instances.

Durante l'installazione in Windows, viene creata una default instance chiamata "DB2" ("db2inst1" su Linux). Questo è rappresentato dal box verde nella figura 4.3. Un instance è semplicemente un ambiente indipendente in cui le applicazioni possono essere eseguite ed i database possono essere creati. Puoi creare instance multiple su un data server e usarle per scopi diversi. Per esempio, un'instance può essere usata per tenere i database usati per la produzione, un'altra instance può essere usata per i database in un ambiente di prova ed un altro per un ambiente di sviluppo. Tutte queste instance sono indipendenti; cioè eseguire operazioni su un'instance non ha effetto sulle altre instance.

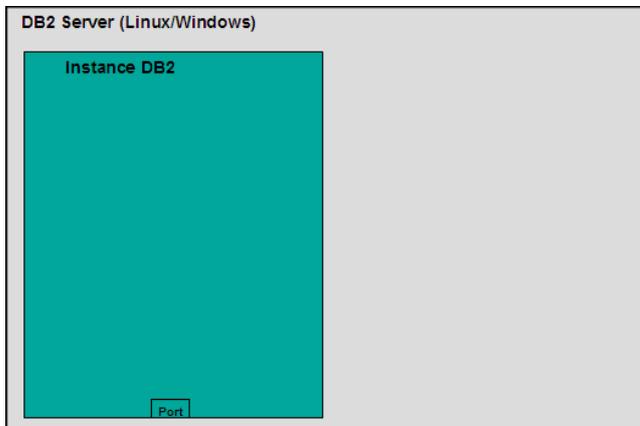


Figure 4.3 – The default DB2 instance created / Figura 4.3 – La default instance di DB2 creata

To create a new DB2 instance, use the command `db2icrt <instance name>`, where `<instance name>` is replaced with any 8 character name. For example, to create the instance *myinst*, we use this command:
`db2icrt myinst.`

Per creare una nuova instance DB2, usa il comando `db2icrt <nome instance>`, dove `<nome instance>` è sostituito con qualsiasi nome di 8 caratteri. Per esempio, per creare l'instance *myinst* usiamo questo comando:`db2icrt myinst.`

Figure 4.4 shows a new instance called `myinst` as a separate green box.

La figura 4.4 mostra una nuova instance chiamata `myinst` come un box verde separata.

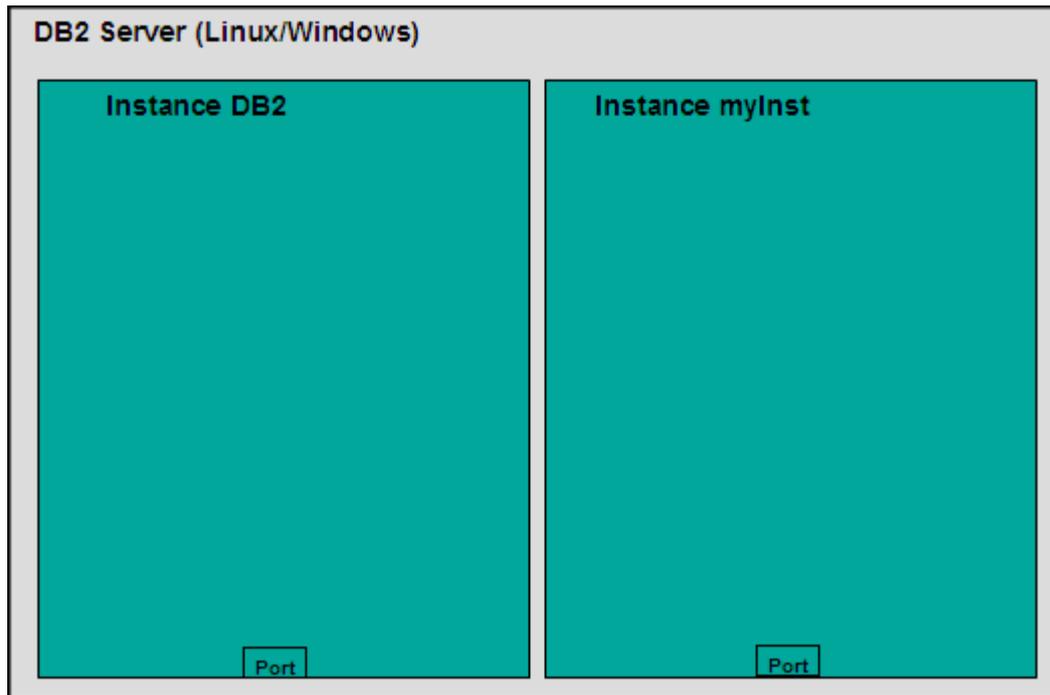


Figure 4.4 – A DB2 server with two instances / Figura 4.4 – Un server DB2 con due instance

Note that each instance has a unique port number. This helps to distinguish between instances when you want to connect to a database in a given instance from a remote client. If you use the DB2 Command Window, you can make any DB2 instance the active one by using this operating system command on Windows:

```
set db2instance=myinst
```

In this example, if you now create a database from the Command Window, it would be created in the instance `myinst`.

Da notare che ogni instance ha un numero di porta univoco. Ciò contribuisce a distinguere le instance quando vuoi collegarti ad un database da un client remoto. Se usi la DB2 Command Window, puoi rendere attiva qualsiasi instance usando questo comando del sistema operativo Windows:

In questo esempio, se creassi un database usando la Command Window, verrebbe inserito nell'instance `myinst`.

To list the instances, run the command:

```
db2ilist
```

On Linux, an instance must match a Linux operating system user; therefore, to switch between instances you can simply switch users (with the `su` command).

Table 4.1 shows some useful instance level commands.

Per elencare le instance, esegui il comando:

Su Linux, un'instance deve corrispondere ad un utente del sistema operativo; quindi, per passare da un'instance ad un'altra puoi semplicemente cambiare utente (con il comando `su`).

La tabella 4.1 mostra alcuni utili comandi a livello dell'instance.

| Command / Comando | Description | Descrizione |
|-------------------------------|---|--|
| <code>db2start</code> | Starts the current instance | Inizia l'instance corrente |
| <code>db2stop</code> | Stops the current instance | Ferma l'instance corrente |
| <code>db2icrt</code> | Creates a new instance | Crea una nuova instance |
| <code>db2idrop</code> | Drops an instance | Abbandona un'instance |
| <code>db2ilist</code> | Lists the instances you have on your system | Elenca le instance che hai sul tuo sistema |
| <code>db2 get instance</code> | Lists the current active instance | Elenca l'instance corrente attiva |

Table 4.1 – Useful DB2 commands at the instance level / Tabella 4.1 - Comandi utili di DB2 a livello dell'instance

Some of the above commands can instead be performed via the Control Center. For example, in the Control Center, if you expand the *Instances* folder and right-click the desired instance you can choose *Start* which is equivalent to issuing the `db2start` command from the DB2 Command Window, or *Stop*, which is equivalent to issuing a `db2stop` command as shown in Figure 4.5.

Alcuni dei comandi precedenti possono essere eseguiti usando il Control Center. Per esempio, nel Control Center, se espandi la cartella *Instances* e fai clic con il tasto destro sull'instance desiderata puoi scegliere *Start* che equivale a immettere il comando `db2start` dalla Command Window DB2, o *Stop*, che è equivalente a immettere un comando `db2stop` come mostrato in figura 4.5.

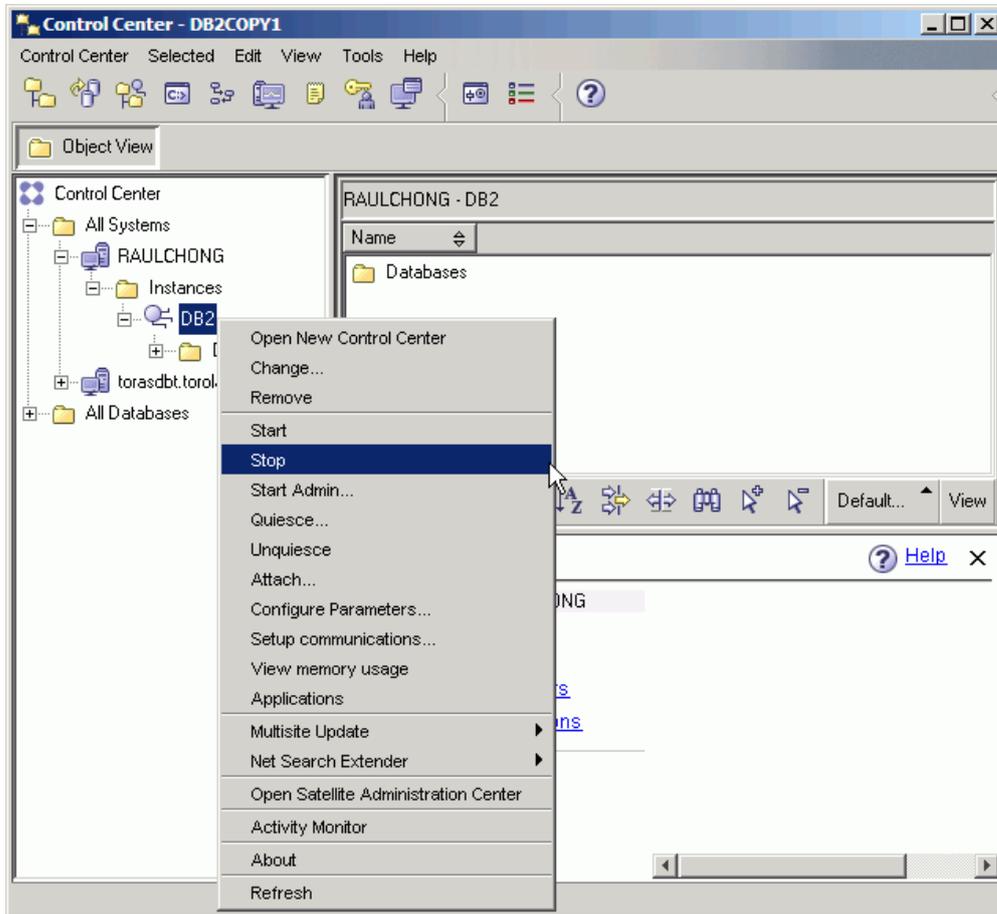


Figure 4.5 – Instance commands from the Control Center / Figura 4.5 – Comandi di instance dal Control Center

To create a database in the active instance, issue this command from the DB2 Command Window:

```
db2 create database mydb1
```

To list all the databases created, run the command:

```
db2 list db directory
```

Within any one instance, you can create many databases. A database is a collection of objects such as tables, views, indexes, and so on. Databases are independent units, and therefore, do

Per creare un database nell'instance attiva, immetti questo comando dalla Command Window DB2:

Per elencare tutti i database creati, esegui il comando:

All'interno di qualsiasi instance, puoi creare molti database. Un database è una collezione di oggetti come tabelle, view, indici e così via. I database sono unità indipendenti e quindi, non

not share objects with other databases. Figure 4.6 shows a representation of a database “MYDB1” created inside instance “DB2”.

condividono gli oggetti con altri database. La figura 4.6 mostra una rappresentazione di un database “MYDB1” creato all'interno del instance “DB2”.

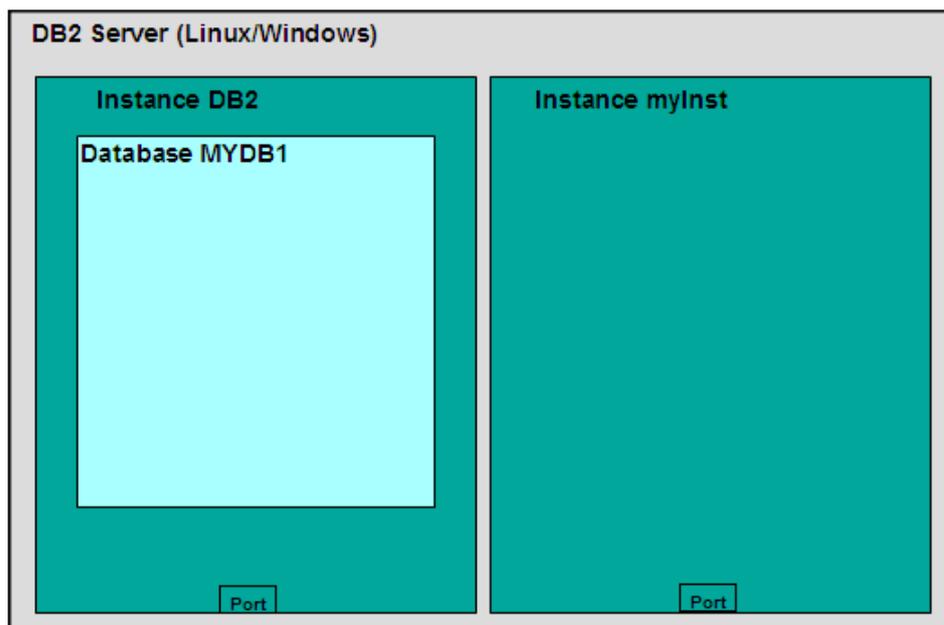


Figure 4.6 – Database “MYDB1” created in instance “DB2” / Figura 4.6 – Il database “MYDB1” creato nell’instance “DB2”

Table 4.2 shows some commands you can use at the database level.

La tabella 4.2 mostra alcuni comandi che puoi usare al livello del database.

| Command/SQL statement / Comandi / SQL statement | Description | Descrizione |
|--|---|---|
| db2 create database | Creates a new database | Crea un nuovo database |
| db2 drop database | Drops a database | Abbandona un database |
| db2 connect to <database_name> | Connects to a database | Si connette ad un database |
| db2 create table/create view/create index | SQL statements to create table, views, and indexes respectively | Comandi SQL per creare rispettivamente tabelle, view e indici |

Table 4.2 - Commands/SQL Statements at the database level / Tabella 4.2 - Comandi/SQL statement a livello di database

If we want to create another database with the same name (MYDB1) but in instance “myinst”, the following commands from the DB2 Command Window would be issued:

```
db2 list db directory
set db2instance=myinst
db2 create database mydb1
set db2instance=db2
```

Figure 4.7 shows the new database “MYDB1” created in instance “myinst”.

Se vogliamo creare un altro database con lo stesso nome (MYDB1) all'interno dell'instance "myinst", si dovranno eseguire i seguenti comandi dalla Command Window DB2:

La figura 4.7 mostra il nuovo database "MYDB1" creato nell'instance "myinst".

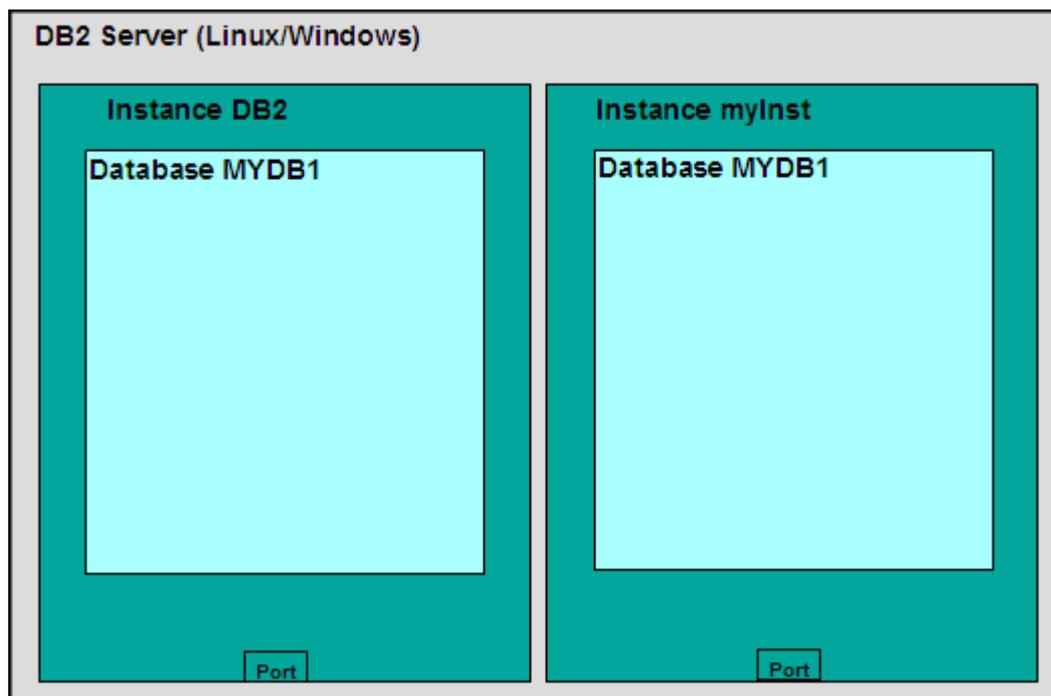


Figure 4.7 – Database “MYDB1” created in instance “myinst” / Figura 4.7 – Il database “MYDB1” creato nell’instance “myinst”

When a database is created, there are several objects created by default: Table spaces, tables, a buffer pool and log files. Creating these objects takes a bit of time, that’s why the `create database` command requires a few

Quando un database è creato, vengono anche prodotti di default diversi oggetti: gli spazi di tabella, le tabelle, un buffer pool e i file di log. La creazione di questi oggetti richiede un po' di tempo, ecco perché il comando `create database`

minutes for processing. Figure 4.8 shows three table spaces created by default. Table spaces will be discussed in more detail in Chapter 6, DB2 Architecture; but for now, think of table spaces as the logical layer between logical tables, and physical resources such as disks and memory.

ha bisogno di alcuni minuti per completare la sua esecuzione. La figura 4.8 mostra tre spazi di tabella creati di default. Gli spazi di tabella saranno discussi con maggiore dettaglio nel capitolo 6, L'architettura DB2; per ora, pensa agli spazi di tabella come uno strato logico fra le tabelle logiche e quelle fisiche come i dischi e la memoria.

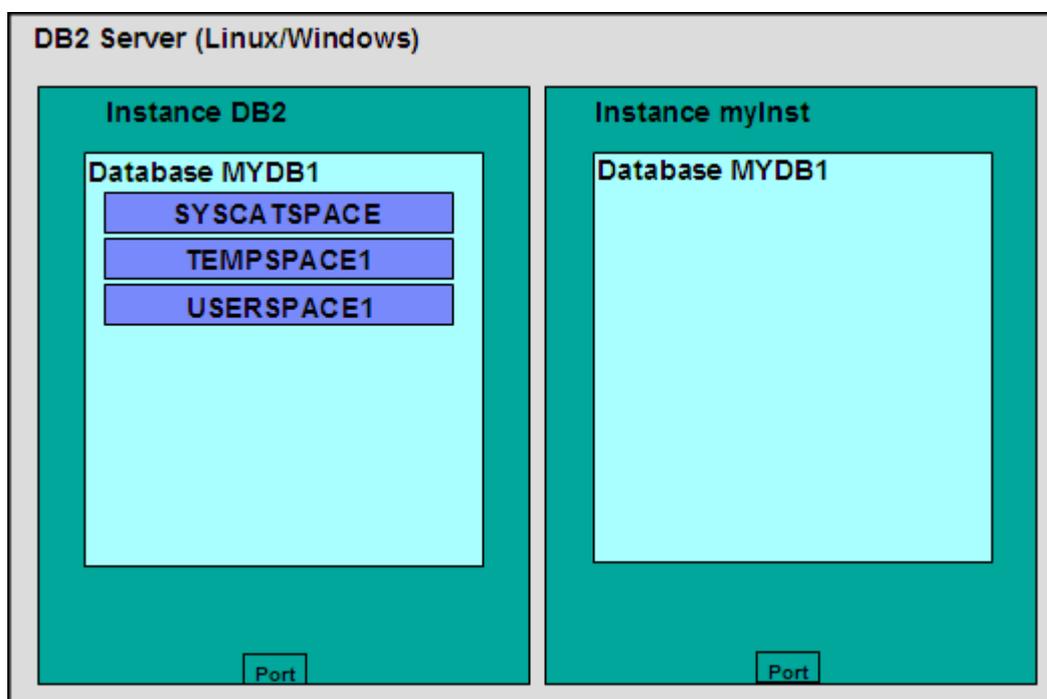


Figure 4.8 –Table spaces created by default when a database is created / Figura 4.8 - Spazi di tabella creati di default quando un database è creato

Table space SYSCATSPACE contains the Catalog tables. The Catalog is also known in other relational database management systems as the data dictionary. It basically contains system information that should not be modified or deleted; otherwise the database will not work correctly.

Lo spazio di tabella SYSCATSPACE contiene le tabelle del Catalogo. Il Catalogo in altri sistemi di gestione di database relazionali è conosciuto come il dizionario dei dati. Contiene fondamentalmente le informazioni di sistema che non dovrebbero essere modificate o cancellate; altrimenti il database potrebbe non funzionare correttamente.

Table space TEMPSPACE1 is used by

Lo spazio di tabella TEMPSPACE1 è

DB2 when it needs additional space to perform some operations such as sorts. Table space USERSPACE1 is normally used to store user database tables if there is no table space specified when creating a table.

You can also create your own table spaces using the CREATE TABLESPACE statement. Figure 4.9 shows the table space MYTBLS1 created inside database MYDB1 on instance DB2. When you create a table space, you specify the disks to use and the memory (buffer pool) to use. Therefore, if you have a “hot” table, that is, a table that is used very often, you can allocate the fastest disks and the most memory by assigning a table space with these characteristics.

In Figure 4.9, we show two other objects created by default: A buffer pool called IBMDEFAULTBP, and the log files.

A buffer pool is basically cache used by the database. You can create one or more buffer pools, but there should always be one buffer pool with a page size that matches the page size of existing table spaces. Pages and page size will be discussed in more detail in Chapter 6, DB2 Architecture.

The log files are used for recovery. When you work on a database, not only is the information stored in the disks for the database, but while you are working on the database, log files store all the operations executed on the data. Think of logs as temporary files where an “autosave” operation is performed. Logs are discussed in more detail in Chapter

usato da DB2 quando ha bisogno di più spazio per eseguire alcune operazioni come i sort. Lo spazio di tabella USERSPACE1 è usato normalmente per la memorizzazione delle tabelle di database dell'utente se non ne viene specificato uno quando la tabella è creata.

Puoi anche creare i tuoi spazi di tabella usando l'istruzione CREATE TABLESPACE. La figura 4.9 mostra lo spazio di tabella MYTBLS1 creato all'interno del database MYDB1 sull'instance DB2. Quando creerai uno spazio di tabella, specificherai i dischi e la memoria (buffer pool) da usare. Di conseguenza, se hai una tabella "hot", cioè, una tabella che è usata molto spesso, puoi riservare i dischi più veloci e la maggior parte della memoria assegnando uno spazio di tabella con queste caratteristiche.

Nella figura 4.9, mostriamo altri due oggetti creati di default: Un buffer pool chiamato IBMDEFAULTBP e i file di log.

Un buffer pool è fondamentalmente una cache usata dal database. Puoi creare uno o più buffer pool, ma ci dovrebbe sempre essere un buffer pool con una dimensione di pagina che è uguale al page size degli spazi attuali di tabella. Le pagine e la dimensione delle pagine sono discussi in maggiore dettaglio nel capitolo 6, L'architettura DB2.

I file di log sono usati per il recupero. Quando lavori su un database, ogni operazione eseguita sui dati viene memorizzata nei file di log. Puoi pensare ai log come file temporanei su cui viene eseguita una funzione "autosave". I log sono discussi con maggiore dettaglio nel capitolo 11: Backup e Recovery.

11: Backup and Recovery.

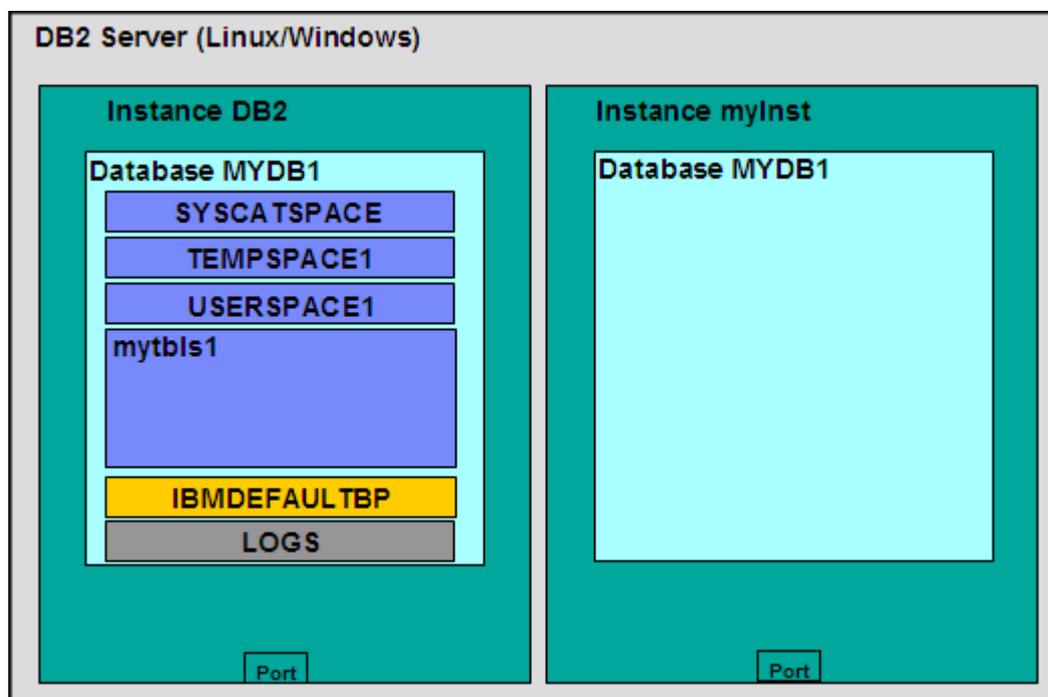


Figure 4.9 – Buffer pool and logs created by default / Figura 4.9 - Buffer pool e log creati di default

Earlier we discussed that instances are independent environments, and therefore, a database with the same name could be created in several instances. Just like instances, databases are also independent units; therefore, objects in one database have no relationship to objects in another database.

In precedenza abbiamo discusso come le instance siano ambienti indipendenti e quindi, un database con lo stesso nome potrebbe essere creato in parecchie instance. Come le instance, anche i database sono unità indipendenti; quindi, gli oggetti tra diversi database non hanno alcun rapporto tra loro.

Figure 4.10 shows the table space "mytbls1" inside both the database MYDB1 and the database SAMPLE, within instance DB2. This is valid because the databases are independent units. Note that Figure 4.10 does not show the other default objects of database SAMPLE due to space constraints in the figure.

La figura 4.10 mostra la parte interna dello spazio di tabella "mytbls1" sia nel database MYDB1 che in SAMPLE, all'interno dell'instance DB2. Questo è valido perché i database sono unità indipendenti. Nota che la figura 4.10 non mostra gli altri oggetti di default del database SAMPLE a causa di vincoli di spazio nella figura.

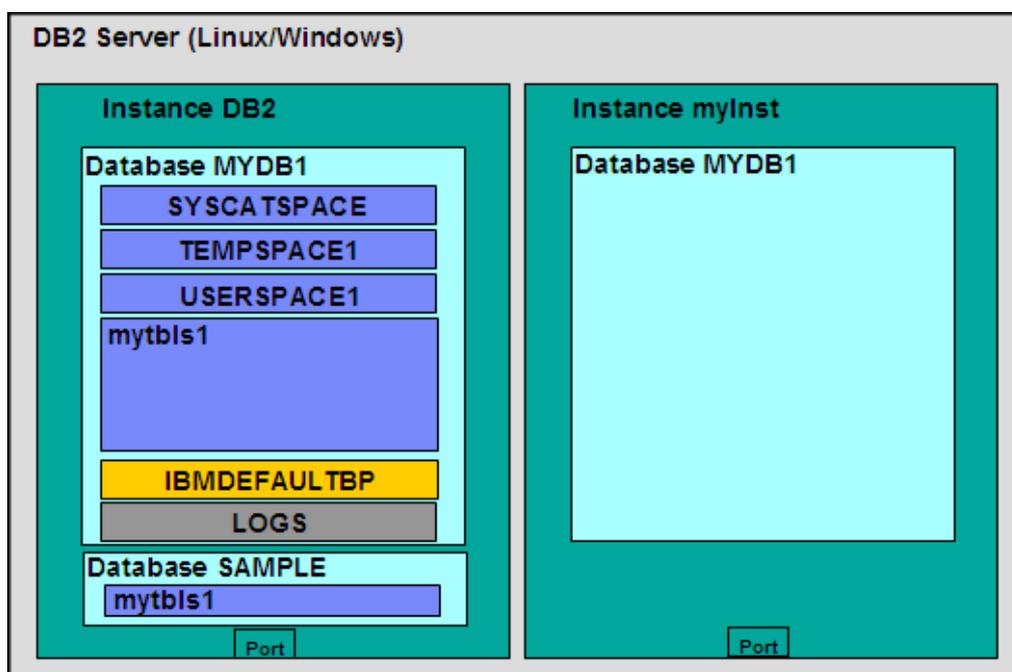


Figure 4.10 – Table spaces with the same name in different databases / Figura 4.10 - Spazi di tabella con lo stesso nome in database diversi

Once you have created a table space, you can create objects inside the table space such as tables, views and indexes. This is illustrated in Figure 4.11.

Una volta che hai creato uno spazio di tabella, al suo interno puoi creare oggetti come tabelle, view e indici. Come è illustrato nella figura 4.11.

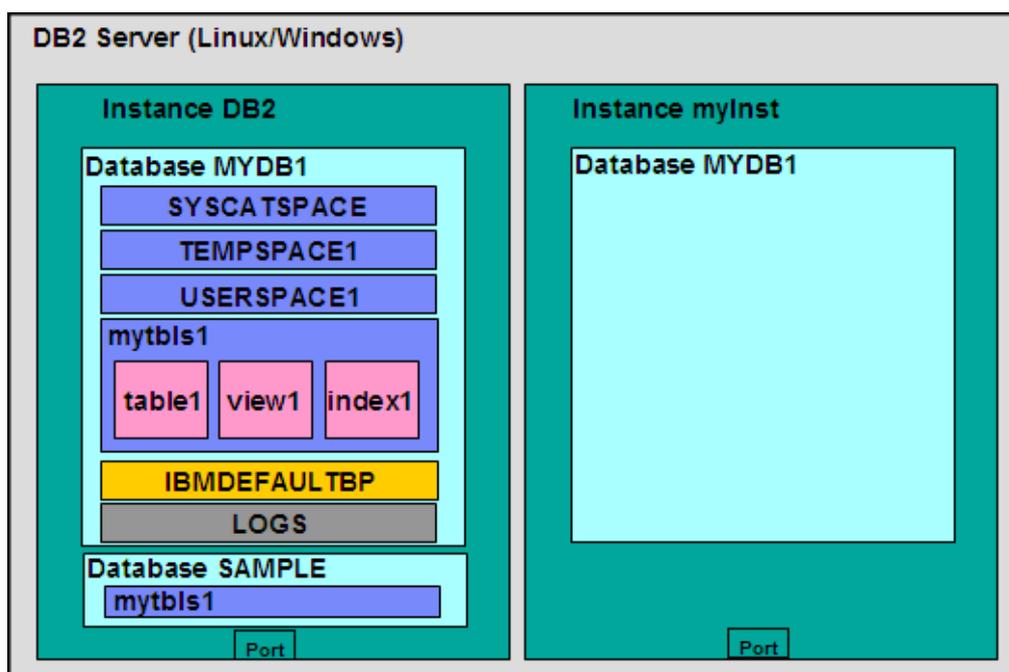


Figure 4.11 – Tables, views, indexes created inside the table space / Figura 4.11 - Tabelle, view, e indici creati all'interno dello spazio di tabella

Quicklab #2 – Create a new database / Creare un nuovo databaseObjective

In this Quicklab, you will create a new database using the Create Database wizard in the Control Center.

Procedure

1. From the Control Center Object Tree pane, right-click the *All Databases* folder, select the *Create Database* item, and choose the *With Automatic Maintenance* item. This launches the Create Database Wizard.
2. Specify the database name and location in the *Name* page of the wizard. Use the following values:

Database Name: EXPRESS

Default Drive (Windows): C:

Default Path: (Linux): /home/db2inst1

Alias: This will default to EXPRESS if left blank

Comment: This is optional and can be left blank

Click on the *Next* button to continue to the next page of the wizard.

3. In the *Specify where to store your data* page, don't make any changes, and click *Next*.
4. In the *Select your maintenance strategy* page, leave the default (*Yes, I can specify an offline ...*), and click *Next*.

Obiettivo

In questo Quicklab, creerai un nuovo database usando il Create Database wizard nel Control Center.

Procedura

1. Dalla vista ad albero nel Control Center, fai clic con il tasto destro sulla cartella *All Databases*, seleziona *Create Database* e scegli *With Automatic Maintenance*. Questo lancia il Create Database Wizard.
2. Specifica il nome e la destinazione del database nella pagina *Name* del wizard. Usa i seguenti valori:

Database Name: EXPRESS

Default Drive (Windows): C:

Default Path: (Linux): /home/db2inst1

Alias: This will default to EXPRESS if left blank

Comment: This is optional and can be left blank

Clicca *Next* per passare alla pagina successiva del wizard.

3. Nella pagine *Specify where to store your data*, non devi fare alcun cambiamento, premi soltanto *Next*.
4. Nella pagine *Select your maintenance strategy*, lascia la scelta di default (*Yes, I can specify an offline ...*), e clicca *Next*.

5. Specify the offline maintenance time window in the *Timing* page of the wizard. Specify two or more hours a week when DB2 can perform automated maintenance tasks to preserve the health of your database. For now, configure the window to start at 1AM every Monday through Thursday for a 6 hour period. Click the *Next* button.
 6. Configure notification on the *Mail Server* page of the wizard. DB2 can automatically send an email or a page if a problem or unhealthy condition is detected. If you want to configure this, indicate an available SMTP server for DB2 to use for sending email. For this lab we don't have an SMTP server, so leave this blank and click *Next*.
 7. Review the options selected on the *Summary* page of the wizard. Click the *Finish* button to begin the database creation process. Database creation usually takes a few minutes, during which time a progress indicator is displayed.
5. Specifica il periodo di manutenzione offline nella pagina *Timing* del wizard. Imposta due o più ore alla settimana in cui DB2 può eseguire le attività di manutenzione automatizzate per conservare la consistenza del tuo database. Per ora, configuralo per iniziare all'una di mattina da lunedì fino a giovedì per un periodo di 6 ore. Fatto questo premi *Next*.
 6. Configura la notifica alla pagina *Mail Server* del wizard. DB2 può inviare automaticamente un email o una pagina se un problema o una condizione non valida è stata rilevata. Se vuoi configurare questo, indica un server SMTP disponibile per DB2 da usare per l'invio delle email. Per questo laboratorio non abbiamo un server SMTP, quindi lascia questo spazio vuoto e clicca *Next*.
 7. Controlla le opzioni selezionate alla pagina *Summary* del wizard. Clicca il tasto *Finish* per far cominciare il processo di creazione del database. La creazione del database richiede solitamente alcuni minuti, nel frattempo è visualizzato un indicatore di progresso.

4.1 DB2 configuration / Configurazione di DB2

DB2 parameters can be configured using the Configuration Advisor Tool. To access the configuration advisor through the Control Center, right click on a database and choose "Configuration Advisor". Based on your answers to some questions about your system resources and workload, the configuration advisor will provide a list of DB2 parameters that should be changed with suggested values for each. If you would like more detail about DB2 configuration, keep reading; otherwise, use the Configuration Advisor and you are good to work with DB2!

I parametri di DB2 possono essere configurati usando il Configuration Advisor Tool. Per accedere all'advisor della configurazione attraverso il Control Center, fai clic con il tasto destro su un database e scegli "Configuration Advisor". Sulla base delle tue risposte ad alcune domande riguardanti le risorse di sistema e il carico di lavoro, l'advisor della configurazione fornirà una lista di parametri di DB2 che dovrebbero essere cambiati con i valori suggeriti per ciascuno di questi. Se vuoi più dettagli sulla configurazione di DB2, continua a leggere; altrimenti, usa il Configuration Advisor e sarai pronto a lavorare con DB2!

A DB2 server can be configured at four different levels:

- Environment variables
- Database manager configuration file (dbm cfg)
- Database configuration file (db cfg)
- DB2 profile registry

This is also shown in Figure 4.12. In the figure, note where each of the boxes reside. For example, environment variables are set at the operating system level of the server, while database manager configuration file parameters are set at the instance level. Database configuration parameters are set at the database level, and the DB2 profile registry is set either at the operating system or instance level.

Un server DB2 può essere configurato in quattro livelli diversi:

- Variabili di ambiente
- File di configurazione del database manager (dbm cfg)
- File di configurazione del database (db cfg)
- Registro del profilo DB2

Ciò è indicato anche nella figura 4.12. Nella figura, nota dove si trova ciascun box. Per esempio, le variabili di ambiente sono settate a livello del sistema operativo del server, mentre i parametri del file di configurazione del database manager sono settate a livello dell'instance. I parametri di configurazione del database sono settati a livello del database e il registro del profilo DB2 è settato sia a livello del sistema operativo che dell'instance.

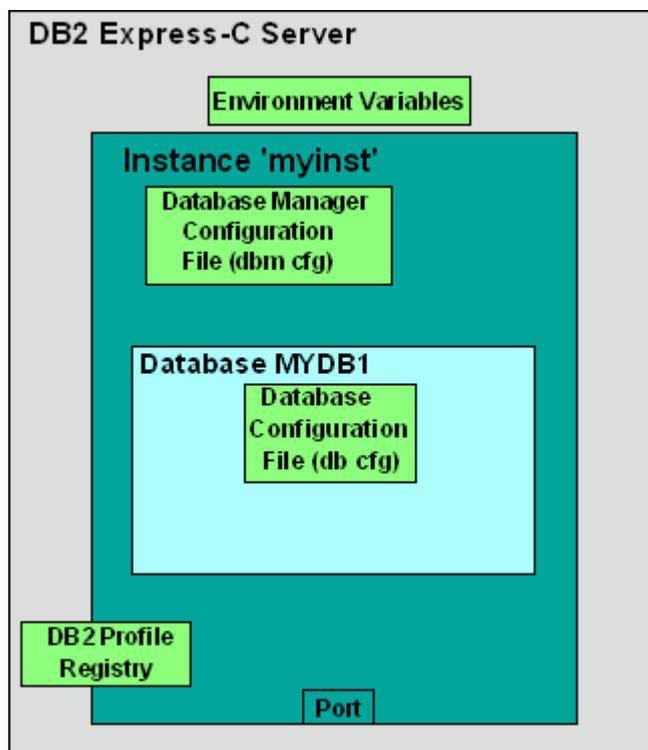


Figure 4.12 – DB2 Configuration / Figura 4.12 - Configurazione di DB2

4.1.1 Environment variables / Variabili d'ambiente

Environment variables are variables set at the operating system level. One key environment variable is DB2INSTANCE. This variable indicates the active instance you are working on, and for which your DB2 commands would apply. For example, to set the active instance to "myinst" in Windows, you can run this operating system command:

```
set db2instance=myinst
```

Le variabili d'ambiente sono variabili settate a livello del sistema operativo. Una variabile d'ambiente chiave è DB2INSTANCE. Questa variabile indica l'instance attiva con cui stai lavorando e su cui i tuoi comandi di DB2 vengono eseguiti. Per esempio, per settare l'instance attiva di "myinst" su Windows, puoi eseguire questo comando del sistema operativo:

4.1.2 Database manager configuration file (dbm cfg) / Il file di configurazione del database manager (dbm cfg)

The Database manager configuration file (dbm cfg) includes parameters that affect the instance and all the databases contained within. The

Il file di configurazione del database manager (dbm cfg) comprende i parametri che influiscono sull'instance e su tutti i database contenuti all'interno. Il

database manager configuration file can be viewed or modified using the command line, or through the DB2 Control Center.

file di configurazione del database manager può essere osservato o modificato usando la Command Line o con il DB2 Control Center.

To work with the DBM CFG from the Control Center, select the instance object from the instance folder of the control center, right-click to reveal the popup menu and select *Configure Parameters*. This is shown in Figure 4.13.

Per lavorare con il DBM CFG dal Control Center, seleziona l' instance object dalla cartella dell'instance del Control Center, fai clic con il tasto destro per far apparire il popup e scegli *Configure Parameters*. Ciò è mostrato nella figura 4.13.

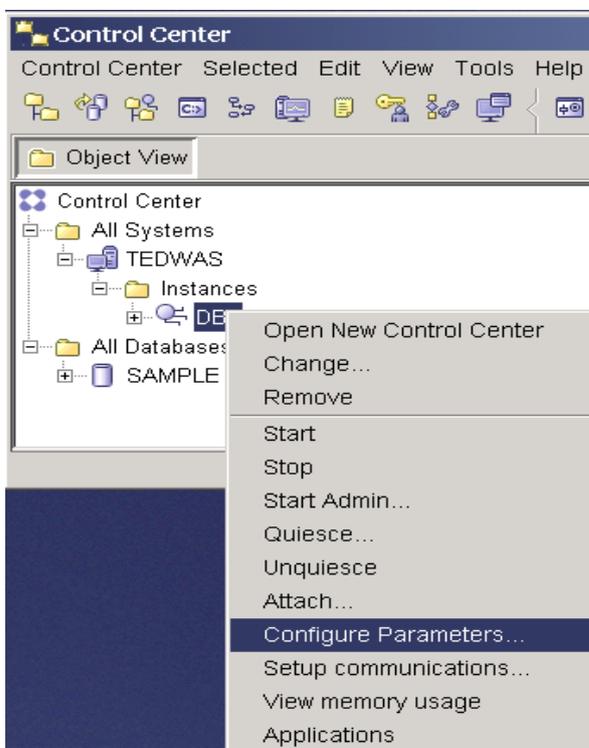


Figure 4.13 – Configuring the dbm cfg from the Control Center / Figura 4.13 - Configurazione di dbm cfg dal Control Center

After choosing *Configure Parameters*, the screen shown in Figure 4.14 will be displayed with the list of dbm cfg parameters.

Dopo la scelta *Configure Parameters*, la schermata mostrata in figura 4.14 sarà visualizzata con la lista dei parametri di dbm cfg.

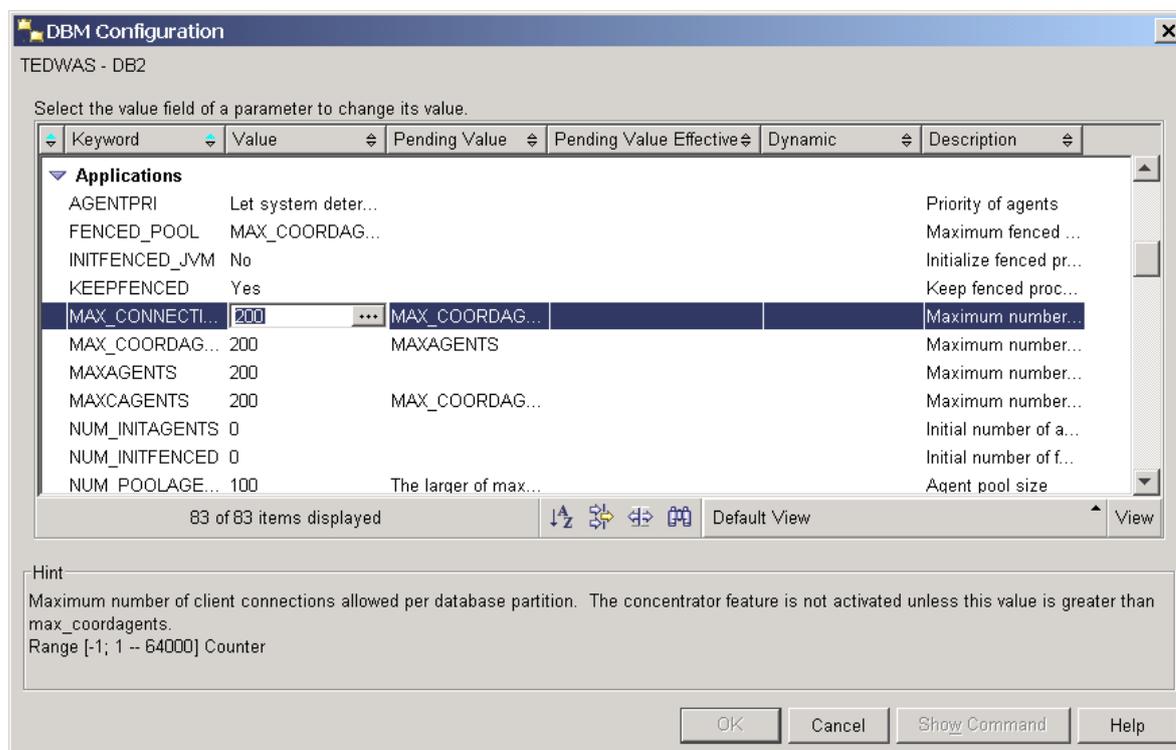


Figure 4.14 –The dbm cfg dialog / Figura 4.14 - La finestra dialogo di dbm cfg

Many parameters are dynamic meaning that changes take effect immediately; however, changes to some parameters may require stopping and starting the instance. From the Command line, this can be done using the `db2stop` and `db2start` commands.

Before an instance can be stopped, all applications must disconnect. If you wish to forcefully stop the instance, you can use the `db2stop force` command.

An instance can also be stopped through the Control Center by clicking on the instance object and selecting either *Stop* or *Start*.

Table 4.3 shows some useful commands to manage the dbm cfg from the Command Line.

Molti parametri sono dinamici quindi il cambiamento è immediato; tuttavia, i cambiamenti di alcuni parametri possono richiedere che l'instance sia fermata e riavviata. Dalla Command Line, questo può essere fatto usando i comandi di `db2stop` e di `db2start`.

Prima che un instance possa essere fermata, tutte le applicazioni devono disconnettersi. Se desideri forzare l'arresto dell'instance, puoi usare il comando `db2stop force`.

Un'instance può anche essere fermata attraverso il Control Center cliccando sull'instance object e selezionando *Stop* o *Start*.

La tabella 4.3 mostra alcuni comandi utili per gestire il dbm cfg dalla Command Line.

| Command | Description | Descrizione |
|--|--|--|
| db2 get dbm cfg | Retrieves information about the dbm cfg | Richiama le informazioni sul dbm cfg |
| db2 update dbm cfg using <parameter_name> <value> | Updates the value of a dbm cfg parameter | Aggiorna il valore di un parametro del dbm cfg |

Table 4.3 - Commands to manipulate the dbm cfg / Tabella 4.3 - Comandi per manipolare il dbm cfg

4.1.3 Database configuration file (db cfg) / Il file di configurazione del database (db cfg)

The database configuration file (db cfg) includes parameters that affect a particular database. The database configuration file can be viewed or modified using the command line, or through the DB2 Control Center.

Il file di configurazione del database (db cfg) comprende i parametri che influenzano un particolare database. Il file di configurazione del database può essere osservato o modificato usando il Command Line, o con il DB2 Control Center.

To work with the DB CFG from the Control Center, select the database object from the database folder of the Control Center, right-click to reveal the popup menu and select Configure Parameters. This is shown in Figure 4.15.

Per lavorare con il DB CFG dal Control Center, seleziona il database dalla cartella Database dal Control Center, fai clic con il tasto destro per mostrare il menù popup e scegli Configure Parameters. Ciò è mostrato nella figura 4.15.

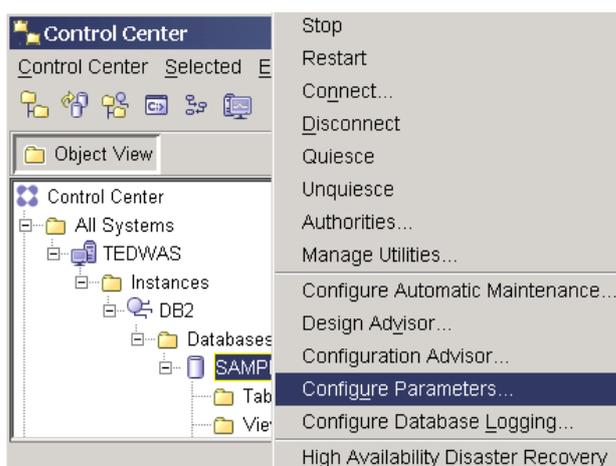


Figure 4.15 – Configuring the db cfg from the Control Center / Figura 4.15 - Configurazione di db cfg dal Control Center

After choosing Configure Parameters, the screen shown in Figure 4.16 will be displayed with the list of db cfg parameters.

Dopo la scelta del Configure Parameters, la schermata che appare in figura 4.16 sarà visualizzata con la lista dei parametri del db cfg.

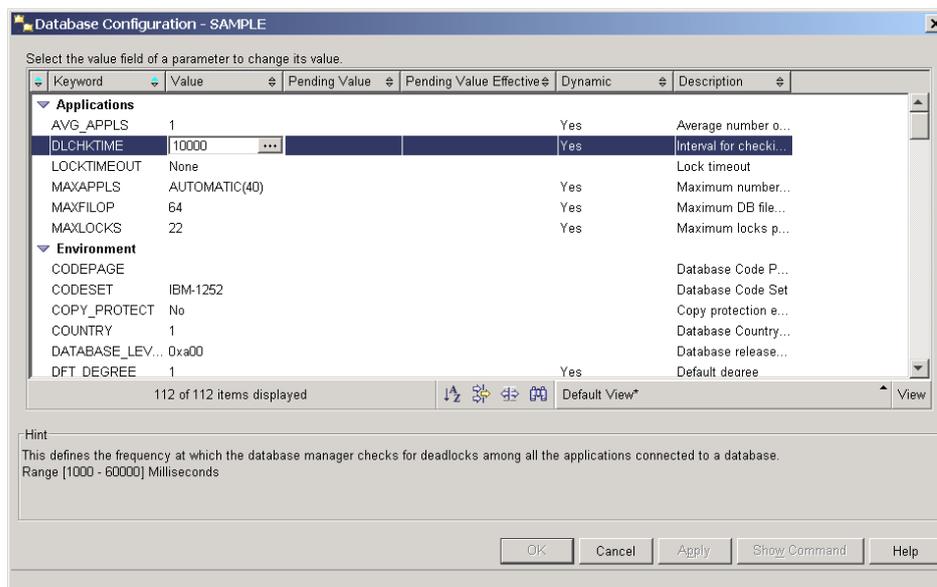


Figure 4.16 –The db cfg / Figura 4.16 - Il db cfg

Table 4.4 shows some useful commands to manage the db cfg from the Command Line.

La tabella 4.4 mostra alcuni comandi utili per gestire il db cfg dal Command Line.

| Command | Description | Descrizione |
|---|---|--|
| <code>get db cfg for <database_name></code> | Retrieves information about the db cfg for a given database | Ottieni le informazioni sul db cfg per un certo database |
| <code>update db cfg for <database_name> using <parameter_name> <value></code> | Updates the value of a db cfg parameter | Aggiorna il valore di un parametro del db cfg |

Table 4.4 - Commands to manipulate the db cfg / Tabella 4.4 - Comandi per manipolare il db cfg

4.1.4 DB2 profile registry / Il registro del profilo di DB2

DB2 profile registry variables include parameters that may be platform specific and can be set globally (affecting all instances), or at the instance level (affecting one particular instance).

Le variabili del registro del profilo di DB2 comprendono i parametri specifici della piattaforma e possono essere settati globalmente (quindi influenzando tutte le instance), o al livello delle instance (influenzando un'instance particolare).

Table 4.5 shows some useful commands to manipulate the DB2 profile registry

La tabella 4.5 mostra alcuni comandi utili per manipolare il registro del profilo di DB2.

| Command | Description | Descrizione |
|---|---|--|
| <code>db2set -all</code> | Lists all the DB2 profile registry variables that are currently set | Elenca tutte le variabili del registro del profilo di DB2 che attualmente sono settate |
| <code>db2set -lr</code> | Lists all the DB2 profile registry variables | Elenca tutte le variabili del registro del profilo di DB2 |
| <code>db2set <parameter>=<value></code> | Assigns a parameter with a given value | Assegna ad un parametro un valore specifico |

Table 4.5 - Commands to manipulate the DB2 profile registry / Tabella 4.5 - Comandi per modificare il registro del profilo di DB2

Table 4.6 shows some of the most commonly used DB2 registry variables

La tabella 4.6 mostra alcune delle variabili di registro più comunemente usate in DB2

| Registry Variable | Description | Descrizione |
|-------------------|---|--|
| DB2COMM | Specifies the communication managers that are started when the database manager is started. | Specifica i gestori di comunicazione che vengono avviati quando il manager del database è avviato. |
| DB2_EXTSECURITY | On Windows, prevents unauthorized access to DB2 by locking DB2 system files | Su Windows, impedisce l'accesso non autorizzato a DB2 bloccando i file di sistema DB2 |
| DB2_COPY_NAME | Stores the name of the DB2 copy currently in use. | Memorizza il nome della copia DB2 attualmente in uso. Per |

| | | |
|--|---|---|
| | To switch to a different DB2 copy installed, run the installpath\bin\db2envvars.bat command. This variable cannot be used for this purpose. | passare ad una differente copia di DB2 installata, esegui il comando installpath\bin\db2envvars.bat . Questa variabile non può essere usata per questo scopo. |
|--|---|---|

Table 4.6 – Commonly used DB2 profile registry variables / Tabella 4.6 - Variabili di registro del profilo di DB2 comunemente usate

For example, to allow for communication using TCPIP, set the DB2COMM registry variable to TCPIP as shown below:

```
db2set db2comm=tcPIP
```

Per esempio, per permettere la comunicazione usando TCPIP, setta la variabile di registro DB2COMM al valore TCPIP come mostrato qui sotto:

4.2 The DB2 Administration Server / Il Server di Amministrazione di DB2

The DB2 Administration Server (DAS) is a daemon process that runs at the DB2 server to allow remote clients to graphically administer the DB2 server. There is only one DAS per physical computer as shown in Figure 4.16.

Il Server di Amministrazione di DB2 (DAS) è un processo demone che avvia il server DB2 per permettere ai client remoti di amministrarlo graficamente. C'è soltanto un DAS per ogni computer fisico come mostrato in figura 4.16.

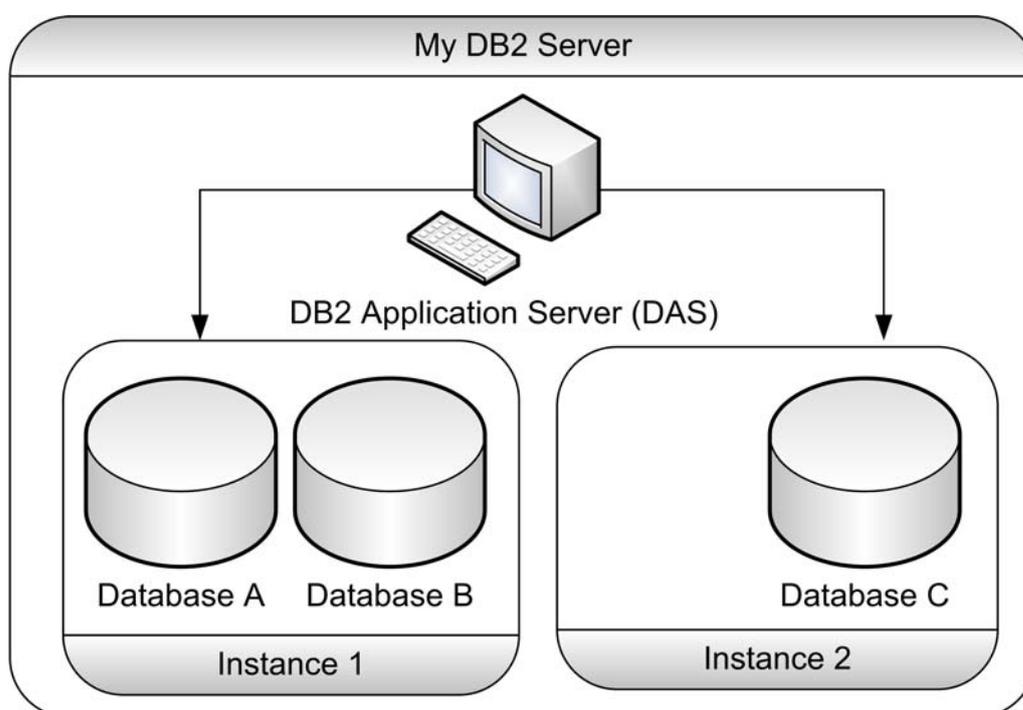


Figure 4.16 –The DB2 Administration Server (DAS) / Figura 4.16 - Il Server di Amministrazione di DB2 (DAS)

Quicklab #3 – Working with instances, databases and configuration / Lavorare con instance, database e configurazione

Objective

In this Quicklab, you will create a new instance, database, and change configuration parameters on a DB2 server on Windows. You can do it from either the Control Center or the Command Window. We provide the instructions using the Command Window.

Procedure

1. From Command Window, create a new instance called *newinst*

```
db2icrt newinst
```

2. In the new instance *newinst*, create a database *newdb* with default values

```
set db2instance=newinst
db2start
db2 create database newdb
```

3. List all the instances in your server

```
db2ilist
```

4. Switch to the *DB2* instance and make sure you really switched

```
set db2instance=db2
db2 get instance
```

Obiettivo

In questo Quicklab, creerai una nuova instance, un database e farai delle modifiche ai parametri di configurazione su un server DB2 installato su Windows. Puoi farlo dal Control Center o dalla Command Window. Forniremo le istruzioni usando la Command Window.

Procedura

1. Dalla Command Window, crea una nuova instance chiamata *newinst*

2. Nella nuova instance *newinst*, crea un database *newdb* con i valori di default

3. Elenca tutti le instance nel tuo server

4. Passa all'instance DB2 e assicurati di esserci veramente passato

- 5.** Change the dbm cfg parameter FEDERATED to a value of YES from NO and verify the change occurred.

```
db2 update dbm cfg using FEDERATED YES
db2 force applications all
db2 terminate
db2stop
db2start
db2 get dbm cfg
```

- 6.** Connect to the SAMPLE database with the userID/psw you are logged on to the operating system

```
db2 connect to sample user <userID> using <psw>
```

- 7.** Review how many applications are running in your current instance

```
db2 list applications show detail
```

- 8.** Open another DB2 Command Window and connect again to the SAMPLE database without specifying a userID/psw. Then review how many connections you have now.

```
db2 connect to sample
db2 list applications
```

- 9.** Force off one of the DB2 command windows

```
db2 force application
```

(application handle obtained from the db2 list applications command for application name "db2bp.exe")

- 5.** Cambia il parametro dbm cfg FEDERATED da NO a YES e verifica che il cambiamento sia avvenuto.

- 6.** Collegati al database SAMPLE con gli stessi userID/psw usati per entrare nel sistema operativo

- 7.** Controlla quante applicazioni stanno girando sulla tua instance corrente

- 8.** Apri un'altra DB2 Command Window e collegati ancora al database SAMPLE senza specificare userID/psw. Poi esamina quanti connessioni hai adesso.

- 9.** Forza una delle DB2 command windows a spegnersi

(l'applicazione è un handle ottenuto dal comando db2 list applications per il nome dell'applicazione)

“db2bp.exe”)

- 10.** Drop the instance *newinst*

```
db2idrop newinst
```

- 10.** Abbandona l'instance *newinst*

- 11.** Drop and recreate the DAS, and start it

```
db2admin stop
db2admin drop
db2admin create
db2admin start
```

- 11.** Abbandona e ricrea il DAS e fallo partire

- 12.** Set the DB2 Registry variable DB2COMM to *tcpip* and *npipe* in your instance

```
db2set db2comm=tcpip,npipe
db2stop
db2start
```

- 12.** Setta la variabile di registro DB2COMM a *tcpip* e *npipe* nella tua instance

- 13.** Unset the DB2COMM registry variable

```
db2set db2comm=
db2stop
db2start
```

- 13.** Elimina la configurazione della variabile di registro DB2COMM

- 14.** Check the current value of the LOGSECOND db cfg parameter, and then change it to a value of 5 and verify the new value

```
db2 connect to sample
db2 get db cfg
db2 update db cfg using LOGSECOND 5
db2 get db cfg
```

- 14.** Controlla il valore corrente del parametro di db cfg LOGSECOND e poi cambiarlo con un valore pari a 5 infine verifica il nuovo valore

5

Chapter 5 – DB2 Tools / Capitolo 5 – Tool DB2

In this chapter, we describe some of the tools you can use with DB2. The red ellipse in Figure 5.1 shows the area of focus in this chapter.

In questo capitolo, descriviamo alcuni dei tool che puoi usare con DB2. L'ellisse rossa nella figura 5.1 mostra l'area di interesse di questo capitolo.

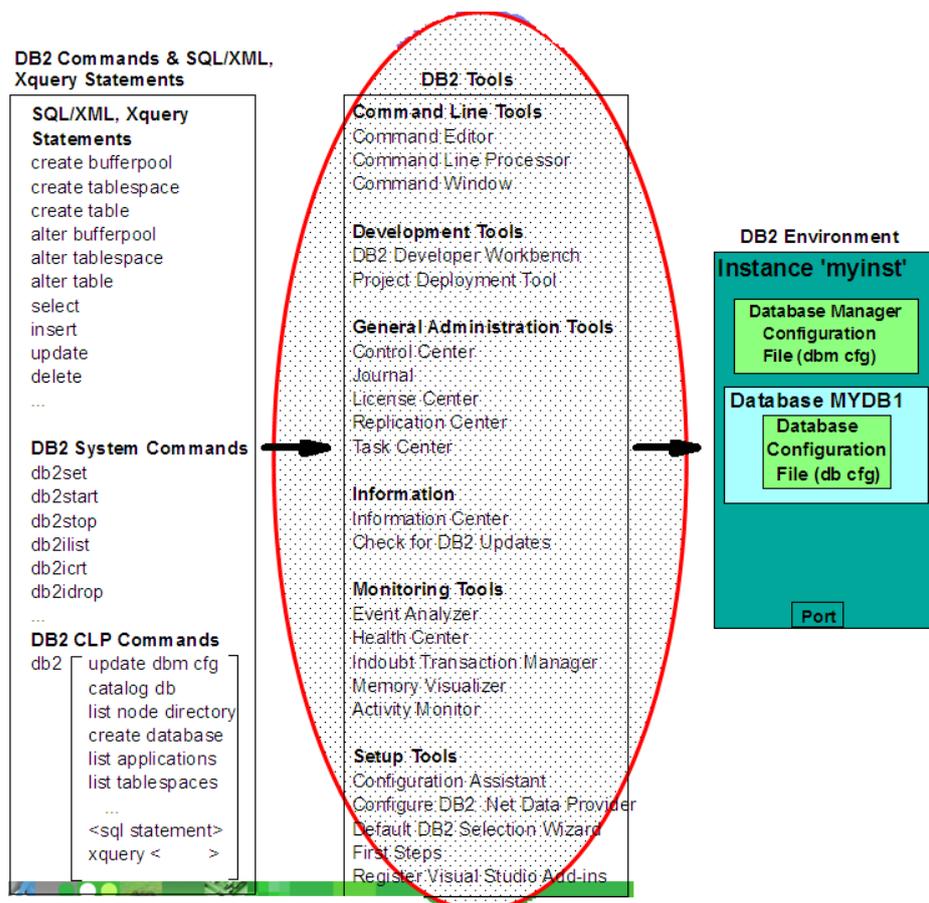


Figure 5.1 – The DB2 big picture: DB2 tools / Figura 5.1 - La big picture di DB2: Tool DB2

Note:

See video presentations about the DB2 Tools and scripting at these links:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4202>

<http://www.channeldb2.com/video/video/show?id=807741:Video:4182>

Nota:

Vedi la video-presentazione di tool e scripting di DB2 a questi link:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4202>

<http://www.channeldb2.com/video/video/show?id=807741:Video:4182>

Figure 5.2 lists all the DB2 Tools available from the IBM DB2 Start Menu shortcuts. Most of these tools are the same on Linux and Windows.

La figura 5.2 elenca tutti i tool di DB2 disponibili dai menù di scelta rapida dell'IBM Start Menù. La maggior parte di questi tool sono gli stessi su Linux e su Windows.

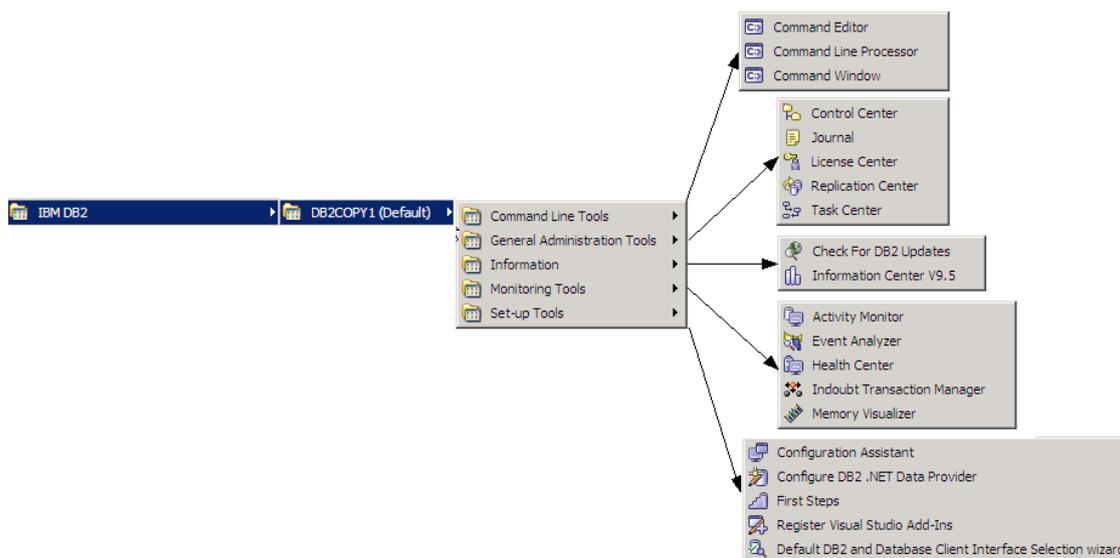


Figure 5.2 – DB2 tools from the IBM DB2 Start menu / Figura 5.2 - Tool di DB2 dal IBM DB2 Start menù

Table 5.1 provides a list of shortcut commands that can be used to start some of the most popular tools in either Linux or Windows.

La tabella 5.1 fornisce un elenco dei comandi di scelta rapida che possono essere usati per avviare alcuni dei tool più conosciuti sia in Linux che in Windows.

| Tool name / Nome del tool | Command / Comando |
|--|--------------------------|
| Command Editor | db2ce |
| Command Line processor | db2 |
| Command Window (Only on Windows platforms) | db2cmd |
| Control Center | db2cc |
| Task Center | db2tc |
| Health Center | db2hc |
| Configuration Assistant | db2ca |
| First Steps | db2fs |

Table 5.1 – Shortcut commands to start some DB2 tools / Tabella 5.1 - Comandi di scelta rapida per avviare alcuni tool di DB2

5.1 Control Center

The primary DB2 tool for database administration is the Control Center, illustrated in Figure 5.3.

Il tool primario di DB2 per la gestione del database è il Control Center, illustrato nella figura 5.3.

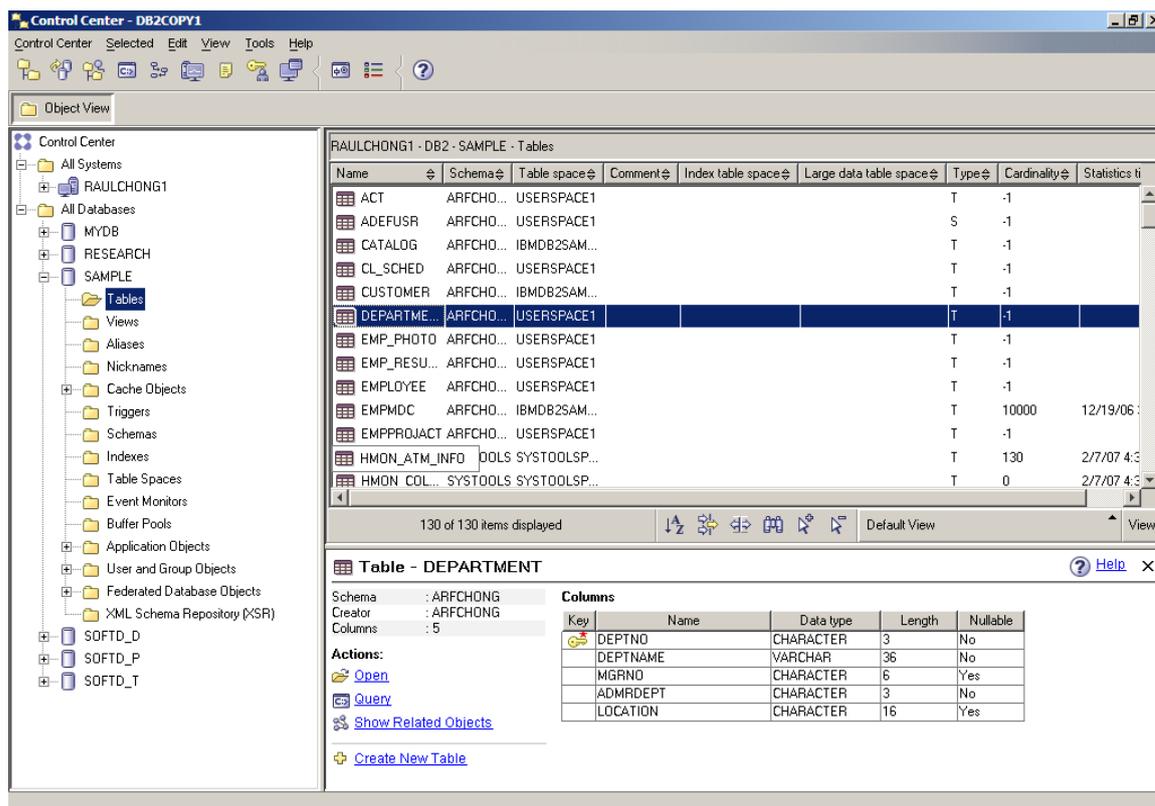


Figure 5.3 - The DB2 Control Center / Figura 5.3 – Il Control Center

The Control Center is a centralized administration tool that allows you to:

- View your systems, instances, databases and database objects;
- Create, modify and manage databases and database objects;
- Launch other DB2 graphical tools

The pane on the left side provides a visual hierarchy for the database objects on your system(s), providing a "folder" for Tables, Views, etc. When you double-click a folder (for example, the Tables folder, as shown in Figure

Il Control Center è un tool di gestione centralizzato che ti permette di:

- Osservare i tuoi sistemi, instance, database e database object;
- Creare, modificare e gestire i database e i oggetti di database;
- Lanciare altri tool grafici di DB2

Il pannello a sinistra fornisce una gerarchia visiva per database object sui tuoi sistemi, fornendo una "folder" per Tabelle, Viste, ecc. Quando fai doppio clic su una cartella (per esempio, la cartella Tabelle, come mostrato nella

5.3), the pane on the top right will list all the related objects, in this case, all the tables associated with the SAMPLE database. If you select a given table in the top right pane, the bottom right pane provides more specific information about that table.

Right clicking on the different folders/objects in the Object tree brings up menus applicable to the given folder/object. For example, right-clicking on an instance and choosing *Configure parameters* would allow you to view and update the database manager configuration file. Similarly, if you right-click on a database and choose *Configure parameters*, you would be able to view and update the database configuration file. The DB2 environment and configuration parameters are discussed in more detail in Chapter 4, DB2 Environment.

The first time you launch the Control Center, you are asked to choose what view you would like to use. The choice of view determines what types of options and database objects are exposed. Figure 5.4 shows the Control Center View dialog box.

figura 5.3), il pannello a destra in alto elencherà tutti gli oggetti relativi, in questo caso, tutte le tabelle associate al database SAMPLE. Se selezioni una specifica tabella nel pannello in alto a destra, il pannello in basso a destra fornisce informazioni specifiche su quella tabella.

Facendo clic destro sulle diverse cartelle/oggetti nel Object tree vengono mostrati i menù applicabili alla cartella/oggetto. Per esempio, fare clic destro su un'instance e scegliere *Configure parameters* ti permetterebbe di osservare ed aggiornare il file di configurazione del database manager. Allo stesso modo, con un clic destro su un database scegliendo *Configure parameters* potrai osservare ed aggiornare il file di configurazione del database. I parametri di configurazione e dell'ambiente DB2 sono discussi con maggiore dettaglio nel capitolo 4, L'ambiente DB2.

La prima volta che lanci il Control Center, ti viene chiesto di scegliere che vista vorrai usare. La scelta della vista determina che tipi di opzioni e di database object sono mostrati. La figura 5.4 mostra la finestra di dialogo della Control Center View.

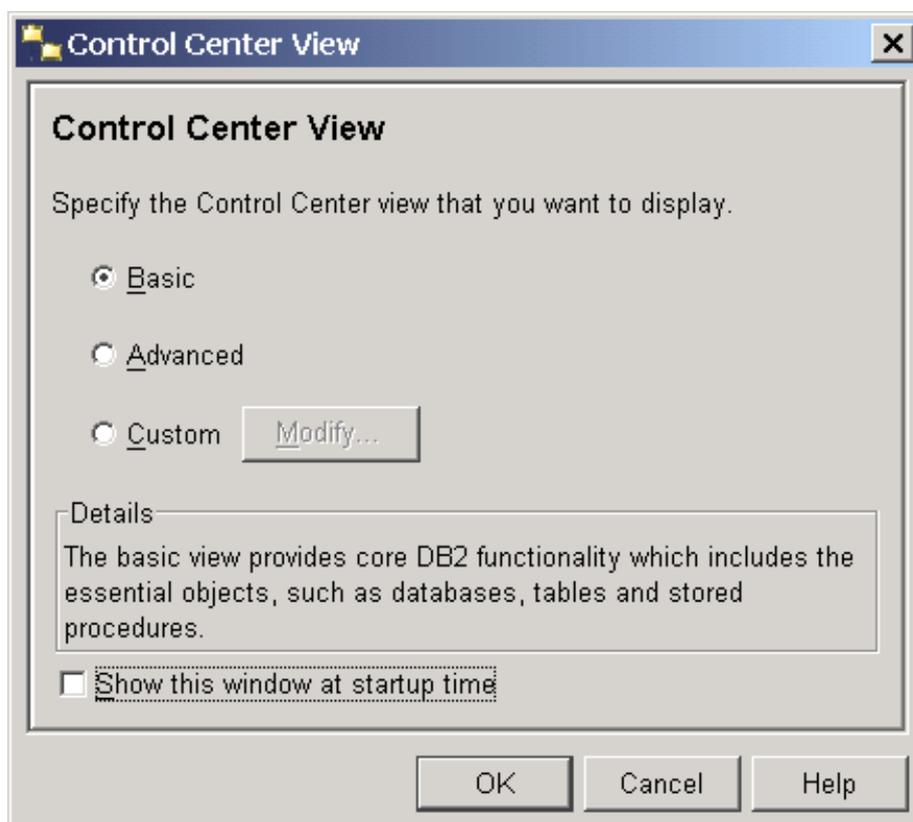


Figure 5.4 - The DB2 Control Center View Dialog Box / Figura 5.4 - La finestra di dialogo della DB2 Control Center

The basic view provides core DB2 functionality.

La vista di base fornisce le funzionalità principali di DB2.

The advanced view shows more options and features.

La vista avanzata mostra ulteriori opzioni e feature.

The custom view allows you to customize the specific features, options, and objects you see.

La vista custom ti permette di personalizzare feature specifiche, opzioni, e gli oggetti che vedi.

To re-invoke the Control Center View dialog, select the *Customize Control Center* option from the Tools menu as shown in Figure 5.5.

Per richiamare la finestra di dialogo del Control Center, seleziona l'opzione *Customize Control Center* dal menù Tools come mostrato in figura 5.5.

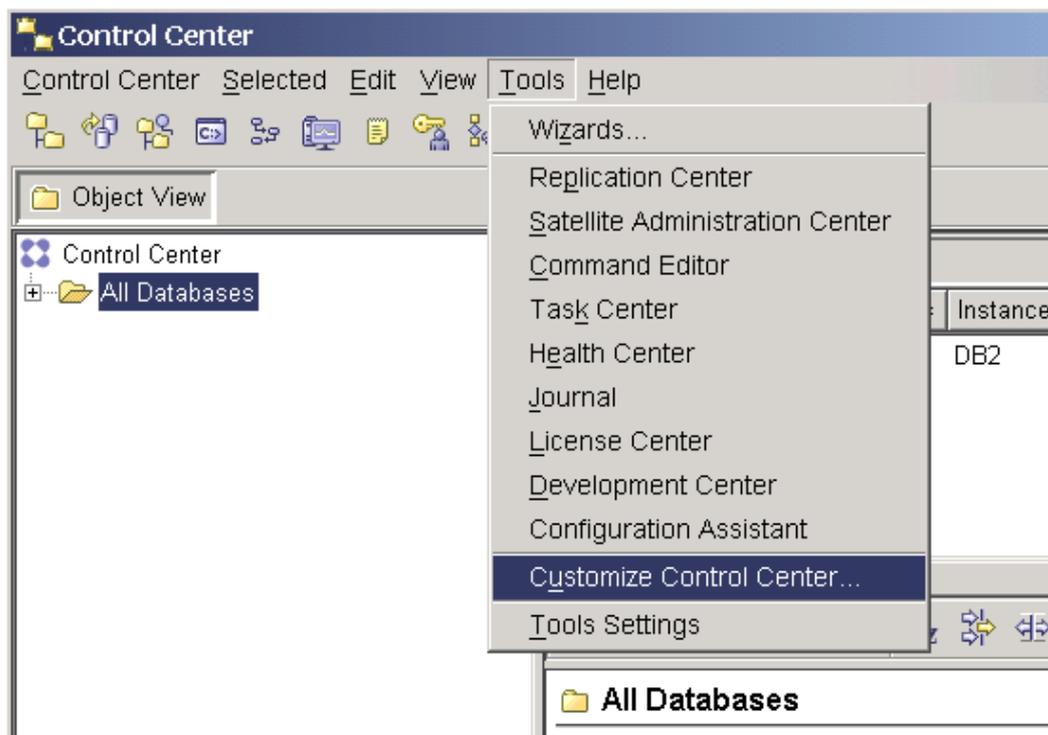


Figure 5.5 – Customizing the Control Center / Figura 5.5 - Personalizzare il Control Center

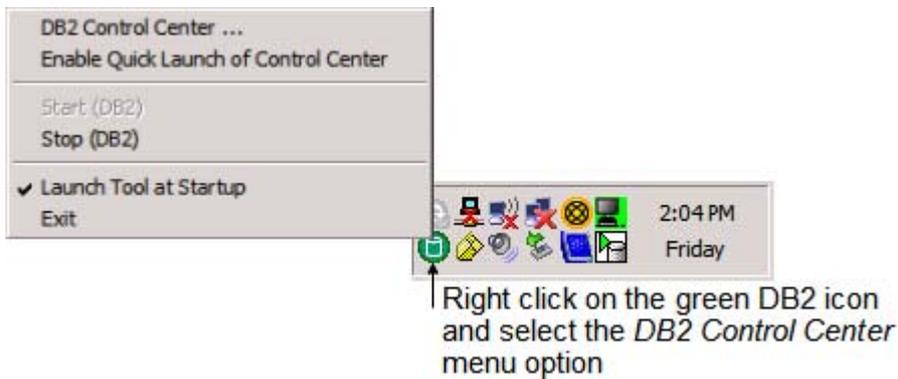
Launching the Control Center / Lanciare il Control Center

There are many ways to launch the Control Center:

- Navigating through the Windows *Start* menu
- By executing `db2cc` on a command prompt
- By clicking the Control Center icon  in the toolbar of any of the other DB2 GUI tools
- From the DB2 icon in the Windows system tray as shown in Figure 5.6 (Right click on the DB2 green icon and select the DB2 Control Center menu option)

Ci sono molti modi di lanciare il Control Center:

- Navigando all'interno dello *Start* menù di Windows
- Eseguendo il comando `db2cc` su un prompt dei comandi
- Cliccando sull'icona  del Control Center nella toolbar di alcuni degli altri tool grafici di DB
- Dall'icona DB2 system tray di Windows come mostrato in figura 5.6 (fare clic destro sull'icona verde di DB2 e selezionare l'opzione del menù DB2 Control Center)



**Figure 5.6 – Launching the DB2 Control Center from the Windows system tray /
Figura 5.6 - Lanciare il DB2 Control Center dal system tray di Windows**

5.2 Command Editor

Using the DB2 Command Editor, you can execute DB2 commands, SQL and XQuery statements, analyze the execution plan of a statement, and view or update query result sets.

Usando il DB2 Command Editor, puoi eseguire i comandi di DB2, gli statement SQL e le XQuery, analizzare il programma di esecuzione di uno statement, ed osservare o aggiornare i set di risultati delle query.

Figure 5.7 shows the Command Editor with a description of its elements.

La figura 5.7 mostra il Command Editor con una descrizione dei relativi elementi.

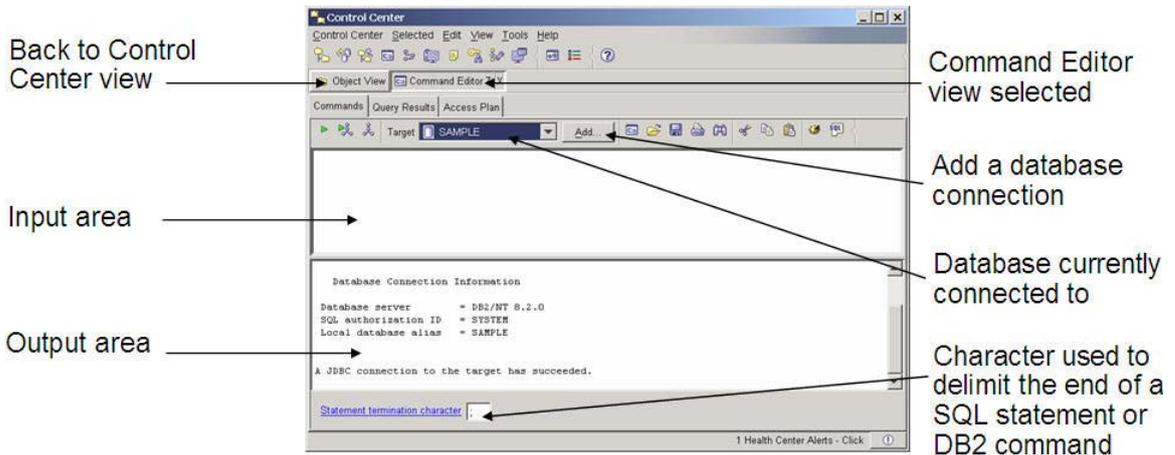


Figure 5.7 – DB2 Command Editor / Figura 5.7 - Il DB2 Command Editor

In the input area, you can input multiple statements, so long as each statement ends with a termination character. If you press the execute button (see Figure 5.8), the statements will be executed one after another. If you explicitly highlight a particular statement, only the highlighted statement will be executed. A database connection must exist in order to carry out any SQL statements, however, one of the statements can be a connect statement.

Nella zona di input, puoi immettere differenti statement, a condizione che ogni statement si concluda con un carattere di terminazione. Se premi il tasto di esecuzione (vedi figura 5.8), gli statement saranno eseguiti uno dopo l'altro. Se evidenzi esplicitamente un particolare statement, solo lo statement evidenziato sarà eseguito. Un collegamento a un database deve esistere per poter effettuare tutte gli statement SQL, tuttavia, uno degli statement può essere uno statement di collegamento.

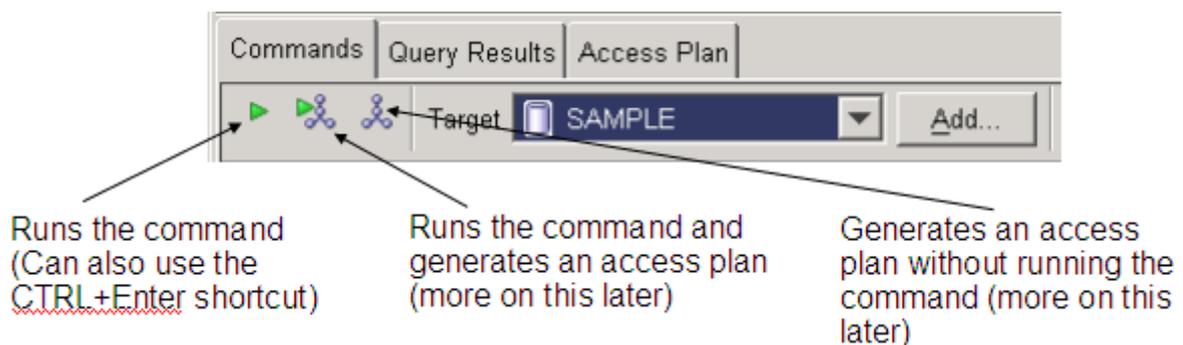


Figure 5.8 – The Command Editor – Commands tab / Figura 5.8 - Il Command Editor - Scheda dei comandi

Launching the Command Editor / Lanciare il Command Editor

You can launch the Command Editor in several ways:

- From the Windows Start Menu:
Start -> Programs -> IBM DB2 -> DB2COPY1 (Default) -> Command Line Tools -> Command Editor
- From a command prompt, type `db2ce`
- From the Tools menu in the Control Center
- Embedded within the Control Center:
 - Right click on the SAMPLE database icon in the Control Center's Object Tree pane and select the *Query* menu item
 - Any time a queryable object is selected (database, table, etc.), you can launch the Command Editor by clicking the *Query* link in the Control Center's Object Detail pane
- From the Control Center, click the Command Editor icon  on the Control Center Toolbar as shown in Figure 5.9

Puoi lanciare il Command Editor in diversi modi:

- Dallo Start menù di Windows:
Start -> Programs -> IBM DB2 -> DB2COPY1 (Default) -> Command Line Tools -> Command Editor
- Dal un prompt dei comandi, scrivendo `db2ce`
- Dal Tool menù nel Control Center
- All'interno del Control Center:
 - Facendo un clic destro sull'icona del database SAMPLE nel pannello di navigazione dell'albero degli oggetti del Control Center e selezionando l'elemento nel menù denominato *Query*
 - In qualunque momento un oggetto su cui possono essere eseguite query è selezionato (database, tabella, ecc.), puoi lanciare il Command Editor cliccando il link *Query* nel pannello Object Detail del Control Center
- Dal Control Center, cliccando sull'icona del Command Editor nella toolbar come mostrato in figura 5.9



Figure 5.9 – The Command Editor icon in Control Center / Figura 5.9 - L'icona del Command Editor nel Control Center

Adding a database connection / Aggiungere un collegamento ad un database

To add a connection to a database, click on the *Add* button (See Figure 5.7). A dialog as shown in Figure 5.10 will appear.

