Related Books of Interest

**DB2 Developer's Guide**
By Craig Mullins
ISBN: 0-13-283642-4
The field’s #1 go-to source for on-the-job information on programming and administering DB2 on IBM z/OS mainframes.
Now, three-time IBM Information Champion Craig S. Mullins has thoroughly updated this classic for the newest versions of DB2 for z/OS: DB2 V9 and V10.
This Sixth Edition builds on the unique approach that has made previous editions so valuable. It brings together condensed, easy-to-read coverage of all essential topics: information otherwise scattered through dozens of IBM and third-party documents. Throughout, Mullins offers focused drill-down on the key details DB2 developers need to succeed, with expert, field-tested implementation advice and realistic examples.

**DB2 SQL Tuning Tips for z/OS Developers**
By Tony Andrews
ISBN: 0-13-303846-7
The Definitive Solutions-Oriented Guide to IBM® DB2® for z/OS®: Now Fully Updated for Both v9 and v10!
DB2 tuning expert Tony Andrews (“Tony the Tuner”) draws on more than 20 years of DB2-related experience, empowering you to take performance into your own hands, whether you’re writing new software or tuning existing systems. Tony shows you exactly how to clear bottlenecks, resolve problems, and improve both speed and reliability.
This book fully reflects the latest SQL programming best practices for DB2 V9 and DB2 V10 on z/OS, techniques that are taught in no other book and are rarely covered in general DB2 SQL courses.

Sign up for the monthly IBM Press newsletter at ibmpressbooks/newsletters
Related Books of Interest

**DB2 9 for Linux, UNIX, and Windows**
**DBA Guide, Reference, and Exam Prep, Sixth Edition**
By George Baklarz and Paul C. Zikopoulos
The sixth edition of this classic offers complete, start-to-finish coverage of DB2® 9 administration and development for Linux®, UNIX®, and Windows® platforms, as well as authoritative preparation for the latest IBM® DB2 certification exam. Written for both DBAs and developers, this definitive reference and self-study guide covers all aspects of deploying and managing DB2 9, including DB2 database design and development; day-to-day administration and backup; deployment of networked, Internet-centered, and SOA-based applications; migration; and much more. You’ll also find an unparalleled collection of expert tips for optimizing performance, availability, and value. Download Complete DB2 V9 Trial Version. Visit ibm.com/db2/9/download.html to download a complete trial version of DB2, which enables you to try out dozens of the most powerful features of DB2 for yourself—everything from pureXML™ support to automated administration and optimization.

**DB2 pureXML Cookbook**
**Master the Power of the IBM Hybrid Data Server**
By Matthias Nicola and Pav Kumar-Chatterjee
ISBN: 0-13-815047-8
DB2® pureXML® Cookbook provides hands-on solutions and best practices for developing and managing XML database applications with DB2. More and more database developers and DBAs are being asked to develop applications and manage databases that involve XML data. Many are utilizing the highly praised DB2 pureXML technology from IBM®. In DB2 pureXML Cookbook, two leading experts from IBM offer the practical solutions and proven code samples that database professionals need to build better XML solutions faster. Organized by task, this book is packed with more than 700 easy-to-adapt “recipe-style” examples covering the entire application lifecycle—from planning and design through coding, optimization, and troubleshooting.

Visit ibmPRESSBOOKS.com for all product information

Listen to the author’s podcast at: ibmPRESSBOOKS.com/podcasts
Enterprise Master Data Management
An SOA Approach to Managing Core Information
By Allen Dreibelbis, Eberhard Hechler, Ivan Milman, Martin Oberhofer, Paul van Run, and Dan Wolfson
ISBN: 0-13-236625-8

Enterprise Master Data Management provides an authoritative, vendor-independent MDM technical reference for practitioners: architects, technical analysts, consultants, solution designers, and senior IT decision makers. Written by the IBM® data management innovators who are pioneering MDM, this book systematically introduces MDM’s key concepts and technical themes, explains its business case, and illuminates how it interrelates with and enables SOA. Drawing on their experience with cutting-edge projects, the authors introduce MDM patterns, blueprints, solutions, and best practices published nowhere else—everything you need to establish a consistent, manageable set of master data, and use it for competitive advantage.
This page intentionally left blank
This page intentionally left blank
The authors and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

© Copyright 2014 by International Business Machines Corporation. All rights reserved.

Note to U.S. Government Users: Documentation related to restricted right. Use, duplication, or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corporation.

IBM Press Program Managers: Steven M. Stansel, Ellice Uffer
Cover design: IBM Corporation
Associate Publisher: Dave Dusthimer
Marketing Manager: Stephane Nakib
Executive Editor: Mary Beth Ray
Publicist: Heather Fox
Editorial Assistant: Vanessa Evans
Development Editor: Jeff Riley
Managing Editor: Kristy Hart
Cover Designer: Alan Clements
Senior Project Editor: Lori Lyons
Copy Editor: Apostrophe Editing Services
Indexer: Heather McNeill
Senior Compositor: Gloria Schurick
Proofreader: Sherri Cain
Manufacturing Buyer: Dan Uhrig

Published by Pearson plc
Publishing as IBM Press

IBM Press offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales, which may include electronic versions and/or custom covers and content particular to your business, training goals, marketing focus, and branding interests. For more information, please contact

U. S. Corporate and Government Sales
1-800-382-3419
corpsales@pearsontechgroup.com.

For sales outside the United States, please contact

International Sales
international@pearsoned.com.
I would like to thank my wife Jin, and my two daughters Meylin and Isabelle, for their understanding, patience, support, and love.
The many weekends and nights spent writing this book meant many sacrifices, and quality family time lost. 
But you all kept me going to complete this book!
I would also like to thank my parents, in-laws, and siblings for their constant support and love.

Raul F. Chong

I didn’t think I would write another book after completing my fourth one a couple years ago.
I would like to sincerely thank Raul for his encouragement (and convincing) to become THE co-author of this book.
It is indeed a proud achievement to find five books (so far) listed when my name is “googled.”
I want to dedicate this book to my lovely family: Heison, Kristen, Ansel, my parents, and my in-laws.
I thank you for your support on giving me private time to work on the chapters over weekends and late nights.

We missed our annual Ride for Heart event this year, didn’t we?
Finally, congratulations to the entire DB2 for LUW team. We did it again!

Clara Liu
# Contents at a Glance

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreword</td>
<td></td>
<td>xxiv</td>
</tr>
<tr>
<td>Chapter 1</td>
<td>Introduction to DB2</td>
<td>1</td>
</tr>
<tr>
<td>Chapter 2</td>
<td>DB2 at a Glance: The Big Picture</td>
<td>31</td>
</tr>
<tr>
<td>Chapter 3</td>
<td>Installing DB2</td>
<td>89</td>
</tr>
<tr>
<td>Chapter 4</td>
<td>Using Database Tools and Utilities</td>
<td>137</td>
</tr>
<tr>
<td>Chapter 5</td>
<td>Understanding the DB2 Environment,</td>
<td>187</td>
</tr>
<tr>
<td></td>
<td>DB2 Instances, and Databases</td>
<td></td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Configuring Client and Server Connectivity</td>
<td>235</td>
</tr>
<tr>
<td>Chapter 7</td>
<td>Working with Database Objects</td>
<td>285</td>
</tr>
<tr>
<td>Chapter 8</td>
<td>Implementing Security</td>
<td>415</td>
</tr>
<tr>
<td>Chapter 9</td>
<td>Understanding Concurrency and Locking</td>
<td>499</td>
</tr>
<tr>
<td>Chapter 10</td>
<td>Maintaining, Backing Up, and Recovering Data</td>
<td>553</td>
</tr>
<tr>
<td>Appendix A</td>
<td>Solutions to the Review Questions</td>
<td>629</td>
</tr>
<tr>
<td>Appendix B</td>
<td>Introduction to SQL</td>
<td>645</td>
</tr>
<tr>
<td>Appendix C</td>
<td>A Comparison of DB2 and Oracle Terminology</td>
<td>675</td>
</tr>
<tr>
<td>Appendix D</td>
<td>Diagnosing Problems</td>
<td>683</td>
</tr>
<tr>
<td>Appendix E</td>
<td>Resources</td>
<td>701</td>
</tr>
<tr>
<td>Index</td>
<td></td>
<td>707</td>
</tr>
</tbody>
</table>
## Contents

**Foreword**

**Chapter 1  Introduction to DB2**  
A Brief History of DB2: From Past to Present  
The Role of DB2 in the Big Data World  
Characteristics of Big Data  
Types of Big Data  
The IBM Big Data Platform  
Integration of DB2 with BigInsights (Hadoop)  
DB2 Editions  
DB2 Express-C  
DB2 Express Server Edition  
DB2 Workgroup Server Edition  
DB2 Enterprise Server Edition  
DB2 Advanced Workgroup Server Edition  
DB2 Advanced Enterprise Server Edition  
DB2 Developer Edition  
DB2 Advanced Recovery Feature  
IBM Data Server Clients and Driver Packages  
Mainframe Host Connectivity  
Database Federation Support  
Database Replication Support  
DB2 Syntax Diagram Conventions  
Case Study  
Summary  
Review Questions  

**Chapter 2  DB2 at a Glance: The Big Picture**  
SQL Statements, XQuery Statements, and DB2 Commands  
SQL Statements  
XQuery Statements  
DB2 System Commands  
DB2 Command Line Processor (CLP) Commands
## Chapter 3 Installing DB2

DB2 Installation: The Big Picture 90  
DB2 Installation System Requirements 94  
Installing DB2 Using the DB2 Setup Wizard 95  
  - Launch the DB2 Setup Wizard on Windows 96  
  - Launch the DB2 Setup Wizard on Linux and UNIX 97  
  - Generate a Response File 97  
  - Select Features to Be Installed 98  
  - Specify a Unique DB2 Copy Name 99  
  - Set User Information for the DB2 Administration Server 99  
  - Create and Configure the DB2 Instance 99  
  - Enable Operating System Security for DB2 Objects (Windows Only) 100  
  - Review Installation Settings and Start the Installation 101  
Root and Non-Root Installation on Linux and UNIX 101  
  - Limitations of Non-Root Installations 102  
  - Installing DB2 with a Non-Root User 103  
  - Enabling Some Root-Based Features in Non-Root Installations 103  
Required User IDs and Groups 105  
  - User IDs and Groups Required for Windows 105  
  - User IDs and Groups Required for Linux and UNIX 106  
Silent Install Using a Response File 107  
  - Creating a Response File Using the DB2 Setup Wizard 109  
  - Installing DB2 Using a Response File on Windows 112  
  - Installing DB2 Using a Response File on Linux and UNIX 112  
Advanced DB2 Installation Methods (Linux and UNIX Only) 113  
  - Installing DB2 Using the db2_install Script 113  
  - Manually Installing the DB2 Payload Files 115  
Installing a DB2 License 116  
Reducing DB2 Product Installation Image Size 118  
  - Customizing DB2 Installation Images 118  
  - Installing DB2 Using a Pruned Installation Image 120  
Installing Multiple DB2 Versions and Fix Packs on the Same Server 121  
  - Coexistence of Multiple DB2 Versions and Fix Packs (Windows) 121  
  - Coexistence of Multiple DB2 Versions and Fix Packs (Linux and UNIX) 124  
  - The db2ls Command (Linux and UNIX) 124  
  - DB2 Administrative Server (DAS) and Multiple DB2 Copies 126  
Installing DB2 Fix Packs 126  
  - Applying Fix Packs to a Non-Root Installation 128  
Upgrading to the Latest DB2 Version 128  
Case Study 1 128  
Case Study 2 130  
Summary 132  
Review Questions 132
Chapter 4 Using Database Tools and Utilities

- Database Tools: The Big Picture 137
- The Command-Line Tools 138
  - The DB2 Command Window 139
  - The DB2 Command Line Processor 141
  - The DB2 Command Line Processor Plus 154
- IBM Data Studio 160
  - IBM Data Studio Workspace and the Task Launcher 161
  - Connection Profiles 162
  - General Database Administration Tools 164
  - General Database Development Tools 171
- IBM Data Studio Web Console 174
- Set-Up Tools 175
  - Configure DB2 .NET Data Provider 176
  - First Steps 176
  - Default DB2 and Database Client Interface Selection Wizard 177
  - The Replication Center 178
- Information Tools 178
  - DB2 Information Center 179
  - Checking for DB2 Updates 180
- Problem Determination Tools 180
  - The db2pd Tool 181
- Case Study 1 181
- Case Study 2 182
- Summary 183
- Review Questions 184

Chapter 5 Understanding the DB2 Environment, DB2 Instances, and Databases

- The DB2 Environment, DB2 Instances, and Databases: The Big Picture 187
- The DB2 Environment 188
- The DB2 Instance 198
  - Creating DB2 Instances 200
  - Creating Client Instances 201
  - Creating DB2 Instances in a pureScale Environment 202
  - Dropping an Instance 202
  - Listing the Instances in Your System 203
- Using the DB2INSTANCE Environment Variable 204
  - Starting a DB2 Instance 204
  - Stopping a DB2 Instance 207
  - Attaching to an Instance 208
  - Configuring an Instance 209
- Working with an Instance from IBM Data Studio 214
- Using the DB2 Commands at the Instance Level 216
Chapter 6  Configuring Client and Server Connectivity  

Client and Server Connectivity: The Big Picture  
The DB2 Database Directories  
The DB2 Database Directories: An Analogy Using a Book  
The System Database Directory  
The Local Database Directory  
The Node Directory  
The Database Connection Services Directory  
The Relationship Between the DB2 Directories  
Supported Connectivity Scenarios  
Scenario 1: Local Connection from a Data Server Client to a DB2 Server  
Scenario 2: Remote Connection from a Data Server Client to a DB2 Server  
Scenario 3: Remote Connection from a Data Server Client to a DB2 Host Server  
Scenario 4: Remote Connection from a Data Server Client to a DB2 Host Server via a DB2 Connect Gateway  
Scenario 5: Remote Connection from an Application to a DB2 Server  
DB2 Packages and the Bind Process  
Automatic Client Reroute Feature  
Application Connection Timeout Support  
TCP/IP Keepalive Timeout Support  
Diagnosing DB2 Connectivity Problems  
Diagnosing Client-Server TCP/IP Connection Problems  
Case Study  
Step 1: Configure the DB2 Connect Gateway Machine  
Step 2: Test the Connection from the DB2 Connect Gateway Machine to the Host  
Step 3: Enable the TCP/IP Listener on the Gateway Machine  
Step 4: Configure a Data Server Client to Connect to the Host via the Gateway  
Summary  
Review Questions  

Chapter 7  Working with Database Objects  

Database Objects: The Big Picture  
Databases  
Database Partitions  
Automatic Storage
Chapter 8  Implementing Security  415

DB2 Security Model: The Big Picture  415
Authentication Methods  417
  Configuring the Authentication Type at a DB2 Server  417
  Configuring the Authentication Type at a DB2 Client  419
  Authenticating Users at the DB2 Server  421
  Authenticating Users Using the Kerberos Security Service  423
  Authenticating Users with Generic Security Service Plug-ins  424
  Authenticating Users at the Data Server Client  427
Administrative Authorities  431
  Managing Administrative Authorities  433
<table>
<thead>
<tr>
<th>Contents</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Object Privileges</td>
<td>438</td>
</tr>
<tr>
<td>Schema Privileges</td>
<td>438</td>
</tr>
<tr>
<td>Table Space Privileges</td>
<td>440</td>
</tr>
<tr>
<td>Table and View Privileges</td>
<td>441</td>
</tr>
<tr>
<td>Index Privileges</td>
<td>444</td>
</tr>
<tr>
<td>Package Privileges</td>
<td>445</td>
</tr>
<tr>
<td>Routine Privileges</td>
<td>446</td>
</tr>
<tr>
<td>Sequence Privileges</td>
<td>448</td>
</tr>
<tr>
<td>Security Label Privileges</td>
<td>449</td>
</tr>
<tr>
<td>SET SESSION AUTHORIZATION Statement and SETSESSIONUSER Privilege</td>
<td>450</td>
</tr>
<tr>
<td>Implicit Privileges</td>
<td>452</td>
</tr>
<tr>
<td>Roles and Privileges</td>
<td>453</td>
</tr>
<tr>
<td>TRANSFER OWNERSHIP Statement</td>
<td>456</td>
</tr>
<tr>
<td>Data Encryption</td>
<td>456</td>
</tr>
<tr>
<td>Label-Based Access Control (LBAC)</td>
<td>458</td>
</tr>
<tr>
<td>Views and LBAC</td>
<td>462</td>
</tr>
<tr>
<td>Implementing an LBAC Security Solution</td>
<td>462</td>
</tr>
<tr>
<td>LBAC in Action</td>
<td>465</td>
</tr>
<tr>
<td>Column Level Security and Referential Integrity</td>
<td>466</td>
</tr>
<tr>
<td>Row and Column Access Control (RCAC)</td>
<td>467</td>
</tr>
<tr>
<td>Built-In SQL Functions and Session Variables Supporting RCAC</td>
<td>468</td>
</tr>
<tr>
<td>Creating Row Permissions</td>
<td>469</td>
</tr>
<tr>
<td>Creating Column Masks</td>
<td>471</td>
</tr>
<tr>
<td>Enforcing Row Permissions and Column Masks</td>
<td>472</td>
</tr>
<tr>
<td>Behavior of INSERT, DELETE, and UPDATE Under RCAC</td>
<td>473</td>
</tr>
<tr>
<td>Implementing a RCAC Security Solution</td>
<td>473</td>
</tr>
<tr>
<td>RCAC in Action</td>
<td>475</td>
</tr>
<tr>
<td>Extending the Case Scenario</td>
<td>476</td>
</tr>
<tr>
<td>Benefits of Using RCAC</td>
<td>478</td>
</tr>
<tr>
<td>Trusted Contexts</td>
<td>479</td>
</tr>
<tr>
<td>Windows Security Considerations</td>
<td>481</td>
</tr>
<tr>
<td>Windows Domain Considerations</td>
<td>481</td>
</tr>
<tr>
<td>Windows Extended Security</td>
<td>483</td>
</tr>
<tr>
<td>Authority and Privilege Metadata</td>
<td>484</td>
</tr>
<tr>
<td>Case Study</td>
<td>486</td>
</tr>
<tr>
<td>Working with Authorities and Privileges</td>
<td>486</td>
</tr>
<tr>
<td>Working with Data Encryption, Ownership Transfer, and Roles</td>
<td>491</td>
</tr>
<tr>
<td>Working with RCAC</td>
<td>492</td>
</tr>
<tr>
<td>Summary</td>
<td>493</td>
</tr>
<tr>
<td>Review Questions</td>
<td>494</td>
</tr>
</tbody>
</table>
### Chapter 9  Understanding Concurrency and Locking

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2 Locking and Concurrency: The Big Picture</td>
<td>500</td>
</tr>
<tr>
<td>Concurrency and Locking Scenarios</td>
<td>500</td>
</tr>
<tr>
<td>Lost Updates</td>
<td>501</td>
</tr>
<tr>
<td>Uncommitted Reads</td>
<td>502</td>
</tr>
<tr>
<td>Nonrepeatable Reads</td>
<td>503</td>
</tr>
<tr>
<td>Phantom Reads</td>
<td>504</td>
</tr>
<tr>
<td>DB2 Isolation Levels</td>
<td>504</td>
</tr>
<tr>
<td>Uncommitted Reads</td>
<td>504</td>
</tr>
<tr>
<td>Cursor Stability</td>
<td>505</td>
</tr>
<tr>
<td>Read Stability</td>
<td>510</td>
</tr>
<tr>
<td>Repeatable Reads</td>
<td>511</td>
</tr>
<tr>
<td>Changing Isolation Levels</td>
<td>512</td>
</tr>
<tr>
<td>Using the DB2 Command Window</td>
<td>512</td>
</tr>
<tr>
<td>Using the DB2 precompile and bind Commands</td>
<td>514</td>
</tr>
<tr>
<td>Using the DB2 Call Level Interface</td>
<td>514</td>
</tr>
<tr>
<td>Using the Application Programming Interface</td>
<td>516</td>
</tr>
<tr>
<td>Working with Statement Level Isolation Level</td>
<td>516</td>
</tr>
<tr>
<td>DB2 Locking</td>
<td>517</td>
</tr>
<tr>
<td>Lock Attributes</td>
<td>518</td>
</tr>
<tr>
<td>Lock Waits</td>
<td>524</td>
</tr>
<tr>
<td>Deadlocks</td>
<td>526</td>
</tr>
<tr>
<td>Lock Deferral</td>
<td>527</td>
</tr>
<tr>
<td>Lock Escalation</td>
<td>528</td>
</tr>
<tr>
<td>Diagnosing Lock Problems</td>
<td>529</td>
</tr>
<tr>
<td>Using the list applications Command</td>
<td>529</td>
</tr>
<tr>
<td>Using the force application Command</td>
<td>531</td>
</tr>
<tr>
<td>Using the Snapshot Monitor</td>
<td>532</td>
</tr>
<tr>
<td>Using Snapshot Table Functions</td>
<td>536</td>
</tr>
<tr>
<td>Using the Event Monitor</td>
<td>536</td>
</tr>
<tr>
<td>Techniques to Avoid Locking</td>
<td>536</td>
</tr>
<tr>
<td>Case Study</td>
<td>538</td>
</tr>
<tr>
<td>Exercises</td>
<td>539</td>
</tr>
<tr>
<td>Setup</td>
<td>539</td>
</tr>
<tr>
<td>Part 1: Testing Isolation CS Without CC</td>
<td>542</td>
</tr>
<tr>
<td>Part 2: Different Access Paths, Different Locking</td>
<td>543</td>
</tr>
<tr>
<td>Part 3: Simulating a Deadlock Situation</td>
<td>544</td>
</tr>
<tr>
<td>Part 4: Testing Isolation CS with CC</td>
<td>546</td>
</tr>
<tr>
<td>Part 5: Testing Isolation UR</td>
<td>547</td>
</tr>
<tr>
<td>Summary</td>
<td>547</td>
</tr>
<tr>
<td>Review Questions</td>
<td>548</td>
</tr>
</tbody>
</table>
## Contents

**Chapter 10  Maintaining, Backing Up, and Recovering Data**  553

- DB2 Data Movement Utilities: The Big Picture  553
  - Data Movement File Formats  555
  - The DB2 EXPORT Utility  557
  - The DB2 IMPORT Utility  559
  - The DB2 Load Utility  562
  - The Ingest Utility  573
  - The db2move Utility  577
- Generating Data Definition Language  579
- DB2 Maintenance Utilities: The Big Picture  580
  - The RUNSTATS Utility  580
  - The REORG and REORGCHK Utilities  582
  - The REBIND Utility and the FLUSH PACKAGE CACHE Command  584
  - Automatic Database Maintenance  585
- Database Backup, Recovery, and Roll Forward Concepts: The Big Picture  585
  - Recovery Scenarios and Strategies  586
  - Unit of Work (Transaction)  587
  - Types of Recovery  588
  - DB2 Transaction Logs  589
  - Logging Methods  596
  - Handling the DB2 Transaction Logs  601
  - Recovery Terminology  602
  - Performing Database and Table Space Backups  602
  - The Backup Files  607
  - Performing Database and Table Space Recovery  608
  - Database and Table Space Roll Forward  614
  - The Recovery History File  618
  - Database Recovery Using RECOVER DATABASE  620
- Case Study  621
- Summary  623
- Review Questions  625

**Appendix A  Solutions to the Review Questions**  629

- Chapter 1 Answers  629
- Chapter 2 Answers  630
- Chapter 3 Answers  632
- Chapter 4 Answers  633
- Chapter 5 Answers  634
- Chapter 6 Answers  635
- Chapter 7 Answers  636
- Chapter 8 Answers  638
- Chapter 9 Answers  640
- Chapter 10 Answers  641
Appendix B  Introduction to SQL  645

Querying DB2 Data  646

Derived Columns  646
The SELECT Statement with COUNT Aggregate Function  648
The SELECT Statement with DISTINCT Clause  648

DB2 Special Registers  649
Scalar and Column Functions  651
The CAST Expression  652
The FROM Clause  653
The WHERE Clause  653

Using FETCH FIRST n ROWS ONLY  653
The LIKE Predicate  654
The BETWEEN Predicate  655
The IN Predicate  655

The ORDER BY Clause  656
The GROUP BY...HAVING Clause  657
Joins  657
Working with NULLs  660
The CASE Expression  661
Adding a Row Number to the Result Set  662

Modifying Table Data  663
Selecting from UPDATE, DELETE, or INSERT  664
The MERGE Statement  666
The UNION,INTERSECT, and EXCEPT Operators  668
The UNION and UNION ALL Operators  668
The INTERSECT and INTERSECT ALL Operators  670
The EXCEPT and EXCEPT ALL Operators  670

Recursive SQL Statements  671

Appendix C  A Comparison of DB2 and Oracle Terminology  675

Product and Functionality Mapping  675
Terminology Mapping  677
DB2 Compatibility Features  680
Data Types, SQL, and Packages Support in DB2  680
PL/SQL Support in DB2  681
Concurrency Control  681
IBM Database Conversion Workbench  681
Appendix D  Diagnosing Problems  683

Problem Diagnosis: The Big Picture  683
The Help (?) Command  684
DB2 First Occurrence Data Capture (FODC)  686
   Administration Notification Log  686
db2diag.log  686
Trap Files  686
Dump Files  687
Core Files (Linux/UNIX Only)  687
DB2 Instance Level Configuration Parameters Related to FODC  687
Administration Notification Log Examples  690
db2diag.log Example  690
Tools for Troubleshooting  692
   DB2VAL  692
   DB2DIAG  692
The db2support Tool  692
The DB2 Trace Facility  693
The db2dart Tool  694
The INSPECT Tool  695
DB2COS  695
DB2PDCFG  697
DB2FODC  697
Searching for Known Problems  699

Appendix E  Resources  701

Index  707
To meet the rapidly growing demand for information, IT infrastructures must not only work faster to provide analytics, they must also work smarter with new database technology created for the era of Big Data. DB2® 10.5 with BLU Acceleration offers businesses a faster, easier and significantly more affordable approach to analytics.

This book describes the increasing interest and demand for Big Data, and introduces big data technologies such as IBM® InfoSphere® Streams, IBM InfoSphere BigInsights™, and IBM InfoSphere Data Explorer. The authors highlight the value of DB2 for data warehouse simplification.

In the last two editions, the authors received great feedback about the visual illustrations throughout the book. With this third edition of the Understanding DB2 series, this book continues to keep the visual learning style with clear explanations for every chapter. Each chapter introduces a topic with a “big picture”. Figures are used extensively and explained thoroughly. The “big pictures” are excellent for beginners to understand the concept and see how each component complements and interacts with others. They are also excellent references for database professionals at intermediate levels.

After “visually” learning the concepts in the chapters, the full case studies that follow will illustrate how to put theory into practice in real life scenarios. There are concept review questions to help you prepare for your DB2 certification exams. This provides a complete introduction as well as practical guide with study material.

This book has been fully updated with DB2 10.1 and 10.5 functions and features. New topics include introducing BLU Acceleration, Adaptive Compression, multi-temperature data management, the newly simplified DB2 portfolio, the complementary tool - IBM Data Studio, and many other new capabilities.

The authors have effectively compiled valuable information into this book that has been collected through their experience of working in the DB2 development lab as well as from working with many DB2 customers and partners globally. As with the previous editions, it will continue to earn a place on the must-read list for every DB2 professional. Enjoy the book!

Judy Huber
Vice President, Distributed Data Servers and Data Warehousing
Director, IBM Canada Laboratory
Preface

We are living in exciting times where data is being heralded as the new gold. The confluence of Big Data Technologies and Cloud Computing is enabling us to analyze vast amounts of data in ways never done before, and allowing the discovery of new information that is impacting everyone’s lives. But for those in the Information Technology (IT) field, keeping up with the skills to be successful on the job is becoming more and more challenging. Understanding new algorithms, new programming paradigms, and new technologies in general require significant time commitment. Although you cannot avoid the time investment needed on your own education, choosing books written in a clear, concise, and visual manner can help you get the most of your investment. This book was designed with this in mind, to minimize the time, money, and effort required to learn DB2 for Linux®, UNIX®, and Windows®. The book visually introduces and discusses the latest version of DB2, DB2 10.5. This version introduces and important feature, BLU acceleration technology, which is particularly important in the big data world.

Who Should Read This Book?

This book is intended for anyone who works with data, and specifically databases, such as database administrators (DBAs), application developers, system administrators, and consultants. This book is a great introduction to DB2, whether you have used it before or you are a beginner. It is also a good study guide for anyone preparing for the IBM DB2 10 Certification exams 610 (DB2 10.1 Fundamentals), or 611 (DB2 10.1 Database Administrator for Linux, UNIX and Windows).

This book will save you time and effort because the topics are presented in a clear and concise manner, and we use figures, examples, case studies, and review questions to reinforce the material as it is presented. The book is different from many others on the subject because of the following.

- Visual learning. The book relies on visual learning as its base. Each chapter starts with a “big picture” to introduce the topics to be discussed in that chapter. Numerous graphics are used throughout the chapters to explain concepts in detail. We feel that figures allow for fast, easy learning and longer retention of the material. If you forget some of the concepts discussed in the book or just need a quick refresher, you will not need to read the entire chapter again. You can simply look at the figures quickly to refresh your memory. For your convenience, some of the most important figures are provided in
color on the book’s Web site (www.ibmpressbooks.com/title/9780133461909). These figures in color can further improve your learning experience.

- Clear explanations. We have encountered many situations when reading other books where paragraphs need to be read two, three, or even more times to grasp what they are describing. In this book we have made every effort possible to provide clear explanations so that you can understand the information quickly and easily.

- Examples, examples, examples. The book provides many examples and case studies that reinforce the topics discussed in each chapter. Some of the examples have been taken from real life experiences that the authors have had while working with DB2 customers.

- Sample exam questions. All chapters end with review questions that are similar to the questions on the DB2 Certification exams. These questions are intended to ensure that you understand the concepts discussed in each chapter before proceeding, and as a study guide for the IBM Certification exams. Appendix A contains the answers with explanations.

Getting Started

If you are new to DB2 and would like to get the most out of this book, we suggest you start reading from the beginning and continuing with the chapters in order. If you are new to DB2 but are in a hurry to get a quick understanding of the product, you can jump to Chapter 2, “DB2 at a Glance: The Big Picture.” Reading this chapter will introduce you to the main concepts of DB2. You can then go to other chapters to read for further details. If you are new to DB2 but have knowledge of Oracle database products, you can review first Appendix C, “A Comparison of DB2 and Oracle Terminology.”

If you would like to follow the examples provided with the book, you need to install DB2. Chapter 3, “Installing DB2,” gives you the details to handle this task.

A Word of Advice

In this book we use figures extensively to introduce and examine DB2 concepts. Although some of the figures may look complex, don’t be overwhelmed by first impressions! The text that accompanies them explains the concepts in detail. If you look back at the figure after reading the description, you will be surprised by how much clearer it is.

This book only discusses DB2 for Linux, UNIX, and Windows, so when we use the term DB2, we are referring to DB2 on those platforms. DB2 for i®, DB2 for z/OS®, and DB2 for VSE and VM are mentioned only when presenting methods that you can use to access these databases from an application written on Linux, UNIX, or Windows. When DB2 for i®, DB2 for z/OS®, and DB2 for VSE and VM are discussed, we refer to them explicitly.

This book was written prior to the official release of DB2 10.5. The authors used a beta copy of the product to obtain screen shots, and perform their tests. It is possible that by the time
this book is published, and the product is officially released, some features and screenshots may have changed slightly.

**Conventions**

Many examples of SQL statements, XPath/XQuery statements, DB2 commands, and operating system commands are included throughout the book. SQL statement keywords are written in uppercase. For example: Use the `SELECT` statement to retrieve data from a DB2 database.

XPath and XQuery statements are case sensitive—for example: `/employee/DEPT/Id` DB2 commands are shown in lowercase mono—for example: The `list applications` command lists the applications connected to your databases.

You can issue many DB2 commands from the Command Line Processor (CLP) utility, which accepts the commands in both uppercase and lowercase. In UNIX operating systems, program names are case-sensitive, so be careful to enter the program name using the proper case. For instance, on UNIX, `db2` must be entered in lowercase. (See Chapter 2, “DB2 at a Glance: The Big Picture,” for a detailed discussion of this.)

Keywords are written in uppercase, unless the keyword is part of a command statement or syntax, or the particular program language uses lowercase.

Database object names used in our examples are shown in italic. For example: The `COUNTRY` table has a `City` column.

Italic is also used for variable names in the syntax of a command or statement. If the variable name has more than one word, it is joined with an underscore. For example: `CREATE SEQUENCE sequence_name`.

In code listings, some code lines are too long to fit the width of the page. When a code line wraps to another line, you will see a code continuation character (➥) at the beginning of the runoff line:

```
Number of pooled fenced processes         (FENCED_POOL) =
➥AUTOMATIC(MAX_COORDAGENTS)
```

Where a concept of a function is new in DB2 10.1 or DB2 10.5, we signify this with an icon as follows:

![V10](icon)

Note that the DB2 certification exams only include material of DB2 version 10.1, not version 10.5

**Contacting the Authors**

We are interested in any feedback that you have about this book. Please contact us with your opinions and inquiries at udb2book@ca.ibm.com.

Depending on the volume of inquiries, we may be unable to respond to every technical question but we’ll do our best. The DB2 forum at https://www.ibm.com/developerworks/community/forums/html/forum?id=11111111-0000-0000-0000-000000000842 is another great way to get assistance from IBM employees and the DB2 user community.
What's New

This book, though with a different title, is an update of the book *Understanding DB2 – Learning Visually with Examples* (2nd Edition), which received great reviews. Though our intention was to keep the same title and same depth in each topic, the size of the book became an issue. As more features are added into the DB2 product, more pages are required to describe them. Rather than reduce the depth of each topic, we decided to split the book. *DB2 Essentials* covers the core topics every DB2 professional should know at the beginner-to-intermediate level. More advanced concepts have been left for another book, which at the time of this writing is in the planning stage.

Since the time the second edition of *Understanding DB2: Learning Visually with Examples* was published, there have been four releases or versions of DB2 for Linux, UNIX, and Windows in that time: DB2 9.7, DB2 9.8, DB2 10.1, and now DB2 10.5; this section highlights what’s new with each of them.

The core of DB2 and its functionality remains mostly the same as in previous versions; therefore, some chapters required minimal updates. On the other hand, some other chapters such as Chapter 4, “Using Database Tools and Utilities,” required substantial changes since most DB2 GUI Tools were deprecated with DB2 9.7, and then discontinued with DB2 10.1 and replaced by IBM Data Studio.

To indicate where something has changed from the previous version of the book, or was added in DB2 10.1 or DB2 10.5, we have used the icon shown below. This is particularly useful for those who have bought the second edition of the *Understanding DB2* book and quickly want to identify what’s new.

![V10](image)

As you will see next, there is only one main feature introduced in DB2 10.5 (BLU acceleration technology) that we discuss in this book; therefore we did not use different icons to distinguish changes or additions between version 10 and version 10.5.

The following sections will briefly introduce the changes or additions in each of these new versions of DB2.

DB2 9.7

DB2 9.7 introduced many features to help administrators and developers of other relational database products migrate their databases and applications to DB2. For example, with DB2 9.7 a migration from an Oracle database to DB2 that would take months in the past, could now be performed in a few hours or days. This was possible because DB2 9.7 introduced several data types, and support for non-standard SQL statements used in Oracle. In addition, the CLPPlus tool was introduced which has a very similar interface and behavior to Oracle’s SQL*Plus. Moreover, Oracle’s PL/SQL language often used in stored procedures could be easily understood in DB2; therefore Oracle’s stored procedures could run with minimal or no modification in DB2. Appendix C, “A Comparison of DB2 and Oracle Terminology,” has a section with more information about this. Other improvements in this release included compression, pureXML®, and security enhancements.
DB2 9.8

DB2 9.8 introduced pureScale® technology, a solution architected based on DB2 on the mainframe data-sharing technology. With pureScale, different DB2 servers share the same data in a cluster environment. DB2 servers can be added to the cluster as the data grows, allowing for scalability, but also, extreme availability. In this book we discuss the basic concepts of pureScale.

DB2 10.1

DB2 10.1 introduced many features and enhancements in different areas. Most of these enhancements are discussed in detail in the book:

- Adaptive compression, for deep compression using dictionaries at the page and table levels.
- Time Travel Query, which allows users to query data in the past, the present or the future.
- Multi-temperature data management, ideal for data warehousing environments where data is classified based on how often it is accessed; thus, assigning the most frequently used data (hot data) to faster devices, and the least frequently used data (cold data) to slower devices.
- Row and Column Access Control (RCAC), which provides security granularity at both, row and column levels.

DB2 10.5

DB2 10.5 introduces new packaging of the product to fit different needs at different price points. One single image is created for most editions, as opposed to one different image per edition. This means that to upgrade from one edition to the other, you don’t need to uninstall the previous edition of DB2, simply apply the license of the new edition, and any feature specific to the new edition would be unlocked. This is particularly helpful in environments where the company’s IT policy required safety procedures in place for new installations of a product. With this approach, there is no new installation required.

Probably the key feature of DB2 10.5 is BLU Acceleration technology, a revolutionary approach of storing data rows in columnar fashion. This is ideal for data warehousing environments, and allows for performance improvements in order of magnitude for analytic workloads. BLU Acceleration is a technology that enables users to work with big data as it dramatically helps with performance. It is also remarkable the ease in which this technology can be implemented by users. All complexities are hidden from regular users who don’t even need to create indexes.

One common denominator in all the new features and changes made to DB2 in these four new versions or releases is that many of the new features and functions were developed to make your life easier!
Acknowledgments

Raul and Clara would like to thank Cristian Molaro and Kshitij Kohli for their extensive technical review of the book. Their suggestions and corrections were invaluable.

Steven Stansel, Susan Visser, and Mary Beth Ray provided guidance and invaluable help throughout the whole process of planning, writing, and publishing the book. Without their help, this book would never have been completed as smoothly as it has been.
About the Authors

Raul F. Chong is a Senior DB2, Big Data and Cloud Program Manager and Technical Evangelist based at the IBM Canada Laboratory. He leads the development and design of several offerings for the Information Management (IM) brand of IBM, with the goal of increasing awareness and growing communities around IBM IM products, such as IBM InfoSphere® BigInsights™, IBM InfoSphere Streams, DB2 database software, IBM Data Studio, InfoSphere Data Architect, and pureQuery® technology. As part of the IM Cloud Computing Center of Competence at the Toronto Lab, Raul leads the development and deployment of projects by the community using DB2 on the Cloud, such as bigdatauniversity.com, and db2oncampus.org. As a technical evangelist, Raul travels worldwide delivering presentations and workshops targeting customers, IBM business partners and the Academia. Raul develops and leads the development of collateral material such as articles, books, videos, courses, and DVDs that help educate users in IBM IM products. He has also participated actively in the development of training material and offerings of IBM IM Certification programs. Raul joined IBM in 1997 and has worked as a DB2 consultant, DB2 technical support specialist, and DB2 Information Developer. Raul has summarized many of his DB2 experiences through the years in the first and second editions of the book Understanding DB2—Learning Visually with Examples for which he is the lead author. He has also co-authored the book DB2 SQL PL Essential Guide for DB2 UDB on Linux, UNIX, Windows, i5/OS, and z/OS (ISBN 0131477005), and other books that are part of the DB2 on Campus book series. In his spare time, Raul enjoys playing with his two little daughters. Raul is fluent in Spanish as he was born and raised in Peru, but he keeps some of the Chinese traditions from his grandparents. He also enjoys reading history and archeology books.