Per aggiungere un collegamento ad un database, clicca sul tasto *Add* (vedi figura 5.7). Comparirà una finestra di dialogo come mostrato in figura 5.10

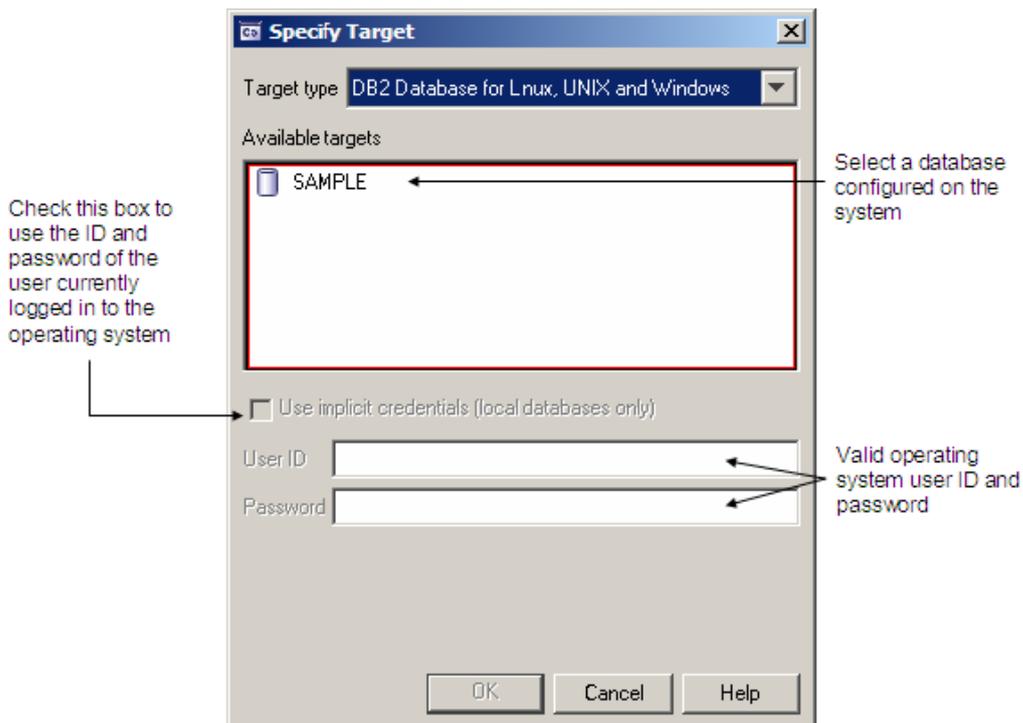


Figure 5.10 – Add a database connection / Figura 5.10 - Aggiungere un collegamento ad un database

5.3 SQL Assist Wizard

If you are not familiar with the SQL language and would like to use an assistant or wizard to generate the SQL code, the SQL Assist Wizard is available from the Command Editor to help you. As shown in Figure 5.11, you invoke it from the Command Editor by clicking on the last icon with the SQL symbol (highlighted in the figure with a red circle)

Se non hai una conoscenza del linguaggio SQL e volessi usare un assistente o un wizard per generare il codice di SQL, SQL Assist Wizard è disponibile dal Command Editor per aiutarti. Come è mostrato in figura 5.11, lo invochi dal Command Editor cliccando sull'ultima icona con il simbolo di SQL (evidenziato nella la figura con un cerchio rosso)

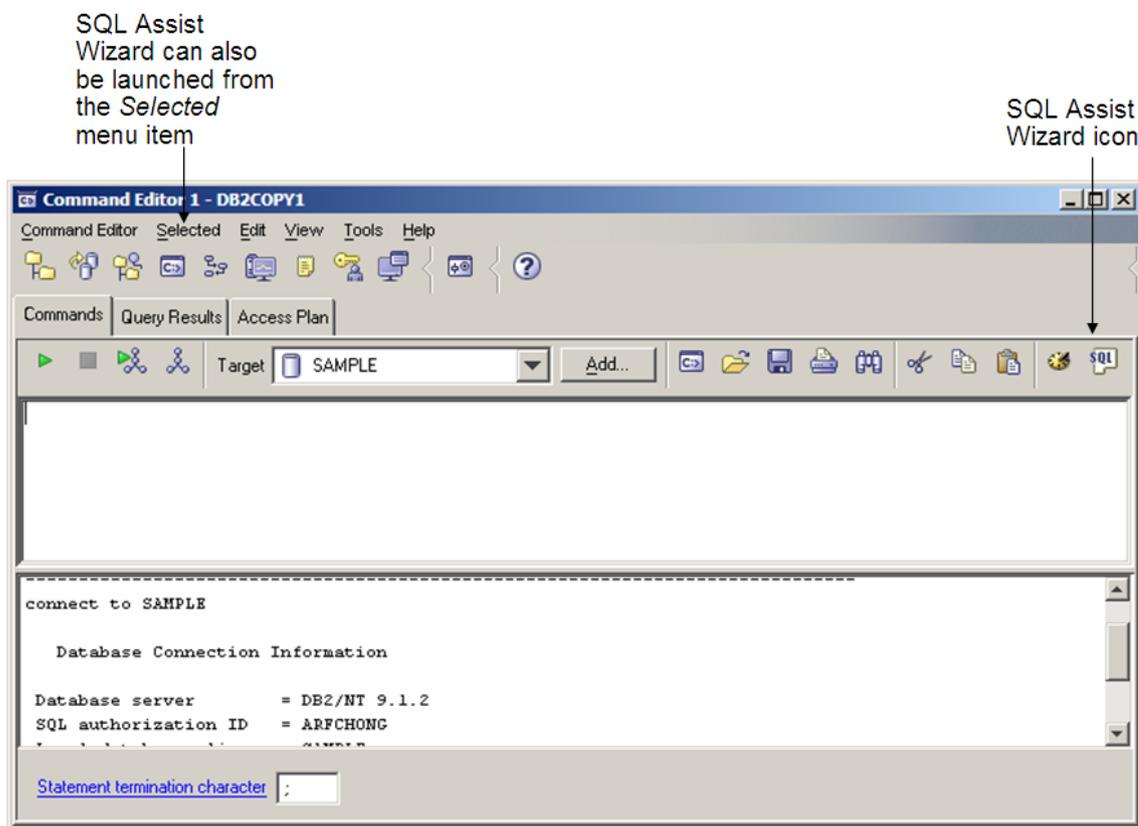


Figure 5.11 – Invoking the SQL Assist Wizard / Figura 5.11 - Invocazione del SQL Assist Wizard

Figure 5.12 shows the SQL Assist wizard. It is fairly straight forward to use. First indicate the type of SQL statement you need assistance with (SELECT, INSERT, UPDATE, DELETE).

La figura 5.12 mostra SQL Assist Wizard. È abbastanza intuitivo da usare. Prima indichi il tipo di statement SQL per cui hai bisogno di assistenza (SELECT, INSERT, UPDATE, DELETE).

Depending on which statement you choose, different options will appear. At the bottom of the window you will see how the SQL statement is constructed as you select different choices in the wizard.

DELETE). A seconda di quale statement scegli, compariranno opzioni diverse. Nella parte inferiore della finestra vedrai come lo statement SQL è costruito mentre selezioni le diverse scelte nel wizard.

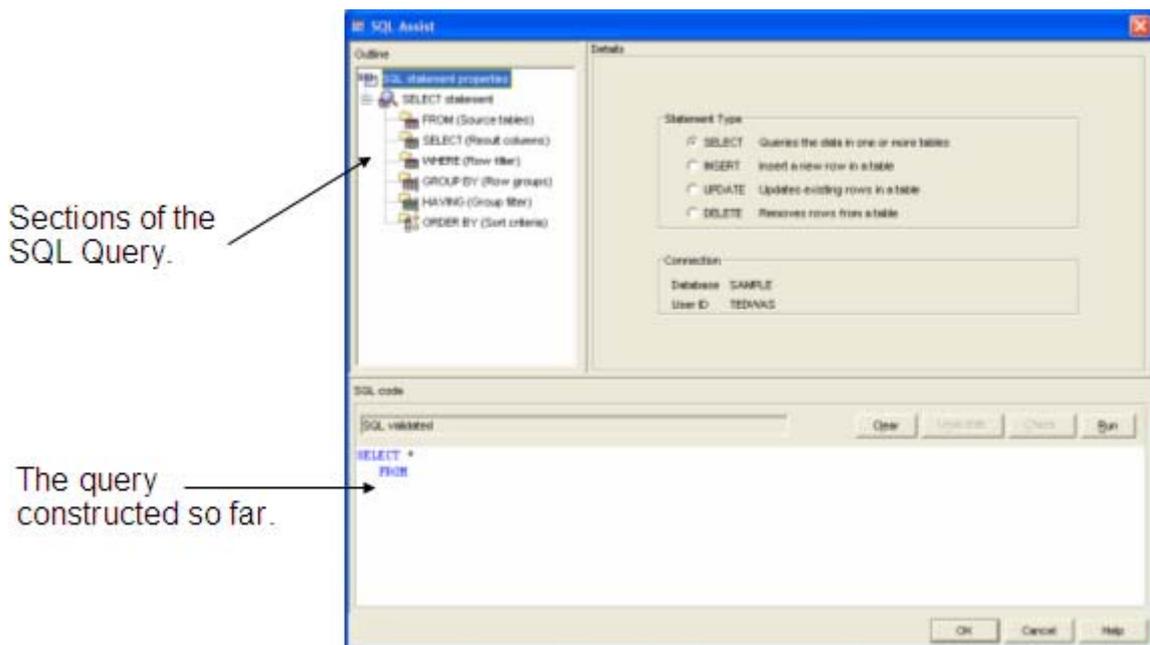


Figure 5.12 – The SQL Assist wizard / Figura 5.12 - SQL Assist wizard

5.4 Show SQL Button / Il tasto Show SQL

Most GUI tools and wizards in DB2 allow you to review the actual command or SQL statement that is created as a result of using the tool or wizard to perform an action. To see this, click on the *Show SQL* button in the tool you are working on, as shown in Figure 5.13 and Figure 5.14

La maggior parte dei GUI tool e dei wizard in DB2 permettono di verificare il comando attuale o lo statement SQL che è creato come risultato dell'uso di tool o wizard per eseguire un'azione. Per vedere questo, clicca sul tasto *Show SQL* nel tool in cui stai lavorando, come mostrato in figura 5.13 e in figura 5.14

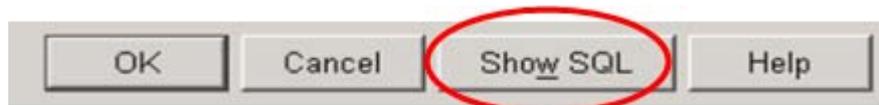


Figure 5.13 – The Show SQL button / Figura 5.13 - Il tasto show SQL

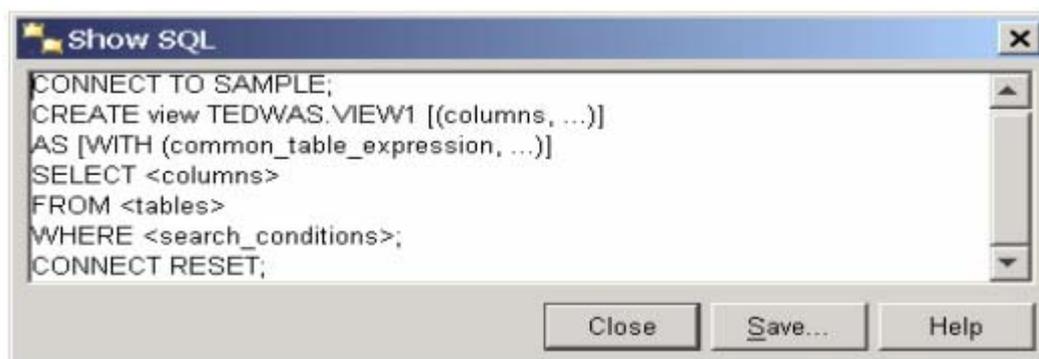


Figure 5.14 – The output of a Show SQL button / Figura 5.14 - L'output del tasto Show SQL

The ability to review the SQL statements and commands is very handy for learning SQL syntax, and for saving the commands or statements to a file for later use. You can also build scripts by reusing these generated commands and statements.

La capacità di esaminare gli statement di SQL e i comandi è molto utile per l'apprendimento della sintassi SQL e per salvare dei comandi o degli statement su un file da usare in seguito. Puoi anche sviluppare script riutilizzando questi comandi e questi statement generati.

Quicklab #4: Populating the EXPRESS Database using scripts / Popolare il database EXPRESS usando gli script

Objective

In this Quicklab, you will populate the EXPRESS database using the Command Editor and two scripts supplied.

Procedure

1. In Quicklab #2 you created the EXPRESS database; now you need to populate it with a few tables and some data. For your convenience, two scripts, called `quicklab4.db2` and `quicklab4.dat` have been created to do this for you. The `quicklab4.db2` script contains the commands used to create the tables and therefore must be run first. The `quicklab4.dat` script contains statements that insert data into the tables. Both scripts can be found in the `quicklabs` zip file accompanying this book. To run these scripts, open Command Editor. Ensure that the new database you created is selected in the drop-down list in the toolbar. If the new database does not appear in the list, add a connection to it using the *Add* button.
2. Click the *Selected* → *Open* menu on the Command Editor and navigate to the folder where the scripts are stored. Select the `quicklab4.db2`

Obiettivo

In questo Quicklab, popolerai il database EXPRESS usando il Command Editor e due script forniti.

Procedura

1. Nel Quicklab #2 hai creato il database EXPRESS; ora devi popolarlo con alcune tabelle e qualche data. Per tua convenienza, due script, chiamati `quicklab4.db2` e `quicklab4.dat` sono stati creati per far questo per te. Lo script `quicklab4.db2` contiene i comandi usati per creare le tabelle e quindi deve essere eseguito per primo. Lo script `quicklab4.dat` contiene gli statement che inseriscono i dati nelle tabelle. Entrambi gli script possono essere trovati nel file di zip dei quicklab che accompagna questo libro. Per eseguire questi script, apri il Comand Editor. Assicurati che il nuovo database che hai creato sia selezionato nel elenco drop-down nella toolbar. Se il nuovo database non compare nella lista, aggiungi un collegamento per mezzo del tasto *Add*.
2. Clicca sul menù *Selected* → *Open* sul Command Editor e scorri fino alla cartella in cui gli script sono memorizzati. Seleziona il file

file and click the *OK* button. The contents of the file should now be displayed in Command Editor's input area. Click the *Run* button to run the script. Verify that there were no errors encountered when running the script.

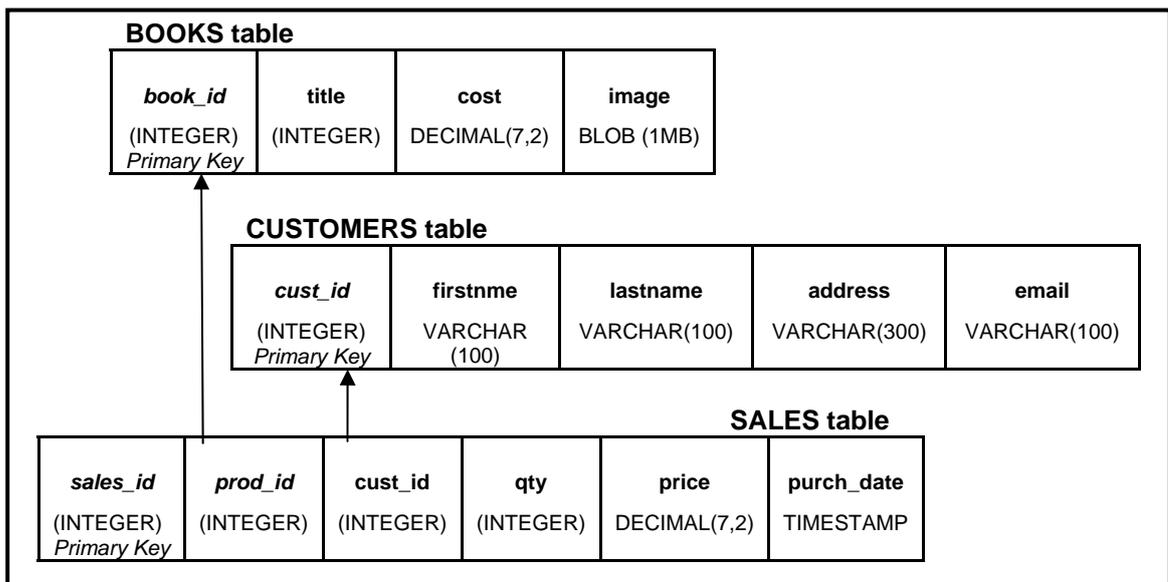
3. Repeat Step (2) for the quicklab4.dat file.

The new database you created is for a very simple Internet bookstore. The *BOOKS* table contains all the information about the books the store carries. The *CUSTOMERS* table contains information about each of the store's customers. Finally, the *SALES* table contains sales data. Whenever a customer purchases a book, a record is made in the *SALES* table. The diagram below shows the design and relationship between the tables.

quicklab4.db2 e clicca il tasto *OK*. Ora, il contenuto del file dovrebbe essere visualizzato nella zona di input del Command Editor. Clicca il tasto *Run* per eseguire lo script. Verifica che non ci siano stati errori riscontrati quando lo script veniva eseguito.

3. Ripeti dal punto (2) per il file quicklab4.dat.

Il nuovo database che avete creato è per un negozio di libri su Internet molto semplice. La tabella *BOOKS* contiene tutte le informazioni sui libri che il negozio contiene. La tabella *CUSTOMERS* contiene le informazioni su ciascuno dei clienti del negozio. Per concludere, la tabella *SALES* contiene i dati delle vendite. Ogni volta che un cliente compra un libro, viene creato un record nella tabella *SALES*. La schemata sotto mostra lo schema ed le relazioni fra le tabelle.



5.5 Scripting

It is always useful to be able to create script files that perform several DB2 commands or SQL statements repeatedly. For example, a DBA may want to run a given script every day to check the row count of important tables.

There are two general forms of scripting:

1. SQL scripts
2. Operating system (shell) scripts.

5.5.1 SQL scripts / Script SQL

SQL scripts include query statements and database commands. These scripts are relatively simple to understand and are platform independent. However, variables or parameters are not supported.

For example, the following commands are saved in a file called `script1.db2`

```
CONNECT TO EXPRESS;
CREATE TABLE user1.mytable
    ( col1 INTEGER NOT NULL,
      col2 VARCHAR(40) ,
      col3 DECIMAL(9,2) );
SELECT * FROM user1.mytable FETCH FIRST 10 ROWS ONLY;
COMMIT;
```

File `script1.db2`

In the above script, all the statements are SQL statements, and each statement is separated by a statement delimiter, in this case a semi-colon. The file name does not need to use the extension "db2". Any extension could be used.

È sempre utile poter creare script file che eseguano ripetutamente molti comandi DB2 o statement SQL. Per esempio, un DBA può volere eseguire uno certo script quotidianamente per controllare il numero di righe delle tabelle importanti.

Ci sono due forme generali di scripting:

1. Script SQL
2. Script del sistema operativo (shell).

Gli script SQL comprendono gli statement di query ed i comandi database. Questi script sono relativamente semplici da capire e sono indipendenti dalla piattaforma. Tuttavia, le variabili o i parametri non sono supportati.

Per esempio, i seguenti comandi sono salvati in un file chiamato `script1.db2`

Nello script precedente, tutti gli statement sono statement SQL, ed ogni statement è separato da un delimitatore di statement, in questo caso un punto e virgola. Il nome del file non necessita dell'estensione "db2". Qualsiasi estensione può essere usata.

Executing SQL Scripts / Esecuzione degli script SQL

An SQL script can be executed from either the Command Editor or the DB2 Command Window on Windows, or through a Linux shell. To run the previous script from the DB2 Command Window or Linux shell, you can use the following command:

```
db2 -t -v -f script1.db2 -z script1.log
```

or / o:

```
db2 -tvf script1.db2 -z script1.log
```

In this command:

- t** indicates statements use the default statement termination character (semi-colon)
- v** indicates verbose mode; causing db2 to echo the command being executed
- f** indicates that the filename specified after this flag is the script file
- z** indicates the following message filename should be used for appending screen output for later analysis (this is optional, but recommended)

When the **-t** flag is used and no line delimiter is specified, the semi-colon is assumed to be the delimiter of the statements. There may be situations where another delimiter is required. For example, a script containing SQL PL code needs to use a different statement termination character other than the default (semicolon), because semicolons are used within SQL PL object definitions to terminate procedural statements.

For example, in the script file below called "*functions.db2*", which contains the necessary Data Definition Language

Uno script SQL può essere eseguito dalla Command Editor o dalla DB2 Command Window su Windows, o attraverso una shell di Linux. Per eseguire lo script precedente dalla Command Window DB2 o dalla shell di Linux, puoi usare il seguente comando:

In questo comando:

- t** indica che gli statement usano il carattere di terminazione di statement di default (punto e virgola)
- v** indica il modo verbose; richiedendo a DB2 di stampare il comando che è eseguito
- f** indica che il nome del file specificato dopo questo flag è il file dello script
- z** indica che il nome del file del messaggio seguente dovrebbe essere usato per aggiungere l'output dello script per un'analisi successiva (questo è facoltativo, ma suggerito)

Quando il flag **-t** è usato e nessun delimitatore di riga è specificato, il punto e virgola è inteso come delimitatore degli statement. Ci possono essere situazioni dove un altro delimitatore è necessario. Per esempio, uno script che contiene codice SQL PL deve usare un carattere di termine statement differente da quello di default (punto e virgola), poiché i punti e virgola sono usati all'interno delle definizioni degli oggetti SQL PL per terminare statement procedurali.

Per esempio, nel file di script seguente, chiamato "*functions.db2*", che contiene il necessario Data Definition Language

(DDL) to create a function, a semi-colon is needed at the end of the SELECT syntax required within the function. For the CREATE FUNCTION statement terminator we are using an exclamation mark (!). If we had used a semi-colon for the statement terminator, there would have been run time conflict from the script, resulting in an error reported by DB2.

```
CREATE FUNCTION f1 ()
  SELECT ... ;
  ...
END!
```

File functions.db2

To inform DB2 that a different statement termination character is being used, use the **-d** flag, followed by the terminator character desired as shown below:

```
db2 -td! -v -f functions.db2 -z functions.log
```

The description of other flags that can be used can be obtained from the Command Window or Linux shell with the command:

```
db2 list command options
```

(DDL) per creare una funzione, un punto e virgola è necessario alla fine della SELECT richiesta all'interno della funzione. Per il terminatore di statement CREATE FUNCTION stiamo usando un punto esclamativo (!). Se avessimo usato un punto e virgola per il terminatore di statement, ci sarebbe stato un conflitto a run time dallo script, con un conseguente errore segnalato da DB2.

Per informare DB2 che un carattere di terminazione di statement differente è stato usato, usa il flag **-d**, seguito dal carattere di terminazione che desideri come mostrato sotto:

La descrizione degli altri flag che possono essere usati può essere ottenuta dalla Command Window o dalla Linux shell con il comando:

5.5.2 Operating system (shell) scripts / Script del sistema operativo (shell)

Operating system scripts provide greater flexibility and power, as they give you the possibility to add additional programming logic. They are platform dependant, but they support parameters and variables. Below is an example of a simple Windows operating system (shell) script.

Gli script del sistema operativo forniscono maggiore flessibilità e potenzialità, poiché danno la possibilità di aggiungere più logica di programmazione. Sono platform dependant, ma supportano i parametri e le variabili. Un esempio di un semplice script del sistema operativo (shell) di Windows segue.

```
set DBPATH=c:
set DBNAME=PRODEXPR
set MEMORY=25
db2 CREATE DATABASE %DBNAME% ON %DBPATH% AUTOCONFIGURE USING
    MEM_PERCENT %MEMORY% APPLY DB AND DBM
db2 CONNECT TO %DBNAME% USER %1 USING %2
del schema.log triggers.log app_objects.log
db2 set schema user1
db2 -t -v -f schema.db2 -z schema.log
db2 -td@ -v -f triggers.db2 -z triggers.log
db2 -td@ -v -f functions.db2 -z functions.log
```

File create_database.bat

To execute this operating system script from the command line, you would issue the following command on Windows:

Per eseguire questo script del sistema operativo dalla command line, su Windows, inserisci il seguente comando:

```
create_database.bat db2admin ibmdb2
```

On Windows using the “bat” extension tells the operating system this is a batch executable file.

Su Windows, usare l'estensione "bat" indica al sistema operativo che questo è un file eseguibile in batch mode.

On Linux, you need to change the mode on the file to indicate the file is executable using a command like `chmod +x`. Afterwards, you can run it in the same manner as listed above.

Su Linux, devi cambiare l'accesso al file per indicare che il file è eseguibile, usando un comando come `chmod +x`. In seguito, puoi eseguirlo nello stesso modo come spiegato in precedenza.

Quicklab #5: Create an installation script for EXPRESS Database / Creare uno script di installazione per il database EXPRESS

Objective

Scripts are a powerful mechanism for performing repetitive tasks such as database statistic collection, backups, and database deployment. Operating system scripts have the advantage of supporting script parameters, making them more flexible. In this Quicklab, you will create an operating system script to deploy the *EXPRESS* database as the *EXPRESS2* database. The script will call the previously generated SQL scripts for database objects. In order to save space, this quicklab shows the scripts and commands specific to the Windows platform. If you prefer to work on Linux, ensure to make the appropriate changes to the instructions below.

Procedure

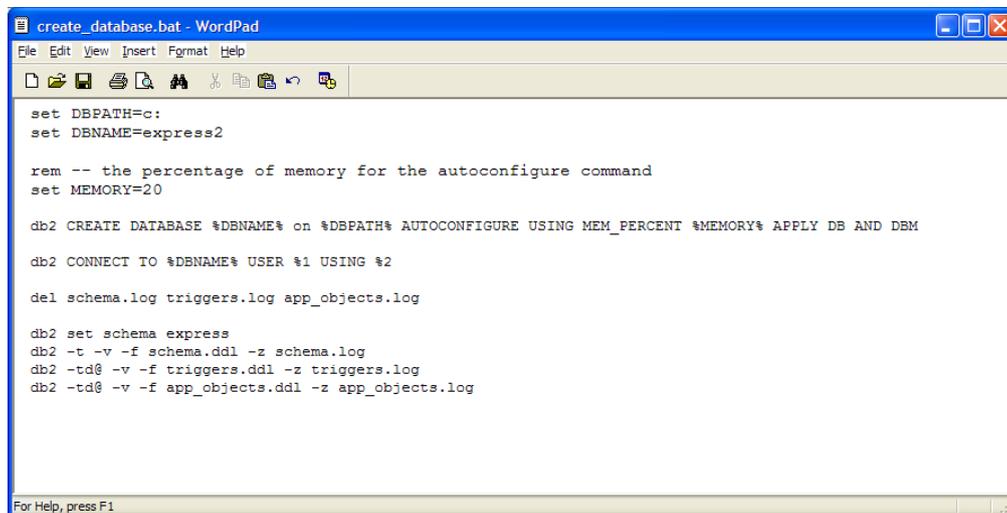
1. Open a text editor, such as Notepad or Wordpad and input the information as shown below.

Obiettivo

Gli script sono un meccanismo potente per eseguire task ripetuti come la raccolta di statistiche del database, i backup e il deployment del database. Gli script del sistema operativo hanno il vantaggio del supporto dei parametri di script, che li rende più flessibili. In questo Quicklab, creerai uno script del sistema operativo per fare il deploy del database *EXPRESS* come database *EXPRESS2*. Lo script chiama gli script SQL generati precedentemente per i database object. Per risparmiare spazio, questo quicklab mostra gli script ed i comandi specifici della piattaforma Windows. Se preferisci lavorare su Linux, assicurati di fare le modifiche necessarie alle istruzioni che seguono.

Procedura

1. Apri un editor testuale, come Notepad o Wordpad ed inserisci le informazioni come mostrato sotto.



```
create_database.bat - WordPad
File Edit View Insert Format Help
set DBPATH=c:
set DBNAME=express2

rem -- the percentage of memory for the autoconfigure command
set MEMORY=20

db2 CREATE DATABASE %DBNAME% on %DBPATH% AUTOCONFIGURE USING MEM_PERCENT %MEMORY% APPLY DB AND DBM

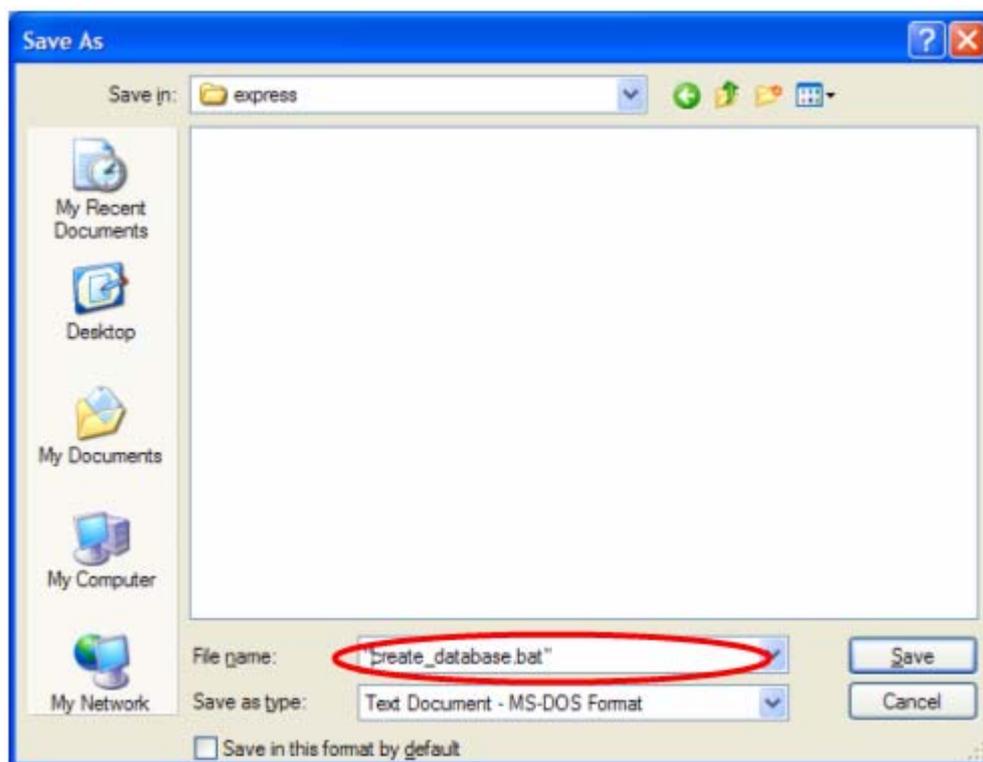
db2 CONNECT TO %DBNAME% USER %1 USING %2

del schema.log triggers.log app_objects.log

db2 set schema express
db2 -t -v -f schema.ddl -z schema.log
db2 -td@ -v -f triggers.ddl -z triggers.log
db2 -td@ -v -f app_objects.ddl -z app_objects.log

For Help, press F1
```

- 2 Save the script file in a directory and call it `create_database.bat`. In the *Save As* window, ensure you choose the *MS-DOS Format* option (in Wordpad). If you save the file with a different format, Wordpad may introduce invisible characters which may cause problems in the execution of the script. Also, put quotes around the file name to ensure that Windows does not append a `.TXT` extension to it as shown in the figure below.
2. Salva il file dello script in una directory e chiamarlo: `create_database.bat`. Nella finestra di dialogo *Save As*, assicurati di scegliere l'opzione di *Formato MS-DOS* (in Wordpad). Se conservi il file in un formato diverso, Wordpad può introdurre caratteri invisibili che possono causare problemi nell'esecuzione dello script. Inoltre, metti degli apici intorno al nome per assicurarti che Windows non aggiunga un'estensione di `.TXT` come mostrato nella figura che segue.

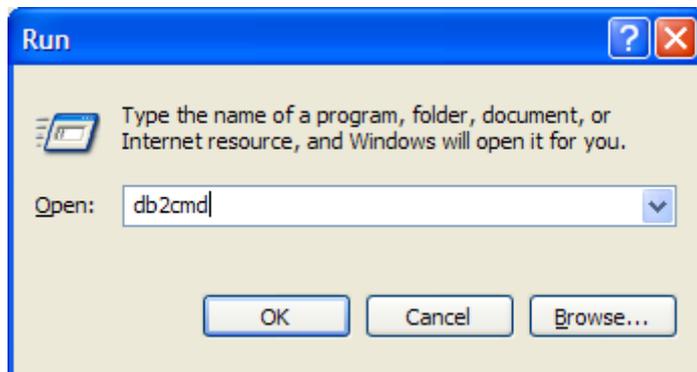


3. To run scripts that interact with DB2, you must have a DB2 command line environment. To open a DB2 Command Window, go to *Start > Program Files > IBM DB2 > DB2COPY1 (default) > Command Line Tools > Command Window.*

Alternatively, you can use *Start > Run*, type `db2cmd` and press *Enter* as shown below.

3. Per eseguire gli script che interagiscono con DB2, devi avere un ambiente command line di DB2. Per aprire una DB2 Command Window, vai su: *Start > Program Files > IBM DB2 > DB2COPY1 (default) > Command Line Tools > Command Window.*

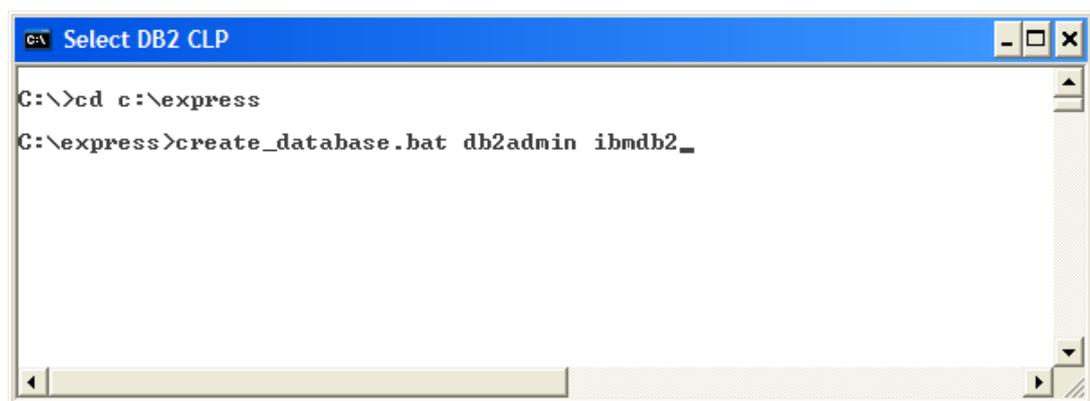
Alternativamente, puoi usare *Start > Run*, digitare `db2cmd` e premere *Enter* come indicato sotto.



4. Then to run the script, enter the commands:

```
cd C:\express
create_database.bat db2admin ibmdb2
```

4. Poi per eseguire lo script, inserisci i comandi:



5. Take a moment to familiarize yourself with the script you just created. Do you understand what is happening on each line?
6. Try to answer the following questions:

- a. Where is the database connection established?
- b. What do the %1 and %2 mean?

5. Prenditi un momento per familiarizzare con lo script che hai appena creato. Capisci che cosa sta accadendo su ogni linea?
6. Prova a rispondere alle seguenti domande:

- a) Dov'è stabilito il collegamento al database?
- b) Che cosa significano %1 e %2?

- c. What does the following line of code do? Where is it used? For what?
`SET DBPATH=C:`
- d. What does the following line of code do?
`del schema.log,
triggers.log,
app_objects.log`
- e. What happens when the script isn't called with any parameters?
- f. Why don't the SQL scripts being called contain `CONNECT TO` statements? How do they connect to the database?
- c) Che cosa fa la seguente linea di codice? Dove è usata? Per che cosa?
`SET DBPATH=C:`
- d) Cosa fa la seguente linea di codice?
`del schema.log,
triggers.log,
app_objects.log`
- e) Cosa accade quando lo script non è chiamato con alcuni parametri?
- f) Perché gli script di SQL chiamati contengono gli statement `CONNECT TO`? Come si collegano al database?

5.6 Task Center

The Task Center GUI tool allows you to create tasks: a set of operations such as running DB2 commands, operating system commands, or scripts.

Subsequent actions can be performed if the task fails or succeeds. For example, if a task which involves backing up an important database at 3:00am in the morning is successful, an email can be sent to the DBA to provide this information. On the other hand, if the backup task fails, the Task Center can page the DBA. Figure 5.15 shows the Task Center.

Il GUI tool Task Center ti permette di creare task: un insieme di operazioni come eseguire: comandi DB2, comandi del sistema operativo, o script. Le azioni successive possono essere eseguite se l'operazione finisce con un successo o con un fallimento. Per esempio, se un'operazione che coinvolge il backup di un database importante alle 3 di mattina ha successo, un e-mail può essere inviata al DBA per fornirgli queste informazioni. D'altra parte, se l'operazione di backup fallisce, il Task Center può far chiamare il DBA. La figura 5.15 mostra il Task Center.

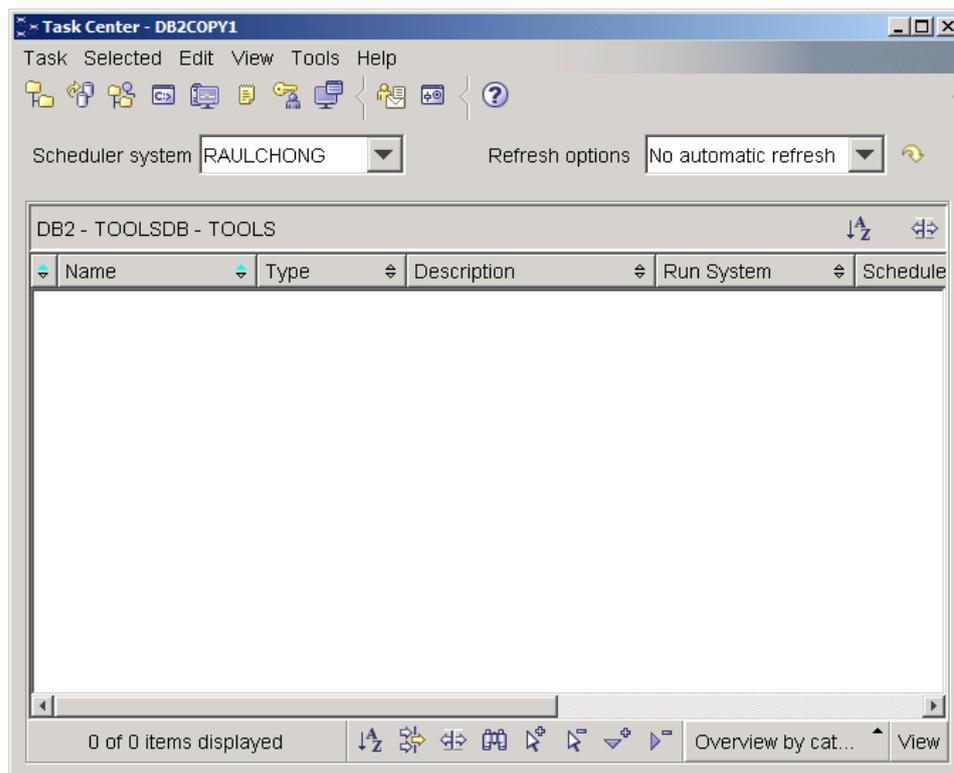


Figure 5.15 – The Task Center / Figura 5.15 - Il Task Center

5.6.1 The Tools Catalog database / Il database Tools Catalog

All the details about your tasks and task scheduling are stored in a separate DB2 database called the Tools Catalog database. This database must already exist in order to schedule tasks. To create a Tools Catalog database you can use this command:

Tutti i particolari sui tuoi task e la programmazione dei tuoi task sono memorizzati in un database DB2 separato chiamato Tools Catalog database. Questo database deve già esistere per programmare i task. Per creare un Tools Catalog database puoi usare questo comando:

```
CREATE TOOLS CATALOG systools CREATE NEW DATABASE toolsdb
```

In the above example, *systools* is the schema name of all tables in the database, and the database name is *toolsdb*. We will talk more about schemas in Chapter 8, Working with database objects.

Nell'esempio precedente, *systools* è il nome dello schema di tutte le tabelle nel database ed il nome del database è *toolsdb*. Parleremo di più degli schemi nel capitolo 8, Lavorare con gli oggetti di database.

Launching the Task Center / Lanciare il Task Center

You can launch the Task Center from the Control Center by clicking on *Tools > Task Center*, as shown in Figure 5.16. Alternatively, you can start this tool from the Windows Start menu: *Start > Programs > IBM DB2 > DB2COPY1 > General Administration Tools > Task Center*

Puoi lanciare il Task Center dal Control Center cliccando su *Tools > Task Center*, come mostrato in figura 5.16. Alternativamente, puoi avviare questo tool dal Windows Start menù: *Start > Programs > IBM DB2 > DB2COPY1 > General Administration Tools > Task Center*

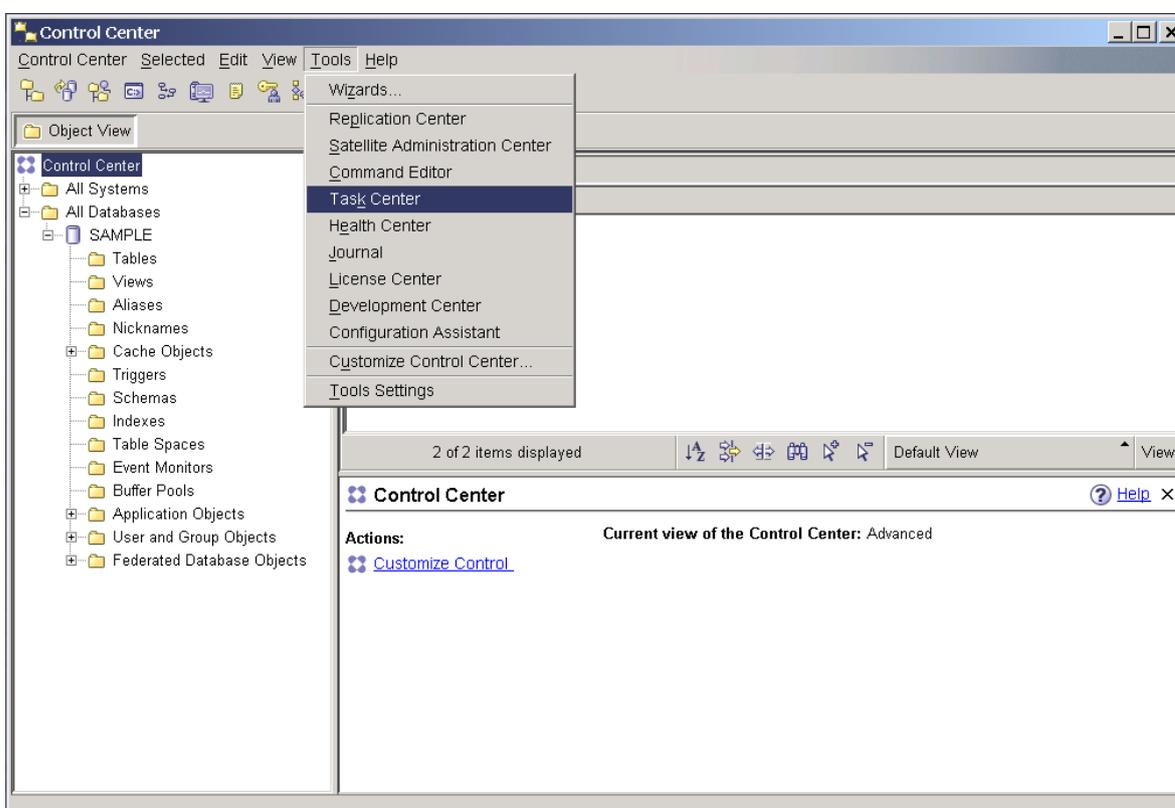


Figure 5.16 – Launching the Task Center / Figura 5.16 Lanciare il Task Center

Scheduling with the Task Center / Schedulare con il Task Center

Any type of script can be scheduled using the Task Center (whether or not it was created through a DB2 GUI tool). Tasks are run at their scheduled time from the system where you created the tools catalog database. We encourage

Qualunque tipo di script può essere programmato usando il Task Center (sia che questo fosse stato creato o meno tramite un GUI tool di DB2). I task sono eseguiti all'ora programmata dal sistema in cui hai creato il database

you to explore the Task Center yourself. Creating a task is straightforward.

tools catalog. Ti incoraggiamo ad esplorare il Task Center da solo. La creazione di una task è facile.

5.7 Journal

The Journal GUI tool provides a DBA with a journal of activities in online form. Figure 5.17 shows the Journal and Table 5.2 shows the information you can obtain from the Journal.

Il GUI tool Journal fornisce ad un DBA una registro delle attività nella forma online. La figura 5.17 mostra il Journal e la Tabella 5.2 mostra le informazioni che puoi ottenere dal Journal.

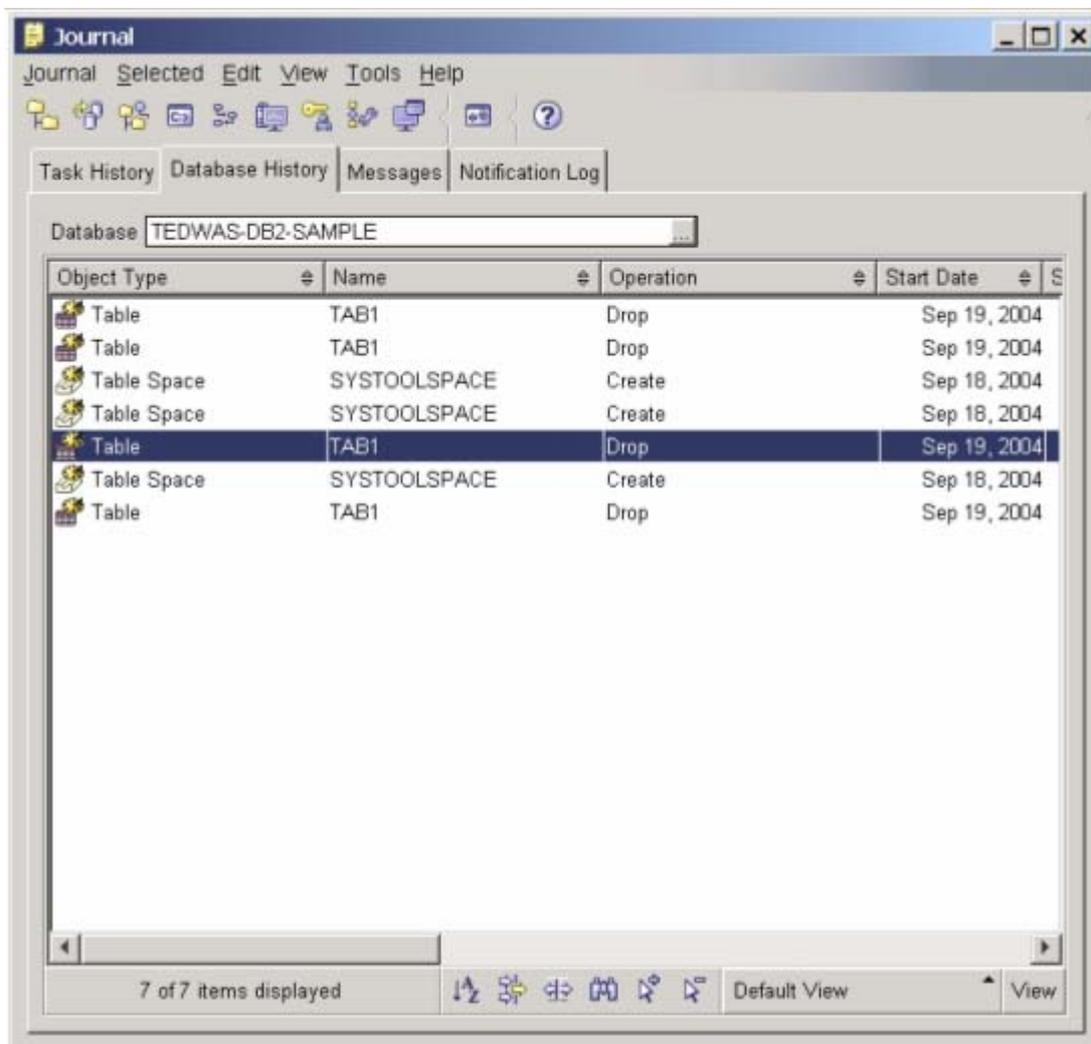


Figure 5.17 –The Journal / Figura 5.17 - Il Journal

| Type of Information / Tipo di informazioni | Description | Descrizione |
|--|--|---|
| Task History | All attempted scheduled tasks and their success status | Mostra tutti i task tentati e programmati e il loro stato di successo o meno |
| Database History | A record of database activities performed (backup, restore, REORGs, etc.) | Un record delle attività del database effettuate (backup, restore, REORG, ecc.) |
| Message | History of messages returned by DB2 tools. This is useful if you want to recall and compare old error messages, or if you close a dialog box too quickly or by accident. | Mostra la storia dei messaggi restituiti dai tool DB2. Ciò è utile se vuoi ricordare e confrontare i vecchi messaggi di errore, o se chiudi troppo rapidamente o accidentalmente una finestra di dialogo. |
| Notification Log | Contains system-level message. Critical errors are recorded here | Contiene i messaggi a livello di sistema. Gli errori critici sono registrati qui. |

Table 5.2 - Information provided in the Journal / Tabella 5.2 - Informazioni fornite nel Journal

Launching the Journal / Lanciare il Journal

You can launch the Journal from the Control Center by clicking on *Tools > Journal*, as shown in Figure 5.18. Alternatively, you can start this tool from the Windows Start menu: *Start > Programs > IBM DB2 > DB2COPY1 > General Administration Tools > Journal*

Puoi lanciare il Journal dal Control Center cliccando sui *Tools > Journal*, come mostrato in figura 5.18. Alternativamente, puoi avviare questo tool dal Windows Start menù: *Start > Programs > IBM DB2 > DB2COPY1 > General Administration Tools > Journal*

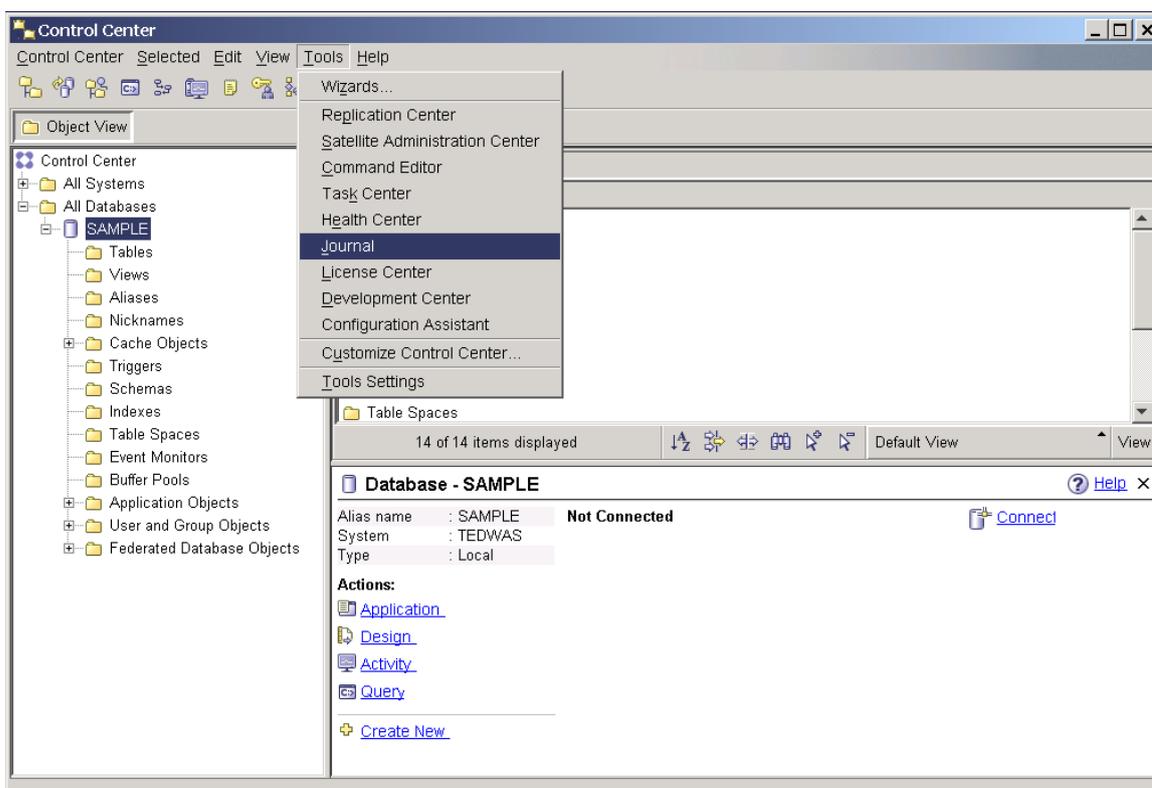


Figure 5.18 – Launching the Journal / Figura 5.18 - Lanciare il Journal

5.8 Health Monitor

The Health Monitor is a default agent that runs within the DB2 Engine, monitoring all aspects of database health (memory, space management, automated activities previously defined, etc.). When some aspect of DB2 is operating outside of the set parameters, an exception is raised and brought to the attention of the DBA. There are three types of alert states:

- Attention: a non-normal state
- Warning: a non-critical state that does not require immediate attention but may indicate a non-optimal system
- Alarm: a critical condition requiring immediate action

The Health Monitor can be turned on or off using the database manager configuration parameter HEALTH_MON.

5.8.1 Health Center

The Health Center is a graphical tool for interacting with the Health Monitor. The Health Center breaks down health alerts on a system by instance, database, and table space levels. Figure 5.19 shows the Health Center.

L'Health Monitor è un agente di default che è eseguito all'interno del motore DB2, controllando tutti gli aspetti di salute del database (memoria, amministrazione di spazio, attività automatizzate precedentemente definite, ecc.). Quando qualche espressione di DB2 è eseguita all'esterno dei parametri settati, un'eccezione è sollevata e portata all'attenzione del DBA. Ci sono tre tipi di stati di alert:

- Attenzione: un stato non-normale
- Avvertimento: uno stato non critico che non richiede l'attenzione immediata ma può indicare un sistema non-ottimale
- Allarme: una condizione critica che richiede un'azione immediata

L'Health Monitor può essere acceso o spento usando il parametro di configurazione HEALTH_MON.

L'Health Center è un tool grafico per l'interazione con l'Health Monitor. L'Health Center intercetta gli alert di salute su un sistema a livello di instance, database e spazi di tabella. La figura 5.19 mostra il Health Center.

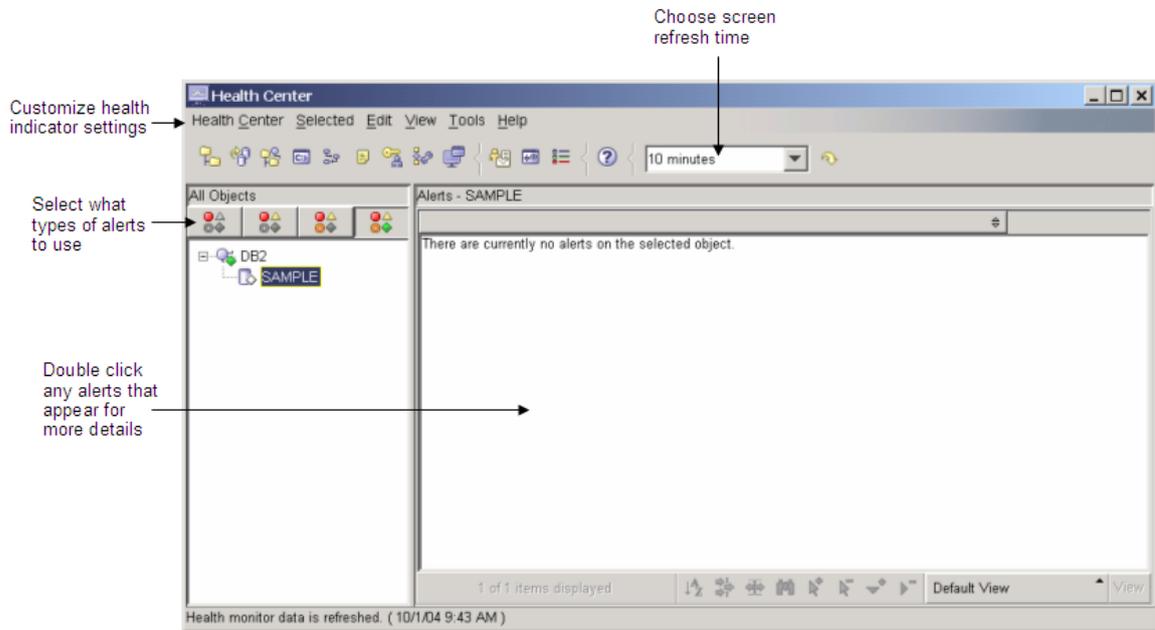


Figure 5.19 – The Health Center / Figura 5.19 - L'Health Center

Launching the Health Center / Lanciare l'Health Center

You can launch the Health Center from the Control Center by clicking the Tools Menu and choosing Health Center. This is shown in Figure 5.20. You can also start this tool from *Start > Programs > IBM DB2 > DB2COPY1 >Monitoring Tools > Health Center*

Puoi lanciare l'Health Center dal Control Center cliccando il Tool Menu e scegliendo l'Health Center. Ciò è indicato nella figura 5.20. Potete anche avviare questo tool da *Start > Programs > IBM DB2 > DB2COPY1 >Monitoring Tools > Health Center*

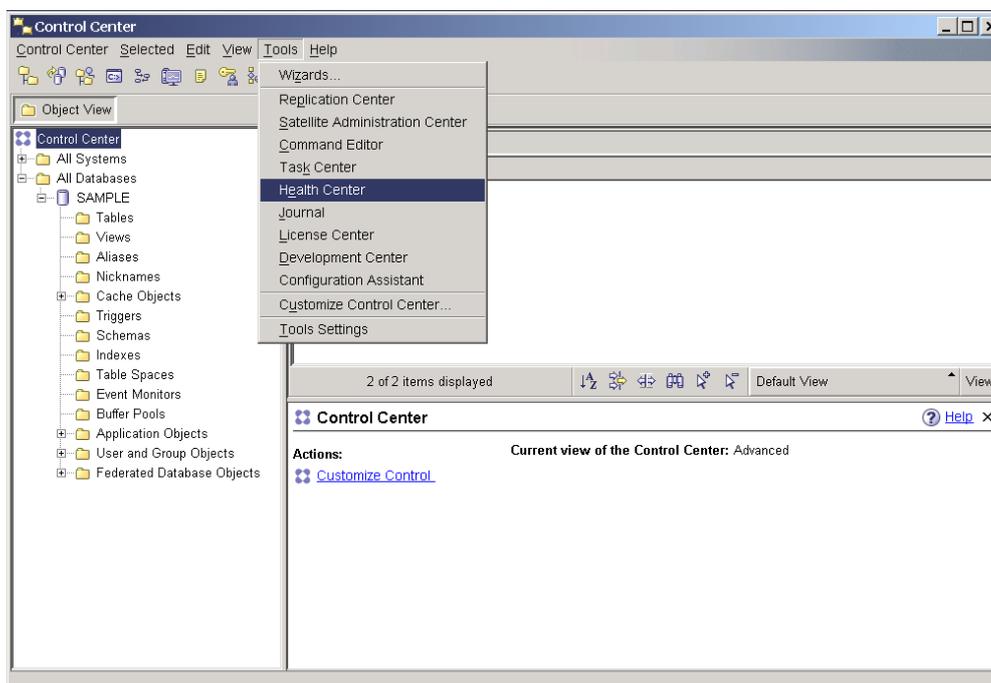


Figure 5.20 – Launching the Health Center / Figura 5.20 - Lanciare l'Health Center

Configuring Health Alert Notification / Configurazione del Health Alert Notification

Once your Health Center is started, you can configure the Alert notification by clicking on the *Health Center menu > Configure > Alert Notification* as shown in Figure 5.21. Alert notification allows you to input contact names with email addresses or pager numbers of the people who will be contacted if an alert is raised.

Una volta che l'Health Center è partito, puoi configurare la notifica degli alert cliccando *Health Center menu > Configure > Alert Notification* come mostrato in figura 5.21. L'Alert Notification ti permette di immettere i nomi dei contatti, con indirizzi e-mail o numeri di "cerca-persone", che saranno contattate se un alert verrà lanciato.

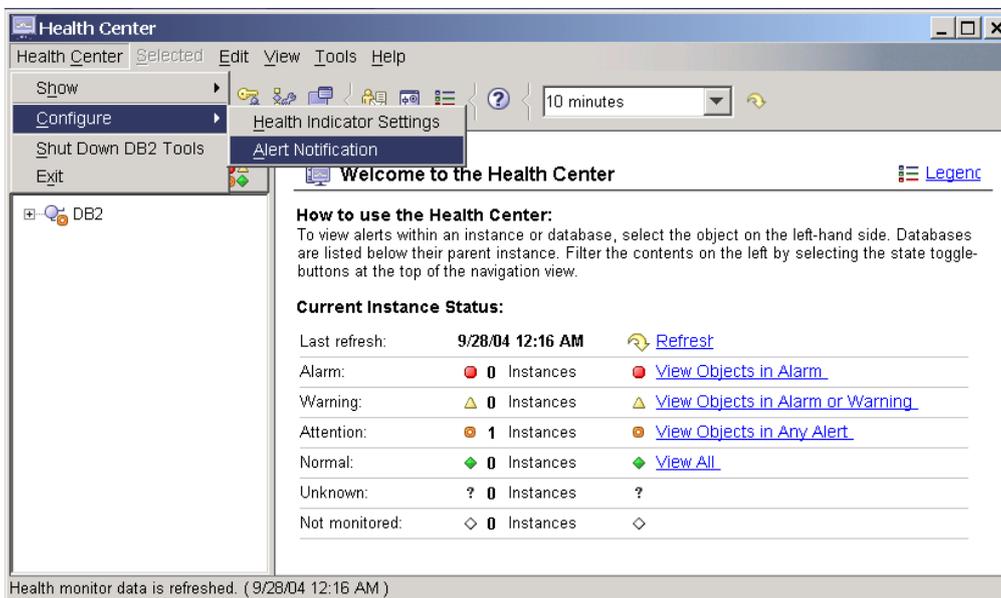


Figure 5.21 – Alert Notification / Figura 5.21 - Alert Notification

**PART II – LEARNING DB2:
DATABASE ADMINISTRATION /
PARTE II – IMPARARE DB2:
AMMINISTRAZIONE DEL
DATABASE**

6

Chapter 6 – DB2 Architecture / Capitolo 6 – L'architettura DB2

In this chapter we briefly discuss the DB2 architecture:

- The DB2 process model
- The DB2 memory model
- The DB2 storage model

In questo capitolo discuteremo brevemente l'architettura DB2:

- Il modello di processo in DB2
- Il modello di memoria di DB2
- Il modello di storage di DB2

Note:

For more information about the DB2 architecture, watch this video:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4482>

Nota:

Per ulteriori informazioni sull'architettura DB2, guardi questo video:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4482>

6.1 DB2 process model / Modello di processo in DB2

Figure 6.1 depicts the DB2 Process Model. In this figure, rectangles represent processes while ellipses represent threads. The main DB2 process is called *db2sysc*. Under this process there are several threads, the main one is also called the *db2sysc*. This is the main thread that spawns other threads. When a remote application tries to connect to the server using an SQL CONNECT statement, the remote listeners for the communication protocol will receive this request and contact a DB2 coordinator agent (*db2agent*).

La figura 6.1 descrive il modello di processo in DB2. Nella figura, i rettangoli rappresentano i processi mentre le ellissi rappresentano i thread. Il processo principale DB2 è chiamato *db2sysc*. Al di sotto di questo processo ci sono parecchi thread, quello principale si chiama anche il *db2sysc*. Questo è il thread principale che genera altri thread. Quando un'applicazione remota prova a collegarsi al server usando uno statement SQL CONNECT, i listener remoti per il protocollo della comunicazione riceveranno questa richiesta e si metteranno in contatto con un agente coordinatore DB2 (*db2agent*).

A DB2 agent is like a little worker that performs operations on behalf of DB2. When the application is local, that is, running on the same server as DB2, the steps are very similar, only that a *db2ipccm* agent handles the request instead of the *db2tpcm* thread. In some cases, such as when parallelism is enabled, a *db2agent* may spawn other agents which appear as *db2agntp* threads. Other agents shown in the figure such as *db2pfchr*, *db2loggr*, *db2dlock* may also be used for different purposes. Most common processes are described in Table 6.1, and most common threads are described in Table 6.2.

Un agente DB2 è come un piccolo operaio che esegue le operazioni di gestione a nome di DB2. Quando l'applicazione è locale, cioè, viene eseguita sullo stesso server di DB2, i passi eseguiti sono molto simili, solo che un agente *db2ipccm* tratta la richiesta anziché il thread *db2tpcm*. In alcuni casi, ad esempio quando il parallelismo è abilitato, un *db2agent* può generare altri agenti che compaiono come thread *db2agntp*. Altri agenti mostrati nella figura come *db2pfchr*, *db2loggr*, *db2dlock* possono anche essere usati per scopi differenti. I processi più comuni sono descritti nella tabella 6.1 ed i thread più comuni sono descritti nella tabella 6.2.

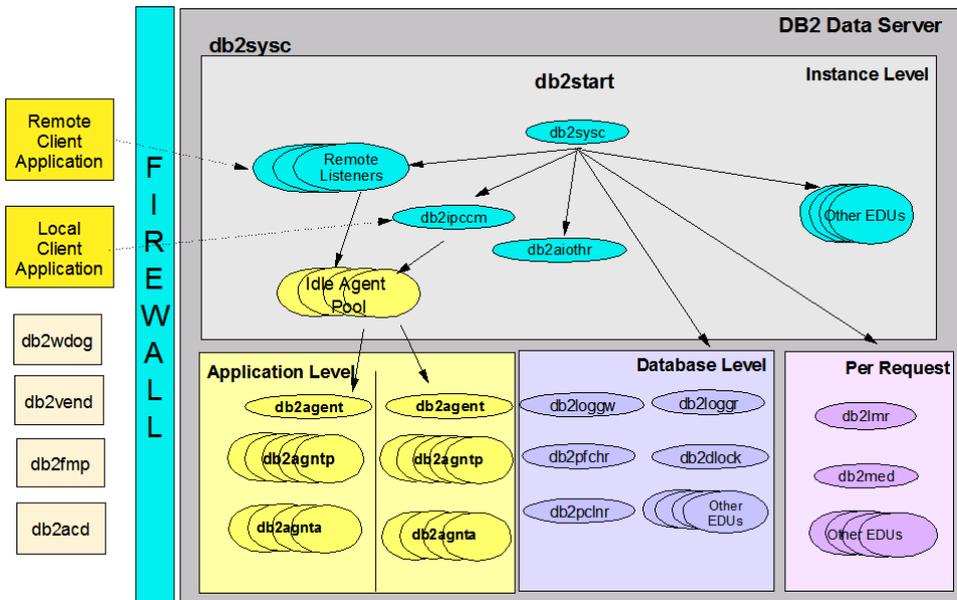


Figure 6.1 – The DB2 Process Model / Figura 6.1 - Il modello processo in DB2

| Process Name / Nome del processo | Description | Descrizione |
|----------------------------------|---|--|
| db2sysc (Linux) | The main DB2 system controller or engine. In DB2 9.5, there is only one | Il controllore di sistema o il motore principale di DB2. In DB2 9.5, c'è |

| | | |
|----------------|--|--|
| db2syscs (Win) | multi-threaded main engine process for the entire partition. All Engine Dispatchable Units (EDUs) are threads inside this process. Without this process, the database server cannot function. | soltanto un processo multi-threaded del motore principale per l'intera partizione. Tutti le Engine Dispatchable Units (EDUs) sono thread all'interno di questo processo. Senza questo processo, il server database non può funzionare. |
| db2acd | The autonomic computing daemon. It is used to perform client side automatic tasks, such as health monitor, automatic maintenance utilities, and the admin scheduler. This process was formerly called <i>db2hmon</i> . | Il demone di calcolo autonomo. È usato per eseguire task automatici lato client, come l'health monitor, programmi di manutenzione automatica, e lo scheduler admin. Questo processo precedentemente era chiamato <i>db2hmon</i> . |
| db2wdog | The DB2 watchdog. The watchdog is the parent of the main engine process, <i>db2sysc</i> . It cleans up resources if the <i>db2sysc</i> process abnormally terminates. | Il watchdog di DB2. Il watchdog è il genitore del processo principale del motore, <i>db2sysc</i> . Pulisce le risorse se il processo <i>db2sysc</i> termina in modo anomalo. |
| db2vend | The fenced vendor process introduced in DB2 9.5 All 3 rd party vendor code runs in this process outside of the engine. 3 rd party vendor code are non-IBM programs that can interact with DB2; for example, log archiving can be managed by a 3 rd party vendor code by specifying a user exit routine parameter to point to this code. | Il processo fenced vendor introdotto in DB2 9. Tutto il codice di terze parti viene eseguito in questo processo all'esterno del motore. I codici di terze parti sono programmi non-IBM che possono interagire con DB2; per esempio, l'archiviazione del log può essere diretta da un codice di un'azienda esterna specificando un parametro user exit routine per puntare a questo codice. |
| db2fmp | Fenced processes that run user code on the server outside the firewall for both stored procedures and user defined functions. This process replaces both the <i>db2udf</i> and <i>db2dari</i> processes that were used in previous versions of DB2. | Processi fenced che eseguono il codice dell'utente sul server fuori del firewall sia per le stored procedure che per le funzioni definite dagli utenti. Questo processo sostituisce sia i <i>db2udf</i> che i processi di <i>db2dari</i> che erano usati nelle versioni precedenti di DB2. |

Table 6.1 – Common DB2 processes / Tabella 6.1 - Processi comuni DB2

| Thread Name | Description | Descrizione |
|-------------|--|--|
| db2sysc | The system controller thread. This thread is responsible for the start-up and shut-down and the management of the running instance | Il thread del controllore di sistema. Questo thread è responsabile per lo start-up, lo shut-down e per l'amministrazione dell'instance che si sta eseguendo. |
| db2tccpm | TCP/IP communication listener | Listener di comunicazione del TCP/IP |
| db2agent | Coordinator agent that performs database operations on behalf of applications (at least 1 per connection, depending if Connection Concentrator is enabled). | Agente coordinatore che esegue operazioni sul database a nome delle applicazioni (almeno 1 per collegamento, dipende se il Connection Concentrator è abilitato). |
| db2agntp | Active subagent spawned if INTRA_PARALLEL is set to YES. Will perform database operations for the application. <i>db2agent</i> will coordinate the work between the different <i>db2agntp</i> subagents. | Subagente attivo generato se INTRA_PARALLEL è impostato a YES. Eseguirà operazioni sul database per l'applicazione. <i>db2agent</i> coordinerà il lavoro fra i diversi subagenti <i>db2agntp</i> . |
| db2pfchr | DB2 asynchronous I/O data prefetcher (NUM_IOSERVERS) | I/O data prefetcher DB2 asincrono (NUM_IOSERVERS) |
| db2pclnr | DB2 asynchronous I/O data writer (NUM_IOCLEANERS) | I/O data writer DB2 asincrono (NUM_IOCLEANERS) |

Table 6.2 – Common DB2 threads / Tabella 6.2 - Thread DB2 comuni

6.2 DB2 memory model / Modello di memoria di DB2

The DB2 memory model consists of different areas in memory at the instance level, database level and application and agent level as shown in Figure 6.2. We will not explain in detail each of the different areas in memory in this book, but just provide a brief overview.

Il modello di memoria DB2 consiste di zone differenti nella memoria al livello dell'instance, al livello del database e al livello dell'applicazione e dell'agente come mostrato in figura 6.2. In questo libro, non spiegheremo in dettaglio ciascuna delle zone differenti della memoria, ma forniremo soltanto una breve descrizione.

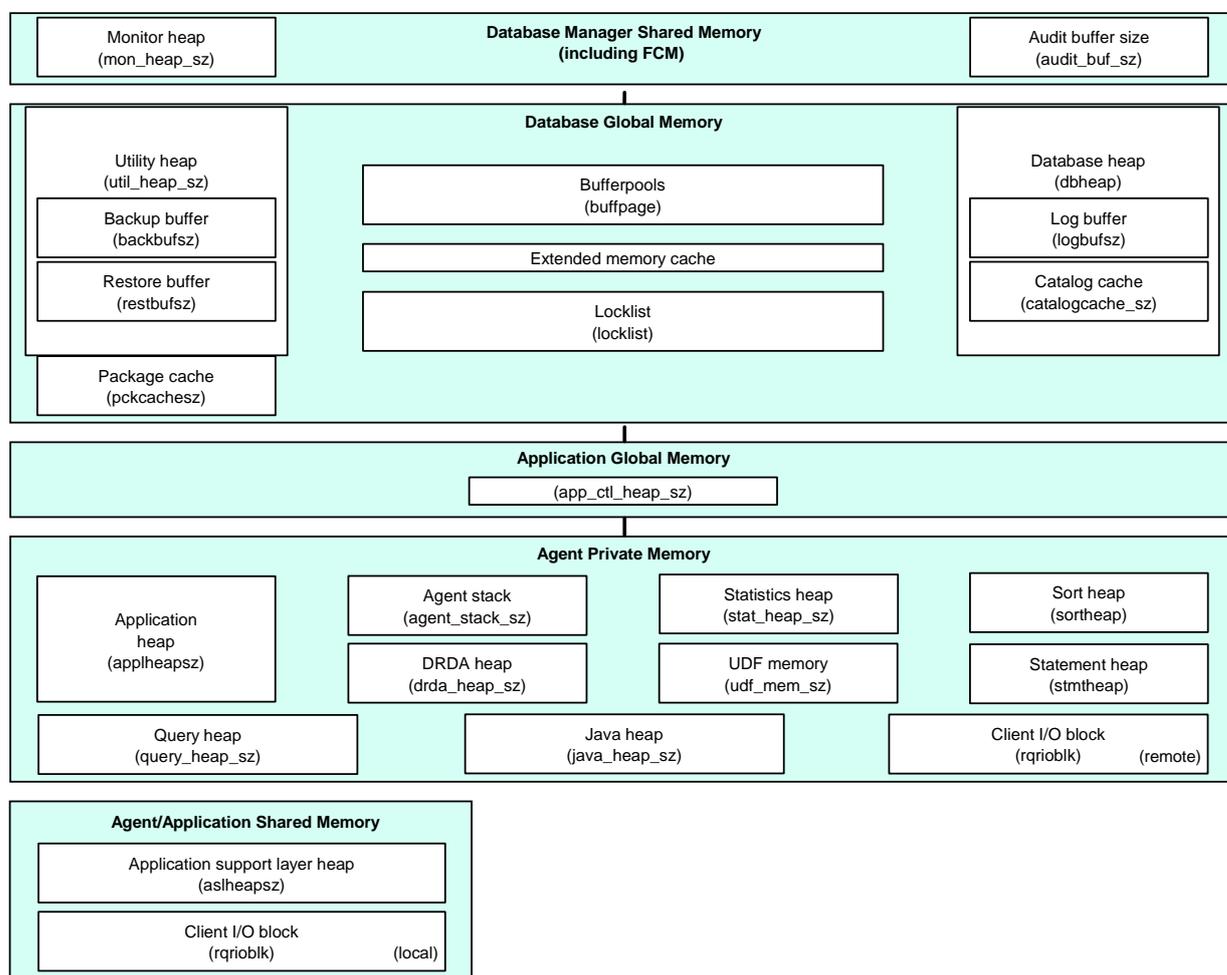


Figure 6.2 – The DB2 memory model / Figure 6.2 – Il memory model di DB2

When an instance is started, the database manager shared memory is allocated. This normally does not take much space. When you first connect to a database, the Database Global Memory is allocated. In this block, the buffer pool is one of the most important parts, especially for improving query performance. The size of the buffer pools will determine how large the entire Database Global Memory will be.

Quando un instance è lanciata, la memoria condivisa del manager del database viene allocata. Ciò normalmente non occupa molto spazio. Quando all'inizio ci si collega ad un database, la memoria globale del database è allocata. In questo blocco, il buffer pool è uno delle parti più importanti, soprattutto per il miglioramento della prestazione della query. La dimensione dei buffer pool determinerà quanto grande sarà l'intera memoria globale del database.

Agent private memory is the memory used by each DB2 agent. Without using connection concentrator, each connection requires one agent. Typically an agent can use approximately 3 to 5 MB. With connection concentrator, several connections can use one agent, therefore reducing the need for more physical memory.

La memoria privata dell'agente è la memoria usata da ogni agente DB2. Senza usare il connection concentrator, ogni collegamento richiede un agente. Un agente può usare tipicamente da 3 a 5 MB. Con il connection concentrator, parecchi collegamenti possono usare un agente, riducendo quindi l'esigenza di memoria fisica.

6.3 DB2 storage model / Modello di storage di DB2

In this section we will describe the following concepts:

- Pages and extents
- Buffer pool
- Table space

In questa sezione descriveremo i seguenti concetti:

- Pagine ed extent
- Buffer pool
- Spazio di tabella

6.3.1 Pages and extents / Pagine e extent

A page is the minimum unit of storage in DB2. Allowed page sizes are: 4K, 8K, 16K and 32K. An extent is a grouping of pages. Working with one page at a time in DB2 would be costly in terms of performance; therefore, DB2 works with extents at a time instead. The page size and extent size are defined when working with buffer pools and table spaces as we will see in the next

Una pagina è l'unità minima di storage in DB2. Le dimensioni di pagina permesse sono: 4K, 8K, 16K e 32K. Un extent è un raggruppamento di pagine. In DB2, lavorando con una pagina alla volta potrebbe essere costoso in termini di prestazioni; quindi, DB2 funziona con più extent alla volta. La dimensione delle pagine e degli extent è definita quando si lavora con i buffer pool e gli

sections.

spazi di tabella come vedremo nelle sezioni seguenti.

6.3.2 Buffer pools / I buffer pool

A buffer pool is real memory cache for table and index data. It improves performance by reducing direct sequential I/O and it promotes asynchronous reading (pre-fetching) and writing. That is to say, DB2 anticipates what pages will be needed and pre-fetches them from the disk to the buffer pool so they are ready to use.

Un buffer pool è un cache di memoria reale per i dati tabella e indice. Migliora la prestazioni riducendo l'I/O sequenziale diretto e promuove la lettura asincrona (pre-fetching) e la scrittura. DB2 prevede che pagine saranno necessarie ed esegue il pre-fetch dal disco ai buffer pool in modo che siano pronte per l'uso.

Buffer pools are allocated in memory units of 4K, 8K, 16K, and 32K pages. There should be at least one buffer pool per database, and at least one matching buffer pool for a table space of a given page size.

I buffer pool sono allocati in unità di memoria delle pagine di 4K, 8K, 16K e 32K. Ci dovrebbe essere almeno uno buffer pool per database, e almeno uno buffer pool di corrispondenza per uno spazio di tabella per un certa misura di pagina.

Creating a buffer pool / Creare un buffer pool

To create a buffer pool you can use the `CREATE BUFFERPOOL` statement. Alternatively, using the Control Center you can right click on the Buffer pool folder within a given database and choose `Create` as shown in Figure 6.3

Per creare un buffer pool si può usare lo statement `CREATE BUFFERPOOL`. Alternativamente, usando il Control Center si può fare click destro del mouse sulla cartella Buffer Pool all'interno di un certo database e scegliere `Create` come illustrato in figura 6.3

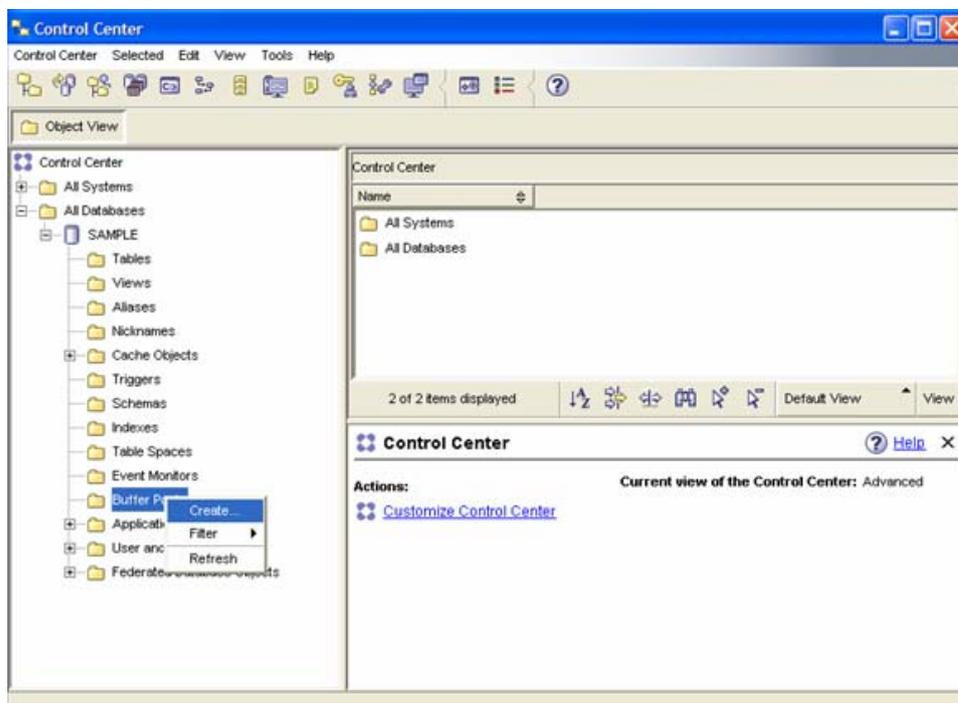


Figure 6.3 – Creating a buffer pool / Figura 6.3 – Creare un buffer pool

After clicking on **Create**, the **Create Buffer Pool Dialog** will appear as shown in Figure 6.4.

Dopo il clic su **Create**, comparirà la finestra di dialogo **Create Buffer Pool** come illustrato in figura 6.4.

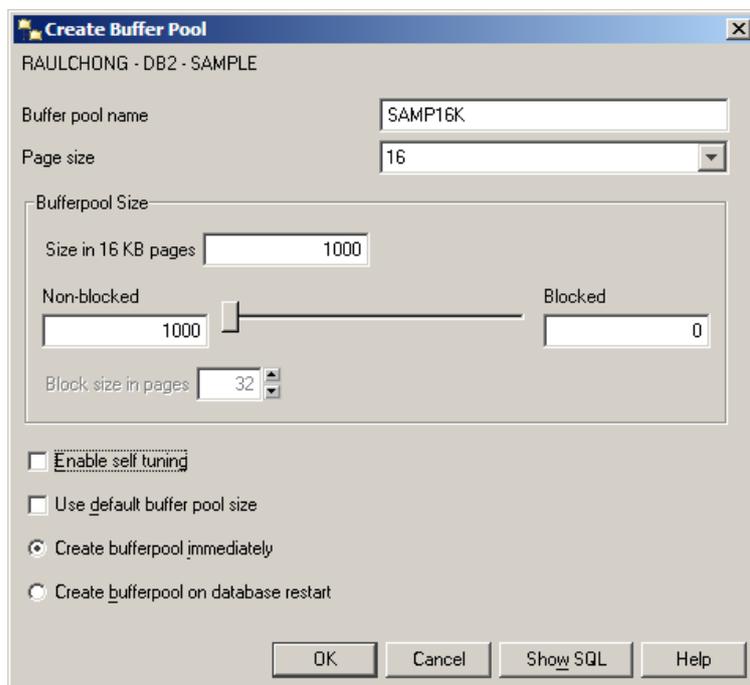


Figure 6.4 – Create a buffer pool dialog box / Figura 6.4 – finestra di dialogo per creare un buffer pool

Most entries in Figure 6.4 are self explanatory. The fields “Non-blocked” and “Blocked” refer to the number of pages that should exist as non-blocked and as blocked. Blocked-based buffer pools ensure that contiguous pages on disk are moved to the buffer pool also contiguously in a blocked area. This may improve performance. The number of pages must not be greater than 98 percent of the number of pages for the buffer pool. Specifying the value 0 disables block I/O.