Clara Liu was recently appointed to be the Program Manager of IBM Cross Brand Technical Initiatives. She manages leading edge strategic projects across IBM brands. Her prior management role was with the DB2 Planning team. Her previous consulting experience gave her an insight and solid understanding of customers’ needs. Based on market demand, competition pressure, and objectives of maximizing return of investment, she drives software enhancements into the product with the right balance between leading edge technology and business needs. In many DB2, Warehouse, PureData™ Systems versions and releases, Clara held key responsibilities such as planning product enhancements, managing product offer portfolio and license entitlement,
making software available on fulfillment systems for customers, and planning for smooth migration paths for customers when products reach end of life. Over the years, she delivered many product hands-on and video demonstrations with her team. Those are great assets for the IBM Sales and Marketing team and customers who want to ‘see’ and ‘play’ with the technologies. Due to the board involvement in numerous phases of the database product life cycle, Clara has established great networking within the development organization as well as across the business teams. As a mother of two, Clara had coauthored five books (including this one), all focusing in her technical expertise, DB2 for Linux, UNIX and Windows.
This page intentionally left blank
Using Database Tools and Utilities

How do you work with DB2? How do you issue SQL and/or XQuery statements and enter DB2 commands? Are there graphical tools that can make your administration tasks easier? This chapter provides the answers to all of these questions. DB2 provides a wide range of tools, both graphical and command-driven, to help you work with DB2.

In this chapter, you learn about

• DB2 command line tools
• IBM Data Studio
• IBM Data Studio Web Console
• Set up tools
• DB2 information tools
• DB2 problem determination tools

Database Tools: The Big Picture

DB2 comes with a comprehensive and complimentary set of tools for you to perform basic database administration and development tasks. Figure 4.1 shows a high level overview of tools available with DB2. They are categorized into command line tools, graphical tools, and other tools to assist you with setup. In most cases you can perform the same DB2 commands, SQL statements, and XQuery statements using the command line and graphical tools.
Chapter 4  Using Database Tools and Utilities

IBM Information Center

DB2 on Linux, UNIX, or Windows

Command Line Tools
- Command Line Processor (CLP)
- Command Line Processor Plus (CLPPlus)
- Command Window (Windows only)

Other Tools
- First Steps
- Configure DB2 .NET Data Provider
- Default DB2 and Database Client Interface Selection Wizard
- Replication Center
- Check for DB2 Updates

Linux, UNIX, or Windows

*DB2 install is not necessary

Graphical Tools
- IBM Data Studio
  (Linux and Windows only)
- IBM Data Studio Web Console
- Advanced tools included in DB2
  Advanced and Developer editions only
  - InfoSphere Optim Performance Manager Extended Edition
  - InfoSphere Optim Configuration Manager
  - InfoSphere Optim Query Workload Tuner
  - InfoSphere Optim pureQuery Runtime for LUW
  - InfoSphere Data Architect
    (Linux and Windows only)

Support Portal

Figure 4.1  Database tools the big picture

NOTE
The Control Center tools and all related components such as wizards and advisors were discontinued in DB2 10. IBM Data Studio can be used to manage DB2 environment and DB2 data-centric applications. IBM Data Studio comes with every DB2 edition.

The Command-Line Tools

All DB2 operations are invoked by DB2 commands, SQL statements, or XQuery statements. For example, to back up a database, you use the BACKUP DATABASE command. To create a table, you use the CREATE TABLE SQL statement. To parse an XML document you use the FLWOR expression. All of these commands, SQL statements, and XQuery statements can be entered using the command-line tools.

The command-line tools consist of the Command Line Processor (CLP), the Command Line Processor Plus (CLP Plus), and the Command Window (Windows platform only). Because they are command-driven, you must have some knowledge of DB2 commands and SQL statements to use them. Figure 4.2 demonstrates the relationship between them. Compare each line in the Windows machine versus the Linux/UNIX machine. The equivalent line in each machine has been aligned in the figure.
NOTE
In this chapter, we use the generic term **DB2 commands** to refer to DB2 system commands, DB2 CLP, and DB2 CLP Plus commands. When a section is only applicable to a given type of command, it is indicated. Refer to section “SQL Statements, XQuery Statements, and DB2 Commands,” for an explanation about the differences between these commands.

Windows Machine

**Windows Command Prompt**

- `C:` `dir`
- `C:` `db2start`
- `C:` `clpplus`

**DB2 Command Window**

- `C:` `dir`
- `C:` `db2start`
- `C:` `db2 <CLP Command>` (To invoke the CLP in non-interactive mode)
- `C:` `db2` (To invoke the CLP in interactive mode)

**DB2 Command Line Processor (CLP) in interactive mode**

- `db2 => <CLP Command>`
- `db2 => list applications`
- `db2 => select * from department`
- `db2 => xquery <a>hello</a>`

**DB2 CLP Plus**

- SQL>`CONNECT db2inst1@localhost:50000/sample`
- SQL>`START resourceReport.sql`
- SQL>`RUN`
- SQL>`SPOOL resourceReport.out`
- SQL>`SET LINESIZE 100`

Linux and UNIX Machine

**Linux / UNIX Shell**

- `/home/user1 $ ls`
- `/home/user1 $ db2start`
- `/home/user1 $ db2 <CLP Command>` (To invoke the CLP in non-interactive mode)
- `/home/user1 $ db2` (To invoke the CLP in interactive mode)

**DB2 Command Line Processor (CLP) in interactive mode**

- `db2 => <CLP Command>`
- `db2 => list applications`
- `db2 => select * from department`
- `db2 => xquery <a>hello</a>`

**DB2 CLP Plus**

- SQL>`CONNECT db2inst1@localhost:50000/sample`
- SQL>`START resourceReport.sql`
- SQL>`RUN`
- SQL>`SPOOL resourceReport.out`
- SQL>`SET LINESIZE 100`

Figure 4.2  The CLP, CLP Plus, and the DB2 Command Window

The DB2 Command Window

The DB2 Command Window is only available on Windows; this is due to some architecture differences in Windows versus Linux and UNIX. If you are familiar with the Linux and UNIX platforms, you can think of the Command Window on Windows as the Linux/UNIX shell. Figure 4.2 illustrates this: The commands and statements inside the DB2 Command Window box to the left of the figure are equivalent to the ones inside the Linux/UNIX shell box on the right.

To start the Command Window, click **Start > Programs > IBM DB2 > DB2COPY1 (Default) > Command Window**. Alternatively, to invoke the Command Window from a
Windows command prompt, issue the command `db2cmd`. This command spawns another window that displays **DB2 CLP** in the title bar. Note that the Command Window looks like any Windows command prompt except for this title bar.

From the Windows command prompt, you can perform operating system commands and DB2 system commands but not DB2 CLP commands, DB2 CLPPlus commands, SQL statements, or XQuery statements. However, you can perform all of these from a DB2 Command Window.

Refer to the DB2 Information Center, under Database administration > Interfaces > Commands, for a complete list of different types of commands. Figure 4.2 shows a few examples. From the Windows command prompt you can execute the following:

**Operating system commands:**
- `dir`

**DB2 system commands:**
- `db2start`

In the DB2 Command Window, you can perform these commands as well as DB2 CLP commands and SQL statements:

**DB2 CLP commands:**
- `db2 list applications`

**SQL statements:**
- `db2 SELECT * FROM department`

**XQuery statements:**
- `db2 "xquery <a>hello</a>"`

If you try to execute a CLP command, SQL statement, or XQuery statement from a Windows command prompt, you receive the following error as illustrated in Figure 4.3:

```
DB21061E Command line environment not initialized
```

In the figure, you also see how the same statement works from the DB2 Command Window after it is invoked with the `db2cmd` command.

**Figure 4.3** Invoking the DB2 Command Window from a Windows command prompt
The DB2 Command Line Processor

The DB2 Command Line Processor (CLP) is an application written in the C language that contains embedded SQL. It provides you with a text-based interface to the DB2 engine that lets you issue CLP commands, SQL statements, and XQuery statements. The CLP executable is called `db2`, and it is stored under the DB2_install_directory/sqllib/bin directory.

**NOTE**

We recommend you learn how to use the Command Line Processor, as it is the common tool available with all DB2 versions and clients.

Methods to Work with the CLP

There are three ways to issue a DB2 command or SQL statement with the CLP: interactive mode, noninteractive mode, and noninteractive mode using a file as input. These methods are discussed in the following sections.

**Method 1: Interactive Mode**

You start the CLP in interactive mode by clicking **Start > Programs > IBM DB2 > DB2COPY1 (Default) > Command Line Processor**. Alternatively, from the Command Window or Linux/UNIX shell, you start the CLP in interactive mode by entering `db2` and pressing **Enter**, as shown in Figure 4.4.

![The Command Line Processor in interactive mode](image)

Figure 4.4  The Command Line Processor in interactive mode

After you invoke the CLP in interactive mode, a few messages appear on the screen, and then your command prompt changes to `db2 =>`. This prompt indicates that you are in interactive mode and you can type any DB2 CLP command, SQL statement, or XQuery statement.
Table 4.1 lists some common CLP interactive mode commands. The underlined letter in the command shows the shortcut that you can use to invoke the command. Figure 4.5 shows a few examples of the commands in Table 4.1 in action.

Table 4.1 Useful CLP Commands for Working with the CLP in Interactive Mode

<table>
<thead>
<tr>
<th>Command</th>
<th>Explanation</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>history</td>
<td>Lists the last 20 commands entered and prefixes each with a number. The maximum number of commands kept in memory can be customized with the DB2 registry variable DB2_CLP_HISTSIZE (see Chapter 5 for information about DB2 registry variables).</td>
<td>History</td>
</tr>
<tr>
<td>runcmd &lt;n&gt;</td>
<td>Re-executes command number n from the list given by the history command. If n is not specified (or n = -1), the previous command is invoked.</td>
<td>To re-execute the third command in the history list: r 3</td>
</tr>
<tr>
<td>edit &lt;n&gt;</td>
<td>Edits the command number n using an editor defined by the DB2 registry variable DB2_CLP_EDITOR. If not set, this uses the vi editor on Linux/UNIX and Notepad on Windows.</td>
<td>To edit the fifth command in the history list: e 5</td>
</tr>
<tr>
<td>Exclamation mark (!)</td>
<td>This is the escape character that enables you to issue operating system commands from within the CLP interactive mode.</td>
<td>!dir</td>
</tr>
</tbody>
</table>

Figure 4.5 Examples of CLP commands in interactive mode
Method 2: Noninteractive Mode

Working with the CLP in noninteractive mode is equivalent to working with the DB2 Command Window (on Windows) or the Linux/UNIX shell. If you start the CLP in interactive mode, entering the `quit` command takes you to the CLP in noninteractive mode. In this mode you need to prefix the CLP command or SQL statement with `db2`, which calls the `db2` executable. Otherwise, you receive an error. For example

```sql
db2 connect to sample
db2 list applications all
db2 select * from employee
db2 "xquery <name>Raul</name>"
```

Using this method you can execute operating system commands in addition to DB2 commands, SQL statements, and XQuery statements from the same window or session.

In practice, many DB2 users prefer to work in the CLP noninteractive mode environment because they can use some shortcut key strokes, such as pressing the up arrow key to repeat the last commands on Windows or taking advantage of operating system mechanisms like piping the output of the CLP to the `more` command on Linux and UNIX to display the output in portions.

Every time you issue the `db2` executable, a “CLP session” is created where a front-end process is invoked. This takes the rest of the statement as input and then closes the process. For example, you issue the following command:

```sql
db2 list db directory
```

db2 invokes a CLP front-end process that takes `list db directory` as input. After the CLP digests this command, it implicitly issues the `quit` command to end the CLP front-end process. The front-end and back-end processes are discussed in more detail later in this chapter. Figure 4.6 shows the CLP in noninteractive mode.

When invoking the CLP in noninteractive mode, enclosing the CLP command, SQL statement, or XQuery statement double quotes (""") might be required if these contain special characters that the operating system interprets as wildcard characters. This is especially important on Linux and UNIX platforms. If double quotes are not used, the error message that DB2 reports vary depending on where the wildcard character is used in the statement. For example, you issue this statement:

```sql
db2 select * from employee
```

You might receive the following error message because the asterisk (*) is a wildcard character:

```
SQL0104N An unexpected token "*" was found following "select "
```

To avoid parsing errors, use double quotes as shown here:

```sql
db2 "select * from employee"
```
A more deceiving example occurs when you use the greater than (>) character. Here is an example:

```
db2 select lastname from employee where salary > 10000
```

The command is first parsed by the operating system, which interprets > 10000 as the redirection of the output to the file 10000. After executing the statement just given, your current directory has a new file with the name 10000 containing a DB2 syntax error message because only select lastname from employee where salary was passed to DB2. Again, to resolve this problem, make sure to enclose the statement in double quotes.

```
db2 "select lastname from employee where salary > 10000"
```

This is particularly important as well when working with XQuery because XML documents use tags enclosed in angle brackets (< and >), which the operating system interprets completely differently.

**Method 3: Noninteractive Mode Using a File as Input**

The CLP can use a file containing one or more CLP commands or SQL statements and process them one after the other. This is ideal to develop DB2 database scripts. For example, Figure 4.7 shows the contents of the file myInput.txt, which we use as input to the CLP.
The Command-Line Tools 145

Figure 4.7  Input file to be used by the CLP

To execute this DB2 script file, the -f command option (for file) followed by the file name is required to indicate to the CLP that this file contains the input. (CLP command options are described in detail in the next section.) If the input file contains a statement terminator character, the -t command option (for terminator) is required to indicate a terminator character is present. By default, the statement terminator is a semicolon (;). If you want to use a different terminator, the -d character option (for delimiter) indicates which delimiter character is being used as the terminator. Use the -v option (for verbose) to echo the command you are executing. Figure 4.8 provides an example of invoking the CLP using these command options.

Figure 4.8  Invoking the CLP in noninteractive mode using a file as input

NOTE
The input file must be a text file. Be aware that invisible characters can cause DB2 CLP fail to process the file. If using the Notepad application on Windows, for example, saving the text file with Unicode encoding rather than ANSI encoding causes the following error:

DB21007E End of file reached while reading the command.
If you prefix each of the CLP commands with `db2` (the CLP executable) in a file and remove the terminator characters, you are effectively converting this file into an operating system script rather than a DB2 script. Depending on the operating system, you might have to make additional modifications. For example, on Windows, you need to use `rem` for comments. You might also need to change the file name so that the `.bat` extension is used. Figure 4.9 shows this for the file `myOS_Input.bat`.

![Figure 4.9 Invoking DB2 CLP commands and SQL statements in a Windows script file](image)

On Linux and UNIX platforms, use the pound sign (`#`) for comments. You might also need to change the permissions of the file so that it is executable. Typically you can use this command to change the file permissions:

```
chmod +x myOS_Input.txt
```

Figure 4.10 shows the same script for a Linux or UNIX platform.

![Figure 4.10 Invoking DB2 CLP commands and SQL statements in a Linux/UNIX script file](image)
NOTE
DB2 scripts do not accept parameters, but operating system scripts do. In other words, if you need to invoke your scripts with parameters, you need to use operating system scripts.

CLP Command Options
The CLP is just another program designed to interact with DB2. Like many other programs, the CLP has been designed to accept several parameter options. The CLP command list command options displays the available CLP command option parameters (see Figure 4.11).

Figure 4.11  CLP command options

To turn on an option, use a dash (-) in the command line. To turn off an option, use a plus symbol (+). Some options are on (or off) by default. For example, to enable auto-commit, invoke the CLP as follows.

db2 -c insert into employee (firstnme) values ('Raul')
After you execute this command, a `COMMIT` statement is automatically issued because auto-commit is enabled. (As you can see in Figure 4.11, the Auto-Commit option was already on by default, so including `-c` in this example is not necessary.)

To disable auto-commit, invoke the CLP as follows.

```
db2 +c insert into employee (firstname) values ('Raul')
```

Note that specifying a command option in the `db2` command applies only to that session of the CLP. Issuing the `db2` command without an option uses the default command option values, or the ones contained in the `DB2OPTIONS` registry variable, which we discuss later in this section.

You can also change a command option when working with the CLP in interactive mode using the following command:

```
update command options using option1 value1 option2 value2 ...
```

Figure 4.12 shows an example where the `v` option (verbose) is used. This option causes the command or statement to be repeated or echoed when executed as discussed earlier. In Figure 4.12, note that the `SELECT * FROM department` statement is echoed.

![Figure 4.12 The CLP in interactive mode](image)

If you would like the changes to your CLP options to be effective across all your CLP sessions, you can set the `DB2OPTIONS` registry variable with the desired options. In the command shown next, the `DB2OPTIONS` registry variable is set so that any command executed is echoed (`-v` option), and the output is spooled in the file `myfile.log` (`-z myfile.log` option). The changes take effect immediately for the current session and for any other new CLP sessions you start.

```
db2set db2options="-v -z myfile.log"
```
To reset the values to the default, issue this command:

```
db2set db2options=
```

To display all defined values for the current instance:

```
db2set -all
```

DB2 registry variables are explained in detail in Chapter 5, “Understanding the DB2 Environment, DB2 Instances, and Databases.”

### Obtaining Help Information from the CLP

One of the most useful CLP commands is the help command represented by a question mark (?). This command provides help on SQL error codes (SQLCODE), DB2 messages, and CLP command syntax. For example

```
db2 ? SQL0104N
```

```
db2 ? DB21004E
```

```
db2 ? list applications
```

In addition, using the help command by itself displays the entire list of CLP commands, as shown in Figure 4.13.

---

**Figure 4.13** Output of the command `db2 ?`
**NOTE**
The help (?) command can display CLP command syntax, but not SQL statement syntax. Refer to the DB2 Information Center for SQL statement syntax.

Figure 4.14 shows more examples of the help (?) command.

**Using Line Continuation**
There are two ways to use line continuation from the CLP: with the backslash character and with the delimiter terminator character.
Method 1: Using the Backslash (\) Character

You can use the backslash (\) character in either interactive or noninteractive mode. Figure 4.15 first shows an example of using the interactive mode, followed by an example of using noninteractive.

![Figure 4.15](image)

Notice that after entering \ and pressing Enter, the prompt changes to

```
db2 (cont.) =>
```

Method 2: Using a Delimiter Terminator Character with the CLP in Interactive Mode

Using this method, the CLP is invoked in interactive mode using the terminator delimiter option. For example

```
db2 -td!
```

After entering this command and pressing Enter, the CLP is invoked in interactive mode. You can wrap commands onto multiple lines until you type the terminator character, which is the exclamation mark (!) in the example shown in Figure 4.16.
Figure 4.16 Line continuation in the CLP using a delimiter termination character in interactive mode

Use this method when you have statements that include carriage returns. If you copy and paste one of these statements into the CLP, the carriage returns cause the statement to continue in another line, which is acceptable because the CLP processes the command after the terminator character is entered.

The following statement has one carriage return character after `staff` and one after `Edwards`; therefore, use a delimiter termination character described in method 2 to start the DB2 CLP in interactive mode:

```
select * from staff
where name = 'Edwards'
and job = 'Sales'
```

After you copy and paste the statement into the CLP, enter the terminator character and press Enter to execute it. Remember that the termination character is an exclamation mark (!) in this example.

The CLP Front-End and Back-End Processes
The CLP has both front-end and a back-end processes. The front-end allows you to perform actions without connecting to a database. For example, issuing the command `db2 list db directory` does not require a connection to a database.
The back-end process is needed when you perform actions against a database. It is created when you connect to the database in a CLP session and can be identified by the application name db2bp. Figure 4.17 shows the output of the list applications command, which shows this thread, indicating a connection to the SAMPLE database.

![Figure 4.17 The CLP back-end process](image)

To remove the connection to a database, issue the connect reset statement, the terminate command, or the disconnect statement. Connect reset and terminate work even if the process is in the middle of a unit of work. Disconnect only works when there is no active unit of work. Closing a window or session without previously issuing a terminate command closes the CLP application and front-end process and removes the connection to the database, but it does not guarantee that the back-end process will be terminated.

**NOTE**

The terminate command is the only one that guarantees the back-end process is indeed terminated. Even if the list applications command does not display the db2bp back-end process running, use the terminate command to be certain.

It is important to make sure that the back-end process is terminated because in some circumstances, a change to a parameter, environment variable, or DB2 registry variable does not take effect until this is performed.

**NOTE**

We recommend issuing a terminate command before a db2stop command. This prevents the back-end process from maintaining an attachment to an instance that is no longer active.
The DB2 Command Line Processor Plus

The DB2 Command Line Processor Plus (CLPPlus) is a command-line user interface that enables you to

- Execute Operating system commands
- Execute DB2 system commands
- Develop, edit, and execute SQL statements and XQuery statements
- Compile and run DB2 stored procedures and functions
- Work with scripts and run command-line reports
- Support SQL*Plus scripts that many DBAs and application developers are familiar with

To start the CLPPlus, on Windows 7, click Start > Programs > IBM DB2 DB2COPY1 (Default) > Command Line Processor Plus. Alternatively, issue the command `clpplus` on the Windows command prompt or the Linux and UNIX shell. You get the SQL> prompt as shown in Figure 4.18.

Take a look at the program icon on the top left corner of the CLPPlus window. The icon is a Java application icon. That’s right. CLPPlus is a Java application. CLPPlus requires Java 1.5 or later to execute. In DB2 10, Java 1.7 is installed and its path is set up with DB2 installation. If you encounter CLPPlus start-up issue, ensure Java is in your PATH.

![Figure 4.18 The CLPPlus Window](image-url)
CLPPlus can run both OS and DB2 commands. To run OS commands, simply use the `!` (exclamation mark) operator. Figure 4.19 illustrates how to use the `ls` and `grep` commands in CLPPlus.

![Figure 4.19 Use of the `!` operator in CLPPlus](image)

To work with the DB2 database from the CLPPlus, you need to first connect to the database. In Figure 4.20, it shows you two methods to connect to the database in CLPPlus. The first example is simply to enter the `CONNECT` command. CLPPLUS prompts you for the database name, hostname, port number, user ID, and password. You may also enter all these information in the `CONNECT` command as follows.

```
CONNECT demoadm@bob:50001/sample
```

To disconnect from the database, enter `DISCONNECT`.

**NOTE**

CLPPlus supports many commands. Refer to the DB2 Information Center for the list of commands and usage example.
Chapter 4  Using Database Tools and Utilities

Working with the SQL Buffer

The SQL buffer is an in-memory working area where CLPPlus keeps copies of the most recently entered SQL statements or SQL Procedural Language (SQL PL) block. CLPPlus provides many commands to help manage the SQL buffer.

Let’s use an example to demonstrate how to work with the SQL buffer. In Listing 4.1, you can see that a compound statement is used. A compound statement is bound by the keywords BEGIN and END that contain multiple statements. In this example, it uses the DECLARE variable statement that is part of SQL PL, a SET statement to assign a value to the variable, and an INSERT statement. If you are following the example on your system, you need to first create the CUSTOMER_STATISTICS table as shown in Listing 4.2.

Listing 4.1 Example of a Compound Statement Stored in File sqlpl.txt

```
BEGIN ATOMIC

DECLARE v_custCount INTEGER;
-- find out customer count and store value in variable v_custCount
SET v_custCount = (SELECT COUNT(*) FROM customer);
```

Figure 4.20  Using the CONNECT command in CLPPlus
The compound statement is stored in the `sqlpl.txt` file. First, load the file into the SQL buffer with the `GET` command. Then `RUN` the script (see Figure 4.21).

After reviewing the output here, you decided to print the customer count returned from the compound statement. You can update the script from the CLPPlus using the `EDIT` command. The default editor is used. In this example, Notepad is used.

The line `CALL DBMS_OUTPUT.PUT_LINE` (highlighted in Figure 4.22) is added to print an output comment on the screen after the `INSERT` statement is executed successfully.
After you save the file and exit the editor, the SQL buffer is updated with the new version of the script.

To display the output comment you just added to the standard output (that is the screen), you need to `SET SERVEROUTPUT ON`. The output from the `DBMS_OUTPUT` message buffer is redirected to the standard output. Then run the script again with the `RUN` command. You now get the customized output as shown in Figure 4.23.
Formatting CLPPlus output

CLPPlus has a lot of options for working with reports. Here is a simple query to display the employee information. As you can see in Figure 4.24, the output is wrapped, and it is hard to read. You can improve the appearance of the report by using some CLPPlus options to format and customize the output of the query.

A few formatting options are used as described here. See Figure 4.25 for the customized output.

- Set the line size to 120 characters for the output the better fit the screen:
  ```
  SET LINESIZE 120
  ```

- Apply formatting rules to the `salary`, `comm`, and `bonus` columns. Add dollar signs in front of the values and add a comma at the unit of thousand:

  ```
  COLUMN salary FORMAT $999,999.99
  COLUMN comm FORMAT $999,999.99
  COLUMN bonus FORMAT $999,999.99
  ```

- Do not print the column `firstname`:

  ```
  COLUMN firstname NOPRINT
  ```

- Display the values of the `lastname` column right-justified:

  ```
  COLUMN lastname JUSTIFY RIGHT
  ```
IBM Data Studio

IBM Data Studio is included in every DB2 edition. IBM Data Studio provides a single integrated environment for database administration and application development. You can perform tasks that are related to database modeling and design, developing database applications, administering and managing databases, tuning SQL performance, and monitoring databases all in one single tool. It is an ideal tool that can greatly benefit a team environment with different roles and responsibilities.

IBM Data Studio comes in three favors: full client, administration client, and web console. The full client includes both the database administrative and the application development capabilities. The development environment is Eclipse-based. This offers a collaborative development environment by integrating with other advanced Eclipse-based tools such as InfoSphere Data Architect and InfoSphere Optim pureQuery Runtime. Note that some of the advanced InfoSphere tools are only included in the DB2 Advanced editions and the DB2 Developer Edition. You can also separately purchase the advanced tools.

The administration client is a subset of the full client. It still provides a wide range of database administrative functionality such as DB2 instance management, object management, data management, and query tuning. Basic application development tasks such as SQL Builder, query...
formatting, visual explain, debugging, editing, and running DB2 routines are supported. Use the full client for advanced application development features.

The web console, as the name implies, is a web-based browser interface that provides health monitoring, job management, and connection management.

**NOTE**
IBM Data Studio also provides collaborative database development tools for DB2 for z/OS, DB2 for i, Informix, and other non-IBM databases. To see a list of IBM Data Studio features by data server, refer to the documentation at http://www.ibm.com/support/docview.wss?uid=swg27022148.

**IBM Data Studio Workspace and the Task Launcher**
When you have successfully installed the IBM Data Studio, you are asked to provide a workspace name. A **workspace** is a folder that saves your work and projects. It refers to the desktop development environment, which is an Eclipse-based concept.

Task Launcher is displayed, which highlights the following category of tasks:

- Design
- Develop
- Administer
- Tune
- Monitor

Each category is described in more detail in its own tab. Click any tab, and you see the key and primary tasks listed in the box on the left. See Figure 4.26 to get an idea on how to navigate the Task Launcher.

As an example, the figure shows you the Develop tasks. You can find the key development tasks on the left. On the top right, it lists more tasks related to development. On the bottom right, IBM Data Studio provides a few documentation links where you can learn more about development. Where appropriate, it also suggests the advanced tools available in the InfoSphere Optim portfolio that apply to the task you have selected.
Connection Profiles

Every task you were to perform against a database requires to first establish a database connection. To connect to a database from IBM Data Studio, open the **Database Administration** perspective. On the top right corner, click the **Open Perspective** icon and select **Database Administration**.

On the **Administration Explorer**, right-click the white space or under the **New** menu, select **New Connection** to a database. From the **New Connection** window, you see that you can use the IBM Data Studio to connect to different IBM data sources, as well as non-IBM data sources. Select the database manager and enter the necessary connection parameters. Figure 4.28 shows an example.
Open the Database Administration perspective

Pull down the JDBC driver drop-down menu, and you can select the type of JDBC driver to use. JDBC type 4 driver is used by default.

Use the Test Connection button to ensure the connection information you enter is valid. Click Finish.

At this point, you have created a connection profile. Connection profiles contain information about how to connect to a database such as indicating the type of authentication to be used when connecting the database, specifying default schema, and configuring tracing options. Other team members can import the connection profiles to their own IBM Data Studio and be able to deploy a set of consistent connection settings.
To update the connection profile, right-click the database and select **Properties**. Properties for the database are displayed as shown in Figure 4.29.

**General Database Administration Tools**

There are few other useful administration tasks available in the menu illustrated in Figure 4.29.

The **Manage Connection** function enables you to rename the connection profile, delete the connection profile, change the user ID and password, and duplicate the profile. The **Back Up and Restore** function enables you to setup a database or table space backups. In the appropriate editor, you can specify the type of backup, location of the backup images, and performance options for the backup. Database backup and recovery is discussed in Chapter 10, “Maintaining, Backing Up, and Recovering Data.”
The Set Up and Configure function enables you to configure the database. Database configuration and this IBM Data Studio function are covered in detail in Chapter 5. Notice from the menu, you can launch the Configure Automatic Maintenance editor. DB2 provides automatic maintenance capabilities for performing database backups, reorganizing tables and indexes, and updating the database statistics as necessary. The editor enables you customize the automatic maintenance policy (see Figure 4.30).

The Manage Database function enables you to start and stop the database. In DB2, that means activating and deactivating the database. Activating a database allocates all the necessary database memory and services or processes required. Deactivating a database releases the memory and stops DB2 services and processes.

The Monitor function launches the IBM Data Studio Web Console. Refer to the section, “IBM Data Studio Web Console,” for introduction of the tool.

Figure 4.29  Updating the connection profile
The *Generate DDL* function uses the DB2 command-based tool `db2look` to extract the Data Definition Language (DDL) statements for the identified database objects or the entire database. This function and tool come handy when you want to mimic a database, a set of database objects, or the database statistics to another database. As a result of the Generate DDL function in IBM Data Studio or the DB2 command `db2look`, you receive a DDL script. The script contains statements to re-create the database objects you have selected. See Figure 4.31 for a reference of the types of statements you can generate using the IBM Data Studio.

For complete options for the DB2 command `db2look`, refer to the DB2 Information Center.
The `Start Tuning` function configures the database to enable query tuning. You might receive a warning indicating that you need to activate the InfoSphere Optim Query Workload Tuner (OQWT) license for advanced tuning capability. Note that IBM DB2 Advanced Enterprise Server Edition comes with OQWT. Follow the instructions to apply the product license or click `Yes` to configure the database server for tuning with the features complementary in the IBM Data Studio.

When the database is configured to use the tuning advisors and tools, you are presented with the Query Tuner Workflow Assistant, as shown in Figure 4.32.
From the **Query Tuner Workflow Assistant**, you can obtain a statement from various sources and tune the statement. In the Capture view, it gives you a list of sources where you can capture the statements. Figure 4.33 shows an example on capturing the SQL statements from the Package Cache. This example captures over 100 statements. Right-click the statement in which you are interested and select **Show SQL statement** or **Run Single-Query Advisors and Tools on the Selected Statement**.

Run the query advisors and tools on the selected statement. You can now enter the Invoke view. The tool collects information and statistics and generates a data access plan (see Figure 4.34).
Figure 4.33  Capturing SQL statements for tuning

Figure 4.34  Tune query in progress
When the query tuning activities are complete, you are brought to the Review view. It presents you the analysis results and an advisor recommendation, such as the one shown in Figure 4.35. The tool documentation recommends gathering and re-collecting all of relevant statistics of the query.

You can also review the access plan graph generated by the DB2 explain function (see Figure 4.36 for an example). Remember to save the analysis for future references and compare them if needed.

The Manage Privileges function allows you to grant database privileges to the users. Refer to Chapter 8, “Implementing Security,” for details about privileges and database access controls.
IBM Data Studio consolidates the database administration and database development capabilities. From the Task Launcher – Develop, you find a list of key development tasks such as creating and running SQL statements, debugging stored procedures, and user-defined functions (UDFs). Each task brings you to a tool that helps you accomplish it.

**SQL and XQuery Editor**

The SQL and XQuery editor helps you create and run SQL scripts that contain more than one SQL and XQuery statements. To launch the editor, open the Data Project Explorer; under SQL Scripts select New > SQL or XQuery Script. As shown in Figure 4.37, a sample SQL script is entered. You can configure the run options for the script.

---

Figure 4.36  Sample access plan graph
The editor formats the SQL statements nicely and provides syntax highlights for easier reading as you enter the SQL statements. The functionality content assist is also very useful. It lists all the existing schemas in the database so that you can just select one from the drop-down menu. The editor also parses the statement and validates the statement syntax. You can validate the syntax in scripts with multiple database parsers and run scripts against multiple database connections.

**SQL Query Builder**

The SQL Query Builder enables you to create a single SQL statement, but it does not support XQuery. As the name implies, the tool helps you build an SQL statement. It helps you look at the underlying database schema or build an expression, as shown in Figure 4.38.
**NOTE**

To get to the SQL Query Builder, click the Data perspective button available at the top right of the window. In the Data Project Explorer view, expand the data profile, right-click the **SQL Scripts** folder, and then click **New > SQL or XQuery Script**. This launches the SQL Query Builder.

![Figure 4.38  SQL Query Builder](image)

**Database Routines Editor and Debugger**

Stored procedures and user-defined functions (UDFs) are database application objects that encapsulate application logic at the database server rather than in application-level code. Use of application objects help reduce overhead of SQL statements and the results that are passed through the network. Stored procedures and UDFs are also called routines. IBM Data Studio supports routines development and debugging.
From the Data Project Explorer, create a new Data Development Project. In the project, you can create various types of database application objects such as stored procedures and UDFs (see Figure 4.39). To debug a routine, right-click the routine and select **Debug**.

**Figure 4.39** Creating a stored procedure

**IBM Data Studio Web Console**

As a database administrator, it is important to have a good understanding of your database environment. Some of the things you need to know are the kinds of activities that are happening on the system, the data server status, and the database connection status. The IBM Data Studio Web Console provides health monitoring, job management, and connection management for DB2 databases. It can be used in a single-user environment or a multiuser environment to share monitored features and analysis across database servers.

The web console is comprised of a server component and a client component. The server component must be running to monitor database health, issue alerts, and manage scheduled database maintenance jobs. The client component is a web interface used to create and manage jobs and view and analyze alerts.
Set-Up Tools

Shown in Figure 4.40 are the key web console tasks:

- View health summary
- View alerts
- View application connections
- Manage database jobs

Figure 4.40  IBM Data Studio Web Console Task Launcher

Set-Up Tools

DB2 comes with a number of tools to help you setup the environment. Using these tools, you can create the SAMPLE database, which can be used to explore DB2 features. Other tools are Configure DB2 .NET Data Provider, First Steps, Default DB2 and Database Client Interface Selection Wizard, and the Replication Center.
Configure DB2 .NET Data Provider

As its name implies, the Configure DB2 .NET Data Provider tool helps you configure the DB2 .NET Data provider, also known as the IBM Data Server Provider for .NET. There are 32-bit and 64-bit versions of the IBM Data Server Provider for .NET, each supporting the 32-bit and 64-bit versions of the .NET Framework version 2.0, 3.0, 3.5, and 4.0 CLR, respectively.

First Steps

The IBM DB2 First Steps tool is a good starting point for new DB2 users who want to get familiar with the product. It can be launched from the DB2 program menu on Windows or execute the `db2fs` command. Figure 4.41 shows the operations you can perform using this tool.

![DB2 First Steps](image)

The Create Sample Database button enables you to create a database called `SAMPLE` on your local system. This database is provided with the DB2 product for testing and learning purposes. It comes with a set of predefined tables. You can work with this database just like any other. The equivalent DB2 command to create the `SAMPLE` database is `db2sampl`.

Figure 4.41   The DB2 First Steps
After the database is created, you can create tables using the IBM Data Studio or SQL statements. The SQL statements to create database objects are described in Chapter 7, “Working with Database Objects.”

**Default DB2 and Database Client Interface Selection Wizard**

This tool only applies to Windows platforms. The Default DB2 and Database Client Interface Selection Wizard (available from the DB2 Programs menu) enables you to select or change the default DB2 install copy on your computer when there is one or more DB2 copies on it. Applications use this DB2 install copy by default. This tool can also be used to set the default IBM database client interface (ODBC/CLI driver and .NET provider) copy. You can also launch this wizard by running the `db2swtch.exe` command located in the `sqllib\bin` directory of your DB2 install copy. Figure 4.42 shows the Default DB2 and Database Client Interface Selection Wizard.

![Figure 4.42 The Default DB2 and Database Client Interface Selection Wizard](image)
The Replication Center

The Replication Center enables you to set up and manage your replication environment. You can easily follow the required steps by using the Replication Center Launchpad. Use DB2 replication when you want to propagate data from one location to another. For example, let’s say your users perform transactional queries to your database throughout the day. At the same time, another group of users performs reporting queries to the same database several times a day. When both types of queries are executed at the same time, the performance of your database degrades because the type of workload is different, causing a lot of contention. To solve this problem, you can replicate the database so that one database is used for transactional queries through the day, and the other database, the replicated one, is used for reporting queries. Figure 4.43 shows the Replication Center.

![The Replication Center](image)

**Figure 4.43** The Replication Center

Information Tools

To keep DB2 users informed on current and new product information, IBM offers valuable information via different channels such as webcasts, best practice documentations, and product
videos. Additionally, every DB2 edition ships two information tools, namely the DB2 Information Center and the Check for DB2 Update tool.

**DB2 Information Center**

The DB2 Information Center gives you access to all DB2 documentation. It comes with a fast search engine, enabling you to search on any given topic. The DB2 Information Center can be accessed in three different ways:

- Locally on the database server after installing the DB2 Information Center from a separate media DVD.
- Through a designated server on your company’s intranet. The DB2 Information Center must be installed on that server.

Figure 4.44 shows the Information Center accessed through the Internet. On the left panel there is a list of topics from which you can choose. Each of these topics can be drilled down to subtopics, and selecting a specific subtopic makes the contents panel on the right side display more information. At the top left corner of the Information Center, you find the Search field. Use this field to input any topic or keyword you want to search in the DB2 manuals. Then click the **GO** button.