Once the buffer pool has been created, it would be displayed in the Control Center as shown in Figure 6.5.

La maggior parte delle opzioni visualizzate nella figura 6.4 sono evidenti. I campi “Non-blocked” e “Blocked” si riferiscono al numero di pagine che dovrebbero esistere come non-blocked e come blocked. I buffer pool blocked-based assicurano che le pagine contigue sul disco siano spostate verso il buffer pool contiguamente in una zona blocked. Questo può migliorare le prestazioni. Il numero di pagine non deve essere più del 98 per cento del numero di pagine per il buffer pool.

Specificando il valore 0 si disabilita il block I/O. Una volta che il buffer pool è stato creato, sarà visualizzato nel Control Center come illustrato in figura 6.5.

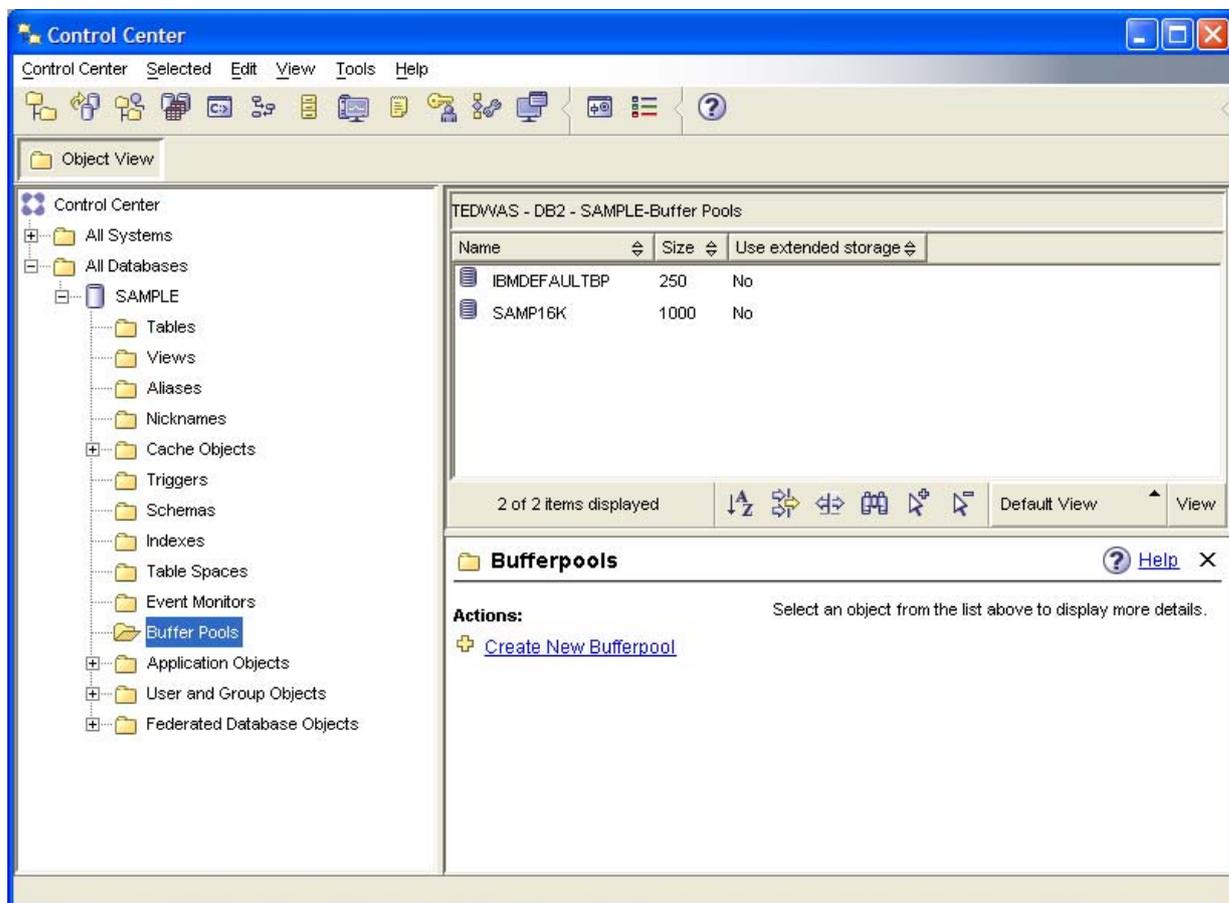


Figure 6.5 – The Control Center after the creation of buffer pool “SAMP16K” / Figura 6.5 – Il Control Center dopo la creazione del buffer pool “SAMP16K”

6.3.3 Table spaces / Spazi di tabella

Table spaces are a logical interface between logical tables and the system's physical memory (buffer pool) and containers (disks). Use the `CREATE TABLESPACE` statement to create a table space where you can specify:

- The page size for the table space (4KB, 8KB, 16KB, or 32KB). The page size must correspond to a buffer pool with the same page size.
- The buffer pool name associated to this table space

Gli spazi di tabella sono un'interfaccia logica fra le tabelle logiche e la memoria fisica del sistema (il buffer pool) e i contenitori (i disk). Usare lo statement `CREATE TABLESPACE` per creare uno spazio di tabella in cui puoi specificare:

- La dimensione della pagina per lo spazio di tabella (4KB, 8KB, 16KB, o 32KB). La dimensione della pagina deve corrispondere a un buffer pool con la stessa dimensione della pagina.
- Il nome del buffer pool associato a questo spazio di tabella

- An extent size
- A pre-fetch size
- Un extent size
- Un pre-fetch size

Table space types / Tipi di spazio di tabella

There are three types of table spaces:

- Regular
These are for user tables. For example, the USERSPACE1 table space create by default is a regular table space.
- Large
These are used to optionally separate LOB data into its own table space. It is also used for storing XML data for databases created with pureXML support – where the database was created as UNICODE and using the XML data type for columns. Large table spaces are the default.
- Temporary
There are two types of temporary table spaces:
 - System temporary
These are used by DB2 for internal operations, such as sorts. For example, the TEMPSPACE1 table space, created by default when you create a database, is a system temporary table space.
 - User temporary
These are used to create User Defined Global Temporary tables (temporary tables in-memory). They are often confused with system temporary table spaces.

Ci sono tre tipi di spazi di tabella:

- Regular
Questi sono usati per le tabelle utente. Per esempio, lo spazio di tabella USERSPACE1 creato di default è uno spazio di tabella regolare.
- Large
Questi sono usati facoltativamente per separare i dati LOB nel relativo proprio spazio di tabella. Inoltre è usato per la memorizzazione dei dati XML per i database creati con il supporto di pureXML – dove il database era stato creato come UNICODE e usando il tipo di dati XML per le colonne. Gli spazi di tabella large sono di default.
- Temporary
Ci sono due tipi di spazi di tabella temporary:
 - System temporary
Questi sono usati da DB2 per operazioni interne come gli ordinamenti. Per esempio, lo spazio di tabella TEMPSPACE1, creato di default quando si crea un database, è uno spazio di tabella temporary del sistema.
 - User temporary
Questi sono usati per creare le tabelle User Defined Global Temporary (tabelle provvisorie in-memory). Sono confusi spesso con gli spazi di tabella temporary del sistema.

Table space management / Amministrazione degli spazi di tabella

Table spaces can be classified based on how they are managed. This can be specified in the CREATE TABLESPACE statement:

Gli spazi di tabella possono essere classificati in base a come sono amministrati. Ciò può essere specificato nello statement CREATE TABLESPACE.

Managed by system / Amministrato dal sistema

This type of table space is known as System Managed Storage (SMS). This means the operating system manages the storage. They are easy to manage, and the containers are file system directories. The space is not pre-allocated, but the files grow dynamically. Once you specify the containers, these are fixed at creation time and other containers cannot be added later, unless a redirected restore is used. When using SMS table spaces the table data, index and LOB data cannot be spread across different table spaces.

Questi tipi di spazi di tabella sono conosciuti come System Managed Storage (SMS). Ciò significa che il sistema operativo amministra la memorizzazione. Sono facili da amministrare e i contenitori sono directory del file system. Lo spazio non è pre-allocato, ma i file crescono dinamicamente. Una volta che si specificano i contenitori, questi sono fissati al tempo della creazione ed altri contenitori non possono essere aggiunti dopo, a meno che non venga usato un restore reindirizzato. Quando si usano spazi di tabella SMS, le tabelle dati, di indice e i dati di LOB non possono essere sparsi attraverso gli spazi di tabella differenti.

Managed by database / Amministrato dal database

This type of table space is known as Database Managed Storage (DMS). This means that DB2 manages the storage. Management of the space requires more manual intervention from a DBA. Containers can be pre-allocated files or raw devices. For raw devices, data is written directly without O/S caching.

Questo tipo di spazio di tabella è conosciuto come Database Managed Storage (DMS). Ciò significa che DB2 amministra la sua memorizzazione. L'amministrazione dello spazio richiede più intervento manuale da un DBA. I contenitori possono essere file pre-allocati o dispositivi raw. Per i dispositivi raw, i dati sono scritti direttamente senza il caching del SO.

Containers can be added, dropped or resized. DMS table spaces are best for performance, and table data, index, and LOB data can be split into separate table spaces, which improves performance.

I contenitori possono essere aggiunti, eliminati o ridimensionati. Gli spazi di tabella DMS sono migliori per le prestazioni e per le tabelle dati, indice ed i dati di LOB possono essere divisi in separati spazi di tabelle, che migliorano le prestazioni.

Managed by automatic storage / Amministrato da storage automatico

This type of table space is managed by automatic storage, and can benefit from the ease of use similar to SMS table spaces, but with the best performance and flexibility of DMS table spaces. Therefore, starting with DB2 9, this is the default type of table space. For these table spaces, a user first specifies a logical group of storage devices for. No explicit container definitions are provided. Containers are automatically created across the storage paths. Growth of existing containers and addition of new ones is completely managed by DB2.

To allow for automatic storage, you first need to create a database with automatic storage enabled (this is the default behavior) and associate a set of storage paths with it. After creation, if needed, you can redefine the storage paths using a database RESTORE operation. Then, you can create table spaces to use automatic storage (again, this is the default behavior).

Questo tipo di spazio di tabella è amministrato da storage automatico e può trarre beneficio dalla facilità d'uso simile agli spazi di tabella SMS, ma con miglior prestazioni e flessibilità degli spazi di tabella DMS. Di conseguenza, a partire da DB2 9, questo è il tipo di spazio di tabella di default. Per questi spazi di tabella, un utente prima specifica un gruppo logico di dispositivo di storage. Non è fornita nessuna definizione esplicita del contenitore. I contenitori sono creati automaticamente attraverso i percorsi di memorizzazione. La crescita dei contenitori esistenti e l'aggiunta di nuovi contenitori è completamente amministrata da DB2.

Per permettere lo storage automatico, si deve prima creare un database con storage automatico abilitato (questo è il comportamento di default) e associarlo ad un insieme di percorsi di memorizzazione. Dopo la creazione, se necessario, si può ridefinire il percorso di memorizzazione usando l'operazione di database RESTORE. Poi, si possono creare gli spazi di tabella per usare lo storage automatico (di nuovo, questo è il comportamento di default).

Automatic storage example / Esempio di storage automatico

First create the database with automatic storage enabled as in these examples:

- Automatic storage is enabled by default:

```
CREATE DATABASE DB1
```

- Automatic storage is explicitly specified:

```
CREATE DATABASE DB1 AUTOMATIC STORAGE YES
```

- Automatic storage is enabled by default, but the storage paths are indicated:

```
CREATE DATABASE DB1 ON /data/path1, /data/path2
```

- Automatic storage is disabled explicitly:

```
CREATE DATABASE DB1 AUTOMATIC STORAGE NO
```

Next, create the table space with automatic storage enabled as in these examples:

- Automatic storage for table spaces is also enabled by default:

```
CREATE TEMPORARY TABLESPACE TEMPTS
```

- Automatic storage is explicitly specified for the table space:

```
CREATE TABLESPACE TS2 MANAGED BY AUTOMATIC STORAGE
```

- Automatic storage is implicitly specified, the initial size is allocated, along with how much it will increase, and the maximum size it can increase:

```
CREATE TABLESPACE TS1
  INITIALSIZE 500 K
  INCREASESIZE 100 K
  MAXSIZE 100 M
```

Prima si crea il database con storage automatico abilitato come in questi esempi:

- Lo storage automatico è abilitato di default:

- Lo storage automatico è specificato esplicitamente:

- Lo storage automatico è abilitato di default, ma i percorsi di memorizzazione sono indicati:

- Lo storage automatico è esplicitamente disabilitato:

Dopo, si crea lo spazio di tabella con storage automatico abilitato come in questi esempi:

- Lo storage automatico per gli spazi di tabella è abilitato di default:

- Lo storage automatico è specificato esplicitamente per lo spazio di tabella:

- Lo storage automatico è specificato implicitamente, la dimensione iniziale è assegnata insieme a quanto aumenterà e la misura massima che potrà raggiungere:

How data is stored in table spaces / Come i dati sono memorizzati negli spazi di tabella

By default, DB2 will write to disk extents at a time striped across containers. For example, if you have a 4K table space with an extent size of 8 using 3 raw containers on a DMS table space, this means that 32K of data (4K x 8 pages per extent = 32K) will be written to one disk before writing to the next. This is shown in Figure 6.6. Note that tables do not share extents.

Di default, DB2 scriverà su un blocco di area alla volta suddivisa tra vari dischi. Per esempio, se si ha uno spazio di tabella di 4K con un extent di dimensione 8 usando 3 contenitori raw su uno spazio di tabella DMS, questo significa che 32K di dati (4K x 8 pagine per extent = 32K) saranno scritti su un disco prima di scrivere sul seguente. Ciò è mostrato in figura 6.6. Nota che le tabelle non condividono gli extent.

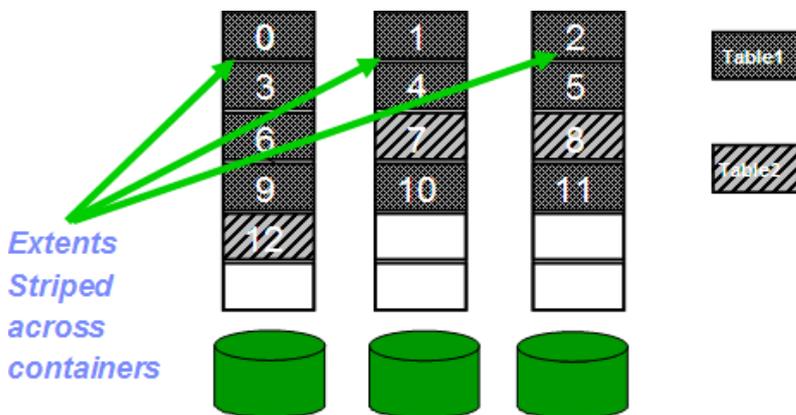


Figure 6.6 – Writing data in table spaces / Figura 6.6 – Scrivere i dati negli spazi di tabella

Creating a table space using the Control Center / Creare un spazio di tabella usando il Control Center

To create a table space from the Control Center, right click on the Table Spaces folder within a given database and choose *Create* as shown in Figure 6.7. The “Create table space wizard” will appear, as shown in Figure 6.8.

Per creare uno spazio di tabella dal Control Center, premere il tasto destro sulla cartella Table Spaces all'interno un certo database e scegliere *Create* come mostrato in figura 6.7. Il “Create table space wizard”, comparirà, come mostrato in Figura 6.8.

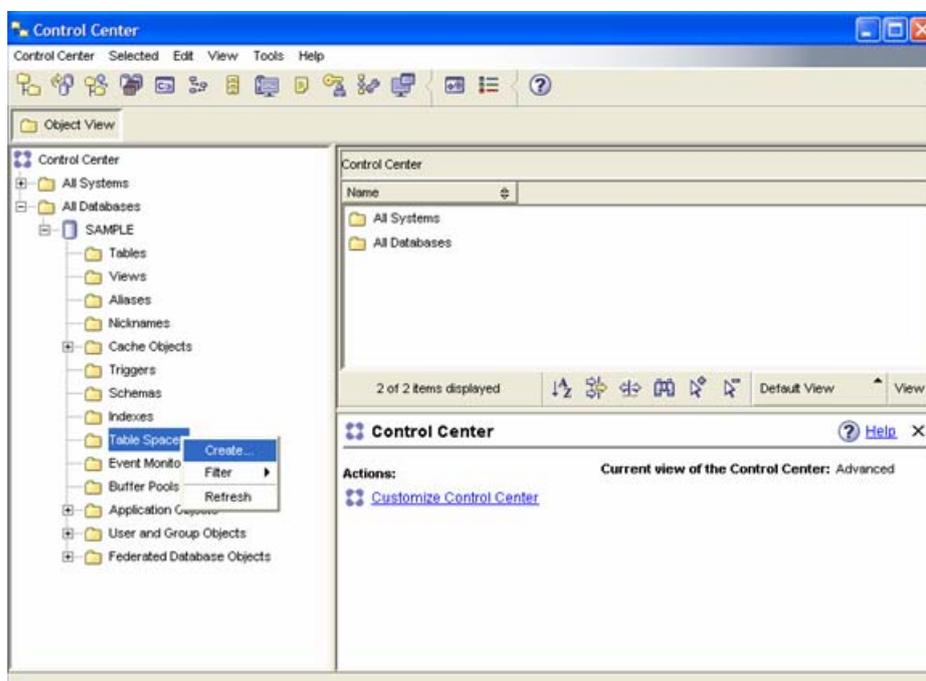


Figure 6.7 – Creating a table space from the Control Center / Figura 6.7 – Creare un spazio di tabella dal Control Center



Figure 6.8 – Create Table Space Wizard / Figura 6.8 - Create Table Space Wizard

7

Chapter 7 – DB2 Client Connectivity / Capitolo 7 - Connettività del client DB2

This chapter covers the setup required to connect from a DB2 client to a DB2 server using TCPIP. Note that a DB2 server comes with a client component, so a DB2 server can also behave as a client to connect to another DB2 server. There are several ways to set up DB2 client connectivity; however, in this chapter we discuss only the easiest method which is using the Configuration Assistant.

Questo capitolo riguarda il setup richiesto per collegarsi da un client DB2 ad un server DB2 usando TCPIP. Nota che un server DB2 è provvisto di un componente client, in modo che un server DB2 possa anche comportarsi come client per collegarsi ad un altro server DB2. Ci sono diversi metodi per il setup della connettività del client DB2; tuttavia, in questo capitolo discutiamo soltanto il metodo più facile che usa il Configuration Assistant.

Note:

For more information about the DB2 client connectivity, watch this video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4222>

Nota:

Per ulteriori informazioni sull'connettività del client DB2, guardi questo video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4222>

7.1 DB2 Directories / Directory di DB2

DB2 directories are binary files that store information about which databases you can connect to from your machine. There are four directories:

Le directory DB2 sono file binari che memorizzano informazioni riguardo i database ai quali è possibile collegarsi dal proprio computer. Ci sono quattro directory:

1. System database directory

This directory is like a table of contents of a book. It shows all the databases, whether they are local or remote, that

Questa directory è come un indice di un libro. Mostra tutti i database, sia locali che remoti, a cui puoi collegarti. Per un

you can connect to. For a local database, it will have a pointer to the *Local database directory*. For a remote database, it will have a pointer to an entry in the *Node directory*. To review the contents of this directory issue the command:

```
list db directory
```

2. Local database directory

This directory contains information about databases that you can connect to and that reside on your machine. To review the contents of this directory, issue the command:

```
list db directory on <drive/path>
```

3. Node directory

This directory includes information about how to connect to a given database. For example, if the TCP/IP protocol is used, a TCP/IP node entry would include the IP address of the server where the DB2 database you are trying to connect resides, and the port of the instance where this database resides. To review this directory, issue the command:

```
list node directory
```

4. DCS directory

This directory will only appear if you have installed the DB2 Connect software to connect to DB2 on z/OS (mainframe), or DB2 on i5/OS. To review the contents of this directory, issue the command:

```
list dcs directory
```

Reviewing and updating the contents of all of these directories can be performed through the Configuration Assistant GUI tool.

database locale, avrà un riferimento al *Local database directory*. Per un database remoto, avrà un riferimento ad un'entry nel *Node directory*. Per esaminare il contenuto di questo directory, usa il comando:

Questa directory contiene le informazioni sui database a cui puoi collegarti e che risiedono sul tuo computer. Per esaminare il contenuto di questa directory, usa il comando:

Questa directory comprende le informazioni su come collegarsi ad un dato database. Per esempio, se viene usato il protocollo TCP/IP, un'entry del node TCP/IP comprenderebbe l'indirizzo IP del server dove risiede il database DB2 a cui stai provando a collegarti e la porta dell'instance dove risiede questo database. Per esaminare questa directory, usa il comando:

Questa directory comparirà soltanto se hai installato il software DB2 Connect per collegarti a DB2 su z/OS (mainframe), o DB2 su i5/OS. Per esaminare il contenuto di questa directory, usa il comando:

L'analisi e l'aggiornamento del contenuto di tutte queste directory possono essere eseguiti tramite lo strumento Configuration Assistant GUI.

7.2 Configuration Assistant

Using the Configuration Assistant GUI tool, you can easily configure connectivity between a DB2 client and a DB2 server.

To launch the Configuration Assistant on Windows, you can choose: *Start > Programs > IBM DB2 > DB2COPY1 > Set-up Tools > Configuration Assistant*

From the Command line, you can start the tool using the command `db2ca`. Figure 7.1 shows the Configuration Assistant.

Per mezzo dello strumento Configuration Assistant GUI, puoi configurare facilmente la connettività fra un client DB2 e un server DB2.

Per lanciare il Configuration Assistant su Windows, puoi selezionare: *Start > Programs > IBM DB2 > DB2COPY1 > Set-up Tools > Configuration Assistant*.

Da linea di comando puoi avviare il tool usando il comando `db2ca`. La figura 7.1 mostra il Configuration Assistant.

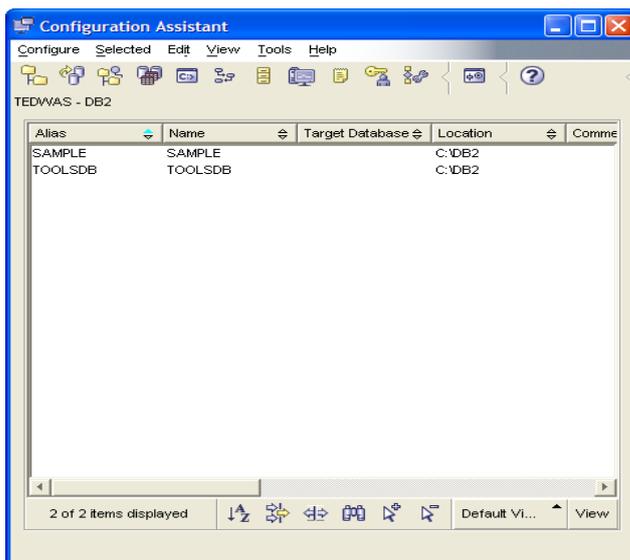


Figure 7.1 – The Configuration Assistant / Figura 7.1 – Il Configuration Assistant

7.2.1 Setup required at the server / Setup richiesto al server

There are two things that need to be set up at the server:

1) DB2COMM

This registry variable determines which communication protocol listeners should be monitoring requests from clients. Typically TCPIP is the communication protocol most used. Changing this parameter requires an instance re-start. To review and change the value of DB2COMM in the Configuration Assistant, choose *Configure -> DB2 Registry* as shown in Figure 7.2 and Figure 7.3.

Ci sono due cose che devono essere configurate sul server:

1) DB2COMM

Questa variabile di registro determina quali protocolli di comunicazione in ascolto dovrebbero monitorare richieste dai client. Tipicamente il protocollo di comunicazione più usato è TCPIP. Il cambiamento di questo parametro richiede un restart dell'instance. Per esaminare e cambiare il valore di DB2COMM nel Configuration Assistant, scegli *Configure -> DB2 Registry* come indicato in figura 7.2 ed in figura 7.3.

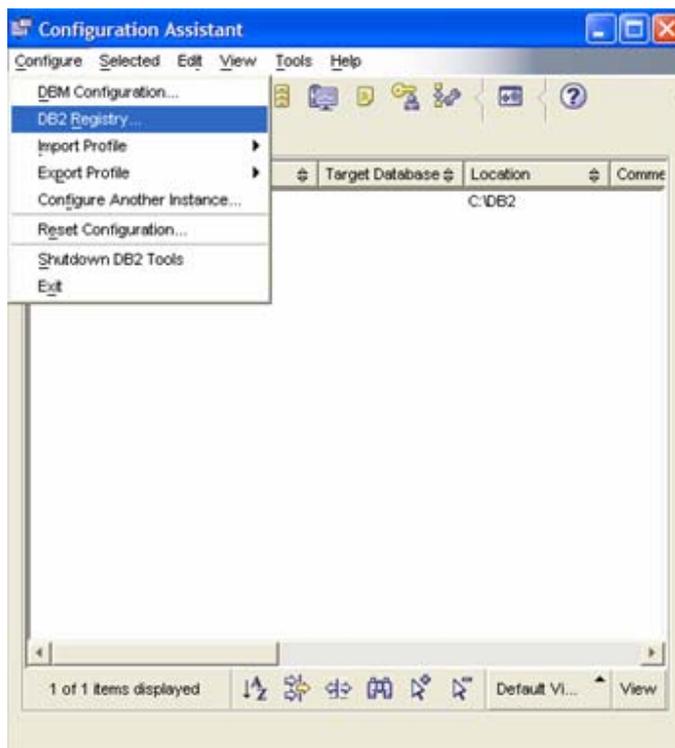


Figure 7.2 – Accessing the DB2 Registry / Figura 7.2 – Accedere al Registro DB2

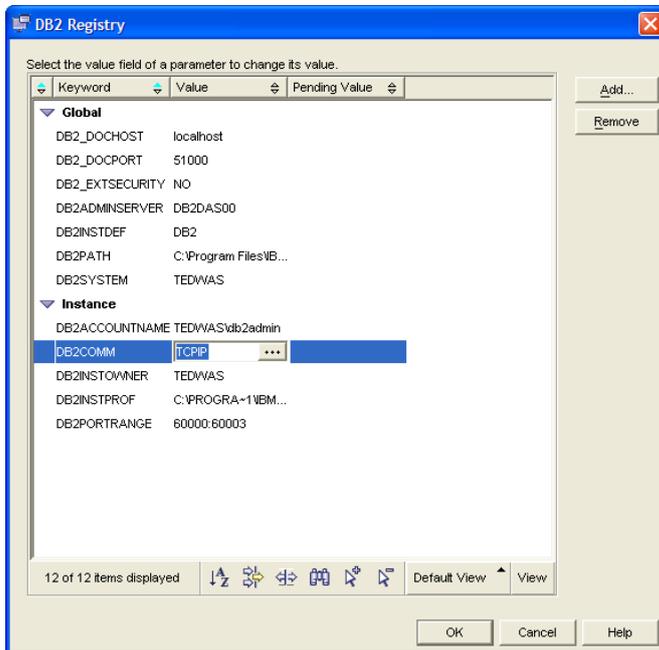


Figure 7.3 – Verifying the value of the DB2COMM DB2 Registry variable / Figura 7.3 – Verificare il valore della variabile di registro di DB2COMM DB2

2) SVCENAME

This dbm cfg parameter should be set to the service name (as defined in the TCPIP services file) or to the port number to use when you want to access databases of this instance. From the Configuration Assistant, choose *Configure > DBM configuration* as shown in Figure 7.4.

2) SVCENAME

Questo parametro del dbm cfg dovrebbe essere settato al nome di servizio (come definito nel file di servizi TCPIP) o al numero della porta da usare quando vuoi accedere ai database di questa instance. Dal Configuration Assistant, scegli *Configure > DBM configuration* come indicato in figura 7.4.

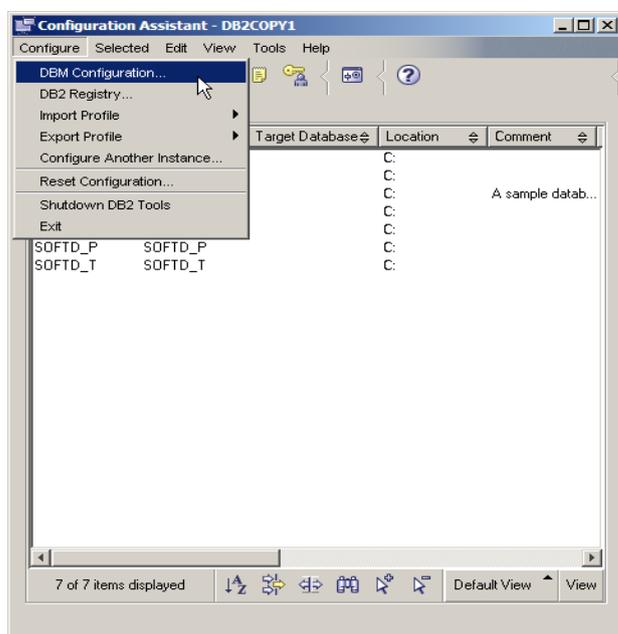


Figure 7.4 –Reviewing the dbm cfg from the Configuration Assistant / Figura 7.4 - Esaminare il file dbm cfg dal Configuration Assistant

Once you are in the DBM Configuration window, find the Communications section, and look for SVCENAME. You can change the value to a string or even to a port number if needed. This is shown in Figure 7.5.

Una volta che sei nella finestra DBM Configuration, trova la sezione Comunicazioni e cerca SVCENAME. Puoi cambiare il valore ad una stringa o persino al numero di una porta, se necessario. Questo è indicato in figura 7.5.

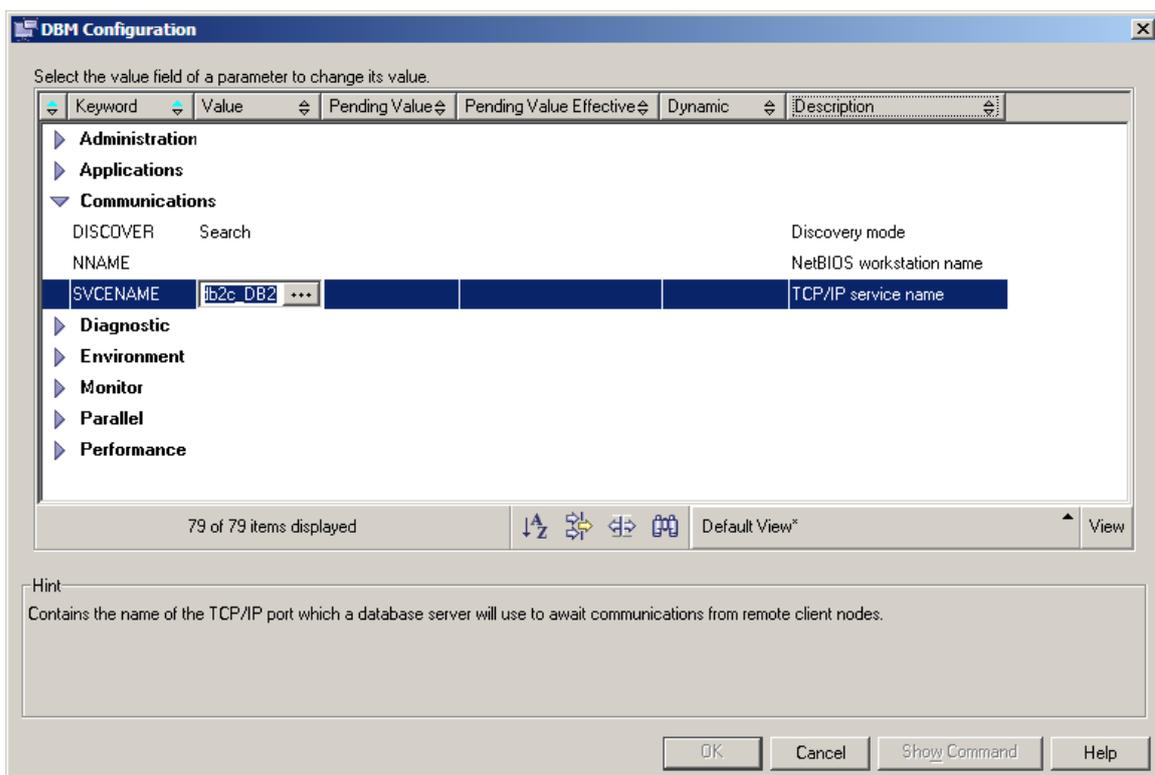


Figure 7.5 –Reviewing the SVCENAME dbm cfg parameter / Figura 7.5 – Esaminare il parametro SVCENAME del dbm cfg

7.2.2 Setup required at the client / Setup richiesto al client

At the client, you need to know this information beforehand:

1. The name of the database you want to connect to.
2. The port number of the DB2 instance at the server where the

Per quanto riguarda il client, è necessario conoscere in anticipo queste informazioni:

1. Il nome del database a cui vuoi collegarti.
2. Il numero di porta dell'instance del server DB2 in cui risiede il

database resides. You can also use a service name, as long as there is a matching entry in the TCPIP services file.

3. The operating system user ID and password to connect to the database. This user ID must have been previously defined at the server.

The above information can be input from the DB2 client using the Configuration Assistant. First, launch the Add Database Wizard by clicking on the *Selected -> Add Database Using Wizard* choice, as shown in Figure 7.6.

database. Puoi anche usare un nome di servizio, se c'è un'entry corrispondente nel file di servizi di TCPIP.

3. Lo user ID e la password del sistema operativo per collegarsi al database. Questo user ID deve essere stato precedentemente definito sul server.

Le informazioni sopra indicate possono essere immesse dal client DB2 usando il Configuration Assistant. In primo luogo, lancia il Add Database Wizard cliccando sulla voce *Selected -> Add Database Using Wizard*, come indicato in figura 7.6.

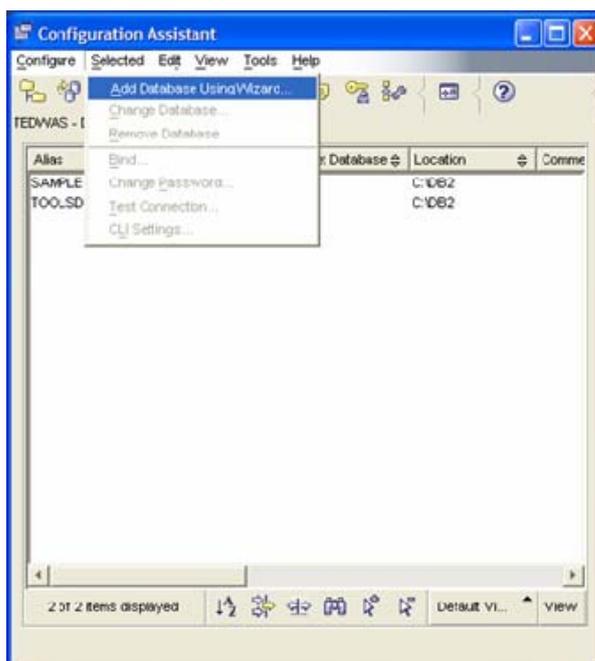


Figure 7.6 – Invoking the Add Database Wizard / Figura 7.6 - Invocazione di Add Database Wizard

You can also get to this wizard by right clicking on the white space in the Configuration Assistant and choosing *Add Database Using Wizard*.

Puoi anche raggiungere questo wizard facendo clic destro sullo spazio bianco nel Configuration Assistant e scegliendo *Add Database Using Wizard*.

Figure 7.7 shows the Add Database Wizard.

La figura 7.7 mostra il Add Database Wizard.

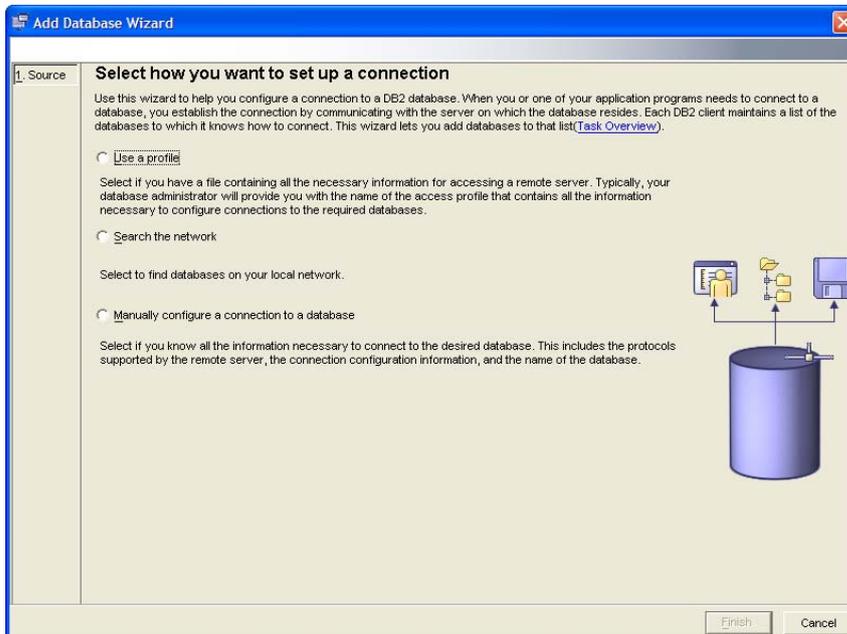


Figure 7.7 –Add Database Wizard / Figura 7.7 - Add Database Wizard

In the Add Database Wizard, there are three options:

Nel Add Database Wizard ci sono tre opzioni:

Use a Profile / Usare un profilo

There may be situations when you need to configure many clients to connect to the same DB2 server. In these situations, it is convenient to perform all configurations from one client, and store these configurations into a “profile” file. With this file, you can load all the information directly to other clients. In Figure 7.7, if you choose *Use a Profile* you would be loading the information from an existing “profile”. More details are provided later in this chapter describing how to create client and server profiles.

Ci possono essere situazioni in cui serve configurare molti client a collegarsi allo stesso server DB2. In queste situazioni, è conveniente eseguire tutte le configurazioni da un client e memorizzare queste configurazioni in un file di “profilo”. Con questo file, puoi caricare tutte le informazioni direttamente ad altri client. Nella figura 7.7, se scegli *Use a Profile* potresti caricare le informazioni da un “profilo” esistente. Maggiori dettagli sono forniti successivamente in questo capitolo descrivendo come creare i profili dei client e dei server.

Search the network / Cercare la rete

This method, also known as “Discovery”, tells DB2 to search the network for a given server, instance, and database. For this method to work, the DAS must be running on each DB2 server where databases are to be discovered. With this method, there are two ways to perform the search:

- Search:
Search the entire network. This is not recommended if your network is large and with many hubs, as it would take a long time to retrieve data from every system

- Known:
Search the network for a known server at an address you provide.

The two methods are illustrated in Figure 7.8.

Questo metodo, anche conosciuto come “Discovery”, istruisce DB2 su come cercare la rete per un dato server, instance e database. Per far funzionare questo metodo, il DAS deve essere in esecuzione su ogni server DB2 in cui i database devono essere scoperti. Con questo metodo, ci sono due modi di effettuare la ricerca:

- Search:
Cercare nell'intera rete. Ciò non è consigliabile se la tua rete è grande e con molti hub, poiché richiederebbe molto tempo per reperire i dati da ogni sistema

- Known:
Cercare la rete per un server conosciuto ad un indirizzo che fornisci.

I due metodi sono illustrati nella figura 7.8.

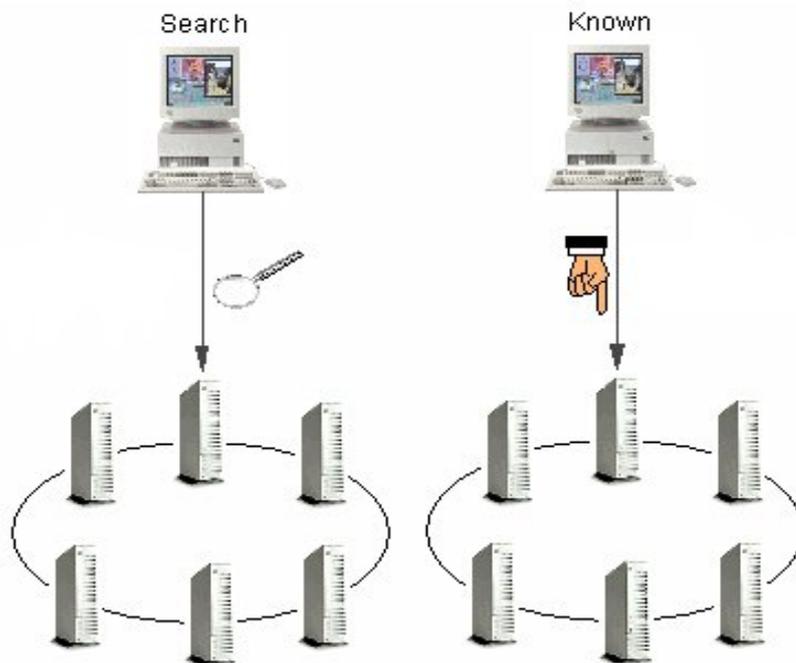


Figure 7.8 –The Search and Known search (or Discovery) methods / Figura 7.8 – I metodi di ricerca Search e di Known (o Discovery)

There may be circumstances when an administrator would not like clients to search the network for databases with confidential information. This can be prevented at the DAS, the instance or the database level.

Figure 7.9 provides details about this.

Ci possono essere circostanze nelle quali un amministratore non vorrebbe che i client cercassero i database con informazioni confidenziali nella rete. Ciò può essere evitato al DAS, a livello instance o database.

La figura 7.9 fornisce i dettagli su questo.

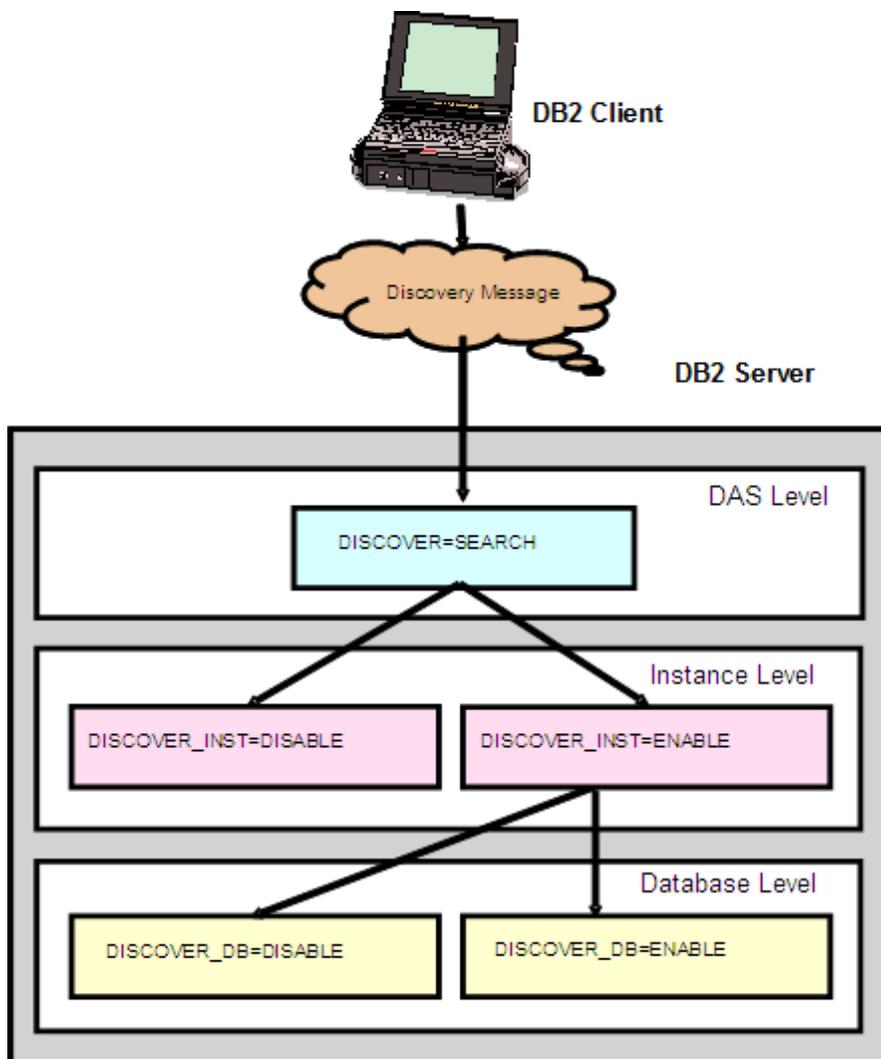


Figure 7.9 – Configuring parameters to allow for discovery / Figura 7.9 - Parametri di configurazione per permettere la scoperta

Figure 7.9 shows the different levels where you can enable or disable discovery. At the DAS level, you can give the DISCOVER parameter a value of SEARCH or KNOWN. At the instance level, the DISCOVER_INST dbm cfg parameter can be set to DISABLE or ENABLE. Finally, at the database level, the DISCOVER_DB parameter can also be set to ENABLE or DISABLE. Setting these parameters accordingly provides you granularity for database discovery.

La figura 7.9 mostra i livelli diversi dove puoi permettere o disabilitare la scoperta. Al livello DAS puoi dare al parametro DISCOVER un valore di SEARCH o KNOWN. Al livello instance il parametro del dbm cfg DISCOVER_INST può essere fissato a DISABLE o ENABLE. Per concludere, al livello database il parametro DISCOVER_DB può anche essere fissato a ENABLE o DISABLE. L'impostazione dei questi parametri ti

fornisce di conseguenza la granularità per la scoperta dei database.

Manually configure a connection to a database / Configurare manualmente un collegamento ad un database

Using this method, you manually add host name, port numbers and database information to the Configuration Assistant, which will then generate "catalog" commands to execute the connectivity configuration. The Configuration Assistant will not check that the information is correct. You will know it is incorrect if you cannot connect to a server. Also, ensure the user ID and password you provide to connect to the remote database is correct. By default the authentication takes place on the DB2 server you are trying to connect to, therefore, you must provide a user ID and password defined on that server.

Usando questo metodo, aggiungi manualmente il nome dell'host, i numeri di porta e le informazioni del database al Configuration Assistant, che genererà i comandi "catalogo" per eseguire la configurazione di connettività. Il Configuration Assistant non verificherà che le informazioni siano corrette. Saprai che sono errate se non riesci a collegarti ad un server. Inoltre, assicurati che lo user ID e la password che fornisci per collegarti al database remoto siano corrette. Di default l'autenticazione avviene sul server DB2 a cui stai provando a collegarti, quindi devi fornire uno user ID e password definiti su quel server.

7.2.3 Creating Client and Server Profiles / Creare i profili del client e del server

If you are configuring a large number of servers or clients, rather than set up each one individually, you can set up one, then export a profile (i.e. configuration file) from it, and then apply the profile to the other clients/servers. This saves a lot of administration time when setting up the environment.

Se stai configurando un gran numero di server o client, piuttosto di fare il setup per ciascuno individualmente, puoi fare il setup per uno, poi esportare un profilo (cioè un file di configurazione) da esso, e dopo applicare il profilo agli altri server/client. Con questo si risparmia molto tempo d'amministrazione quando si configura l'ambiente.

To create a customized profile from the Configuration Assistant, click on the Configure Menu, then select *Export Profile > Customize*, as shown in Figure 7.10

Per creare un profilo personalizzato dal Configuration Assistant, clicca sul menu Configure, poi scegli *Export Profile > Customize*, come indicato in figura 7.10

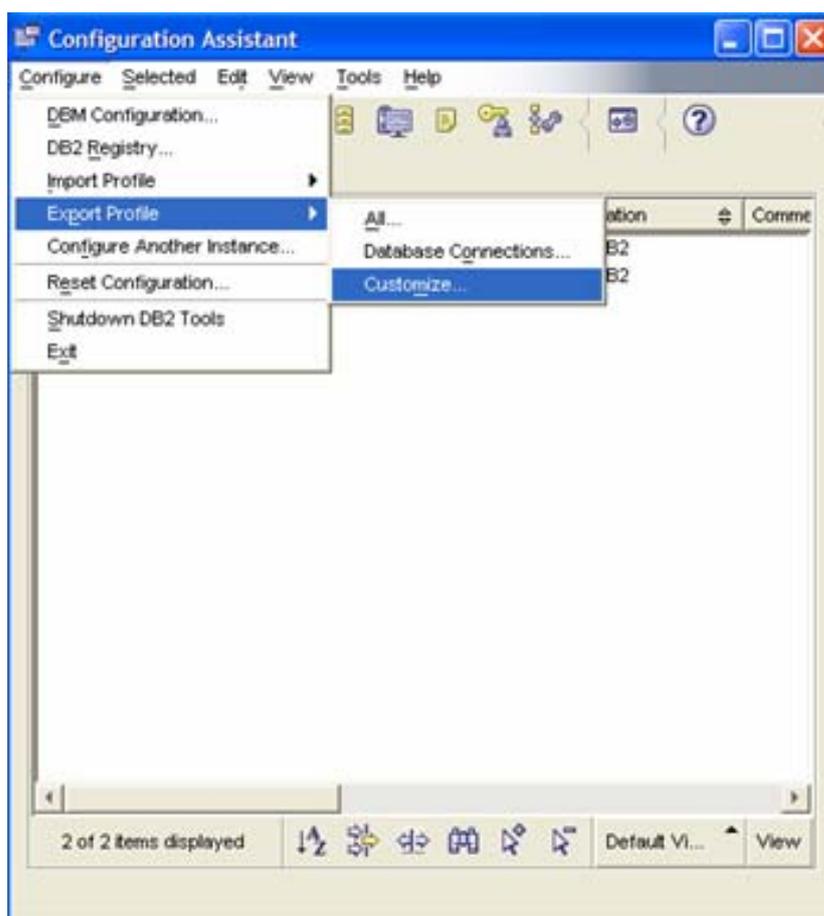


Figure 7.10 – Exporting a Profile / Figura 7.10 - Esportazione del profilo

Figure 7.11 shows the fields that need to be completed to export a profile.

La figura 7.11 mostra i campi che devono essere completati per esportare un profilo.

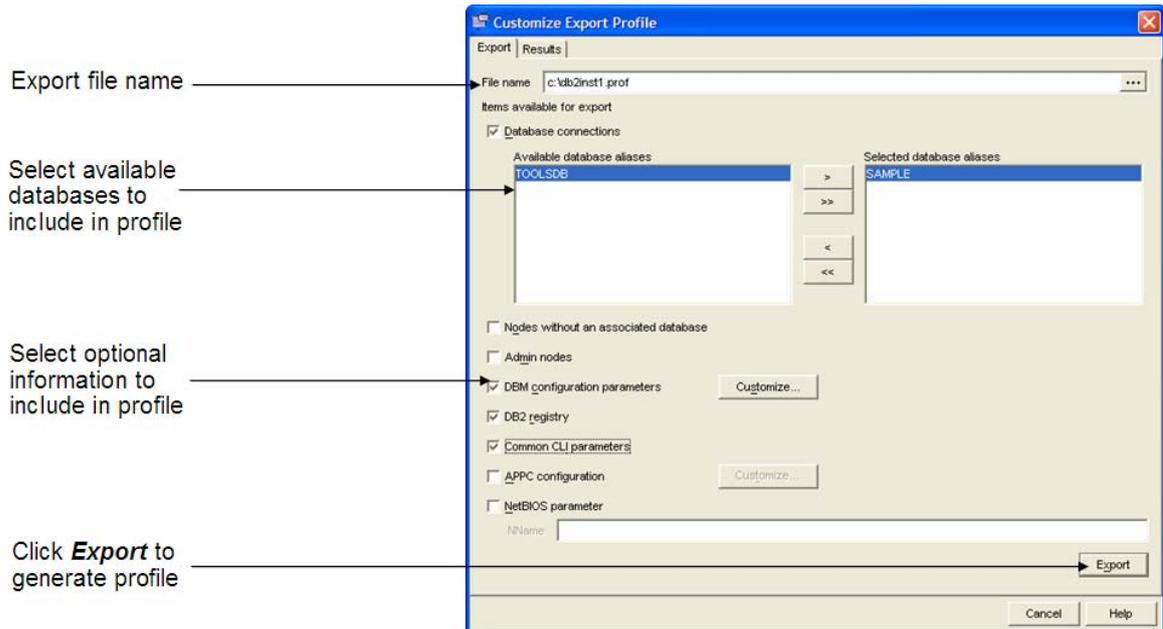


Figure 7.11 – Customize Export Profile dialog / Figura 7.11 – Personalizzare il dialogo Export Profile

Figure 7.12 show the results after clicking *Export* in the Customize Export Profile dialog.

La figura 7.12 mostra i risultati dopo aver cliccato su *Export* nel dialogo Customize Export Profile.

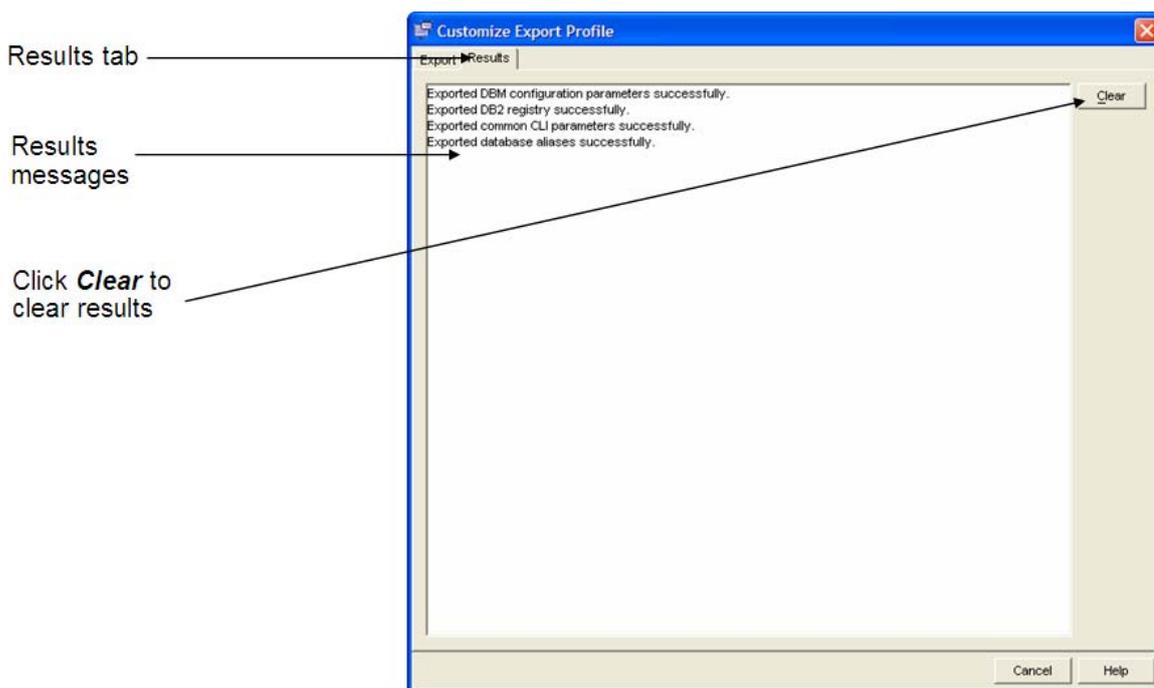


Figure 7.12 – Export Profile results / Figura 7.12 - Risultati dell'Export di un profilo

To import a customized profile from the Configuration Assistant, click on the Configure Menu, then select Import Profile > Customize, as shown in Figure 7.13.

Per importare un profilo personalizzato dal Configuration Assistant, cliccare sul menu Configure, poi scegliere *Import Profile > Customize*, come indicato in figura 7.13.

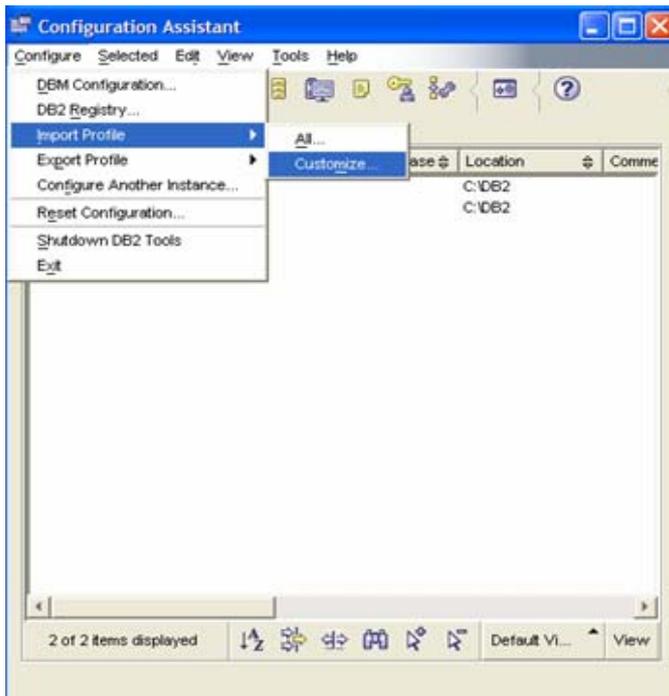


Figure 7.13 – Importing a profile / Figura 7.13 - Importazione del profilo

Figure 7.14 shows the fields that need to be completed to import a profile.

La figura 7.14 mostra i campi che devono essere completati per importare un profilo.

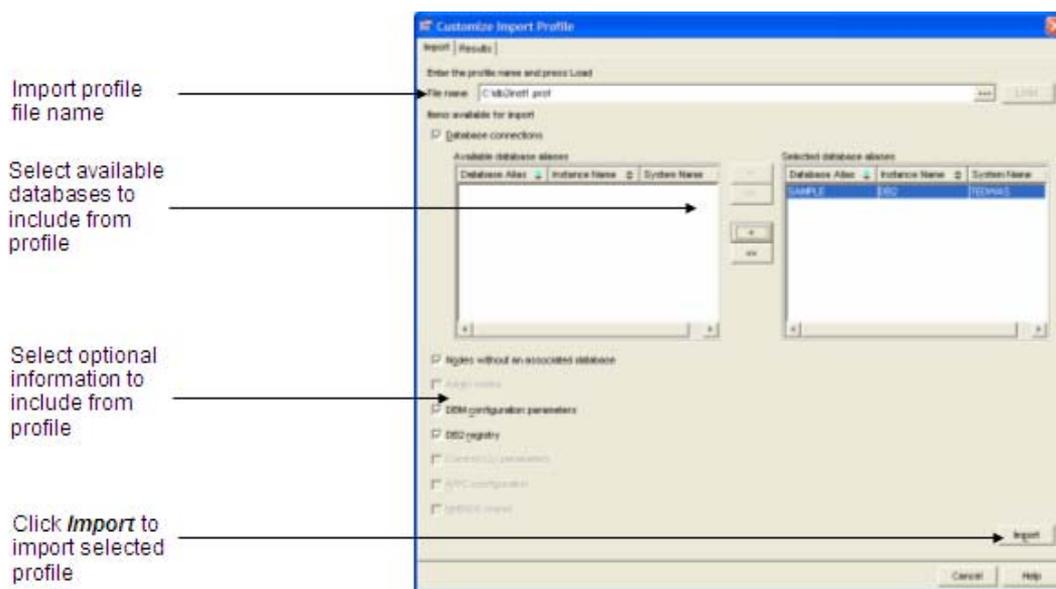


Figure 7.14 – Customize Import Profile / Figura 7.14 - Personalizzare il profilo Import

Quicklab #6: Using the Configuration Assistant / Usare il Configuration Assistant

Objective

The Configuration Assistant can be used to quickly and easily configure remote database connections. In this Quicklab, you will catalog a database residing on a remote DB2 server (represented by your neighbor's workstation, using both *Search* and *Discover* modes. Once the database is cataloged, you will be able to access it as if it were on your local system. DB2 performs all the communication processes "under the covers".

Note: This Quicklab assumes you are working within a network. If this is not the case, you can always use your own computer as both the client and server machines and follow the instructions for configuration below to connect to your own system.

Procedure

1. Ask your neighbor (or instructor) for the following information:

Remote

Database Info:

(PR) Protocol
 (IP) IP Address or hostname
 (PN) Instance Port Number
 (DB) Database Name

TCPIP

SAMPLE

Hints:

- To obtain the hostname on Windows, type *hostname*

Obiettivo

Il Configuration Assistant può essere usato per configurare rapidamente e facilmente le connessioni remote ai database. In questo Quicklab, catalogherai un database che risiede su un server DB2 remoto (rappresentato dalla workstation di lavoro del tuo vicino, usando sia la modalità *Search* che *Discover*. Una volta che il database è catalogato, potrai accedergli come se fosse sul tuo sistema locale. DB2 esegue tutti i processi relativi alla comunicazione "sotto copertura".

Nota: Questo Quicklab presuppone che stia lavorando all'interno di una rete. In caso contrario, puoi utilizzare il tuo computer sia come server che come client seguendo le istruzioni che seguono per la configurazione per il collegamento al proprio sistema.

Procedura

1. Chiedere al tuo vicino (o all'istruttore) le seguenti informazioni:

Suggerimenti:

- Per ottenere l'hostname su Windows, scrivi *hostname* da

-
- | | |
|---|---|
| <p>from a command window</p> <ul style="list-style-type: none">- To obtain the IP address on Windows, type <i>ipconfig</i> from a command window <ol style="list-style-type: none">2. Open the Configuration Assistant. (Hint: it is accessible through the <i>Start</i> menu).3. Open the <i>Selected</i> menu and select <i>Add Database Using Wizard</i>.4. On the <i>Source</i> page of the wizard, select the <i>Manually Configure a Connection to a Database</i> option. Click the <i>Next</i> button to move to the next page of the wizard.5. On the <i>Protocol</i> page of the wizard, select the TCP/IP option. Click the <i>Next</i> button to move to the next page of the wizard.6. On the <i>TCP/IP</i> page of the wizard, enter the full hostname or IP address that you wrote down in step (1). Enter the Port number you wrote down in step (1). Click the <i>Next</i> button to move to the next page of the wizard. | <p>linea di comando</p> <ul style="list-style-type: none">- Per ottenere l'indirizzo IP su Windows, scrivi <i>ipconfig</i> da linea di comando <ol style="list-style-type: none">2. Apri il Configuration Assistant. (Suggerimento: è accessibile attraverso il menu <i>Start</i>).3. Apri il menu <i>Selected</i> e scegli <i>Add Database Using Wizard</i>.4. Nella pagina <i>Source</i> del wizard, seleziona l'opzione <i>Manually Configure a Connection to a Database</i>. Clicca il tasto <i>Next</i> per andare alla pagina seguente del wizard.5. Nella pagina <i>Protocol</i> del wizard, seleziona l'opzione del TCP/IP. Clicca il tasto <i>Next</i> per andare alla pagina seguente del wizard.6. Nella pagina <i>TCP/IP</i> del wizard, inserisci l'hostname completo o l'indirizzo IP che avevi annotato al punto (1). Inserisci il numero della porta che avevi annotato al punto (1). Clicca il tasto <i>Next</i> per andare alla pagina seguente del wizard. |
|---|---|

Note: The option for *Service Name* can be used if you have an entry in the *local Services* file with a port number defined corresponding to the port the remote server instance is listening for. When you use this option, DB2 will look in the services file on the local machine, not on the server.

Nota: L'opzione per *Service Name* può essere usata se hai una voce nel file *locale Services* con un numero di porta definito che corrisponde alla porta sulla quale sta ascoltando il server remoto. Quando questa opzione è in uso DB2 cercherà nei file di servizi sul computer locale, non sul server. Devi aggiungere una

You must add an entry to this file if you want to use this option.

voce a questo file se vuoi usare quest'opzione.

7. On the *Database* page of the wizard, enter the name of the database defined on the remote server that you wrote down in step (1) in the *Database Name* field. Note how the *Database Alias* field is automatically filled out with the same value. The database alias is a name that local applications will use to connect to this database. Since you already have a local database called *SAMPLE* defined, DB2 will not let you catalog another database with the same name. You must therefore use a different alias name. For this example, change the database alias to *SAMPLE1*. You can enter an optional comment about this database if you want. Click the *Next* button to move to the next page of the wizard.
 8. On the *Data Source* page of the wizard, you can optionally register this new database (data source) as an ODBC data source. This automatically registers it in the Windows ODBC Manager for you. For this example, un-check the *Register this database for ODBC* since you will not be using ODBC. Click the *Next* button to move to the next page of the wizard.
 9. On the *Node Options* page of
7. Alla pagina *Database* del wizard, inserisci il nome del database definito sul server remoto che hai annotato al punto (1) nel campo *Database Name*. Nota come il campo *Database Alias* sia compilato automaticamente con lo stesso valore. Il database alias è un nome che le applicazioni locali useranno per connettersi a questo database. Poiché hai già definito un database locale chiamato *SAMPLE*, DB2 non ti lascerà catalogare un altro database con lo stesso nome. Devi quindi usare uno pseudonimo differente. Per questo esempio, cambia il pseudonimo del database a *SAMPLE1*. Se vuoi è possibile inserire un commento facoltativo su questo database. Clicca il tasto *Next* per andare alla pagina seguente del wizard.
 8. Alla pagina *Data Source* del wizard, puoi facoltativamente registrare questo nuovo database (data source) come fonte di dati di ODBC. Ciò la registra automaticamente per te nel ODBC Manager di Windows. Per questo esempio, deseleziona la spunta su *Register this database for ODBC* poiché non userai ODBC. Clicca il tasto *Next* per andare alla pagina seguente del wizard.
 9. Alla pagina *Node Options* del

the wizard, specify the operating system of the server where the remote database is located. Since all workstations in this lab use Microsoft Windows, ensure the *Windows* item in the drop-down list is selected. The Instance name field should be set to *DB2*. If it is not, set its value to *DB2*. Click the *Next* button to move to the next page of the wizard.

wizard, specifica il sistema operativo del server in cui il database remoto è localizzato. Poiché tutte le workstation in questo laboratorio usano Microsoft Windows, assicurati che il parametro *Windows* nel elenco a discesa sia selezionato. Il campo Instance name dovrebbe essere settato a *DB2*. Se così non è, fissa il relativo valore a *DB2*. Clicca il tasto *Next* per andare alla pagina seguente del wizard.

10. This *System Options* page of the wizard gives you the opportunity to ensure the system and hostname are correct, and to verify the operating system setting. Click the *Next* button to move to the next page of the wizard.
11. The *Security Options* page of the wizard allows you to specify where you want user authentication to take place and what method you want to use. Select the option *Use authentication value in server's DBM Configuration*. This will use the method specified by the `AUTHENTICATION` parameter in the remote Instance's configuration file. Click the *Finish* button to catalog the remote database and close the wizard. A confirmation box should appear. Click the *Test Connection* button to ensure you can connect successfully to the database.

10. Questa pagina *System Options* del wizard ti dà l'opportunità di assicurarti che il sistema e l'hostname siano corretti e di verificare il setting del sistema operativo. Clicca il tasto *Next* per andare alla pagina seguente del wizard.
11. La pagina *Security Options* del wizard ti permette di specificare dove vuoi effettuare l'autenticazione utente e che metodo vuoi usare. Seleziona l'opzione *Use authentication value in server's DBM Configuration*. Questo utilizzerà il metodo specificato dal parametro `AUTHENTICATION` nel file di configurazione dell'instance remota. Clicca il tasto *Finish* per catalogare il database remoto e per chiudere il wizard. Dovrebbe comparire un box di conferma. Clicca il tasto *Test Connection* per assicurarti che tu possa collegarti al database con successo.

Also, ensure the username and password you provide is a valid one *defined on the remote server* (since it is likely that the Server's AUTHENTICATION parameter is set to the value SERVER). If the test connection succeeds, then you have successfully cataloged the remote database. If it does not succeed, go back through the wizard and make sure all the correct values are specified. (Click the *Change* button to go back through the wizard settings).

12. Open the Control Center and try viewing the different tables in the newly cataloged remote database.
13. Go back to the Configuration Assistant and try to catalog a different database, this time using *Search the Network* option. Step through the wizard the same way you did for manually configuring the connection. Note that, on large networks, searched discovery could take a long time to return results.

Inoltre, assicurati che lo username e la password che fornisci siano valori validi *definiti sul server remoto* (poiché è probabile che il parametro AUTENTICAZIONE del server sia fissato al valore SERVER). Se il test della connessione va a buon fine significa che hai catalogato con successo il database remoto. Se non ha successo, vai indietro attraverso il wizard e assicurati che tutti i valori specificati siano corretti. (Clicca il tasto *Change* per andare indietro attraverso le schermate del wizard).

12. Apri il Control Center e prova ad osservare le diverse tabelle nel database remoto recentemente catalogato.
13. Vai indietro al Configuration Assistant e prova a catalogare un database diverso, questa volta usando l'opzione *Search the Network*. Utilizza il wizard come quando hai fatto la configurazione manuale della connessione. Nota che, sulle grandi reti, la searched discovery potrebbe richiedere molto tempo per restituire i risultati.

8

Chapter 8 – Working with Database Objects / Capitolo 8 – Lavorare con gli oggetti di database

This chapter discusses database objects such as schemas, tables, views, indexes, sequences, and so on. Some advanced database application objects such as triggers, user defined functions (UDFs) and stored procedures are discussed in Chapter 14, SQL PL stored procedures, and Chapter 15, Inline SQL PL, Triggers, and UDFs.

Questo capitolo discute gli oggetti di database come gli schemi, le tabelle, le view, gli indici, le sequenze e così via. Alcuni oggetti applicativi avanzati dei database come i trigger, le funzioni definite dall'utente (UDFs) e le stored procedure sono discusse nel capitolo 14, Stored procedure di SQL PL, e nel capitolo 15, SQL PL inline, Trigger, e UDF.

Note:

For more information about working with database objects, watch this video:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4242>

Nota:

Per maggiori informazioni sul funzionamento degli oggetti di database, guardi questo video:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4242>

8.1 Schema

Schemas are *name spaces* for a collection of database objects. They are primarily used to:

- Provide an indication of object ownership or relationship to an application
- Logically group related objects together

All DB2 database objects have a two-part fully-qualified name; the schema is the first half of that name:

Gli schemi sono *name space* per una collezione di oggetti di database.

Sono usati soprattutto per:

- Fornire un'indicazione della proprietà o del rapporto dell'oggetto con un'applicazione
- Raggruppare logicamente gli oggetti correlati

Tutti gli oggetti di database DB2 hanno un nome composto da due parti fully-qualified; lo schema è la prima metà di

<schema_name>.<object_name>

A fully-qualified object name must be unique. When you connect to a database and create or reference an object without specifying the schema, DB2 uses the user ID you connected to the database with for the schema name. For example, if you connect to the SAMPLE database with user "arfchong", and create a table using the CREATE TABLE statement:

```
CREATE TABLE artists ...
```

The fully qualified name of the created table is actually arfchong.artists.

quel nome:

Il nome di un oggetto fully-qualified deve essere unico. Quando ti colleghi ad un database e crei o fai riferimento a un oggetto senza specificare lo schema, DB2 usa l'user ID con cui sei collegato al database come nome dello schema. Per esempio, se ti colleghi al database SAMPLE con lo user "arfchong" e crei una tabella usando lo statement CREATE TABLE:

Il nome fully qualified della tabella creata è in realtà arfchong.artists.

8.2 Tables / Tabelle

A table is a collection of related data logically arranged in columns and rows. The statement below provides an example of how to create a table using the CREATE TABLE statement.

```
CREATE TABLE artists
(artno          SMALLINT    not null,
 name          VARCHAR(50)  with default 'abc',
 classification CHAR(1)     not null,
 bio           CLOB(100K)   logged,
 picture       BLOB(2M)     not logged compact
)
IN mytbls1
```

In the following sections, we will describe the main parts of this CREATE TABLE statement

Una tabella è una collezione di dati correlati che sono organizzati logicamente in colonne e in righe. Lo statement qui sotto offre un esempio di come creare una tabella usando lo statement CREATE TABLE.

Nelle seguenti sezioni, descriveremo le parti principali dello statement CREATE TABLE.

8.2.1 Data types / Tipi di data

Figure 8.1 lists the data types supported in DB2.

La figura 8.1 elenca i tipi di dati supportati in DB2.

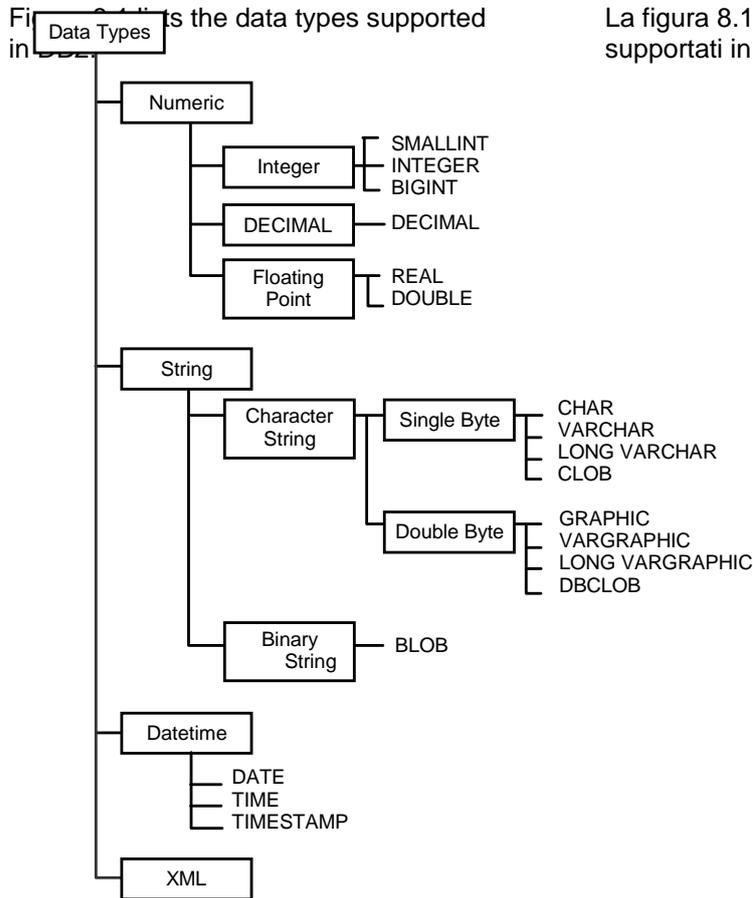


Figure 8.1 – DB2 built-in data types / Figura 8.1 - Tipi di dati built-in di DB2

Large Object (LOB) data types / Large Object (LOB) data type

Large object data types are used to store large character strings, large binary strings or files as shown in Figure 8.2.

I tipi large object sono usati per memorizzare stringhe di caratteri, stringhe binarie o file di grandi dimensioni come indicato in figura 8.2.

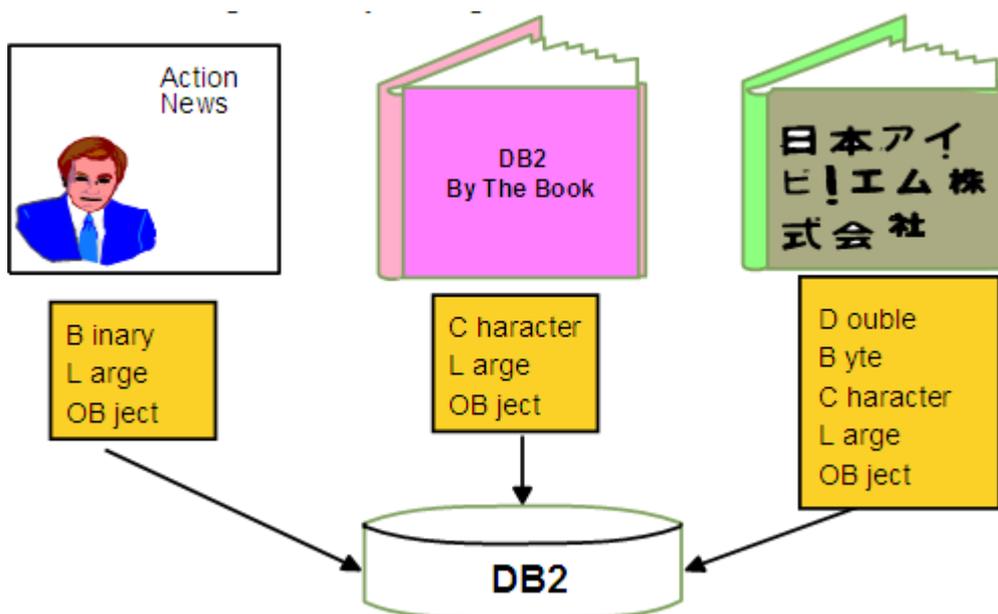


Figure 8.2 – LOBs data types / Figura 8.2 - Tipi di dati LOB

These large object binaries are usually abbreviated for clarity: a binary large object is a BLOB, a character large object is a CLOB, and a double byte character large object is also known as a DBCLOB.

Questi oggetti binari di grandi dimensioni sono abbreviati solitamente per chiarezza: un oggetto binario è un BLOB, un oggetto di caratteri è un CLOB e un grande oggetto di caratteri double-byte è anche conosciuto come DBCLOB.

User-defined types / Tipi definiti dall'utente

DB2 allows you to define your own data types, based on the built-in data types. These are known as user-defined types (UDTs). UDTs are useful when:

- There is a need to establish context for values
- There is a need to have DB2 enforce data typing

DB2 permette di definire tipi di dati propri, basati su tipi di dati built-in. Questi sono conosciuti come tipi definiti dall'utente (UDTs). Gli UDTs sono utili quando:

- C'è la necessità di stabilire un contesto per i valori
- C'è la necessità di applicare a DB2 un data typing

The following statements illustrate an example of how and when to use UDTs:

```
CREATE DISTINCT TYPE POUND AS INTEGER WITH COMPARISONS

CREATE DISTINCT TYPE KILOGRAM AS INTEGER WITH COMPARISONS

CREATE TABLE person
  (f_name    VARCHAR(30),
   weight_p  POUND NOT NULL,
   weight_k  KILOGRAM NOT NULL )
```

In this example, two UDTs are created: POUND and KILOGRAM. Both are built based on the built-in data type INTEGER. The WITH COMPARISONS clauses defined as part of the syntax indicate that casting functions with the same name as the data types will also be created.

The table `person` uses the two new UDTs in columns `weight_p` and `weight_k`, respectively. If we now issue the following statement:

```
SELECT F_NAME FROM PERSON
WHERE weight_p > weight_k
```

You will receive an error message because two columns with different data types are being compared. Even though `weight_p` and `weight_k` use the POUND and KILOGRAM data types respectively, both of which were created based on the INTEGER data type, by creating UDTs, you make this type of comparison impossible. This is exactly what you want, because in real life, what would a comparison between pounds and kilograms mean? It would not make sense.

In the next example, you would like to compare the column `weight_p` with an

I seguenti statement illustrano un esempio di come e quando usare gli UDTs:

In questo esempio, due UDTs sono generati: POUND e KILOGRAM. Entrambi sono costruiti basandosi su tipi di dati built-in INTEGER. Le clause WITH e COMPARISONS definite come componente della sintassi indicano che saranno inoltre create le funzioni di casting con lo stesso nome dei tipi di dati.

La tabella `person` utilizza i due nuovi UDTs nelle colonne `weight_p` e `weight_k`, rispettivamente. Se ora inseriamo il seguente statement:

Riceverai un messaggio di errore perché due colonne con diversi tipi di dati sono confrontate. Anche se `weight_p` e `weight_k` usano i tipi di dati POUND e KILOGRAM rispettivamente, entrambi sono stati creati basandosi sul tipo di dato INTEGER; creare UDTs, rende questo tipo di confronto impossibile. Ciò è esattamente quello che vuoi, perché in realtà, cosa significherebbe un confronto fra le libbre e i chilogrammi? Non avrebbe senso.

Nell'esempio seguente, vorresti paragonare la colonna `weight_p` con

integer; however, these two data types are different, and therefore you would receive an error unless you use a casting function.

As you can see from the statement below, we use the casting function POUND() so that this comparison is possible. As indicated earlier, the POUND() casting function was created with the UDT when invoking the WITH COMPARISONS clause of the CREATE DISTINCT TYPE statement.

```
SELECT F_NAME FROM PERSON
WHERE weight_p > POUND(30)
```

Null values / Valori nulli

A null value represents an unknown state. However, the CREATE TABLE statement can define a column using the NOT NULL clause. This ensures that the column contains a known data value. You can also specify a default value for the column if NOT NULL is declared. The next statement provides examples of this behaviour:

```
CREATE TABLE Staff (
  ID          SMALLINT NOT NULL,
  NAME       VARCHAR(9),
  DEPT       SMALLINT NOT NULL with default 10,
  JOB        CHAR(5),
  YEARS      SMALLINT,
  SALARY     DECIMAL(7,2),
  COMM       DECIMAL(7,2) with default 15
)
```

un numero intero; tuttavia, questi due tipi di dati sono differenti e quindi riceveresti un errore a meno che non usi una funzione di casting.

Come puoi vedere dallo statement qui sotto, useremo la funzione di casting POUND() in modo che questo confronto sia possibile. Come indicato prima, la funzione di casting POUND() è stata creata con l'UDT quando si invoca la clausola WITH COMPARISONS dello statement.

Un valore nullo rappresenta una condizione sconosciuta. Tuttavia, lo statement CREATE TABLE può definire una colonna usando la clausola NOT NULL. Ciò assicura che la colonna contenga un valore di dati conosciuto. Puoi anche specificare un valore di default per la colonna se NOT NULL è dichiarato. Lo statement seguente fornisce un esempio di questo comportamento:

8.2.2 Identity columns / Colonne identity

An identity column is a numeric column which automatically generates a unique numeric value for each inserted row. There can only be one identity column per table.

There are two ways to generate values for an identity column, depending on how it was defined:

- **Generated always:** values are always generated by DB2. Applications are not allowed to provide an explicit value.
- **Generated by default:** values can be explicitly provided by an application or, if no value is given, then DB2 generates one. DB2 cannot guarantee uniqueness. This option is intended for data propagation, and for the unloading and reloading of a table.

Let's take a look at the following example:

```
CREATE TABLE subscriber(subscriberID INTEGER GENERATED ALWAYS AS
                           IDENTITY (START WITH 100
                           INCREMENT BY 100) ,
                           firstname VARCHAR(50) ,
                           lastname  VARCHAR(50) )
```

In the example, the column *subscriberID* is an INTEGER defined as an identity column that is always generated. The value generated will start from 100, and it will be incremented by 100.

Una colonna identity (identificativo) è una colonna numerica che crea automaticamente un valore numerico unico per ogni riga inserita. Ci può essere soltanto una colonna identity per tabella.

Ci sono due metodi per generare i valori per una colonna identity, a seconda di come è stata definita:

- **Generated always:** i valori sono generati sempre da DB2. Alle applicazioni non è permesso fornire un valore esplicito.
- **Generated by default:** i valori possono essere forniti esplicitamente da un'applicazione o, se nessun valore è dato, DB2 ne genera uno. DB2 non può garantire l'unicità. Questa opzione è intesa per la propagazione dei dati e per cancellare e ricaricare una tabella.

Diamo un'occhiata al seguente esempio:

Nell'esempio, la colonna *subscriberID* è un INTEGER definito come una colonna identity che è generata sempre. Il valore generato comincerà da 100 e sarà incrementato di 100.

8.2.3 SEQUENCE objects / Oggetti di SEQUENCE

Sequence objects generate a unique number across the database. Unlike identity columns, sequences are independent of tables. The following statements provide an example:

```
CREATE TABLE t1 (salary int)
```

```
CREATE SEQUENCE myseq
  START WITH 10
  INCREMENT BY 1
  NO CYCLE
```

```
INSERT INTO t1 VALUES (nextval for myseq)
INSERT INTO t1 VALUES (nextval for myseq)
INSERT INTO t1 VALUES (nextval for myseq)
```

```
SELECT * FROM t1
```

```
SALARY
```

```
-----
          10
          11
          12
  3 record(s) selected.
```

```
SELECT prevval for myseq FROM sysibm.sysdummy1
```

```
1
-----
          12
  1 record(s) selected
```

PREVVAL provides the current value of the sequence, while NEXTVAL provides the next value.

The above example also uses SYSIBM.SYSDUMMY1. This is a system catalog table that contains one

Gli oggetti di sequenza (Sequence objects) generano un numero unico attraverso il database. Diverso delle colonne di identità, le sequenze sono indipendenti dalle tabelle. I seguenti statement forniscono un esempio:

PREVVAL fornisce il valore corrente della sequenza, mentre NEXTVAL fornisce il valore seguente.

L'esempio di sopra inoltre usa SYSIBM.SYSDUMMY1. Questa è una tabella del catalogo di sistema che

column and one row. It can be used in situations where a query requires that the output based on only one value is returned. System catalog tables are described in the next section.

contiene una colonna ed una riga. Può essere utilizzata nelle situazioni in cui una query richieda che sia ritornato un output basato soltanto su un valore. Le tabelle del catalogo di sistema sono descritte nella sezione seguente.