![The DB2 Information Center](image-url)
Checking for DB2 Updates

The Information Center website is periodically updated with new documentation; however, if you have installed the Information Center locally, make sure to check for updates regularly. Use the Check For DB2 Updates option in the DB2 menu to launch the InstallShield Update Service, which is shown in Figure 4.45.

![Figure 4.45 Checking for DB2 updates](image)

From this site, you can download the refreshed DB2 Information Center image and install it on your server. You can also obtain information about updates to the DB2 code and news about DB2 in general.

Problem Determination Tools

Although people never want to encounter problems in their database systems, it happens, so it is important to be able to perform a logical and systematic diagnosis of a database system to identify source of problems. One of the best ways to collect diagnostic data in DB2 data is by using the `db2pd` tool.
The db2pd Tool
The db2pd tool is a command-line monitoring and troubleshooting tool that collects immediate statistics for DB2 instances and databases. This tool does not degrade the DB2 engine performance because it acquires the information directly from memory, rather than gathering data on the fly. Use this tool for troubleshooting, problem determination, database monitoring, performance tuning, and as an aid in application development design.

The db2pd tool provides many options to display information about database transactions, table spaces, table statistics, dynamic SQL, database configurations, and many other database details; for example:

- To display the operating system information, issue
  db2pd -osinfo
- To display all instance-related information, issue
  db2pd -inst
- To display all database-related information to the sample database, issue
  db2pd -db sample
- Use the db2pd -help command to display all the available options. This tool is not available through the graphical interface.

Case Study 1
You recently installed DB2 Express-C on your Windows laptop. During the installation, the DB2 instance was created. Now you want to start using DB2 by creating a database. Because you are new to DB2, you decide to use the DB2 First Steps tool (refer to Figure 4.41).

You click the Create Sample Database button from the First Steps. Next you choose the option XML and SQL objects and data, which creates a Unicode database, and click OK. Wait for a few minutes for the sample database to be created. Alternatively, you can create the sample database from a Windows command prompt or a Linux/UNIX shell using this command:

db2sampl -sql -xml

After the database is created, you launch the DB2 Command Line Processor by choosing Start > Programs > IBM DB2 > DB2COPY1 > Command Line Processor. You should see a prompt like this: db2 =>, which is the DB2 CLP prompt. From the prompt, connect to the sample database.

connect to sample

You want to examine the table spaces created by the tool; issue the following command:

list tablespaces
Five table spaces are created by default. You wonder what sample user tables were created. The following command lists all tables defined under the schema of the user ID currently connected to the database.

```
list tables
```

To list tables of a specific schema (xyz, for example), use this command:

```
list tables for schema xyz
```

Issue `quit` to exit the CLP interactive mode. Sample codes and scripts are installed with the installation that works with the `SAMPLE` database. Run the file `tbread.db2` that contains SQL statements to query the sample tables. Issue the following in the CLP.

```
db2 -tvf ..\samples\clp\tbread.db2
```

You now want to get familiar with the CLPPlus. Issue the command `clpplus` to start the command line interface. But first, connect to the `SAMPLE` database.

```
CONNECT userid@localhost:50001/sample
```

Load a sample file into the SQL buffer with the `GET` command.

```
GET ..\samples\sqlpl\resultset.db2
```

Run the script with the `RUN` command. You get an error complaining about the end label "@". Use the `EDIT` command to remove the @ sign at the end of the file. Save and close the editor.

Run the script again. This time it should work, and a stored procedure is successfully created. Call the stored procedure with the following command:

```
CALL median_result_set(?)
```

Congratulations! You have successfully created a Unicode database, queried few tables, and executed SQL scripts using the command-line tools CLP and CLPlus.

**Case Study 2**

Using the `SAMPLE` database you created in the First Steps, you now explore the IBM Data Studio graphical interface. Launch the IBM Data Studio full client. Open the Database Administration Explorer and create a database connection to the `SAMPLE` database. After a database connection is established, navigate to the `SAMPLE` database folder.

The `SAMPLE` database already has a set of tables defined in it. However, you decide to create a table of your own. To do so, you right-click the `Tables` folder and choose `Create Table`. You are presented with the following selections:

- Identify the schema for the new table. Select the user ID you logged in with. We discuss the significance of schemas in Chapter 7; for now it is sufficient to enter your user ID.
- The Properties pane is opened. Enter the name of the table you want to create, for example, `Table1`. 
• Go to the Columns tab, click the Add icon. Enter the name of the first column for Table1, for example, Col1. Choose the data type from the pull-down menu, for example, INTEGER. You could create more columns by repeating this step, but one column is sufficient for now.

There are other tabs in which you can define the properties for the new table. However, completing these two windows is enough to create the table. Click the Review and deploy changes icon to generate the CREATE TABLE statement. Click the Advanced Options button to customize the deployment such as generating REORG and RUNSTATS commands. Click Run to run the statements now. Note that you can also schedule to deploy these changes some other time, such as the next maintenance window.

Table1 is displayed under the Tables view. To display the contents of the table, right-click on the table name and choose Browse data. Since nothing has been inserted into Table1, no contents are displayed. To insert a row into the table, right-click the table name, and click Edit Data. Enter a value under Col1. Click the Play button to commit the changes.

Your colleague, who is a DB2 expert, drops by your office and offers his help for any problems you may have. You tell him you would like to get familiar with how to tune SQL statements. He asks you to right-click on the database name, and select Start Tuning.

In the Capture tab, you enter the SQL statement you want to tune and get advice on. Click the Invoke Advisors and Tools button to run the explain command against the statement.

You are now in the Review tab where you are presented with recommendations based on the query, the current database statistics, and existing indexes defined in the database. You also check out the Access Plan Graph to understand how the query result is obtained. Lastly, you review the summary report and save the report for future reference.

Though you have not finished exploring all the functionality in IBM Data Studio, this exercise has made you realize how easy to use and powerful it is!

Summary

This chapter introduced most of the tools that are available in DB2. They come in two categories: the command-driven and the graphical user interface (GUI) tools. To use the command-line tools you need to have some knowledge of DB2 commands and SQL statements. If you aren’t familiar with these, the GUI tools come in handy.

The command-line tools include the Command Line Processor (CLP), the CLP Plus, the Command Window (only on the Windows platform), and the Command Editor. The IBM Data Studio is a GUI that comes with every DB2 edition. From the IBM Data Studio, you can launch different perspectives to manage and administer your instances and databases. The SQL and XQuery editor helps you create and run SQL scripts. If you want to leverage the power of routines such as stored procedures, user-defined functions, and triggers, the database routines editor and debugger is what you need. It helps you with many tasks from developing and testing the routines to deploying and debugging the routines.
The IBM Data Studio Web Console provides health monitoring, job management, and connection management for DB2 databases. It can be used in a single-user environment or a multiuser environment to share monitored features and analysis across database servers.

The web console generates reports and keep track of your databases according to criteria you provide. These reports are handy for investigating performance problems and setting benchmarks for your database.

**Review Questions**

1. Which IBM tool can be used to schedule SQL scripts in DB2?
2. The DB2 Command Window is only available on Windows. What is the equivalent tool on the Linux/UNIX platforms?
3. Which registry variable needs to be changed to set autocommit to off permanently for the CLP?
4. When is it handy to start the CLP in interactive mode with a different terminator character as in `db2 -td!`?
5. Which command is necessary to guarantee the CLP back-end process is terminated?
6. When would you choose to use CLPPlus over CLP?
7. Which tool can be used to develop SQL user-defined functions?
8. Can the IBM Data Studio be used to perform database development for DB2 for i databases?
9. It’s 9:00 a.m., and you would like to investigate a problem that happened at 3:00 a.m. Where do you look for more information?
10. How can you obtain the most current information about a given DB2 topic?
11. Which of the following tools can be used to execute SQL statements against a DB2 database?
    A. Command Window
    B. Command Editor
    C. Command Line Processor
    D. Command Line Processor Plus
12. Which of the following is the default termination character for files processed by the DB2 CLP?
    A. :
    B. ;
    C. |  
    D. $
13. If you have the following CLP input file named samp.sql, how many commits will occur during the processing of the db2 -tvf samp.sql command?

```
connect to sample;
select * from org;
select * from dept;
connect reset;
```

A. 0  
B. 1  
C. 2  
D. 3  
E. 4  

14. In which function of the IBM Data Studio would you be able to view a data access explain plan visually?

A. Configure Automatic Maintenance editor  
B. IBM Data Studio Web Console  
C. SQL and XQuery Editor  
D. Query Tuner Workflow Assistant  

15. Which of the followings are tasks of the IBM Data Studio Web Console?

A. View alerts  
B. Manage database jobs  
C. View data access plan graphs  
D. Deploy and debug stored procedures  
E. Manage data connection profiles  

16. If you have the following CLP input file named samp.sql, how many commits will occur during the processing of the db2 +c -tvf samp.sql command?

```
connect to sample;
select * from org;
select * from dept;
```

A. 0  
B. 1  
C. 2  
D. 3  
E. 4
17. If you have the following CLP input file named `samp.sql`, which of the following commands will run this file successfully?

```sql
connect to sample@
select * from org@
select * from dept@
connect reset@
```

A. `db2 -t@f samp.sql`
B. `db2 -td@ -f samp.sql`
C. `db2 -t@ -f samp.sql`
D. `db2 -td@f samp.sql`

18. If your application receives the SQL code `-911`, which of the following commands can be used to get its description?

A. `db2 ? -911`
B. `db2 ? 911N`
C. `db2 ? SQL-911`
D. `db2 ? SQL911N`

19. Which of the following commands cannot be run from the CLP in interactive mode?

A. History
B. Edit
C. Runcmd
D. Repeat

20. Which two of the following can be performed from the CLP in interactive mode?

A. `db2 ? SQL911N`
B. `db2stop`
C. `list applications`
D. `select * from staff`
## Index

### Symbols
- `>>---` symbol, 24
- `-->` symbol, 24
- `--->` symbol, 24
- `--->` symbol, 24
- `/` (backslash line continuation character), 151
- `!` (delimiter terminator character), 151-152
- `!` (exclamation mark) operator, 155
- `?` (help command), 149-150
- `;` (semicolons), commands/statements, 70
- `10.5 offering portfolio, 14-15
  Advanced Enterprise Server Edition (AESE), 17
  Advanced Recovery Feature, 18
  Advanced Workgroup Server Edition (AWSE), 17
  Developer Edition, 18
  Enterprise Server Edition (ESE), 16-17
  Express Server, 16
  Express-C, 15-16
- `Workgroup Server Edition (WSE), 16`
- `RBAC`
  assigning privileges, 455
  granting, 454, 491
  revoking, 454, 491
- `RCAC`
  benefits, 479
  built-in functions, 468
  case scenario, 476-478
  case study, 492-493
  creating column masks, 471
  creating row permissions, 467-468
  defined, 467
  DELETE operations, 473
  enforcing column masks, 473
  enforcing row permissions, 472
  examples, 475-476
  implementing, 473-475
  INSERT operations, 473
  overview, 467-468
  session variables, 469
  UPDATE operations, 473
  access paths, 543-544

## A
- `access control concurrency. See also locking isolation levels. See isolation levels
  lost updates, 501-502
  nonrepeatable reads, 503
  overview, 500
  uncommitted reads, 502`
- `LBAC, 458-461
  column-level security, 466
  implementing, 462-465
  overview, 458
  security label components, 460-461
  security policies, 460
  table access examples, 465-466
  views, 462
  phantom reads, 504
  privileges. See privileges

---

707
access plan graphs, 170
ACCESSCTRL authority, 431
active logs, 594-595
adaptive row compression, 368
ADD clause (ALTER MODULE statement), 402
administration
authorities, 431
case study, 486-490
database-level, 431
hierarchy, 432-434
instance-level, 431
managing with IBM Data Studio, 484
operating system/security facility groups, 435
reviewing, 484
system catalog views, 484-485
system groups, assigning, 435
client (Data Studio), 161
notification logs, 686
examples, 690
Linux/UNIX, 686
Windows, 686, 696
tools (Data Studio), 164-171
Configure Automatic Maintenance editor, 165
Generate DDL, 166
Manage Connection, 164
Manage Database, 165
Monitor, 165
Query Tuner Workflow Assistant, 167-170
Set Up and Configure, 165
Start Tuning, 167
ADMIN_MOVE_TABLE stored procedure, 555
admin_priority keyword
(autoconfigure command), 227
Advanced Workgroup Server Edition (AWSE), 17
AESE (Advanced Enterprise Server Edition), 17, 42
AFTER triggers, 396-397
ALL PRIVILEGES privilege, 442
ALTER privilege, 442, 448
ALTER statements
BUFFERPOOL, 329
DATABASE PARTITION GROUP, 68, 308
MASK, 473
MODULE, 402
PERMISSION, 472
PROCEDURE, 399
SEQUENCE, 403
STOGROUP, 314
TABLE
check constraints, 358
row compression, 368
rows, locking, 519
TABLESPACE, 325
ALTERIN privilege, 438
APARs (Authorized Program Analysis Reports), 126
APIs (application programming interfaces), 516
applications
programming interfaces (APIs), 516
remote DB2 server connections, 264-265
automatic client reroute, 267-268
connection timeouts, 268-269
TCP/IP keepalive timeouts, 269
utilities, binding, 265-266
snapshot with locking information, 533-534
archival logs, 597-600
configuration parameters
ARCHRETRYDELAY, 600
FAILARCHPATH, 600
LOGARCHMETH1, 597-599
LOGARCHMETH2, 599
LOGARCHOPT1/LOGARCHOPT2, 600
NUMARCHRETRY, 600
overview, 597
ARCHRETRYDELAY configuration parameter, 600
arrays (security labels), 460
ASC (non-delimited ASCII) file format, 556
attach command, 208
attaching to instances, 208-209
authentication, 417
Data Server clients, 427-429
all clients trusted, 428
server, 429
TRUST_ALLCLNTS parameter, 427
TRUST_CLNTAUTH parameter, 428
trusted clients only, 429
at DB2 servers, 422-423
GSS plug-ins, 424-425
Database Manager parameters, 426
example, 425
samples, 425
unsupported, 425
Kerberos, 423-424
types, configuring at client, 418-421
server, 417-418
AUTHENTICATION parameter, 418
authorities. See administration, authorities
Authorized Program Analysis Reports (APARs), 126
auto-commit, disabling, 148
autoconfigure command
invoking
ADMIN_CMD stored
procedure, 228
CREATE DATABASE
command, 228
keywords, 227
listing, 227
automatic client reroute, 267-268
automatic maintenance, 585
automatic storage, 296-297
AUTOMATIC STORAGE
clause (CREATE DATABASE
command), 297
automatic storage table spaces, 310
creating, 322
nonautomatic, compared, 323
overview, 322
AWSE (Advanced Workgroup
Server Edition), 17, 42

B
back-end processes (CLP), 152-153
backslash line continuation
character (\), 151
BACKUP DATABASE
command, 607
backup in progress table space
state, 570
backup pending table space state, 570
backups
case study, 623
configuration files, 57
databases
performing, 602-604
restoring, 608-609
rolling forward, 614-616
dropped tables, recovering, 616-618
files, 607-608
names, 607
Windows, 607
incremental, 605-606
online versus offline access, 603
recovery
history file, 618-620
RECOVER DATABASE
command, 620-621
redirected restores, 610-614
table spaces
performing, 605
restoring, 609-610
rolling forward, 616
BEFORE triggers, 396, 397
BETWEEN predicate (SELECT
statement), 655
big data, 3-4
characteristics, 4-5
DB2 integration with
BigInsights, 11
data from DB2, obtaining, 11-12
filtering/aggregating/
transforming data
warehouse data, 12-13
defined, 3
history, 4
IBM platform, 6-8
costs, reducing, 9
raw data analysis, 8-9
streaming data analysis, 10
unlocking big data, 8
warehouse simplification, 10-11
types
in motion, 5
in place, 6
at rest, 6
Big Data University, 705
Big SQL, 645
BigInsights
DB2 integration, 11
data from DB2, obtaining, 11-12
filtering/aggregating/
transforming data
warehouse data, 12-13
features, 8-9
BIGINT (big integer) data type, 334
binary large objects (BLOBs), 336
bind command, 514
BIND privilege, 445
binding utilities, 265-266
bitemporal tables, 377
BLOBs (binary large objects), 336
blocks, 384
indexes, 385-387
maps, 387
blog websites, 699,
BLU Acceleration, xxix, 3,
339, 378
BOOLEAN data types, 334
bp_resizeable keyword
(autoconfigure command), 227
buffer pools
creating, 326-329, 404
default, 299
defined, 53, 287
dropping, 330
definition, 298
editing, 329-330
overview, 326
partitioned databases, 68
build phase (LOAD utility), 563
built-in data types, 332
boolean, 334
date and time, 336
numeric, 333-334
string, 334-336
XML, 337
built-in functions, 400
encryption/decryption, 457
RCAC, 468