8.2.4 System catalog tables / Tabelle del catalogo di sistema

Each database has its own system catalog tables and views. These store *metadata* about the database objects. You can query these tables just like any normal database tables. Three schemas are used to identify the system catalog tables:

- SYSIBM: base tables, optimized for DB2 use
- SYSCAT: views based on the SYSIBM tables, optimized for ease of use
- SYSSTAT: database statistics

Ogni database ha le proprie tabelle e view del catalogo di sistema. Questi memorizzano i *metadati* degli oggetti di database. Potete applicare query a queste tabelle come a tutte le tabelle normali dei database. Tre schemi sono usati per identificare le tabelle del catalogo del sistema:

- SYSIBM: tabelle base, ottimizzate per l'uso di DB2
- SYSCAT: view basate sulle tabelle di SYSIBM, ottimizzate per facilitarne l'utilizzo
- SYSSTAT: statistiche del database

The following are some examples of catalog views:

- SYSCAT.TABLES
- SYSCAT.INDEXES
- SYSCAT.COLUMNS
- SYSCAT.FUNCTIONS
- SYSCAT.PROCEDURES

Di seguito alcuni esempi delle viste del catalogo:

8.2.5 Declared temporary tables / Tabelle provvisorie dichiarate

Declared temporary tables are tables created in memory and are used by an application and then dropped automatically when the application terminates. These tables can only be accessed by the application that created them. No entry exists in any DB2 catalog table. Accessing these tables provides very efficient performance because there is no catalog contention, no locking of rows, no default logging (logging is optional), and no authority checking. There is also index support for these temporary tables, that is, any standard index can be created on a temporary table. You can also run RUNSTATS against these tables.

Le tabelle provvisorie dichiarate sono tabelle create in memoria e usate da un'applicazione ed quindi cancellate automaticamente quando l'applicazione termina. Queste tabelle possono essere raggiunte soltanto dall'applicazione che le ha create. Nessuna entry esiste in tutta la tabella del catalogo DB2. L'accesso a queste tabelle fornisce prestazioni molto efficienti perché non c'è conflitto del catalogo, locking delle righe, nessun logging di default (il logging è facoltativo) e nessun controllo di autorizzazione. C'è inoltre il supporto sugli indici per queste tabelle provvisorie, cioè, qualsiasi indice standard può essere creato su una tabella provvisoria. Puoi anche eseguire RUNSTATS su queste tabelle.

Declared temporary tables reside inside a user temporary table space, which must be defined prior to creating any declared temporary tables. The following statements provide an example on how to create three declared temporary tables:

Le tabelle provvisorie dichiarate risiedono all'interno di uno spazio di tabella provvisoria dell'utente, che deve essere definito prima di creare qualunque tabella provvisoria dichiarata. I seguenti statement forniscono un esempio su come creare tre tabelle provvisorie dichiarate:

```
CREATE USER TEMPORARY TABLESPACE apptemps
    MANAGED BY SYSTEM USING ('apptemps');
```

```
DECLARE GLOBAL TEMPORARY TABLE tempemployees
    LIKE employee NOT LOGGED;
```

```
DECLARE GLOBAL TEMPORARY TABLE tempdept
    (deptid CHAR(6), deptname CHAR(20))
    ON COMMIT DELETE ROWS NOT LOGGED;
```

```
DECLARE GLOBAL TEMPORARY TABLE tempprojects
    AS ( fullselect ) DEFINITION ONLY
    ON COMMIT PRESERVE ROWS NOT LOGGED
    WITH REPLACE IN TABLESPACE apptemps;
```

When a declared temporary table is created, its schema is SESSION, and must be specified. The user ID used to create a temporary table will have all privileges on the table. Each application which creates a temporary table will have its own independent copy as shown in Figure 8.5.

Quando una tabella provvisoria dichiarata è creata, il relativo schema è SESSION e deve essere specificato. L'utente ID usato per creare una tabella provvisoria avrà tutti i privilegi sulla tabella. Ogni applicazione che crea una tabella provvisoria avrà la propria copia indipendente secondo le indicazioni di figura 8.5.



Figure 8.5 – Scope of declared global temporary tables / Figura 8.5 - Scope di tabelle provvisorie dichiarate globali

Quicklab #7: Creating a new table / Creare un nuova tabella

Objective

So far, you have been using the existing tables in the *SAMPLE* database to illustrate concepts. Eventually, you will need to create your own tables in a database. In this Quicklab, you will use the *Create Table Wizard* to create two new tables in the *SAMPLE* database.

Procedure

1. Launch the Create Table Wizard as previously shown in the presentation. (*Control Center -> All Databases -> SAMPLE -> (right-click) Tables object -> Create ... option*)
2. Define the table name, column definitions, and any constraints. The table will be used to store information about the office supplies used by a project in the *SAMPLE* database. Each time supplies are purchased, a row will be added to this table. The table will have six columns:
 - *product_id*: unique identifier of the item being purchased
 - *description*: description of the item
 - *quantity*: the quantity purchased
 - *cost*: the cost of the item
 - *image*: a picture of the item (if available)
 - *project_num*: the project

Obiettivo

Finora, hai utilizzato le tabelle esistenti nel database *SAMPLE* per illustrare i concetti. Finalmente, dovrai creare le tue tabelle in un database. In questo Quicklab, userai il *Create Table Wizard* per creare due nuove tabelle nel database *SAMPLE*.

Procedura

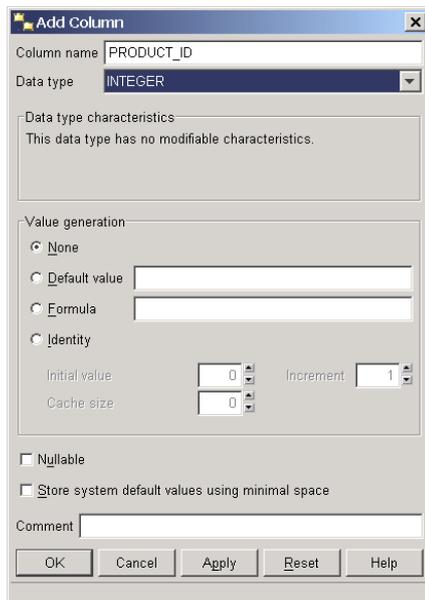
1. Lancia il Create Table Wizard come indicato precedentemente nella presentazione. L'opzione... *Control Center -> All Databases -> SAMPLE -> (right-click) Tables object -> Create*
2. Definisci il nome della tabella, le definizioni delle colonne e tutti i vari vincoli (constraint). La tabella sarà usata per memorizzare le informazioni sulle forniture di un ufficio usati da un progetto nel database *SAMPLE*. Ogni volta che viene acquistata una fornitura, una riga verrà aggiunta a questa tabella. La tabella avrà sei colonne:
 - *product_id*: identificativo univoco dell'articolo acquistato
 - *description*: descrizione dell'articolo
 - *quantity*: quantità acquistata
 - *cost*: costo del prodotto
 - *image*: immagine dell'articolo (se disponibile)
 - *project_num*: progetto per cui

this product has been purchased for

questo prodotto è stato acquistato

3. In the first page of the wizard, for the schema name, enter the user ID you are currently logged on as, and use the following table name: *SUPPLIES*. You can also optionally enter a comment. Click the *Next* button to continue to the next page of the wizard.
4. From this page, you can add columns to the table. Click the *ADD* button to add columns.

3. Nella prima pagina del wizard, per il nome dello schema, inserisci l'utente con cui sei loggato e utilizza il seguente nome di tabella: *SUPPLIES*. Puoi anche facoltativamente inserire un commento. Clicca sul tasto *Next* per continuare alla pagina seguente del wizard.
4. Da questa pagina, puoi aggiungere le colonne alla tabella. Clicca sul tasto *ADD* per aggiungere le colonne.



Enter the column name "*product_id*" and select the data type: *INTEGER*. Uncheck *Nullable*, and click the *Apply* button to define the column.

Inserisci il nome "*product_id*" della colonna e seleziona il tipo di dati: *INTEGER*. Deseleziona *Nullable* e clicca sul tasto *Apply* per definire la colonna.

Repeat this step for the remaining columns of the table using the options shown in the table below. Once all columns have been added (Applied), click the *OK* button and the list of the columns you just created should be summarized. Click the *Next* button to continue to the next page of the wizard.

Ripeti questo punto per le colonne restanti della tabella usando le opzioni indicate nella tabella qui sotto. Una volta che tutte le colonne sono state aggiunte (Applied), clicca *OK* e la lista delle colonne che hai appena creato dovrebbe essere ricapitolata. Clicca sul tasto *Next* per continuare alla pagina seguente del wizard.

| Column Name | Nome di colonna | Attributes |
|---------------------------|----------------------------|---|
| product_id (completed) | product_id (completato) | INTEGER, NOT NULL |
| description | descrizione | VARCHAR, length 40, NOT NULL |
| quantity | quantità | INTEGER, NOT NULL |
| cost | costo | DECIMAL, Precision 7, Scale 2, NOT NULL |
| image | immagine | BLOB, 1MB, NULLABLE, NOT LOGGED |
| project_num | project_num | CHAR, length 6, NOT NULL |

Note: The NOT LOGGED option can be specified when declaring LOB columns. It is mandatory for columns greater than 1GB in size. It is also generally recommended for LOBs larger than 10MB as changes to large columns can quickly fill the log file. Even if NOT LOGGED is used, changes to LOB files during a transaction can still be successfully rolled back. Also notice that the *image* column is the only one defined as a "NULLABLE" column. Why do you think that the column was defined like this?

Nota: L'opzione NOT LOGGED può essere specificata quando dichiarare le colonne LOB. È obbligatorio per le colonne di dimensioni maggiori di 1GB. Inoltre viene suggerita generalmente per i LOB più grandi di 10MB, dato che i cambiamenti alle colonne di grandi dimensioni possono riempire rapidamente il file log. Anche se NOT LOGGED è usato, i cambiamenti ai file LOB durante la transazione possono ancora essere annullati con successo. Inoltre nota che la colonna *image* è l'unica definita come colonna "NULLABLE". Perché pensi che la colonna sia stata definita così?

5. At this point, all the mandatory information for creating a table has been provided. By skipping the other pages, you are choosing the default values for those options. You can always add keys and constraints after a table has been created.
 6. Add a constraint to the table to restrict values on the *quantity* column. On the *Constraint* page of the wizard, click the *ADD* button. In the *Check Name* field, enter: *valid_quantities*. In the *Check Condition* field, enter: *quantity > 0*. Click the *OK* button. You should see a summary of the constraint you just added in the *Constraint* page of the wizard. Click the *Next* button to continue to the next page of the wizard.
 7. You can continue going through the wizard, changing the other parameters of the table. Alternatively, you can skip to the *Summary* page, or simply click the *Finish* button to create the table.
 8. From Control Center, click on the *Tables* folder under the *SAMPLE* database in the Object Tree pane. The table you just created should now appear in the list. It might be necessary to refresh the Control Center view in order to see the changes.
5. A questo punto, tutte le informazioni obbligatorie per la creazione della tabella sono state fornite. Salta le altre pagine, scegliendo i valori di default per quelle opzioni. Puoi comunque aggiungere sempre chiavi e vincoli dopo che una tabella è stata creata.
 6. Aggiungi un vincolo alla tabella per limitare i valori sulla colonna *quantity*. Alla pagina *Constraint* del wizard, clicca *ADD*. Nel campo *Check Name*, inserisci: *valid_quantities*. Nel campo *Check Condition*, inserisci: *quantity > 0*. Clicca *OK*. Dovresti vedere un riassunto dei vincoli che hai appena aggiunto nella pagina *Constraint* del wizard. Clicca sul tasto *Next* per continuare alla pagina seguente del wizard.
 7. Puoi continuare a passare attraverso il wizard, cambiando gli altri parametri della tabella. Alternativamente, puoi saltare direttamente alla pagina *Summary*, o semplicemente cliccare *Finish* per creare la tabella.
 8. Dal Control Center, clicca sulla cartella *Tables* sotto il database *SAMPLE* nell'albero degli oggetti. La tabella che hai appena creato dovrebbe ora comparire nella lista. Potrebbe essere necessario aggiornare la vista del Control Center per vedere i cambiamenti.

8.3 Views / View

A view is a representation of the data in tables. The data for the view is not stored separately, but obtained when the view is invoked. Nested views, that is, a view created based on other views, are supported. All information about views is kept in the following DB2 catalog views: SYSCAT.VIEWS, SYSCAT.VIEWDEP, and SYSCAT.TABLES. Here is an example of how to create and use a view.

Una view (vista) è una rappresentazione dei dati nelle tabelle. I dati per le view non sono memorizzati separatamente, ma sono ottenuti quando la view è invocata. Le view annidate, cioè, una view creata basandosi su altre view, sono supportate. Tutte le informazioni sulle view sono mantenute nelle seguenti view del catalogo DB2: SYSCAT.VIEWS, SYSCAT.VIEWDEP e SYSCAT.TABLES. Ecco qui un esempio di come creare ed usare una view.

```
CONNECT TO MYDB1;

CREATE VIEW MYVIEW1
  AS SELECT ARTNO, NAME, CLASSIFICATION
  FROM ARTISTS;

SELECT * FROM MYVIEW1;
```

Output :

| ARTNO | NAME | CLASSIFICATION |
|-------|-----------|----------------|
| 10 | HUMAN | A |
| 20 | MY PLANT | C |
| 30 | THE STORE | E |
| ... | | |

8.4 Indexes / Indici

An index is an ordered set of keys each of which points to a row in a table. An index allows for uniqueness, and it also improves performance. Some of the characteristics that you can define on indexes:

- The index order can be ascending or descending
- The index keys can be unique or non-unique
- Several columns can be used for the index (this is called a *compound index*)

Un indice è un insieme ordinato di chiavi, ognuna delle quali indica una riga in una tabella. Un indice permette l'unicità ed inoltre migliora le prestazioni. Alcune delle caratteristiche che puoi definire sugli indici:

- L'ordine dell'indice può essere ascendente o discendente
- Le chiavi dell'indice possono essere uniche o non-uniche
- Più colonne possono essere usate per l'indice (questo è chiamato *indice compound*)

- If the index and the physical data are clustered in similar index sequence, they are a cluster index
- Se l'indice ed i dati fisici sono clustered (raggruppati) in una simile sequenza di indice, sono un indice cluster

For example:

```
CREATE UNIQUE INDEX artno_ix ON artists (artno)
```

Per esempio:

8.4.1 Design Advisor

The Design Advisor is an excellent tool to advise you on the optimal design of your database for a given SQL workload. The design advisor can help you with the design of your indexes, Materialized Query Tables (MQTs), Multi-dimension clustering (MDC), and the database partitioning feature. The Design Advisor is invoked from the Control Center; right-click on a database and select *Design Advisor* as shown in Figure 8.6.

Il Design Advisor è uno strumento eccellente per consigliarti sul design ottimale del tuo database per un dato workload SQL. Il Design Advisor può aiutarti con: il design dei tuoi indici, le Materialized Query Tables (MQTs), il Clustering Multi-dimension (MDC) e la funzione di partitioning del database. Il Design Advisor è invocato dal Control Center; fai clic destro su un database e seleziona *Design Advisor* secondo le indicazioni della figura 8.6.

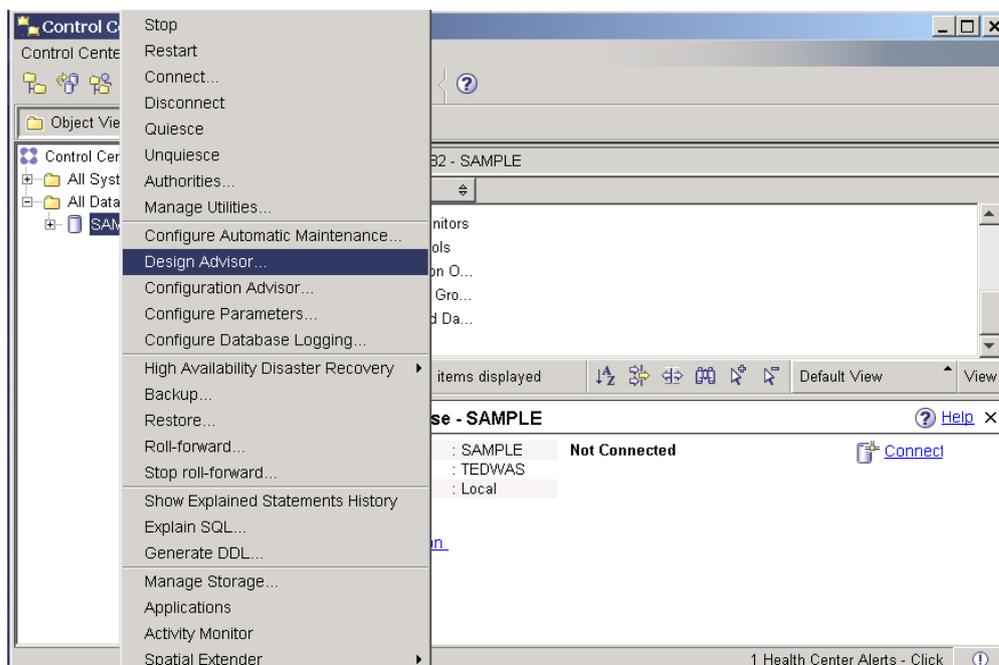


Figure 8.6 – Invoking the Design Advisor from the Control Center / Figura 8.6 - Invocare il Design Advisor dal Control Center

Figure 8.7 shows the Design Advisor. Follow the steps in this wizard to obtain the design recommendations from DB2.

La figura 8.7 mostra il Design Advisor. Segui i punti in questo wizard per ottenere le specifiche per il design da DB2.

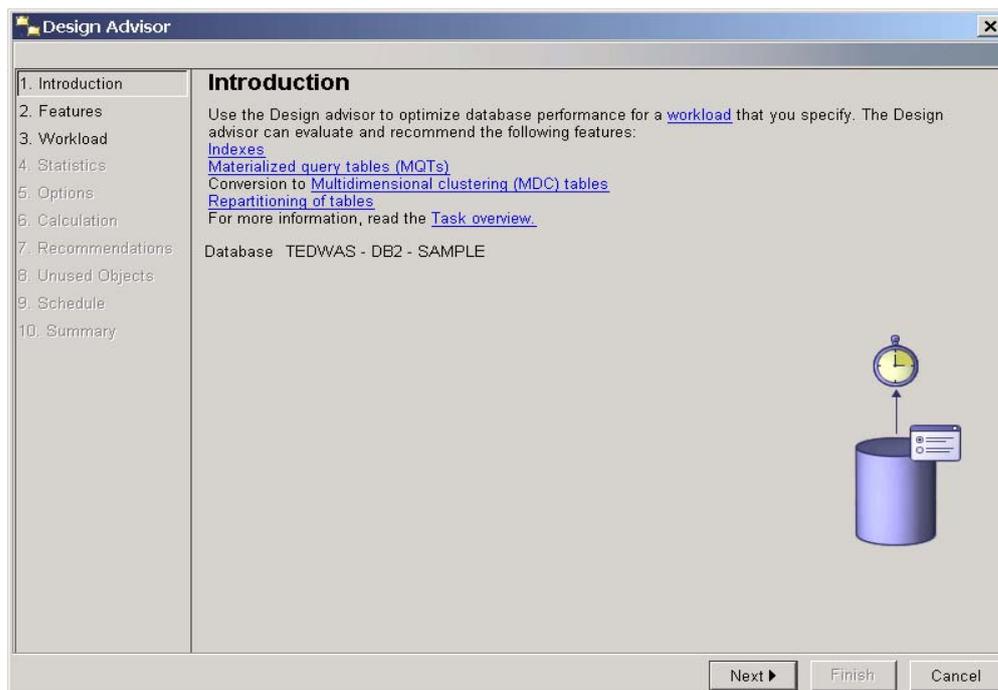


Figure 8.7 –The Design Advisor / Figura 8.7 - Il Design Advisor

8.5 Referential integrity / Integrità referenziale

Referential integrity allows your database to manage relationships between tables. You can establish parent-child type of relationships between tables as shown in Figure 8.8. In the figure, there are two tables, DEPARTMENT and EMPLOYEE, related by the department number. The WORKDEPT column in the EMPLOYEE table can only contain department numbers that already exist in the DEPARTMENT table. This is because in this example, the DEPARTMENT table is the parent table, and the EMPLOYEE table is the child, or dependent, table. The figure also shows the necessary CREATE TABLE statement for the EMPLOYEE table needed to establish the relationship.

L'integrità referenziale permette al tuo database di gestire le relazioni fra le tabelle. Puoi stabilire il tipo di relazione padre-figlio fra le tabelle secondo le indicazioni di figura 8.8. Nella figura, ci sono due tabelle, DEPARTMENT e EMPLOYEE, collegate dal numero di reparto. La colonna WORKDEPT nella tabella EMPLOYEE può contenere soltanto i numeri di reparto che già esistono nella tabella DEPARTMENT. Ciò perché in questo esempio, la tabella DEPARTMENT è la tabella padre e la tabella EMPLOYEE è la tabella figlio, o dipendente. La figura inoltre mostra la necessità dello statement CREATE TABLE per la tabella EMPLOYEE, fondamentale per stabilire la relazione.

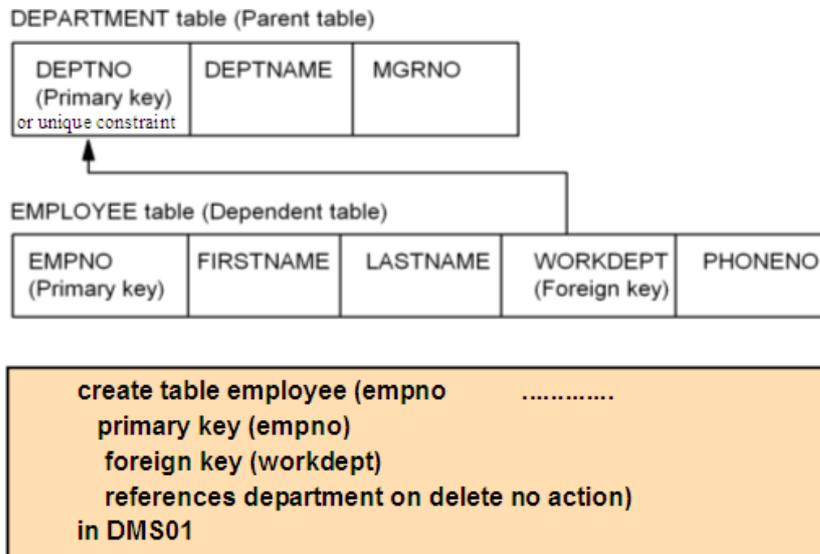


Figure 8.8 –An example of referential integrity between tables / Figura 8.8 - Un esempio d'integrità referenziale fra le tabelle

In referential integrity, the following concepts are often used:

Nell'integrità referenziale, i seguenti concetti sono usati spesso:

| Concept / Concetto | Description | Descrizione |
|--------------------------------------|--|---|
| Parent table / Tabella padre | A controlling data table in which the parent key exists | Una tabella dati controllante in cui la chiave padre esiste |
| Dependent table / Tabella dipendente | A table dependent on the data in the parent table. It also contains a foreign key. For a row to exist in a dependent table, a matching row must already exist within a parent table. | Una tabella dipendente dai dati nella tabella padre. Contiene inoltre una chiave esterna. Affinché una riga esista in una tabella dipendente, una riga corrispondente deve esistere già all'interno di una tabella padre. |
| Primary Key / Chiave primaria | Defines the parent key of the parent table. It cannot contain NULL values and values must be unique. A primary key consists of one or more columns within a table. | Definisce la chiave padre della tabella padre. Non può contenere valori NULL e questi valori devono essere univoci. Una chiave primaria consiste di una o più colonne all'interno di una tabella. |
| Foreign Key / Chiave esterna | References the primary key of a parent table | Riferimento alla chiave primaria di una tabella padre |

Data in tables can be related to data in one or more tables with referential integrity. Constraints can also be imposed on data values so that they conform to a certain property or business rule. For example, if a table column stores the sex of a person, the constraint can enforce that the only values allowed are "M" for male, and "F" for female.

I dati nelle tabelle possono essere collegati con dati in una o più tabelle con integrità referenziale. Le costrizioni possono anche essere imposte a valori di dati in modo che si conformino ad una determinata proprietà o regola di business. Per esempio, se una colonna della tabella memorizza il sesso di una persona, il vincolo può imporre che gli unici valori permessi siano "M" per il maschio e "F" per la femmina.

9

Chapter 9 – Data Movement Utilities / Capitolo 9 – Utility di Data Movement

The tools or commands described in this section are used to move data within the same database or across databases in the same or different platforms. Figure 9.1 provides an overview of the data movement utilities

I tool o i comandi descritti in questa sezione sono utilizzati per spostare i dati all'interno dello stesso database o attraverso i database nelle stesse o su diverse piattaforme. La figura 9.1 fornisce una descrizione delle utility per il movimento di dati:

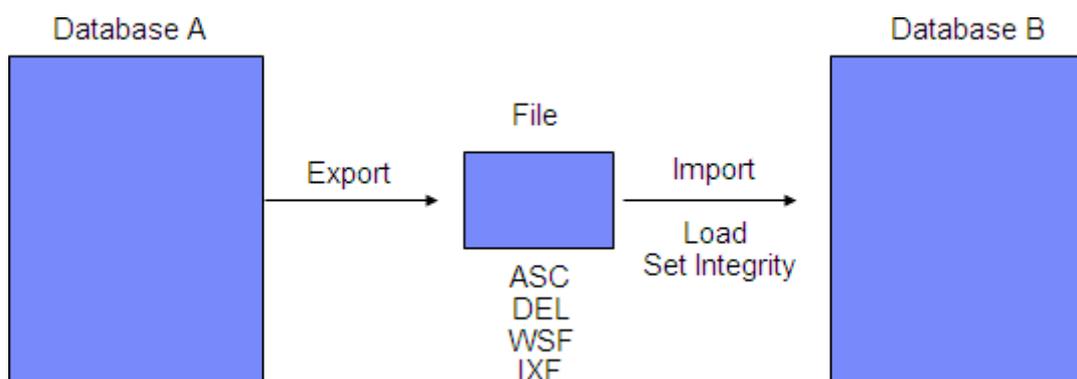


Figure 9.1 – Data movement utilities / Figura 9.1 - Programmi di utilità del movimento di dati

In Figure 9.1 there are two databases, database A, and B. Using the EXPORT utility, one can export the data from a table into a file. The file can have any of these formats:

ASC = ASCII
DEL = Delimited ASCII
WSF = Worksheet format
IXF = Integrated Exchange Format

ASC and DEL files are text files that can be opened and reviewed in any text editor. WSF is a format that can be

Nella figura 9.1 ci sono due database, database A e B. Usando l'utility EXPORT, si possono esportare i dati da una tabella in un file. Il file può avere alcuni di questi formati:

I file ASC e DEL sono file di testo che possono essere aperti ed esaminati in qualsiasi editor. WSF è un formato che

move data to spreadsheets such as Excel or Lotus® 1-2-3. IXF is a format that not only includes the data but also the Data Definition Language (DDL) of the table in question. This is convenient because when the table needs to be reconstructed, it can be done directly from a file with an IXF format, while this would not be possible if you use the other formats.

Once the data has been exported to a file, the IMPORT utility can be used to import the data from the file into another table. The table must exist beforehand for the ASC, DEL and WSF format, but it does not need to exist for the IXF format. Another method to load the data into a table is to use the LOAD utility. The LOAD utility is faster as it goes directly to the database pages without interacting with the DB2 engine; however, this method does not make a check for constraints, and triggers will not be fired. To guarantee consistency of the data loaded using LOAD, the SET INTEGRITY command is often used afterwards.

The next sections describe the EXPORT, IMPORT, and LOAD utilities in more detail.

può essere usato per migrare i dati verso spreadsheet come Excel o Lotus® 1-2-3. IXF è un formato che non solo comprende i dati ma anche il Data Definition Language (DDL) della tabella in questione. Questo è conveniente perché quando la tabella deve essere ricostruita, può essere fatto direttamente da un file con un formato IXF, mentre questo non sarebbe possibile usando gli altri formati.

Una volta che i dati sono stati esportati in un file, l'utility IMPORT può essere usata per importare i dati da un file in un'altra tabella. La tabella deve esistere in anticipo per il formato ASC, DEL e WSF, ma non deve esistere per il formato IXF. Un altro metodo per caricare i dati in una tabella è di usare l'utility LOAD. L'utility LOAD è più veloce poiché va direttamente alle pagine del database senza interagire con il motore DB2; tuttavia, questo metodo non fa un controllo sulle restrizioni ed i trigger non saranno attivati. Per garantire la consistenza dei dati caricati usando LOAD, spesso di seguito si usa il comando INTEGRITY.

Le sezioni seguenti descrivono più in dettaglio le utility EXPORT, IMPORT, e LOAD.

Note:

For more information about working with data movement utilities, watch this video:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4262>

Nota:

Per maggiori informazioni sul funzionamento delle utility di trasferimento dei dati, guardi questo video:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4262>

9.1 EXPORT utility / Utility EXPORT

The EXPORT utility is used to extract data from a table into a file as discussed earlier. Behind the scenes, an SQL SELECT operation is what is really being performed. The following example exports to the file *employee.ixf* of IXF format 10 rows from the table *employee*.

```
EXPORT TO employee.ixf OF IXF
  SELECT * FROM employee
  FETCH FIRST 10 ROWS ONLY
```

We encourage you to try the above example. The employee table is part of the SAMPLE database, so you first need to connect to this database created in a previous chapter.

If you prefer to work with GUI tools, the EXPORT utility can also be invoked from the Control Center as shown in Figure 9.2.

L'utility EXPORT è usata per estrarre i dati da una tabella in un file come discusso precedentemente. Dietro le quinte, una SQL SELECT è ciò che realmente sta eseguendo. Il seguente esempio esporta al file *employee.ixf* di formato IXF, 10 righe dalla tabella *employee*.

Ti incoraggiamo a provare l'esempio precedente. La tabella employee fa parte del database SAMPLE, dunque devi prima collegarti a questo database creato nel capitolo precedente.

Se preferisci lavorare con i tool GUI, l'utility EXPORT può anche essere invocata dal Control Center come indicato in figura 9.2.

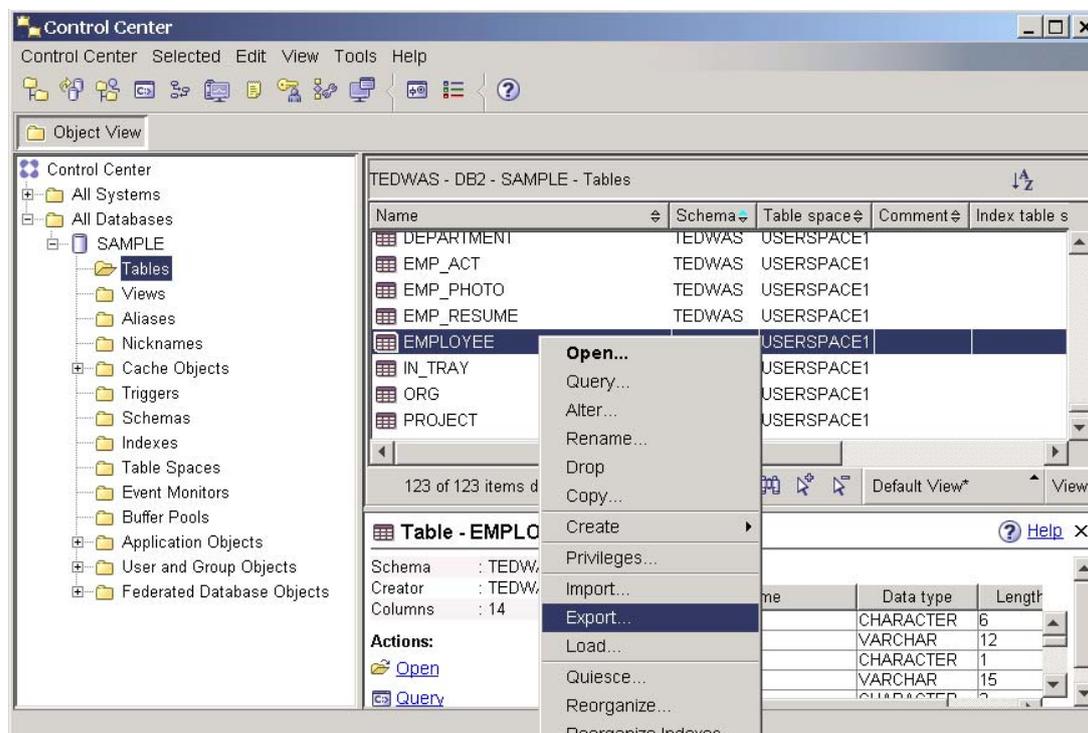


Figure 9.2 – Launching the EXPORT table dialog / Figura 9.2 - Lanciare la finestra di dialogo EXPORT

As shown in the figure, you first select the employee table by clicking it once, and then right click on the table to obtain a pop-up menu from where you can choose the Export option. After choosing this option, a wizard will come up. Simply follow the steps the wizard provides to complete the operation.

Come mostrato in figura, prima selezioni la tabella employee cliccando una volta e poi fai clic destro sulla tabella per ottenere un menù pop-up da dove puoi scegliere l'opzione Export. Dopo questa scelta, si aprirà un wizard. Segui semplicemente i punti che il wizard fornisce per completare l'operazione.

9.2 IMPORT utility / Utility IMPORT

The IMPORT utility is used to load data from a file into a table as discussed earlier. Behind the scenes, an SQL INSERT operation is really being executed. As an INSERT operation is being executed, any triggers are activated, all constraints are enforced immediately, and the database buffer pool is used. The following example loads all the data from the IXF formatted file *employee.ixf* into the table *employee_copy*.

L'utility IMPORT è usata per caricare i dati da un file in una tabella come discusso precedentemente. Dietro le quinte, in realtà, si sta eseguendo una SQL INSERT. Mentre si esegue la INSERT, tutti i trigger sono attivati, tutte le restrizioni sono rispettate immediatamente e il buffer pool del database è usato. Il seguente esempio carica tutti i dati dal file *employee.ixf* con il formato IXF nella tabella *employee_copy*.

The REPLACE_CREATE option is one of many options available with the IMPORT utility. This option will replace the contents of the *employee_copy* table if it previously existed before the IMPORT utility was executed, or it will create the table and load the data if the table didn't already exist. We encourage you to try the example below, but you need to have run the EXPORT utility in the previous section.

L'opzione di REPLACE_CREATE è una delle molte opzioni disponibili con l'utility IMPORT. Quest'opzione sostituirà il contenuto della tabella *employee_copy* se esisteva prima dell'IMPORT, o creerà la tabella e caricherà i dati se la tabella non esisteva già. Ti incoraggiamo a provare l'esempio seguente, ma prima devi avere eseguito l'utility EXPORT nella sezione precedente.

```
IMPORT FROM employee.ixf OF IXF
REPLACE_CREATE
INTO employee_copy
```

If you prefer to work from the Control Center, you can launch the IMPORT utility by selecting any table, right-clicking on it, and choosing the Import option as shown in Figure 9.3.

Se preferisci lavorare dal Control Center, puoi lanciare l'utility IMPORT selezionando qualunque tabella, fai clic destro e scegli l'opzione IMPORT come indicato in figura 9.3.

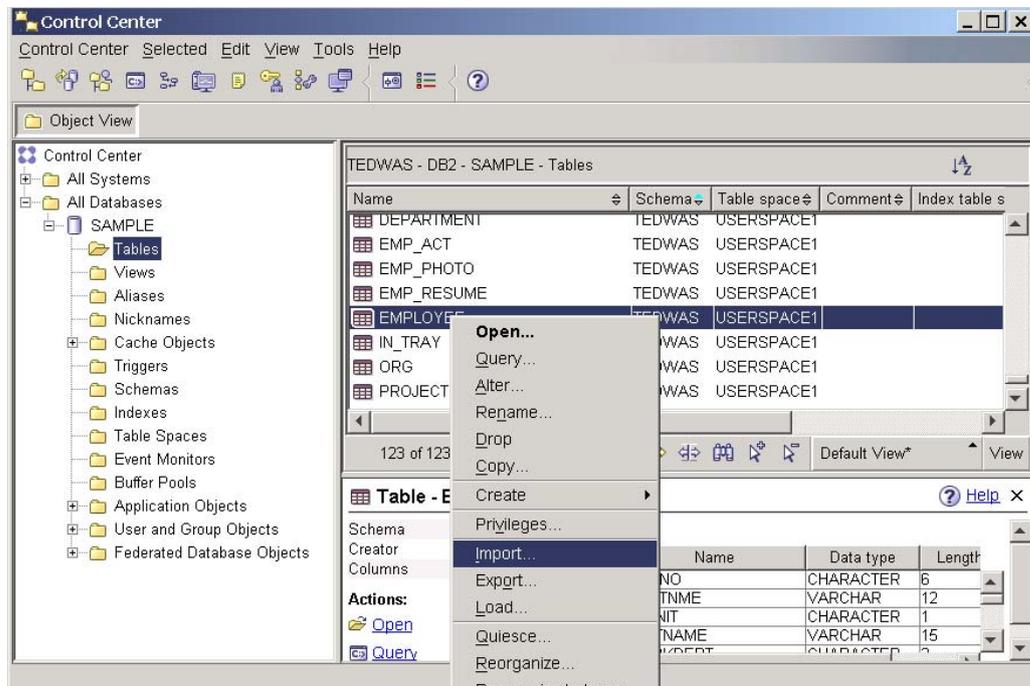


Figure 9.3 – Launching the IMPORT dialog / Figura 9.3 - Lanciare la finestra di dialogo IMPORT

9.3 LOAD

The LOAD utility is a faster way to load data from a file into a table. As discussed before, the LOAD utility does not go through the DB2 engine, so therefore triggers are not activated, the buffer pool is not used and constraints can be enforced but only as a separate step. On the other hand, a LOAD operation is faster than IMPORT as it is a low level data loader directly accessing the data pages on disk. It works in three phases: LOAD, BUILD, and DELETE.

The following example loads all the data from the IXF formatted file *employee.ixf* into the table *employee_copy*. The REPLACE option is one of the many options available with LOAD. In this

L'utility LOAD è un modo più veloce per caricare i dati da un file in una tabella. Come discusso prima, l'utility LOAD non passa attraverso il motore di DB2, quindi i trigger non sono attivati, il buffer pool non è usato, e le restrizioni possono essere rispettate soltanto come un passo separato. Dall'altro lato, un'operazione di LOAD è più veloce dell'IMPORT poiché è un caricamento dei dati a basso livello che accede alle pagine di dati direttamente sul disco. Funziona in tre fasi: LOAD, BUILD e DELETE.

Il seguente esempio caricherà tutti i dati dal file *employee.ixf* in formato IXF nella tabella *employee_copy*. L'opzione REPLACE è uno delle molte opzioni disponibili con LOAD. In questo caso è

case it is used to REPLACE all the contents of the *employee_copy* table.

usato per fare il REPLACE (sostituzione) di tutto il contenuto della tabella *employee_copy*.

```
LOAD FROM employee.ixf OF IXF
REPLACE INTO employee_copy
```

After executing the above command (which you can try out), the table space where your table resides may have been placed in CHECK PENDING state. This means you need to run the SET INTEGRITY command to check the consistency of your data. The following example shows you how:

Dopo l'esecuzione del comando precedente (che puoi provare), lo spazio di tabella in cui la vostra tabella risiede può essere posto nello stato CHECK PENDING. Ciò significa la necessità di eseguire il comando SET INTEGRITY per controllare la consistenza dei vostri dati. Il seguente esempio ti mostra come si fa:

```
SET INTEGRITY FOR employee_copy
ALL IMMEDIATE UNCHECKED
```

If you prefer to work from the Control Center, you can launch the LOAD and the SET INTEGRITY utilities, as shown in Figure 9.4 and 9.5 respectively.

Se preferisci lavorare dal Control Center, puoi lanciare LOAD e l'utility SET INTEGRITY, come indicato nelle figure 9.4 e 9.5, rispettivamente.

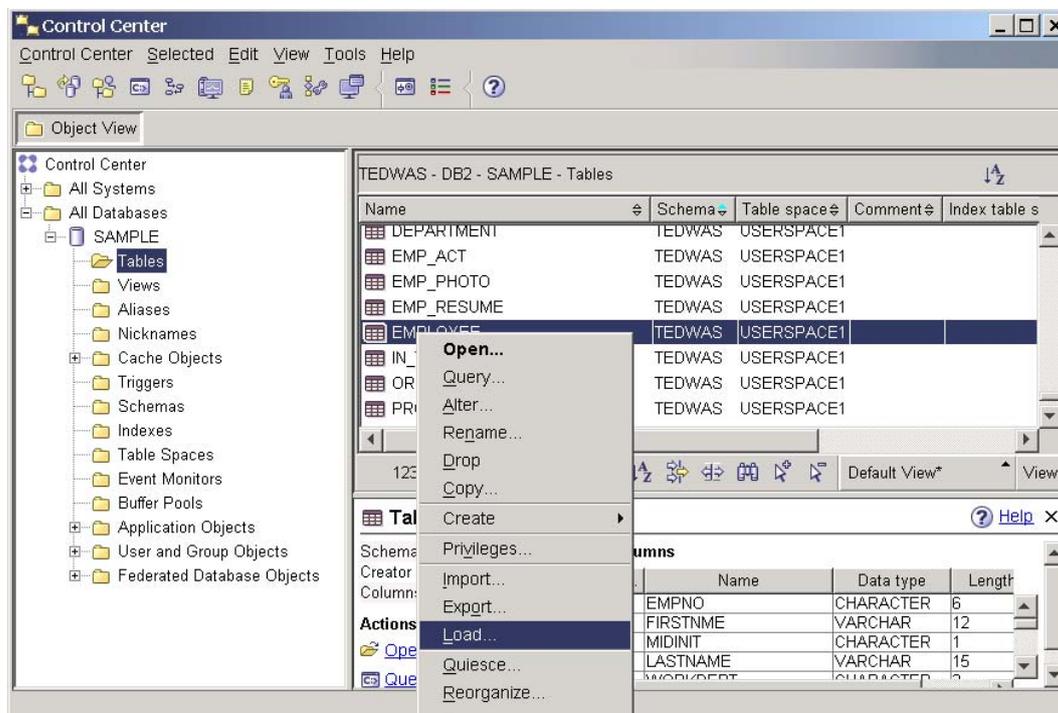


Figure 9.4 – Launching the LOAD wizard / Figura 9.4 - Lanciare il wizard LOAD

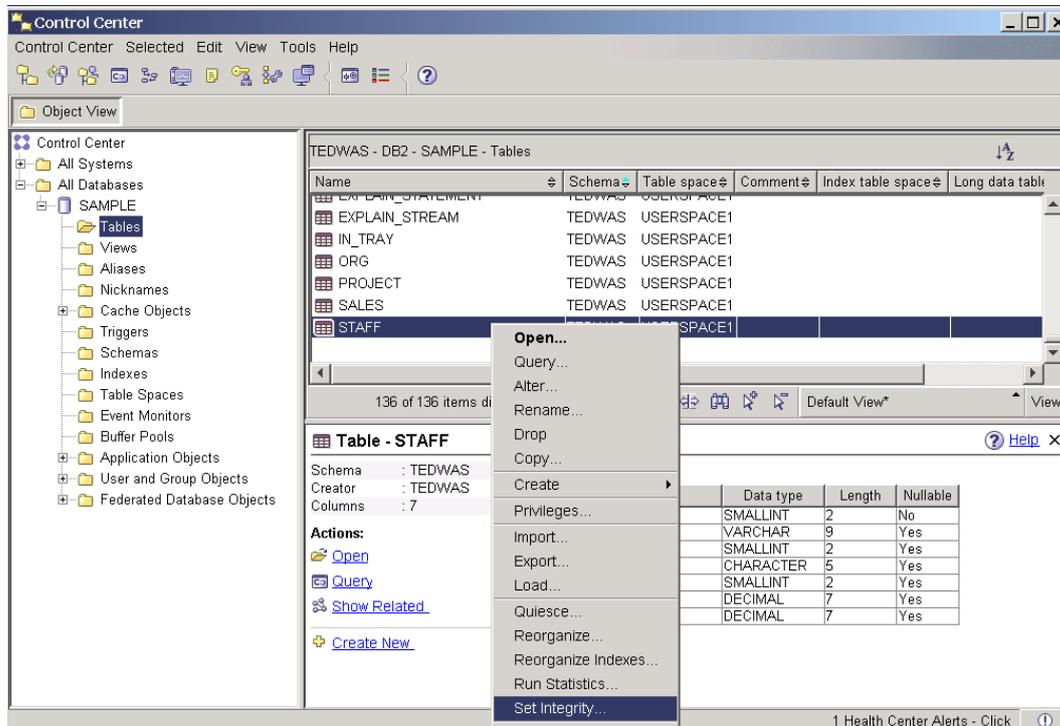


Figure 9.5 – Launching the SET INTEGRITY wizard / Figura 9.5 - Lanciare il wizard SET INTEGRITY

9.4 The db2move utility / Utility db2move

The EXPORT, IMPORT, and LOAD utilities work on one table at a time. Though you could write a script to generate the above commands for each table in a database, another utility called *db2move* can do this for you. The *db2move* utility can only work with IXF files, and the file names will automatically be generated by *db2move*. The examples below show how to run *db2move* with the export, and import options respectively using the SAMPLE database.

```
db2move sample export
db2move sample import
```

The Control Center does not have an option for *db2move*.

Le utility EXPORT, IMPORT e LOAD lavorano su una tabella alla volta. Nonostante potresti scrivere uno script per generare i comandi precedenti per ogni tabella in un database, esiste un'altra utility chiamata *db2move* che può fare questo per te. L'utility *db2move* può funzionare soltanto con i file IXF ed i nomi del file saranno generati automaticamente da *db2move*. I seguenti esempi mostrano come eseguire *db2move* con le opzioni di export e import, rispettivamente usando il database SAMPLE.

Il Control Center non ha un'opzione per *db2move*.

9.5 The db2look utility / Utility db2look

While EXPORT, IMPORT, LOAD and db2move utilities allow you to move data from one table to another, either within one database or across several databases, the db2look utility can be used to extract the DDL statements, database statistics and table space characteristics for a database and store them in a script file that can later be run on another system.

For example, if you would like to clone a database from a DB2 server running on Linux to a DB2 server running on Windows; you could first run the db2look utility on the DB2 Linux server to obtain the structure of the database and store this structure on a script file. You would then copy this script file to the DB2 Windows server, and execute the script to start building the cloned database. At this point, the structure of the database has been cloned.

The next step would be to run the db2move utility with the export option in the DB2 Linux server, and then copy all the generated files to the DB2 Windows server, then execute the db2move with the import or load options. Once this is done, your database would be fully cloned from one server to another on different platforms.

The above scenario may be needed when working with databases on different platforms such as Linux and Windows. If both servers are running on the same platform, you would likely use the `backup` and `restore` commands, which make this process easier and more straight-forward. The backup and restore commands are discussed in more detail in a later chapter of this book.

Mentre EXPORT, IMPORT, LOAD e db2move ti permettono di spostare i dati da una tabella ad un'altra, sia all'interno di un database o attraverso parecchi database, l'utility db2look può essere usata per estrarre gli statement DDL, le statistiche del database e le caratteristiche dello spazio di tabella per un database e per memorizzarle in uno script file che può successivamente essere eseguito su un altro sistema.

Per esempio, se volessi clonare un database da un server DB2 che funziona su Linux ad un server DB2 che funziona su Windows; potresti prima eseguire l'utility db2look sul server DB2 su Linux per ottenere la struttura del database e per memorizzare questa struttura su uno script file. Poi copierai questo file di script sul server DB2 su Windows ed eseguirai lo script per cominciare a costruire il database clonato. A questo punto, la struttura del database sarà stata clonata.

Il punto seguente consiste nell'eseguire l'utility db2move con l'opzione EXPORT nel server DB2 su Linux e poi copiare tutte i file generati al server DB2 su Windows, e poi eseguire db2move con le opzioni di import o load. Una volta che questo sarà stato fatto, il tuo database sarà completamente clonato da un server ad un altro su piattaforme diverse.

Lo scenario precedente può essere necessario quando si lavora con database su piattaforme diverse come Linux e Windows. Se entrambi i server stanno girando sulla stessa piattaforma, probabilmente userai i comandi di `backup` e di `restore`, che rendono questo processo più facile. I comandi di backup e restore sono discussi più in dettaglio in un capitolo successivo di questo libro.

The following example extracts the table space and buffer pool layouts, along with the DDL statements from the SAMPLE database, and stores them into the file `sample.ddl`. We encourage you to run the command below and review the output text file `sample.ddl`.

Il seguente esempio estrae il layout dello spazio della tabella e del buffer pool, con gli statement DDL dal database SAMPLE e le memorizza nel file `sample.ddl`. Ti incoraggiamo ad eseguire il comando seguente ed esaminare il file di testo in output `sample.ddl`.

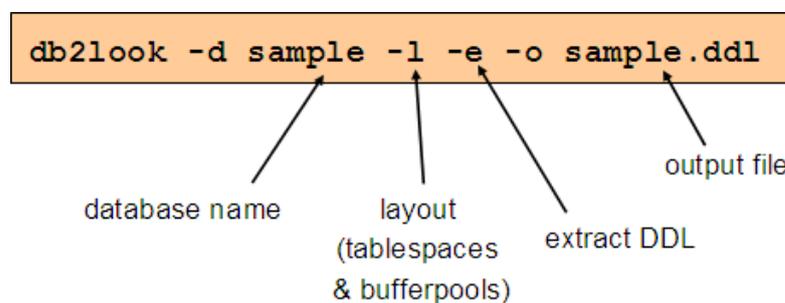


Figure 9.6 – db2look syntax / Figura 9.6 – Il sintassi di db2look

The `db2look` command has too many options to describe in this book; however you can use the `-h` flag to obtain a brief description of the available options:

Il comando `db2look` ha troppe opzioni per descriverle tutte in questo libro; tuttavia puoi usare il flag `-h` per ottenere una breve descrizione delle opzioni disponibili:

```
db2look -h
```

The `db2look` utility can also be invoked from the Control Center as shown in Figure 9.6.

L'utility `db2look` può anche essere invocata dal Control Center come indicato in figura 9.6.

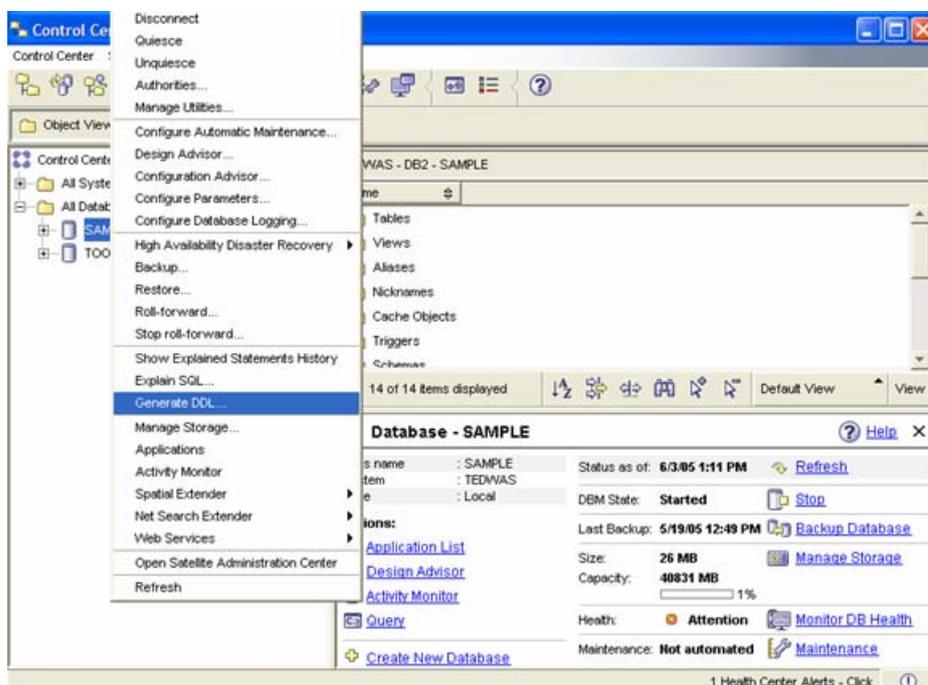


Figure 9.6 – Extracting DDL from the Control Center / Figura 9.6 – Estrarre DDL dal Control Center

In Figure 9.6, select the database from which you want to obtain the DDL, right click on it, and choose *Generate DDL*. The Generate DDL window appears, showing several extraction options, as shown in Figure 9.7.

Nella figura 9.6, selezioni il database da cui vuoi ottenere il DDL, fai clic destro e scegli *Generate DDL*. La finestra Generate DDL compare, mostrando parecchie opzioni di estrazione, come indicato in figura 9.7.

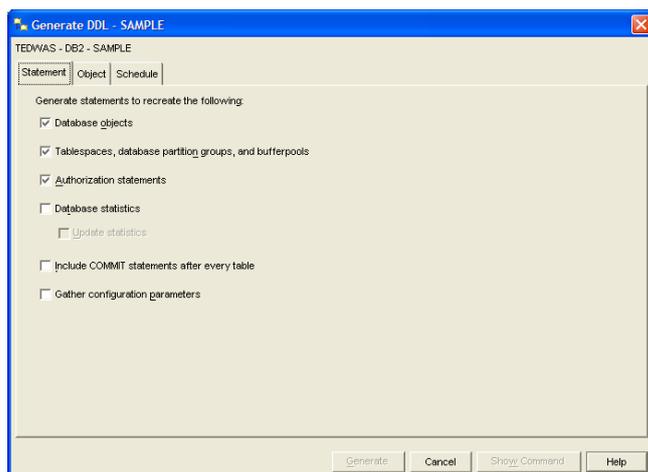


Figure 9.7 - Extracting DDL from the Control Center / Figura 9.7 - Estrarre DDL dal Control Center

Quicklab #8 – Extracting DDL for the EXPRESS database / Estrazione del DDL per il database EXPRESS

Objective

When you clone a database, your goal should be to make the re-creation of the database as straightforward and repeatable as possible. This is usually done using SQL scripts, which can be immediately executed after DB2 has been installed. In this Quicklab, you will extract the object definitions from the *EXPRESS* database (created in Quicklab #2) using the Control Center.

Procedure

1. Open the Control Center.
2. Right-click on the *EXPRESS* database in the object tree and select the *Generate DDL* menu item. This launches the *Generate DDL* window.
3. In the *Generate DDL* window, specify options for the generated DDL, as shown below. If you created additional objects in your environment, such as table spaces, buffer pools, etc., you would select them here. Since you have not created these types of objects, uncheck the box.

Database statistics have not been included because the production environment will likely contain a different set of statistics than the development

Obiettivo

Quando cloni un database, il tuo obiettivo dovrebbe essere di rendere la ricreazione del database il più possibile diretta e ripetibile. Ciò è fatto solitamente usando gli script di SQL, che possono essere eseguiti immediatamente dopo che DB2 è stato installato. In questo Quicklab, estrarrai le definizioni dell'oggetto dal database EXPRESS (generato nel Quicklab #2) usando il Control Center.

Procedura

1. Apri il Control Center.
2. Fai clic destro sul database *EXPRESS* nell'albero degli oggetti e seleziona la voce *Generate DDL* del menù. Ciò lancia la finestra di dialogo di *Generate DDL*.
3. Nella finestra *Generate DDL*, specifica le opzioni per il DDL generato, come indicato di seguito. Se crei oggetti supplementari nel tuo ambiente, come spazi di tabella, buffer pool, ecc., li selezionerai qui. Poiché non hai generato questi tipi di oggetti, deseleziona il box.

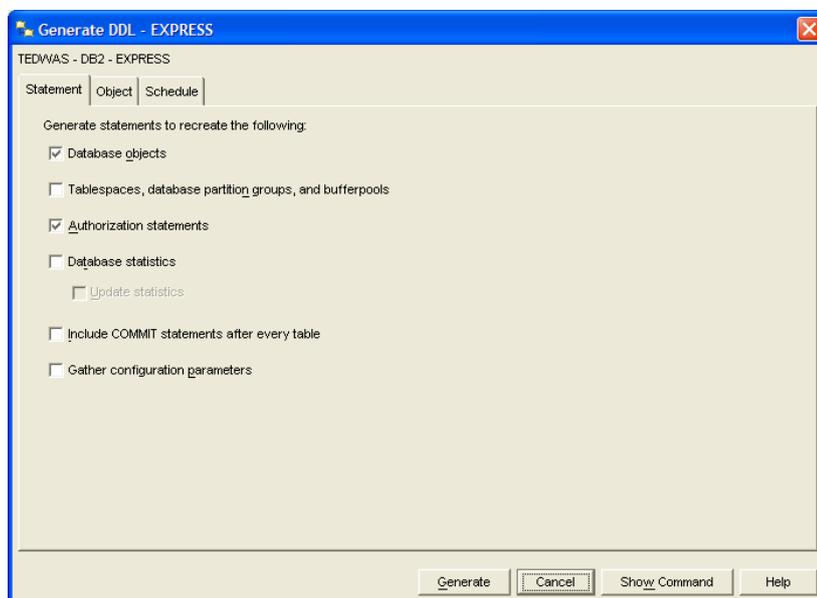
Le statistiche del database non sono state incluse perché l'ambiente di produzione probabilmente contiene un insieme differente delle

environment.

Similarly, configuration parameters will likely be different as well. In your own environment, if everything is configured exactly the way it will be deployed, you may choose to include those additional options.

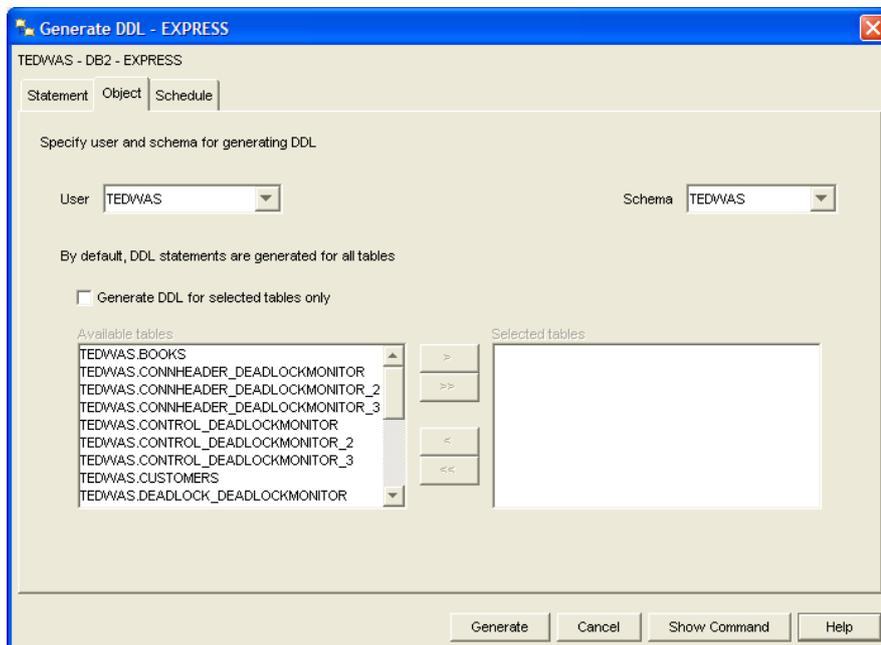
statistiche dall'ambiente di sviluppo.

In modo analogo anche i parametri di configurazione probabilmente saranno differenti. Nel tuo ambiente, se tutto è configurato esattamente nel modo in cui sarà fatto il deploy, puoi scegliere di includere le opzioni supplementari.

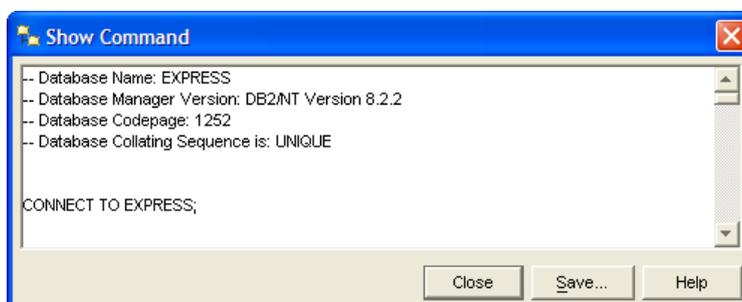


4. Move to the *Object* tab. You are able to specifically choose which objects you want to generate DDL. In this case, select the user and schema you have been using to create all your objects in and generate the DDL for all objects in that schema. Click the *Generate* button to start DDL generation.

4. Spostati al tab *Object*. Puoi specificamente scegliere gli oggetti su cui vuoi generare il DDL. In questo caso, seleziona l'user e lo schema che stavi usando per generare tutti i tuoi oggetti all'interno e genera il DDL per tutti gli oggetti in quello schema. Clicca il tasto *Generate* per iniziare la generazione del DDL.



5. Review the resulting DDL. The result of the previous step is a single script with all the SQL statements for the chosen objects. You will now organize this script into logical groupings.
5. Esamina il DDL ottenuto. Il risultato del punto precedente è un singolo script con tutti gli statement SQL per gli oggetti scelti. Ora organizza questo script in raggruppamenti logici.
6. Create a directory called C:\express in the file system and save the generated DDL file in this new directory to a file called schema.ddl. (Click the Save button)
6. Crea una directory chiamata C:\express nel file system e salva il file DDL generato in questa nuova directory in un file chiamato schema.ddl. (Clicca il tasto Save)



- | | |
|---|---|
| <p>7. Open the newly saved file in Command Editor. (Hint: From Command Editor, choose <i>File => Open</i>)</p> | <p>7. Nel Command Editor, apri il nuovo file salvato. (Suggerimento: Dal Command Editor, scegli <i>File => Open</i>)</p> |
| <p>8. Although we only really wanted the DDL for tables, you will notice DDL for other database objects is included as well. Move all the CREATE TRIGGER statements into a separate new file called <code>triggers.ddl</code>. Even though we only created one trigger, it is generally a best practice to separate objects by types.</p> | <p>8. Anche se realmente abbiamo voluto soltanto il DDL per le tabelle, nota che è incluso anche il DDL per altri oggetti del database. Muovi tutti gli statement CREATE TRIGGER in un nuovo file (separato) chiamato <code>triggers.ddl</code>. Anche se abbiamo creato soltanto un trigger, è generalmente un'ottima pratica separare gli oggetti dai tipi.</p> |
| <p>9. For now, we also recommend removing all the statements:</p> <ul style="list-style-type: none"> • CONNECT TO database • DISCONNECT statements <p>You should have two scripts at this point: DDL for tables, views, indexes, and constraints:</p> <pre style="margin-left: 40px;">C:\express\schema.ddl</pre> <p>DDL for triggers:</p> <pre style="margin-left: 40px;">C:\express\triggers.ddl</pre> | <p>9. Per ora, inoltre consigliamo di rimuovere tutti gli statement:</p> <ul style="list-style-type: none"> • CONNECT TO database • DISCONNECT <p>Dovreste avere due script a questo punto: DDL per tabelle, viste, indici e constraint:</p> <p>DDL per trigger:</p> |
| <p>10. Cleanse the script for deployment:</p> <ul style="list-style-type: none"> • Remove unnecessary comments (e.g. <code>-- CONNECT TO...</code>) • Separate the functions and procedures into their own files (useful when there are a lot of functions and procedures). You might also want to group them by function or application (e.g. | <p>10. Pulire lo script per il deploy:</p> <ul style="list-style-type: none"> • Rimuovere le osservazioni inutili (per esempio <code>-- CONNECT TO...</code>) • Separare le funzioni e le procedure nei loro file (utili quando ci sono molte funzioni e procedure). Potresti anche volerle raggruppare per funzione o per applicazione (per |

billing.ddl,
math.ddl,
tringfunc.ddl, etc.)

esempio, billing.ddl,
math.ddl,
stringfunc.ddl, ecc.)

11. You may have noticed that a special character is being used to delimit the end of the triggers, functions and procedures (@). This is necessary in order to delimit the end of the `CREATE <object>` statement as opposed to the end of a procedural statement within the object.

11. Puoi notare che un carattere speciale (@) è usato per delimitare la fine dei trigger, delle funzioni e delle procedure. Ciò è necessario per delimitare la fine dello statement `CREATE <object>` in contrasto con la fine di uno statement procedurale all'interno dell'oggetto.

10

Chapter 10 – Database Security / Capitolo 10 – Sicurezza del Database

This chapter discusses how security is handled in DB2. Figure 10.1 provides a basic overview.

Questo capitolo discute come viene trattata la sicurezza in DB2. La figura 10.1 ne fornisce una visione di base.

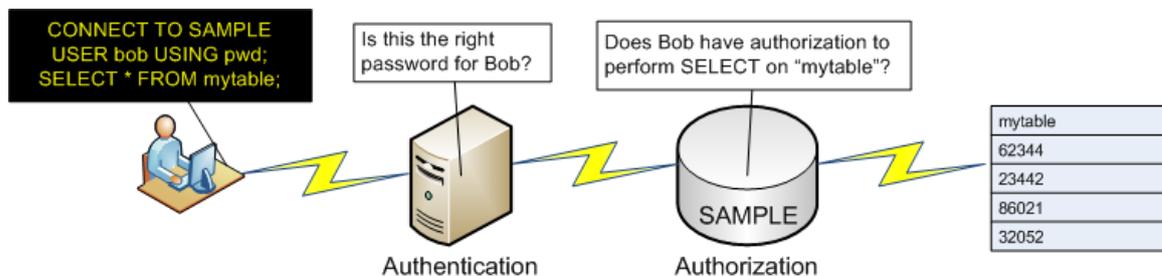


Figure 10.1 – DB2 security overview / Figura 10.1 - Overview della sicurezza in DB2

As shown in Figure 10.1, DB2 security consists of two parts:

Come mostrato in figura 10.1, la sicurezza in DB2 consiste di due parti:

Authentication

It is the process by which the user identity is validated. Authentication is performed by a security facility outside of DB2 (typically by an operating system, some network authentication method, or a custom-built authentication plug-in). OS-based authentication is the default. When using OS-based authentication, the user ID and password are flown to the database server (e.g. as part of a connect statement). The database server then invokes the OS authentication to validate the user ID and password.

Autenticazione

È il processo tramite cui l'identità dell'utente è validata. L'autenticazione è effettuata da una funzione di sicurezza fuori da DB2 (tipicamente da un sistema operativo, alcuni metodi di autenticazione della rete, o da un plug-in custom-built di autenticazione). L'autenticazione OS-based è quella di default. Quando si usa l'autenticazione OS-based, lo user ID e la password sono mandate al database server (per esempio come componente dello statement connect). Il database server poi invoca l'autenticazione dell'OS per validare lo user ID e la password.

Authorization

At this stage, DB2 checks if the authenticated user may perform the requested operation. The authorization information is stored in a DB2 catalog and a DBM configuration file. For example, in Figure 10.1, user "bob" connects to the SAMPLE database with this statement:

```
CONNECT TO sample USER bob USING pwd
```

Both "bob" and "psw" are passed to the operating system or external authentication facility to perform the authentication approval, verifying that a user named "bob" is already defined, and that the password provided matches that user. If this part is successful, the operating system will return security control to DB2. Next, when user "bob" executes a statement such as:

```
SELECT * FROM mytable
```

Now DB2 takes over security control to perform the authorization check and confirm that user "bob" has SELECT privilege on table "mytable". If the authorization check fails, DB2 will return an error message, otherwise the statement will be executed against "mytable".

Autorizzazione

In questa fase, DB2 controlla se lo user autenticato può eseguire l'operazione richiesta. Le informazioni di autorizzazione sono memorizzate in un catalogo DB2 e in un file di configurazione DBM. Per esempio, nella figura 10.1, lo user "bob" si collega al database SAMPLE con questo statement:

Sia "bob" che "psw" sono passati al sistema operativo o alle funzioni esterne di autenticazione per effettuare l'approvazione dell'autenticazione, verificando che un user chiamato "bob" sia già definito e che la password fornita sia uguale per quell'user. Se questa parte riesce, il sistema operativo restituirà il controllo di sicurezza a DB2. Successivamente, quando lo user "bob" esegue uno statement come:

DB2 assume il controllo di sicurezza per effettuare il controllo di autorizzazione e per confermare che l'user "bob" ha il privilegio di SELECT sulla tabella "mytable". Se il controllo di autorizzazione fallisce, DB2 ritorna un messaggio di errore, altrimenti lo statement sarà eseguito su "mytable".

Note:

For more information about working with DB2 security, watch this video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4267>

Nota:

Per maggiori informazioni sul funzionamento con la sicurezza DB2, guardi questo video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4267>

10.1 Authentication / Autenticazione

Although the actual authentication is performed by the operating system (or another external security facility), DB2 does decide at which level this authentication occurs.

The DBM CFG parameter AUTHENTICATION, set at the DB2 server, has a range of possible values. For example, when the parameter is set to SERVER (the default), the authentication is performed by the operating system/external security facility on the server. However, if AUTHENTICATION is set to CLIENT, the authentication is performed by the operating system/external security facility at the client. This is shown in Figure 10.2.

Anche se l'autenticazione attuale è realizzata dal sistema operativo (o da un'altra funzione di sicurezza esterna), DB2 decide a quale livello questa autenticazione deve avvenire.

Il parametro AUTHENTICATION di DBM CFG, fissato sul server DB2, ha una gamma di valori possibili. Per esempio, quando il parametro è fissato su SERVER (default), l'autenticazione è realizzata dal sistema operativo (o funzione di sicurezza esterna) sul server. Tuttavia, se il parametro AUTHENTICATION è fissato al valore CLIENT, l'autenticazione è realizzata dal sistema operativo (o funzione di sicurezza esterna) sul client. Ciò è mostrato nella figura 10.2.

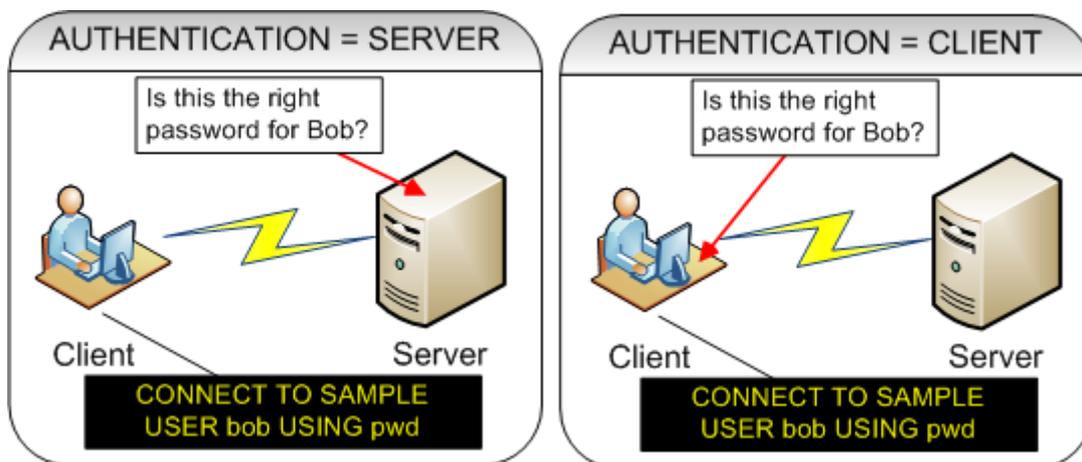


Figure 10.2 – Where authentication takes place / Figura 10.2 - Dove avviene l'autenticazione

The AUTHENTICATION parameter can be set to any of the values listed in Table 10.1

Il parametro AUTHENTICATION può essere fissato ad alcuni valori elencati in tabella 10.1

| Command / Comando | Description | Descrizione |
|--------------------------------|--|--|
| SERVER (default) | Authentication takes place at the server | L'autenticazione avviene sul server |
| CLIENT | Authentication takes place on the client | L'autenticazione avviene sul client |
| SERVER_ENCRYPT | Like SERVER except user IDs and passwords are encrypted | Come SERVER ma gli user ID e le password sono cifrati |
| KERBEROS | Authentication takes place using a Kerberos security mechanism | L'autenticazione avviene usando un meccanismo di sicurezza Kerberos |
| SQL_AUTHENTICATION_DATAENC | Server authentication plus connections must use data encryption | I server di autenticazione e i collegamenti devono usare la crittografia di dati |
| SQL_AUTHENTICATION_DATAENC_CMP | Like above, except data encryption only used when available | Come sopra, ma la crittografia di dati si usa soltanto quando disponibile |
| GSSPLUGIN | Authentication uses an external GSS API-based plug-in security mechanism | L'autenticazione usa un meccanismo di sicurezza esterna con un plug-in GSS API-based |

Table 10.1 – Valid AUTHENTICATION parameter values / Tabella 10.1 - Valori di parametro validi per AUTENTICAZIONE

10.2 Authorization / Autorizzazione

Authorization consists of the privileges and authorities that are stored in DB2 system tables and are managed by DB2.

A privilege allows a user to execute a single type of operation against the database, such as CREATE, UPDATE, DELETE, INSERT, etc.

L'autorizzazione consiste in privilegi e autorità che sono memorizzati in tabelle di sistema DB2 e sono gestiti da DB2.

Un privilegio permette a un user di eseguire un singolo tipo di operazione sul database, come CREATE, UPDATE, DELETE, INSERT, ecc.

An authority is a predefined role consisting of several privileges.

Un'autorità è un ruolo predefinito che consiste di parecchi privilegi.

Figure 10.3 shows the different authorities and privileges in DB2.

La figura 10.3 mostra le diverse autorità ed privilegi in DB2.

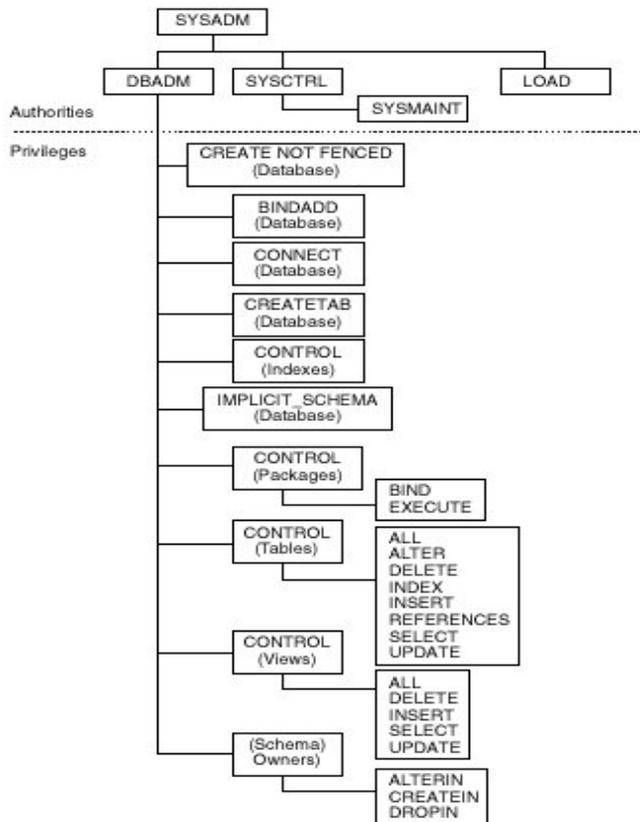


Figure 10.3 – Authorities and privileges / Figura 10.3 - Autorità e privilegi

Table 10.2 shows the different functions that each authority can perform. As you can see, SYSADM has the most authority while SYSMON has the least.

La tabella 10.2 indica le diverse funzioni che ogni autorità può effettuare. Come puoi vedere, SYSADM ha la maggior parte della autorità mentre SYSMON ha il minimo.

| <i>Function / Funzione</i> | <i>SYSADM</i> | <i>SYSCTRL</i> | <i>SYSMAINT</i> | <i>SYSMON</i> | <i>DBADM</i> | <i>LOAD</i> |
|--------------------------------------|---------------|----------------|-----------------|---------------|--------------|-------------|
| Update DBM CFG | Y | | | | | |
| Grant/revoke DBADM | Y | | | | | |
| Establish/change SYSCTRL | Y | | | | | |
| Establish/change SYSMAINT | Y | | | | | |
| Establish/change SYSMON | Y | | | | | |
| Force users off database | Y | Y | | | | |
| Create/drop database | Y | Y | | | | |
| Restore to new database | Y | Y | | | | |
| Update DB CFG | Y | Y | Y | | | |
| Backup database/table space | Y | Y | Y | | | |
| Restore to existing database | Y | Y | Y | | | |
| Perform roll-forward recovery | Y | Y | Y | | | |
| Start/stop instance | Y | Y | Y | | | |
| Restore table space | Y | Y | Y | | | |
| Run trace | Y | Y | Y | Y | | |
| Obtain monitor snapshots | Y | Y | Y | | | |
| Query table space state | Y | Y | Y | | | |
| Prune log history files | Y | Y | Y | | | |
| Quiesce table space | Y | Y | Y | | Y | Y |
| LOAD tables | Y | | | | Y | Y |
| Set/unset check pending state | Y | | | | Y | |
| Create/drop event monitors | Y | | | | Y | |

Table 10.2 - DB2 authorities and privileges / Tabella 10.2 - Autorità e privilegi DB2

In order to grant SYSADM, SYSCTRL or SYSMAINT authority to a group, the DBM CFG parameters SYSADM_GROUP, SYSCTRL_GROUP, and SYSMAINT_GROUP can be assigned to an operating system group.

Per assegnare l'autorità di SYSADM, SYSCTRL o SYSMAINT ad un gruppo, i parametri DBM CFG SYSADM_GROUP, SYSCTRL_GROUP, e SYSMAINT_GROUP possono essere assegnati ad un gruppo del sistema operativo.

For example, to give SYSADM authority to the operating system group "db2admns", you can issue this command:

Per esempio, per dare l'autorità di SYSADM al gruppo "db2admns" del sistema operativo, puoi inserire questo comando:

```
update dbm cfg using SYSADM_GROUP db2admns
```

Each DB2 instance has its own authority group definitions.

Ogni instance DB2 ha le proprie definizioni del gruppo di autorità.

On Windows, these parameters are empty by default, which means the local Windows Administrators group will be SYSADM. On Linux the instance owner group is the default SYSADM group.

Su Windows, questi parametri sono vuoti di default, il che significa che il gruppo Windows Administrators sarà SYSADM. Su Linux il gruppo proprietario del instance è il gruppo SYSADM di default.

10.3 DBADM authority / Autorità di DBADM

The DBADM (DataBase ADMinistrator) authority is a super user for the database. It is not an authority at the instance level; therefore it is not listed in the previous section. In order to grant DBADM authority, use the GRANT statement as shown in the example below.

L'autorità DBADM (DataBase ADMinistrator) è un super user per la base di dati. Non è un'autorità a livello di instance; quindi non è elencato nella sezione precedente. Per assegnare l'autorità di DBADM, usa lo statement GRANT come indicato nell'esempio seguente.

```
connect to sample
grant DBADM on database to user <userid>
```

In the above example, you first need to connect to the database, in this case the "sample" database, and then you can grant DBADM to a user. To grant DBADM authority you need to be SYSADM.

Nell'esempio precedente, devi prima collegarti al database, in questo caso il database "SAMPLE" e poi puoi assegnare DBADM ad un user. Per assegnare autorità di DBADM devi essere SYSADM.

Note that a DBADM cannot create table spaces, even though they are objects inside a database, because a table space deals with containers (disk) and buffer pools (memory) which are physical resources of the system.

Si noti che un DBADM non può creare gli spazi di tabella, anche se sono oggetti all'interno di un database, perché uno spazio di tabella si occupa dei contenitori (disc) e dei buffer pool (memoria) che sono risorse fisiche del sistema.

10.4 The PUBLIC group / Il gruppo PUBLIC

DB2 defines an internal group called PUBLIC. Any user identified by the operating system or network authentication service is implicitly a member of the PUBLIC group. When a database is created, certain privileges are granted to PUBLIC automatically:

- CONNECT,
- CREATETAB,
- IMPLICIT_SCHEMA,
- BINDADD

DB2 definisce un gruppo interno chiamato PUBLIC. Qualsiasi user identificato dal sistema operativo o dal servizio di autenticazione della rete è implicitamente un membro del gruppo PUBLIC. Quando un database è creato, determinati privilegi sono assegnati automaticamente a PUBLIC:

For added security, we recommend revoking all privileges from the PUBLIC group as shown below:

Per una sicurezza aggiuntiva, ti suggeriamo di revocare tutti i privilegi dal gruppo PUBLIC come indicato di seguito:

```
REVOKE CONNECT          ON DATABASE FROM PUBLIC
REVOKE CREATETAB       ON DATABASE FROM PUBLIC
REVOKE IMPLICIT_SCHEMA ON DATABASE FROM PUBLIC
REVOKE BINDADD         ON DATABASE FROM PUBLIC
```

10.5 The GRANT and REVOKE statements / Gli statement GRANT e REVOKE

The GRANT and REVOKE statements are part of the SQL standards, and are used to give or remove privileges to a user or group. Below are some examples of these statements:

Gli statement GRANT e REVOKE fanno parte degli standard SQL e sono usati per dare o rimuovere privilegi ad uno user o ad un gruppo. Qui di seguito alcuni esempi di questi statement:

To grant the SELECT privilege on table T1 to the user USER1:

Per assegnare il privilegio SELECT sulla tabella T1 all'user USER1:

```
GRANT SELECT ON TABLE T1 TO USER user1
```

To grant all privileges on table T1 to the group GROUP1:

Per assegnare tutti i privilegi sulla tabella T1 al gruppo GROUP1:

```
GRANT ALL ON TABLE T1 TO GROUP group1
```

To revoke all privileges on table T1 from group GROUP1:

Per revocare tutti i privilegi sulla tabella T1 dal gruppo GROUP1:

```
REVOKE ALL ON TABLE T1 FROM GROUP group1
```

To grant EXECUTE privilege on procedure p1 to user USER1:

Per assegnare il privilegio GRANT EXECUTE sulla procedura p1 allo user USER1:

```
GRANT EXECUTE ON PROCEDURE p1 TO USER user1
```

To revoke EXECUTE privilege on procedure p1 from user USER1:

Per revocare il privilegio EXECUTE sulla procedura p1 dallo user USER1:

```
REVOKE EXECUTE ON PROCEDURE p1 FROM USER user1
```

10.6 Authorization and privilege checking / Controllo di privilegi e autorizzazione

The easiest way to check for authorization and privileges is through the Control Center. Figure 10.4 shows how to launch the Table Privileges dialog for the EMPLOYEE table from the Control Center.

Il modo più facile per controllare l'autorizzazione e i privilegi è usare il Control Center. La Figura 10.4 mostra come lanciare la finestra di dialogo Table Privileges per la tabella EMPLOYEE dal Control Center.

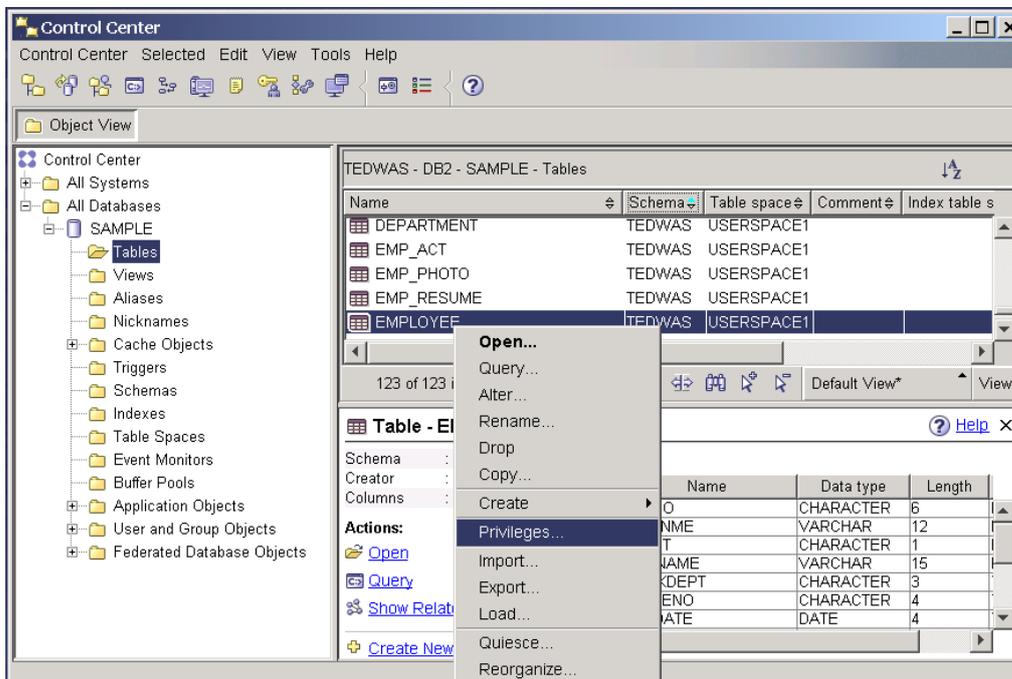


Figure 10.4 - Launching the Table Privileges dialog / Figura 10.4 - Lanciare la finestra di dialogo Table Privileges

As shown by Figure 10.4, you select the desired table, right-click on it, and choose Privileges. Once selected, the Table Privileges dialog box appears as shown in Figure 10.5. This figure also explains the different fields and elements of the dialog box.

Come indicato in figura 10.4, seleziona la tabella che vuoi, fai clic destro e scegli Privileges. Una volta selezionato, la finestra di dialogo Table Privileges compare come indicato in figura 10.5. Questa figura spiega anche i differenti elementi e campi della finestra di dialogo.

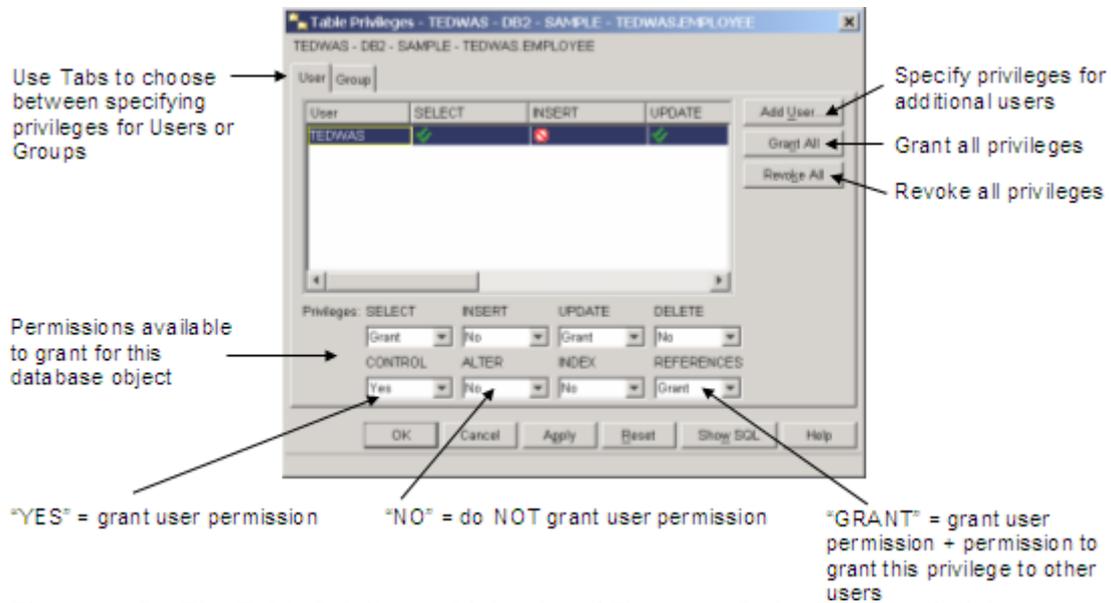


Figure 10.5 – The Table Privileges Dialog box / Figura 10.5 - La finestra di dialogo Table Privileges

Alternatively, you can query the DB2 SYSCAT catalog views which contain the authorization information. For example, if you would like to know if user DB2ADMIN has SELECT privilege on table T2, and would like to know who granted this privilege, you could run a query like this:

```
SELECT grantor, grantee, selectauth
FROM syscat.tabauth
WHERE tabname = 'T2'
```

| GRANTOR | GRANTEE | SELECTAUTH |
|----------|----------|------------|
| ARFCHONG | DB2ADMIN | Y |

In the above example, user ARFCHONG granted SELECT privilege to user DB2ADMIN.

Alternativamente, puoi interrogare la view del catalogo DB2 SYSCAT che contengono le informazioni di autorizzazione. Per esempio, se volessi sapere se l'user DB2ADMIN ha il privilegio SELECT sulla tabella T2 e volessi conoscere chi ha assegnato questo privilegio, puoi eseguire una query come questa:

Nell'esempio precedente, lo user ARFCHONG ha assegnato il privilegio SELECT allo user DB2ADMIN.

10.7 Group privilege considerations / Considerazioni sui privilegio dei gruppi

To make DB2 administration easier, place users into groups, and then grant those groups the required privileges.

Per rendere la gestione di DB2 più facile, disponi gli user nei gruppi e poi assegna a quei gruppi i privilegi richiesti.

When a group is granted privileges, members of the group are granted implicit privileges inherited through group memberships.

Quando a un gruppo vengono assegnati privilegi, ai membri del gruppo sono assegnati implicitamente i privilegi ereditati attraverso l'appartenenza al gruppo.

When a user is removed from a group, they lose the implicit group privileges, but still retain any previous privileges that were explicitly granted: Privileges that were explicitly given to a user must be explicitly revoked from the user.

Quando un user è rimosso da un gruppo, perde i privilegi impliciti del gruppo, ma mantiene ancora tutti i privilegi precedenti che sono stati assegnati esplicitamente: i privilegi che sono stati dati in questa maniera ad uno user devono essere revocati esplicitamente allo user.

Quicklab #9 – Granting and revoking user permissions / Assegnare e revocare i permessi ad un user

Objective

So far, you have been using the instance administrator account (SYSADM) to issue all the database commands. This account has full access to all the utilities, data, and database objects. Therefore, it is very important to safeguard this account in order to avoid accidental or deliberate data loss. In most cases, you will want to create different user accounts or groups with a limited set of permissions. In this lab, you will create a new user account, then assign it specific privileges.

Procedure

1. Open the Windows Computer Management console by right-clicking on the *My Computer* icon on the desktop, and selecting the *Manage* menu item.
2. Expand the *System Tools* selection in the tree on the left pane of the window and then expand the *Local Users and Groups* folder. Right-click on the *User* folder and select the *New User* item.
3. In the *New User* window, enter the following information: in the *User name* field, enter "customer" and in the *Full name* field, enter "Customer1". In the

Obiettivo

Fino a questo momento hai usato un'istanza dell'account amministratore (SYSADM) per eseguire tutti i comandi verso il database. Questo account ha pieno accesso a tutti i programmi di utilità, ai dati ed agli oggetti di database. Di conseguenza, è molto importante salvaguardare questo account per evitare la perdita accidentale o intenzionale dei dati. Nella maggior parte dei casi, vorresti creare diversi account o gruppi di user con permessi limitati. In questo laboratorio, genererai un nuovo user account, e poi gli assegnerai degli specifici privilegi.

Procedura

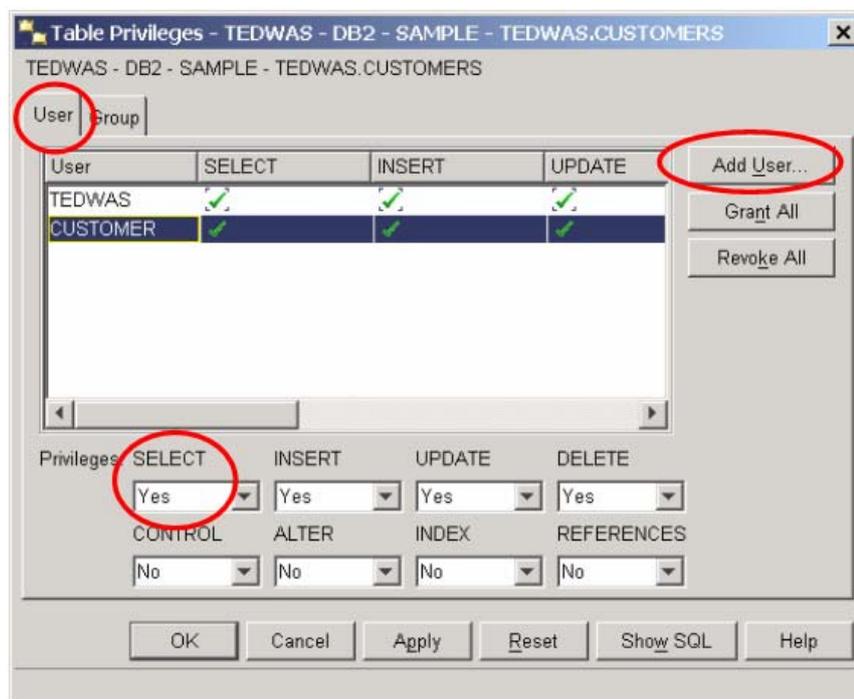
1. Apri la console Windows Computer Management con un clic destro sull'icona *My Computer* sul desktop e seleziona l'opzione del menù *Manage*.
2. Espandi la selezione dei *System Tools* nell'albero sul pannello sinistro della finestra e poi espandi la cartella *Local Users e Groups*. Fai clic destro sulla cartella *User* e seleziona *New User*.
3. Nel finestra *New User*, inserisci le seguenti informazioni: nel campo *User name*, scrivi "customer" e nel campo *Full name*, inserisci "Customer1". Nel campo *Description*, inserisci

Description field, enter "A typical bookstore customer". In the *Password* and *Confirm password* fields, enter "ibmdb2". Remove the checkmark from the *User must change password on next logon* option, and click the *Create* button to create the new user.

"A typical bookstore customer". Nei campi *Password* e *Confirm password*, "ibmdb2". Rimuovi il checkmark dall'opzione "User must change password on next logon", e fai clic sul tasto *Create* per creare il nuovo user.

4. Ensure the advanced view is being used in the DB2 Control Center. To switch to the advanced view, select the *Customize Control Center* menu item from the Control Center *Tools* menu. Select the *Advanced* option and click the *OK* button.
 5. Expand the Control Center object tree in the left object tree pane to *All Databases > EXPRESS > Tables*.
 6. Grant the required privileges to the newly created user. From the list of tables in the
4. Assicurati che la vista avanzata sia utilizzata nel DB2 Control Center. Per passare alla vista avanzata, seleziona il punto del menù *Customize Control Center* dal menù *Tools* nel Control Center. Seleziona l'opzione *Advanced* e clicca il tasto *OK*.
 5. Espandi l'albero Control Center nel pannello sinistro *All Databases > EXPRESS > Tables*.
 6. Assegna i privilegi richiesti allo user appena creato. Dalla lista delle tabelle nel database

- | | |
|--|--|
| <p><i>EXPRESS</i> database, right click the <i>CUSTOMERS</i> table, and select the <i>Privileges</i> item to view the <i>Table Privileges</i> window.</p> <p>7. Click the <i>Add User</i> button and select the <i>customer</i> user just created. Click the <i>OK</i> button to close the <i>Add User</i> dialog box.</p> <p>8. You will notice that the <i>customer</i> user has been added to the user list, but has no privileges assigned. To grant <i>SELECT</i>, <i>INSERT</i>, <i>UPDATE</i>, and <i>DELETE</i> privileges to the user, change each drop down box to <i>Yes</i>. An Internet customer should be able to view/add/update/delete their account data. We do not give the user the other permissions because they do not require them. Click the <i>OK</i> button to close the <i>Table Privileges</i> window and accept the changes you made.</p> | <p><i>EXPRESS</i>, clicca con il tasto destro sulla tabella <i>CUSTOMERS</i> e seleziona l'oggetto <i>Privileges</i> per osservare la finestra <i>Table Privileges</i>.</p> <p>7. Clicca sul tasto <i>Add User</i> e seleziona lo user <i>customer</i> appena creato. Clicca sul tasto <i>OK</i> per chiudere la finestra <i>Add User</i>.</p> <p>8. Nota che l'user <i>customer</i> è stato aggiunto alla lista degli user, ma non ha ancora privilegi assegnati. Per assegnare i privilegi <i>GRANT SELECT</i>, <i>INSERT</i>, <i>UPDATE</i>, e <i>DELETE</i> all'user, cambia ogni casella di scelta a <i>Yes</i>. Un cliente Internet dovrebbe poter osservare/aggiunge/aggiornare/cancellare i propri dati dell'account. Non diamo all'user altri permessi perché non li richiedono. Clicca <i>OK</i> per chiudere la finestra <i>Table Privileges</i> e per accettare i cambiamenti fatti.</p> |
|--|--|



9. Repeat Steps 7-9 for the *BOOKS* and *SALES* tables. For the *BOOKS* table, only grant the SELECT privilege because the customer should not be able to modify any of the store's inventory data. For the *SALES* table, only grant the SELECT and INSERT privileges. The customer should NOT have the DELETE or UPDATE privilege because only store employees should have access to modify sales transactions.
9. Ripeti i punti 7-9 per le tabelle *BOOKS* e *SALES*. Per la tabella *BOOKS*, assegna soltanto il privilegio SELECT perché il cliente non dovrebbe potere modificare alcun dato di inventario del negozio. Per la tabella *SALES*, assegna soltanto i privilegi SELECT ed INSERT. Il cliente NON dovrebbe avere il privilegio DELETE o UPDATE perché soltanto gli impiegati del negozio dovrebbero avere accesso per modificare le transazioni di vendita.
10. Connect to the database using the *customer* user ID created above. Try to SELECT data from the customers table. What happens? Try to DELETE or UPDATE data in the *SALES* table. What happens?
10. Collegati al database usando l'user ID *customer* creato precedentemente. Prova una SELECT dei dati dalla tabella dei clienti. Che cosa accade? Prova una DELETE o un UPDATE dei dati nella tabella *SALES*. Che cosa

In this Quicklab, we only created one user; however, your application may contain many different types of users. Experiment with creating other users and assigning them privileges. You can also create groups of users and assign privileges to those groups, rather than to each individual user specifically.

accade?

In questo Quicklab, abbiamo creato soltanto un user; tuttavia, la tua applicazione può contenere molti tipi diversi di user. Sperimentarlo con la generazione di altri user e assegnando loro i privilegi. Puoi anche creare gruppi di user ed assegnare privilegi a quei gruppi, piuttosto che ad ogni user specificamente.

11

Chapter 11 – Backup and Recovery / Capitolo 11 - Backup e Recovery

In this chapter we discuss DB2 database logging, how to make a full or partial copy of your database using the BACKUP utility, and how to recover your data using the RESTORE utility.

In questo capitolo discutiamo il logging di un database DB2, come fare una copia completa o parziale del tuo database usando l'utility BACKUP e come recuperare i tuoi dati usando l'utility RESTORE.

Note:

For more information about logging, backup, and recovery, watch this video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4282>

Nota:

Per ulteriori informazioni su logging, backup e recovery, guardi questo video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4282>

11.1 Database Logging / Logging del database

If you were working with a text editor, every time you want to ensure your document is saved, you click the “save” button. In the database world, a COMMIT statement does just that. Every time a COMMIT statement is executed, you guarantee that whatever changes were made to the data, they will be saved somewhere.

In a similar way, when you work with a text document, sometimes you will see at the bottom right corner a brief message saying “auto-saving”. In the database world, this happens as well, because any operation you perform against the data, such as an UPDATE,

Se stai lavorando con un text editor, ogni volta che vuoi assicurarti che il tuo documento sia salvato, cliccherai il tasto “save”. Nel mondo dei database, lo statement COMMIT fa giusto quello. Ogni volta che uno statement COMMIT viene eseguito, garantischi che qualsiasi cambiamento che era stato fatto ai dati, verrà conservato in qualche luogo.

In modo simile, quando lavori con un documento di testo, a volte vedrai nell'angolo inferiore a destra un breve messaggio che dice “auto-saving”. Anche nel mondo dei database questo accade, perché qualsiasi operazione sui dati che effettui, come un UPDATE,

INSERT or DELETE, will be saved somewhere as you perform it.

That “somewhere” in the preceding paragraphs refers to the database logs. The database logs are stored on disk and are used to record actions of transactions. If there is a system or database crash, logs are used to playback and redo committed transactions during a recovery.

Figure 11.1 provides a graphical overview of what happens when you are working with a database in terms of logging.

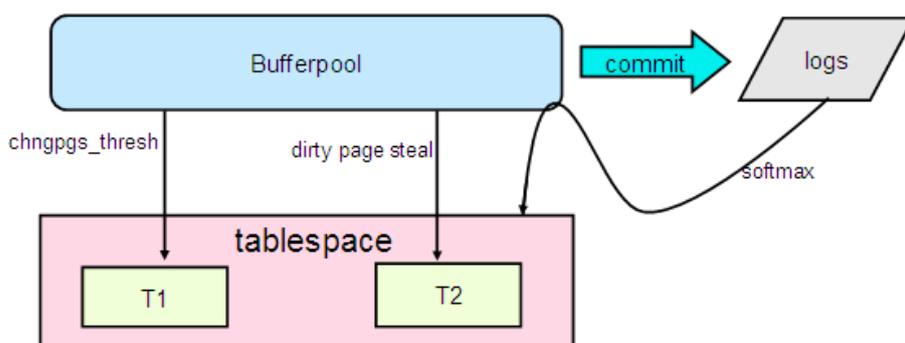


Figure 11.1 – Database logging / Figura 11.1 - Logging del database

In Figure 11.1, we see a table space and logs. Both of them reside on disks, although we recommend that they are not kept on the same disk. When an UPDATE operation takes place for example, the pages for the row(s) in question will be brought to the buffer pool (memory). The update changes will be performed in the buffer pool, and the old and new values will be stored in the log files, sometimes immediately, and sometimes when a log buffer is full. If a COMMIT is issued after the UPDATE, the old and new value will be stored in the log files immediately. This process is repeated for many other SQL operations that are performed on the database. Only when certain conditions are met, such as reaching the change

INSERT o DELETE, verrà conservato in qualche luogo mentre lo esegui.

Quel “luogo” nei paragrafi precedenti si riferisce ai log dei database. I log dei database sono memorizzati sul disco e sono usati per registrare le azioni delle transazioni. Se c’è un crash del database o del sistema, i log sono usati per il playback e rifanno le transazioni commitate durante un recovery.

La figura 11.1 fornisce una descrizione grafica di ciò che accade quando stai lavorando con un database in termini di logging.

Nella figura 11.1, vediamo uno spazio di tabelle ed i log. Entrambi risiedono sul disco, anche se suggeriamo che non stiano sullo stesso disco. Quando un’operazione di UPDATE avviene, per esempio, le pagine per le righe in questione saranno portate al buffer pool (memoria). I cambiamenti dell’aggiornamento saranno effettuati nel buffer pool, ed i vecchi e nuovi valori saranno memorizzati nel file di log, a volte immediatamente, e a volte quando un log buffer è pieno. Se un COMMIT è eseguito dopo un’ UPDATE, il vecchio e il nuovo valore saranno memorizzati immediatamente nel file di log. Questo processo è ripetuto per molte alter operazioni di SQL che sono realizzate sul database. Soltanto quando

page threshold specified in parameter CHNGPGS_THRES, will the pages in the buffer pool be “externalized” or written to the table space disk. The CHNGPGS_THRES parameter indicates the percentage of the buffer pool with “dirty” pages, that is, pages containing changes.

determinate condizioni sono riscontrate, come il raggiungimento della soglia della pagina del cambiamento specificata nel parametro CHNGPGS_THRES, le pagine nel buffer pool saranno “esteriorizzate” o saranno scritte sul disco dello spazio di tabella. Il parametro CHNGPGS_THRES indica la percentuale del buffer pool con pagine “dirty”, cioè, pagine che contengono cambiamenti.

From a performance point of view, it does not make sense to perform two writes for each COMMIT operation: One to write to the logs, and another one to write to the table space disk; that’s why “externalization” of the data to the table space disk only occurs when parameters such as the “chngpgs_thres” threshold are reached.

Dal punto di vista della prestazione, non ha senso effettuare due scritture per ciascun COMMIT: Uno per scrivere sul log e l’altro per scrivere sul disco dello spazio di tabella; ecco perché l’“esteriorizzazione” dei dati al disco dello spazio di tabella accade soltanto quando i parametri come la soglia “chngpgs_thres” sono raggiunti.

11.2 Types of logs / Tipi di log

There are two types of logs:

Ci sono due tipi di log:

Primary logs / Log primari

These are pre-allocated and the number of primary logs available is determined by the LOGPRIMARY db cfg parameter.

Questi sono pre-allocati e il numero dei log primari disponibili è determinato dal parametro LOGPRIMARY del cfg di db.

Secondary logs / Log secondari

These are dynamically allocated when needed by DB2. The maximum number of secondary logs is set by the db cfg parameter LOGSECOND. Dynamically allocating a log is costly; therefore, for day to day operations, ensure that you stay within your primary log allocation. Secondary log files are deleted when all the connections to a database are terminated.

Questi sono allocati dinamicamente quando DB2 ne ha bisogno. Il numero massimo dei log secondari è fissato dal parametro LOGSECOND del cfg di db. Allocare dinamicamente un log è costoso; quindi, per le operazioni giornaliere, assicurati di rimanere all’interno della tua primary log allocation. I file dei log secondari sono cancellati quando tutti i collegamenti ad un database sono terminati.

Infinite logging is possible if you set LOGSECOND to a value of -1; however, this is not recommended, and you may run out of file system space.

Il logging infinito è possibile se fissato LOGSECOND ad un valore di -1; tuttavia, questo non è suggerito e potresti finire lo spazio nel file system.

11.3 Types of logging / Tipi di logging

There are two types of logging: circular logging (default) and archive logging.

Ci sono due tipi di logging: circular logging (default) e archive logging.

11.3.1 Circular logging

Figure 11.2 demonstrates how circular logging works.

La figura 11.2 dimostra come funziona il circular logging.

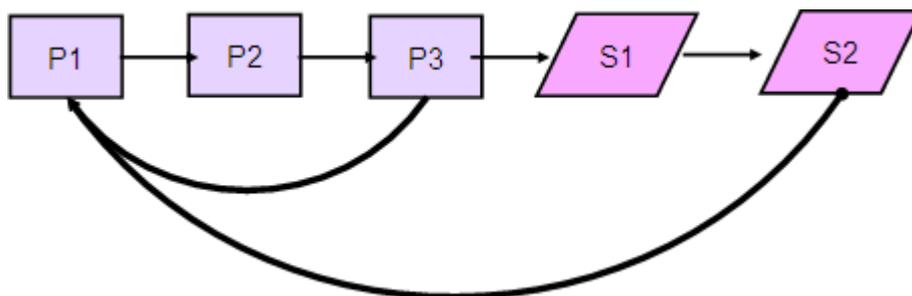


Figure 11.2 – Working with primary and secondary logs / Figura 11.2 – Lavorare con i log primari e secondari

In Figure 11.2 there are 3 primary logs, therefore we can assume that LOGPRIMARY was set to 3. For simplicity, let's just say there is only one transaction being performed in this example. As the transaction is performed, space starts filling up the log file P1, and then P2. If a commit occurs and the information is later externalized to the table space disk, then P1 and P2 can be overwritten, because the information is no longer needed for crash recovery (which will be discussed in more detail later in this chapter). If, on the other hand, the transaction is so long that it uses P1, P2, P3, and still needs more log space because the transaction has not been committed nor

Nella figura 11.2 ci sono 3 log primari, quindi possiamo supporre che LOGPRIMARY era stato regolato a 3. Per semplicità, diciamo che c'è soltanto una transazione effettuata in questo esempio. Mentre la transazione è realizzata, lo spazio comincia a riempirsi sul file di log P1 e poi su P2. Se avviene un commit e le informazioni sono successivamente esteriorizzate al disco dello spazio di tabella, allora P1 e P2 possono essere soprascritti, perché le informazioni non sono più necessarie per il recupero dopo un crash (che sarà discusso più in dettaglio successivamente in questo capitolo). Se, da l'altro lato, la transazione è così lunga che usa P1, P2, P3 ed ha ancora

externalized, then a secondary log (S1 in the figure) would be dynamically allocated. If the transaction still continues, more secondary logs are allocated until the maximum LOGSECOND logs are allocated. If more logs are needed, an error message indicating a log full condition is reached will be returned to the user, and the transaction will be rolled back.

bisogno di più spazio del log perché la transazione non è stata completata né esternalizzata, allora un log secondario (S1 nella la figura) verrà allocato dinamicamente. Se la transazione continua ancora, più log secondari sono allocati finché sono allocati al massimo LOGSECOND. Se più log sono necessari, un messaggio di errore che indica una condizione di log full viene catturato e sarà ritornata all'user e la transazione sarà rolled back (annullata).

11.3.2 Archival logging or log retain / Archival logging o log retain

In archival logging, also known as log retain logging, the log files are not overwritten, but are kept online or offline. Online archive logs are kept with the active logs which are still needed for crash recovery. Offline archive logs are moved to another media such as tape, and this can be done with USEREXIT routines. To enable archival logging set the LOGRETAIN db cfg parameter to YES.

Nell'archival logging, anche conosciuto come retain logging, i file di log non sono sovrascritti, ma sono mantenuti online o offline. I log di archivio online sono mantenuti con i log attivi che sono ancora necessari per il recupero dopo un arresto. I log di archivio offline sono spostati verso gli altri media come tape, e questo può essere fatto con le routine USEREXIT. Per permettere archival logging fissa il parametro del cfg di db LOGRETAIN a un valore di YES.

Archival logging is normally used in production systems, and because the logs are kept, it allows for database recovery back to as early as the oldest log file in most situations. With archival logging, a DBA can recover (to some extent) from errors caused by humans. For example, if a user of a system inadvertently starts performing an incorrect transaction that lasts for days, when the problem is detected later, the DBA could restore the system back to the time before the problem was introduced. However, there may be some manual manipulation required to rerun the transaction correctly.

L'archival logging è utilizzato normalmente nei sistemi di produzione e poiché i log sono mantenuti, permette il recupero di database fin al più vecchio file di log nella maggior parte delle situazioni. Con l'archival logging, un DBA può recuperare (in parte) dagli errori causati dalle persone. Per esempio, se un utente del sistema involontariamente comincia a realizzare una transazione errata che dura per giorni, quando il problema viene rilevato più tardi, il DBA potrebbe ristabilire il sistema al momento prima che il problema fosse introdotto. Tuttavia, potrebbe essere richiesta una certa manipolazione manuale per ritornare correttamente la transazione.

Archival logging is required for roll forward recovery and on-line backup. Figure 11.3 depicts the archival logging process.

L'archival logging è richiesto per il recupero del roll forward e il backup online. La figura 11.3 descrive il processo dell'archival logging.

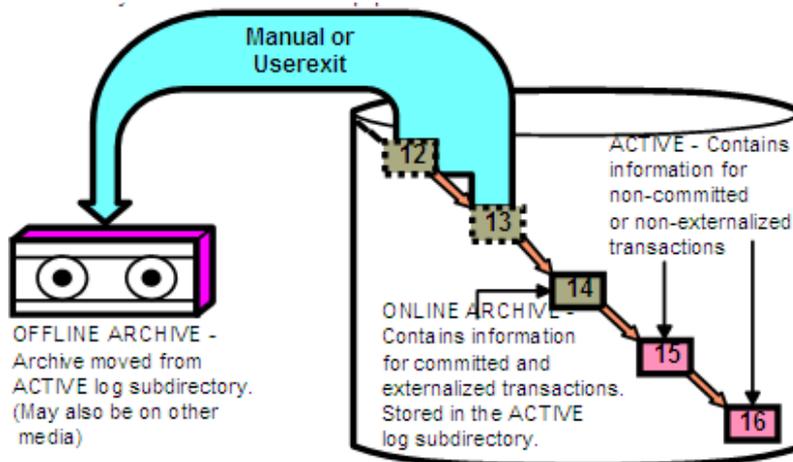


Figure 11.3 – Archival logging / Figura 11.3 – Archival logging

11.4 Database logging from the Control Center / Database logging dal Control Center

You can configure database logging from the Control Center by right-clicking on the database in question, and choosing “Configure Database Logging”. This is depicted in Figure 11.4.

Puoi configurare il database logging dal Control Center con un clic destro sul database in questione e scegliendo “Configure Database Logging”. Questo è mostrato in Figura 11.4.

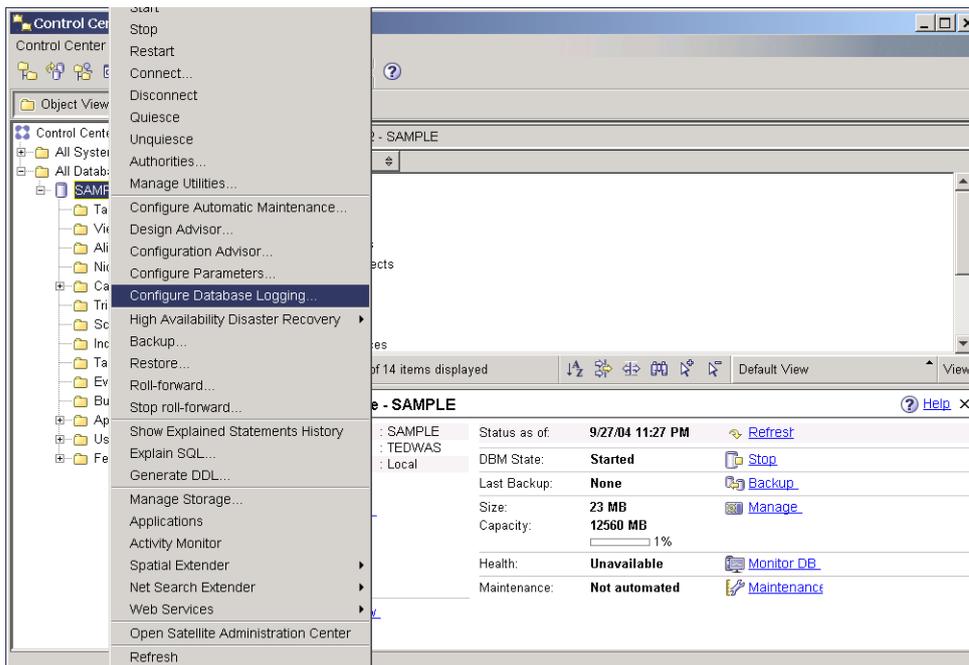


Figure 11.4 – Configuring database logging from the Control Center / Figura 11.4 – Configurare database logging dal Control Center

Figure 11.5 shows the Database Logging Wizard, where you can choose circular logging or archival logging.

La figura 11.5 mostra il Database Logging Wizard, in cui puoi scegliere circular logging o archival logging.

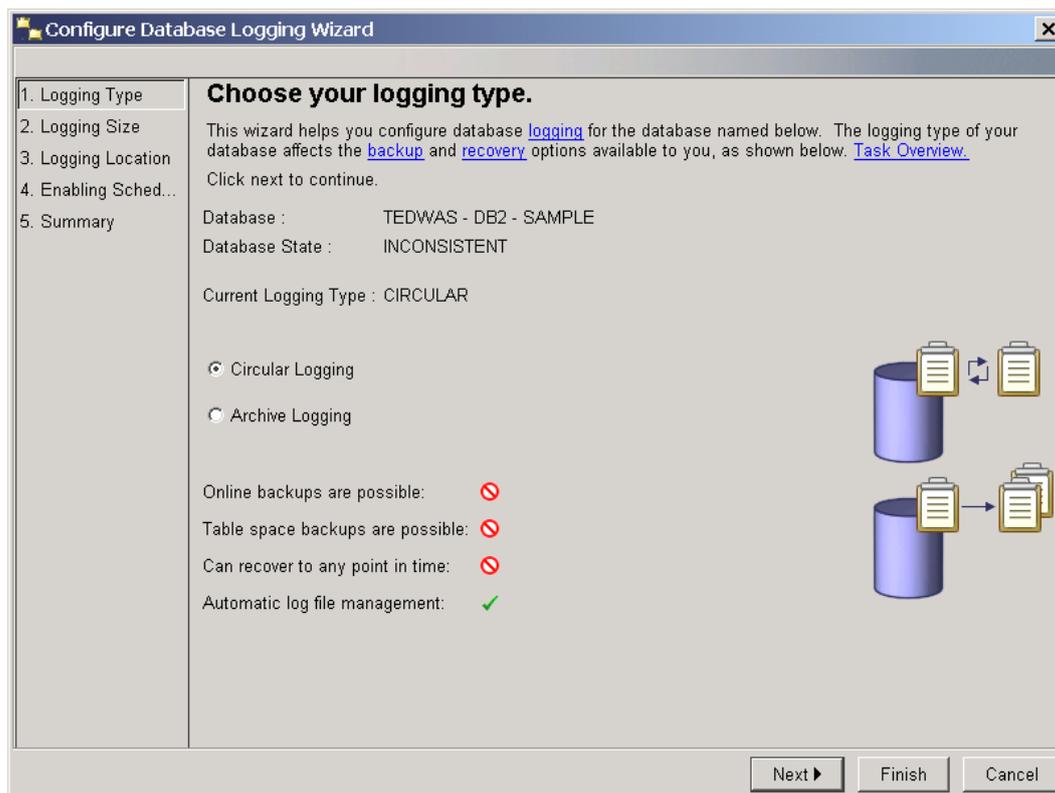


Figure 11.5 – Database Logging Wizard / Figura 11.5 - Database Logging Wizard

11.5 Logging parameters / Parametri di logging

There is a number of DB CFG parameters related to logging. Table 11.1 lists the main parameters.

Ci sono un certe numero di parametri di DB CFG relativi al logging. La tabella 11.1 elenca i parametri principali.

| Parameter / Parametro | Description | Descrizione |
|-----------------------|---|---|
| logbufsz | The amount of memory to use as a buffer for log records before writing these records to disk. | La quantità di memoria da usare come un buffer per il log record prima della scrittura di questi record al disco. |

| | | |
|--------------------|--|---|
| logfilisz | The size of each configured log, in number of 4KB pages | La dimensione di ogni log configurato, il numero di pagine da 4KB |
| logprimary | The number of primary logs of size logfilisz that will be created | Il numero dei log primari della dimensione di logfilisz che saranno creati |
| logsecond | The number of secondary log files that are created and used for recovery, if needed. | Il numero dei file di log secondari che sono creati ed usati per il recupero, se necessario. |
| logpath/newlogpath | The location in which active logs and future archived logs are placed. | Il luogo in cui i log attivi ed i futuri log archiviati sono messi. |
| mirrorlogpath | To protect the logs on the primary log path from disk failure or accidental deletion, you can specify that an identical set of logs be maintained on a secondary (mirror) log path | Per proteggere i log sul percorso primario dei log da un guasto del disco o cancellazione accidentale, puoi specificare che un insieme identico di log sia mantenuto su un percorso secondario del log (mirror) |
| loghead | The name of the log file that is currently active | Il nome del file di log che è attualmente attivo |
| userexit | Enable userexit program to copy logs offline | Permettere al programma userexit di copiare i log offline |
| softmax | Limits cost of crash recovery | Limita il costo del recupero da un arresto |
| logretain | Enables Archive Logging mode | Permette il modo Archive Logging |
| overflowlogpath | Similar to the OVERFLOW LOG PATH option of the ROLLFORWARD command; however, instead of specifying the OVERFLOW LOG PATH option for every ROLLFORWARD command issued, you can set this configuration parameter once. | Simile all'opzione OVERFLOW LOG PATH del comando ROLLFORWARD; tuttavia, invece di specificare l'opzione OVERFLOW LOG PATH per ogni comando ROLLFORWARD immesso, puoi fissare questo parametro di configurazione soltanto una volta. |
| blk_log_dsk_ful | Set to prevent disk full errors from being generated when DB2 cannot create a new log file in the active log path. Instead, DB2 will attempt to create the log file every five minutes until it | Fissato per prevenire la generazione di errori di disco pieno quando DB2 non può creare un nuovo file di log nel percorso attivo del log. Invece, DB2 tenterà di creare il file di |

| | | |
|--------------|--|--|
| | succeeds. Unblocked, read-only SQL may continue. | log ogni cinque minuti fino a che non riesce. Sbloccato, L'SQL read-only può continuare. |
| max_log | Percent of max active log space by transaction | Le percentuali di massimo spazio di log attivo per transazione |
| num_log_span | Number. of active log files for 1 active UOW | Numero di file di log attivi per 1 UOW attivo |

Table 11.1 – Logging parameters / Tabella 11.1 - Parametri di logging

11.6 Database backup / Backup del database

The DB2 `backup` command allows you to take a snapshot copy of your database at the time the command is executed. The simplest syntax that you can use to run this command is:

```
BACKUP DATABASE <dbname> [ TO <path> ]
```

Most commands and utilities can be performed online or offline. Online implies that other users may be connected and performing operations on the database while you execute your command. Offline means that no other users are connected to the database while you perform your operation. To allow for an online operation, add the keyword `ONLINE` to the command syntax, otherwise, by default the command will be assuming you are executing it offline.

For example, if you want to back up the database *sample* to the path `C:\BACKUPS` you can issue this command from the DB2 Command Window/Linux shell:

```
db2 BACKUP DB sample TO C:\BACKUPS
```

Note that the `C:\BACKUPS` directory must exist prior to executing the command. Also ensure there are no

Il comando `backup` di DB2 permette di fare una copia istantanea del database al momento in cui il comando è eseguito. La sintassi più semplice che puoi usare per far funzionare questo comando è:

La maggior parte dei comandi e delle utility possono essere effettuati online o offline. Online implica che altri utenti possano essere collegati e stiano eseguendo operazioni sul database mentre esegui il tuo comando. Offline significa che nessun altro utente è collegato al database mentre esegui l'operazione. Per eseguire un'operazione online, aggiungi la parola chiave `ONLINE` alla sintassi del comando, altrimenti, di default il comando presupporrà che stai eseguendolo offline.

Per esempio, se vuoi fare il back up del database *sample* nel path `C:\BACKUPS` puoi inserire questo comando dalla Command Window DB2 o dalla shell Linux:

Si noti che la cartella `C:\BACKUPS` deve esistere prima dell'esecuzione del comando. Inoltre assicurarti che non ci

connections to the database when you execute the above command, otherwise you will receive an error message since an offline backup cannot be performed when there are connections.

sia nessuno collegato al database quando esegui il comando precedente, altrimenti riceverai un messaggio di errore poiché un backup offline non può essere effettuato quando ci sono dei collegamenti attivi.

To find out if there are connections to databases in an instance, issue this command from the DB2 command window or Linux shell:

Per scoprire se ci sono collegamenti al database in un'instance, inserisci questo comando dalla command window DB2 o dalla shell Linux:

```
db2 list applications
```

To force all the connections from all databases in an instance, issue this command from the DB2 command window or Linux shell:

Per forzare tutti i collegamenti da tutti i database in un'instance, inserisci questo comando dalla command window DB2 o dalla shell Linux:

```
db2 force applications all
```

You may not want to run this last command in a production environment with many users, otherwise you would receive many calls from angry co-workers! Note as well that the last command runs asynchronously. This means that when you try to run the `backup` command right after, it may still not work. Wait a few seconds, and repeat the `backup` command if you received an error the first time.

Probabilmente non vorrai eseguire questo ultimo comando in un ambiente di produzione con molti utenti, altrimenti riceverei molte chiamate da colleghi arrabbiati! Nota pure che l'ultimo comando viene eseguito in modo asincrono. Ciò significa che quando provi a eseguire il comando `backup` immediatamente dopo, potrebbe non funzionare ancora. Attendi alcuni secondi e ripeti il comando `backup` se ricevi un errore la prima volta.

After a successful execution of the `backup` command, a new file containing the backup database image is created. The name of this file follows the convention shown in Figure 11.6.

Dopo una successiva esecuzione del comando `backup`, un nuovo file contenente l'immagine del backup del database è creato. Il nome di questo file segue la convenzione che appare in figura 11.6.

Linux/UNIX/Windows



Figure 11.6 – Backup image naming convention / Figura 11.6 - Convenzione per nominare immagini di backup

A type of “0” means that the backup is a full backup. A type of “3” for example, would mean that it is just a table space backup. The node is fixed to NODE0000 for non-partitioned databases, which is the case for all DB2 editions except DB2 Enterprise Edition with the DPF feature. The catalog node is also fixed to CATN0000. Refer to the DB2 manuals for more details.

When several backups are taken and stored in the same path, the timestamp at the end of the file name is used to distinguish between the backup images. As we will see on the next section, the RESTORE command can use this timestamp to restore from a specific backup.

Un tipo “0” significa che il backup è un backup completo. Un tipo “3” per esempio, significherebbe che è solo un backup dello spazio di tabella. Il nodo è fissato a NODE0000 per i database non-partitioned (divisi), che è il caso di tutte le edizioni DB2 tranne DB2 Enterprise Edition con il feature DPF. Il nodo del catalogo inoltre è fissato a CATN0000. Fai riferimento ai manuali DB2 per maggiori particolari.

Quando più backup sono presi e memorizzati nello stesso percorso, il timestamp alla fine del nome del file è usato per distinguere le immagini di backup. Come vedremo nella sezione seguente, il comando RESTORE può usare questo timestamp per fare il ripristino da un backup specifico.

Quicklab #10 – Scheduling a backup / Schedulare un backup

Objective

Although DB2 is able to automate several database maintenance activities, sometimes you will want to customize when certain activities occur. In this Quicklab, you will create a customized nightly backup schedule for the *EXPRESS* database.

Obiettivo

Anche se DB2 può automatizzare parecchie attività di manutenzione del database, a volte vorrai impostare quando determinate attività devono avvenire. In questo Quicklab, genererai un programma personalizzato per schedulare il backup del database *EXPRESS*.

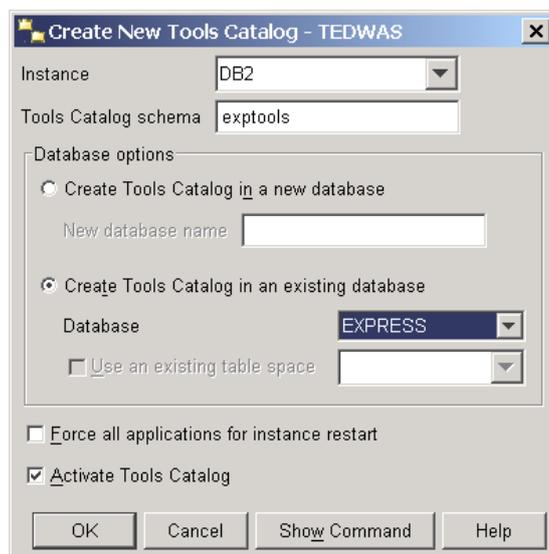
Procedure

1. From the Control Center object tree, navigate to Control Center => *All Databases*. Right-click on the *EXPRESS* database and select the *Backup* item. This launches the *Backup Wizard*.
2. The *Introduction* page of the wizard summarizes the current state of the database including the time of the last backup and logging method. Click the *Next* button to move to the next page of the wizard.
3. On the *Image* page of the wizard, select the destination of the backup image. You will typically select a different physical drive than where the existing database is stored. For now, create a new folder in the file system called `C:\db2backup`, and specify that folder as the backup location. In the wizard, select the *File System* item from the *Media Type* drop-down list. Click the *Add* button, select the

Procedura

1. Dall'albero del Control Center, naviga => *All Databases*. Fai clic destro sul database *EXPRESS* e seleziona la voce *Backup*. Ciò lancia il *Backup Wizard*.
2. La pagina *Introduzione* del wizard ricapitola lo stato attuale del database compreso il periodo dell'ultimo backup e del metodo di logging. Clicca il tasto *Next* per muoverti verso la pagina seguente del wizard.
3. Alla pagina *Image* del wizard, seleziona la destinazione dell'immagine di backup. Selezionerai tipicamente un drive fisico diverso da dove il database attuale è memorizzato. Per ora, crea una nuova cartella nel file system chiamato `C:\db2backup` e specifica quella cartella come il luogo dei backup. Nel wizard, seleziona l'elemento del *File System* dal drop-down list *Media Type*. Clicca il tasto *Add*, seleziona la

- folder you just created, and then click the *OK* button. Click the *Next* button to move to the next page of the wizard.
4. You can explore the *Options* and *Performance* pages, but the default options are usually sufficient because DB2 automatically performs the database backup in the most optimal way. Navigate to the *Schedule* page when you are finished exploring.
5. On *Schedule* page, if the scheduler has not yet been enabled, choose to enable it now. Select the system to create the tools catalog on and create a new tools catalog. Specify a schema for the tools catalog and choose to create it in the existing *EXPRESS* database.
- cartella che hai appena creato e poi clicca il tasto *OK*. Clicca il tasto *Next* per muoverti verso la pagina seguente del wizard.
4. Puoi esplorare le pagine di *Options* e *Performance*, ma le opzioni di default sono solitamente sufficienti perché DB2 effettua automaticamente il backup del database nel modo migliore. Naviga alla pagina *Schedule* quando hai finito l'esplorazione.
5. Alla pagina *Schedule*, se lo scheduler ancora non è stato abilitato, scegli di abilitarlo ora. Seleziona il sistema su cui creare il tool catalogo e crea un nuovo tool catalogo. Specifica uno schema per il tool catalogo e scegli di generarlo nel database *EXPRESS* esistente.



The tools catalog holds metadata about all the

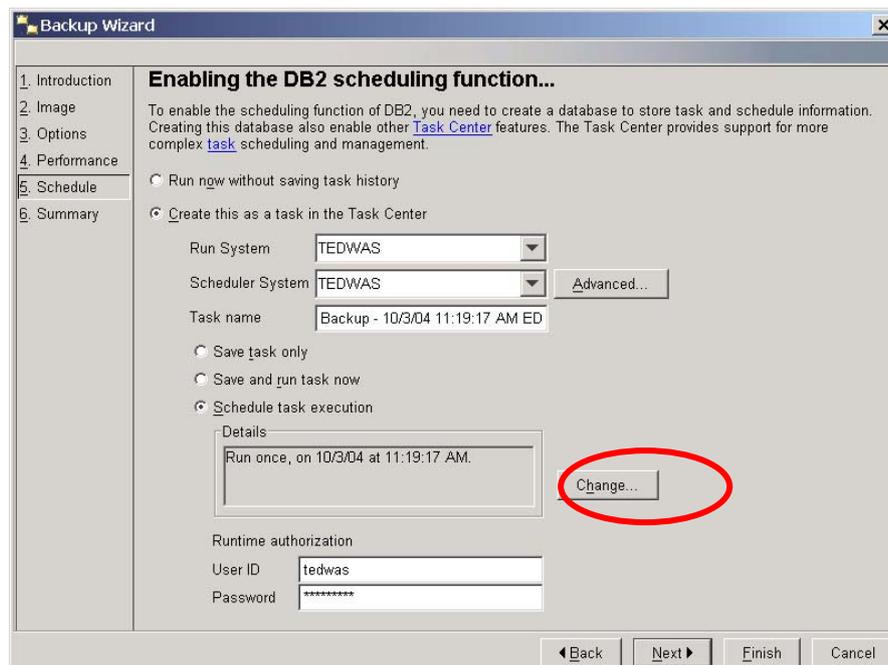
Il tool catalogo tiene i metadati su tutti i task schedulati. Clicca il

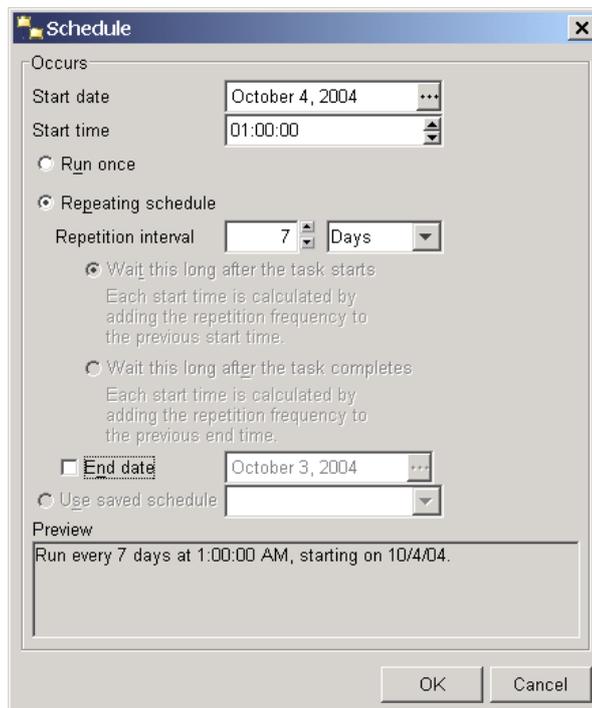
scheduled tasks. Click the *OK* button to continue. Click the *Next* button to move to the next page of the wizard once the tools catalog has been created.

6. On the *Schedule* page, choose to create a schedule for task execution. Schedule the backup to run each day, starting at 1AM. Click the *Next* button to move to the next page.

tasto *OK* per continuare. Clicca il tasto *Next* per passare alla pagina seguente del wizard una volta che il tool catalogo è stato creato.

6. Alla pagina *Schedule*, scegli di creare un programma per l'esecuzione. Programma il backup per essere eseguito ogni giorno, a partire dall' 1AM. Clicca il tasto *Next* per muoverti verso la pagina seguente.





7. On the *Summary* page, you can review the scheduled tasks that will be created. When you have reviewed the changes, click the *Finish* button to create the task.

9. Launch Task Center to view or modify the newly created backup task.

7. Alla pagina *Summary*, puoi esaminare i task schedulati che saranno creati. Quando hai esaminato i cambiamenti, clicca il tasto *Finish* per creare il task.

9. Lancia il Task Center per osservare o modificare il nuovo task di backup appena creato.

11.7 Database recovery / Recovery del database

A database recovery implies restoring your database from a backup and/or logs. If you just restore from a backup, you would be recreating the database as it existed at the time the backup was taken.

If archival logging was enabled before the backup, you can not only restore using a backup image, but also from the logs. As we will see in the next section, a roll-forward recovery allows you to restore from a backup, and then apply (roll-forward) the logs to the end of the logs, or to a specific point in time.

Note that the term *recovery* is used often in this section, but the command used for recovery is called *RESTORE*.

Un recupero di database implica il ristabilimento del tuo database da un backup e/o dai log. Se ristabilisci semplicemente da un backup, creerai un database come esisteva al momento in cui il backup era stato fatto.

Se l'archival logging era abilitato prima del backup, puoi non solo ristabilire usando un'immagine del backup, ma anche dai log. Come vedremo nella sezione seguente, un recupero "roll-forward" permette di ripristinare da un backup e poi applicare (roll-forward) i log alla fine dei log, o ad un punto specifico nel tempo.

Nota che il termine *recovery* è utilizzato spesso in questa sezione, ma il comando usato per il recupero è chiamato *RESTORE*.

11.7.1 Recovery types / Tipi di recupero

There are three types of recovery:

Ci sono tre tipi di recupero:

- **Crash or restart recovery / Crash o restart recovery**

Assume you are working on a desktop computer running important transactions to a DB2 database. Suddenly there is a power outage, or someone accidentally unplugs the power cord: what will happen to the database?

Presuppone che stai lavorando ad un desktop computer che fa funzionare le transazioni importanti ad un database DB2. C'è improvvisamente un'interruzione elettrica, o qualcuno disconnette casualmente il cavo di alimentazione: che cosa accadrà al database?

The next time you start your computer, and start DB2, crash recovery will automatically be executed. In crash recovery, DB2 will automatically run the command `RESTART DATABASE` and will read and redo/undo the transactions based on the active logs. When this command completes, you are

La prossima volta che avvii il tuo computer ed inizi DB2, il crash recovery sarà eseguito automaticamente. Nel crash recovery, DB2 farà funzionare automaticamente il comando `RESTART DATABASE` e leggerà e rifarà/annullerà le transazioni basate sui log attivi. Quando

guaranteed that your database will be in a consistent state, that is, whatever was committed will be saved, and whatever was not committed will be rolled back.

questo comando è completato, ti è garantito che il tuo database sarà in una condizione consistente, cioè, ciò che era stato completato sarà conservato e ciò che non era stato completato sarà annullato(rolled back).

- **Version or image recovery / Recupero di versione o di immagine**

This type of recovery implies that you are restoring only from a backup image; therefore, your database would be put in the state it was at the time the backup was taken. Any transactions performed on the database after the backup was taken would be lost.

Questo tipo di recupero implica che stai ripristinando soltanto da un'immagine di backup; quindi, il tuo database sarà messo nella condizione che era quando il backup era stato fatto. Tutte le transazioni realizzate sul database dopo che il backup è stato fatto andranno perse.

- **Roll-forward recovery**

With this type of recovery, you not only RESTORE from a backup image, but you also run the ROLLFORWARD command to apply the logs on top of the backup so that you can recover to a specified point in time. This type of recovery minimizes data loss.

Con questo tipo di recupero, non solo si ripristina da un'immagine di BACKUP, ma si fa anche funzionare il comando ROLLFORWARD per applicare i log sul backup in modo da poterlo recuperare da un punto specifico nel tempo. Questo tipo di recupero minimizza la perdita dei dati.

11.7.2 Database restore / Restore del database

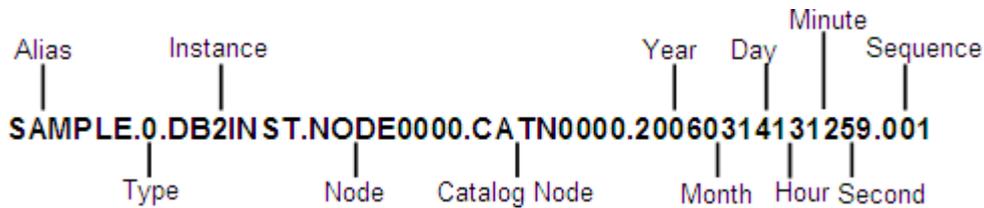
Use the RESTORE command to recover a database from a backup image. The following syntax is the simplest that can be used for this command:

Usa il comando RESTORE per recuperare un database da un'immagine di backup. La seguente sintassi è la più semplice che possa essere usata per questo comando:

```
RESTORE DATABASE <dbname> [from <path>] [taken at <timestamp>]
```

For example, if you had a backup image file of the *sample* database with this name:

Per esempio, se hai un file di immagine di backup del database *sample* con questo nome:



You could perform the following:

Potresti effettuare quanto segue:

```
RESTORE DB sample FROM <path> TAKEN AT 20060314131259
```

11.8 Other operations with BACKUP and RESTORE / Altri funzionamenti con BACKUP e RESTORE

The following lists some of the things that you can also do with the BACKUP and RESTORE commands. We encourage you to review the DB2 manuals for additional details.

- Backup a database in a 32-bit instance, and restore it on a 64-bit instance
- Restore over an existing database
- Use of a redirected restore when restoring into a system where there are a different number of disks than what was specified in the backup image
- Backup or restore just by table space, rather than the entire database
- Delta and incremental backups are allowed; delta backups record only the changes from backup to the next, while incremental backups record all the changes and accumulates them on each backup image
- Backup from flash copy (correct hardware required)
- Recover dropped tables (if the option was enabled for a given table)
- Backup from one platform (e.g. Windows) and restoring to another platform (e.g. Linux) is not possible. Use db2look and db2move for this scenario

I seguenti punti elencano alcune delle cose che potresti anche fare con i comandi BACKUP e RESTORE. Ti incoraggiamo ad esaminare i manuali DB2 per particolari aggiuntivi.

- Esecuzione del backup di un database in istanze a 32-bit e recuperarle su istanze a 64-bit
- Recupero sopra un database esistente
- Uso di un recupero riorientato quando ripristini un sistema in cui c'è un numero differente di dischi di quanti specificati nell'immagine di backup
- Backup o recupero dallo spazio di tabella, piuttosto che dell'intero database
- Delta ed sostegni incrementali sono permessi; i sostegni di delta registrano soltanto i cambiamenti da un backup al seguente, mentre i sostegni incrementali registrano tutti i cambiamenti e li accumulano su ogni immagine di backup
- Backup dalla flash copy (è richiesto hardware apposito)
- Recupero delle tabelle eliminate (se l'opzione è stata permessa per una data tabella)
- Il backup da una piattaforma (per esempio Windows) e il ripristino su un'altra piattaforma (per esempio Linux) non è possibile. Usa db2look e db2move per questo scenario

12

Chapter 12 – Maintenance Tasks / Capitolo 12 - Attività di Manutenzione

This chapter discusses some of the tasks required to keep your database well maintained. The overall direction in DB2 is to automate most of these tasks. DB2 Express-C edition, like all current DB2 editions, includes these automated capabilities. This self management capability is a great benefit to small and medium size companies who cannot hire a full time DBA to manage the data server. On the other hand, if a DBA is hired, he or she will have more free time to perform advanced activities that will add value to a company's bottom line.

Questo capitolo discute alcuni task richiesti per mantenere in buono stato il tuo database. L'idea generale in DB2 è quella di automatizzare la maggior parte di questi task. L'edizione di DB2 Express-C, come tutte le edizioni attuali di DB2, comprende queste capacità automatizzate. Questa capacità di auto-amministrazione è un notevole beneficio per le aziende medie e piccole che non possono assumere un DBA a tempo pieno per amministrare un data server. D'altra parte, se un DBA è assunto, lui o lei avrà maggiore tempo libero per effettuare attività avanzate aggiungendo valore al profitto dell'azienda.

Note:

For more information about maintenance tasks, watch this video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4302>

Nota:

Per ulteriori informazioni sulle attività di manutenzione, guardi questo video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4302>

12.1 REORG, RUNSTATS, REBIND

There are three main maintenance tasks in DB2, as depicted in Figure 12.1: REORG, RUNSTATS and REBIND.

Ci sono tre attività principali di manutenzione in DB2, come rappresentato nella figura 12.1: REORG, RUNSTATS e REBIND.

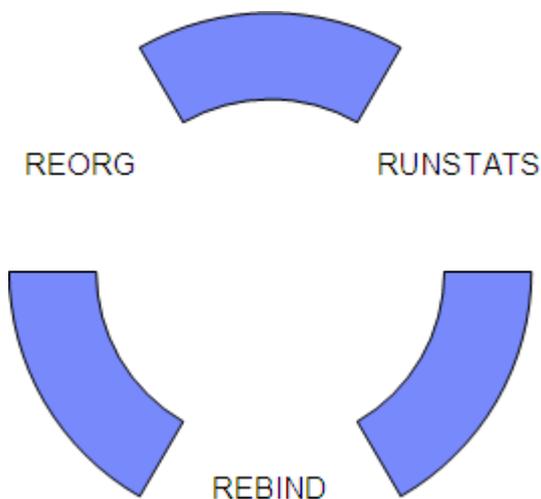


Figure 12.1 – Maintenance tasks: REORG, RUNSTATS, REBIND / Figura 12.1 - Attività di manutenzione: REORG, RUNSTATS, REBIND

Figure 12.1 shows that the maintenance tasks are performed in circular fashion. If a REORG is performed, it is recommended to also run a RUNSTATS, followed by a REBIND. After some time, the tables in a database will be modified due to UPDATE, DELETE and INSERT operations. At that time the cycle will start again with a REORG.

La figura 12.1 indica che le attività di manutenzione sono effettuate in modo circolare. Se un REORG è eseguito, è suggerito di eseguire anche un RUNSTATS, seguito da un REBIND. Dopo un certo tempo, le tabelle in un database saranno modificate a causa di operazioni di UPDATE, DELETE e INSERT. In quel momento il ciclo comincerà di nuovo con un REORG.

12.1.1 The REORG command / Il comando REORG

Over time, as you perform INSERT, UPDATE and DELETE operations on your database, your data starts getting more and more fragmented across the database pages. The REORG command reclaims wasted space and reorganizes data to make retrieval more efficient. Tables that frequently modified will benefit the most from REORG. You can REORG indexes as well as tables, and a REORG can be performed online or offline.

Col passare del tempo, poiché saranno eseguite operazioni di INSERT, UPDATE e DELETE sul tuo database, i dati cominceranno a diventare sempre più frammentati tra le pagine del database. Il comando REORG recupera lo spazio sprecato e riorganizza i dati per rendere il recupero più efficiente. Le tabelle che sono modificate frequentemente traggono il maggiore vantaggio da un REORG. Puoi effettuare un REORG sugli indici così come sulle tabelle, e un REORG può essere eseguito online o offline.

Offline REORG is faster and more efficient, but does not permit access to the table, while an online REORG allows access to the table, but can consume a lot of system resources; this works best for small tables.

Il REORG offline è più veloce e più efficiente, ma non permette l'accesso alla tabella, mentre un REORG online permette l'accesso alla tabella, ma può richiedere molte risorse di sistema; quindi questo funziona meglio su piccole tabelle.

Syntax / Sintassi:

```
REORG TABLE <tablename>
```

Example / Esempio:

```
REORG TABLE employee
```

The REORGCHK command can be used before a REORG to determine whether a table or index needs to be fixed.

Il comando REORGCHK può essere usato prima del REORG per determinare se una tabella o un indice deve essere riparato.

12.1.2 The RUNSTATS command / Il comando RUNSTATS

The DB2 Optimizer is “the brain” of DB2. It finds the most efficient access paths to locate and retrieve data. The optimizer is system cost-aware, and uses statistics of the database objects that are stored in catalog tables to maximize the database performance. For example, catalog tables have statistics about how many columns are present in a table, how many rows there are, how many and what type of indexes are available for a table, and so forth.

L'Optimizer DB2 è “il cervello” di DB2. Trova i percorsi di accesso più efficienti per trovare e recuperare i dati. L'ottimizzatore è cost-aware del sistema e usa le statistiche dei database object che sono memorizzati in tabelle di catalogo per massimizzare la prestazione del database. Per esempio, le tabelle di catalogo hanno statistiche di quante colonne sono presenti in una tabella, quante righe ci sono, quanti e che tipo di indici sono disponibili per una tabella e così via.

Statistics information is not updated dynamically. This is by design, as you would not want DB2 to be updating the statistics constantly for every operation performed to the database; this would negatively affect the entire database performance. Instead, DB2 provides the RUNSTATS command to update these statistics. It is essential to keep database statistics up to date. The DB2 optimizer can make radical changes in the access path if it thinks a table has 1

Le informazioni statistiche non sono aggiornate dinamicamente. Ciò deriva dal design, non si vuole che DB2 aggiorni costantemente le statistiche per ogni operazione eseguita sul database; ciò influisce negativamente su tutte le prestazioni del database. Invece, DB2 offre il comando RUNSTATS per aggiornare queste statistiche. È essenziale per mantenere le statistiche del database aggiornate. L'ottimizzatore DB2 può fare cambi

row versus 1 million rows. When database statistics are up to date, DB2 can choose a better access plan. The frequency of statistics gathering should be determined by how often the data in the table changes.

radicali nel percorso di accesso se pensa che una tabella abbia 1 riga rispetto a 1 milione di righe. Quando le statistiche del database sono aggiornate, DB2 può scegliere un percorso di accesso migliore. La frequenza di recupero delle statistiche dovrebbe essere determinata secondo quanto spesso i dati nella tabella cambiano.

Syntax / Sintassi:

```
RUNSTATS ON TABLE <schema.tablename>
```

Example / Esempio:

```
RUNSTATS ON TABLE myschema.employee
```

12.1.3 BIND & REBIND

After successfully running a RUNSTATS command, not all queries will use the latest statistics. Static SQL access plans are determined when you issue a BIND command, so the statistics used at the time may not be the same as the current ones. Figure 12.2 helps illustrate this idea.

Dopo aver eseguito con successo un comando RUNSTATS, non tutte le query useranno le ultime statistiche. I percorsi di accesso statici di SQL sono determinati quando fornisci un comando BIND, quindi le statistiche usate al momento possono non essere le stesse di quelle correnti. La figura 12.2 aiuta ad illustrare quest'idea.

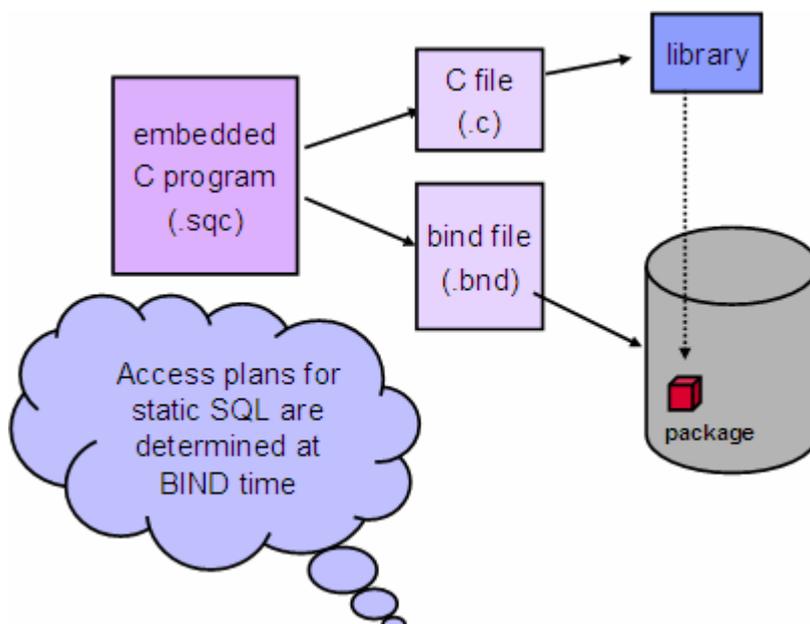


Figure 12.2 – Static SQL bind process / Figura 12.2 – Il processo statico di SQL bind

In Figure 12.2 an embedded C program (stored as a file with a “sqc” extension) is precompiled. After pre-compilation, two files are generated, a “.c” file containing the C code with all the SQL commented out; and a “.bnd” file containing all the SQL statements. The C file with the “.c” extension is compiled as usual with a C compiler, creating a “library” as shown in the top right hand side of the figure. The “.bnd” file is similarly bound, generating a package that is stored in the database. Binding is equivalent to compiling the SQL statements where the best access plan is determined based on the statistics available at the time, and then storing them in the package.

Now, what happens if 1 million rows are inserted into a table used in the SQL for this embedded C program? After the insertion, if a **RUNSTATS** is performed, the statistics will be updated, however the package will not be automatically updated to recalculate the access path based on the latest statistics. The `db2rbind` command can be used to rebind all the existing packages to take into account the latest stats.

Syntax / Sintassi:

```
db2rbind database_alias -l <logfile>
```

Example / Esempio:

To rebind all the packages of the *SAMPLE* database and store the output log in the file `mylog.txt`, issue this command:

```
db2rbind sample -l mylog.txt
```

Nella figura 12.2 un programma embedded C (memorizzato come un file con estensione “sqc”) è precompilato. Dopo la precompilazione, sono generati due file, un file “.c” che contiene il codice C con tutto l'SQL commentato; e un file “.bnd” che contiene tutti gli statement SQL. Il file C con estensione “.c” è compilato come al solito con il compilatore C, creando una “library” come indicato in alto a destra nella figura. Sul file “.bnd” viene fatto il binding, generando un package che è memorizzato nel database. Il binding è equivalente alla compilazione degli statement SQL dove il migliore percorso di accesso è determinato in base alle statistiche disponibili in quel momento e poi memorizzandole nel package.

Ora, che cosa accade se 1 milione di righe sono inserite in una tabella usata nel SQL per questo programma embedded C? Dopo l'inserimento, se un **RUNSTATS** è eseguito, le statistiche saranno aggiornate, ma comunque il package non sarà aggiornato automaticamente per ricalcolare il percorso di accesso basato sulle ultime statistiche. Il comando `db2rbind` può essere usato per il rebind di tutti i package esistenti per considerare l'ultime statistiche.

Per fare il rebind di tutti i package del database *SAMPLE* e memorizzare il log di output nel file `mylog.txt`, inserisci questo comando:

12.1.4 Maintenance tasks from the Control Center / Attività di manutenzione dal Control Center

From the Control Center you can REORG and RUNSTATS. Figure 12.3 shows you how.

Dal Control Center puoi eseguire i comandi REORG e RUNSTATS. La figura 12.3 illustra come.

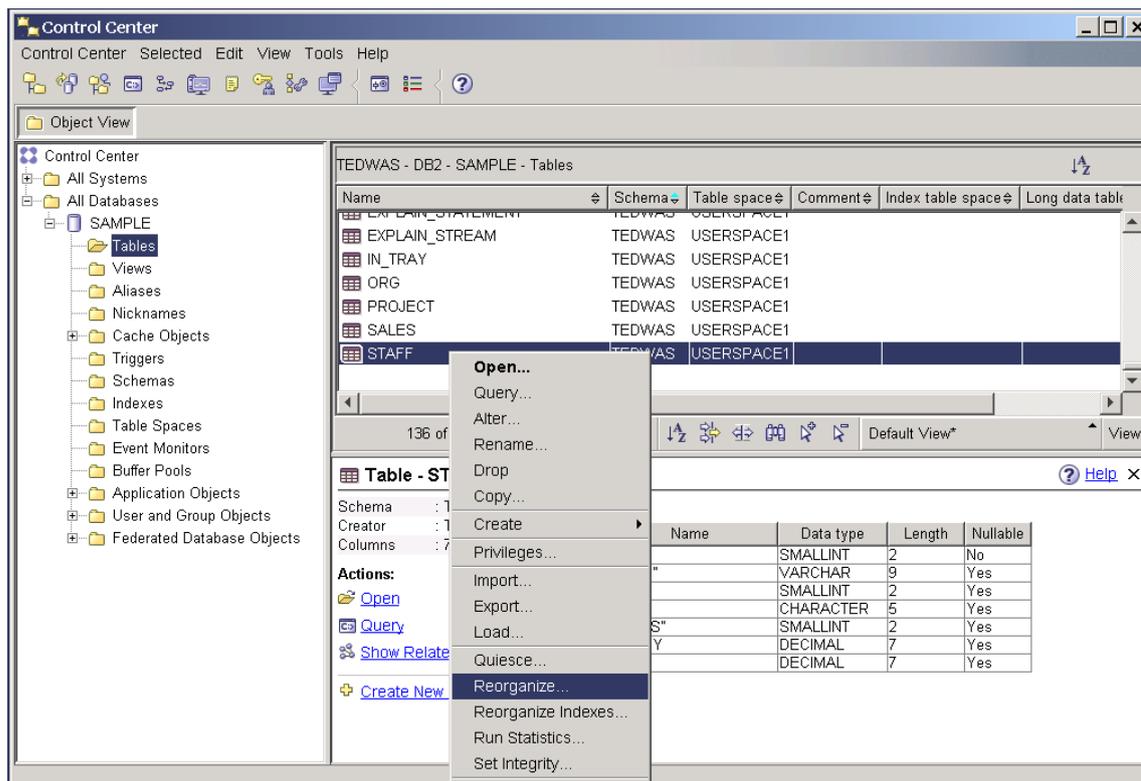


Figure 12.3 – REORG and RUNSTATS from the Control Center / Figura 12.3 – Eseguire REORG e RUNSTATS dal Control Center

You choose the table you would like to operate against, right-click on it and choose Reorganize (for REORG) or Run Statistics (for RUNSTATS).

Scegli la tabella su cui vorresti lavorare, fai un clic destro sulla tabella e scegli Reorganize (per il REORG) o scegli Run Statistics (per RUNSTATS).

The database operational view / La operational view del database

When you select a database, the database operational view on the bottom right side of the Control Center will provide information about the database, such as its size, when it was backed up last, whether automatic

Quando selezioni un database, la operational view del database nella parte in basso a destra del Control Center fornirà informazioni sul database, come la dimensione, l'ultima volta che è stato fatto un back up, se la

maintenance is set, etc. This view allows you to quickly identify maintenance needs for your database. Figure 12.4 shows this information.

manutenzione automatica è fissata, ecc. Questa view ti permette di identificare rapidamente la manutenzione necessaria per il tuo database. La figura 12.4 mostra questi informazioni.

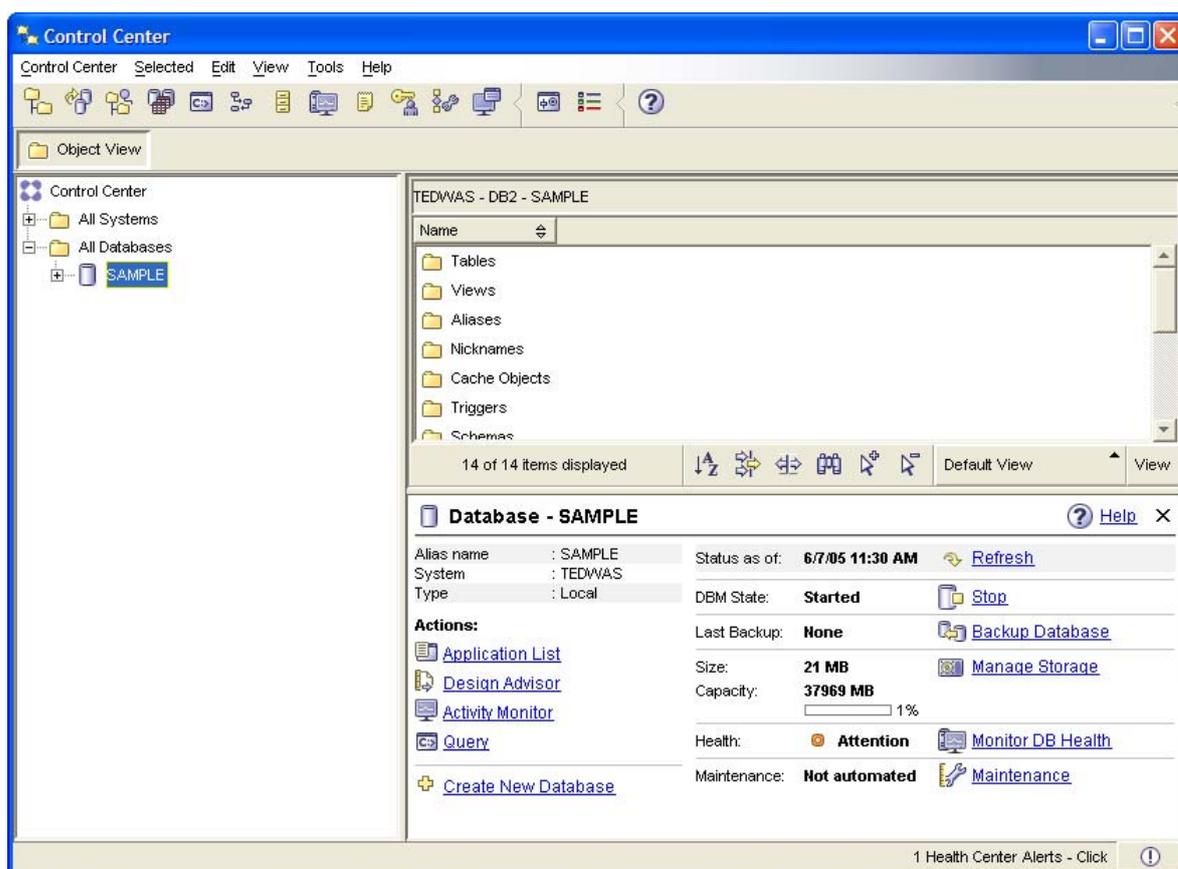


Figure 12.4 – The database operational view from the Control Center / Figura 12.4 - La operational view del database dal Control Center

12.2 Maintenance Choices / Scelte di manutenzione

There are three ways to perform maintenance tasks:

1. **Manual maintenance:** You perform maintenance activities manually when the need arises.
2. **Create scripts to perform maintenance:** You can create scripts with the maintenance commands, and schedule them regularly for execution.
3. **Automated maintenance:** Have DB2 automatically look after maintenance for you (REORG, RUNSTATS, BACKUP).

In this section we concentrate on automated maintenance.

Automatic maintenance consists of the following:

- The user defines a *maintenance window* where tasks can be executed with minimal disruption. For example, if the system has the least activity on Sundays from 2:00am to 4:00am, this time frame would work as a maintenance window.
- There are two maintenance windows: one for online operations, and another one for offline operations.
- DB2 will perform maintenance operations automatically only when needed during the maintenance window

Ci sono tre modi per eseguire le attività di manutenzione:

1. **Manutenzione manuale:** Eseguire manualmente le attività di manutenzione quando ce n'è bisogno.
2. **Manutenzione con script:** Creare script con i comandi di manutenzione e programmare l'esecuzione regolare di questi.
3. **Manutenzione automatizzata:** Fare in modo che DB2 si occupi automaticamente della manutenzione (REORG, RUNSTATS, BACKUP).

In questa sezione ci concentriamo sulla manutenzione automatizzata.

La manutenzione automatica consiste nel seguente:

- L'utente definisce una *maintenance window* in cui i task possono essere eseguiti con una interruzione minima. Per esempio, se il sistema ha meno attività le domeniche dalle 2 alle 4 di mattina, questo periodo di tempo funzionerebbe come maintenance window.
- Ci sono due maintenance window: una per le operazioni online e l'altra per le operazioni offline.
- DB2 esegue automaticamente le operazioni di manutenzione soltanto quando è necessario durante la maintenance window

From the Control Center, you can launch the Configure Automated Maintenance Wizard as shown in Figure 12.5.

Dal Control Center, puoi lanciare il Configure Automated Maintenance Wizard come mostrato in figura 12.5.

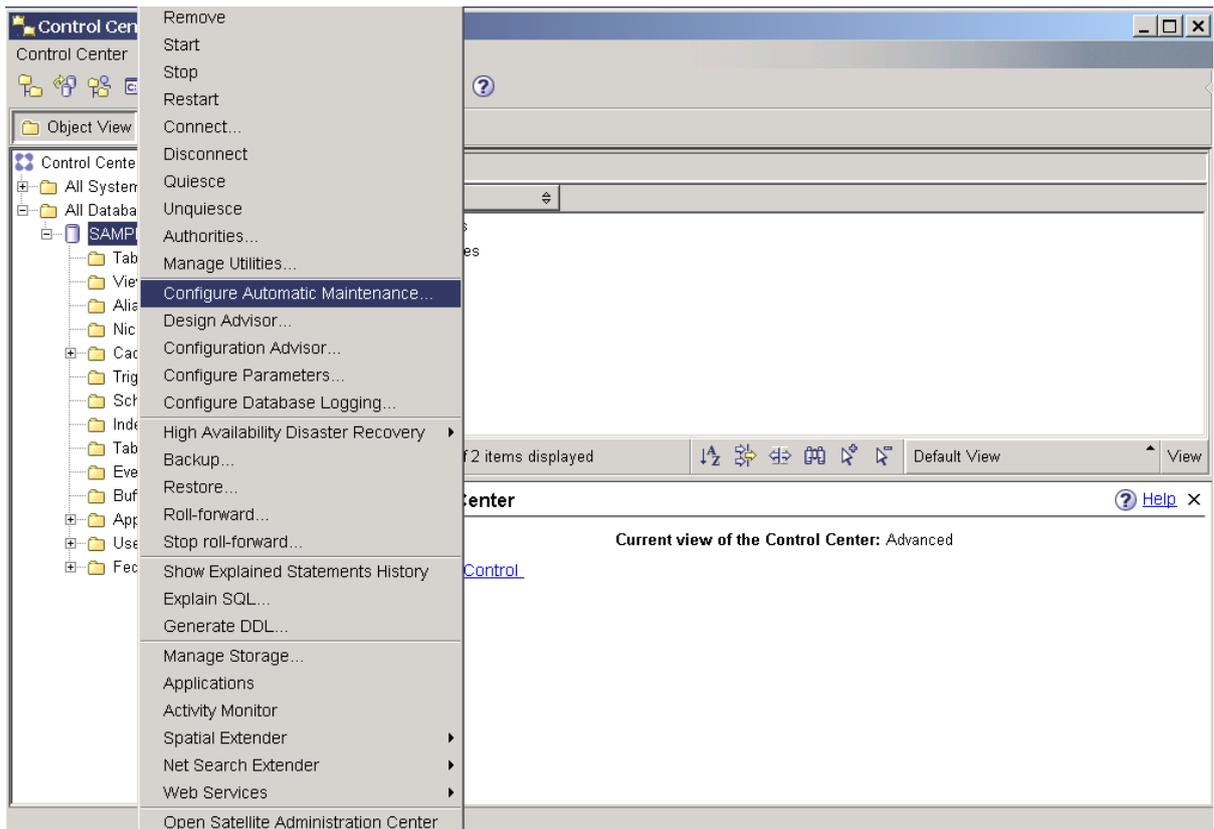


Figure 12.5 – Launching the Configure Automated Maintenance Wizard / Figura 12.5 – Lanciare il Configure Automated Maintenance Wizard

Figure 12.6 shows the Configure Automated Maintenance Wizard.

La figura 12.6 mostra il Configure Automated Maintenance Wizard.

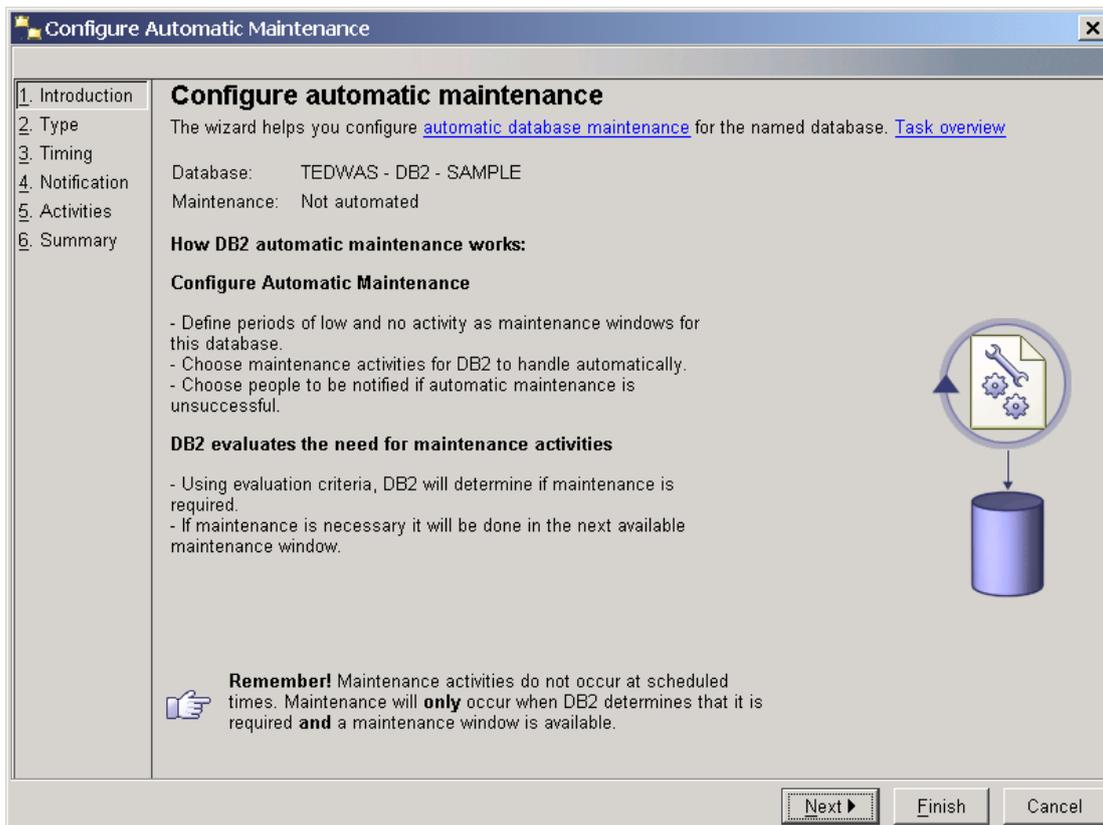


Figure 12.6 –The Configure Automated Maintenance Wizard / Figura 12.6 – Il Configure Automated Maintenance Wizard

Quicklab #11 – Configuring automated maintenance / Configurare la manutenzione automatizzata

Objective

In this Quicklab, with a few easy steps, you will configure automatically maintenance on the DB2 SAMPLE database.

Procedure

1. From the Control Center object tree, right-click on the *SAMPLE* database and select the *Configure Automatic Maintenance* menu item. This launches the *Configure Automatic Maintenance* wizard.
2. The *Introduction* page of the wizard displays the current automated maintenance settings. If you created the database with the automated maintenance option, then automated maintenance is already configured. You can use this wizard to re-configure the automated maintenance options. Click the *Next* button to move to the next page of the wizard.
3. The *Type* page of the wizard asks you to choose between disabling all automated maintenance, or changing your automated maintenance settings. Select the option to change the current automated maintenance settings. Click the *Next*.

Obiettivo

In questo Quicklab, con alcuni semplici passi, configurerai la manutenzione automatica sul database SAMPLE di DB2.

Procedura

1. Dall'albero del Control Center, fai clic destro sul database *SAMPLE* e seleziona l'elemento del menù *Configure Automatic Maintenance*. Questo lancerà il *Configure Automatic Maintenance* wizard.
2. La pagina *Introduction* del wizard visualizza gli attuali setting automatici di manutenzione. Se hai creato il database con l'opzione di manutenzione automatica, allora la manutenzione automatizzata è già configurata. Puoi usare questo wizard per modificare le opzioni di manutenzione automatica. Clicca *Next* per passare alla pagina seguente del wizard.
3. La pagina *Type* del wizard ti chiede di scegliere fra disabilitare tutte le attività di manutenzione automatica, o cambiare le tue impostazioni di manutenzione. Selezioni l'opzione per cambiare le attuali impostazioni di manutenzione automatica. Clicca *Next*.

4. The *Timing* page of the wizard asks you to specify the maintenance windows. Configure the *Offline* window to be every Saturday and Sunday night from midnight to 6AM as shown below. Click the *Change* button beside the offline maintenance window preview pane and choose the desired times. After specifying the required information, click the *OK* button to return to the wizard. Leave the online window as is (online maintenance can occur anytime). Click the *Next* button.

4. La pagina *Timing* del wizard richiede di specificare le maintenance windows (finestre di manutenzione). Configura la finestra *Offline* affinché sia: ogni sabato e domenica sera dalla mezzanotte alle 6 di mattina, come indicato sotto. Clicca *Change* accanto al pannello di preview della finestra di manutenzione e scegli i periodi che desideri. Dopo aver specificato le informazioni richieste, clicca *OK* per ritornare al wizard. Lascia la finestra online così com'è (la manutenzione online può avvenire in qualunque momento). Clicca *Next*.

Change Maintenance Window Specification - Offline Activity

Specify when automatic maintenance can occur.

During the specified time.

Outside the specified time.

Specify the start time and the duration of the maintenance window. The start time is specified using a 24-hour clock.

Start time

Duration hours

Specify how often this maintenance window occurs. A valid maintenance window must meet the conditions specified on both the Days of the Week tab and the Days of the Month tab.

Days of the week | Days of the month

All

Only on selected days

Monday Friday

Tuesday Saturday

Wednesday Sunday

Thursday

Preview

Offline automatic maintenance can occur **during** the following window.