call level interface (CLI),
514-515
CASE expression (SELECT statement), 661-662
case studies
backups, 623
CLP
launching, 181
table spaces, examining, 181-182
CLPPlus, 182
concurrency/locking, 538-539
configuration files, saving, 229
connectivity
Data Server clients host connection via the gateway, configuring, 279
DB2 Connect gateway, configuring, 278
host connection, testing, 278-279
TCP/IP listeners, enabling, 279
data movement utilities, 622-623
Data Studio, 182-183
Database Manager
configuration file
parameter, editing, 84
viewing, 84
database objects
buffer pools, 404
check constraints, 406-407
database connections, 404
databases, creating, 404
indexes, 405
referential constraints, 406
schemes, 407
tables, creating, 405
databases
configuration file, viewing, 84
configuring, 231
creating, 84, 181
listing, 81-84
tables/table spaces, checking, 84
IBM PureData System, 26-27
installation, ESE
new copy, 128-130
scratch with Setup Wizard, 130-131
instances
configuring, 231
current, viewing, 81
listing, 81, 230
registry variables
listing, 230
setting, 230
unsetting, 230
security
administrative authorities/privileges, 486-490
RCAC, 492-493
roles, granting/revoking, 491
tables, creating, 407-408
case-sensitivity
CLP commands, 80
configuration parameters, 82
database object names, 83
registry variables, 81
SQL statements, 81
stored data, 82
system commands, 80
XQuery statements, 81
CAST expression (SELECT statement), 652-653
catalog command, 49, 256
catalog database command, 420
catalog db command, 237
catalog DCS database command, 237
catalog partitions, 67
catalog table space (SYSCATSPACE), 52, 67, 311
catalog TCPIP node command, 237
CDI (Continuous Data Ingest). See ingest utility
cells (MDC tables), 385
CF (cluster caching facility), 75
CGTTs (Create Global Temporary Tables), 371
creating, 372
DGTTs, compared, 371
CHAR columns, 335
check constraints, 358-360,
406-407
adding, 358-359
turning on/off, 359
Check for DB2 Updates menu option, 43
circular logging, 596
classic row compression, 368
classifications
partition groups, 305-306
stored procedures, 399
table spaces, 310-311
tables, 338-341
triggers, 396
UDFs, 400-401
UDTs, 337
views, 391-392
deletable, 392
insertable, 393-394
read-only, 394
updatable, 393
CLI (call level interface), 514-515
CLIENT authentication type, 419, 421
clients, 19
authentication, 427-429
SQL operations
implications, 355-358
unique keys, 354
row-organized table
conversions, 379
security labels, 460
UNION/UNION ALL
operations, 668-669
Command Line Processor Plus.
See CLPPlus
Command Line Processor. See CLP
Command Window, 139-140
commands
invoking, 140
support, 140
isolation levels, changing,
512-513
starting, 140
command-line tools, 39, 138-139
CLP, 141
command parameters,
147-149
executable, 141
front-end/back-end
processes, 152-153
help, 149-150
interactive mode, 141-142
launching, 181
line continuation, 150-152
noninteractive mode,
143-149
table spaces examining,
181-182
CLPPlus
case study, 182
databases, connecting/
disconnecting, 155-156
features, 154
OS commands, running,
155
output, formatting, 159
SQL buffer, 156-158
starting, 154
window, 154
Command Window, 139-140
invoking commands, 140
starting, 140
supported commands, 140
commands
; (semicolons), 70
attach, 208
autoconfigure
invoking with ADMIN_
CMD stored procedure,
228
invoking with CREATE
DATABASE command,
228
keywords, 227
listing, 227
backup database, 607
bind, 514
case-sensitivity
CLP commands, 80
system commands, 80
catalog, 49, 256
catalog database, 420
catalog db, 237
catalog DCS database, 237
catalog TCP/IP node, 237
CLP, 36-37
help, 149-150
interactive mode, 142
parameters, 147-149
CLPPlus, 37
Command Window
invoking, 140
support, 140
create database, 49, 51, 237
autoconfigure command,
invoking, 228
AUTOMATIC
STORAGE clause, 297
partitions, 65-66
syntax, 297
create database partition
group, 67
DAS, listing of, 217
dasrt, 217
dasdrop, 217
database level, 226
db2admin
create, 217
drop, 217
start, 217
stop, 217
db2convert, 379
db2dart, 694
db2fodc, 697-698
db2iauto, 206
db2icrt, 200-201, 216
db2idrop, 202, 216
db2ilist, 203, 216
db2icm, 116
db2look, 579-580
db2ls, 125-126
db2move, 577-579
db2pdcfg, 697
db2prereqcheck, 94
db2set, 195
parameters, 197
registry variables, 196
db2setup, 112
db2start, 205, 216
db2stop, 207, 216
db2swtch.exe, 122
db2trc, 693
db2val, 692
directories, 237
drop database, 237, 300
executing in partitioned
databases, 69-70
export, 189, 557
flush package cache, 584
force application, 530-532
get admn cfg, 217
get db cfg, 226
get db cfg for database_name,
222
get dbm cfg, 213, 216
help (?), 684-685
import, 560
INGEST, 574-575
instance level, 216
list applications, 529-531
list database directory, 237
list database directory on path/drive, 237
LIST DATABASE PARTITION GROUPS, 68
list db directory, 237, 239, 300
list db directory on drive/path, 242
list db directory on path/drive, 237
list dcs directory, 237, 244
LIST HISTORY BACKUP, 618
list node directory, 237, 243
list tablespaces show detail, 572
list utilities, 571-572
load, 563-564
load client, 565-567
load query, 570-571
precompile, 514
QUIESCE TABLESPACES FOR TABLE, 518
RECOVER DATABASE, 620-621
reorgchk, 582-583
reset admin cfg, 217
reset db cfg, 226
reset db cfg for database_name, 223
reset dbm cfg, 214, 216
RESTORE, 611
RESTORE DATABASE, 608
set, 189
set current lock timeout, 525
set integrity, 569
SET SCHEMA, 332
setup, 112
system, 36
Text Search, 37
uncatalog db, 237
uncatalog DCS database, 237
uncatalog node, 237
update admin cfg, 217
update db cfg, 226
update db cfg for database_name, 221
update dbm cfg, 213, 216
common table expressions (CTE), 671
communication protocols client to remote DB2 server connections, 252
DB2 instance, configuring, 99
compressing tables, 369
concurrency access paths, choosing, 543-544
case study, 538-539
isolation levels. See isolation levels
locking. See locking
lost updates, 501-502
nonrepeatable reads, 503
Oracle compatibility, 681
overview, 500
phantom reads, 504
uncommitted reads, 502
conference website, 705
Configuration Advisor configuration parameter values, passing, 228
running automatically/manually, 228
starting, 227
configuration files backing up, 57
DAS
resetting, 217
updating, 217
viewing, 217
Database Manager parameters current values, viewing, 213-214, 222-223
editing, 84
GSS plug-in, 426
listing, 209-212
resetting, 214, 223
updating, 212-213, 221-222
verifying, 272
viewing, 217-221
databases current parameter values, viewing, 222-223
resetting parameters, 223
updating parameters, 221-222
viewing, 84, 217-221
node. See node configuration file
saving, 229
collection levels (environment), 187-188
configuration manager, 42
configuration parameters, 48-49
case-sensitivity, 82
database level, 48-49
changes taking effect, 49
storing, 48
updating, 48
viewing, 48
DIAGLEVEL, 686, 689
DIAGPATH, 688
DIAGSIZE, 689
graphical tool, 48
instance level, 48
changes taking effect, 48
storing, 48
updating, 48
viewing, 48
logging
ARCHRETRYDELAY, 600
FAILARCHPATH, 600
LOGARCHMETH1, 597-599
LOGARCHMETH2, 598
LOGARCHOPT1/
LOGARCHOPT2, 600
MIRRORLOGPATH, 601
NUMARCHRETRY, 600
MAXLOCKS, 520
NOTIFYLEVEL, 686, 687
Configure DB2.NET Data
Provider tool, 176
Configure Parameters Editor, 48
configuring
authentication types at
client, 418-421
server, 417-418
DAS, 99
databases
case study, 231
configuration file
content, viewing, 217-221
current parameter values,
viewing, 222-223
Data Studio, 223-225
resetting parameters, 223
servers, 252-253
updating parameters,
221-222
instances in Data Studio, 215
instances with DBM
configuration file, 209-214
current parameter values,
viewing, 213-214
listing, 209-212
parameter default values,
resetting, 214
updating parameters,
212-213
Connect, 20
Connect gateway
configuring, 278
remote client connection to
host servers, 262-263
CONNECT statement, 518
connectivity
case study
Data Server clients host
connection via the
gateway, configuring, 279
DB2 Connect gateway,
configuring, 278
host connections, testing,
278-279
TCP/IP listeners,
configuring, 279
databases, 155-156, 404
directories
book analogy, 238-239
commands, 237
DCS, 51, 244
local connections, 245
local database, 50,
241-242
node, 50-51, 242-244
remote databases, 246-247
remote host DB2 server,
247
system database, 49,
239-241
local connection from data
server client to DB2 server,
249-250
mainframe host, 20
overview, 235-236
partitioned databases
clients, 62
servers, 62
problems, 269
client configuration,
verifying, 272-274
client-host diagnostic
flowchart, 276
client-server diagnostic
flowchart, 275
database connection
flowchart, 269
diagnosis aids, 277-278
server configuration,
verifying, 270-272
profiles (Data Studio),
162-165
creating, 162-164
updating, 164
pureScale interconnect, 74
remote. See remote
connections
timeouts, 268-269
constraints, 350
check
adding, 358-359
case study, 406-407
turning on/off, 359
violations, 359
data validation, 568
informational, 360-361
referential. See referential
integrity
turning off, 359
unique, 351-352
containers
defined, 52
DMS table spaces
device, 321
file, 321-322
SMS table spaces, creating,
319-320
table spaces, 312
Continuous Data Ingest. See
ingest utility
Control Center tools, 138
CONTROL privilege
indexes, 444
packages, 445
tables/views, 442
coordinator partitions, 69
copies (DB2)
benefits, 121
multiple, installing, 121
DAS, 126
Linux/UNIX, 124
products/features
installed, viewing, 125-126
Windows, 121-123
names, 99
core files, 687
COUNT aggregate function
(SELECT statement), 648
crash recovery, 588
create database command, 49, 51, 237
autoconfigure command, invoking, 228
AUTOMATIC STORAGE clause, 297
partitions, 65-66
syntax, 297
create database partition group command, 67
Create Global Temporary Tables. See CGTTs
CREATE statements
BUFFERPOOL, 68, 326
database objects, 288
DATABASE PARTITION GROUP, 307
FUNCTION, 400
INDEX, 379
MASK, 471
MODULE, 402
PERMISSION, 470
PROCEDURE, 398
SCHEMA, 331
SECURITY LABEL COMPONENT, 463
SEQUENCE, 401
STOGROUP, 313
TABLE, 341
check constraints, 358
DATA INITIALY DEFERRED clause, 370
informational constraints, 360
NOT LOGGED INITIALLY clause, 362
ORGANIZE BY COLUMN clause, 378
partitions, 363
REFERENCES clause, 354
REFRESH DEFERRED clause, 370
row compression, 368
VALUE COMPRESSION clause, 369
TABLESPACE, 69, 317
automatic storage table spaces, 322
DMS table spaces, creating, 320
SMS table spaces, 319
TRIGGER, 396
TYPE, 337
UNIQUE, 352
USER MAPPING, 80
USER TEMPORARY TABLESPACE, 371
VIEW, 390, 394
CREATEIN privilege, 438
cross-platform development, 27
CS (Cluster Services), 75
CS (cursor stability), 505-509
currently committed disabled, 506-508
enabling, 508-509
testing, 546
overview, 505-506
without CC, testing, 542-543
CTE (common table expressions), 671
cumulative incremental backups, 605-606
CURRENT DATE (CURRENT_ DATE) register, 650
CURRENT ISOLATION register, 650
CURRENT LOCK TIMEOUT register, 650
CURRENT PACKAGE PATH register, 650
CURRENT PATH (CURRENT_PATH) register, 650
CURRENT SCHEMA (CURRENT_SCHEMA) register, 650
CURRENT TIME (CURRENT_TIME) register, 650
CURRENT TIMESTAMP (CURRENT_TIMESTAMP) register, 650
CURRENT USER (CURRENT_USER) register, 650
CURSOR file format, 557, 564
cursor stability (CS), 505-509
customizing installation images, 118-120
D
DAS (Database Administration Server), 45, 216
admin configuration file
resetting, 217
updating, 217
viewing, 217
commands, listing of, 217
configuring, 99
creating
Linux/UNIX, 217
Windows, 217
dropping
Linux/UNIX, 217
Windows, 217
multiple DB2 copies, 126
overview, 216
starting, 45, 217
stopping, 45, 217
users, creating, 216
dasct command, 217
dasdrop command, 217
data
case-sensitivity, 81
distribution across partitions, 71-72
encryption. See encryption loading into tables, 464, 474
movement
file formats, 555, 557
utilities. See data
movement utilities multitemperature, 314-315
types. See data types
Data Control Language (DCL) statements, 34
Data Definition Language (DDL), 33, 579-580
DATA INITIALLY
DEFERRED clause (CREATE TABLE statement), 370
Data Manipulation Language (DML) statements, 33
data movement utilities case study, 622-623
db2move, 577-579
EXPORT, 557-559
column names, specifying, 558-559
multiple tables, 558
file formats
ASC (non-delimited ASCII), 556
CURSOR, 557
WSF, 557
IMPORT, 559-562
columns, selecting to import, 561
failures, restarting, 562
import command, 560
import modes, 560-561
regular commits, 562
ingest, 573-576
benefits, 573
INGEST command, 574
named pipes, 575
phases, 573
LOAD, 562
clients, loading from, 567-568
CURSOR file format, 564
data validation against constraints, 568
exception tables, 565-567
load command, 563-564
phases, 562-563
progress monitoring, 568-570
rejected records dump files, 564-565
table space states, 569-570
table states, 569
loading data utility comparison, 576
overview, 554-555
Data Server clients. See clients
Data Studio, 39-41
Administer tab, 41
administration client, 161
authorities, managing, 484
case study, 182-183
Configure Parameters Editor, 48
connection profiles, 162-165
creating, 162-164
updating, 164
database administration tools, 164-171
access plan graphs, 170
Configure Automatic Maintenance editor, 165
Generate DDL, 166
Manage Connection, 164
Manage Database, 165
Monitor, 165
Query Tuner Workflow Assistant, 167-170
Set Up and Configure, 165
Start Tuning, 167
database development tools, 171-174
Database Routines Editor and Debugger, 173-174
SQL and XQuery editor, 171-172
SQL Query Builder, 172
databases, configuring, 223-225
deprecated parameters, 224
quiesced mode, 225
restarting databases, 225
starting/stopping databases, 224
Design tab, 41
Develop tab, 41
downloading, 40
features by data server website, 161
full client, 160
instances, 214-215
configuring, 215
starting/stopping, 214
viewing, 214
Monitor tab, 41
overview, 160
package isolation levels, viewing, 514
privileges, managing, 486
Task Launcher, 40, 161-162
Tune tab, 41
web console, 161, 174-175
overview, 174
tasks, 175
workspace, 161
data types
built-in, 332
boolean, 334
date and time, 336
numeric, 333-334
string, 334-336
XML, 337
choosing, 339
Oracle compatibility, 676-681
user-defined, 337-338
classifications, 337
creating, 337
dropping, 338
DATABASE ACCESS authority, 431
Database Administration Server.  
See DAS
Database Connection Services directory.  See DCS
Database Conversion Workbench.  See DCW
database-managed space table spaces.  See DMS table spaces
Database Manager configuration file parameters
  current values, viewing, 213-214, 222-223
  editing, 84
  GSS plug-in, 426
  listing, 209-212
  resetting, 214, 223
  updating, 212-213, 221-222
  verifying, 272
  viewing, 217-221
Database Routines Editor and Debugger, 173-174
databases, 51
  administration and performance classroom courses, 703
  administrative authorities, 431
  automatic storage, 296-297
  backups
    files, 607-608
    incremental, 605-606
    online versus offline access, 603
  performing, 603-604
  commands, 226
  configuration file, viewing, 84
  configuration parameters, 48-49
  changes taking effect, 49
  storing, 48
  updating, 48
  viewing, 48
configuring
  case study, 231
  configuration file contents, viewing, 217-221
  current parameter values, viewing, 222-223
  Data Studio, 223-225
  resetting parameters, 223
  updating parameters, 221-222
connections, 404
  CLPPlus, 155-156
  local, 244
  creating, 51, 84, 297-298, 404
  First Steps, 181
  multipartition environment, 302-304
  single-partition environment, 301-302
  temporary data/user data separate table space, 304
defined, 51, 290
design considerations, 228-229
dropping, 300
federation, 21
  allowing, 77
  implementing, 79
  Informix access, 80
  nicknames, 79
  wrappers, 80
host connections
  clients via gateway, 279
  testing, 278-279
listing, 81-84, 300
lock snapshots, 534-536
locking, 518
maintenance.  See maintenance
  names, verifying, 274
objects
  case-sensitivity, 83
  creating/dropping, 288
default, 299
interaction, 288-290
listing of, 286-287
names, 287
overview, 285
objects.  See objects
partitioning, 58-62
buffer pools, 68
catalog, 67
commands, executing, 69-70
  coordinator, 69
  creating, 65-66
data distribution, 71-72
  Data Server client connection, 62
  DB2NODE environment variable, 70-71
defined, 58
  groups.  See groups, partition
  illustration, 60
  instances, 64-65
  multiple, 60
  node configuration file, 62-64, 292-297
  overview, 290-292
  server interpartition communication, 62
  single, 58
  SQL statements, executing, 69-70
  table partitioning/MDC, combining, 388-389
  table spaces, 69
queries, 51
quiesced mode, 225
recovery, 608-609
  archival logging, 597-600
  backups, 608-609
  circular logging, 596
  crash, 588
dropped tables, 616-618
  handling transaction logs, 601-602
history file, 618-620
infinite logging, 598
log mirroring, 598-601
logging methods versus recovery methods, 602
overview, 585
RECOVER DATABASE command, 620-621
recoverable versus nonrecoverable databases, 602
redirected restores, 610-614
roll forward, 589, 614
strategies, 586-587
transaction logs. See transaction logs
transactions, 587
types, 588
version, 588
restarting support, 22-23
restoring backups, 608-609
redirected restores, 610-614
rolling forward, 614-616
SAMPLE, 304
servers
client connections, accepting, 252-253, 259
existence, verifying, 271
snapshot with locking information, 533
starting/stopping, 224
tables/table spaces, checking, 84
DATA_ENCRYPT
authentication type, 419, 421
DATA_ENCRYPT_CMP
authentication type, 419
date and time data types, 336
DATE data type, 336
db cfg file, 84
DB2
Advanced Recovery Feature, 18
documentation website, editions, 13-18
10.5 offering portfolio, 14-15
Advanced Enterprise Server Edition (AESE), 17
Advanced Workgroup Server Edition (AWSE), 17
database federation support, 21
Developer, 18
Enterprise Server Edition (ESE), 16-17
Express Server, 16
Express-C, 15-16
LUW (Linux, UNIX, Windows), 13
recovery, 588
replications support, 22-23
upgrading, 14, 128
website, 18
Workgroup Server Edition (WSE), 16
z/OS, VM/VSE, i, 13
environment. See environment
Express-C website,
history, 1-3
pureScale, 3
pureXML capabilities, 2
SQL compatibility, 3
UDB (Universal Database), 2
version 10, 3
version 10.5, 3
Information Center. See Information Center
instances, 44, 198
DB2 Essentials: Understanding
DB2 in a Big Data World,
43, 701
DB2 for Linux, UNIX, and Windows websites
DeveloperWorks forum, 699
main web page,
Technical support website, 699
DB2 instance, configuring, 99
db2admin create command, 217
db2admin drop command, 217
db2admin start command, 217
db2admin stop command, 217
db2ckbp tool, 699
DB2COMM registry variable, 271
db2convert command, 379
db2cos script, 695-696
db2dart tool, 694-695
db2diag.log, 686
examples, 690-691
troubleshooting tool, 692
DB2_EVALUNCOMMITTED registry variable, 527
db2fode utility, 697-698
DB2_GRP_LOOKUP registry variable, 482-483
db2i auto command, 206
db2icrt command, 200-201, 216
db2idrop command, 202, 216
db2ilist command, 203, 216
db2_install script, 93, 113-114
installation, performing, 113
keywords, 114
parameters, 114
DB2INSTANCE environment variable, 46, 204
current setting, checking, 46
value, setting, 46
db2iprune utility
fix packs, 120
input file example, 118-120
Linux/UNIX sample, 118
overview, 118
pureScale support, 120
Windows sample, 118
db2licm command, 116
db2look command, 555, 579-580
db2ls command, 125-126
db2move utility, 577-579
db2mtkr, 699
DB2NODE environment variable, 70-71
db2nodes.cfg file, 62-64
database partitions, 292-297	edition, 296
eight two-way SMP Linux servers with eight partitions in total, 294-295
eight-way SMP Linux server with four partitions, 294
four four-way SMP UNIX servers with eight partitions, 295-296
Linux/UNIX columns, 292
Windows columns, 293
edition, 64
example, 62-63
location, 294
storing, 63
db2pd tool, 181
db2pdcfg command, 697
db2prereqcheck command, 94
db2profile script file for Linux/UNIX, 191-194
db2relocatedb tool, 555
db2spgn utility, 111-112
db2set command, 195
parameters, 197
registry variables, 196
db2setup command, 112
DB2_SKIPDELETED registry variable, 528
DB2_SKIPINSERTED registry variable, 528
db2start command, 205, 216
db2stop command, 207, 216
db2support utility, 692-693
db2swtch.exe command, 122
db2top tool, 699
db2trc command, 693
db2val command, 692
DBADM authority, 431
DBCLOBs (double-byte character large objects), 336
dbm.cfg file. See Database Manager configuration file parameters
DCL (Data Control Language) statements, 34
DCS (Database Connection Services) directory, 51, 244
commands, 237
contents, viewing, 51, 244
entries, adding, 51
remote client connection to DB2 host database, 259
DCW (Database Conversion Workbench)
overview, 681
task launcher, 682
tasks, 682
website, 682
DDL (Data Definition Language), 33, 579-580
deadlocks, 526-527, 542-545
DECFLOAT data type, 334
DECIMAL data type, 334
Declare Global Temporary Tables. See DGTTs
DECRIPT_BIN function, 457
DECRIPT_CHAR function, 457
decryption, 456-458, 457. See also encryption
DEFAULT clause (CREATE TABLE statement), 344-345
Default DB2 and Database Client Interface Selection Wizard, 177
default option fields syntax, 24
DEL (delimited ASCII) file format, 555
deletable views, 392
delete pending table space state, 570
delete phase (LOAD utility), 563
DELETE privilege, 442
DELETE statements
FROM, 356-357, 664
SELECT from, 664
deleting rows, 664
delimited ASCII (DEL) file format, 555
delimiter terminator character (!), 151-152
delta backups, 605-606
derived columns, 646-648
Developer Edition, 18
DeveloperWorks website, development tools (Data Studio), 171-174
Database Routines Editor and Debugger, 173-174
SQL and XQuery editor, 171-172
SQL Query Builder, 172
device containers, 321
DGTTs (Declare Global Temporary Tables), 371
CGTTs, compared, 371
creating, 371
DIAGLEVEL configuration parameter, 686, 689
diagnosing problems, 683
help (?) command, 684-685
information, collecting
administration notification logs, 686, 690
core files, 687
db2diag.log, 686, 690-691
DIAGLEVEL parameter, 689
DIAGPATH parameter, 688
DIAGSIZE parameter, 689
dump files, 687
FODC, 686
NOTIFYLEVEL parameter, 687
trap files, 686-687
overview, 683-684
searching for known problems, 699
tools
db2ckbkp, 699
db2cos script, 695-696
db2dart, 694-695
db2diag.log, 692
db2fodc utility, 697-698
db2mtrk, 699
db2pdcf, 697
db2support, 692-693
db2top, 699
db2val command, 692
INSPECT, 695
summary, 698
trace utility, 693-694
DIAGPATH configuration parameter, 688
DISTINCT clause (SELECT statement), 648-649
distribution data across partitions, 71-72
disks, 71
DML (Data Manipulation Language) statements, 33
DMS (database-managed space) table spaces
director...
database objects, 288
databases, 300
indexes, 382
instances, 45, 56, 202-203
partition groups, 68, 310
partitions from partition
groups, 68
row permissions, 472
schemas, 332
sequences, 403
storage groups, 314
stored procedures, 399
table spaces, 325-326
UDTs, 338
user tables, 344
views, 391
dump files, 564-565, 687
e
registry variable level indicator, 196
EDIT script, 157-158
editing
buffer pools, 329-330
db2nodes.cfg database partitions, 296
environment variables, 46
isolation levels
APIs, 516
call level interface, 514-515
DB2 command window, 512-513
precompile/bind commands, 514
statement level, 516-517
node configuration file, 64
partition groups, 308
profile registries, 195
sequences, 403
storage groups, 314
stored procedures, 399
table spaces, 325
editions, 13-18
10.5 offering portfolio, 14-15
Advanced Enterprise Server Edition (AESE), 17
Advanced Workgroup Server Edition (AWSE), 17
database federation support, 21
Developer, 18
Enterprise Server Edition (ESE), 16-17
Express Server, 16
Express-C, 15-16
LUW (Linux, UNIX, Windows), 13
recovery, 588
replication support, 22-23
trial copies, 94
upgrading, 14, 128
website, 18
Workgroup Server Edition (WSE), 16
z/OS, VM/VSE, i, 13
editors
Database Routines Editor and Debugger, 173-174
SQL and XQuery, 171-172
ENABLE QUERY OPTIMIZATION clause
(CREATE TABLE statement), 360
ENCRIPT function, 457
encryption, 456-458
built-in functions, 457
case study, 491
DATA_ENCRYPT authentication value, 457-458
SSL, 457
ENFORCED clause (CREATE TABLE statement), 360
Enterprise Server Edition (ESE), 16-17, 128-130
environment
buffer pools, 53
configuration levels, 187-188
configuration parameters, 48-49
database level, 48-49
graphical tool, 48
instance level, 48
DAS, 45
database transaction logs, 53
databases, 51
directories
DCS, 51
local database, 50
node, 50-51
structure (Windows), 53-57
system database, 49
indexes, 53
instances, 44-45
creating, 45
dropping, 45
Linux/UNIX, 45
starting, 45
stopping, 45
LOBs, 53
overview, 43
profile registry
defined, 194
editing, 195
registries, 194-195
registry variables, 46-47
level indicators, 196
search order, 196-197
setting at Instance Node-Level, 196
starting/stopping instances for changes to take effect, 197
storage groups, 53
table spaces, 52-53
tables, 53
variables, 46
contents, viewing, 189
creating/editing, 46
DB2INSTANCE. See
DB2INSTANCE
environment variable
DB2NODE, 70-71
permanently setting, 190
setting, 189-194
temporarily setting, 189
error messages, 684
ESE (Enterprise Server Edition), 16-17, 128-130
Event Monitor, 536
EXCEPT operator, 668
EXCEPT/EXCEPT ALL
operators, 670-671
exception tables, 565-567
exclamation mark (!) operator,
155
EXECUTE privileges
packages, 445
routines, 446
EXPLAIN authority, 431
EXPORT utility, 557-559
column names, specifying,
558-559
environment variables, 189
example, 557-558
keywords, 558
multiple tables, 558
Express-C edition, 15-16
Express Server Edition, 16
Extensible Markup Language
(XML) data type, 337
extents, 315-317
external functions, 401