Time 00:00 - 06:00 (6 hours)

Days of the week Saturday, Sunday

Days of the month ALL

Activities using this window None

OK Cancel Help

5. On the *Notification* page of the wizard, you can set up a contact in case an automated maintenance activity fails. Skip this step for now. Click the *Next* button.
 6. On the *Activities* page of the wizard, you can choose to individually automate or not to automate specific activities as well as choose to be notified of particular activities. In this example, ensure that all the *Automate* checkboxes are checked and the *Notify* checkboxes are unchecked. Click the *Next* button.
 7. Before proceeding to the next page of the wizard, you should configure the backup location of the database. Ideally, you want to store backups on a different physical drive in case of disk failure. From the *Activities* page, select the *Backup database* option, and then click the *Configure Settings* button.
 8. On the *Backup Criteria* tab of the Configure Settings window, choose the *Balance Database Recoverability with Performance* option. On the *Backup Location* tab, select the existing backup location and click the *Change* button. Specify a different location to perform the backup (ensure that enough room exists on the drive). On the *Backup Mode*
5. Sulla pagina *Notification* del wizard, puoi settare un contatto nel caso in cui un'attività di manutenzione automatica fallisca. Salta questo punto per ora. Clicca *Next*.
 6. Sulla pagina *Activities* del wizard, puoi scegliere di rendere automatiche o di non automatizzare attività specifiche individualmente così come è possibile scegliere di essere informati su particolari attività. In questo esempio, assicurati che tutte le checkbox *Automate* siano spuntate e che la checkbox *Notify* non lo sia. Clicca *Next*.
 7. Prima di proseguire alla pagina seguente del wizard, dovresti configurare una destinazione per i backup del database. Idealmente, devi memorizzare i backup su un drive fisico diverso per non avere problemi in caso di un guasto al disco. Dalla pagina *Activities*, seleziona l'opzione *Backup database* e poi clicca sul tasto *Configure Settings*.
 8. Sulla scheda *Backup Criteria* della finestra Configure Settings, scegli l'opzione *Balance Database Recoverability with Performance*. Sulla scheda Backup Location, seleziona una posizione di backup esistente e clicca sul tasto *Change*. Specifica una posizione differente per eseguire il backup (assicurandoti che ci sia abbastanza spazio sul drive).

tab, ensure that *Offline Backup* is selected. Click the *OK* button to close the *Backup Criteria* tab. Click the *Next* button.

9. The *Summary* page of the *Configure Automated Maintenance* wizard contains a summary of the choices you selected. Click the *Finish* button to accept and implement the changes.

Sulla scheda *Backup Mode*, assicurati che *Offline Backup* sia selezionato. Clicca *OK* per chiudere la scheda *Backup Criteria*. Clicca *Next*.

9. La pagina *Summary* del wizard *Configure Automated Maintenance* contiene un sommario delle scelte che hai selezionato. Clicca il tasto *Finish* per accettare ed applicare i cambiamenti.

13

Chapter 13 – Concurrency and Locking / Capitolo 13 - Concorrenza e Locking

This chapter discusses how to allow multiple users to access the same database at the same time without interfering with each other, and keeping their operations consistent. We will discuss the concepts of transactions, concurrency and locking.

Questo capitolo discute come dare a più utenti l'accesso allo stesso database nello stesso momento senza interferire l'un l'altro, mantenendo le loro operazioni consistenti. Discuteremo i concetti di transazione, concorrenza e locking.

Note:

For more information about concurrency and locking, watch this video:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4322>

Nota:

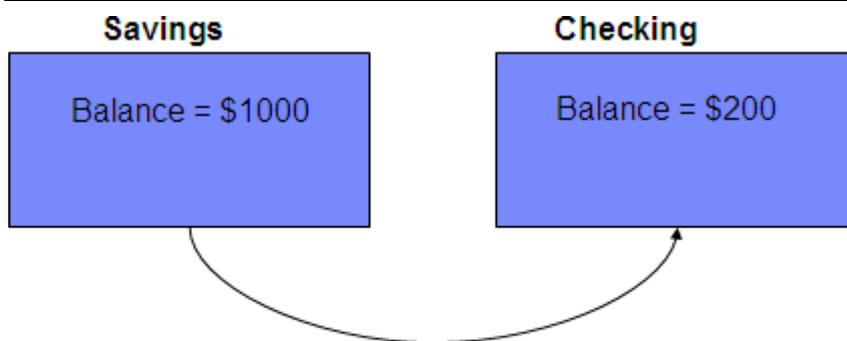
Per ulteriori informazioni sulla concorrenza e sul locking, guardi questo video:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4322>

13.1 Transactions / Transazioni

A transaction or unit of work consists of one or more SQL statements which, when executed, should be considered as a single unit; that is, if one of the statements in the transaction fails, the entire transaction fails, and any statements executed up to the point of failure are rolled back. A transaction ends with a COMMIT statement, which also signifies the start of a new transaction. Figure 13.1 provides an example of a transaction.

Una transazione o un'unità di lavoro consiste di uno o più statement SQL che, una volta eseguiti, dovrebbero essere considerati come una singola unità; cioè se uno degli statement nella transazione fallisce, tutte le transazioni falliscono, e ognuno degli statement eseguiti fino al punto del fallimento deve essere annullato (rolled back). Una transazione si conclude con uno statement COMMIT, che indica anche l'inizio di una nuova transazione. La figura 13.1 fa vedere un esempio di una transazione.



Transfer \$100 from Savings to Checking:

- Debit \$100 from Savings account
- Credit \$100 to Checking account

Figure 13.1 –An example of a transaction / Figura 13.1 - Un esempio di una transazione

In Figure 13.1, for example, you want to transfer 100 dollars from your savings account to your checking account. The following sequence of events may be required to achieve this task:

Debit \$100 from the savings account
Credit \$100 to the checking account

If the above sequence of events is not treated as a single unit of work, a transaction, imagine what would happen if a power failure occurred after the debit from the savings account, but before the checking account is credited. You would lose \$100!

Nella figura 13.1, per esempio, si vogliono trasferire 100 dollari dal tuo libretto di risparmio al tuo conto corrente. La seguente serie di eventi potrebbe essere necessaria per realizzare questa operazione:

Addebitare \$100 dal libretto di risparmio
Accreditare \$100 al conto corrente

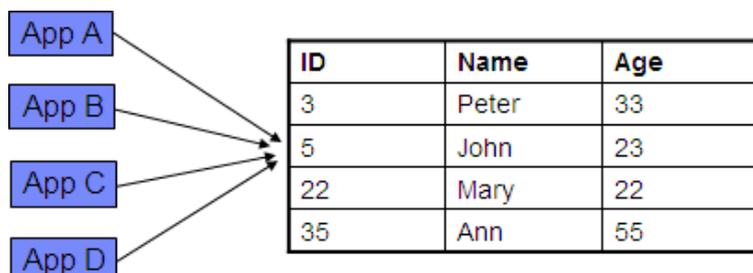
Se la serie di eventi qui sopra non venisse trattata come una singola unità di lavoro, transazione, immagina che cosa accadrebbe se si presentasse un'interruzione di corrente dopo il debito dal libretto di risparmio, ma prima che il conto corrente venga accreditato. Perderesti \$100!

13.2 Concurrency / Concorrenza

Concurrency implies that several users can work at the same time on the same database objects. DB2 was designed as a multi-user database. Access to data must be coordinated properly and transparently using a mechanism to ensure data integrity and consistency. Consider Figure 13.2 as an example.

La concorrenza implica che più utenti possano lavorare allo stesso tempo sugli stessi oggetti del database. DB2 è stato progettato come un database multi-utente. L'accesso ai dati deve essere coordinato correttamente e in modo trasparente usando un meccanismo per assicurare l'integrità e

la consistenza di dati. Considera la figura 13.2 come esempio.



**Figure 13.2 –An example of concurrency, and the need for concurrency control /
Figura 13.2 - Un esempio di concorrenza e l'esigenza del controllo di concorrenza**

In Figure 13.2, there are four applications, App A, App B, App C, and App D that are trying to access the same row (row 2) in a table. Without any concurrency control, all of the applications could perform operations against the same row. Assuming all of the applications are updating the Age column for row 2 with different values, the application which performs the update the last will likely be the “winner” in this situation. It should be obvious in this example that some sort of concurrency control is required to guarantee consistent results. This concurrency control is based on using locks.

Locking and concurrency concepts go hand in hand. Locking temporarily stops other applications from performing their operation until another operation finishes. The more locking there is in a system, the less concurrency is possible. On the other hand, the less locking there is in a system, the more concurrency is possible.

Nella figura 13.2, ci sono quattro applicazioni, App A, App B, App C ed App D che stanno provando ad accedere alla stessa riga (la riga 2) in una tabella. Senza alcun controllo di concorrenza, tutte le applicazioni potrebbero operare sulla stessa riga. Assumendo che tutte le applicazioni stiano aggiornando la colonna Age per la riga 2 con differenti valori, l'applicazione che esegue l'aggiornamento probabilmente sarà l'ultima e quindi il “winner” in questa situazione. Dovrebbe essere evidente in questo esempio che una certa sorta di controllo di concorrenza sia richiesto per garantire che i risultati siano consistenti. Questo controllo di concorrenza è basato sull'uso del locking.

I concetti di concorrenza e di locking vanno di pari passo. Il locking ferma temporaneamente le altre applicazioni dall'eseguire le loro operazioni finché un'altra operazione non abbia finito. Maggiore è il locking in un sistema, minore è la possibilità di concorrenza. Dall'altro lato, meno locking c'è in un sistema, maggiore è la concorrenza possibile.

Locks are acquired automatically as needed to support a transaction and are released when the transaction terminates (using either a `COMMIT` or `ROLLBACK` command). Locks can be acquired on tables or rows. There are two basic types of locks:

- Share locks (S locks) – acquired when an application wants to read and prevent others from updating the same row
- Exclusive locks (X locks) – acquired when an application updates, inserts, or deletes a row

Now consider Figure 13.3, which is similar to Figure 13.2, but it now shows a lock.

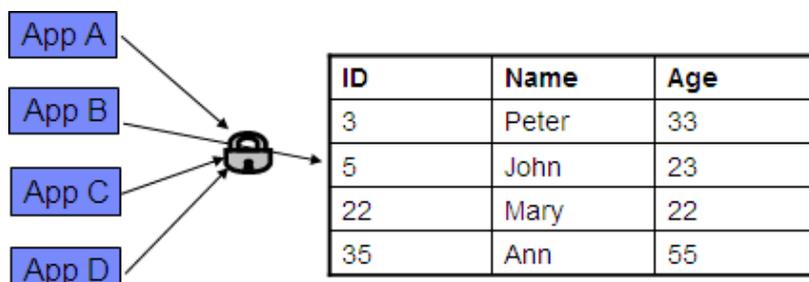


Figure 13.3 –An example of concurrency, and the need for locks / Figura 13.3 - Un esempio di concorrenza e l'esigenza di locks (serrature)

For example, in Figure 13.2, if App B is the first one accessing row 2, and is performing an `UPDATE`, App B holds an X lock on the row. When App A, App C and App D try to access the same row, they won't be able to `UPDATE` it because of the X lock. This control allows for consistency and integrity of the data.

I locking vengono acquisiti automaticamente non appena richiesti per supportare una transazione e sono rilasciati quando la transazione termina (usando il comando `COMMIT` o `ROLLBACK`). I locks possono essere acquisiti sulle tabelle o sulle righe. Alla base, ci sono due tipi di locking:

- Share locks (S locks) - acquisiti quando un'applicazione vuole leggere ed impedire ad altre applicazioni l'aggiornamento della stessa riga
- Exclusive locks (X locks) - acquisito quando un'applicazione aggiorna, inserisce, o cancella una riga.

Ora considera la figura 13.3, che è simile alla figura 13.2, ma mostra un locking.

Per esempio, nella figura 13.2, se App B è il primo ad accedere alla riga 2, e sta eseguendo un `UPDATE`, App B mantiene un lock X sulla riga. Quando App A, App C ed App D proveranno ad accedere alla stessa riga, non potranno fare un `UPDATE` a causa del lock X. Questo controllo permette la consistenza ed l'integrità dei dati.

13.3 Problems without concurrency control / Problemi senza controllo di concorrenza

Without some form of concurrency control, the following problems may be encountered

- ▶ Lost update
- ▶ Uncommitted read
- ▶ Non-repeatable read
- ▶ Phantom read

Senza una certa forma di controllo di concorrenza, si possono incontrare i seguenti problemi

- ▶ Perdita dell'aggiornamento
- ▶ Lettura non portata a termine
- ▶ Lettura non-ripetibile
- ▶ Lettura fantasma

13.3.1 Lost update / Update perso

Lost update is a problem similar to the one explained earlier in this section where the application performing the last update, will be the "winner".

La perdita dell'aggiornamento è un problema simile a quello spiegato precedentemente in questa sezione dove l'applicazione che eseguiva l'ultimo aggiornamento, era il "winner".

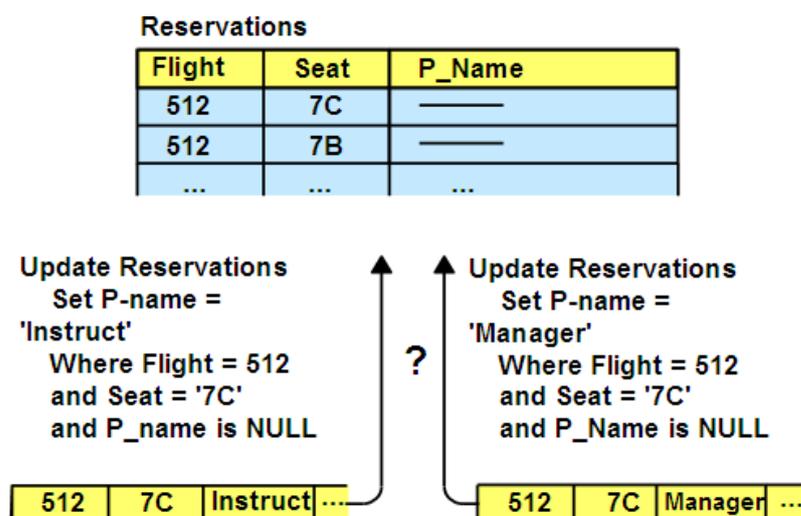


Figure 13.4 – Lost Update / Figura 13.4 – Perdita dell'aggiornamento

In Figure 13.4 there are two applications attempting to update the same row. The one on the left is application App1, and the one on the right is application App2. The sequence of events is then:

1. App1 updates a row
2. App2 updates the same row
3. App1 commits
4. App2 commits

Nella figura 13.4 ci sono due applicazioni che tentano di aggiornare la stessa riga. Quella a sinistra è l'applicazione App1 e quella a destra è l'applicazione App2. La serie di eventi quindi è:

1. App1 aggiorna una riga
2. App2 aggiorna la stessa riga
3. App1 fa commit
4. App2 fa commit

App1's update is lost when App2 make its update, hence the term "Lost Update".

L'aggiornamento di App1 è perso quando App2 fa l'aggiornamento, da qui il termine "perdita dell'aggiornamento".

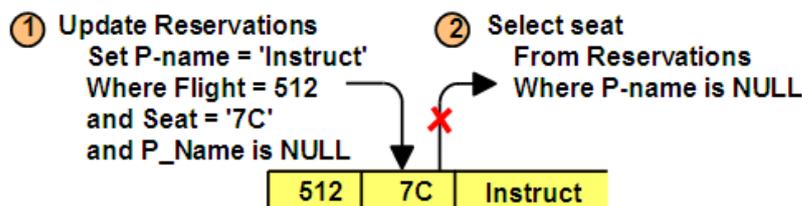
13.3.2 Uncommitted read / Read non committata

An uncommitted read, or "dirty read" allows for an application to read information that has not been committed, and therefore is not necessarily accurate.

Una lettura (read) non committata (portata a termine), o un "dirty read" ("lettura sporca") permette a un'applicazione di leggere informazioni che non sono state commitate e quindi non sono necessariamente corrette.

Reservations

| Flight | Seat | P_Name |
|--------|------|--------|
| 512 | 7C | _____ |
| 512 | 7B | _____ |
| ... | ... | |



③ Roll back

④ Incorrect results set

Figure 13.5 – Uncommitted Read / Figura 13.5 – Read non portata a termine

Figure 13.5 follows this sequence of events:

La figura 13.5 segue questa serie di eventi:

1. App1 updates a row
2. App2 reads the new value from that row
3. App1 rolls back its changes to that row

1. App1 aggiorna una riga
2. App2 legge il nuovo valore da quella riga
3. App1 fa i roll back i relativi ai cambiamenti su quella riga

App2 is reading uncommitted data, and hence invalid data, which is why this problem is called an "uncommitted read".

App2 sta leggendo i dati non portati a termine e quindi i dati non sono validi, ecco perché questo problema è chiamato "uncommitted read" (lettura non portata a termine).

13.3.3 Non-repeatable read / Read non-ripetibile

A non-repeatable read implies that you cannot obtain the same result after performing the same read in the same operation.

Una lettura (read) non-ripetibile implica che non puoi ottenere lo stesso risultato dopo l'esecuzione della stessa lettura nella stessa operazione.

| FLIGHT | SEAT | NAME | DESTINATION | ORIGIN |
|--------|------|------|-------------|----------|
| 512 | 7B | — | DENVER | DALLAS |
| | | | | |
| | | | | |
| 814 | 8A | — | SAN JOSE | DENVER |
| | | | | |
| 134 | 1C | — | HONOLULU | SAN JOSE |
| | | | | |

Figure 13.6 – Non-repeatable Read / Figura 13.6 - Read non-ripetibile

In Figure 13.6, consider if you are trying to book a flight from Dallas to Honolulu. The sequence of events is:

1. App1 opens a cursor (also known as a result set) obtaining what you see in Figure 13.6
2. App2 deletes a row that qualified for the cursor (for example, the row with destination "San Jose")
3. App2 commits changes
4. App1 closes and reopens the cursor

In this case, since App1 would not get the same data on a repeated read, it cannot reproduce the data set; that's why this problem is called "non-repeatable read".

13.3.4 Phantom read / Read fantasma

The phantom read problem is similar to the non-repeatable read problem, but the difference is that on subsequent fetches, you may obtain additional rows

Nella figura 13.6, considera di star provando a prenotare un volo da Dallas ad Honolulu. La serie di eventi è:

1. App1 apre un cursore (anche conosciuto come un insieme di risultati) ottenendo quello che vedi nella figura 13.6
2. App2 cancella una riga qualificata per il cursore (per esempio, la riga con la destinazione "San Jose")
3. App2 fa un commit dei cambiamenti
4. App1 chiude e riapre il cursore

In questo caso, poiché App1 non otterrebbe gli stessi dati ripetendo la lettura, non può riprodurre l'insieme di dati; ecco perché questo problema è chiamato "lettura non-ripetibile".

Il problema della lettura (read) fantasma è simile al problema della lettura non-ripetibile, ma la differenza è che sulle acquisizioni successive, puoi ottenere

rather than fewer rows. Figure 13.7 provides an example of this problem.

file in più piuttosto che file in meno. La figura 13.7 fornisce un esempio di questa problema.

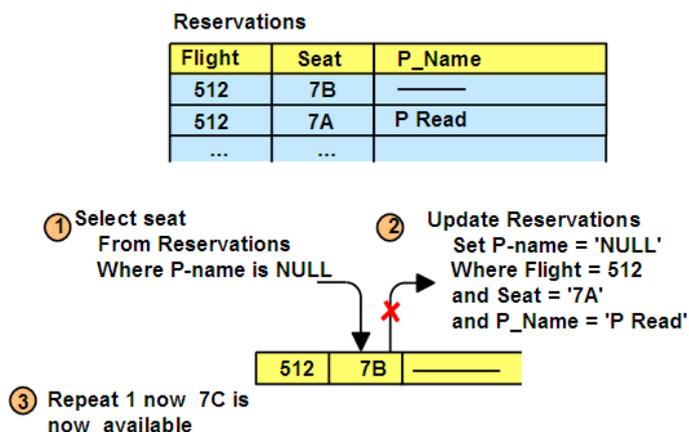


Figure 13.7 – Phantom read / Figura 13.7 – Read fantasma

Figure 13.7 shows the following sequence of events:

La figura 13.7 mostra la seguente serie di eventi:

1. App1 opens a cursor
2. App2 adds a row to the database that would qualify for the cursor
3. App2 commits changes
4. App1 closes and reopens cursor

1. App1 apre un cursore
2. App2 aggiunge una riga al database che sarebbe qualificata per il cursore
3. App2 fa un commit dei cambiamenti
4. App1 chiude e riapre il cursore

In this case, App1 would not get the same data on a repeated read, it would get more rows, that's why this problem is called "phantom read".

In questo caso, App1 non otterrebbe gli stessi dati ripetendo la lettura, esso otterrebbe più file, ecco perché questo problema è chiamato "lettura fantasma".

13.4 Isolation Levels / Livelli di isolamento

You can think of isolation levels as locking policies where, depending on the isolation level chosen, you may get different behaviors for database locking with an application.

Puoi pensare ai livelli di isolamento come politiche di locking dove, a seconda del livello di isolamento scelto, può ottenere comportamenti diversi per il locking del database con un'applicazione.

DB2 provides different levels of protection to isolate data:

DB2 fornisce diversi livelli di protezione per isolare i dati:

- | | |
|-------------------------|--------------------------------------|
| ▶ Uncommitted Read (UR) | ▶ Lettura non portata a termine (UR) |
| ▶ Cursor Stability (CS) | ▶ Stabilità del cursore (CS) |
| ▶ Read Stability (RS) | ▶ Stabilità della lettura (RS) |
| ▶ Repeatable Read (RR) | ▶ Lettura ripetibile (RR) |

13.4.1 Uncommitted read / Read non committato

Uncommitted read is also known as “dirty read”. It is the lowest level of isolation, and provides the highest degree of concurrency. No row locks are obtained on read operations, unless another application attempts to drop or alter a table; and update operations act as if using the cursor stability isolation level.

Un read non committato (una lettura non portata a termine) è anche conosciuta come “dirty read” (lettura sporca). È il più basso livello di isolamento, e fornisce il più alto grado di concorrenza. Nessun lock di riga è ottenuto sulle operazioni di lettura, a meno che un'altra applicazione tenti di cancellare o alterare una tabella; e le operazioni di aggiornamento si comportano come quando si usa il livello di isolamento di stabilità del cursore.

Problems still possible with this isolation level:

- ▶ Uncommitted read
- ▶ Non-repeatable read
- ▶ Phantom read

Problemi ancora possibili con questo livello di isolamento:

- ▶ Lettura non portata a termine
- ▶ Lettura non-ripetibile
- ▶ Lettura fantasma

Problems prevented with this isolation level:

- ▶ Loss of update

Problemi evitati con questo livello di isolamento:

- ▶ Perdita di aggiornamento

13.4.2 Cursor stability / Stabilità del cursore

Cursor stability is the default isolation level. It provides a minimal degree of locking. Basically, with this isolation level the “current” row of a cursor is locked. If the row is only read, the lock is held until a new row is fetched or the unit of work is terminated. If the row is updated, the lock is held until the unit of work is terminated.

La stabilità del cursore è il livello di isolamento di default. Fornisce un grado minimo di locking. Fondamentalmente, con questo livello di isolamento la riga “corrente” di un cursore è bloccata. Se la riga è soltanto letta, il lock è tenuto fino a che una nuova riga viene acquisita o fino a che l'unità di lavoro è terminata. Se la riga è aggiornata, il lock è tenuto fino a che l'unità di lavoro è terminata.

Problems still possible with this isolation

Problemi ancora possibili con questo

level:

- ▶ Non-repeatable read
- ▶ Phantom read

Problems prevented with this isolation level:

- ▶ Loss of update
- ▶ Uncommitted read

livello di isolamento:

- ▶ Lettura non-ripetibile
- ▶ Lettura fantasma

Problemi evitati con questo livello di isolamento:

- ▶ Perdita di aggiornamento
- ▶ Lettura non portata a termine

13.4.3 Read stability / Stabilità della read

With read stability, all the rows an application retrieves within a unit of work are locked. For a given cursor, it locks all rows that qualify for the result set. For example, if you have a table containing 10,000 rows and the query returns 10 rows, then only those 10 rows are locked. Read stability uses a moderate degree of locking.

Con la stabilità di read (lettura), tutte le righe che un'applicazione richiama nell'unità di lavoro sono bloccate. Un certo cursore dato, farà lock su tutte le file che saranno qualificate per l'insieme dei risultati. Per esempio, se hai una tabella che contiene 10.000 righe e la query ritorna 10 righe, allora soltanto quelle 10 righe sono bloccate. La stabilità di lettura usa un grado moderato di locking.

Problems still possible with this isolation level:

- ▶ Phantom read

Problems prevented with this isolation level:

- ▶ Loss of update
- ▶ Uncommitted read
- ▶ Non-repeatable read

Problemi ancora possibili con questo livello di isolamento:

- ▶ Lettura fantasma

Problemi evitati con questo livello di isolamento:

- ▶ Perdita di aggiornamento
- ▶ Lettura non portata a termine
- ▶ Lettura non-ripetibile

13.4.4 Repeatable read / Read ripetibile

Repeatable read is the highest isolation level. It provides the highest degree of locking, and the least concurrency. Locks are held on all rows processed to build the result set; that is, rows not necessarily in the final result set may be locked. No other application can update, delete, or insert a row that would affect the result set until the unit of work completes. Repeatable read guarantees that the same query issued by an application more than once in a unit of

Il read (la lettura) ripetibile è il livello più elevato di isolamento. Dà il più alto grado di locking e il minor grado di concorrenza. I locks sono tenuti su tutte le righe processate per costruire l'insieme del risultato; cioè righe non necessariamente nell'insieme del risultato finale possono essere bloccate. Nessun'altra applicazione può aggiornare, cancellare, o inserire una riga che ha effetto sull'insieme del risultato finché l'unità di lavoro non

work will give the same result each time.

finisce. Le letture ripetibili garantiscono che la stessa query eseguita da un'applicazione, più di una volta in un'unità di lavoro, darà sempre lo stesso risultato.

Problems still possible with this isolation level:

- ▶ none

Problemi ancora possibili con questo livello di isolamento:

- ▶ Nessuno

Problems prevented with this isolation level:

- ▶ Loss of update
- ▶ Uncommitted read
- ▶ Non-repeatable read
- ▶ Phantom read

Problemi evitati con questo livello di isolamento:

- ▶ Perdita di aggiornamento
- ▶ Lettura non portata a termine
- ▶ Lettura non-ripetibile
- ▶ Lettura fantasma

13.4.5 Comparing isolation levels / Confronto tra i livelli di isolamento

Figure 13.8 compares the different isolation levels for a fetch. In the figure, we see that isolation level uncommitted read (UR) takes no locks. Isolation level cursor stability (CS) takes a lock for row 1 when it is fetching it, but releases it as soon as it fetches row 2, and so on. For isolation levels read stability (RS) or repeatable read (RR), any row that is fetched will be locked, and the lock is not released until the end of the transaction (A commit point).

La figura 13.8 paragona i diversi livelli di isolamento per un'acquisizione dei dati. Nella figura vediamo che il livello di isolamento "lettura non portata a termine" (UR) non prende alcun lock. Il livello di isolamento "stabilità del cursore" (CS) prende un lock per la riga 1 quando sta per acquisirla, ma la libera non appena va a prendere la riga 2 e così via. Per il livello di isolamento "stabilità di lettura" (RS) o "lettura ripetibile" (RR), qualunque riga sia acquisita sarà bloccata e non verrà rilasciata fino alla conclusione della transazione (al punto del commit).

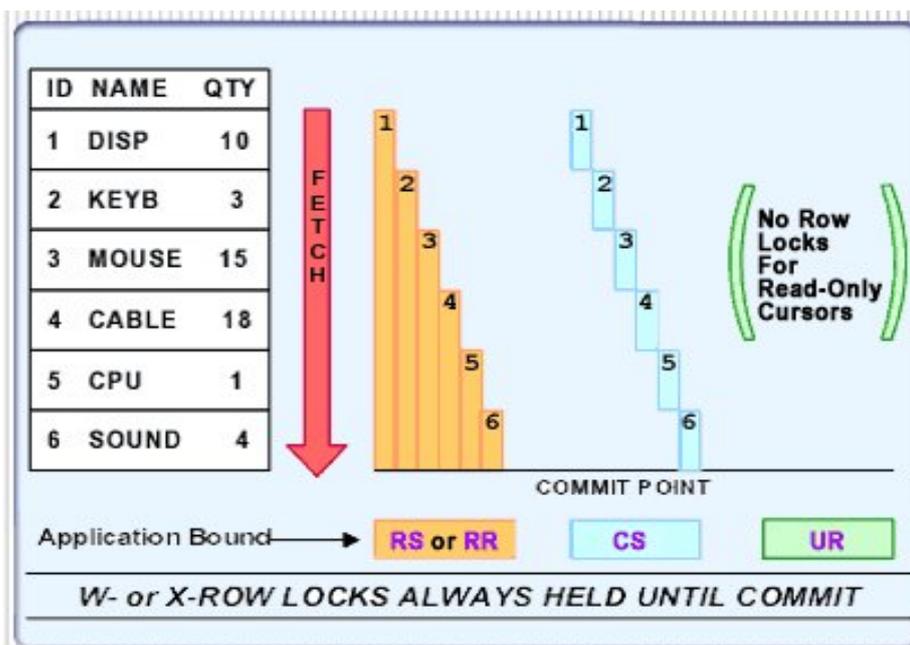


Figure 13.8 – Comparing isolation levels for a fetch / Figura 13.8 - Confronto tra i livelli di isolamento per un'acquisizione dei dati

13.4.6 Setting the isolation level / Settare il livello di isolamento

Isolation levels can be specified at many levels:

- ▶ Session (application)
- ▶ Connection
- ▶ Statement

The isolation level is normally defined at the session or *Application* Level. If no isolation level is specified in your application, it defaults to cursor stability. For example, table 13.1 shows the possible isolation levels for a .NET or JDBC program and how these properties, when set, match a DB2 isolation level.

I livelli di isolamento possono essere specificati a più livelli:

- ▶ Sessione (dell'applicazione)
- ▶ Collegamento
- ▶ Statement

Il livello di isolamento è definito normalmente al livello di sessione o di *Applicazione*. Se nessun livello di isolamento è specificato nella tua applicazione, quello di default è "stabilità del cursore". Per esempio, la tabella 13.1 mostra i livelli possibili di isolamento per un programma JDBC o .NET e come queste proprietà, una volta fissate, abbinino un livello di isolamento DB2.

| DB2 | .NET | JDBC |
|-----------------------|-----------------|------------------------------|
| Uncommitted Read (UR) | ReadUncommitted | TRANSACTION_READ_UNCOMMITTED |
| Cursor Stability (CS) | ReadCommitted | TRANSACTION_READ_COMMITTED |
| Read Stability (RS) | RepeatableRead | TRANSACTION_REPEATABLE_READ |
| Repeatable Read (RR) | Serializable | TRANSACTION_SERIALIZABLE |

Table 13.1 - Comparison of isolation level terminology / Tabella 13.1 - Confronto di terminologia del livello di isolamento

Statement isolation level can be set using the WITH {isolation level} clause. For example:

```
SELECT ... WITH {UR | CS | RS | RR}
```

Example scenario:

An application needs to get a "rough" count of how many rows are in a table. Performance is of utmost importance. Cursor stability isolation level is required with the exception of one SQL

Il livello di isolamento degli statement può essere fissato usando la clausola WITH {isolamento livello}. Per esempio:

Scenario d'esempio:

Un'applicazione deve ottenere un calcolo "approssimativo" di quante righe ci sono in una tabella. La prestazione è di massima importanza. Il livello di isolamento di stabilità del cursore è

statement:

```
SELECT COUNT(*) FROM tabl WITH UR
```

richiesto con l'eccezione di uno statement SQL:

For embedded SQL, the level is set at bind time, for dynamic SQL, the level is set at run time.

Per SQL embedded, il livello è fissato al momento del binding, per SQL dinamico, il livello è fissato a run-time.

Choosing which isolation level to use depends on your application. If your application does not need exact counts as in the above example, choose UR isolation. If your application requires very tight control on the data it works with, choose RR isolation.

Scegliere quale livello di isolamento usare dipende dalla tua applicazione. Se la tua applicazione non ha bisogno di conteggi esatti come nell'esempio sopra, scegli l'isolamento UR. Se la tua applicazione richiede il controllo molto stretto sui dati con cui funziona, scegli l'isolamento RR.

13.5 Lock escalation

Every lock made by DB2 consumes some memory. When the optimizer thinks it is better to have one lock on the entire table, rather than multiple row locks, lock escalation occurs. Figure 13.9 illustrates this.

Ogni lock fatto da DB2 consuma un po' di memoria. Quando l'ottimizzatore pensa che sia meglio avere un lock sull'intera tabella, piuttosto che più lock di riga, il lock escalation avviene. La figura 13.9 illustra questo.

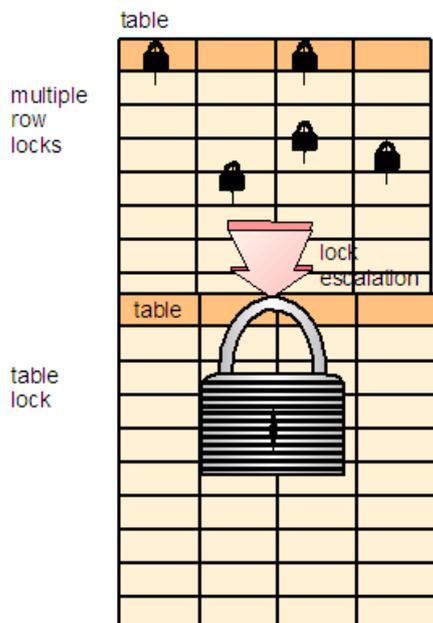


Figure 13.9 – Lock escalation / Figura 13.9 - Lock escalation

There are two main database configuration parameters related to lock escalation:

- ▶ **LOCKLIST** – The amount of memory (in 4k pages) reserved to manage locks for all connected applications. The default is fifty 4K (200 K)pages on Windows
- ▶ **MAXLOCKS** –Maximum percentage of the entire lock list that a single application can use up. The default is 22%.

Therefore, if the default values are used, lock escalation occurs when a single application requires more than 44K of lock memory ($200\text{ K} * 22\% = 44\text{K}$). If lock escalation occurs frequently with these settings, increase the value of LOCKLIST and MAXLOCKS. Lock escalation is not good for performance as it reduces concurrency. The DB2 diagnostic log file (db2diag.log, which is normally located in the C:\Program Files\IBM\SQLLIB\DB2 directory) can be used to determine whether lock escalation is occurring.

Ci sono due parametri principali di configurazione del database relativi al lock escalation:

- ▶ **LOCKLIST** – La quantità di memoria (in pagine di 4k) riservata per gestire i lock per tutte le applicazioni collegate. Questo di default è cinquanta pagine di 4K (200 K) su Windows
- ▶ **MAXLOCKS** – Percentuale massima dell'intera lista di lock che una singola applicazione può consumare. Questo di default è 22%.

Di conseguenza, se i valori di default sono usati, la lock escalation avviene quando una singola applicazione richiede più di 44K della memoria dei lock ($200\text{ K} * 22\% = 44\text{K}$). Se la lock escalation avviene frequentemente con queste impostazioni, puoi aumentare il valore di LOCKLIST e di MAXLOCKS. la lock escalation non va bene in termini di prestazioni poiché riduce la concorrenza. Il file di log diagnostico DB2 (db2diag.log, che è situato normalmente nel directory C:\Program Files\IBM\SQLLIB\DB2) può essere usato per determinare se la lock escalation sta avvenendo.

13.6 Lock monitoring / Monitoraggio del locking

You can monitor the use of locks using DB2 application lock snapshots. To turn on the snapshots for locks, issue this command:

```
UPDATE MONITOR SWITCHES USING LOCK ON
```

Puoi monitorare l'uso dei lock usando i lock snapshot della applicazione DB2. Per accendere ai lock snapshot, esegui questo comando:

After the switch is turned on, monitoring information will be collected. To obtain a report of the locks at a given time, issue this command:

Dopo che l'interruttore viene acceso, le informazioni per il monitoraggio saranno raccolte. Per ottenere un report dei lock da un certo momento, esegui questo comando:

```
GET SNAPSHOT FOR LOCKS FOR APPLICATION AGENTID <handle>
```

Figure 13.9 shows the output for a sample application lock snapshot.

La figura 13.9 mostra un esempio dell'output per un lock snapshot dell'applicazione.

```

Application Lock Snapshot

Snapshot timestamp                = 11-05-2002
00:09:08.672586

Application handle                = 9
Application ID                    =
*LOCAL.DB2.00B9C5050843
Sequence number                  = 0001
Application name                  = db2bp.exe
Authorization ID                  = ADMINISTRATOR
Application status                = UOW Waiting
Status change time               = Not Collected
Application code page             = 1252
Locks held                       = 4
Total wait time (ms)             = 0

List Of Locks
Lock Name                        = 0x05000700048001000000000052
Lock Attributes                   = 0x00000000
Release Flags                     = 0x40000000
Lock Count                       = 255
Hold Count                       = 0
Lock Object Name                  = 98308
Object Type                       = Row
Tablespace Name                   = TEST4K
Table Schema                      = ADMINISTRATOR
Table Name                       = T2
Mode                             = X

```

Figure 13.9 – Application Lock Snapshot / Figura 13.9 – Lock Snapshot della Applicazione

13.7 Lock wait

When two or more applications need to perform an operation on the same object, one of them may have to wait to obtain the needed lock. By default, an application will wait indefinitely. The time an application waits for a lock is controlled by the database configuration parameter LOCKTIMEOUT. The default value of this parameter is -1 (infinite wait).

The CURRENT LOCK TIMEOUT register can be used to set the lock wait for a given connection. By default, this register is set to the value of LOCKTIMEOUT. Use the SET LOCK TIMEOUT statement to change its value. Once the value of this register is set for a connection, it will persist across transactions. Example:

```
SET LOCK TIMEOUT=WAIT n
```

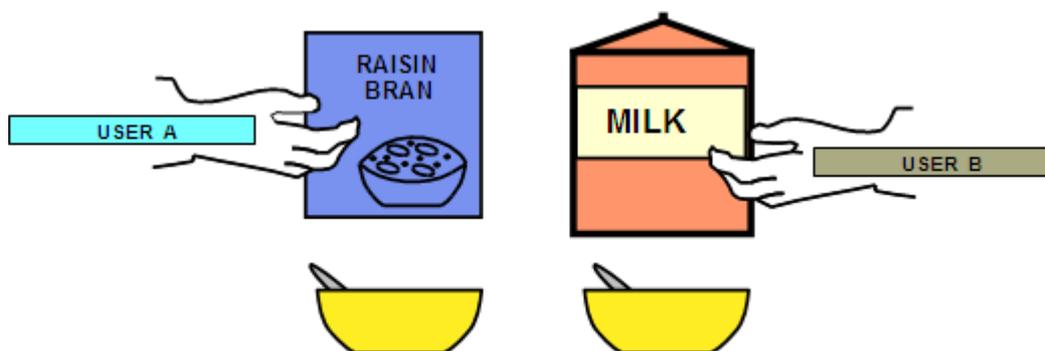
Quando due o più applicazioni devono eseguire un'operazione sullo stesso oggetto, una di loro può dover attendere per ottenere il lock necessario. Di default, un'applicazione attenderà in modo indefinito. Il tempo per cui un'applicazione attende un lock è controllato dal parametro di configurazione del database LOCKTIMEOUT. Il valore di default di questo parametro è -1 (attesa infinita).

Il registro CURRENT LOCK TIMEOUT può essere usato per fissare il lock wait per un certo collegamento. Di default, questo registro è fissato al valore LOCKTIMEOUT. Usa lo statement SET LOCK TIMEOUT per cambiare il valore. Una volta che il valore di questo registro è fissato per un collegamento, persisterà attraverso le transazioni. Esempio:

13.8 Deadlock causes and detection / Cause e rilevazione del deadlock

A deadlock occurs when two or more applications connected to the same database wait indefinitely for a resource. The waiting is never resolved because each application is holding a resource that the other needs. Deadlocks are an application design issue most of the time. Figure 13.10 illustrates a deadlock.

Un deadlock accade quando due o più applicazioni collegate allo stesso database attendono in modo indefinito per una risorsa. L'attesa non è mai risolta perché ogni applicazione sta tenendo una risorsa di cui l'altra ha bisogno. Spesso, i deadlock sono questioni da risolvere durante il design dell'applicazione. La figura 13.10 illustra un deadlock.



within a UNIT OF WORK (UOW) INSERT CEREAL AND MILK into BOWL
Figure 13.10 – Deadlock scenario / Figura 13.10 – Scenario di deadlock

In Figure 13.10, user A is holding the raisin bran and will not let go until he gets the milk. On the other hand, user B is holding the milk, and will not let go until he gets the raisin bran. Therefore, we have a deadlock situation.

Nella figura 13.10, l'utente A sta tenendo la crusca di uva passa e non la lascerà finché non otterrà il latte. Dall'altro lato, l'utente B sta tenendo il latte e non lo lascerà finché non otterrà la crusca di uva passa. Di conseguenza, abbiamo una situazione di deadlock.

To simulate a deadlock situation in DB2, follow these steps:

1. Open two DB2 Command Windows (which we will call "CLP1" and "CLP2". respectively) representing two different applications connecting to the database

Per simulare una situazione di deadlock in DB2, segui questi punti:

1. Apri due DB2 Command Window (che chiameremo "CLP1" e "CLP2". rispettivamente) che rappresenteranno le due applicazioni differenti che si collegano al database.

2. From CLP1 issue the commands:

```
db2 connect to sample
db2 +c update employee set firstnme = 'Mary' where empno
='000050'
```

First we are connecting to the SAMPLE database, and then issuing an update statement on the row with "empno = 50000" on the employee table. The "+c" option in the statement indicates that we do not want the DB2 Command Window to automatically commit the statement. We are doing this on

2. Da CLP1 esegui I comandi:

Primo ci colleghiamo al database SAMPLE e poi eseguiamo uno statement di aggiornamento sulla riga con "empno = 50000" sulla tabella employee. L'opzione "+c" nello statement indica che non vogliamo che la DB2 Command Window faccia un commit automaticamente. Facciamo questo

purpose so we hold the locks.

di proposito così manteniamo il lock.

3. From CLP2 issue the commands:

```
db2 connect to sample
db2 +c update employee set firstnme = 'Tom' where empno
='000030'
```

In the CLP2 window, which represents the second application, we are also connecting to the SAMPLE database, but are updating another row in the employee table.

3. Da CLP2 esegui i comandi:

Nella finestra CLP2, che rappresenta la seconda applicazione, ci stiamo collegando sempre al database SAMPLE, ma stiamo aggiornando un'altra riga nella tabella employee.

4. From CLP1 issue:

```
db2 +c select firstnme from employee where empno = '000030'
```

After pressing Enter to execute the above SELECT statement, the SELECT may seem to hang. It actually is not hanging, but waiting for the release of the exclusive lock that was taken by CLP2 on this row in step 3. At this point, if LOCKTIMEOUT has been left with its default value of -1, the CLP1 application would wait forever.

4. Da CLP1 esegui:

Dopo aver premuto Enter per eseguire lo statement SELECT di sopra, può sembrare che la SELECT si arresti. In realtà non si sta piantando, ma sta attendendo il rilascio del lock esclusivo che è stato preso da CLP2 su questa riga al punto 3. A questo punto, se LOCKTIMEOUT fosse stato lasciato con il suo valore di default -1, l'applicazione CLP1 attenderebbe per sempre.

5. From CLP2 issue:

```
db2 +c select firstnme from employee where empno = '000050'
```

By issuing the above SELECT statement, we are now creating a deadlock. This SELECT statement will also seem to hang, as it is waiting for the release of the exclusive lock that was taken by CLP1 on this row in step 2.

5. Da CLP2 esegui:

Eseguendo lo statement SELECT di sopra, stiamo creando un deadlock. Anche questo statement SELECT sembrerà piantarsi, siccome sta attendendo il rilascio del lock esclusivo che è stato preso da CLP1 su questa riga al punto 2.

In the above deadlock scenario, DB2 will check for the database configuration parameter DLCHKTIME. This parameter sets the time interval for checking for deadlocks. For example, if

Nello scenario di deadlock precedente, DB2 controllerà se c'è il parametro di configurazione del database DLCHKTIME. Questo parametro fissa l'intervallo di tempo per controllare i

this parameter is set to 10 seconds, DB2 will check every 10 seconds if a deadlock has occurred. If indeed a deadlock happened, DB2 will use an internal algorithm to determine which of the two transactions should be rolled back, and which one should continue.

If you are experiencing numerous deadlocks, you should re-examine your existing transactions and see if any restructuring is possible.

deadlock. Per esempio, se questo parametro è fissato a 10 secondi, DB2 controllerà ogni 10 secondi se è avvenuto un deadlock. Se avviene effettivamente un deadlock, DB2 userà una procedura interna per determinare su quale delle due transazioni deve essere fatto un roll back e quale deve continuare.

Se stai incontrando numerosi deadlock, dovresti riesaminare le tue transazioni esistenti e vedere se è possibile fra qualche cambiamento.

13.9 Concurrency and locking best practices / Scelte ottimali per concorrenza e locking

The following are some tips to follow in order to allow for the best possible concurrency:

1. Keep transactions as short as possible. This can be achieved by issuing frequent COMMIT statements (even for read-only transactions) when your application logic allows it.
2. Log transaction information only when required.
3. Purge data quickly using:

```
ALTER TABLE ACTIVATE NOT LOGGED INITIALLY WITH EMPTY TABLE
```

4. Perform data modifications in batches/groups. For example:

```
DELETE FROM (
  SELECT * FROM tedwas.t1 WHERE c1 = ... FETCH FIRST 3000 ROWS
  ONLY)
```

5. Use concurrency features in DB2 data movement tools.
6. Set the database level LOCKTIMEOUT parameter (suggested times are between 30-120 seconds). Don't leave it to the default of -1. You can also use session-based lock timeout.
7. Do not retrieve more data than is required. For example, use the FETCH FIRST n ROWS ONLY clause in SELECT statements.

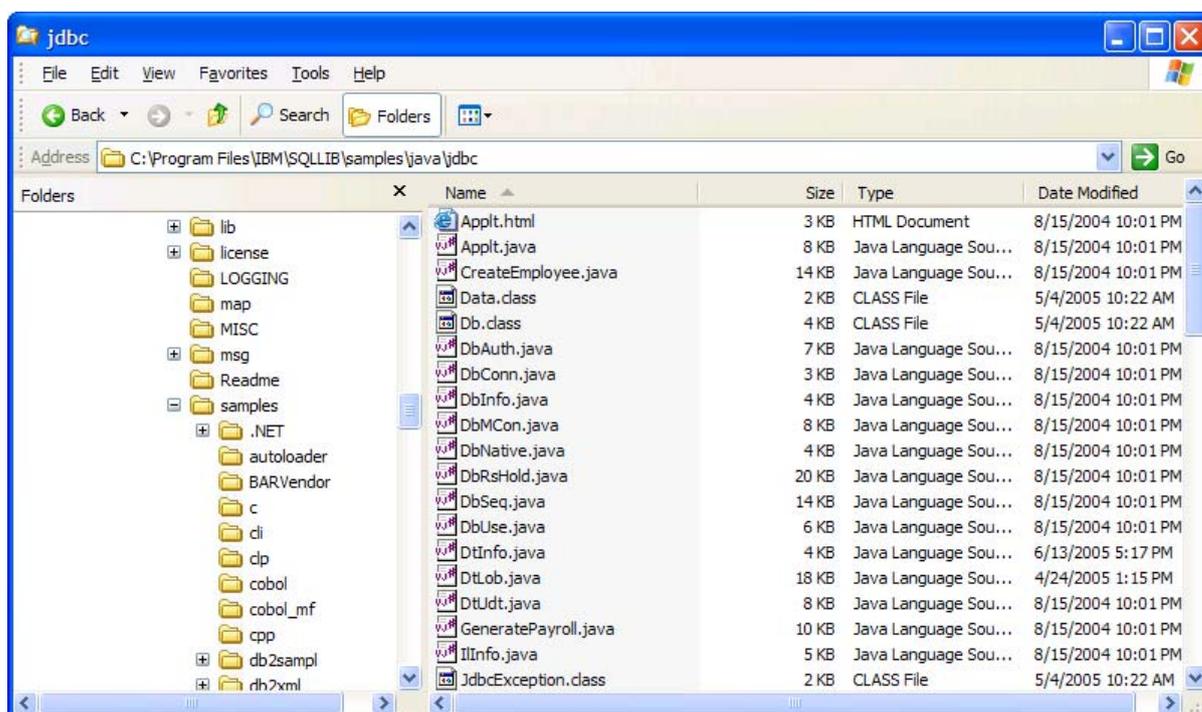
Qui di seguito ci sono alcuni punti da seguire per permettere la migliore concorrenza:

1. Fare transazioni più brevi possibile. Ciò può essere realizzato eseguendo di frequente lo statement COMMIT (anche per le transazioni read-only) quando la tua logica di applicazione lo permette.
 2. Salvare nel Log le informazioni di transazione soltanto quando richiesto.
 3. Eliminare i dati rapidamente usando:
- ```
ALTER TABLE ACTIVATE NOT LOGGED INITIALLY WITH EMPTY TABLE
```
4. Eseguire le modifiche di dati nei batch/gruppi. Per esempio:
- ```
DELETE FROM (
  SELECT * FROM tedwas.t1 WHERE c1 = ... FETCH FIRST 3000 ROWS
  ONLY)
```
5. Usare le feature di concorrenza nei tool di movimento di dati di DB2.
 6. Fissare il parametro di livello del database LOCKTIMEOUT (tempo suggerito fra 30-120 secondi). Non lasciarlo al valore di default -1. Puoi anche usare il lock session-based time-out.
 7. Non richiamare più dati di quanti siano necessari. Per esempio, usa la clausola FETCH FIRST n ROWS ONLY negli statement SELECT.

**PART III – LEARNING DB2:
APPLICATION DEVELOPMENT /
PARTE III – IMPARARE DB2:
SVILUPPO DI APPLICAZIONI**

In part III of this book, we discuss in depth application database objects such as: stored procedures, user-defined functions (UDFs), and triggers. Note that you can learn how to program in different languages using DB2 as the data server by reviewing the sample applications that come with the DB2 server installation in the `SQLLIB\samples` directory. The figure below shows the sample Java programs provided with DB2 on a Windows platform.

Nella parte III di questo libro, discuteremo in modo approfondito database object di applicazioni come: stored procedure, funzioni definite dall'utente (UDFs) e trigger. Tieni conto che puoi imparare a programmare con diversi linguaggi usando DB2 come data server esaminando le applicazioni di esempio che trovi con l'installazione del server DB2 nella directory `SQLLIB\samples`. La figura qui sotto mostra i programmi Java di esempio forniti con DB2 su una piattaforma di Windows.



Sample Java programs that come with DB2 / Programmi Java di esempio che trovi con DB2

14

Chapter 14 – SQL PL Stored Procedures / Capitolo 14 - Stored Procedure di SQL PL

In this chapter we discuss stored procedures. A stored procedure is a database application object that can encapsulate SQL statements and business logic. Keeping part of the application logic in the database provides performance improvements as the amount of network traffic between the application and the database is considerably reduced. In addition, stored procedures provide a centralized location to store your code, so other applications can reuse the same procedures.

DB2 stored procedures can be written using SQL PL, C/C++, Java, Cobol, CLR (Common Language Runtime) supported languages, and OLE. In this chapter, we focus on SQL PL procedures because of their popularity and simplicity.

Figure 14.1 illustrates how stored procedures work.

In questo capitolo discuteremo le stored procedure. Una stored procedure è un oggetto delle applicazioni database che può incapsulare dichiarazioni SQL e business logical. Mantenendo parte della logica dell'applicazione nel database permette un miglioramento delle prestazioni poiché la quantità di traffico di rete fra l'applicazione ed il database è nettamente ridotta. In più, le stored procedure offrono un luogo centralizzato per memorizzare il tuo codice, in modo che altre applicazioni possano riusare le stesse procedure.

Le stored procedure di DB2 possono essere scritte usando SQL PL, C/C++, Java, COBOL, i linguaggi supportati di CLR (Common Language Runtime), e OLE. In questo capitolo, ci concentreremo sulle procedure di SQL PL data la loro popolarità e semplicità.

La figura 14.1 illustra come funzionino le stored procedure.

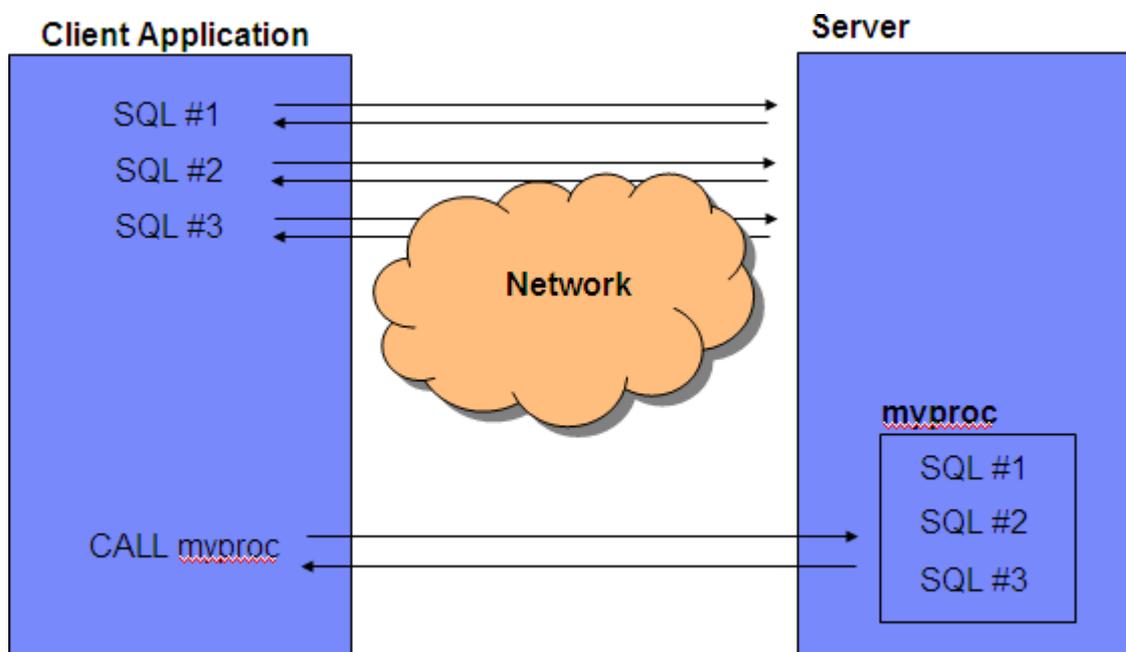


Figure 14.1 – Network traffic reduction with stored procedures / Figura 14.1 - Riduzione del traffico di rete con le stored procedure

At the top left corner of the figure, you see several SQL statements executed one after the other. Each SQL is sent from the client to the server, and the server returns the result back to the client. If many SQL statements are executed like this, network traffic increases.

On the other hand, at the bottom right corner, you can see the stored procedure “myproc” held on the server, which contains the same SQL; and then at the client (on the left side), the CALL statement is used to call the procedure. This second method of calling the procedure is more efficient, as there is only one call statement that goes through the network, and one result returned to the client.

Stored procedures can also be helpful for security purposes in your database. For example, you can let users access tables or views only through stored

In alto a sinistra nella figura, si vedono parecchi statement SQL eseguiti uno dopo l'altro. Ogni statement SQL è trasmesso dal client al server poi il server ritorna il risultato indietro al client. Se molti statement SQL sono eseguiti come questo, il traffico di rete aumenta.

D'altra parte, in basso a destra, si vede la stored procedure “myproc” tenuta sul server, che contiene lo stesso SQL; e poi sul client (dalla parte di sinistra), la dichiarazione CALL è usata per chiamare la procedura. Questo secondo metodo di chiamare la procedura è più efficiente, poiché c'è soltanto una dichiarazione di chiamata che passa attraverso la rete ed un solo risultato ritornato al client.

Le stored procedure possono anche essere utili per scopi di sicurezza nel tuo database. Per esempio, puoi permettere agli utenti l'accesso alle

procedures; this helps lock down the server and keep users from accessing information they are not supposed to access. This is possible because users do not require explicit privileges on the tables or views they access through stored procedures; they just need to be granted sufficient privilege to invoke the stored procedures.

tabelle o alle view soltanto attraverso le stored procedure; ciò aiuta il “lock down” del server e evita che gli utenti abbiano accesso ad informazioni a cui non devono accedere. Ciò è possibile perché gli utenti non richiedono privilegi espliciti sulle tabelle o sulle view a cui accedono con le stored procedure; devono soltanto aver i privilegi sufficienti per invocare le stored procedure.

Note:

For more information about SQL PL stored procedures, watch this video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4343>

Nota:

Per ulteriori informazioni sulle stored procedure di SQL PL, guardi questo video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4343>

14.1 The IBM Data Studio / L'IBM Data Studio

IBM Data Studio is a comprehensive solution that helps you develop and manage database applications throughout the data management lifecycle. Some of the features that can be performed with Data Studio are:

- Create, alter, and drop DB2 database objects (with impact analysis)
- Explore and edit data - relational and XML
- Visually build SQL and XQuery statements
- Optimize queries using Visual Explain
- Develop, debug, and deploy stored procedures SQL and Java™
- Develop user defined functions (UDFs)
- Develop SQLJ applications
- Develop queries and routines for

IBM Data Studio è una soluzione esauriente che ti aiuta a sviluppare e gestire le applicazioni di database durante il ciclo di vita della gestione dei dati. Alcune delle feature che possono essere eseguite con IBM Data Studio sono:

- Creare, modificare e eliminare i database object di DB2 (con analisi di impatto)
- Esplorare e modificare i dati - relazionali e XML
- Costruire visivamente l'SQL e le dichiarazioni XQuery
- Ottimizzare le query che usano il Visual Explain
- Sviluppare, debuggare, e deployare le stored procedure SQL e Java™
- Sviluppare le funzioni user defined(UDFs)
- Sviluppare le applicazioni SQLJ
- Sviluppare le query e le routine per

- | | |
|---|--|
| <ul style="list-style-type: none"> - pureXML applications - Perform data movement tasks - Collaborate and share projects with team members - Quickly build SOAP and REST Web Services - Discover database object relationships with physical data models (diagraming) - Visualize data distribution across tables | <ul style="list-style-type: none"> - le applicazioni pureXML - Eseguire task di movimento di dati - Collaborare e condividere i progetti con i membri del team - Sviluppare rapidamente web service SOAP e di REST - Scoprire le relazioni tra i database object e i modelli di dati fisici (diagrammi) - Visualizzare la distribuzione dei dati attraverso le tabelle |
|---|--|

IBM Data Studio is based on the Eclipse platform. It is a separate image (i.e. not part of the DB2 installation image), but it is also free of charge. Data Studio images can be downloaded from the “Download” tab on the ibm.com/db2/express web site. Figure 14.2 shows IBM Data Studio.

Il IBM Data Studio è basato sulla piattaforma di Eclipse. È un'immagine separata (cioè non è parte dell'immagine dell'installazione di DB2), ma è inoltre gratuita. Le immagini del Data Studio possono essere download dalla scheda “Download” sul sito web di ibm.com/db2/express. Figura 14.2 mostra il IBM Data Studio.

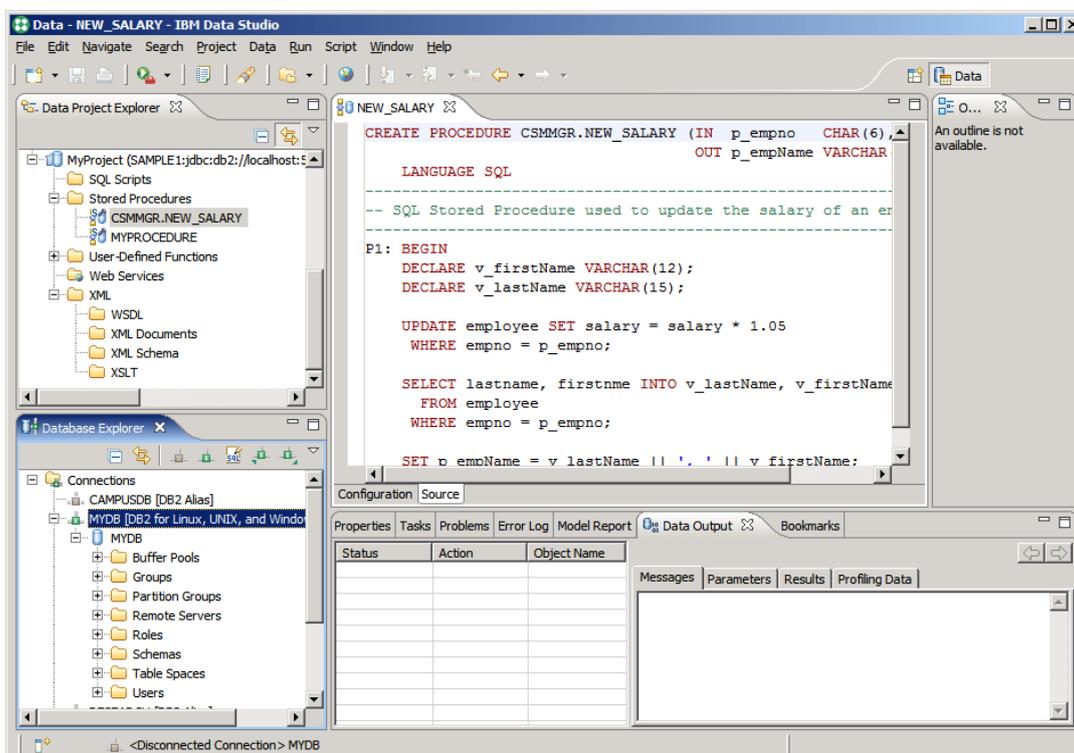


Figure 14.2 – IBM Data Studio / Figura 14.2 – IBM Data Studio

14.1.2 Create a stored procedure in Data Studio / Creare una stored procedure nel Data Studio

To create a Java or SQL PL stored procedure in Data Studio, follow the following steps. Note that stored procedures in other languages cannot be created from Data Studio.

Per creare una stored procedure Java o SQL PL nel Data Studio, segui i seguenti punti. Nota che le stored procedure in altri linguaggi non possono essere creati dal Data Studio.

Step 1: Create a Data Studio project

Punto 1: Creare un progetto Data Studio

From the Data Studio menu, choose *File -> New -> Project* and choose *Data Development Project*. This is shown in Figure 14.3.

Dal menu di Data Studio, scegli *File -> New -> Project* e scegli *Data Development Project*. Ciò è indicato nella figura 14.3.

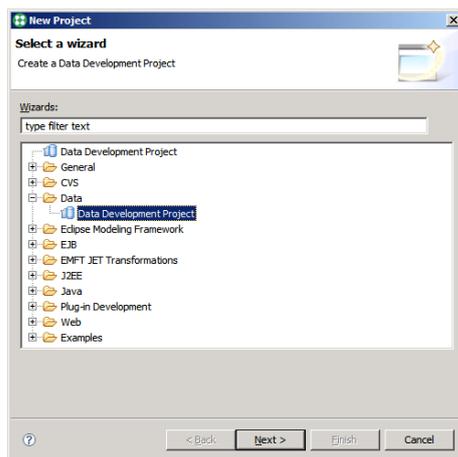


Figure 14.3 – The data development project / Figura 14.3 - Il progetto di sviluppo di dati

Follow the steps from the wizard to input a name for your project, indicate which database you want to connect to as part of your project, and specify the JDK directory (the one provided by default is usually the correct one).

Segui i punti dal wizard per immettere un nome per il tuo progetto, indicare a quale database vuoi collegarti come parte del tuo progetto, e specificare la directory della JDK (quella fornita di default è solitamente quella corretta).

Step 2: Create a stored procedure

When the project is created, the left side of the data perspective will show your project. In Figure 14.4 you can see the “myProject” project created and expanded.

Punto 2: Creare una stored procedure

Quando il progetto è stato creato, la parte sinistra della prospettiva dati mostrerà il tuo progetto. Nella figura 14.4 puoi vedere il progetto “myProject” creato ed esteso.

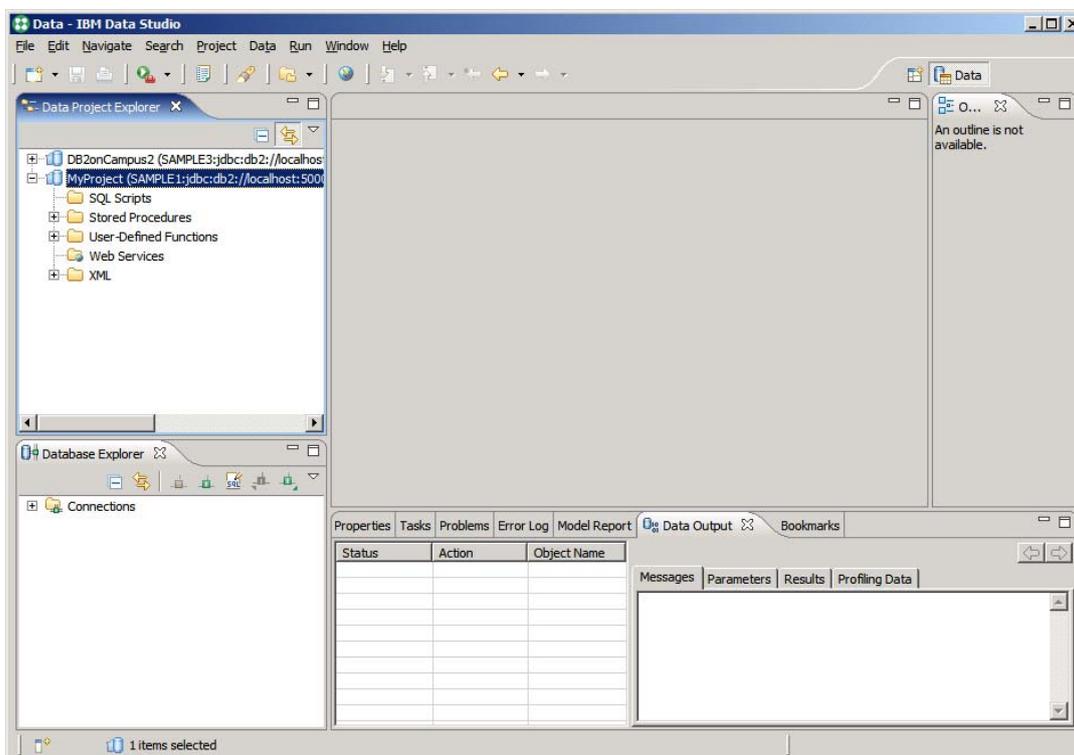


Figure 14.4 – The “myProject” project / Figura 14.4 – Il progetto “myProject”

Figure 14.4 shows different folders for your project. When you want to create a stored procedure, right-click on the Stored Procedures folder and choose *New -> Stored Procedure*. Complete the information requested by the New Stored Procedure wizard such as the project to associate the procedure with, the name and language of the procedure (Note that only SQL PL and Java are supported within IBM Data Studio), and the SQL statements to use in the procedure. By default, Data Studio gives you an example SQL

La figura 14.4 mostra diverse cartelle all'interno del tuo progetto. Quando vuoi creare una stored procedure, fai clic destro sulla cartella Stored Procedures e scegli *New -> Stored Procedure*. Completa le informazioni richieste dal New Stored Procedure wizard come il nome e il linguaggio della procedura (Nota che soltanto SQL PL e Java sono supportati all'interno dell'IBM Data Studio), e gli statement SQL da usare nella procedura. Di default, Data Studio ti dà un esempio di uno statement SQL. A questo punto, puoi cliccare su *Finish*

statement. At this point, you can click *Finish* and a stored procedure is created using some template code and the SQL statement provided before as an example. This is shown in Figure 14.5.

e una stored procedure sarà creata usando un certo codice di template e lo statement SQL fornito prima come esempio. Ciò è indicato nella figura 14.5.

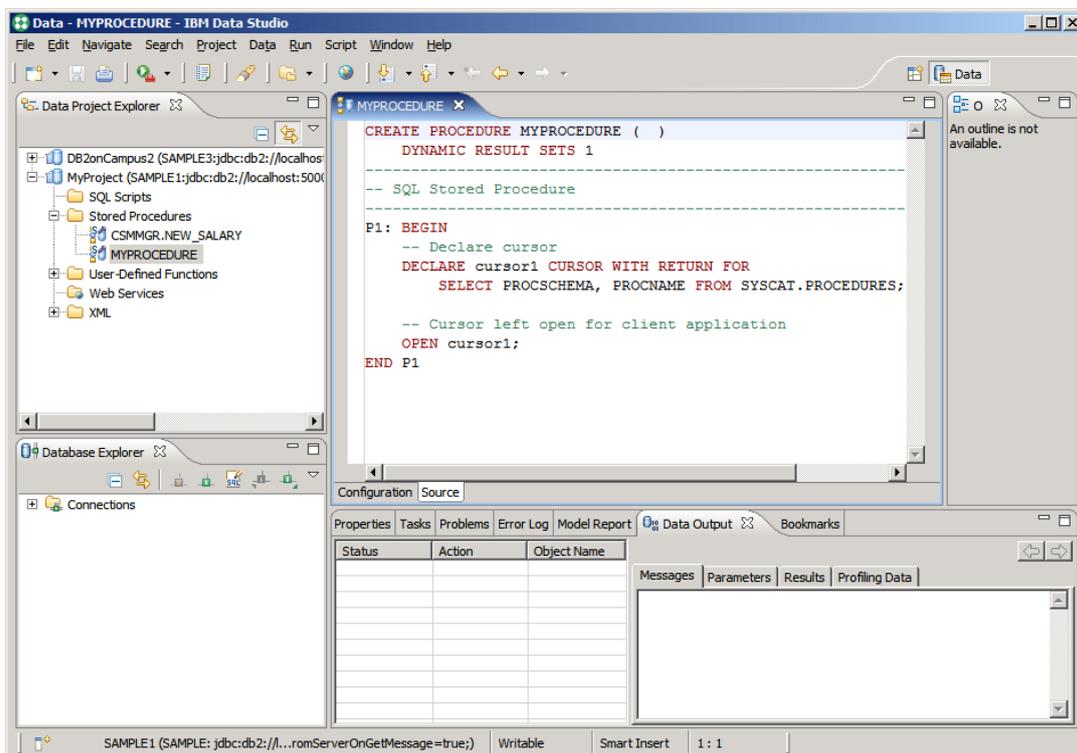


Figure 14.5 – A sample stored procedure / Figura 14.5 – Un esempio di stored procedure

In Figure 14.5, the code for the sample stored procedure “MYPROCEDURE” was generated. You can replace all of this code with your own code. For simplicity, we will continue in this book using the above sample stored procedure as if we had written it.

Nella figura 14.5, il codice per la stored procedure di esempio “MYPROCEDURE” è stato generato. Puoi sostituire tutto questo codice con il tuo codice. Per semplicità, continueremo in questo libro seguendo l’esempio sopra di stored procedure come se l’avessimo scritta noi.

Step 3: Compile (deploy) a stored procedure

Once the stored procedure is created, you compile and deploy it by right-

Punto 3: Compilare (deploy) una stored procedure

Una volta che la stored procedure è creata, puoi compilarla e deployarla

clicking on it in the left panel, and choosing “Deploy”. Figure 14.6 illustrates this step.

facendo un clic destro su di essa nel pannello di sinistra e scegliendo “Deploy”. La figura 14.6 illustra questo punto.

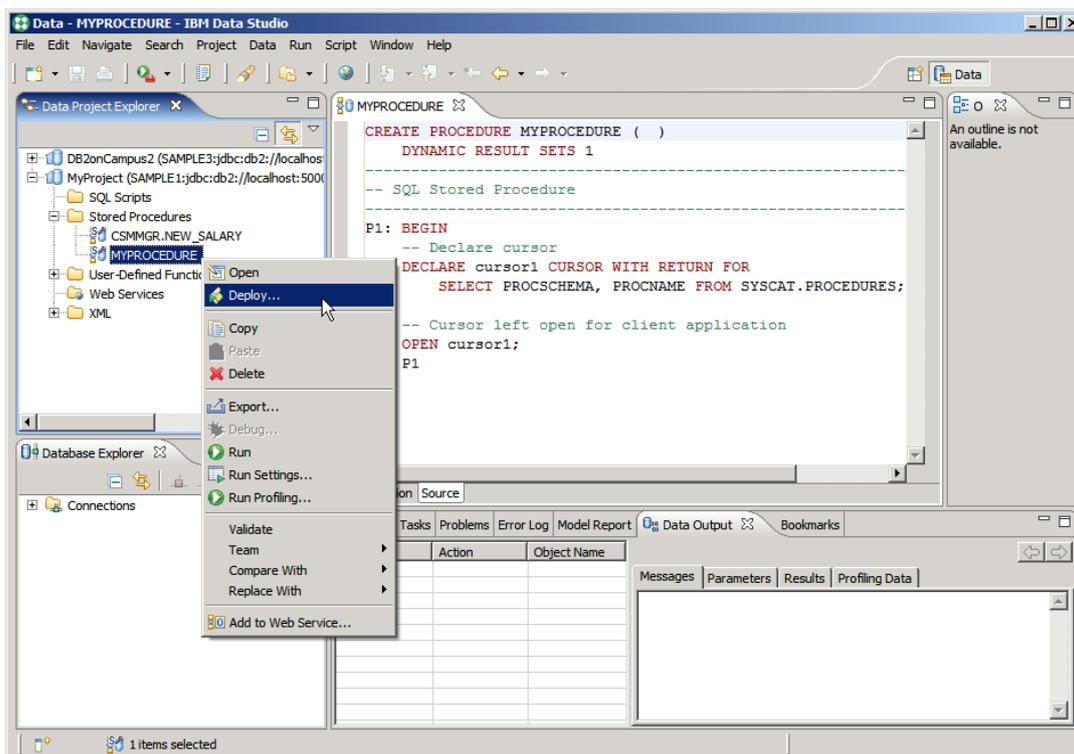


Figure 14.6 – Deploying a stored procedure / Figura 14.6 – Deploy di una stored procedure

Step 4: Run a stored procedure

Once the stored procedure has been deployed, you can run it by right-clicking on it and choosing *Run*. The results would appear in the *Results* tab at the bottom right corner of the window.

To run a stored procedure from the Command Window or the Command Editor, you can use `CALL <procedure name>`

Punto 4: Eseguire una stored procedure

Una volta che la stored procedure è stata deployata, puoi eseguirla facendo clic destro su di essa e scegliendo *Run*. I risultati dovrebbero comparire nel tab *Results* nell'angolo in basso a destra della finestra.

Per eseguire una stored procedure dalla Command Window o dal Command Editor, puoi usare `CALL <procedure name>`

14.2 SQL PL stored procedures basics / Principi fondamentali delle stored procedure di SQL PL

SQL Procedural Language (SQL PL) stored procedures are easy to create and learn. They have the best performance in DB2. SQL PL stored procedures (or simply “SQL stored procedures”) are the focus of this chapter.

Le stored procedure del SQL Procedural Language (SQL PL) sono facili da creare e imparare. Hanno la migliore prestazione in DB2. Le stored procedure di SQL PL (o semplicemente “SQL stored procedure”) sono al centro di questo capitolo.

14.2.1 Stored procedure structure / Struttura delle stored procedure

The basic store procedure syntax is shown here.

La sintassi di base delle stored procedure è mostrata qui.

```
CREATE PROCEDURE proc_name [( {optional parameters} )]  
[optional procedure attributes] <statement>
```

Where <statement> is a single statement, or a set of statements grouped by BEGIN [ATOMIC] ... END

Dove lo <statement> è una singola dichiarazione, o un insieme di dichiarazioni raggruppate da BEGIN [ATOMIC] ... END

14.2.2 Optional stored procedure attributes / Attributi facoltativi delle stored procedure

The following describes some of the optional stored procedure attributes:

Ciò che segue descrive alcuni degli attributi facoltativi delle stored procedure:

- LANGUAGE SQL
This attribute indicates the language that the stored procedure will use. LANGUAGE SQL is the default value. For other languages, such as Java or C use LANGUAGE JAVA or LANGUAGE C, respectively.
- RESULT SETS <n>
This is required if your stored procedure will be returning n result sets.
- SPECIFIC my_unique_name
This is a unique name that can be
- LANGUAGE SQL
Questo attributo indica il linguaggio che la stored procedure userà. LANGUAGE SQL è il valore di default. Per altri linguaggi, come Java o C, usa rispettivamente LANGUAGE JAVA o LANGUAGE C.
- RESULT SETS <n>
Questo è richiesto se la tua stored procedure ritornerà n insiemi di risultati.
- SPECIFIC my_unique_name
Questo è un nome unico che può

given to a procedure. A stored procedure can be overloaded, that is, several stored procedures can have the same name, but with different number of parameters. By using the SPECIFIC keyword you can provide one unique name for each of these stored procedures, and this can ease management of stored procedures. For example, to drop a stored procedure using the SPECIFIC keyword, you can issue this statement: `DROP SPECIFIC PROCEDURE`. If the SPECIFIC keyword had not been used you would have had to use a `DROP PROCEDURE` statement and put the name of the procedure with the parameters so DB2 would know which of the overloaded procedures you wanted to drop.

essere dato ad una procedura. Una stored procedure può essere overloaded, cioè, parecchie stored procedure possono avere lo stesso nome, ma con un diverso numero di parametri. Usando la parola chiave SPECIFIC puoi fornire un nome unico per ciascuna di queste stored procedure, e questo può facilitare l'amministrazione delle stored procedure. Per esempio, per eliminare una stored procedure usando la parola chiave SPECIFIC, puoi fornire questa dichiarazione: `DROP SPECIFIC PROCEDURE`. Se la parola chiave SPECIFIC non è stata usata avresti dovuto usare una dichiarazione `DROP PROCEDURE` e mettere il nome della procedura con i parametri in modo che DB2 conosca quale delle procedure overloaded desideri eliminare.

14.2.3 Parameters / Parametri

There are three types of parameters in an SQL PL stored procedure:

- IN - Input parameter
- OUT - Output parameter
- INOUT - Input and Output parameter

For example:

```
CREATE PROCEDURE proc (IN p1 INT, OUT p2 INT, INOUT p3 INT)
```

When calling the procedure, all the parameters must be provided in the CALL statement. For example, to call the above stored procedure you would specify:

```
CALL proc (10,?,4)
```

The question mark (?) is used for OUT parameters in the CALL statement.

Ci sono tre tipi di parametri in una stored procedure di SQL PL:

- IN - parametro di input
- OUT - parametro di output
- INOUT - Parametro di input e di output

Per esempio:

Quando chiami la stored procedure, tutti i parametri devono essere forniti nella dichiarazione CALL. Per esempio, per chiamare la stored procedure precedente specificheresti:

Il punto interrogativo (?) è usato per i parametri OUT nella dichiarazione di CALL.

Here is another example of a stored procedure with parameters that you can try:

```
CREATE PROCEDURE P2 ( IN    v_p1 INT,
                    INOUT v_p2 INT,
                    OUT   v_p3 INT)

LANGUAGE SQL
SPECIFIC myP2
BEGIN
    -- my second SQL procedure
    SET v_p2 = v_p2 + v_p1;
    SET v_p3 = v_p1;
END
```

Qui un altro esempio di una stored procedure con i parametri che puoi provare:

To call the procedure from the Command Editor use:

```
call P2 (3, 4, ?)
```

Per chiamare la procedura dal Command Editor usa:

14.2.4 Comments in an SQL PL stored procedure / Commenti in una stored procedure di SQL PL

There are two ways to specify comments in an SQL PL stored procedure:

- Using two dashes. For example:
--This is an SQL-style comment
--Questo è un commento SQL-style
- Using a format similar to the C language. For example:
/* This is a C-style coment */
/* Questo è un commento C-style */

Ci sono due modi per specificare i commenti in una stored procedure SQL PL:

- Usando due trattini. Per esempio:
Usando un formato simile al linguaggio C. Per esempio:

14.2.5 Compound statements / Statement composti

A compound statement in a stored procedure is a statement consisting of several procedural instructions and SQL statements encapsulated by the keywords BEGIN and END. When the ATOMIC keyword follows the BEGIN keyword, the compound statement is treated as one unit, that is, all of the instructions or statements in the

Una dichiarazione di blocco in una stored procedure è una dichiarazione che consiste di parecchie istruzioni procedurali e di SQL statement incapsulate tra le parole chiavi BEGIN e END. Quando la parola chiave ATOMIC segue la parola chiave BEGIN, la dichiarazione di blocco è trattata come un'unità, cioè, tutte le istruzioni o gli

compound statement must be successful in order for the entire compound statement to be successful. If one of the statements is not, then everything is rolled back. Figure 14.7 shows a compound statement structure.

statement nella dichiarazione di blocco devono avere successo in cascata affinché l'intera dichiarazione di blocco termini con successo. Se uno degli statement non ha successo, allora viene fatto il rollback di tutto quanto. La figura 14.7 mostra una struttura di dichiarazione di blocco.

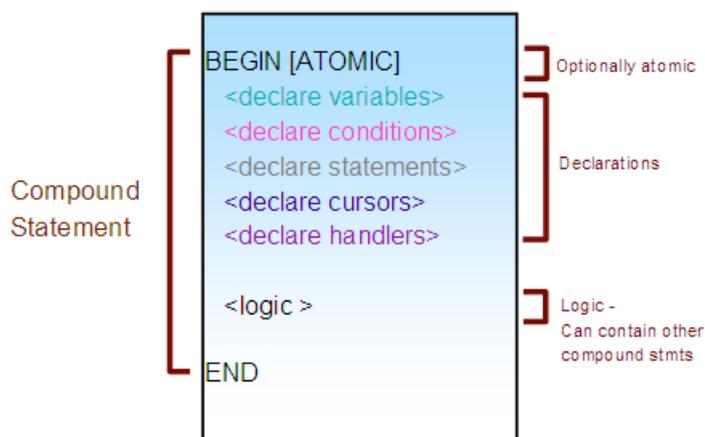


Figure 14.7 – Compound statements / Figura 14.7 - Dichiarazione di blocchi

14.2.6 Variable declaration / Dichiarazione dei variabile

To declare a variable, use the DECLARE statement:

Per dichiarare una variabile, usa la dichiarazione DECLARE:

```
DECLARE var_name <data type> [DEFAULT value];
```

Here are some examples:

Ecco qui alcuni esempi:

```
DECLARE temp1 SMALLINT DEFAULT 0;
DECLARE temp2 INTEGER DEFAULT 10;
DECLARE temp3 DECIMAL(10,2) DEFAULT 100.10;
DECLARE temp4 REAL DEFAULT 10.1;
DECLARE temp5 DOUBLE DEFAULT 10000.1001;
DECLARE temp6 BIGINT DEFAULT 10000;
DECLARE temp7 CHAR(10) DEFAULT 'yes';
DECLARE temp8 VARCHAR(10) DEFAULT 'hello';
DECLARE temp9 DATE DEFAULT '1998-12-25';
DECLARE temp10 TIME DEFAULT '1:50 PM';
DECLARE temp11 TIMESTAMP DEFAULT '2001-01-05-12.00.00';
DECLARE temp12 CLOB(2G);
DECLARE temp13 BLOB(2G);
```

14.2.7 Assignment statements / Statement di assegnazione

To assign a value to a variable, use the SET statement. For example:

```
SET total = 100;
```

The above statement is equivalent to:

```
VALUES(100) INTO total;
```

Additionally, any variable can be set to NULL:

```
SET total = NULL;
```

A condition is raised if more than one row fetches only the first row from a table:

```
SET total = (select sum(c1) from T1);  
SET first_val = (select c1 from T1 fetch first 1 row only)
```

You can also set variables according to external database properties:

```
SET sch = CURRENT SCHEMA;
```

Per assegnare un valore ad una variabile, usa la dichiarazione SET. Per esempio:

La dichiarazione precedente è equivalente a:

Inoltre, alcune variabile possono essere settate a NULL:

Una condizione è sollevata se più di una riga va a prendere soltanto la prima riga da una tabella:

Puoi anche settare le variabili secondo proprietà esterne del database:

14.3 Cursors / Cursori

A cursor is a result set holding the result of a SELECT statement. The syntax to declare, open, fetch, and close a cursor follows:

```
DECLARE <cursor name> CURSOR [WITH RETURN <return target>]  
    <SELECT statement>;  
OPEN <cursor name>;  
FETCH <cursor name> INTO <variables>;  
CLOSE <cursor name>;
```

Un cursore è un insieme di risultati che mantiene il risultato di uno statement SELECT. La sintassi per dichiarare, aprire, fare il fetch e chiudere un cursore è la seguente:

When a cursor is declared, the WITH RETURN clause can be used with these values:

- CLIENT: the result set will return to client application
- CALLER: the result set is returned

Quando un cursore viene dichiarato, la WITH RETURN clause può essere usata con questi valori:

- CLIENT: l'insieme dei risultati ritornerà all'applicazione del client
- CALLER: l'insieme dei risultati sarà

to client or stored procedure that made the call

ritornato al client o alla stored procedure che ha fatto la chiamata

Here is an example of a stored procedure using a cursor:

Qui troviamo un esempio di una stored procedure che usa un cursore:

```
CREATE PROCEDURE set ()
DYNAMIC RESULT SETS 1
LANGUAGE SQL
BEGIN
DECLARE cur CURSOR WITH RETURN TO CLIENT
FOR SELECT name, dept, job
FROM staff
WHERE salary > 20000;
OPEN cur;
END
```

14.4 Flow control / Controllo di flusso

Like in many other languages, SQL PL has several statements that can be used to control the flow of the logic. Below we list some of the flow control statements supported:

Come in molti altri linguaggi, SQL PL ha parecchie dichiarazioni che possono essere usate per controllare il flusso della logica. Sotto elenchiamo alcune dichiarazioni di controllo di flusso supportate:

CASE (selects an execution path (simple search)) / (seleziona un percorso di esecuzione (ricerca semplice))

IF

FOR (executes body for each row of table) / (esegue il body della stored procedure per ogni riga della tabella)

WHILE

ITERATE (forces next iteration. Similar to CONTINUE in C) / (forza il passaggio alla prossima iterazione. Simile a CONTINUE in C)

LEAVE (leaves a block or loop. "Structured Goto") / (lascia un blocco o un ciclo.