F
FAILARCHPATH configuration
parameter, 600
federation, 78
allowing, 78
implementing, 79
Informix access, 80
nicknames, 79
support, 21
wrappers, 80
fenced stored procedures, 399
FETCH FIRST n ROWS ONLY
clause (SELECT statement),
654
files
backup, 607-608
names, 607
Windows, 607
configuration
backing up, 57
saving, 229
core, 687
DAS configuration
resetting, 217
updating, 217
viewing, 217
data movement formats, 555
ASC (non-delimited
ASCII), 556
CURSOR, 557
DEL (delimited ASCII),
555
PC/IXF (PC version of
IXF), 557
WSF, 557
database configuration
parameters
current values, viewing,
213-214, 222-223
editing, 84
listing, 209-212
resetting, 214, 223
updating, 212-213,
221-222
verifying, 272
viewing, 84, 217-221
db2diag.log, 686
examples, 690-691
troubleshooting tool, 692
DMS table space containers,
321-322
dump, 564-565, 687
input, 144-149
logs. See logs
node configuration, 62-64
database partitions,
292-297
database partitions,
292-297
database partitions,
292-297
database partitions,
292-297
database partitions,
292-297
database partitions,
292-297
database partitions,
292-297
database partitions,
292-297
FODC, 686
First Steps tool, 176-177, 181
fix packs
applying, 127
db2iprune utility, 120
downloading, 127
installing, 126-128
multiple, installing, 121
DAS, 126
Linux/UNIX, 124
products/features
installed, viewing,
125-126
Windows, 121-123
non-root installations, 128
FLOAT data type, 334
flush package cache command,
584
FLWOR (FOR, LET, WHERE, ORDER, RETURN) expression, 35
FODC (First Occurrence Data Capture), 686
force application command, 530-532
foreign keys, 353-354
formatting CLPPlus output, 159
FROM clause (SELECT statement), 653
front-end processes (CLP), 152-153
full client Data Studio, 160
full outer joins, 659
functions
   built-in, 400
      encryption/decryption, 457
      RCAC, 468
COALESCE, 660
column, 652
COUNT (SELECT statement), 648
OVER, 662-663
ROWNUMBER, 662-663
snapshot table, 536
table, 324
UDFs
   classifications, 400-401
   creating, 400
   defined, 287

G
[g] registry variable level indicator, 196
Generic Security Service. See GSS plug-ins
get admin cfg command, 217
get dbm cfg command, 213, 216
get dbm cfg command, 213, 216
get scripts, 157
GETHINT function, 457
global group security, 481
Global-Level Profile Registry, 195
global level (registry variables), 47
Global Switcher, 122
GPFS, 75-76
GRANT EXEMPTION ON RULE statement, 465
GRANT statement (privileges), 438-439
   indexes, 444
   packages, 445
   routines, 446
   security labels, 449
   sequences, 448
SETSESSIONUSER, 451
table spaces, 440
tables/views, 441
GRAPHIC data types, 335
GROUP BY clause (SELECT statement), 657
GROUP PLUGIN parameter, 426
   groups
      partition, 67-68
      classifications, 305-306
      creating, 67, 307-308
      default, 67, 299, 306
      defined, 286
      dropping, 68, 310
      editing, 308
      listing, 309
      overview, 305
      partitions, adding/dropping, 68
      viewing, 68
storage
      creating, 313
      default, 299
      defined, 53, 286
dropping, 314
   editing, 314
   multitemperature data, 314-315
table spaces, 312-315
   user, required
      Linux/UNIX, 105-107
      Windows, 105
GSS (Generic Security Service) plug-ins, 424
Database Manager parameters, 426
example, 425
sample plug-ins
   client-side, 425
   group, 425
   server-side, 425
unsupported, 425
GSSPLUGIN authentication type, 419, 421
GSS_SERVER_ENCRYPT authentication type, 419

H
Hadoop
costs, reducing, 9
DB2 integration, 11
data from DB2, obtaining, 11-12
filtering/aggregating/transforming data
warehouse data, 12-13
history, 4
HAVING clause (SELECT statement), 657
help command (?), 149-150, 684-685
history
   big data, 4
   DB2, 1-3
      pureScale, 3
      pureXML capabilities, 2
      SQL compatibility, 3
UDB (Universal Database), 2
version 10, 3
version 10.5, 3
hosts
costs, reducing, 9
raw data analysis, 8-9
streaming data analysis, 10
unlocking big data, 8
warehouse simplification, 10-11
IBM
Academic Initiative program website, 26
Big Data Platform, 6-8
certification program and tutorials website, 703
Cognos Business Intelligence Reporting, 17
Data Magazine website, 705
data server clients/driver packages, 19
Data Studio, 39-41
Administer tab, 41
administration client, 161
authorities, managing, 484
case study, 182-183
Configure Parameters Editor, 48
costs, reducing, 9
IBM DEFAULTGROUP partition group, 67, 306
IBM FTPGROUP partition group, 67, 306
IBM TEMPGROUP partition group, 67, 306
IBMCATGROUP partition group, 67, 306
IBMDEFAULTGROUP partition group, 67, 306
IBM FTPGROUP partition group, 67, 306
IBM TEMPGROUP partition group, 67, 306
IDA (InfoSphere Data Architect), 42
identity columns, 347-350
IDUG (International DB2 Users Group), 705
images (installation), 94
customizing, 118-120
pruned, installing, 120
implicit privileges, 452-453
import command, 560
IMPORT utility, 559-562
import command, 560
import modes, 560-561
regular commits, 562
IN predicate (SELECT statement), 655-656
incremental backups, 605-606
Index Advisor, 382
index copy phase (LOAD utility), 562
INDEX privilege, 442
indexes, 379
block, 385-387
clustering, 382-383
creating, 379-380, 405
defined, 53, 287
dropping, 382
high/low cluster ratios, 582
Index Advisor, 382
physical characteristics
statistics, updating. See
RUNSTATS utility
privileges, 444-445
query performance, 382
reorganizing, 582-584
sales records example,
380-382
temporary tables, 372
infinite logging, 598
Information Center, 23, 179
accessing, 179
overview, 43
updates, checking, 180
website, 23, 179, 699,
information tools, 179, 702
Check for DB2 Updates menu
option, 43, 180
Information Center. See
Information Center
informational constraints,
360-361
Informix access, 80
InfoSphere
Data Architect (IDA), 42
Optim portfolio tools, 42
ingest utility, 573-576
named pipes, 575
phases, 573
inner joins, 657-659
input files (CLP noninteractive
mode), 144-149
invoking CLP, 145
Linux/UNIX script file, 146
requirements, 145
Windows script file, 146
INSERT import mode, 560
INSERT privilege, 442
INSERT statements, 663
INTO, 355-356
SELECT from, 665
insertable views, 393-394
INSERT_UPDATE import
mode, 560
INSPECT tool, 695
installation
case studies, ESE
new copy installation,
128-130
scratch with Setup
Wizard, 130-131
DB2 Setup Wizard, 95
db2_install script, 93,
113-114
installation, performing,
113
keywords, 114
parameters, 114
fix packs, 126-128
images, 94
customizing, 118-120
pruned, installing, 120
license certificates, 116-117
Linux/UNIX
non-root, 92
root, 91, 101
methods, 91
multiple versions and fix
packs, 121
DAS, 126
Linux/UNIX, 124
products/features
installed, viewing,
125-126
Windows, 121-123
non-root
limitations, 102-103
performing, 102
root, compared, 101
root-based features,
enabling, 103-104
payload files, 93, 115-116
response files
creating with response
file generator utility,
111-112
creating with Setup
Wizard, 109-111
defined, 107
Linux/UNIX, 112-113
PROD_SERVER.rsp
excerpt, 108-109
version-sensitive, 113
Windows, 112
Setup Wizard
DAS, configuring, 99
DB2 copy names, 99
DB2 instance, 99
ESE from scratch
installation case study,
130-131
features to install,
selecting, 98
installation, starting, 101
Linux/UNIX, launching,
97
object security, 100-101
response files, creating,
97, 109-111
settings, reviewing, 101
Windows, launching,
96-97
silent, 93
system requirements, 94
user IDs/groups required
Linux/UNIX, 106-107
Windows, 105
Windows, 90
Instance Node-Level Profile
Registry, 195
instances, 44-45
administrative authorities,
431
architectural perspective, 198
attaching to, 208-209
client, creating, 202
commands, 216
configuration parameters, 48
changes taking effect, 48
storing, 48
updating, 48
viewing, 48
configuring in Data Studio, 215
configuring with DBM
configuration file, 209-214
current parameter values,
viewing, 213-214
listing, 209-212
parameter default values,
resetting, 214
updating parameters,
212-213
creating, 45, 200-201
Linux/UNIX, 201
Windows, 200-201
current, viewing, 81
Data Studio, 214-215
configuring, 215
starting/stopping, 214
viewing, 214
databases, listing, 81-84
DB2, configuring, 99
DB2INSTANCE
environment variable, 204
default, 200
defined values, viewing, 149
design considerations,
228-229
dropping, 45, 56, 202-203
Linux/UNIX, 45
listing, 203, 230
owners, 200
partitioned databases, 64-65
Profile Registry, 194, 195
pureScale, 76-77, 202
registry variables, 47
starting, 45, 205-207
Linux/UNIX, 206-207
Windows, 205-206
stopping, 45, 207-208
user perspective, 198
viewing, 81
INSTEAD OF triggers, 396, 397
INTEGER data type, 333
integration (BigInsights), 11
data from DB2, obtaining,
11-12
filtering/aggregating/
transforming data
warehouse data, 12-13
interactive mode (CLP), 141-142
International DB2 Users Group
>IDUG), 705
Internet Protocol Version 6. See
IPv6
INTERSECT operator, 668
INTERSECT/INTERSECT ALL
operators, 670
IPv6 (Internet Protocol Version
6), 253
isolation keyword (autoconfigure
command), 227
isolation levels, 504
bound packages, viewing,
514-515
case study, 538-539
changing with
APIs, 516
call level interface,
514-515
DB2 command window,
512-513
precompile/bind
commands, 514
comparing, 511
cursor stability. See CS
read stability, 510
repeatable reads, 511
statement level, 516-517
uncommitted reads, 504
UR, testing, 547
is_populated keyword
(autoconfigure command), 227

J
Jaql, 12
JDBC (Java Database
Connectivity), 516
joins, 657-659
full outer, 659
inner, 657-659
left outer, 659
right outer, 659

K
keepalivetimeout parameters,
269
Kerberos authentication, 419,
421, 423-424
keywords
autoconfigure command, 227
CREATE TABLESPACE
statement, 317
db2_install script, 114
KRB_SERVER_ENCRYPT
authentication type, 419

L
language identifiers, 96
large objects. See LOBs
large table spaces, 311
LBAC (Label-Based Access
Control), 458-461
column-level security, 466
implementing, 462-465
data, loading, 464
granting security labels,
465
rule exemptions, 465
security labels, 463, 464
security policies, defining,
463
tables, creating, 464
overview, 458
security labels, 460-461
arrays, 460
choosing, 463
creating, 463
privileges, 449-450
sets, 460
trees, 461
types, 460