"Structured Goto")

LOOP (infinite loop) / (iterazione infinita)

REPEAT

GOTO

RETURN

CALL (procedure call) / (chiamata a procedura)

14.5 Calling stored procedures / Chiamare le stored procedure

The following code snippets show how to CALL stored procedures using different programming languages.

I seguenti frammenti di codice mostrano come chiamare (CALL) le stored procedure usando diversi linguaggi di programmazione.

Example calling a stored procedure from a CLI/ODBC application / Esempio di come chiamare una stored procedure da un'applicazione di CLI/ODBC

```
SQLCHAR *stmt = (SQLCHAR *)
"CALL MEDIAN_RESULT_SET( ? )" ;
SQLDOUBLE sal = 20000.0; /* Bound to parameter marker in
stmt */
SQLINTEGER salind = 0; /* Indicator variable for sal */

sqlrc = SQLPrepare(hstmt, stmt, SQL_NTS);
sqlrc = SQLBindParameter(hstmt, 1, SQL_PARAM_OUTPUT,
SQL_C_DOUBLE, SQL_DOUBLE, 0, 0, &sal, 0, &salind);
SQLExecute(hstmt);

if (salind == SQL_NULL_DATA)
    printf("Median Salary = NULL\n");
else
    printf("Median Salary = %.2f\n\n", sal );

/* Get first result set */
sqlrc = StmtResultPrint(hstmt);
/* Check for another result set */
sqlrc = SQLMoreResults(hstmt);
if (sqlrc == SQL_SUCCESS) {
    /* There is another result set */
    sqlrc = StmtResultPrint(hstmt);
}
```

For more details, see the DB2 sample file:

sqlllib/samples/sqlproc/resultset.c

Per maggiori dettagli, vedi il file DB2 sample:

Example calling a stored procedure from a VB.NET application / Esempio di come chiamare una stored procedure da un'applicazione VB.NET

```
Try
    \ Create a DB2Command to run the stored procedure
    Dim procName As String = "TRUNC_DEMO"
    Dim cmd As DB2Command = conn.CreateCommand()
    Dim parm As DB2Parameter
```

```
cmd.CommandType = CommandType.StoredProcedure
cmd.CommandText = procName

` Register the output parameters for the DB2Command
parm              = cmd.Parameters.Add("v_lastname",
DB2Type.VarChar)
parm.Direction   = ParameterDirection.Output
parm              = cmd.Parameters.Add("v_msg",
DB2Type.VarChar)
parm.Direction   = ParameterDirection.Output

` Call the stored procedure
Dim reader As DB2DataReader = cmd.ExecuteReader

Catch myException As DB2Exception
  DB2ExceptionHandler(myException)
Catch
  UnhandledExceptionHandler()
End Try
```

Example calling a stored procedure from a Java application / Esempio di come chiamare una stored procedure da un'applicazione Java

```
try
{
  // Connect to sample database
  String url = "jdbc:db2:sample";
  con = DriverManager.getConnection(url);

  CallableStatement cs = con.prepareCall("CALL
trunc_demo(?, ?)");

  // register the output parameters
  callStmt.registerOutParameter(1, Types.VARCHAR);
  callStmt.registerOutParameter(2, Types.VARCHAR);

  cs.execute();
  con.close();
}
catch (Exception e)
{
  /* exception handling logic goes here */
}
```

14.6 Errors and condition handlers / Handler per errori e condizioni

In DB2, the SQLCODE and SQLSTATE keywords are used to determine the successful or unsuccessful execution of an SQL statement. These keywords need to be explicitly declared in the outermost scope of the procedure as follows:

```
DECLARE SQLSTATE CHAR(5);  
DECLARE SQLCODE INT;
```

DB2 will set the values of the above keywords automatically after each SQL operation. For the SQLCODE, the values are set as follows:

- = 0, successful.
- > 0, successful with warning
- < 0, unsuccessful
- = 100, no data was found. (i.e.: FETCH statement returned no data)

For the SQLSTATE, the values are set as follows:

- success: SQLSTATE '00000'
- not found: SQLSTATE '02000'
- warning: SQLSTATE '01XXX'
- exception: all other values

The SQLCODE is RDBMS specific, and more detailed than the SQLSTATE. The SQLSTATE is standard among RDBMSs but is very general in nature. Several SQLCODEs may match one SQLSTATE. SQLCODEs and SQLSTATEs are discussed in more detail in Appendix A, Troubleshooting

A condition can be raised by any SQL statement and would match an SQLSTATE. For example, a specific condition like SQLSTATE '01004' is raised when a value is truncated during an SQL operation. Rather than using SQLSTATE '01004' to test for this condition, names can be assigned. In

In DB2, le parole chiave SQLSTATE e SQLCODE sono usate per determinare l'esecuzione con successo o meno di uno statement SQL. Queste keyword devono essere dichiarate esplicitamente nello scope più esterno della procedura come segue:

DB2 fisserà automaticamente i valori delle keyword precedenti dopo ogni operazione SQL. Per SQLCODE, i valori sono fissati come segue:

- = 0, successo.
- > 0, successo con avvertimento
- < 0, insuccesso
- = 100, nessun dato è stato trovato. (cioè: La dichiarazione FETCH non ha ritornato dati)

Per SQLSTATE, i valori sono fissati come segue:

- successo: SQLSTATE '00000'
- non trovato: SQLSTATE '02000'
- warning: SQLSTATE '01XXX'
- eccezione: tutti gli altri valori

SQLCODE è specifico di RDBMS, ed è più dettagliato di SQLSTATE. SQLSTATE è standard fra i RDBMS ma originariamente è molto generale. Parecchi SQLCODE possono corrispondere a un SQLSTATE. SQLCODE e SQLSTATE sono discussi più in dettaglio nel appendice A, Risoluzione dei problemi

Una condizione può essere sollevata da un qualsiasi statement SQL e corrisponderebbe a un SQLSTATE. Per esempio, una condizione specifica come SQLSTATE '01004' è sollevata quando un valore è troncato durante un'operazione SQL. Piuttosto che usare SQLSTATE '01004' per provare questa

this particular example, the name “trunc” can be assigned to condition SQLSTATE ‘01004’ as shown below:

condizione, possono essere assegnati dei nomi. In questo particolare esempio, il nome ‘trunc’ può essere assegnato alla condizione SQLSTATE ‘01004’ mostrata sotto.

```
DECLARE trunc CONDITION FOR SQLSTATE '01004'
```

Other predefined general conditions are:

- SQLWARNING
- SQLEXCEPTION
- NOT FOUND

Altre condizioni predefinite generali sono:

Condition handling / Trattamento delle condizioni

To handle a condition, you can create a condition handler which must specify:

- Which conditions it handles
- Where to resume execution (based on the type of the handler: CONTINUE, EXIT or UNDO)
- The actions to perform to handle the condition. The actions can be any statement, including control structures.

Per trattare una condizione, puoi creare un condition handler che deve specificare:

- Quale condizioni tratta
- Dove riprendere l'esecuzione (basato sul tipo di handler: CONTINUE, EXIT o UNDO)
- Le azioni da eseguire per trattare la condizione. Le azioni possono essere dichiarazioni qualsiasi, comprese strutture di controllo.

If an SQLEXCEPTION condition is raised, and there is no handler, the procedure terminates and returns to the client with an error.

Se una condizione SQLEXCEPTION è sollevata e non c'è un handler, la procedura termina e ritorna al client con un errore.

Types of handlers / Tipi di handler

There are three types of handlers:

CONTINUE – This handler is used to indicate that after an exception is raised, and the handler handles the condition, the flow will CONTINUE to the next statement after the statement that raised the condition.

Ci sono tre tipi di handler:

CONTINUE - questo handler è usato per indicare che quando un'eccezione sarà sollevata, e l'handler avrà trattato la condizione, il flusso continuerà (CONTINUE) dallo statement successivo a quello che ha sollevato la condizione.

EXIT – This handler is used to indicate that, after an exception is raised, and the handler handles the condition, the flow will go to the end of the procedure.

EXIT - questo handler è usato per indicare che, quando un'eccezione sarà sollevata, e l'handler avrà trattato la condizione, il flusso passerà direttamente alla fine della procedura.

UNDO – This handler is used to indicate that after an exception is raised, and the handler handles the condition, the flow will go to the end of the procedure, and will undo or roll back any statements performed.

UNDO - questo handler è usato per indicare che quando un'eccezione sarà sollevata, e l'handler avrà trattato la condizione, il flusso andrà alla fine della procedura, e annullerà o farà il rollback di tutti gli statement effettuati.

Figure 14.8 illustrates the different condition handlers and their behavior.

La figura 14.8 illustra i diversi handler di condizioni ed il loro comportamento.

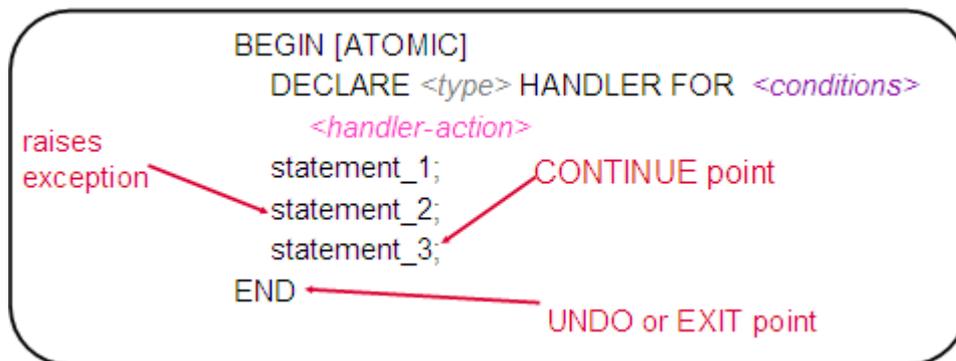


Figure 14.8 – Type of condition handlers / Figura 14.8 - Tipi di condition handler

14.7 Dynamic SQL / SQL dinamico

In dynamic SQL, as opposed to static SQL, the entire SQL statement is not known at run time. For example if `col1` and `tablename` are variables in this statement, then we are dealing with dynamic SQL:

Nell'SQL dinamico, in contrasto con l'SQL statico, l'intera dichiarazione SQL non è conosciuta a run time. Per esempio se `col1` e `tablename` sono variabili in questo statement, allora stiamo avendo a che fare con SQL dinamico:

```
'SELECT ' || col1 || ' FROM ' || tablename;
```

Dynamic SQL is also recommended for DDL to avoid dependency problems and package invalidation. It is also required to implement recursion.

L'SQL dinamico è anche raccomandato per il DDL per evitare problemi di dipendenza e invalidazione dei package. È anche richiesto per implementare la ricorsione.

Dynamic SQL can be executed using two approaches:

- Using the EXECUTE IMMEDIATE statement – this is ideal for single execution SQL
- Using the PREPARE statement along with the EXECUTE statement - ideal for multiple execution SQL

The following code snippet provides an example of Dynamic SQL using the two approaches. The example assumes a table T2 has been created with this definition:

```
CREATE TABLE T2 (c1 INT, c2 INT)
```

```
CREATE PROCEDURE dyn1 (IN value1 INT, IN value2 INT)
SPECIFIC dyn1
BEGIN
DECLARE stmt varchar(255);
DECLARE st STATEMENT;

SET stmt = 'INSERT INTO T2 VALUES (?, ?)';

PREPARE st FROM stmt;

EXECUTE st USING value1, value1;
EXECUTE st USING value2, value2;

SET stmt = 'INSERT INTO T2 VALUES (9,9)';
EXECUTE IMMEDIATE stmt;
END
```

L'SQL dinamico può essere eseguito usando due approcci:

- Usando lo statement EXECUTE IMMEDIATE - questo è ideale per una singola esecuzione SQL
- Usando lo statement PREPARE con lo statement EXECUTE - ideale per esecuzioni multiple di SQL

Il seguente frammento di codice offre un esempio di SQL dinamico usando i due approcci. L'esempio suppone che una tabella T2 sia stata creata con questa definizione:

15

Chapter 15 – Inline SQL PL, Triggers, and UDFs / Capitolo 15 – SQL PL inline, Trigger, e UDF

In this chapter, we discuss inline SQL PL and other database application objects such as user-defined functions (UDFs) and triggers.

In questo capitolo, discutiamo l'inline SQL PL e altre applicazioni database come le funzioni user-defined (UDFs) e i trigger.

Note:

For more information about UDFs and triggers, watch these videos:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4367>
<http://www.channeldb2.com/video/video/show?id=807741:Video:4362>

Nota:

Per ulteriori informazioni su UDF e su trigger, guardi questi video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4367>
<http://www.channeldb2.com/video/video/show?id=807741:Video:4362>

15.1 Inline SQL PL / SQL PL inline

In Chapter 14 we discussed creating stored procedures using the SQL PL language. The SQL PL language can also be used “inline” meaning that it can stand alone without the need to create a stored procedure. The SQL PL used in UDFs and triggers is also inline because the code is added inline with the UDF/trigger code, and it is dynamic SQL in nature. Inline SQL PL supports only a subset of all the SQL PL statements. Here are inline SQL PL supported keywords:

Nel capitolo 14 abbiamo discusso come creare stored procedure usando il linguaggio SQL PL. Il linguaggio SQL PL può anche essere usato “inline” intendendo che può essere stand-alone senza la necessità di creare una stored procedure. L'SQL PL usato nelle UDF e nei trigger è inoltre inline perché il codice si aggiunge inline con il codice di UDF/trigger, ed è SQL dinamico per natura. L'SQL PL inline supporta soltanto un sottoinsieme di tutti gli statement SQL PL. Qui ci sono le keyword supportate da SQL PL inline:

```
DECLARE <variable>
SET
CASE
FOR
GET DIAGNOSTICS
GOTO
IF
RETURN
SIGNAL
WHILE
ITERATE
LEAVE
```

Here are keywords not supported in inline SQL PL:

```
ALLOCATE CURSOR
ASSOCIATE LOCATORS
DECLARE <cursor>
DECLARE ...HANDLER
PREPARE
EXECUTE
EXECUTE IMMEDIATE
LOOP
REPEAT
RESIGNAL
CALL
COMMIT/ROLLBACK
```

Here is an example of dynamic compound SQL using inline SQL PL. If you want to try it out, you can put it as part of a script file, and ensure you create the following tables:

```
CREATE TABLE T1 (c1 INT)
CREATE TABLE T3 (c1 INT)
```

```
BEGIN ATOMIC
  DECLARE cnt          INT DEFAULT 0;
  DECLARE sumevens    INT DEFAULT 0;
  DECLARE err_msg     VARCHAR(1000) DEFAULT '';
  WHILE (cnt < 100) DO
    IF mod(cnt,2) = 0 THEN
      SET sumevens = sumevens + cnt;
    END IF;
    SET cnt=cnt+1;
  END WHILE;
  INSERT INTO T3 values (sumevens);
  SET cnt = (SELECT 0 FROM SYSIBM.SYSDUMMY1);
```

Qui ci sono le keyword non supportate da SQL PL inline:

Ecco un esempio di dynamic compound SQL usando l'SQL PL inline. Se vuoi provarlo, puoi metterlo come parte di uno script file assicurandoti che crei le seguenti tabelle:

```
FOR cur1 AS SELECT * FROM T1 DO
  IF cur1.c1 > 100 THEN
    SET cnt = cnt + 1;
  END IF;
END FOR;

SET err_msg = 'Rows with values > 100 is: ' ||
char(cnt);
SIGNAL SQLSTATE '80000' SET MESSAGE_TEXT = err_msg;
END!
```

If you save the above inline SQL PL into a script file called “myScript.txt” you could execute it as follows:

```
db2 -td! -vf myScript.txt
```

Se salvi l'SQL PL inline mostrato sopra in uno script file chiamato “myScript.txt”, potresti eseguirlo come segue:

15.2 Triggers / I trigger

Triggers are database objects associated with a table that define operations to occur when an INSERT, UPDATE, or DELETE operation is performed on the table. They are activated (or “fired”) automatically. The operations that cause triggers to fire are called *triggering* SQL statements.

I trigger sono database object associati con una tabella che definiscono le operazioni che devono essere eseguite quando un'operazione di INSERT, UPDATE, o DELETE è avvenuta sulla tabella. Sono attivati (o lanciati) automaticamente. Le operazioni che causano il lancio di trigger sono chiamate *triggering* SQL statement.

15.2.1 Types of triggers / Tipi di trigger

There are three types of triggers: “before” triggers, “after” triggers, and “instead of” triggers.

Ci sono tre tipi di trigger: i trigger “before”, i trigger “after”, e i trigger “instead of”.

Before triggers / I trigger “before”

Before triggers are activated before a row is inserted, updated or deleted. The operations performed by this trigger cannot activate other triggers (so INSERT, UPDATE, and DELETE operations are not permitted).

I trigger “before” sono attivati prima che un riga sia inserita, aggiornata o cancellata. Le operazioni realizzate da questo trigger non possono attivare altri trigger (quindi operazioni di INSERT, UPDATE, DELETE non sono permesse).

An example of simple before trigger is shown in Figure 15.1.

Un esempio di un semplice trigger “before” è mostrato nella figura 15.1.

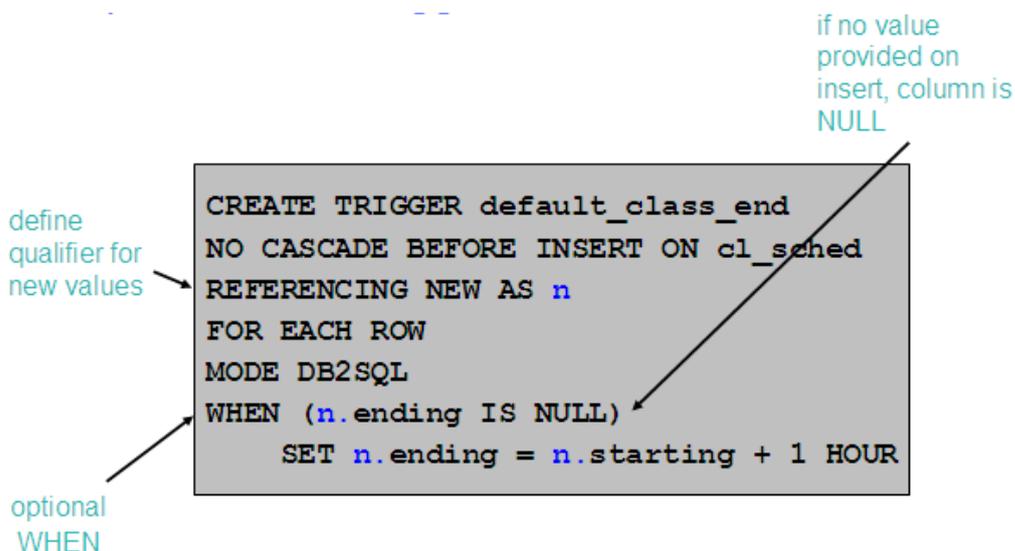


Figure 15.1 – Example of a before trigger / Figura 15.1 - Esempio di un trigger “before”

In Figure 15.1 the trigger “default_class_end” will be triggered before an INSERT SQL statement is performed on the table cl_sched. This table is part of the SAMPLE database, so you can create and test this trigger yourself while connected to this database. The variable “n” in the trigger definition will represent the new value in an INSERT, that is, the value being inserted. The trigger will check the validity of what is being inserted into the table. If the column “ending” has no value during an insert, the trigger will ensure it has the value of the column starting plus 1 hour.

Nella figura 15.1 il trigger “default_class_end” sarà innescato prima che uno statement SQL INSERT sia eseguito sulla tabella cl_sched. Questa tabella fa parte del database SAMPLE, così puoi tu stesso creare e testare questo trigger mentre sei collegato al database. La variabile “n” nella definizione del trigger rappresenterà il nuovo valore in un INSERT, cioè, il valore che è inserito. Il trigger controllerà la validità di ciò che sarà inserito nella tabella. Se la colonna “ending” non ha valore durante un inserimento, il trigger assicurerà che abbia il valore della colonna iniziale più 1 ora.

The following statements show how to test the trigger.

I seguenti statement mostrano come testare il trigger.

```

C:\Program Files\IBM\SQLLIB\BIN>db2 insert into cl_sched
(class_code, day, starting) values ('abc',1,current time)
DB20000I The SQL command completed successfully.
  
```

```
C:\Program Files\IBM\SQLLIB\BIN>db2 select * from cl_sched

CLASS_CODE DAY      STARTING ENDING
-----
042:BF      4 12:10:00 14:00:00
553:MJA     1 10:30:00 11:00:00
543:CWM     3 09:10:00 10:30:00
778:RES     2 12:10:00 14:00:00
044:HD     3 17:12:30 18:00:00
abc        1 11:06:53 12:06:53

6 record(s) selected.
```

The trigger “validate_sched” shown below extends the functionality of the “default_class_end” trigger previously described. Again, you can create it and test it out against the SAMPLE database.

Il trigger “validate_sched” mostrato sotto estende la funzionalità del trigger “default_class_end” precedentemente descritto. Di nuovo, puoi crearlo e testarlo sul database SAMPLE.

```
CREATE TRIGGER validate_sched
NO CASCADE BEFORE INSERT ON cl_sched
REFERENCING NEW AS n
FOR EACH ROW
MODE DB2SQL
BEGIN ATOMIC
-- supply default value for ending time if null
IF (n.ending IS NULL) THEN
    SET n.ending = n.starting + 1 HOUR;
END IF;

-- ensure that class does not end beyond 9pm
IF (n.ending > '21:00') THEN
    SIGNAL SQLSTATE '80000'
    SET MESSAGE_TEXT='class ending time is beyond 9pm';
ELSEIF (n.DAY=1 or n.DAY=7) THEN
    SIGNAL SQLSTATE '80001'
    SET MESSAGE_TEXT='class cannot be scheduled on a
weekend';
END IF;
END
```

After triggers / I trigger “after”

After triggers are activated after the triggering SQL statement has executed to successful completion. The operations performed by this trigger may activate other triggers (cascading is permitted up to 16 levels). After triggers support INSERT, UPDATE and DELETE operations. Below is an example of an after trigger.

I trigger “after” sono attivati dopo che il triggering SQL statement è stato eseguito con successo. Le operazioni eseguite da questo trigger possono attivare altri trigger (l'attivazione a cascata è consentita fino a 16 livelli). I trigger “after” supportano le operazioni di INSERT, di UPDATE, e di DELETE. Sotto un esempio di un trigger “after”.

```
CREATE TRIGGER audit_emp_sal
AFTER UPDATE OF salary ON employee
REFERENCING OLD AS o NEW AS n
FOR EACH ROW
MODE DB2SQL
  INSERT INTO audit VALUES (
    CURRENT TIMESTAMP, ' Employee ' || o.empno || '
salary changed from ' || CHAR(o.salary) || ' to ' ||
CHAR(n.salary) || ' by ' || USER)
```

In this example, the trigger `audit_emp_sal` is used to perform auditing on the column “salary” of the “employee” table. When someone makes a change to this column, the trigger will be fired to write the information about the changed made to the salary into another table called “audit”. The “OLD as o NEW as n” line indicates that the prefix “o” will be used to represent the old or existing value in the table, and the prefix “n” will be used to represent the new value coming from the UPDATE statement. Thus, “o.salary” represents the old or existing value of the salary, and “n.salary” represents the updated value for the column salary data.

In questo esempio, il trigger `audit_emp_sal` è usato per eseguire “il controllo” sulla colonna “salary” della tabella “employee”. Quando qualcuno fa un cambiamento a questa colonna, il trigger sarà lanciato per scrivere le informazioni sul cambiamento fatto sullo stipendio in un'altra tabella chiamata “audit”. La linea “OLD as o NEW as n” indica che il prefisso “o” sarà usato per rappresentare il valore vecchio o il valore esistente nella tabella, ed il prefisso “n” sarà usato per rappresentare il valore nuovo proveniente dallo statement UPDATE. Quindi, “o.salary” rappresenta il valore vecchio o il valore esistente dello stipendio e “n.salary” rappresenta il valore aggiornato del dato della colonna “salary”.

“Instead of” triggers / I trigger “instead of”

Instead of triggers are defined on views. The logic defined in the trigger is executed instead of the triggering SQL statement. For example, if you perform an update operation on a view, the instead of trigger will be fired to actually perform the update on the base tables that form the view.

Triggers cannot be created from IBM Data Studio. They can be created from the Control Center or from the Command line tools (Command Window, Command Line Processor, or the Command Editor).

I trigger “instead of” sono definiti sulle view. La logica definita nel trigger è eseguita al posto del triggering statement SQL. Per esempio, se esegui un'operazione di update su una view, il trigger “instead of” sarà lanciato per eseguire realmente l'aggiornamento sulle tabelle di base che formano la view.

I trigger non possono essere creati dall'IBM Data Studio. Possono essere creati dal Control Center o dai tool della command line (Command Window, Command Line Processor, o Command Editor).

Quicklab #12 – Creating a trigger in the Control Center / Creare un trigger nel Control Center

Objective

Triggers are a database object used to perform business logic when a data modification operation occurs on a table. In this Quicklab, you will create a trigger using the Control Center. This trigger will keep a log of changes made to the *SALES* table for auditing purposes. You will log the user ID who made the change, as well as the time of day the change was made.

Procedure

1. Open the Control Center.
2. For this Quicklab, you will need to create an additional table to be used for logging. Create a table with the following characteristics:

Table name: `saleslog`

First column:

Name: `userid`
Data type: `VARCHAR(128)`
Other attributes: `NOT NULL`

Second column:

Name: `daytime`
Data type: `TIMESTAMP`
Other attributes: `NOT NULL`

Hint: Create this table using the `CREATE TABLE` statement in Command Editor, or use the *Create Table* wizard from Control Center.

Obiettivo

I trigger sono database object usati per eseguire business logic quando un'operazione di modifica sui i dati avviene su una tabella. In questo Quicklab, creerai un trigger usando il Control Center. Questo trigger manterrà un log dei cambiamenti fatti alla tabella *SALES* per scopi di verifica. Annoterai l'user ID che ha fatto il cambiamento, così come l'ora a cui il cambiamento è avvenuto.

Procedura

1. Apri il Control Center.
2. Per questo Quicklab, dovrai creare una tabella supplementare per i log.

Crea una tabella con le seguenti caratteristiche:

Nome della Tabella: `saleslog`

Prima colonna:

Nome: `userid`
Data type: `VARCHAR (128)`
Altri attribuiti: `NOT NULL`

Seconda colonna:

Nome: `daytime`
Data type: `TIMESTAMP`
Altri attribuiti: `NOT NULL`

Suggerimento: Crea questa tabella usando lo statement `CREATE TABLE` nel Command Editor, o usando il wizard *Create Table* dal Control Center.

3. From Control Center, expand the *EXPRESS* database folder. Right-click on the *Triggers* folder and select the *Create* option. The *Create Trigger* window opens.

4. Fill in the following information in the window:

Trigger schema: User ID of the user you are logged in as (should be the default setting)

Trigger name: *audit_sales*

Table or view schema: User ID of the user you are logged in as (should be the default setting)

Table or view name: *SALES*

Time to trigger action: *After*

Operation that causes the trigger to be executed: *Update of columns* (do not specify any columns because we want the trigger to fire when any of the columns are updated).

Comment: *Logs all update actions on Sales table.*

3. Dal Control Center, espandi la cartella del database *EXPRESS*. Fai clic destro sulla cartella *Triggers* e seleziona l'opzione *Create*. La finestra *Create Trigger* si apre.

4. Inserisci le seguenti informazioni nella finestra:

Schema del trigger: Lo user ID dell'utente con cui sei loggato (dovrebbe essere il setting di default)

Nome del trigger: *audit_sales*

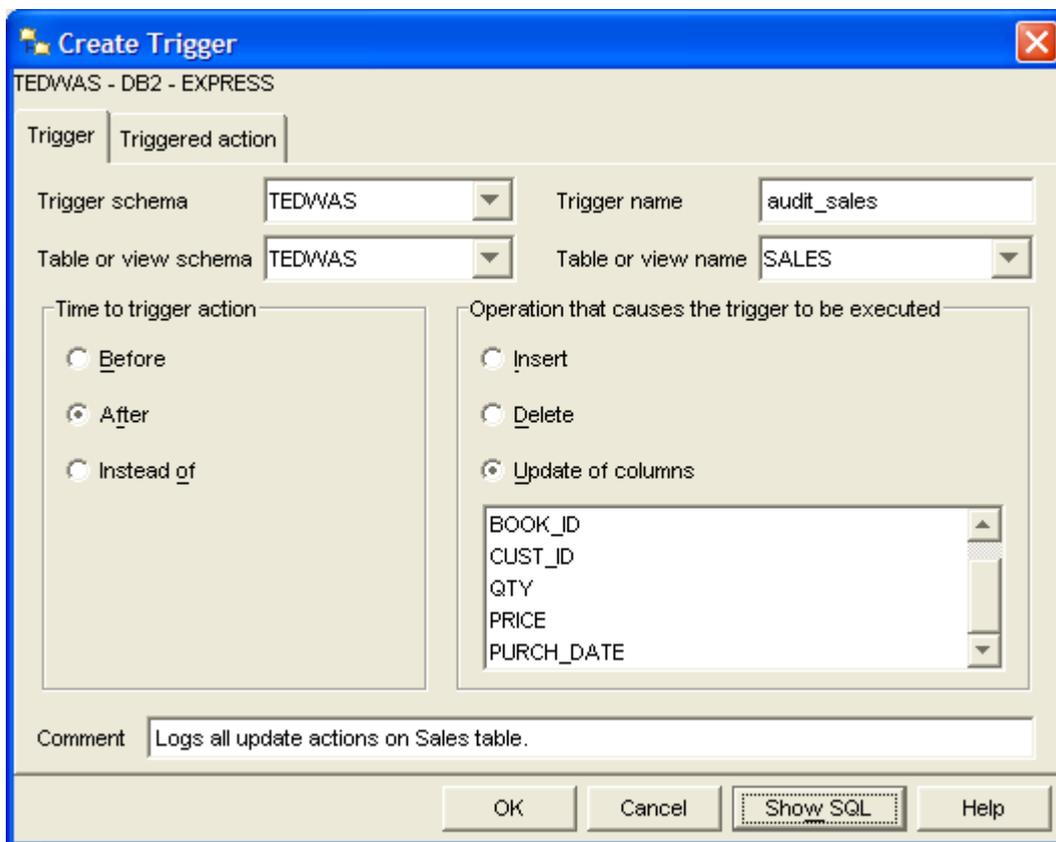
Schema della tabella o della view: L'user ID dell'utente con cui sei loggato (dovrebbe essere il setting di default)

Nome della tabella o della view: *SALES*

Tempistica dell'azione del trigger: *After*

L'operazione che implica l'esecuzione del trigger: *Aggiornamento delle colonne* (non specificare alcuna colonna perché vogliamo che il trigger sia lanciato quando alcune delle colonne sono aggiornate).

Commento: *Annota tutte le azioni dell'aggiornamento sulla tabella Sales.*



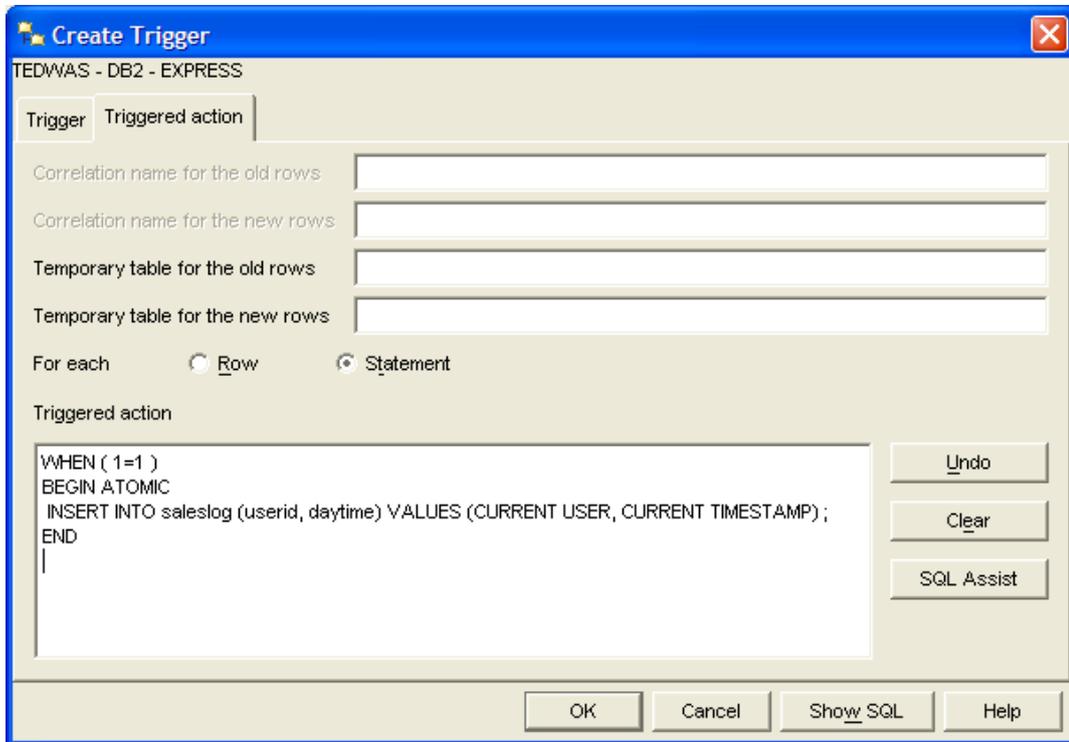
5. On the *Triggered action* tab, select the *For Each STATEMENT* option. Use the following code for the triggered action:

```
WHEN ( 1=1 )
BEGIN ATOMIC
  INSERT INTO saleslog (userid, daytime) VALUES (CURRENT
USER, CURRENT TIMESTAMP);
END
```

(Note: A statement trigger fires once after the statement activating the trigger has completed. A row trigger specifies that the triggered action will execute every time the triggering SQL statement affects a row.)

5. Sul tab *Triggered action*, seleziona l'opzione *For Each STATEMENT*. Usa il seguente codice per l'azione del trigger:

(Nota: Il trigger statement è lanciato una volta dopo che lo statement che ha attivato il trigger ha finito. Un trigger specifica che l'azione del trigger sarà eseguita ogni volta che il triggering statement SQL coinvolge una riga.)



Click the *OK* button to create the trigger.

Clicca *OK* per creare il trigger.

6. You should now be able to view the trigger in the *Triggers* folder in the Control Center.
7. Query the *saleslog* table to ensure there is no data in it. Delete any rows that may be in it (`DELETE FROM saleslog`).
8. Try to update a record in the *sales* table. (Hint: use the Command Editor or the SQL Assist Wizard).
9. Check the contents of the *saleslog* table again. How many rows are in it?

6. Adesso dovresti poter osservare il trigger nella cartella *Triggers* nel Control Center.
7. Interroga la tabella *saleslog* per assicurarti che non contenga dati. Cancella tutte le righe che possono esistere (`DELETE FROM saleslog`).
8. Prova ad aggiornare un record nella tabella *sales*. (Suggerimento: usa il Command Editor o l'SQL Assist Wizard.)
9. Controlla ancora il contenuto della tabella *saleslog*. Quante righe ci sono?

15.3 User-defined functions (UDFs) / Funzioni definite dall'utente (UDFs)

A user-defined function (UDF) is a database application object that maps a set of input data values into a set of output values. For example, a function may take a measurement in inches as input, and return the result in centimeters.

DB2 supports creating functions using SQL PL, C/C++, Java, CLR (Common Language Runtime), and OLE (Object Linking and Embedding). In this book, we focus on SQL PL functions because of their simplicity, popularity, and performance.

There are four types of functions: scalar, table, row, and column functions. In this chapter, we focus only on scalar and table functions.

Una funzione user-defined (UDF) è un database application object che crea un collegamento tra un insieme di valori di dati di input in un insieme di valori di output. Per esempio, una funzione può prendere una misura in pollici come input e ritornare il risultato in centimetri.

DB2 supporta la creazione di funzioni usando SQL PL, C/C++, Java, CLR (Common Language Runtime) e OLE (Object Linking and Embedding). In questo libro, ci focalizzeremo sulle funzioni SQL PL a causa delle loro semplicità, popolarità, e prestazioni.

Ci sono quattro tipi di funzioni: funzioni scalari, di tabella, di riga, di colonna. In questo capitolo, ci concentreremo soltanto sulle funzioni scalari e di tabella.

15.3.1 Scalar functions / Funzioni scalari

Scalar functions return a single value. Scalar functions cannot include SQL statements that will change the database state; that is, INSERT, UPDATE, and DELETE statements are not allowed. Some built-in scalar functions are SUM(), AVG(), DIGITS(), COALESCE(), and SUBSTR().

DB2 allows you to build customized user-defined functions where you can encapsulate frequently used logic. For example, consider a migration of your application from Oracle to DB2. In your application, you widely invoke Oracle's NVL() function. The equivalent built-in function in DB2 is called COALESCE. Rather than renaming all the NVL functions in your application to COALESCE, you could create a user-

Le funzioni scalari ritornano un valore singolo. Le funzioni scalari non possono includere statement SQL che cambieranno lo stato del database; cioè statement di INSERT, UPDATE e DELETE non sono permessi. Alcune funzioni scalari built-in sono SUM(), AVG(), DIGITS(), COALESCE() e SUBSTR().

DB2 ti permette di costruire funzioni personalizzate definite dall'utente dove puoi incapsulare la logica che viene usata di frequente. Per esempio, considera una migrazione delle tue applicazioni da Oracle a DB2. Nella tua applicazione, invochi spesso la funzione NVL() di Oracle. La funzione built-in equivalente in DB2 si chiama COALESCE. Piuttosto che rinominare tutte le funzioni NVL in COALESCE

defined function in DB2, call it NVL and have it invoke the COALESCE function as shown below.

nella tua applicazione, potresti creare una funzione definita dall'utente in DB2, chiamarla NVL e farle invocare la funzione di COALESCE come mostrato sotto.

```
CREATE FUNCTION NVL (p_var1 VARCHAR(30),
                    p_var2 VARCHAR(30))
SPECIFIC nvlvarchar30
RETURNS VARCHAR(30)
RETURN COALESCE(p_var1, p_var2)
```

The COALESCE function then returns the first argument that is not null.

La funzione di COALESCE poi ritorna il primo argomento che non è nullo.

Below is another example of a scalar function. The function is called "deptname" and it returns the department number of an employee based on the employee id.

Sotto è riportato un altro esempio di una funzione scalare. La funzione è chiamata "deptname" e ritorna il numero del dipartimento di un impiegato basato sull'id dell'impiegato.

```
CREATE FUNCTION deptname(p_empid VARCHAR(6))
RETURNS VARCHAR(30)
SPECIFIC deptname
BEGIN ATOMIC
  DECLARE v_department_name VARCHAR(30);
  DECLARE v_err VARCHAR(70);
  SET v_department_name = (
    SELECT d.deptname FROM department d, employee e
    WHERE e.workdept=d.deptno AND e.empno= p_empid);
  SET v_err = 'Error: employee ' || p_empid || ' was not
found';
  IF v_department_name IS NULL THEN
    SIGNAL SQLSTATE '80000' SET MESSAGE_TEXT=v_err;
  END IF;
  RETURN v_department_name;
END
```

To test the function, try the following from the Command Window or Linux/UNIX shell:

Per verificare la funzione, prova il seguente comando dalla Command Window o dallo shell di Linux/UNIX:

```
db2 "values (deptname ('000300'))"
```

Invoking scalar UDFs / Invocare le UDF scalari

Scalar UDFs can be invoked in SQL statements wherever a scalar value is expected, or in a VALUES clause. Here are two examples showing a call to the COALESCE scalar function:

```
SELECT DEPTNAME, COALESCE(MGRNO, 'ABSENT') FROM DEPARTMENT
VALUES COALESCE('A', 'B')
```

Le UDF scalari possono essere invocate negli statement SQL dovunque un valore scalare sia previsto, o in una VALUES clause. Qui ci sono due esempi che mostrano una chiamata alla funzione scalare di COALESCE:

15.3.2 Table functions / Funzioni di tabella

Table functions return a table of rows. You can call them using the FROM clause of a query. Table functions, as opposed to scalar functions, can change the database state; therefore, INSERT, UPDATE, and DELETE statements are allowed. Some built-in table functions are SNAPSHOT_DYN_SQL() and MQREADALL(). Table functions are similar to views, but since they allow for data modification statements (INSERT, UPDATE, and DELETE) they are more powerful. Typically they are used to return a table and keep an audit record.

Le funzioni di tabella ritornano una tabella di righe. Puoi chiamarle usando la FROM clause della query. Le funzioni di tabella, a differenza delle funzioni scalari, possono cambiare lo stato del database; quindi, gli statement INSERT, UPDATE e DELETE sono consentiti. Alcune funzioni di tabella built-in sono SNAPSHOT_DYN_SQL() e MQREADALL(). Le funzioni di tabella sono simili alle view, ma poiché consentono statement di modifica (INSERT, UPDATE e DELETE), sono più potenti. Sono usati tipicamente per ritornare una tabella e per mantenere un record di verifica.

Below is an example of a table function that enumerates a set of department employees:

```
CREATE FUNCTION getEnumEmployee(p_dept VARCHAR(3))
RETURNS TABLE
  (empno CHAR(6),
   lastname VARCHAR(15),
   firstnme VARCHAR(12))
SPECIFIC getEnumEmployee
RETURN
  SELECT e.empno, e.lastname, e.firstnme
  FROM employee e
  WHERE e.workdept=p_dept
```

Di seguito è riportato un esempio di una funzione di tabella che enumera un insieme di impiegati di un dipartimento:

To test the above function, try:

Per verificare la funzione qui sopra, usa:

```
db2 "SELECT * FROM table(getEnumEmployee('D11')) AS t"
```

Invoking table UDFs / Invocare le UDFs di tabella

A table UDF has to be invoked in the FROM clause of an SQL statement. The TABLE() function must be applied and must be aliased. Figure 15.2 shows an example of how to invoke the function "getEnumEmployee" which we had tested earlier as well.

Un UDF di tabella deve essere invocata nella clausola FROM di uno statement SQL. La funzione TABLE() deve essere applicata e deve avere un alias. La figura 15.2 mostra un esempio di come invocare la funzione "getEnumEmployee" che era stata testata anche in precedenza.

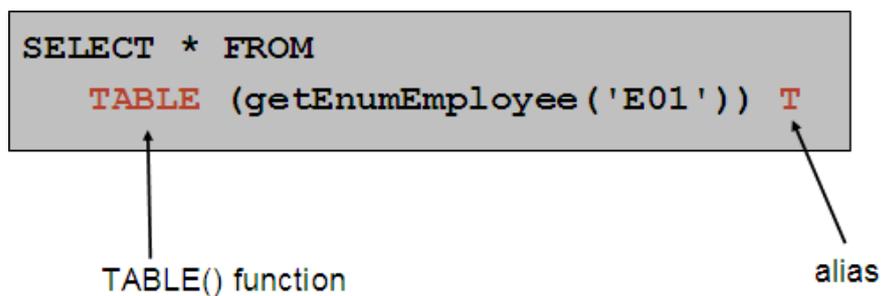


Figure 15.2 – Invoking a table function / Figura 15.2 - Invocare una funzione di tabella

Quicklab #13 – Creating a UDF using the IBM Data Studio / Creare un UDF usando l'IBM Data Studio

Objective

In this Quicklab, you will create a scalar user-defined function in IBM Data Studio. This will give you more experience with Data Studio, as well as improving your familiarity with the SQL PL language for user-defined functions.

Procedure

1. Open the IBM Data Studio (Hint: it is available through the Start menu).
2. From the Data Project Explorer window, choose the project you created in the previous Quicklab and select *Open Project*.
3. Right-click the *User-Defined Functions* folder. Select the *New* menu item. Select the *SQL User-Defined Function* menu item. You could have alternatively selected the *User-Defined Function using Wizard* item if you wanted to be guided through the process using a GUI wizard.
4. The *Editor* view should open with a skeletal function. Modify the code as follows:

```
CREATE FUNCTION booktitle(p_bid INTEGER)
RETURNS VARCHAR(300)
-----
SQL UDF (Scalar)
```

Obiettivo

In questo Quicklab, creerai una funzione scalare definita dall'utente in IBM Data Studio. Questo ti darà maggiore esperienza con il programma Data Studio, inoltre migliorerà la tua confidenza con il linguaggio SQL PL per le funzioni user-defined.

Procedura

1. Apri IBM Data Studio (Suggerimento: è disponibile attraverso il menu Start).
2. Dalla finestra Data Project Explorer, scegli il progetto che hai creato nel precedente Quicklab e seleziona *Open Project*.
3. Clic destro su *User-Defined Functions*. Seleziona l'elemento del menù *New*. Seleziona l'elemento del menù *SQL User-Defined Function*. Potresti selezionare alternativamente l'elemento *User-Defined Function using Wizard* se volessi essere guidato nel processo usando un GUI wizard.
4. La vista dell'Editor dovrebbe aprirsi con una funzione embrionale. Modifica il codice come segue:

```
-----  
SPECIFIC booktitle  
F1: BEGIN ATOMIC  
DECLARE v_book_title VARCHAR(300);  
DECLARE v_err VARCHAR(70);  
SET v_book_title = (SELECT title FROM books WHERE p_bid =  
book_id);  
SET v_err = 'Error: The book with ID ' || CHAR(p_bid) || '  
was not found.';  
IF v_book_title IS NULL THEN SIGNAL SQLSTATE '80000' SET  
MESSAGE_TEXT=v_err;  
END IF;  
RETURN v_book_title;  
END
```

5. Build the function by right-clicking on the function and choosing *Deploy*.
6. Run the function by clicking the *Run* button in the toolbar.
7. Since the function accepts one input parameter, a window appears asking you to fill in a value for the parameter.

Enter the value: 80002

What is the result?

Try again with the value: 1002

What happens this time? (Hint: Look in the *Messages* section of the *Output* view).

8. Close the IBM Data Studio when you are finished.

5. Costruisci la funzione facendo clic destro sulla funzione e scegliendo *Deploy*.
6. Esegui la funzione facendo clic sul tasto *Run* nella toolbar.
7. Siccome la funzione accetta un parametro di input, una finestra si aprirà per chiederti di fornire un valore per il parametro.

Inserisci il valore: 80002

Qual è il risultato?

Prova ancora con il valore: 1002

Che cosa succede questa volta? (Suggerimento: Osserva nella sezione *Messages* della *Output* view).

8. Chiudi IBM Data Studio quando hai finito.

16

Chapter 16 – DB2 pureXML / Capitolo 16 – pureXML DB2

In this chapter we discuss pureXML, the new technology provided starting with DB2 9 to support XML native storage. Many of the examples and concepts discussed in this chapter have been taken from the IBM Redbook: *DB2 9: pureXML overview and fast start*. See the Resources section for more information. Figure 16.1 outlines which section of the DB2 "Big Picture" we will discuss in this chapter.

In questo capitolo discutiamo pureXML, la nuova tecnologia fornita a partire da DB2 9 per supportare lo "storage nativo" di XML. Molti degli esempi e dei concetti discussi in questo capitolo sono stati presi dall'IBM Redbook: *DB2 9: pureXML overview and fast start* (inizio veloce). Vedi la sezione Risorse per maggiori informazioni. La figura 16,1 mostra quale sezione della "Big Picture" di DB2 che discuteremo in questo capitolo.

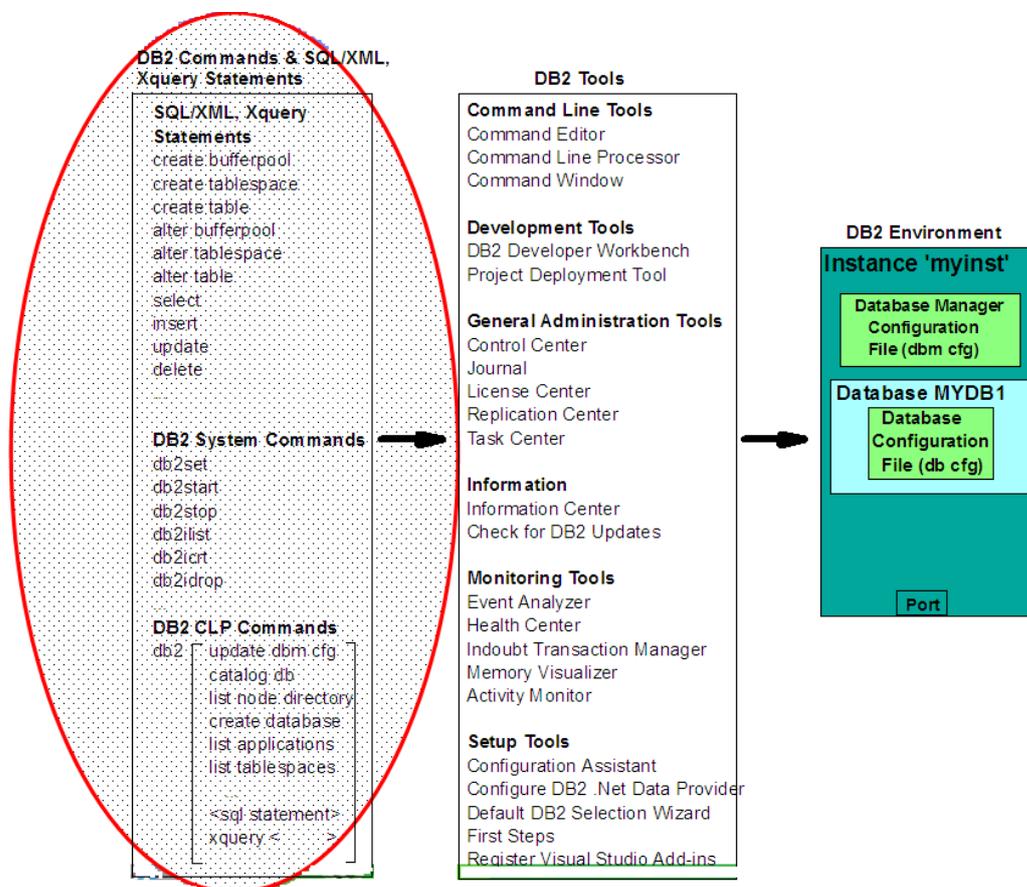


Figure 16.1 – The DB2 Big Picture: DB2 commands, SQL/XML and Xquery / Figura 16.1 - DB2 Big Picture: Comandi di DB2, SQL/XML e XQuery

Note:

For more information about pureXML, watch this video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4382>

Nota:

Per maggiori informazioni su pureXML, guardi questo video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4382>

16.1 Using XML with databases / Usare XML con i database

XML documents can be stored in text files, XML repositories, or databases. There are two main reasons why many companies propose to store them in databases:

I documenti XML possono essere registrati in file di testo, in repository XML, o nei database. Ci sono due motivi principali per cui molte aziende propongono di memorizzare documenti XML nei database:

- Managing large volumes of XML data is a database problem. XML is data like other data, just in a different overall format. The same reasons to store relational data on databases apply to XML data: Databases provide efficient search and retrieval, robust support for persistence of data, backup and recovery, transaction support, performance and scalability.
- Amministrare grandi volumi di dati in formato XML è un problema che richiede l'uso dei database. I dati XML sono a tutti gli effetti dati come gli altri, organizzati però in un diverso formato. Le stesse ragioni che portano a memorizzare i dati relazionali nei database valgono per i dati XML: i database forniscono ricerca e recupero efficiente dei dati, un robusto supporto per la persistenza dei dati, backup e ripristino, supporto alle transazioni, performance e scalabilità.
- Integration: by storing relational and XML documents together, you can integrate new XML data with existing relational data, and combine SQL with XPath or XQuery in one query. Moreover, relational data can be published as XML, and vice versa. Through integration, databases can better support Web applications, SOA, and web services.
- Integrazione: memorizzando documenti relazionali e dati XML insieme, si possono integrare i nuovi dati XML con dati relazionali già esistenti, e unire SQL con XPath o con XQuery in un'unica query. Inoltre, i dati relazionali possono essere pubblicati in formato XML, e viceversa. Attraverso l'integrazione, i database possono supportare meglio le applicazioni Web, SOA, e i web service.

16.2 XML databases / Database XML

There are two types of databases for storing XML data:

- XML-enabled databases
- Native XML databases

Ci sono due tipi di database per registrare dati XML:

- Database XML-enabled
- Database nativi XML

16.2.1 XML-enabled databases / Database XML-enabled

An XML-enabled database uses a relational model for the core data storage model. This requires a mapping between the XML (hierarchical) data model and the relational data model, or else storing the XML data as a character large object. While this can be considered as an "old" technology, it is

Un database XML-enabled usa un modello relazionale per la memorizzazione dei dati al suo interno. Questa tecnica richiede una mappatura tra il modello dati XML (gerarchico) e il modello dati relazionale, oppure la memorizzazione dei dati XML come un grande oggetto di tipo carattere. Questo

still being used by many database vendors. Figure 16.2 explains in more detail the two options for XML-enabled databases.

approccio può essere considerato come una "vecchia" tecnologia, ma esso è ancora usato da molti produttori di database. La figura 16,2 spiega più in dettaglio le due opzioni per i database XML-enabled.

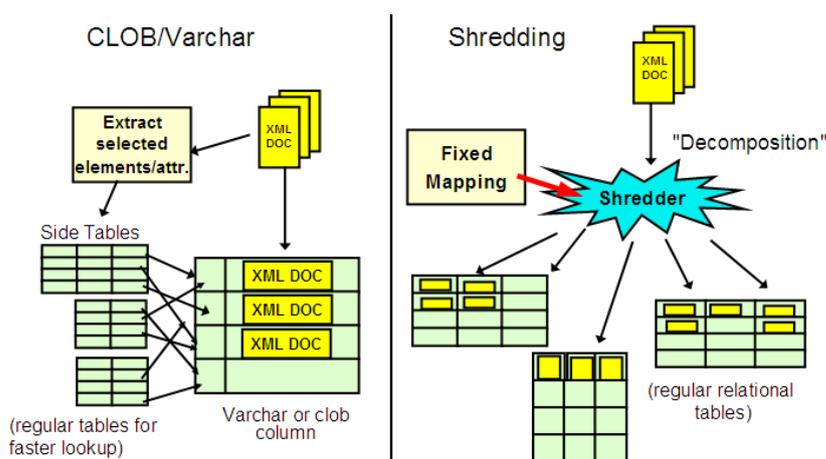


Figure 16.2 – Two options to store XML in XML-enabled databases / Figura 16.2 - Le due opzioni per memorizzare dati XML in database XML-enabled

The left side of Figure 16.2 shows the "CLOB and varchar" method of storing XML documents in a database. Using this method, an XML document is stored as an unparsed string in either a CLOB or a varchar in the database. If the XML document is stored as a string, when you want to retrieve part of the XML document, your program will have to retrieve the string, and perform the parsing to find what you want. This method is not very flexible.

La parte sinistra della figura 16.2 mostra il metodo "CLOB" e "varchar" per registrare documenti XML in un database. Usando questo metodo, un documento XML viene registrato come una stringa non parsificata sia nel formato dati CLOB che in un varchar all'interno database. Se il documento XML è registrato come una stringa, quando si vuole recuperare parte del documento XML, il programma dovrà estrarre l'intera stringa, e successivamente eseguire il parsing ed effettuare l'analisi per trovare quello che si desidera. Questo metodo non risulta molto flessibile.

The other option for XML-enabled databases is called shredding or decomposition and is illustrated on the right hand side of Figure 16.2. Using this method, an entire XML document is shredded into smaller parts which are stored in tables. Using this method, the

L'altra opzione per i database XML-enabled è chiamata shredding o decomposizione ed è illustrata nella parte destra della figura 16.2. Usando questo metodo, un intero documento XML è "spezzettato" in parti più piccole le quali sono poi registrate in

hierarchical model of an XML document is forced into the relational model. This is also not good for flexibility: a change in the XML document is not easily propagated into the corresponding tables and many other tables may need to be created. This method is also not good for performance: if you need to get the original XML document back, you need to perform an expensive SQL operation, which grows even more expensive when more tables are joined.

tabelle. Utilizzando questo metodo, il modello gerarchico di un documento XML viene forzato al modello relazionale. Anche questo non è vantaggioso per la flessibilità: un cambiamento nel documento XML non viene facilmente propagato alle tabelle corrispondenti e ci può essere la necessità di creare nuove tabelle. Anche le prestazioni risentono dei limiti imposti dal modello: nel momento in cui il documento originale XML deve essere ricostruito, si deve eseguire una costosa operazione SQL, la quale diventa ancor più onerosa quando viene fatto il join di più tabelle.

16.2.2 Native XML Databases / Database Native XML

Native XML databases use the hierarchical XML data model to store and process XML internally. The storage format is the same as the processing format: there is no mapping to the relational model, and XML documents are not stored as images. When XPath or XQuery statements are used, they are processed natively by the engine, and not converted to SQL. This is why these databases are known as “native” XML databases. DB2 starting with version 9 is currently the only commercial data server providing this support.

I database Native XML usano il modello gerarchico di XML per memorizzare e processare internamente dati XML. Il formato di memorizzazione è lo stesso di quello usato per la fase di elaborazione: non c'è alcuna mappatura al modello relazionale, e i documenti XML non vengono memorizzati come immagini. Quando le istruzioni di XPath o di XQuery sono usate, i dati sono trattati nativamente dall'engine, e non sono convertiti in SQL. Ecco perché questi database sono conosciuti come database "nativi" XML. DB2, a partire dalla versione 9, è attualmente l'unico data server commerciale che offre questo supporto.

16.3 XML in DB2

Figure 16.3 below outlines how relational data and hierarchical data (XML documents) are stored in DB2 starting with version 9. In the figure, assume the table `dept` is defined as follows:

La figura 16.3 qui sotto delinea come sono registrati in DB2, a partire dalla versione 9, i dati relazionali e i dati gerarchici (cioè i documenti XML). Nella figura, si assume che la tabella `dept` è definita come segue:

```
CREATE TABLE dept (deptID CHAR(8), ..., deptdoc XML);
```

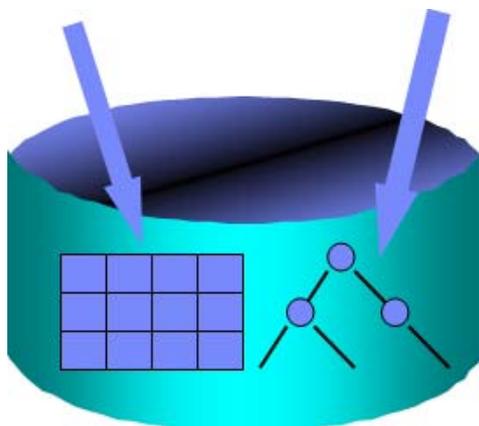


Figure 16.3 – XML in DB2 / Figura 16.3 - XML in DB2

Note that the table definition uses a new data type, XML, for the deptdoc column.

The left arrow in the figure indicates the relational column deptID stored in relational format (tables), while the XML column deptdoc is stored in parsed hierarchical format.

Figure 16.4 illustrates that starting with DB2 9, there are now four ways to access data:

- Use SQL to access relational data
- Use SQL with XML extensions to access XML data
- Use XQuery to access XML data
- Use XQuery to access relational data

Si osservi che la definizione della tabella usa un nuovo data type, XML, per la colonna deptdoc. La freccia a sinistra nella figura indica la colonna relazionale deptID registrata in formato relazionale (tabelle), mentre la colonna XML di deptdoc è registrata in un formato gerarchico parsificabile. La figura 16.4 illustra che a partire dalla versione DB2 9, ci sono ora quattro modi per accedere ai dati:

- Usare SQL per accedere a dati relazionali
- Usare SQL con le estensioni XML per accedere a dati XML
- Usare XQuery per accedere a dati XML
- Usare XQuery per accedere a dati relazionali

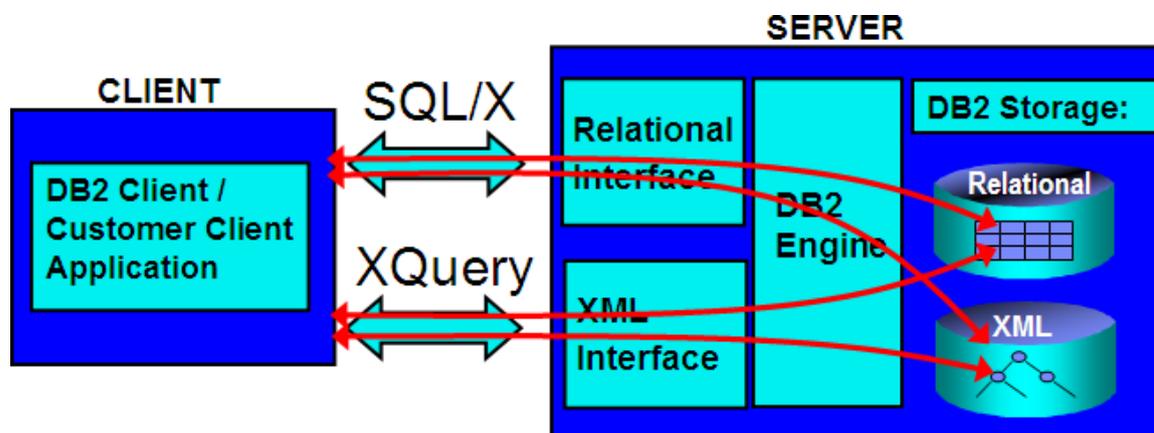


Figure 16.4 – Four ways to access data in DB2 / Figura 16.4 - Quattro modi per accedere ai dati in DB2

Therefore, while an SQL person may see DB2 as a world class RDBMS that also supports XML, an XML person would see DB2 as a world class XML repository that also supports SQL.

Dunque, mentre un utente SQL può considerare DB2 come un RDBMS di prima classe che supporta anche XML, una persona che usa XML considererebbe DB2 come un repository XML che supporta anche SQL.

Note that IBM uses the term pureXML instead of "native XML" to describe this technology. While other vendors still use the old technologies of CLOB/varchar or shredding to store XML documents, they call those old technologies "native XML". To avoid confusion, IBM decided to use the new term pureXML, and to trademark this name so that no other database or XML vendor can use this same term to denote some differing technology. In DB2 9.1, pureXML support was provided only when the database was created as a Unicode database. This restriction has been lifted with DB2 9.5, where you can use pureXML in a unicode or non-unicode database.

Si noti che IBM usa il termine pureXML invece di "native XML" per descrivere questa tecnologia. Mentre gli altri produttori, che usano ancora vecchie tecnologie CLOB, varchar o shredding per registrare documenti XML, chiamano quelle vecchie tecnologie "native XML". Per evitare confusione, IBM ha deciso di usare il nuovo termine pureXML, e di fare di questo nome un marchio di fabbrica cosicché nessun altro fornitore di database o di XML potesse usare questo stesso termine per indicare diverse tecnologie. In DB2 9,1, il supporto di pureXML era fornito soltanto quando il database veniva creato come database Unicode. Questa limitazione è stata rimossa con DB2 9.5, dove è possibile usare il pureXML sia in un database unicode sia non-unicode.

16.3.1 pureXML technology advantages / I vantaggi della tecnologia pureXML

Many advantages are provided by using pureXML technology.

1. You can seamlessly leverage your relational investment, given that XML documents are stored in columns of tables using the new XML data type.
2. You can reduce code complexity. For example, Figure 16.5 illustrates a PHP script written both with and without using pureXML. Using pureXML (the green box) the lines of code are reduced. This not only means that the code is less complex, but the overall performance is improved as there are fewer lines to parse and maintain in the code.

Si ottengono molti vantaggi usando la tecnologia pureXML.

1. Si può usare facilmente il preesistente database relazionale, dato che i documenti XML sono registrati nelle colonne delle tabelle usando il nuovo tipo di dato XML.
2. Si può ridurre la complessità del codice. Per esempio, la figura 16.5 illustra uno script PHP scritto sia senza che con l'uso di pureXML. Usando pureXML (riquadro verde) le linee di codice sono ridotte. Questo non significa solo che il codice è meno complesso, ma le prestazioni generali sono migliorate poiché ci sono meno linee da parsare e mantenere nel codice.

```

<?php
$conn = db2_connect($dbname, $dbuser, $dbpass);

/* Insert Customer Documents */

$stmt = db2_prepare($conn, "VALUES (NEXT VALUE FOR
Cid)");
db2_execute($stmt);
list($Cid) = db2_fetch_array($stmt);

$fileContents = file_get_contents
("customers/c1.xml");

$stmt = db2_prepare($conn, "INSERT INTO xmldatacustomer
(Cid, Info) VALUES (?, ?)");
if(!db2_execute($stmt, array($Cid, $fileContents)))
{
    echo db2_stmt_errormsg($stmt);
}

/* Insert Product Documents */

$fileContents = file_get_contents
("products/p1.xml");
$dom = simplexml_load_string($fileContents);

$prodID = (string) $dom["pid"];

$stmt = db2_prepare($conn, "INSERT INTO xmldataproduct
(Pid, Description) VALUES (?, ?)");
if(!db2_execute($stmt, array($prodID,

```

```

<?php
db2_execute($stmt);
list($Cid) = db2_fetch_array($stmt);

$fileContents = file_get_contents
("customers/c1.xml");
$dom = simplexml_load_string($fileContents);

$custName = (string) $dom->name;
$custCountry = (string) $dom->addr["country"];
$custStreet = (string) $dom->addr->street;
$custCity = (string) $dom->addr->city;
$custProvince = (string) $dom->addr->("prov-state");
$custZip = (string) $dom->addr->("pcode-zip");
$custPhone = (string) $dom->phone;

$stmt = db2_prepare($conn, "INSERT INTO sqicustomer
(Cid, Name, Country, Street, City, Province, Zip,
Phone, Info) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?)");
if(!db2_execute($stmt, array($Cid, $custName,
$custCountry, $custStreet, $custCity, $custProvince,
$custZip, $custPhone, $fileContents)))
{
    echo db2_stmt_errormsg($stmt);
}

/* Insert Product Documents */

$fileContents = file_get_contents
("products/p1.xml");
$dom = simplexml_load_string($fileContents);

$prodID = (string) $dom["pid"];

```

Figure 16.5 – Code complexity with and without pureXML / Figura 16.5 - La complessità del codice con e senza pureXML

3. Changes to your schema are easier using XML and pureXML technology. Figure 16.6 illustrates an example of this increased flexibility. In the figure, assume that you had a database consisting of the tables *Employee* and *Department*. Typically with a non-XML database, if your manager asked you to store not only one phone number per employee (the home phone number), but also a second phone number (a cell phone number), then you could add an extra column to the *Employee* table and store the cell phone number in that new column.

However, this method would be against the normalization rules of relational databases. If you want to preserve these rules, you should instead create a new Phone side table, and move all phone information to this table. You could then also add the cell phone numbers as well. Creating a new "Phone" table is costly, not only because large amounts of pre-existing data needs to be moved, but also because all the SQL in your applications would have to change to point to the new table.

Instead, on the left side of the figure, we show how this could be done using XML. If employee "Christine" also has a cell phone number, a new tag can be added to put this information. If employee "Michael" does not have a cell phone number, we just leave it as is.

3. Le modifiche al tuo schema sono più semplici usando la tecnologia XML e pureXML. La figura 16.6 illustra un esempio dell'aumento di flessibilità. Nella figura, si assuma di avere un database che consiste delle tabelle *Employee* e *Department*. Tipicamente con un database non-XML, se il manager chiedesse di registrare non solo un numero di telefono per dipendente (il numero di telefono di casa), ma anche un secondo numero di telefono (ad esempio un numero di cellulare), allora si potrebbe aggiungere una colonna addizionale alla tabella *Employee* e registrare il numero del cellulare in quella nuova colonna.

Tuttavia, questo metodo sarebbe contro le regole di normalizzazione dei database relazionali. Se si vogliono conservare queste regole, si dovrebbe creare invece una nuova e ulteriore tabella "Phone" e spostare l'informazione relativa al telefono su questa tabella. Successivamente si potrebbe anche aggiungere il numero di telefono cellulare. La creazione della nuova tabella "Phone" è costosa, non soltanto perché grandi quantità di dati pre-esistenti devono essere spostati, ma anche perché tutte le query SQL nelle applicazioni dovrebbero essere cambiate per far riferimento alla nuova tabella.

Invece, sul lato sinistro della figura, è mostrato come questo potrebbe essere fatto usando XML. Se il dipendente "Christine" ha anche un numero di cellulare, una nuova etichetta può essere aggiunta per mettere tali informazioni. Se il dipendente "Michael" non ha un numero di telefono cellulare, semplicemente lo si lascia così com'è.

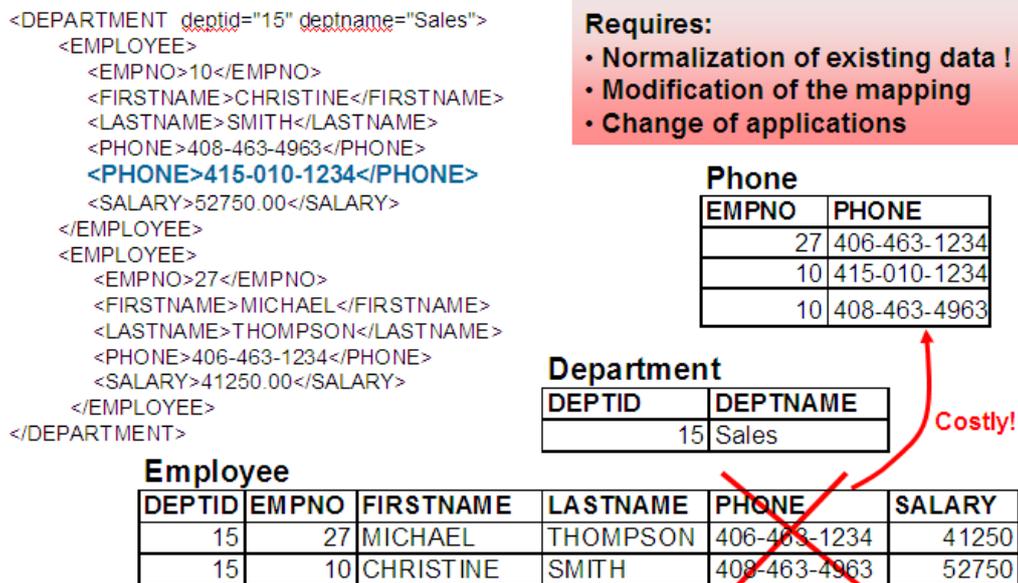


Figure 16.6 – Increased data flexibility using XML / Figura 16.6 - Aumento di flessibilità dei dati usando XML

4. You can improve your XML application performance. Tests performed using pureXML technology showed huge improvements in performance for several XML applications. Figure 16.7 shows the test results for a company that switched to pureXML from older technologies. The middle column shows the results using the old method of working with XML, and the third column shows the results using pureXML.
4. Si possono migliorare le prestazioni delle applicazioni XML. Test effettuati utilizzando la tecnologia pureXML hanno dimostrato grandi miglioramenti nelle prestazioni in parecchie applicazioni XML. La figura 16.7 mostra i risultati delle prove per un'azienda che ha cambiato in pureXML le più vecchie tecnologie. La colonna nel mezzo mostra i risultati usando il vecchio metodo di lavorare con XML, la terza colonna mostra i risultati usando il pureXML.

| Task | Other relational DB | DB2 with pureXML |
|--|---|-----------------------|
| Development of search and retrieval business processes / Sviluppare processi aziendali di ricerca e recupero | CLOB: 8 hrs / ore Shred: 2 hrs / ore | 30 min. |
| Relative lines of I/O code / Linee di codice I/O relative | 100 | 35 (65% reduction) |

| | | |
|---|--------------------|-----------------|
| Add field to schema / Aggiungere un campo allo schema | 1 week / settimana | 5 min. |
| Queries | 24 - 36 hrs / ore | 20 sec - 10 min |

Figure 16.7 – Increased performance using pureXML technology / Figura 16.7 - Prestazioni aumentate usando la tecnologia pureXML

16.3.2 XPath basics / Principi di XPath

XPath is a language that can be used to query XML documents. Figure 16.8 shows an XML document, and Figure 16.9 illustrates the same document represented in parsed-hierarchical (also called “node” or “leaf”) format. We will use the parsed-hierarchical format to explain how XPath works.

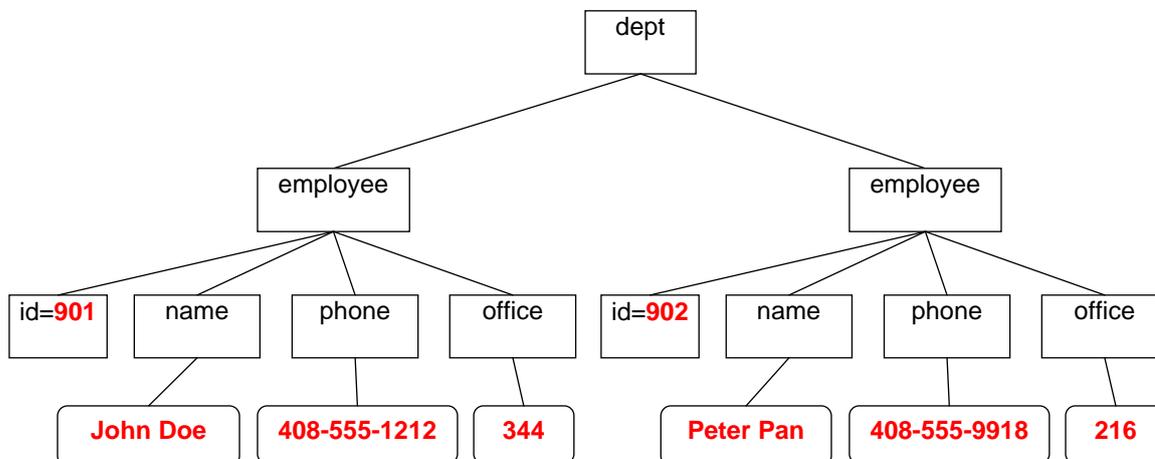
XPath è un linguaggio che può essere usato per eseguire query su documenti XML. La figura 16.8 mostra un documento XML, la figura 16.9 illustra lo stesso documento rappresentato in formato parsed-gerarchico (chiamato anche "nodo" o "foglia"). Useremo il formato parsed-gerarchico per spiegare come funziona XPath.

```

<dept bldg="101">
  <employee id="901">
    <name>John Doe</name>
    <phone>408 555 1212</phone>
    <office>344</office>
  </employee>
  <employee id="902">
    <name>Peter Pan</name>
    <phone>408 555 9918</phone>
    <office>216</office>
  </employee>
</dept>

```

Figure 16.8 – An XML document / Figura 16.8 - Un documento XML



**Figure 16.9 – Parsed-hierarchical representation of the XML document in Fig 16.8 /
Figura 16.9 - Rappresentazione parsed-gerarchica del documento XML della figura
16.8**

A quick way to learn XPath is to compare it to the change directory (`cd`) command in MS-DOS or Linux/UNIX. Using the `cd` command you traverse a directory tree as follows:

```
cd /directory1/directory2/...
```

Similarly, in XPath you use slashes to go from one element to another within the XML document. For example, using the document in Figure 16.9 in XPath you could retrieve the names of all employees using this query:

```
/dept/employee/name
```

XPath expressions / Espressioni XPath

XPath expressions use fully qualified paths to specify elements and/or attributes. An "@" sign is used to specify an attribute. To retrieve only the value (text node) of an element, use the "text()" function. Table 16.1 shows XPath queries and the corresponding results using the XML document from Figure 16.9.

Un modo rapido per imparare XPath è quello di confrontarlo con il comando per cambiare directory (`cd`) in MS-DOS o in Linux/Unix. Usando il comando `cd` si attraversa un albero di directory come segue:

Allo stesso modo, in XPath si usa lo slash per andare da un elemento ad un altro all'interno del documento XML. Per esempio, usando il documento della figura 16.9 in XPath potreste richiamare i nomi di tutti gli impiegati usando questa query:

Le espressioni XPath utilizzano path completi per specificare elementi e/o attributi. Il simbolo "@" si usa per specificare un attributo. Per richiamare soltanto il valore (nodo testo) di un elemento, si usa la funzione "text()". La tabella 16.1 mostra le query XPath ed i risultati corrispondenti usando il documento XML della figura 16,9.

| XPath | Result / Risultato |
|----------------------------|---|
| /dept/@bldg | 101 |
| /dept/employee/@id | 901 902 |
| /dept/employee/name | <name>Peter Pan</name> <name>John Doe</name> |
| /dept/employee/name/text() | Peter Pan John Doe |

**Table 16.1 – XPath expression examples / Tabella 16.1 - Esempi di espressioni XPath
XPath wildcards / Caratteri jolly di XPath**

There are two main wildcards in XPath:

- "*" matches any tag name
- "/" is the "descendent-or-self" wildcard

Ci sono due wildcard principali in XPath:

- "*" corrisponde a qualsiasi nome di tag
- "/" è la "descendent-or-self" wildcard

Table 16.2 provides more examples using the XML document from Figure 16.9.

La tabella 16.2 offre ulteriori esempi usando il documento XML della figura 16.9.

| XPath | Result / Risultato |
|-------------------------|---|
| /dept/employee/*/text() | John Doe 408 555 1212 344 Peter Pan 408 555 9918 216 |
| /dept*/@id | 901 902 |
| //name/text() | Peter Pan John Doe |
| /dept//phone | <phone>408 555 1212</phone> <phone>408 555 9918</phone> |

Table 16.2 – XPath wildcard examples / Tabella 16.2 - Esempi di wildcard di XPath

XPath predicates / Predicati di Xpath

Predicates are enclosed in square brackets []. As an analogy, you can think of them as the equivalent to the WHERE clause in SQL. For example [@id="902"] can be read as: "WHERE attribute id is equal to 902". There can be multiple predicates in one XPath expression. To specify a positional predicate, use [n] which means the nth child would be selected. For Example, employee[2] means that the second employee should be selected. Table 16.3 provides more examples.

I predicati sono racchiusi tra parentesi quadre []. Come analogia, si possono pensare come equivalenti alla clausola WHERE nell' SQL. Per esempio [@id="902"] può essere letto come: "WHERE attributo id uguale a 902" Ci possono essere predicati multipli in un'espressione XPath. Per specificare un predicato posizionale, usare [n] che significa la selezione dell'ennesimo figlio. Ad esempio, dipendente [2] significa che viene selezionato il secondo dipendente. Vedere la tabella 16.3 per ulteriori esempi.

| XPath | Result / Risultato |
|--|------------------------|
| /dept/employee[@id="902"]/name | <name>Peter Pan</name> |
| /dept[@bldg="101"]/employee[office >"300"]/name | <name>John Doe</name> |
| //employee[office="344" OR office="216"]/@id | 901 902 |
| /dept/employee[2]/@id | 902 |

Table 16.3 – XPath predicate examples / Tabella 16.3 - Esempi di predicati XPath

XPath: the parent axis / Xpath: l'asse padre

Similar to MS-DOS or Linux/UNIX, you can use a "." (dot) to indicate in the expression that you are referring to the current context, and a ".." (dot dot) to refer to the parent context. Table 16.4 provides more examples.

Analogamente a MS-DOS o Linux/UNIX, si può usare un "." (punto) nell'espressione per indicare che si sta facendo riferimento al contesto attuale, e un ".." (punto punto) per fare riferimento al contesto padre. La tabella 16.4 offre maggiori esempi.

| XPath | Result / Risultato |
|--|------------------------|
| /dept/employee/name[../@id="902"] | <name>Peter Pan</name> |
| /dept/employee/office[../>"300"] | <office>344</office> |
| /dept/employee[office > "300"]/office | <office>344</office> |

| | |
|--|-----|
| /dept/employee[name="John Doe"]/../@bldg | 101 |
| /dept/employee/name[.="John Doe"]/..../@bldg | 101 |

Table 16.4 – XPath parent axis / Tabella 16.4 - Asse padre in XPath

16.3.3 XQuery defined / XQuery definito

XQuery is a query language created for XML. XQuery supports path expressions to navigate XML hierarchical structure. In fact, XPath is a subset of XQuery; therefore, everything we learned earlier about XPath applies to XQuery too. XQuery supports both typed and untyped data. XQuery lacks null values because XML documents omit missing or unknown data. XQuery returns sequences of XML data. It's important to note that XQuery and XPath expressions are case sensitive.

XQuery è un linguaggio d'interrogazione creato per XML. XQuery supporta le espressioni dei path per navigare la struttura gerarchica di XML. Infatti, XPath è un sottoinsieme di XQuery; quindi, tutto ciò che abbiamo appreso in precedenza riguardo XPath si applica anche a XQuery. XQuery supporta sia dati tipizzati sia non tipizzati. XQuery non supporta i valori nulli, perché i documenti XML omettono dati mancanti o sconosciuti. XQuery restituisce sequenze di dati XML. È importante notare che le espressioni di XQuery e XPath sono case sensitive.

XQuery supports the FLWOR expression. If we use SQL for an analogy, it is equivalent to a SELECT-FROM-WHERE expression. The next section describes FLWOR in more detail.

XQuery supporta l'espressione FLWOR. Se usiamo SQL in analogia, è equivalente ad un'espressione SELECT-FROM-WHERE. La sezione seguente descrive FLWOR in maggiore dettaglio.

XQuery: FLWOR expression / Xquery: Espressione FLWOR

FLWOR stands for:

- FOR: iterates through a sequence, binds a variable to items
- LET: binds a variable to a sequence
- WHERE: eliminates items of the iteration
- ORDER: reorders items of the iteration
- RETURN: constructs query results

FLWOR sta per :

- FOR: itera attraverso una sequenza, assegna una variabile agli oggetti
- LET: lega una variabile ad una sequenza
- WHERE: elimina gli elementi dell'iterazione
- ORDER: riordina gli elementi dell'iterazione
- RETURN: costruisce i risultati della query

It is an expression that allows manipulation of XML documents, enabling you to return another expression. For example, assume you have a table with this definition:

È un'espressione che permette la manipolazione di documenti XML, permettendo di restituire un'altra espressione. Per esempio, supponiamo che si abbia una tabella con questa definizione:

```
CREATE TABLE dept (deptID CHAR(8), deptdoc XML);
```

And the following XML document is inserted in the deptdoc column:

E il seguente documento XML è inserito nella colonna deptdoc:

```
<dept bldg="101">
  <employee id="901">
    <name>John Doe</name>
    <phone>408 555 1212</phone>
    <office>344</office>
  </employee>
  <employee id="902">
    <name>Peter Pan</name>
    <phone>408 555 9918</phone>
    <office>216</office>
  </employee>
</dept>
```

Then the following XQuery statement using the FLWOR expression could be run:

Poi la seguente dichiarazione di XQuery usando l'espressione di FLWOR potrebbe essere eseguita:

```
xquery
for $d in db2-fn:xmlcolumn('dept.deptdoc')/dept
let $emp := $d//employee/name
where $d/@bldg > 95
order by $d/@bldg
return
  <EmpList>
    { $d/@bldg, $emp }
  </EmpList>
```

This would return the following:

Il risultato sarebbe il seguente:

```
<EmpList bldg="101">
  <name>
    John Doe
  </name>
  <name>
    Peter Pan
  </name>
</EmpList>
```

16.3.4 Inserting XML documents / Inserimento di documenti XML

Inserting XML documents into a DB2 database can be performed using the INSERT SQL statement, or the IMPORT utility. XQuery cannot be used for this purpose as this has not yet been defined in the standard.

L'inserimento di documenti XML in un database DB2 può essere effettuato usando la dichiarazione INSERT SQL, o l'utility IMPORT. XQuery non può essere usato per tale scopo poiché questo non è ancora stato definito nello standard.

Let's examine the following script, which can be run from the DB2 Command Window or Linux shell using this statement:

Esaminiamo il seguente script, che può essere eseguito dalla Command Window di DB2 o in una shell Linux usando questa dichiarazione:

```
db2 -tvf table_creation.txt
```

table_creation.txt

```
-- (1)
drop database mydb
;

-- (2)
create database mydb using codeset UTF-8 territory US
;

-- (3)
connect to mydb
;

-- (4)
create table items (
  id          int primary key not null,
  brandname   varchar(30),
  itemname    varchar(30),
  sku         int,
  srp         decimal(7,2),
  comments    xml
);

-- (5)
create table clients(
  id          int primary key not null,
  name        varchar(50),
  status      varchar(10),
  contact     xml
);

-- (6)
insert into clients values (77, 'John Smith', 'Gold',
```

```
'<addr>111 Main St., Dallas, TX, 00112</addr>')
;

-- (7)
IMPORT FROM "D:\Raul\clients.del" of del xml from "D:\Raul" INSERT
INTO CLIENTS (ID, NAME, STATUS, CONTACT)
;

-- (8)
IMPORT FROM "D:\Raul\items.del" of del xml from "D:\Raul" INSERT
INTO ITEMS (ID, BRANDNAME, ITEMNAME, SKU, SRP, COMMENTS)
;
```

Note that this script file and related files are provided in the zip file `expressc_book_quicklabs.zip` that accompanies this book. Follow along as we describe each line in the script.

1. Drop the database "mydb". This is normally done in script files to perform cleanup. If "mydb" didn't exist before, you will receive an error message, but this is OK.
2. Create the database "mydb" using the codeset UTF-8. A UNICODE database is required to support pureXML, so this step is necessary to create the database as a UNICODE database.
3. Connect to the newly created database "mydb". This is necessary to create objects within the database.
4. Create the table "items". Note that the last column in the table (column "comments") is defined as an XML column using the new XML data type.
5. We create the table "clients". Note that the last column in the table (column "contact") is also defined with the new XML data type.

Da notare che questo file script e i file correlati sono forniti nel file zip `expressc_book_quicklabs.zip` che accompagna questo libro. Segue una descrizione di ogni linea dello script.

1. Eseguire il drop del database "mydb". Questo è fatto normalmente attraverso il file di script per l'esecuzione del cleanup. Se "mydb" non esisteva, viene ricevuto un messaggio di errore, ma questo è accettabile.
2. Creare il database "mydb", usando il codeset UTF-8. Un database deve essere UNICODE per supportare il pureXML, questo punto è fondamentale per creare un database di tipo UNICODE.
3. Connettersi al nuovo database "mydb" appena creato. Questo è necessario per creare oggetti all'interno del database.
4. Creare la tabella "items". Notare che l'ultima colonna nella tabella (la colonna "comments") è definita come una colonna XML usando il nuovo tipo di dati XML.
5. Creare la tabella "clients". Notare che l'ultima colonna nella tabella (colonna "contatto") è definita con il nuovo tipo di dati XML.

- | | |
|--|--|
| <p>6. Using this SQL INSERT statement, you can insert an XML document into an XML column. In the INSERT statement you pass the XML document as a string enclosed in single quotes.</p> | <p>6. Usando questa dichiarazione SQL INSERT, si può inserire un documento XML in una colonna di tipo XML. Nella dichiarazione dell'INSERT passare il documento XML come una stringa racchiusa tra apici singoli .</p> |
| <p>7. Using an IMPORT command, you can insert or import several XML documents along relational data into the database. In (7) you are importing the data from the clients.del file (a delimited ascii file), and you also indicate where the XML data referenced by that clients.del file is located (for this example, in D:\Raul).</p> | <p>7. Usando un comando IMPORT, si possono inserire o importare parecchi documenti XML insieme ai dati relazionali nel database. Si importano i dati dal file clients.del (un file in formato ascii), e si indica anche dove i dati XML referenziati da quel file clients.del si trovano (per questo esempio, in D:\Raul).</p> |

We will take a more careful look at file clients.del, but first, let's see the contents of directory D:\Raul (Figure 16.10).

Prenderemo in esame più attentamente il file clients.del, ma prima, vediamo il contenuto della directory D:\Raul (Figura 16,10).

| Name | Size | Type |
|-----------------|------|--------------|
| Client3227.xml | 1 KB | XML Document |
| Client4309.xml | 1 KB | XML Document |
| Client5681.xml | 1 KB | XML Document |
| Client8877.xml | 1 KB | XML Document |
| Client9077.xml | 1 KB | XML Document |
| Client9177.xml | 1 KB | XML Document |
| ClientInfo.xsd | 2 KB | XML Schema |
| clients.del | 1 KB | DEL File |
| Comment3926.xml | 1 KB | XML Document |
| Comment4023.xml | 1 KB | XML Document |
| Comment4272.xml | 1 KB | XML Document |
| items.del | 1 KB | DEL File |

Figure 16.10 - Contents of D:\Raul directory with XML documents / Figura 16.10 - Il contenuto del directory D:\Raul con documenti XML

These are the contents of the file clients.del

Questo è il contenuto del file clients.del

clients.del

```
3227,Ella Kimpton,Gold,<XDS FIL='Client3227.xml' />,
8877,Chris Bontempo,Gold,<XDS FIL='Client8877.xml' />,
9077,Lisa Hansen,Silver,<XDS FIL='Client9077.xml' />,
9177,Rita Gomez,Standard,<XDS FIL='Client9177.xml' />,
5681,Paula Lipenski,Standard,<XDS FIL='Client5681.xml' />,
4309,Tina Wang,Standard,<XDS FIL='Client4309.xml' />
```

In the clients.del file, "XDS FIL=" is used to point to a specific XML document file.

Nel file clients.del, "XDS FIL=" è usato per puntare ad uno specifico file di documento XML.

Figure 16.11 shows the Control Center after running the above script.

La figura 16.11 mostra il Control Center dopo aver eseguito lo script qui sopra.

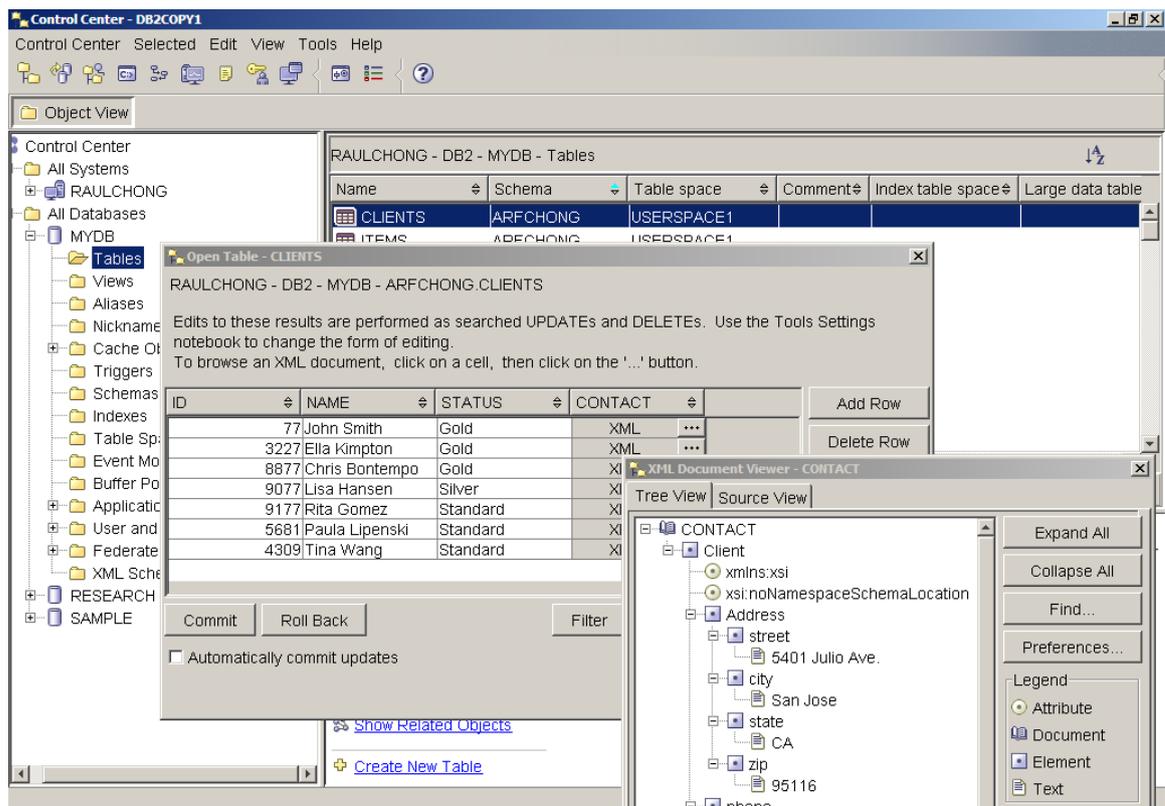


Figure 16.11 – The Control Center after running table_creation.txt / Figura 16.11 - II Control Center dopo aver eseguito table_creation.txt

Note that in the figure, we show the contents of the CLIENTS table. The last column “Contact” is an XML column. When you click on the button with three dots, another window opens showing you the XML document contents. This is shown in the bottom right corner of the figure 16.11.

Notare che nella figura, si mostra il contenuto della tabella CLIENTI. L'ultima colonna "Contact" è una colonna di tipo XML. Quando si fa clic sul tasto con i tre punti, un'altra finestra si apre che mostra il contenuto del documento XML. Questo è mostrato nell'angolo in basso a destra della figura 16.11.

16.3.5 Querying XML data / Query su dati XML

There are two ways to query XML data in DB2:

- Using SQL with XML extensions (SQL/XML)
- Using XQuery

In both cases, DB2 follows international XML standards.

Ci sono due modi per fare query in DB2 su dati XML:

- Usando SQL con le estensioni XML (SQL/XML)
- Usando XQuery

In entrambi i casi, DB2 segue gli standard internazionali di XML.

Querying XML Data with SQL/XML / Query su dati XML con SQL/XML

Using plain SQL statements allows you to work with rows and columns. An SQL statement can be used to work with full XML documents; however, it would not help when attempting to retrieve only part of the document. In such cases, you need to use SQL with XML extensions (SQL/XML).

Usando semplici istruzioni SQL permette di lavorare con righe e colonne. Una dichiarazione SQL può essere utilizzata per lavorare con documenti XML; tuttavia, non aiuterebbe quando si tenta di richiamare soltanto una parte del documento. In questi casi, bisogna usare SQL con le estensioni di XML (SQL/XML).

Table 16.5 describes some of the SQL/XML functions available with the SQL 2006 standard.

La tabella 16.5 descrive alcune delle funzioni SQL/XML disponibili con lo standard SQL 2006.

| Function name / Nome della funzione | Description | Descrizione |
|-------------------------------------|--|--|
| XMLPARSE | Parses character or large object binary data, produces XML value | Fa il parsing di caratteri o dati binari di grandi dimensioni, producendo dati XML |
| XMLSERIALIZE | Converts an XML value into character or large object binary data | Converte un valore XML in caratteri o dati binari di grandi dimensioni |

| | | |
|-------------|---|---|
| XMLVALIDATE | Validates XML value against an XML schema and type-annotates the XML value | Valida il valore XML rispetto a uno schema XML e annotata il tipo di valore XML |
| XMLEXISTS | Determines if an XQuery returns a results (i.e. a sequence of one or more items) | Determina se una XQuery ritorna risultati (i.e. una sequenza di uno o più elementi) |
| XMLQUERY | Executes an XQuery and returns the result sequence | Esegue una XQuery e ritorna la sequenza di risultati |
| XMLTABLE | Executes an XQuery, returns the result sequence as a relational table (if possible) | Esegue una XQuery, ritorna la sequenza di risultati come una tabella relazionale (se possibile) |
| XMLCAST | Cast to or from an XML type | Fa il cast da o a un tipo di dato XML |

Table 16.5 – SQL/XML Functions / Tabella 16.5 - Funzioni SQL/XML

The following examples can be tested using the “mydb” database created earlier.

Example 1

This is an example of a sample query problem. Imagine that you need to locate the names of all clients who live in a specific zip code. The “clients” table stores customer addresses, including zip codes, in an XML column. Using XMLEXISTS, you can search the XML column for the target zip code and then restrict the return result set accordingly.

```
SELECT name FROM clients
WHERE xmlexists(
  '$c/Client/Address[zip="95116"]'
  passing clients.contact as "c"
)
```

The first line is an SQL clause specifying that you want to retrieve information in the “name” column of the “clients” table.

The WHERE clause invokes the XMLEXISTS function, specifying the XPath expression that prompts DB2 to

Gli esempi seguenti possono essere esaminati usando il database “mydb” creato in precedenza.

Esempio 1

Questo è un esempio di un problema di una tipica query. Immaginate di dover individuare i nomi di tutti i clienti che vivono ad un determinata area postale. La tabella “clients” registra gli indirizzi dei clienti, compresi i codici postali, in una colonna di tipo XML. Usando XMLEXISTS, si può cercare la colonna di tipo XML per il codice postale desiderato e successivamente restringere il set di risultati ritornati di conseguenza.

La prima riga è una clausola di SQL che specifica il fatto che si vogliono richiamare le informazioni nella colonna “nome” della tabella “clients”.

La clausola WHERE invoca la funzione XMLEXISTS, specificando l'espressione XPath che richiede a DB2 di navigare

navigate to the “zip” element and check for a value of 95116

The “\$c/Client/Address” clause indicates the path inside the XML document hierarchy where DB2 can locate the “zip” element. A dollar sign (\$) is used to specify a variable; therefore “c” is a variable. This variable is then defined by this line: `passing clients.contact as "c"`. Here, “clients” is the name of the table and “contact” is the name of the column with an XML data type. In other words, we are passing the XML document to the variable “c”.

DB2 inspects the XML data contained in the “contact” column, navigates from the root “Client” node down to the “Address” node and then to the “zip” node and finally determines if the customer lives in the target zip code. The `XMLEXISTS` function evaluates to “true” and DB2 returns the name of the client associated with that row.

In DB2 9.5, the above query could be simplified as follows:

```
SELECT name FROM clients
WHERE xmlexists(
  '$CONTACT/Client/Address[zip="95116"]'
)
```

A variable with the same name as an XML column is created automatically by DB2. In the above example, the variable `CONTACT` is created automatically by DB2. Its name matches the name of the XML column `CONTACT`.

Example 2

Let’s consider how to solve the problem of how to create a report listing the e-mail addresses of “Gold” status

l'elemento "zip" e di controllare se c'è un valore di 95116

La clausola "\$c/Client/Address" indica il path all'interno della struttura gerarchica del documento XML dove DB2 può localizzare l'elemento "zip". Il segno dollaro (\$) è usato per specificare una variabile; quindi "c" è una variabile. Questa variabile è poi definita da questa riga: `passing clients.contact as "c"`. Qui, "clients" è il nome della tabella e "contact" è il nome della colonna con un tipo di dati XML. In altre parole, stiamo passando il documento XML alla variabile "c".

DB2 esamina i dati XML contenuti nella colonna "contact", naviga dal nodo radice "Client" fino al nodo "Address" e poi al nodo "zip", ed infine determina se il cliente abita nell'area del codice postale richiesto. La funzione `XMLEXISTS` valuta come "true" e DB2 ritorna il nome del cliente associato a quella riga.

In DB2 9.5, la query precedente potrebbe essere semplificata come segue:

Una variabile con lo stesso nome di una colonna XML è creata automaticamente da DB2. Nell'esempio precedente, la variabile `CONTACT` è creata automaticamente da DB2. Il suo nome corrisponde al nome della colonna XML `CONTACT`.

Esempio 2

Consideriamo come risolvere il problema di creare un report che elenchi gli indirizzi di posta elettronica di

customers. The following query could be run:

```
SELECT xmlquery('$c/Client/email' passing contact as "c")
FROM clients
WHERE status = 'Gold'
```

The first line indicates we want to return the email address which is an element of the XML document (not a relational column). As in the previous example, "\$c" is a variable that contains the XML document. In this example we use the XMLQUERY function which can be used after a SELECT, while the XMLEXISTS function can be used after a WHERE clause.

Example 3

There may be situations when you would like to present XML data as tables. This is possible with the XMLTABLE function as shown in the example below.

```
SELECT t.comment#, i.itemname, t.customerID, Message
FROM items i,
xmltable('$c/Comments/Comment' passing i.comments as "c"
columns Comment# integer path 'CommentID',
CustomerID integer path 'CustomerID',
Message varchar(100) path 'Message') AS t
```

The first line specifies the columns to be included in your results set. Columns prefixed with the "t" variable are based on XML element values.

The second line invokes XMLTABLE function to specify the DB2 XML column containing the target data ("i.comments") and the path within the column's XML documents where the elements of interest are located.

The "columns" clause, spanning lines 4 to 6, identifies the specific XML elements that will be mapped to output columns in the SQL result set specified on line 1. Part of this mapping involves

clienti con status "Gold". La seguente query potrebbe essere eseguita:

La prima riga indica che vogliamo ritornare l'indirizzo e-mail che è un elemento del documento XML (non una colonna relazionale). Come nell'esempio precedente, "\$c" è una variabile che contiene il documento XML. In questo esempio usiamo la funzione XMLQUERY che può essere usata dopo una SELECT, mentre la funzione XMLEXISTS può essere usata dopo una clausola WHERE.

Esempio 3

Ci possono essere situazioni in cui si vuole presentare i dati XML come tabelle. Ciò è possibile con la funzione XMLTABLE come è mostrato nell'esempio qui sotto.

La prima riga indica le colonne che saranno incluse nel set dei risultati. Colonne con prefisso "t" sono basate sui valori dell'elemento XML.

La seconda riga richiama la funzione XMLTABLE per specificare la colonna DB2 XML contenente i dati target ("i.comments") ed il path all'interno della colonna dei documenti XML dove gli elementi di interesse sono situati.

La clausola "columns", che copre le linee da 4 a 6, identifica gli elementi specifici di XML che verranno associate a colonne di output nel set di risultati di SQL specificati nella linea 1. Parte di

specifying the data types to which the XML element values will be converted. In this example all XML data is converted to traditional SQL data types.

Example 4

Now let's explore a simple example in which you include an XQuery FLWOR expression inside an XMLQUERY SQL/XML function.

```
SELECT name, xmlquery(
  'for $e in $c/Client/email[1] return $e'
  passing contact as "c"
)
FROM clients
WHERE status = 'Gold'
```

The first line specifies that the customer names and the output from the XMLQUERY function will be included in the result set. The second line indicates the first "email" sub-element of the "Client" element is to be returned. The third line identifies the source of our XML data ("contact" column). The fourth line tells us that this column is coming from the "clients" table; and the fifth line indicates that only "Gold" customers are of interest.

Example 5

This example again demonstrates the XMLQUERY function which takes an XQuery FLWOR expression, but note that this time we are returning not only XML, but also HTML.

```
SELECT xmlquery('for $e in $c/Client/email[1]/text()
  return <p>{$e}</p>'
  passing contact as "c")
FROM clients
WHERE status = 'Gold'
```

questa mappatura richiede di specificare i tipi di dato a cui i valori dell'elemento XML saranno convertiti. In questo esempio tutti i dati XML sono convertiti nei tradizionali tipi di dato SQL.

Esempio 4

Adesso esploriamo un semplice esempio nel quale si include un'espressione di XQuery FLWOR all'interno di una funzione XMLQUERY SQL/XML.

La prima riga specifica che i nomi dei clienti e dell'output dalla funzione XMLQUERY saranno inclusi nel set dei risultati. La seconda riga indica il primo sotto-elemento "email" dell'elemento "Client" che deve essere ritornato. La terza riga identifica la fonte dei nostri dati XML (la colonna "contatto"). La quarta riga ci dice che questa colonna proviene dalla tabella "clients"; e la quinta riga indica che soltanto i clienti "Gold" sono rilevanti.

Esempio 5

Questo esempio dimostra ancora una volta la funzione XMLQUERY che riprende un'espressione XQuery FLWOR, ma è da notare che questa volta stiamo ritornando non solo XML, ma anche HTML.

The return clause of XQuery enables you to transform XML output as needed. Using the text() function in the first line indicates that only the text representation of the first email address of qualifying customers is of interest. The second line specifies that this information is to be surrounded by HTML paragraph tags.

La clausola di ritorno di XQuery permette di trasformare l'output XML secondo le proprie esigenze. Usando la funzione text() nella prima riga indica che soltanto la rappresentazione testuale del primo indirizzo di e-mail dei clienti qualificati è rilevante. La seconda riga specifica che tali informazioni devono essere incluse all'interno del tag di paragrafo dell'HTML.

Example 6

The following example uses the XMLELEMENT function to create a series of item elements, each of which contain sub-elements for the ID, brand name, and stock keeping unit (SKU) values obtained from corresponding columns in the "items" table. Basically, you can use the XMLELEMENT function when you want to convert relational data to XML data.

Esempio 6

Il seguente esempio usa la funzione XMLELEMENT per creare una serie di elementi item, di cui ciascuno contiene sotto-elementi per l'ID, la marca, e i valori del stock keeping unit (SKU) ottenuti dalle corrispondenti colonne nella tabella "items". In sostanza, si può usare la funzione XMLELEMENT quando si vogliono convertire dati relazionali in dati XML.

```
SELECT
  xmlelement (name "item", itemname),
  xmlelement (name "id", id),
  xmlelement (name "brand", brandname),
  xmlelement (name "sku", sku)
FROM items
WHERE srp < 100
```

The above query would be provide output like this:

La query di sopra fornirebbe un output simile a questo:

```
<item>
  <id>4272</id>
  <brand>Classy</brand>
  <sku>981140</sku>
</item>
...
<item>
  <id>1193</id>
  <brand>Natural</brand>
  <sku>557813</sku>
</item>
```

Query XML Data with Xquery / Query di dati XML con XQuery

In the previous section, we looked at how to query XML data using SQL with XML extensions. SQL was always the primary query method, and XPath was embedded inside SQL. In this section, we discuss how to query XML data using XQuery. This time, XQuery will be the primary query method, and in some cases, we will use SQL embedded inside XQuery (using the "db2-fn:sqlquery" function). When using XQuery, we will invoke a few functions, and will also use the FLWOR expression.

Nella sezione precedente, abbiamo esaminato come eseguire query su dati XML usando SQL con le estensioni di XML. SQL era sempre il metodo di query principale, e XPath è stato integrato all'interno di SQL. In questa sezione, discutiamo come eseguire query su dati XML usando XQuery. Questa volta, XQuery sarà il metodo principale di query, ed in alcuni casi, useremo SQL integrato all'interno di XQuery (usando la funzione "db2-fn:sqlquery"). Quando useremo XQuery, invocheremo alcune funzioni, ed inoltre utilizzeremo anche l'espressione FLWOR.

Example 1

A simple XQuery to return customer contact data

```
xquery db2-fn:xmlcolumn('CLIENTS.CONTACT')
```

Always prefix any XQuery expression with the `xquery` command so that DB2 knows it has to use the XQuery parser, otherwise DB2 will assume you are trying to run an SQL expression. The `db2-fn:xmlcolumn` function is a function that retrieves the XML documents from the column specified as the parameter. It is equivalent to the following SQL statement, as it is retrieving the entire column contents:

```
SELECT contact FROM clients
```

Example 2

In this example, we use the FLWOR expression to retrieve client fax data

```
xquery
  for $y in db2-fn:xmlcolumn('CLIENTS.CONTACT')/Client/fax
  return $y
```

Esempio 1

Una semplice XQuery per ritornare dati sui contatto dei clienti

Usare sempre il comando `xquery` come prefisso di qualsiasi espressione di XQuery in modo tale che DB2 sappia di dover usare il parser XQuery, altrimenti DB2 assumerà che stiate provando ad eseguire un'espressione SQL. La funzione `db2-fn:xmlcolumn` è una funzione che richiama i documenti XML dalla colonna specificata come il parametro. E' equivalente alla seguente istruzione di SQL, che richiama l'intero contenuto della colonna:

Esempio 2

In questo esempio, usiamo l'espressione FLWOR per richiamare i dati del fax del cliente:

The first line invokes the XQuery parser. The second line instructs DB2 to iterate through the fax sub-elements contained in the CLIENTS.CONTACT column. Each fax element is bound to the variable \$y. The third line indicates that for each iteration, the value "\$y" is returned.

La prima riga invoca il parser XQuery. La seconda riga incarica DB2 di iterare attraverso i sotto-elementi del fax contenuti nella colonna CLIENTS.CONTACT. Ogni elemento del fax è legato alla variabile \$y. La terza riga indica che per ogni iterazione, il valore "\$y" viene ritornato.

The output of this query is similar to this one (it may include the namespace by default, but we don't show it below, otherwise this output would be harder to read as it may span several lines):

L'output di questa query è simile a questo (potrebbe includere il namespace di default, ma non viene mostrato qui sotto, altrimenti l'output sarebbe difficile da leggere poiché occuperebbe parecchie righe):

```
<fax>4081112222</fax>
<fax>5559998888</fax>
```

Example 3

This next example queries XML data and returns the results as HTML.

Esempio 3

Il prossimo esempio interroga i dati XML e ritorna i risultati in formato HTML.

```
xquery
  <ul> {
    for $y in db2-fn:xmlcolumn('CLIENTS.CONTACT')/Client/Address
    order by $y/zip
    return <li>{$y}</li>
  }
</ul>
```

The sample HTML returned looks like this:

Il sample HTML ritornato assomiglia a questo:

```
<ul>
<li>
<address>
  <street>9407 Los Gatos Blvd.</street>
  <city>Los Gatos</city>
  <state>ca</state>
  <zip>95302</zip>
</address>
</li>
<address>
<street>4209 El Camino Real</street>
  <city>Mountain View</city>
  <state>CA</state>
  <zip>95302</zip>
</address>
```

```
</li>
...
</ul>
```

Example 4

The following example shows how to embed SQL within XQuery by using the db2-fn:sqlquery function. The db2-fn:sqlquery function executes an SQL query and returns only the selected XML data. The SQL query passed to db2-fn:sqlquery must only return XML data. This XML data can then be further processed by Xquery.

```
xquery
  for $y in
    db2-fn:sqlquery(
      'select comments from items where srp > 100'
    )/Comments/Comment
  where $y/ResponseRequested='Yes'
  return (
    <action>
      {
        $y/ProductID
        $y/CustomerID
        $y/Message
      }
    </action>
  )
```

In the example, the SQL query filters rows based on the condition that the "srp" column has a value greater than 100. From those rows filtered, it will pick the "comments" column, which is the XML column. Next XQuery (or XPath) is applied to go to sub-elements.

Note: DB2 is case insensitive and treats all table and column names in uppercase while XQuery is case sensitive. The above resource functions are XQuery interface functions so all the table names and column names should be passed to these function in uppercase. Passing the object names in lowercase may result in an undefined object name error.

Esempio 4

Il seguente esempio mostra come inserire SQL all'interno di XQuery usando la funzione db2-fn:sqlquery. La funzione db2-fn:sqlquery esegue una query SQL e ritorna soltanto i dati XML selezionati. La query SQL passata a db2-fn:sqlquery deve ritornare soltanto dati XML. Questi dati XML possono poi essere processati ulteriormente da Xquery.

Nell'esempio, la query SQL filtra le righe in base alla condizione che la colonna "srp" abbia un valore più grande di 100. Da quelle righe filtrate, selezionerà la colonna "comments", che è la colonna XML. La prossima Xquery (o XPath) è applicata per visitare sotto-elementi.

Nota: DB2 è case insensitive e tratta tutti i nomi di colonne e tabelle in maiuscolo, mentre XQuery è case sensitive. Le funzioni-risorsa qui sopra sono funzioni dell'interfaccia di XQuery così tutti i nomi delle tabelle e delle colonne dovrebbero essere passati a queste funzioni in maiuscolo. Passando i nomi degli oggetti in minuscolo può accadere un errore del tipo undefined object name.

16.3.6 Joins with SQL/XML / Join con SQL/XML

This section describes how to perform JOIN operations between two XML columns of different tables, or between one XML column and one relational column. Assume you have created two tables with these statements:

```
CREATE TABLE dept (unitID CHAR(8), deptdoc XML)

CREATE TABLE unit (unitID CHAR(8) primary key not null,
                   name CHAR(20),
                   manager VARCHAR(20),
                   ...
                   )
```

You can perform a JOIN operation in either of two ways:

Method 1:

```
SELECT u.unitID
FROM dept d, unit u
WHERE XMLEXISTS (
  '$e//employee[name = $m]'
  passing d.deptdoc as "e", u.manager as "m")
```

In line 3 of this statement shows that the JOIN operation occurs between the element "name" which is a sub-element of the "deptdoc" XML column in table "dept", and the "manager" relational column in the table "unit".

Method 2:

```
SELECT u.unitID
FROM dept d, unit u
WHERE u.manager = XMLCAST(
  XMLQUERY('$e//employee/name '
  passing d.deptdoc as "e")
  AS char(20))
```

In this alternate method, the relational column is on the left side of the JOIN. If the relational column is on the left side of the equation, a relational index may be used instead of an XML index.

Questa sezione descrive come eseguire operazioni di JOIN tra due colonne XML di tabelle diverse, o tra una colonna XML e una colonna relazionale. Assumete di aver creato due tabelle con queste dichiarazioni:

Si può eseguire un'operazione di JOIN in qualunque dei due seguenti modi:

Metodo 1:

Nella riga 3 di questa dichiarazione viene mostrato come l'operazione di JOIN avviene tra il "nome" dell'elemento, che è un sotto-elemento della colonna XML "deptdoc" nella tabella "dept", e la colonna relazionale "manager" nella tabella "unit".

Metodo 2:

Con questo metodo alternativo, la colonna relazionale è sul lato sinistro del JOIN. Se la colonna relazionale è sul lato sinistro dell'equazione, un indice relazionale può essere usato invece di un indice XML.

16.3.7 Joins with Xquery / Join con XQuery

Assume the following tables have been created:

```
CREATE TABLE dept(unitID CHAR(8), deptdoc XML)
CREATE TABLE project(projectDoc XML)
```

If we use SQL/XML, the query would look as follows:

```
SELECT XMLQUERY (
  '$d/dept/employee' passing d.deptdoc as "d")
FROM dept d, project p
WHERE XMLEXISTS (
  '$e/dept [@deptID=$p/project/deptID]'
  passing d.deptdoc as "e", p.projectDoc as "p")
```

The equivalent query using XQuery would be:

```
xquery
for $dept in db2-fn:xmlcolumn("DEPT.DEPTDOC")/dept
for $proj in db2-fn:xmlcolumn("PROJECT.PROJECTDOC")/project
where $dept/@deptID = $proj/deptID
return $dept/employee
```

This second method is easier to interpret -- variable "\$dept" holds the XML document of the XML column "deptdoc" in table "dept". The variable "\$proj" holds the XML document of the XML column "projectdoc" in table "project". Then line 4 performs the JOIN operation between an attribute of the first XML document and an element of the second XML document.

Si assumo che le seguenti tabelle siano state create:

Se usiamo SQL/XML, la query assomiglierebbe alla seguente:

La query equivalente usando XQuery potrebbe essere:

Questo secondo metodo è più facile da interpretare – la variabile "\$dept" contiene il documento XML della colonna XML "deptdoc" nella tabella "dept". La variabile "\$proj" contiene il documento XML della colonna XML "projectdoc" nella tabella "project". Poi la riga 4 esegue l'operazione di JOIN tra un attributo del primo documento XML e un elemento del secondo documento XML.

16.3.8 Update and delete operations / Operazioni di update e delete

Update and delete operations on XML data can be performed in one of two ways:

- ▶ Using SQL UPDATE and DELETE statements
- ▶ Using the TRANSFORM expression

For the first way using SQL UPDATE and DELETE statements, the update or delete occurs at the document level; that is, the entire XML document is replaced with the updated one. For example, if in the example below all we'd like to change is the <state> element, the entire XML document is actually replaced.

```
UPDATE clients SET contact=(
  xmlparse(document
    '<Client>
      <address>
        <street>5401 Julio ave.</street>
        <city>San Jose</city>
        <state>CA</state>
        <zip>95116</zip>
      </address>
      <phone>
        <work>4084633000</work>
        <home>4081111111</home>
        <cell>4082222222</cell>
      </phone>
      <fax>4087776666</fax>
      <email>newemail@someplace.com</email>
    </Client>')
  WHERE id = 3227
```

For the second way, you can perform sub-document updates using the TRANSFORM expression, which is a lot more efficient. This allows you to replace, insert, delete or rename nodes in an XML document. You can also change the value of a node without

Le operazioni di update e delete su dati XML possono essere eseguite in due modi:

- ▶ Usando le dichiarazioni SQL UPDATE e DELETE
- ▶ Usando l'espressione TRANSFORM

Il primo modo implica l'uso delle dichiarazioni SQL UPDATE e SQL DELETE, l'aggiornamento o la cancellazione avviene al livello del documento; cioè l'intero documento XML è sostituito con quello aggiornato. Ad esempio, se nell'esempio qui sotto volessimo cambiare soltanto l'elemento <state>, l'intero documento XML sarebbe effettivamente sostituito.

Per il secondo modo, si possono eseguire gli aggiornamenti del sub-documento usando l'espressione TRANSFORM, che è molto più efficiente. Ciò ci permette di sostituire, inserire, cancellare o rinominare i nodi in un documento XML. Si può anche

replacing the node itself, typically to change an element or attribute value—which is a very common type of update. This is new with DB2 9.5.

modificare il valore di un nodo senza sostituire il nodo stesso, tipicamente per cambiare un elemento o un valore dell'attributo—che è un tipo di update molto comune. Questo è una novità in DB2 9.5.

The TRANSFORM expression is part of the XQuery language, you can use it anywhere you normally use XQuery, for example in a FLWOR expression or in the XMLQUERY function in an SQL/XML statement. The most typical use is in an SQL UPDATE statement to modify an XML document in an XML column.

L'espressione TRANSFORM fa parte del linguaggio XQuery, si può usare dovunque si usi normalmente XQuery, per esempio, in un'espressione FLWOR o nella funzione XMLQUERY in una dichiarazione SQL/XML. L'uso più tipico è in una dichiarazione SQL UPDATE per modificare un documento XML in una colonna di XML.

Here is the syntax of the TRANSFORM expression:

Ecco la sintassi dell'espressione TRANSFORM:

```
>>-transform--| copy clause |--| modify clause |--| return clause
|-><
```

copy clause

```
      .-'.-----'.
      v                                     |
|--copy---$VariableName---:---CopySourceExpression-+-----|
```

modify clause

```
|--modify--ModifyExpression-----|
```

return clause

```
|--return--ReturnExpression-----|
```

The `copy` clause is used to assign to a variable the XML documents you want to process. In the `modify` clause, you can invoke an `insert`, `delete`, `rename`, or `replace` expression. These expressions allow you to perform updates to your XML document. For example, if you want to add new nodes to the document, you would use the `insert` expression, to delete nodes from an XML document, use the

La clausola `copy` è usata per assegnare ad una variabile i documenti XML che si vogliono trattare. Nella clausola `modify`, si può invocare l'espressione di `insert`, `delete`, `rename`, o `replace`. Queste espressioni permettono di eseguire gli aggiornamenti al documento XML. Per esempio, se si vogliono aggiungere nuovi nodi nel documento, useremo l'espressione `insert`, per eliminare i nodi di un documento XML, useremo

delete expression, to rename an element or attribute in the XML document, use the rename expression, and to replace an existing node with a new node or sequence of nodes, use the replace expression. The replace value of the expression can only be used to change the value of an element or attribute. The return clause returns the result of the transform expression.

l'espressione delete, per rinominare un elemento o un attributo nel documento XML, useremo l'espressione rename, e per sostituire un nodo esistente con un nuovo nodo o una sequenza di nodi, useremo l'espressione replace. Il valore di sostituzione dell'espressione può essere usato soltanto per cambiare il valore di un elemento o il valore di un attributo. La clausola di return ritorna il risultato dell'espressione transform.

Here is an example of an UPDATE statement using the TRANSFORM expression.

Ecco un esempio di una dichiarazione UPDATE usando l'espressione TRANSFORM.

```
(1)-- UPDATE customers
(2)-- SET contactinfo = xmlquery( 'declare default element namespace
(3)--                               "http://posample.org";
(4)--     transform
(5)--     copy $newinfo := $c
(6)--         modify do insert <email2>my2email.gm.com</email2>
(7)--             as last into $newinfo/customerinfo
(8)--     return $newinfo' passing contactinfo as "c")
(9)-- WHERE id = 100
```

In the above example, lines (1), (2), and (9) are part of the SQL UPDATE syntax. In Line (2) the XMLQUERY function is invoked, which calls the transform expression in line (4). The transform expression block goes from line (4) to line (8), and it is used to insert a new node into the XML document containing the email2 element. Note that updating the elements in an XML document through a view is not supported.

Nell'esempio precedente, le righe (1), (2), e (9) fanno parte della sintassi SQL UPDATE. Nella riga (2) la funzione XMLQUERY viene invocata, questa chiama l'espressione di trasformazione nella riga (4). Il blocco dell'espressione di trasformazione va dalla riga (4) alla riga (8), ed è usato per inserire un nuovo nodo nel documento XML che contiene l'elemento email2. Notare che l'update degli elementi in un documento XML attraverso una view non è supportato.

Deleting entire XML documents from tables is as simple as when using the SELECT statement in SQL/XML. Use the SQL DELETE statement and specify any necessary WHERE predicates needed.

L'eliminazione di interi documenti XML dalle tabelle è semplice come quando si usa uno statement SELECT in SQL/XML. Usa l'istruzione SQL DELETE e specifica tutti i predicati WHERE necessari.

16.3.9 XML indexing / L'indicizzazione XML

In an XML document, indexes can be created for elements, attributes, or for values (text nodes). Below are some examples. Assume the table below was created:

```
CREATE TABLE customer(info XML)
```

And assume this was one of the XML documents stored:

```
<customerinfo Cid="1004">
  <name>Matt Foreman</name>
  <addr country="Canada">
    <street>1596 Baseline</street>
    <city>Toronto</city>
    <state>Ontario</state>
    <pcode>M3Z-5H9</pcode>
  </addr>
  <phone type="work">905-555-4789</phone>
  <phone type="home">416-555-3376</phone>
  <assistant>
    <name>Peter Smith</name>
    <phone type="home">416-555-3426</phone>
  </assistant>
</customerinfo>
```

1) This statement creates an index on the attribute "Cid"

```
CREATE UNIQUE INDEX idx1 ON customer(info)
GENERATE KEY USING
xmlpattern '/customerinfo/@Cid'
AS sql DOUBLE
```

2) This statement creates an index on the element "name":

```
CREATE INDEX idx2 ON customer(info)
GENERATE KEY USING
xmlpattern '/customerinfo/name'
AS sql VARCHAR(40)
```

3) This statement creates an index on all elements "name":

In un documento XML, gli indici possono essere creati per gli elementi, per gli attributi o per i valori (text nodes). Di seguito vi sono alcuni esempi. Si assuma che la tabella seguente sia stata creata:

Ed inoltre si assuma che il seguente sia uno dei documenti XML salvati:

1) Questa istruzione crea un indice sull'attributo "Cid"

2) Questa istruzione crea un indice sull'elemento "name":

3) Questa istruzione crea un indice su tutti gli elementi "name":

```
CREATE INDEX idx3 ON customer(info)
  GENERATE KEY USING
  xmlpattern '//name'
  AS sql VARCHAR(40);
```

4) This statement creates an index on all text nodes (all values). This is not recommended, as it would be too expensive to maintain the index for update, delete and insert operations, and the index would be too large:

4) Questa istruzione crea un indice su tutti i text nodes (tutti i valori). Questo non è raccomandabile, poiché sarebbe troppo costoso mantenere l'indice per le operazioni update, delete e insert e l'indice sarebbe troppo grande:

```
CREATE INDEX idx4 ON customer(info)
  GENERATE KEY USING
  xmlpattern '//text()'
  AS sql VARCHAR(40);
```

QuickLab #14 - SQL/XML and Xquery / SQL/XML e XQueryObjective

You have seen several examples of SQL/XML and XQuery syntax and have been introduced to the DB2 Command Editor and IBM Data Studio. In this lab, you will test your SQL/XML and XQuery knowledge while gaining experience with these tools. We will use the “mydb” database created using the **table_creation.txt** script file which was explained earlier in this chapter.

Procedure

1. Create the “mydb” database and load the XML data, as discussed earlier in the chapter.
2. Using either the Command Editor or IBM Data Studio:
 1. Retrieve all the comments XML documents from the ITEMS table in two ways, but only using Xquery
 2. Why would issuing this SQL statement not return the exact same output?:

```
SELECT comments
  FROM items
```
 - c) Retrieve the ID and BRANDNAME for the records whose XML documents have a ResponseRequested element with a value of “No”

Obiettivo

Si sono visti diversi esempi di sintassi di XQuery e di SQL/XML ed è stato presentato il Command Editor di DB2 e l'IBM Data Studio. In questo laboratorio, metterai alla prova la tua conoscenza di XQuery e di SQL/XML acquisendo nel frattempo l'esperienza con questi tool. Useremo il database "mydb" creato usando il file di script table_creation.txt che è stato spiegato precedentemente in questo capitolo.

Procedura

1. Crea il database "mydb" e carica i dati XML, come discusso in precedenza nel capitolo.
2. Usando o il Command Editor o l'IBM Data Studio:
 - a) Recupera tutti i commenti sui documenti XML dalla tabella ITEMS in due modi, ma soltanto usando Xquery
 - b) Perché l'utilizzo di questo statement SQL non ritornerebbe lo stesso identico output?

```
SELECT comments
  FROM items
```
 - c) Recupera l'ID ed il BRANDNAME per i record per i quali i documenti XML hanno un elemento di Response Requested con il valore "No"

SOLUTIONS / SOLUZIONE:

2a)

```
xquery db2-fn:xmlcolumn('ITEMS.COMMENTS')
xquery db2-fn:sqlquery("select comments from items")
```

2b)

The output is different because SQL returns NULL values when a value is not present, while XQuery doesn't return anything. /

L'output è diverso perché l'SQL ritorna valori NULL quando un valore non è presente, mentre XQuery non ritorna niente.

2c)

```
SELECT id, brandname FROM items WHERE
XMLEXISTS('$c/Comments/Comment [ResponseRequested="No"] '
passing ITEMS.COMMENTS as "c
```

17

Chapter 17 –Developing with Java, PHP, and Ruby / Capitolo 17 - Sviluppare con Java, PHP, e Ruby

This chapter discusses the basics of application development in Java, PHP and Ruby on Rails using a DB2 server. The purpose of this chapter is not to teach these languages, but to provide information pertinent to using them with DB2.

Questo capitolo discute i principi dello sviluppo di applicazioni in Java, PHP ed Ruby on Rails usando un server DB2. Lo scopo di questo capitolo non è di insegnare questi linguaggi, ma di fornire informazioni pertinenti al loro utilizzo con DB2.

Note:

For more information about this topic, watch this video:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4402>

Nota:

Per ulteriori informazioni su questo argomento, guardi questo video:

<http://www.channeldb2.com/video/video/show?id=807741:Video:4402>

17.1 Application development in Java / Sviluppo di applicazioni in Java

The IBM DB2 driver for JDBC (also known as the JCC driver) has been optimized for all DB2 servers in all platforms. The jar file db2jcc.jar (com.ibm.db2.jcc) includes the type 2 and type 4 drivers. The db2jcc.jar file is included with any DB2 client, or can be obtained on its own (IBM DB2 Driver for JDBC and SQLJ) from the DB2 Express-C website (ibm.com/db2/express).

Il driver IBM DB2 per JDBC (anche conosciuto come driver JCC) è stato ottimizzato per tutti i server DB2 in tutte le piattaforme. Il file jar db2jcc.jar (com.ibm.db2.jcc) include i driver type 2 e type 4. Il file jar db2jcc.jar è incluso con qualsiasi client DB2, o può essere ottenuto da solo (driver IBM DB2 per JDBC e SQLJ) dal sito web DB2 Express-C (ibm.com/db2/express).

17.1.1 JDBC Type 2 driver / Il driver JDBC type 2

The JDBC type 2 driver requires a DB2 client to be installed where the JDBC application is running. Figure 17.1 illustrates a JDBC application using the type 2 driver.

Il driver JDBC type 2 necessita di un client DB2 installato dove l'applicazione JDBC è in esecuzione. La figura 17.1 illustra un'applicazione JDBC usando il driver type 2.

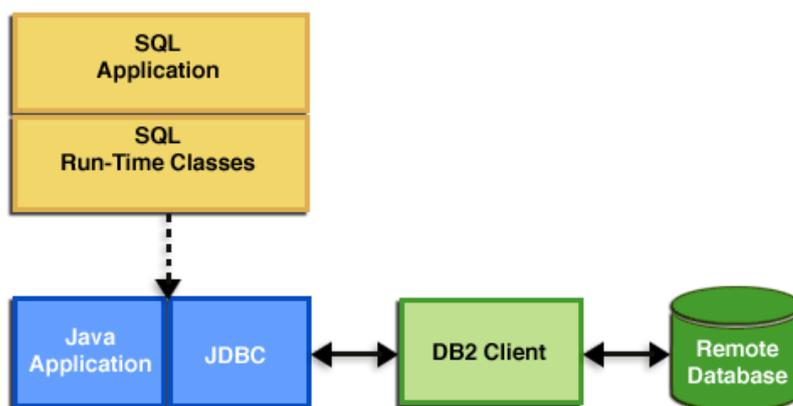


Figure 17.1 – The JDBC type 2 driver / Figura 17.1 – Il driver JDBC type 2

Figure 17.2 provides a listing that shows how to establish a connection using the JDBC Type 2 driver. Note that the URL does not include hostname or port information because this is taken from the DB2 client.

Nella figura 17.2 è illustrato il codice che mostra come stabilire un collegamento usando il tipo 2 driver JDBC. Nota che l'URL non include le informazioni su porta o hostname perché questi sono recuperati dal DB2 client.

```

...
public static final String DB_URL = "jdbc:db2:sample";
Properties connectProperties = new Properties();
connectProperties.put("user", "db2admin");
connectProperties.put("password", "ibmdb2");
Connection connection = null
try
{
    Class.forName("com.ibm.db2.jcc.DB2Driver").newInstance();
    connection = DriverManager.getConnection(url, connectProperties)
}
catch (Exception e)
throw e;
}
...

```

Figure 17.2 – Establishing a connection using the JDBC type 2 driver / Figura 17.2 – Stabilire un collegamento usando il driver JDBC type 2

17.1.2 JDBC Type 4 driver / Il driver JDBC type 4

The JDBC type 4 driver does not require a DB2 client to connect to a DB2 server. Figure 17.3 illustrates a JDBC application using the type 4 driver.

Il driver JDBC type 4 non richiede un client DB2 per connettersi a un server DB2. La figura 17.3 illustra un'applicazione JDBC usando il driver type 4.

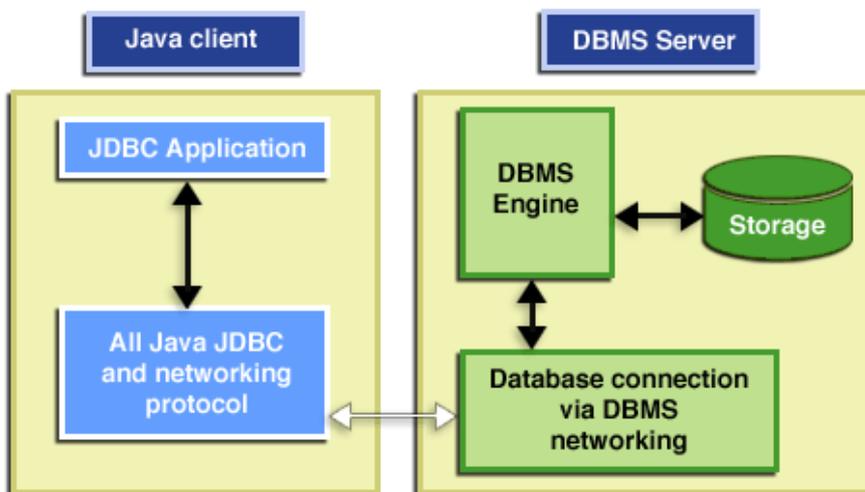


Figure 17.3 – The JDBC type 4 driver / Figura 17.3 - Il driver JDBC type 4

Figure 17.4 provides a listing showing how to establish a connection using the JDBC Type 4 driver. Note that the URL does include the hostname or port information.

La figura 17.4 fornisce il codice che mostra come stabilire una connessione usando il driver JDBC type 4. Nota che l'URL include le informazioni su porta e hostname.

```
...  
public static final String DB_URL = "jdbc:db2://server1:50000/sample";  
Properties connectProperties = new Properties();  
connectProperties.put("user", "db2admin");  
connectProperties.put("password", "ibmdb2");  
Connection connection = null  
try  
{  
    Class.forName("com.ibm.db2.jcc.DB2Driver").newInstance();  
    connection = DriverManager.getConnection(url, connectProperties)  
}  
catch (Exception e)  
    throw e;  
}  
...
```

Figure 17.4 – Establishing a connection using the JDBC type 4 driver / Figura 17.4 - Stabilire una connessione che usa il driver JDBC type 4

17.2 Application development in PHP / Sviluppo di applicazioni in PHP

PHP (PHP Hypertext Preprocessor) is an open source, platform independent scripting language that is designed for Web application development. It is one of the most widely deployed web languages in the world today. The popularity of PHP is based on the following characteristics of the language:

- Rapid, iterative development cycles with a low learning curve
- Robust, high-performance and scalable
- Stable and secure
- An alternative to J2EE™ and .NET on the Web
- Easily integrated into heterogeneous environments/systems
- Proven through widespread deployment
- Vibrant well established community

PHP is part of the LAMP stack (Linux, Apache HTTP Server, MySQL, **PHP** / Perl / Python). This is an open source web technology stack, often available on ISPs for reasonable monthly fees.

Il PHP (PHP Hypertext Preprocessor) è un linguaggio di scripting, open-source e platform independent, che è progettato per lo sviluppo di applicazioni Web. È uno dei linguaggi web più diffusi nel mondo oggi. La popolarità del PHP è basata sulle seguenti caratteristiche del linguaggio:

- Cicli di sviluppo veloci e iterativi con una curva di apprendimento bassa.
- Robustezza, elevate prestazioni e scalabilità.
- Stabilità e sicurezza
- Un'alternativa a J2EE™ e a .NET sul web.
- Facilmente integrabile negli ambienti/sistemi eterogenei
- Provato attraverso sviluppi diffusi
- Community affermata e vibrante

Il PHP fa parte della stack LAMP (Linux, Apache HTTP Server, MySQL, **PHP** / Perl / Python). Cioè uno stack open-source di tecnologie web, spesso disponibile sugli ISP per ragionevoli motivi di spesa.

17.2.1 DB2 connection options for PHP / Opzioni di connessione di DB2 per PHP

IBM supports access to a DB2 Database from PHP applications through two extensions.

IBM supporta l'accesso a un database DB2 da applicazioni PHP attraverso due estensioni.

ibm_db2:

The `ibm_db2` extension offers a procedural application programming interface to create, read, update and write database operations in addition to

L'estensione `ibm_db2` offre un'interfaccia per la programmazione di applicazioni procedurali per creare, leggere, aggiornare e scrivere le

extensive access to the database metadata. It can be compiled to work with either PHP 4 or PHP 5. The extension is available from the PECL repository under the Apache 2.0 license.

It was developed and is supported by IBM. It has full featured support for stored procedures and LOBs, and is fast, as it has been optimized for DB2.

PDO_ODBC:

The PDO_ODBC is a driver for the PHP Data Objects (PDO) extension, and offers access to DB2 databases through a standard object-oriented database interface introduced in PHP 5.1. It can be compiled directly against DB2 libraries. It provides a standard data access interface for PHP. It is fast, light weight, and object oriented. The PDO_ODBC extension uses DB2 libraries for native access, and has been built into PHP 5.1. For more information, see these sites:

- <http://pecl.php.net/package/pdo>
- http://pecl.php.net/package/PDO_ODBC

operazioni di database in aggiunta ad un vasto accesso ai metadati del database. Può essere compilato per funzionare sia con PHP 4 che con PHP 5. L'estensione è disponibile dal repository PECL sotto la licenza Apache 2.0.

Fu sviluppata ed è supportata da IBM. Ha un supporto completo per le stored procedure e i LOB, ed è veloce, poiché ottimizzato per DB2.

Il PDO_ODBC è un driver per l'estensione PHP Data Objects (PDO), e offre l'accesso ai database DB2 attraverso un'interfaccia database standard e object-oriented introdotta nel PHP 5.1. Può essere compilato direttamente dalle librerie DB2. Fornisce un'interfaccia standard di accesso ai dati per il PHP. È veloce, leggera ed object-oriented. L'estensione PDO_ODBC usa le librerie DB2 per l'accesso nativo ed è stata costruita all'interno di PHP 5.1. Per ulteriori informazioni, consulta questi siti:

Connecting to an uncatalogued DB2 database / Connettersi a un database DB2 non catalogato

Listing 17.1 shows how to connect to a DB2 database using either of the two extensions previously described.

Il listing 17.1 mostra come connettersi a un database DB2 usando l'una o l'altra delle due estensioni descritte precedentemente.

```
$host = 'localhost';
$port = 50000;
$DSN = "DRIVER={IBM DB2 ODBC DRIVER}; PORT=$port;
        HOSTNAME=$host; DATABASE=$database; PROTOCOL=TCPIP;
        USER=$user; PWD=$password";
```

```
-- If using the ibm_db2 extension --
$suconn = db2_connect($DSN, null, null);

-- If using the PDO_ODBC extension --
try {
    $suconn = new PDO("odbc:$DSN", null, null);
}
catch (PDOException $e) { print $e->errormsg(); }
```

Listing 17.1 – Connecting to an uncatalogued DB2 database / Elenco 17.1 – Connettersi a un database DB2 non catalogato

Listing 17.2 provides an example of a simple PHP application using the `ibm_db2` extension.

Il listing 17.2 mostra un esempio di una semplice applicazione semplice PHP usando l'estensione `ibm_db2`.

```
<?php
$sql = "SELECT name, breed FROM ANIMALS WHERE weight < ?";
$conn = db2_connect($database, $user, $password);
$stmt = db2_prepare($conn, $sql);
$res = db2_execute($stmt, array(10));
while ($row = db2_fetch_assoc($stmt)) {
    print "{$row['NAME']} is a {$row['BREED']}. \n";
}
?>
```

Listing 17.2 – A simple PHP application using the `ibm_db2` extension / Elenco 17.2 - Una semplice applicazione PHP usando l'estensione `ibm_db2`

Configuring PHP for `ibm_db2` / Configurare PHP per `ibm_db2`

In Linux or UNIX you may need to modify the `php.ini` file as follows:

In Linux o in UNIX può essere necessario modificare il file `php.ini` come segue:

```
extension=ibm_db2.so
ibm_db2.instance_name=<instance name>
```

On Windows, modify the `php.ini` file as follows:

Su Windows, modificare il file `php.ini` come segue:

```
extension=php_ibm_db2.dll
```

Alternatively, you can download and install the **Zend Core for IBM** application suite as described in the next section, and not have to worry about these configuration issues.

Alternativamente, puoi fare il download e installare la suite di applicativi **Zend Core per IBM** come è descritto nella sezione seguente senza preoccuparti di questi problemi di configurazione.

17.2.2 Zend Core for IBM / Zend Core per IBM

Zend Core is a seamless out-of-the-box PHP development and production environment for business-critical web applications. It delivers reliability, productivity and flexibility needed for running PHP applications. It can be downloaded for free from:

<http://ibm.com/software/data/info/zendcore>

Zend Core for IBM installs DB2 and IDS clients, an optional Apache HTTP Server, PHP 5, and popular PHP extensions including `ibm_db2`, and `PDO_INFORMIX`. Zend Core for IBM can optionally install DB2 Express-C server, IBM Cloudscape™ server, the complete PHP manual, and sample applications for DB2. It comes with an easy to use and configure PHP environment, as shown in Figure 17.5, 17.6 and 17.7.

Zend Core è un ambiente di sviluppo e produzione illimitato e out-of-the-box per PHP per applicazioni web business-critical. Offre affidabilità, produttività e la flessibilità richiesta per l'esecuzione di applicazioni PHP. Può essere scaricato gratuitamente da:

Zend Core per IBM installa DB2 e i client IDS, Apache HTTP Server opzionale, PHP 5, e le estensioni PHP più diffuse compreso `ibm_db2` e `PDO_INFORMIX`. Zend Core per IBM può facoltativamente installare il server DB2 Express-C, il server Cloudscape™ di IBM, il manuale completo di PHP, e le applicazioni di esempio per DB2. Fornisce un ambiente PHP che è facile da usare e configurare, come è mostrato nelle figure 17.5, 17.6 e 17.7.

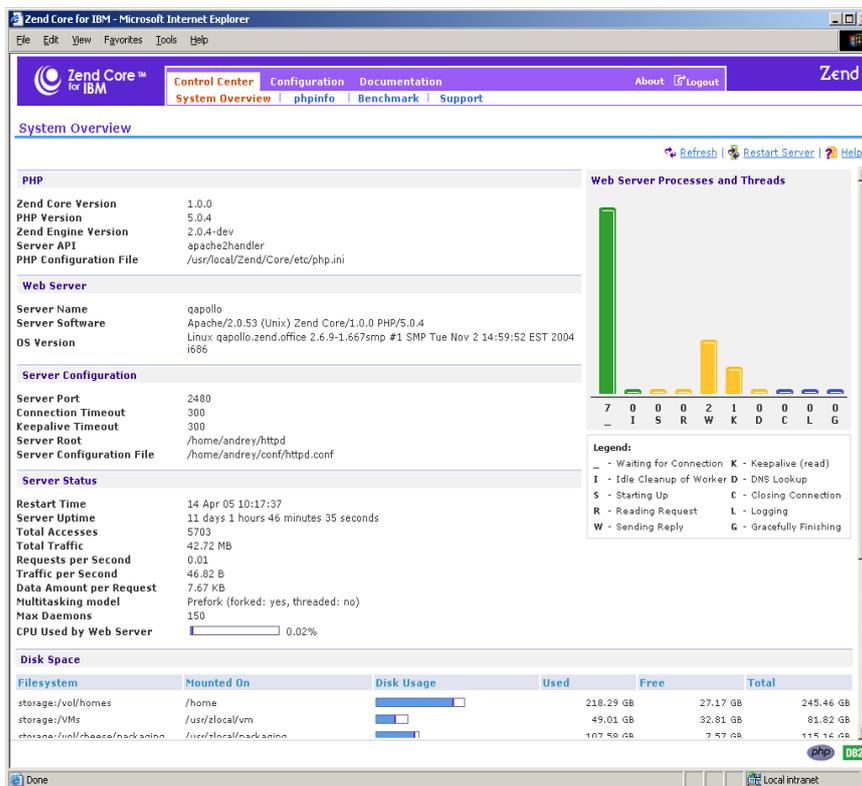


Figure 17.5 - Zend Core management and control interface / Figura 17.5 - Interfaccia Zend Core di gestione e controllo

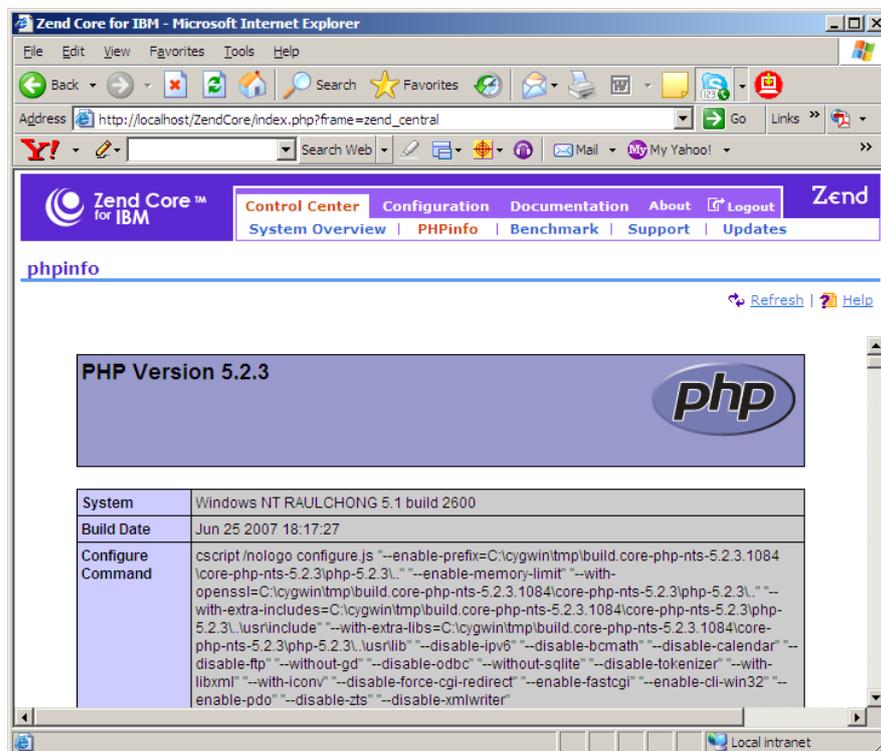


Figure 17.6 - Zend Core PHP configuration interface / Figura 17.6 - Interfaccia Zend Core di configurazione PHP

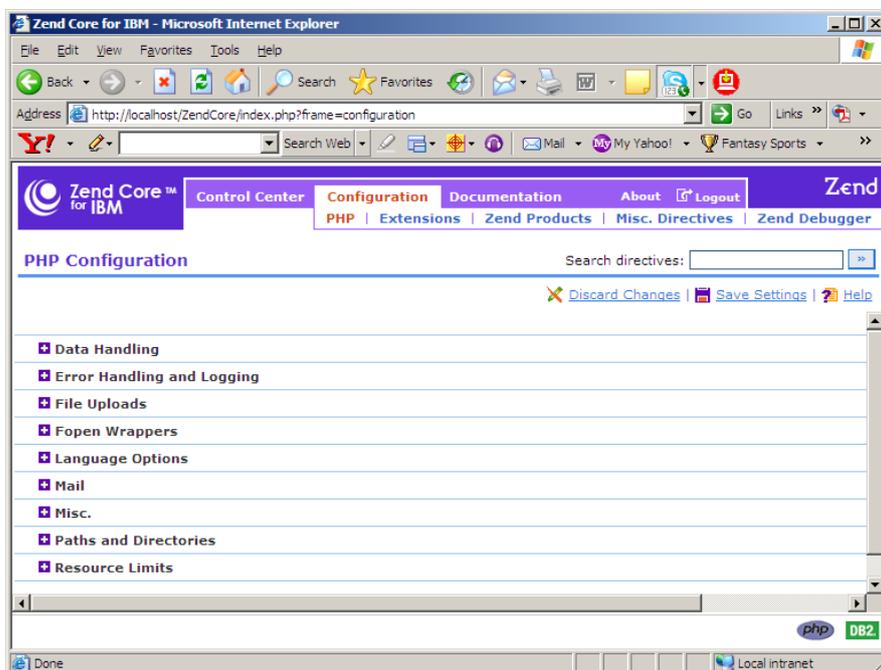


Figure 17.7 - Zend Core PHP configuration interface / Figura 17.7 - Interfaccia Zend Core di configurazione PHP

17.3 Application development in Ruby on Rails / Sviluppo di applicazioni in Ruby on Rails

Ruby is an object-oriented, dynamic, cross platform scripting language. It facilitates rapid development and includes a rich library. Ruby is a simple and fun oriented programming language invented by Yukihiro Matsumoto ("Matz") in 1995.

Rails is a full-stack framework for database-backed web applications written in Ruby. It implements the model-view-control (MVC) architecture. It is incredibly productive and easy to use. Rails is one of the fastest emerging web frameworks since 2004 and was invented by David Heinemeier Hansson.

Ruby è un linguaggio di scripting dinamico, object-oriented e cross-piattaforma. Facilita lo sviluppo veloce e include una ricca libreria. Ruby è un semplice linguaggio di programmazione orientato al divertimento inventato da Yukihiro Matsumoto ("Matz ") nel 1995.

Rails è un full-stack framework per applicazioni web database-backed scritte in Ruby. Implementa l'architettura model-view-control (MVC). È incredibilmente produttivo e facile da usare. Rails è uno dei più veloci e emergenti web frameworks dal 2004 ed è stato inventato da David Heinemeier Hansson.

17.3.1 Startup Toolkit for DB2 on Rails / Startup Toolkit per DB2 on Rails

IBM recognizes the importance of Ruby on Rails in the development community; therefore, it has created a package called **Startup Toolkit for DB2 on Rails**. This is an integrated installer that creates a complete DB2 Ruby on Rails development environment. It can be downloaded and used for free from the IBM alphaWorks Web site:

<http://www.alphaworks.ibm.com/tech/db2onrails>.

The Startup Toolkit for DB2 on Rails:

- Includes an integrated installer
- Helps you to easily install and configure Ruby and Rails
- Installs DB2 Express – C 9 and tools
- Includes an IBM developed DB2 Ruby driver and a DB2 Rails Adapter
- Includes various demos and tutorials

IBM riconosce l'importanza di Ruby on Rails nella community degli sviluppatori; quindi, ha creato un package chiamato **Startup Toolkit per DB2 on Rails**.

Questo è un installer integrato che crea un ambiente di sviluppo completo di DB2 Ruby on Rails. Può essere scaricato e usato gratuitamente dal sito web IBM alphaWorks:

Lo Startup Toolkit per DB2 on Rails:

- Include un installer integrato
- Ti aiuta a installare e configurare facilmente Ruby on Rails
- Installa DB2 Express-C 9 e i tool
- Include un driver DB2 Ruby sviluppato da IBM e un DB2 Rails Adapter
- Include varie demo e tutorial

A

Appendix A – Troubleshooting / Appendice A – Risoluzione dei problemi

This appendix discusses how to troubleshoot problems that may be encountered when working with DB2. Figure A.1 provides a brief overview of the actions to take should a problem arise.

Questa appendice tratta la risoluzione dei problemi che si possono incontrare lavorando con DB2. La figura A.1 offre una rapida panoramica delle azioni da intraprendere se si presenta un problema.

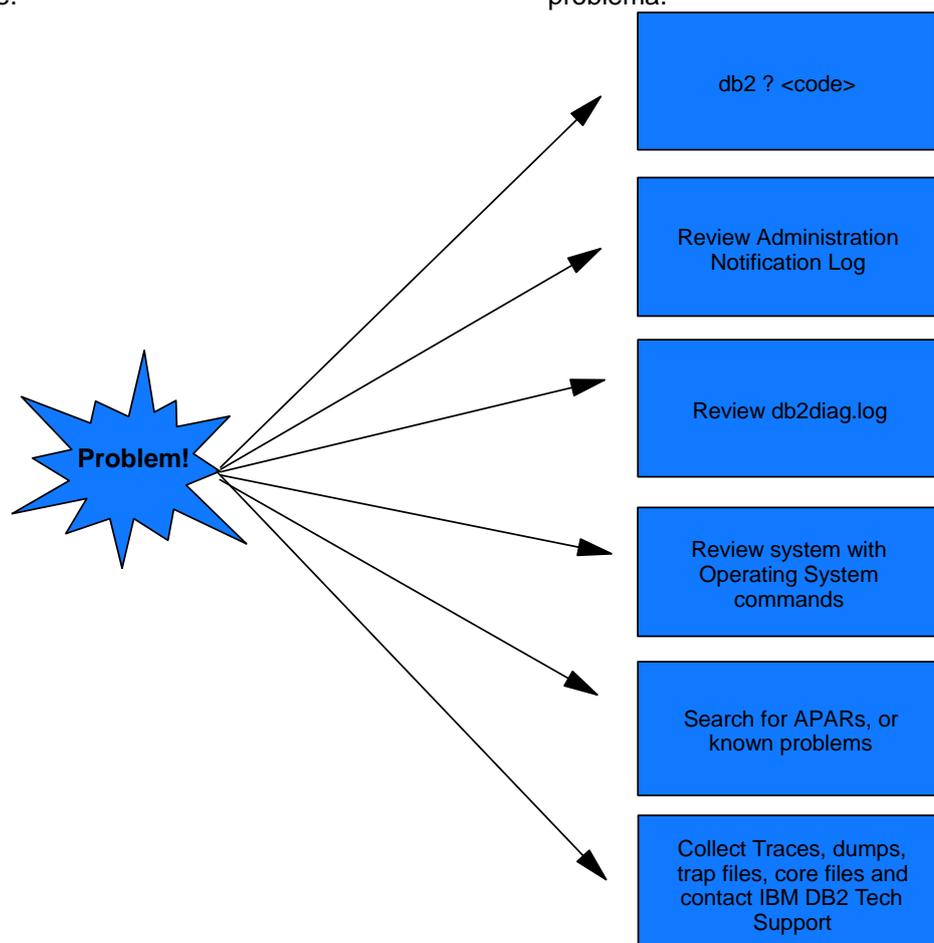


Figure A.1 – Troubleshooting overview / Figura A.1 - Panoramica della risoluzione dei problemi

Note:

For more information about troubleshooting, watch this video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4462>

Nota:

Per ulteriori informazioni sulla risoluzione dei problemi, guardi questo video:
<http://www.channeldb2.com/video/video/show?id=807741:Video:4462>

A.1 Finding more information about error codes / Trovare maggiori informazioni sui codici di errore

To obtain more information about an error code received, enter the code prefixed by a question mark in the Command Editor input area and click the *Execute* button. This is shown in Figure A.2.

Per ottenere maggiori informazioni su un codice di errore ricevuto, immetti il codice preceduto da un punto interrogativo nell'input area del Command Editor e fai clic sul tasto *Execute*. Questo è mostrato nella figura A.2.

Prefix error code by a question mark in the Input area

More information about the error code appears in the Output area

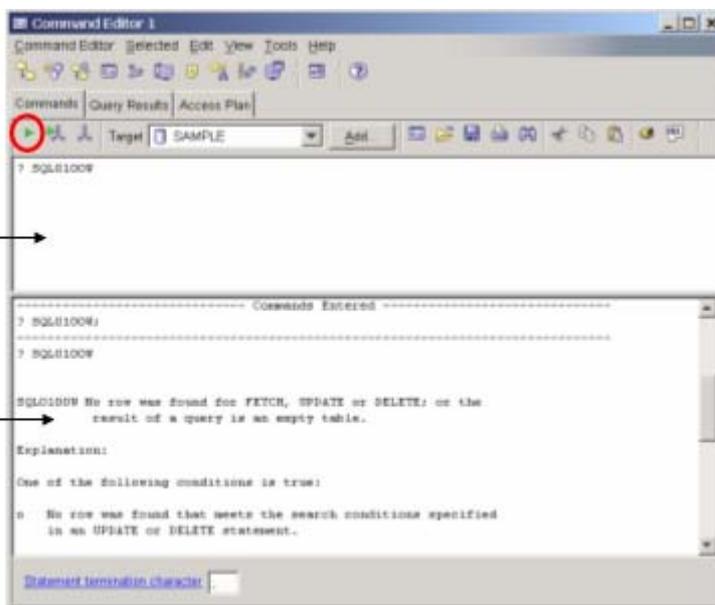


Figure A.2 – Finding more information about DB2 error codes / Figura A.2 - Trovare maggiori informazioni sui codici di errore di DB2

The question mark (?) invokes the DB2 `help` command.

Below are several examples of how to invoke it for help if you receive, for example, the SQL error code “-104”. All of the examples below are equivalent.

```
db2 ? SQL0104N
db2 ? SQL104N
db2 ? SQL-0104
db2 ? SQL-104
db2 ? SQL-104N
```

Il punto interrogativo (?) richiama l'`help` command di DB2.

Qui sotto ci sono parecchi esempi di come invocarlo per avere aiuto se ricevi, per esempio, il codice di errore di SQL “-104”. Tutti gli esempi qui sotto sono equivalenti.

A.2 SQLCODE and SQLSTATE / SQLCODE e SQLSTATE

An SQLCODE is a code received after every SQL statement is executed. The meanings of the values are summarized below:

SQLCODE = 0; the command was successful

SQLCODE > 0; the command was successful, but returned a warning

SQLCODE < 0; the command was unsuccessful and returned an error

The SQLSTATE is a five-character string that conforms to the ISO/ANSI SQL92 standard. The first two characters are known as the SQLSTATE class code:

- A class code of 00 means the command was successful.
- A class code of 01 implies a warning.
- A class code of 02 implies a not found condition.

All other class codes are considered errors.

Un SQLCODE è un codice ricevuto dopo che ogni istruzione SQL è stata eseguita. I significati dei valori sono riassunti qui sotto:

SQLCODE = 0; il comando è stato eseguito con successo

SQLCODE > 0; il comando è stato eseguito con successo, ma ha ritornato un warning

SQLCODE < 0; il comando non è stato eseguito con successo ed ha ritornato un errore

Lo SQLSTATE è una stringa di cinque-carattere che si conforma allo standard ISO/ANSI SQL92. I primi due caratteri sono conosciuti come il codice di classe di SQLSTATE:

- Un codice di classe di 00 indica che il comando è stato eseguito con successo.
- Un codice di classe di 01 implica un warning.
- Un codice di classe di 02 implica una condizione not found.

Tutti gli altri codici di classe sono considerati errori.

A.3 DB2 Administration Notification Log / L'Administration Notification Log di DB2

The DB2 administration notification log provides diagnostic information about errors at the point of failure. On Linux/UNIX platforms, the administration notification log is a text file called <instance name>.nfy (e.g. "db2inst.nfy"). On Windows, all administration notification messages are written to the Windows Event Log.

The DBM configuration parameter `notifylevel` allows administrators to specify the level of information to be recorded:

- 0 -- No administration notification messages captured (not recommended)
- 1 -- Fatal or unrecoverable errors
- 2 -- Immediate action required
- 3 -- Important information, no immediate action required (the default)
- 4 -- Informational messages

L'Administration Notification Log di DB2 offre informazioni diagnostiche sugli errori sul punto di malfunzionamento. Sulle piattaforme Linux/UNIX, l'Administration Notification Log è un file di testo chiamato <instance name>.nfy (per esempio "db2inst.nfy "). Su Windows, tutti i messaggi di notifica sono scritti sul Windows Event Log.

Il parametro di configurazione `notifylevel` del DBM permette agli amministratori di specificare il livello di informazioni che devono essere registrate:

- 0 – Nessun messaggio di administration notification catturato (non raccomandato)
- 1 – Errori fatali o irrecuperabili
- 2 – Azione immediata richiesta
- 3 – Informazioni importanti, nessuna azione immediata necessaria (default)
- 4 – Messaggi informativi

A.4 db2diag.log

The `db2diag.log` provides more detailed information than the DB2 Administration notification log. It is normally used only by IBM DB2 technical support or experienced DBAs. Information in the `db2diag.log` includes:

- The DB2 code location reporting an error.
- Application identifiers that allow you to match up entries pertaining to an application on the `db2diag.log`s of servers and clients.
- A diagnostic message (beginning with "DIA") explaining the reason for the error.
- Any available supporting data, such

Il `db2diag.log` offre informazioni più dettagliate rispetto all'Administration Notification Log di DB2. È usato normalmente soltanto dal supporto tecnico di IBM DB2 o da DBA con esperienza. Le informazioni nel `db2diag.log` includono:

- La posizione nel codice di DB2 che riporta un errore.
- Gli identificatori dell'applicazione che permettono di confrontare le entries pertinenti un'applicazione sui file `db2diag.log` dei server e dei client.
- Un messaggio diagnostico (che inizia con "DIA") che spiega il motivo dell'errore.

as SQLCA data structures and pointers to the location of any extra dump or trap files.

- Qualsiasi dato di supporto disponibile, come strutture di dati di SQLCA e indicatori della posizione di qualsiasi file supplementare di dump o di trap.

On Windows, the db2diag.log is located by default under the directory:

Su Windows, db2diag.log è situato di default sotto la directory:

```
C:\Program Files\IBM\sqlllib\\db2diag.log
```

On Linux/UNIX, the db2diag.log is located by default under the directory:

Su Linux/UNIX, db2diag.log è situato di default sotto la directory:

```
/home/<instance_owner>/sqlllib/db2dump/db2diag.log
```

The verbosity of diagnostic text is determined by the dbm cfg configuration parameter DIAGLEVEL. The range is 0 to 4, where 0 is the least verbose, and 4 is the most. The default level is 3.

La verbosità del testo diagnostico è determinata dal parametro di configurazione DIAGLEVEL del dbm cfg. L'intervallo è 0 - 4, dove 0 è il minimo livello di verbose, e 4 il massimo. Il livello di default è 3.

A.5 CLI traces / Trace di CLI

For CLI and Java Applications, you may turn on the CLI trace facility to troubleshoot your application. This can be done by making changes to the db2cli.ini file at the server where your application is running. Typical entries in the db2cli.ini file are shown below.

Per le applicazioni di Java e di CLI, puoi attivare la traccia di CLI per diagnosticare la tua applicazione. Ciò può essere fatto facendo delle modifiche al file db2cli.ini sul server dove la tua applicazione è in esecuzione. Le tipiche entries del file db2cli.ini sono mostrate qui sotto.

```
[common]
trace=0
tracerefreshinterval=300
tracepathname=/path/to/writeable/directory
traceflush=1
```

Low level tracing (db2trc) is also available, but this is typically only useful for DB2 technical support.

Il tracciamento di livello basso (db2trc) è anche disponibile, ma questo è tipicamente utile soltanto per il supporto tecnico di DB2.

A.6 DB2 Defects and Fixes / Difetti e Fix di DB2

Sometimes a problem you encounter may be caused by a defect in DB2. IBM regularly releases fix packs which contain code fixes for defects (APARs). The fix pack documentation contains a list of the fixes contained in the fix pack. When developing new applications, we always recommend using the latest fix pack to benefit from the latest fixes. To view your current version and fix pack level: from Control Center, select the *About* option from the *Help* menu; from the Command Window, type "db2level". Note that fix packs and official IBM DB2 technical support are only offered in DB2 Express-C if you purchase the 12 months subscription license.

A volte un problema che incontri può essere causato da un difetto in DB2. IBM rilascia regolarmente rilascia dei fix pack che contengono correzioni di codice per i difetti (APARs). La documentazione dei fix pack contiene un elenco delle correzioni contenute negli stessi. Quando sviluppi delle nuove applicazioni, raccomandiamo di usare sempre l'ultimo fix pack per beneficiare delle correzioni più recenti. Per vedere la tua attuale versione e il livello del fix pack: dal Control Center, seleziona l'opzione *About* dal menu *Help* o *Aiuto*; dalla Command Window, immetti "db2level". Nota che i fix packs e il supporto tecnico ufficiale di IBM DB2 sono offerti per DB2 Express-C soltanto se hai acquistato la 12 month subscription license.

Resources / Risorse

Web sites / Siti web

1. DB2 Express-C web site:
www.ibm.com/db2/express
Use this web site to download the image for DB2 Express-C servers, DB2 clients, DB2 drivers, manuals, access to the team blog, mailing list sign up, etc.

Usate questo sito per il download dei file immagine per i server di DB2 Express-C, i client DB2, i driver DB2, i manuali, l'accesso al team blog, registrazioni a mailing list, ecc.
2. DB2 Express forum:
www.ibm.com/developerworks/forums/dw_forum.jsp?forum=805&cat=19
Use the forum to post technical questions when you cannot find the answers in the manuals yourself.

Usa il forum per postare le domande tecniche quando non trovi le risposte nei manuali.
3. DB2 Information Center
<http://publib.boulder.ibm.com/infocenter/db2luw/v9/index.jsp>
The information center provides access to the online manuals. It is the most up to date source of information.

Il centro d'informazione offre l'accesso ai manuali online. È la fonte di informazione più aggiornata.
4. developerWorks
<http://www-128.ibm.com/developerworks/db2>
This Web site is an excellent resource for developers and DBAs providing access to current articles, tutorials, etc. for free.

Questo sito web è una risorsa eccellente per gli sviluppatori e i DBA e offre l'accesso agli articoli attuali, ai tutorial , ecc. tutti gratis.
5. alphaWorks
<http://www.alphaworks.ibm.com/>
This Web site provides direct access to IBM's emerging technology. It is a place where one can find the latest technologies from IBM Research.

Questo sito web offre l'accesso diretto alla tecnologia emergente di IBM. È il posto dove si possono trovare le ultime tecnologie derivate dalla ricerca di IBM.

6. planetDB2
www.planetDB2.com
This is a blog aggregator from many contributors who blog about DB2. Questo è un blog aggregator da molti contributor che fanno blog su DB2.

7. DB2 Technical Support / Il supporto tecnico di DB2
If you purchased the 12 months subscription license of DB2 Express-C, you can download fixpacks from this Web site. Se ai comprato la 12 month subscription license di DB2 Express-C, puoi fare il download dei fix packs da questo sito web:

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

8. ChannelDB2
ChannelDB2 is a social network for the DB2 community. It features content such as DB2 related videos, demos, podcasts, blogs, discussions, resources, etc. for Linux, UNIX, Windows, z/OS, and i5/OS. ChannelDB2 è una social network per la community di DB2. Contiene materiale come i video relativi a DB2, le dimostrazioni, i podcast, i blog, le discussioni, le risorse, ecc. per Linux, UNIX, Windows, z/OS, e i5/OS.

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

Books / Libri

1. Free Redbook: DB2 Express-C: The Developer Handbook for XML, PHP, C/C++, Java, and .NET
Whei-Jen Chen, John Chun, Naomi Ngan, Rakesh Ranjan, Manoj K. Sardana,
August 2006 - SG24-7301-00
<http://www.redbooks.ibm.com/abstracts/sg247301.html?Open>
2. Understanding DB2 – Learning Visually with Examples V9.5
Raul F. Chong, et all. January 2008
ISBN-10: 0131580183
3. DB2 9: pureXML overview and fast start by Cynthia M. Saracco, Don Chamberlin,
Rav Ahuja June 2006 SG24-7298
<http://www.redbooks.ibm.com/abstracts/sg247298.html?Open>
4. DB2® SQL PL: Essential Guide for DB2® UDB on Linux™, UNIX®, Windows™,
i5/OS™, and z/OS®, 2nd Edition
Zamil Janmohamed, Clara Liu, Drew Bradstock, Raul Chong, Michael Gao, Fraser
McArthur, Paul Yip
ISBN: 0-13-100772-6
5. Free Redbook: DB2 pureXML Guide
Whei-Jen Chen, Art Sammartino, Dobromir Goutev, Felicity Hendricks, Ipeei Komi,
Ming-Pang Wei, Rav Ahuja, Matthias Nicola. August 2007
<http://www.redbooks.ibm.com/abstracts/sg247315.html?Open>
6. Information on Demand - Introduction to DB2 9 New Features
Paul Zikopoulos, George Baklarz, Chris Eaton, Leon Katsnelson
ISBN-10: 0071487832
ISBN-13: 978-0071487832
7. Redbook: Developing PHP Applications for IBM Data Servers.
Whei-Jen Chen, Holger Kirstein, Daniel Krook, Kiran H Nair, Piotr Pietrzak
May 2006 - SG24-7218-00
<http://www.redbooks.ibm.com/abstracts/sg247218.html?Open>

Contact emails / Contatti email

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

Mailbox generale di DB2
Express-C: db2x@ca.ibm.com

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

Mailbox generale di DB2 on
Campus: db2univ@ca.ibm.com

Getting started with DB2 couldn't be easier. Read this book to:

- Find out what DB2 Express-C is all about
- Understand DB2 architecture, tools, security
- Learn how to administer DB2 databases
- Write SQL, XQuery, stored procedures
- Develop database applications for DB2
- Practice using hands-on exercises
- Prepare for the DB2 on Campus test

Iniziare con DB2 non può essere più facile. Leggi questo libro per:

- Scoprire di che si tratta parlando di DB2 Express-C
- Capire l'architettura di DB2, i tool, la sicurezza
- Imparare come amministrare database DB2
- Scrivere SQL, XQuery, stored procedure
- Sviluppare applicazioni database per DB2
- Esercitarsi usando gli esercizi hands-on
- Prepararsi per il test di DB2 on Campus

The rapid adoption of XML for application integration, Web 2.0, and SOA is driving the need for innovative hybrid data servers. DB2 Express-C from IBM is a no-charge, no-limits, hybrid data server capable of managing both XML and traditional relational data with ease. No-charge means DB2 Express-C is free to download, free to build your applications, free to deploy into production, and free to redistribute with your solution. And, DB2 does not place any artificial limits on the size of database, number of databases, or number of users.

DB2 Express-C runs on Windows and Linux systems and provides application drivers for a variety of programming languages including C/C++, Java, .NET, PHP, Perl, and Ruby. Optional low-cost subscription and support with additional capabilities is available. If you require even greater scalability or more advanced functionality, you can seamlessly deploy applications built using DB2 Express-C to other DB2 editions such as DB2 Enterprise.

This free edition of DB2 is ideal for developers, consultants, ISVs, DBAs, students, or anyone who intends to develop, test, deploy, or distribute database applications. Join the growing DB2 Express-C user community today and take DB2 Express-C for a test-drive. Start discovering how you can create next generation applications and deliver innovative solutions.

La rapida adozione di XML per integrare applicazioni, per il Web 2,0 e per SOA sta portando all'esigenza di ibridi data server innovativi. DB2 Express-C di IBM è uno di questi gratuito, senza limiti, un data server ibrido capace di controllare facilmente sia XML che dati relazionali tradizionali. Gratuito significa che DB2 Express-C è libero per il download, libero per sviluppare le vostre applicazioni, libero per fare il deploy e libero da ridistribuire con la tua soluzione. Inoltre, DB2 non pone alcun limite artificiale alla dimensione del database, al numero di database, o al numero degli utenti.

DB2 Express-C gira su sistemi Linux e Windows e fornisce i driver per applicazioni scritte in vari linguaggi di programmazione compresi C/C++, Java, .NET, PHP, Perl e Ruby. La subscription low-cost è facoltativa e il supporto con ulteriori potenzialità è disponibile. Se hai bisogno di una scalabilità ancora maggiore o di funzionalità più avanzate, puoi far passare senza sforzi la tua applicazione sviluppata con DB2 Express-C ad altre edizioni di DB2 come DB2 Enterprise.

Questa edizione gratuita di DB2 è ideale per gli sviluppatori, i consulenti, i venditori di software, i DBA, gli studenti, o chiunque intenda mettere a punto, sperimentare, rilasciare, o distribuire applicazioni con database. Unisciti oggi alla crescente community degli utenti di DB2 Express-C e usa DB2 Express-C per un test-drive. Inizia a scoprire come poter creare le applicazioni del futuro e fornire soluzioni innovative.