index
Index 727

security policies, 460, 463
table access examples, 465-466
views, 462
left outer joins, 659
license certificates, installing, 116-117
LIKE predicate (SELECT statement), 654-655
line continuation (CLP), 150-152
\ (backslash character), 151
! (delimiter terminator character), 151-152
Linux
CLP script file, 146
core files, 687
DAS
creating, 217
dropping, 217
users, creating, 216
DB2 installation
db2_install script, 113-114
non-root, 92
payload files manual, 115-116
response files, 112-113
root, 90
sample db2iprune input file, 118
DB2 Setup Wizard
DAS, configuring, 99
DB2 copy names, 99
DB2 instance, 99
features to install, selecting, 98
installation, starting, 101
launching, 97
response files, generating, 97
settings, reviewing, 101
db2nodes.cfg columns, 292
db2profile script file, 191-194
directory locations
local database, 241
node, 242
system database, 239
instances
creating, 45, 200-201
starting, 206-207
license certificates, installing, 116
non-root installation
fix packs, 128
limitations, 102-103
performing, 102
root, compared, 101
root-based features, enabling, 102
root installation
non-root, compared, 101
user IDs/groups required, 106-107
Linux, UNIX, Windows (LUW), 13
list applications command, 529-531
list database directory command, 237
list database directory on path/directory command, 237
LIST DATABASE PARTITION GROUP statement, 309
LIST DATABASE PARTITION GROUPS command, 68
db directory command, 237, 239, 300
db directory on drive/path command, 242
db directory on path/drive command, 237
dcs directory command, 237, 244
LIST HISTORY BACKUP command, 618
list node directory command, 237, 243
list tablespaces show detail command, 572
list utilities command, 571-572
listings
administration notification logs, 690, 696
application snapshot with locking information, 533-534
autoconfigure command, 227
catalog database command, 420
compound statements
storing in sqlpl.txt, 156
table, 157
CREATE MASK statement, 471
CREATE PERMISSION statement, 470
databases
configuration file, 217-221
lock snapshot, 534-536
snapshot with locking information, 533
db2cos script, 696
db2diag.log, 690, 691
db2profile script file for Linux/UNIX, 191-194
DBM configuration file, 209-212
derived columns, 647
EXCEPT/EXCEPT ALL, 670
GRANT syntax diagram
indexes, 444
packages, 445
routines, 446
schema privileges, 439
security labels, 449
sequences, 448
SETSESSIONUSER, 451
table spaces, 440
tables/views, 441
INGEST command, 574-575
INTERSECT/INTERSECT
ALL operators, 670
license certificate, viewing, 117
LIST HISTORY BACKUP
command, 618
list table spaces show detail, 572
list utilities command output, 571-572
load query command output, 570-571
LOCK TABLE statement, 519
MERGE statement
example, 667
syntax, 667
QUIESCE TABLESPACES
FOR TABLE command, 518
R1 and R2 table examples, 668
recursive SQL statements
children table example, 671-672
common table expression
syntax, 671
maximum number of recursive levels, 673
query example, 672-673
result, 673
redirected restore script
example, 612-613
response files
PROD_SERVER.rsp
excerpt, 108-109
Windows example, 110-111
REVOKE syntax diagram
indexes, 444
packages, 446
routines, 447
schema privileges, 439
security labels, 449
sequences, 448
SETSESSIONUSER, 451
table spaces, 441
tables/views, 443
schemas, creating implicitly, 440
SELECT statement
CASE expression, 661-662
CAST expression, 652
COALESCE function, 660
column function, 652
COUNT aggregate function, 648
COUNT function and DISTINCT clause, 649
DISTINCT clause, 648
FETCH FIRST n ROWS ONLY, 654
GROUP BY clause, 657
HAVING clause, 657
inner join, 657
isolation clause, 517
LIKE predicate, 654-655
ORDER BY clause, 656
OVER function, 662
BETWEEN predicate, 655
IN predicate, 655-656
ROWNUMBER function, 662
scalar function, 651
special registers, 649
WHERE clause, 653
SET CURRENT
ISOLATION statement, 513
set current lock timeout
command, 525
SET SESSION
AUTHORIZATION statement, 450
UNION/INTERSECT/
EXCEPT operators, 668
UNION/UNION ALL
operation, 669
load client command, 565-567
load command, 563-564
load in progress table space state, 570
load in progress table state, 569
load pending table space state, 569
load pending table state, 569
load phase (LOAD utility), 563
load query command, 570-571
LOAD utility, 562
clients, loading from, 565-567
CURSOR file format, 564
data validation against constraints, 568
exception tables, 565-567
load command, 563-564
phases, 562-563
progress monitoring, 568-572
rejected records dump files, 564-565
table space states, 569-570
table states, 569
LOBs (large objects), 53
data types, 335
defined, 53
local connections, 244
local connections from data server clients to DB2 servers, 249-250
local database directory, 50, 241-242
commands, 237
contents, viewing, 50, 242
location, 241
lock deferrals, 527-528
lock escalation, 528-529
LOCK TABLE statement, 519
lock waits, 524-526
locking
  access paths, choosing, 543-544
  case study, 538-539
databases, 517
deadlocks, 526-527, 542-545
deferrals, 527-528
escalation, 528-529
isolation levels. See isolation levels.
locklists, 528
lost updates, 502
modes
  compatibility charts, 520-524
  row, 520-521
  summary, 522
  table-level, 519-520
nonrepeatable reads, 503
phantom reads, 504
problems, preventing, 536-538
rows, 519
table spaces, 518-519
  tables, 519
troubleshooting tools
  Event Monitor, 536
  force application command, 530-532
  list applications command, 529-531
  Snapshot Monitor, 532-536
  snapshot table functions, 536
  uncommitted reads, 502
  waits, 524-526
LOCKLIST parameter, 529
locklists, 528
LOGARCHMETH1
  configuration parameter, 597-599
LOGARCHMETH2
  configuration parameter, 598
LOGARCHOPT1/
LOGARCHOPT2
  configuration parameters, 600
LOGARCHMETH1
  configuration parameter, 597-599
LOGARCHMETH2
  configuration parameter, 598
LOGARCHOPT1/
LOGARCHOPT2
  configuration parameters, 600
LOGFILSIZ parameter, 592
LOGPRIMARY parameter, 592
LOGRETM value
  (LOGARCHMETH1 parameter), 599
logs
  administration notification, 686, 690
  examples, 690
  Linux/UNIX, 686
  Windows, 686, 696
archival, 597-600
ARCHRETRYDELAY
  configuration parameter, 600
FAILARCHPATH
  configuration parameter, 600
LOGARCHMETH1
  configuration parameter, 597-599
LOGARCHMETH2
  configuration parameter, 598
LOGARCHOPT1/
LOGARCHOPT2
  configuration parameters, 600
NUMARCHRETRY
  configuration parameter, 600
archival, 597-600
logging methods versus
  recovery methods, 602
mirroring, 598-601
transaction, 589
  handling, 601-602
  log space parameters, 592
offline archive, 596
online archive, 595
overview, 590-592
primary log files, 592-594
secondary log files, 592-594
LOGSECOND parameter, 592
LUW (Linux, UNIX, Windows), 13

M
mainframe host connectivity, 20
maintenance
  automatic, 585
  utilities, 580
    REBIND, 584
    REORG, 582-584
    REORGCHK, 582-584
    RUNSTATS, 581-582
mandatory choices syntax, 24
mandatory fields syntax, 24
materialized query tables (MQTs), 370
MDC (multidimensional clustering)
  benefits, 383-384
  blocks, 384
    indexes, 385-387
    maps, 387
  cells, 385
  creating, 384
database partitioning/table partitioning, combining, 388-389
dimensions, choosing, 388
  slices, 385
memory
  buffer pools
    creating, 326-329
    default, 299
    defined, 287
    dropping, 330
    editing, 329-330
    overview, 326
    partitioned databases, 68
  locklists, 528
  mem_percent keyword
    (autoconfigure command), 227
  MERGE statement, 666-667
  mirroring logs, 598-601
  MIRRORLOGPATH configuration parameter, 601
  modules
    creating, 403
    database objects supported, 404
    defined, 403
    routine definitions, adding, 403
  MON_GET_TABLESPACE function, 324
  monitor switches, turning on, 532
  MQTs (materialized query tables), 370
  multidimensional clustering. See MDC
  multiple versions/fix packs
    installation, 121
    DAS, 126
    Linux/UNIX, 124
    products/features installed, viewing, 125-126
    Windows, 121-123
    copies, selecting, 121-122
    default copy selection, 122-123
  multitemperature data, 314-315

N
  names
    backup files, 607
    columns, specifying for export, 558-559
    copy, 99
    databases, verifying, 274
    objects, 287
    nested views, 395
    news group website, 704
    NEXT VALUE FOR statement, 403
    NEXTVAL FOR statement, 403
    nicknames (federation), 79
    node configuration file, 62-64
    database partitions, 292-297
    editing, 296
    eight two-way SMP Linux servers with eight partitions in total, 294-295
    eight-way SMP Linux server with four partitions, 294
    four four-way SMP UNIX servers with eight partitions, 295-296
    Linux/UNIX columns, 292
    Windows columns, 293
    editing, 64
    example, 62-63
    location, 294
    storing, 63
    node directory, 50-51, 242-244
    commands, 237
    contents, viewing, 50, 243
    location, 242
    remote client connections databases, 255
    host databases, 259
    server connectivity information, verifying, 272-273
  non-delimited ASCII (ASC) file format, 556
  noninteractive mode (CLP), 143-144
  noninteractive mode using input files (CLP), 144-149
  input file requirements, 145
  invoking CLP, 145
  Linux/UNIX, 146-147
  Windows, 146
  non-root installations
    fix packs, 128
    limitations, 102-103
    performing, 102
    root, compared, 101
    root-based features, enabling, 102
  normal table space state, 569
  normal table state, 569
  NOT ENFORCED clause
    (CREATE TABLE statement), 360
  not load restartable table state, 569
  not logged initially tables, 362-363
  NOTE LOGGED INITIALLY clause (CREATE TABLE statement), 362
  NOTIFYLEVEL configuration parameter, 686, 687
  NULL values
    SELECT statement, 660-661
    tables, 346-347
  NUMARCHRETRY configuration parameter, 600
  numeric data types, 333-334
  num_local_apps keyword
    (autoconfigure command), 227
  num_remote_apps keyword
    (autoconfigure command), 227
  num_stmts keyword
    (autoconfigure command), 227
objects. See also specific objects creating/dropping, 288
DDL, generating, 579-580
default, 299
IDs, 320
interaction, 288-290
list of, 286-287
names, 287
overview, 285
privileges, 438
granting, 438
implicit, 452-453
indexes, 444-445
ownership, transferring, 456
packages, 445
revoxing, 438
roles, 454-456
routines, 446-447
schema, 438
security labels, 449-450
sequences, 448-449
SESSION_USER special register, 450
SETSESSIONUSER, 451
table space, 440-441
tables, 441-444
views, 441-444
security, enabling, 100-101
OFF value (LOGARCHMETHI parameter), 599
offline and not accessible table space state, 570
offline archive logs, 596
online archive logs, 595
operating system authorities, 435
operators
EXCEPT/EXCEPT ALL, 670-671
INTERSECT/INTERSECT ALL, 670
UNION, 668-669
UNION ALL, 668-669
optional fields syntax, options. See parameters
Oracle compatibility features, 676-681
concurrency, 681
data types/SQL/packages, 680-681
enabling, 676
PL/SQL, 681
Data Conversion Workbench (DCW), 681
product and functionality mapping, 675-677
Redbook Oracle to DB2 Conversion Guide
Compatibility Made Easy website, 675
terminology mapping, 677
ORDER BY clause (SELECT statement), 656-657
ORGANIZE BY COLUMN clause (CREATE TABLE statement), 378
OVER function (SELECT statement), 662-663
ownership transfers, 491
parameters
AUTHENTICATION, 418
authentication plug-ins, 426
CLP commands, 147-149
auto-commit, disabling, 148
changes effective across all CLP sessions, 148
instance defined values, viewing, 149
interactive mode, 148
turning on/off, 147
values, resetting, 149
configuration, 48-49
case-sensitivity, 81
database level, 48-49
DIAGLEVEL, 686, 689
DIAGPATH, 688
DIAGSIZE, 689
graphical tool, 48
instance level, 48
MAXLOCKS, 520
NOTIFYLEVEL, 686, 687
Database Manager
configuration files
current values, viewing, 213-214, 222-223
editing, 84
listing, 209-212
resetting, 214, 223
updating, 212-213, 221-222
verifying, 272
viewing, 217-221
db2_install script, 114
db2set command, 197
keepalivetimeout, 269
LOCKLIST, 529
log space, 592
logging configuration
ARCHRETRYDELAY, 600
FAILARCHPATH, 600
LOGARCHMETH1, 597-599
LOGARCHMETH2, 598
LOGARCHOPT1/
LOGARCHOPT2, 600
MIRRORLOGPATH, 601
NUMARCHRETRY, 600
TRUST_ALLCLNTS, 427
NO value, 429
YES value, 428
TRUST_CLNAUTH
SERVER value, 429
TRUST_CLNTAUTH, 428
partitioning keys, 72
partitioning maps, 71
partitions (databases), 58-62, 290
buffer pools, 68
catalog, 67
commands, executing, 69-70
coordinator, 69
creating, 65-66
data distribution, 71-72
Data Server client connection, 62
DB2NODE environment
variable, 70-71
defined, 58
groups, 67-68
classifications, 305-306
creating, 67, 307-308
default, 67, 299, 306
defined, 286
dropping, 68, 310
editing, 308
listing, 68, 309
overview, 305
partitions, adding/
dropping, 68
illustration, 60
instances, 64-65
multiple, 60
node configuration file,
62-64, 292-297
editing, 64
example, 62-63
storing, 63
overview, 290-292
server interpartition
communication, 62
single, 58
SQL statements, executing,
69-70
table partitioning/MDC,
combining, 388-389
table spaces, 69
partitions (tables), 363-366
attaching, 365
creating, 363-353
database partitioning/MDC,
combining, 388-389
detaching, 366
range boundaries, 364-365
payload files, installing, 93,
115-116
PC version of IXF (PC/IXF) file
format, 557
PC/IXF (PC version of IXF) file
format, 557
performance
manager, 42
queries, 382
phantom reads, 504
platforms
cross-platform development,
27
IBM Big Data, 6-8
costs, reducing, 9
raw data analysis, 8-9
streaming data analysis,
10
unlocking big data, 8
warehouse simplification,
10-11
plug-ins (GSS), 424-425
Database Manager
parameters, 426
example, 425
samples, 425
unsupported, 425
ports
accessibility, testing, 273
TCP/IP, 62
precompile command, 514
PREVIOUS VALUE FOR
statement, 403
PREVVAL FOR statement, 403
primary keys, 353, 354
primary log files, 592-594
privileges, 438
case study, 486-490
granting, 438
implicit, 452-453
indexes, 444-445
managing with IBM Data
Studio, 486
ownership, transferring, 456
packages, 445
revoking, 438
roles, 454-456
assigning privileges with
roles, 455
assigning privileges
without roles example,
455
granting, 454
revoking, 454
routines, 446-447
schema, 438-440
security labels, 449-450
sequences, 448-449
SESSION_USER special
register, 450
SETSESSIONUSER, 451
system catalog views,
484-485
table space, 440-441
tables, 441-444
views, 441-444
problems
connectivity, 269
client configuration,
verifying, 272-274
client-host diagnostic
flowchart, 276
Index 733

client-server diagnostic flowchart, 275
database connection flowchart, 269
diagnosis aids, 277-278
server configuration, verifying, 270-272
diagnosing. See troubleshooting
error messages, 684
locking. See troubleshooting, locking
searching for known, 699
processes (CLP)
back-end, 152-153
front-end, 152-153
PROD_SERVER.rsp excerpt, 108-109
profile registry
categories, 47
defined, 194
editing, 195
registries, 194-195
variables. See registry variables
protocols
communication
client to remote DB2
server connections, 252
DB2 instance, configuring, 99
IPv6, 253
TCP/IP. See TCP/IP
pruning installation images, 118-120
PureData System, 10-11, 26-27
pureQuery Runtime for LUW, 42
pureScale, 3, 72
architecture, 73-74
CF (cluster caching facility), 75
CS (Cluster Services), 75
file system (GPFS), 75-76
instances, 76-77, 202
interconnectivity, 74
pureXML, 2
Q
queries
databases, 51
performance, 382
time travel, 376-377
time travel. See temporal tables
workload tuner, 42
Query Tuner Workflow Assistant, 167-170
advisor recommendations, reviewing, 170
SQL statements, capturing for tuning, 168
tuning query, 168
QUIESCE TABLESPACES FOR TABLE command, 518
quiesced: EXCLUSIVE table space state, 570
quiesced: SHARE table space state, 570
quiesced: UPDATE table space state, 570
quiescing table spaces, 518-519
R
raw data analysis, 8-9
RBAC (Role-Based Access Control), 454-456
assigning privileges, 455
granting, 454, 491
revoking, 454, 491
RCAC (Row and Column Access Control)
benefits, 479
built-in functions, 468
case scenario, 476-478
case study, 492-493
column masks
creating, 471
enforcing, 473
defined, 467
DELETE operations, 473
eamples, 475-476
implementing, 473-475
activating, 475
data, loading, 474
roles/row permissions, creating, 474-475
tables, creating, 474
INSERT operations, 473
overview, 467-468
row permissions
creating, 467-468
enforcing, 472
session variables, 469
UPDATE operations, 473
RDMA (remote directory memory access), 74
read access only table state, 569
read stability (RS), 510
read-only views, 394
REAL data type, 334
REBIND utility, 584
RECOVER DATABASE command, 620-621
recoverable versus nonrecoverable databases, 602
recovery
Advanced Recovery Feature, 18
databases
backups, restoring, 608-609
RECOVER DATABASE command, 620-621
rolling forward, 614-616
dropped tables, 616-618
history file, 618-620
logging
archival, 597-600
circular, 596
infinite, 598
mirroring, 598-601
recovery methods, compared, 602
overview, 585
recoverable versus nonrecoverable databases, 602
redirected restores, 610-614 strategies, 586-587
table spaces backups, restoring, 609-610
rolling forward, 616
transaction logs, 589
active, 594-595
handling, 601-602
log space parameters, 592
offline archive, 596
online archive, 595
overview, 590-592
primary log files, 592-594
secondary log files, 592-594
transactions, 587
types, 588
roll forward, 589
version, 588
recursive SQL statements, 671-673
children table example, 671-672
common table expression syntax, 671
maximum number of recursive levels, 673
query example, 672-673
results, 673
Redbook Oracle to DB2 Conversion Guide:
Compatibility Made Easy website, 675
redirected restores, 610-614 manual, 611
scripts, 611-614
REFERENCES clause (CREATE TABLE statement), 354
REFERENCES privilege, 442
referential constraints, 352
referential integrity, 352-353
case study example, 406
foreign keys, 353, 354
primary keys, 353, 354
relation columns and XML document elements, 355
SQL operations implications, 355-358
deleting rows from dependent tables, 357
deleting rows from parent tables, 356-357
inserting into dependent tables, 355-356
inserting into parent tables, 355
updating rows from dependent tables, 357-358
updating rows from parent table, 357
unique keys, 354
REFRESH DEFERRED clause (CREATE TABLE statement), 370
registries editing, 195
listing of, 194-195
registry variables, 46-47, 194-197
available, viewing, 47
case-sensitivity, 81
changes taking effect, 47
current, viewing, 47
DB2COMM, 271
DB2_
EVALUNCOMMITTED, 527
DB2_GRP_LOOKUP, 482-483
DB2_SKIPDELETED, 528
DB2_SKIPINSERTED, 528
instance-level versus global-level, 47
level indicators, 196
listing, 230
search order, 196-197
setting, 196, 230
starting/stopping instances for changes to take effect, 197
unsetting, 230
value, setting, 47
regular table spaces, 311
remote connections applications to servers, 264-265
automatic client reroute, 267-268
connection timeouts, 268-269
TCP/IP keepalive timeouts, 269
utilities, binding, 265-266
clients, accepting, 259
clients to host servers, 258
TCP/IP connectivity, 259
DB2 Connect gateway, 262-263
clients to servers, 251
client connections, accepting, 252-253
client requirements, 251-252
communication protocols supported, 252
server requirements, 252
TCP/IP, 253-256
Databases, 246-247
host DB2 server, 247
remote directory memory access (RDMA), 74
removing. See dropping reorg in progress table space state, 570
REORG utility, 582-584
reorgchk command, 582-583
REORCGHK utility, 582-584
repeatable reads (RRs), 511
Drop in Progress Table Space State, 570
remote directory memory access (RDMA), 74
removing. See dropping reorg in progress table space state, 570
REORG utility, 582-584
reorgchk command, 582-583
REORCGHK utility, 582-584
repeatable reads (RRs), 511
repeated options syntax, 24
REPLACE import mode, 560
Replication Center, 178
replication support, 22-23
required user IDs/groups
  Linux/UNIX, 106-107
  Windows, 105
reset admin cfg command, 217
reset db cfg command, 226
reset db cfg for database_name
  command, 223
reset dbm cfg command,
  214, 216
resource websites
Big Data University, 705
blogs, 706
conferences, 705
Data Server Client, 701
database on the Cloud, 705
DB2
database administration
  and performance
classroom courses, 703
documentation, 704
Express-C, 701
Information Center, 702
technical support, 702
DB2 Essentials:
Understanding DB2 in a
  Big Data World, website,
  701
DB2 for Linux, UNIX, and
  Windows main web page,
  702
DeveloperWorks, 704
IBM
  Business Partners, 706
certification program and
tutorials, 703
Data Magazine, 705
Data Studio, 702
DB2 for LUW forum, 704
DB2 support, 702
Information Management
  Best Practices, 704
passport advantage, 701
Redbooks, 704
software, buying, 701
software support, 704
news group, 704
technical materials library,
  704
user groups, 705
response files
  creating
    response file generator
    utility, 111-112
    Setup Wizard, 109-111
    DB2 installation, performing
      Linux/UNIX, 112-113
      Windows, 112
defined, 107
generating, 97
generator utility, 111-112
  PROD_SERVER.rsp excerpt,
    108-109
version-sensitive, 113
restarting databases, 225
RESTORE command, 611
RESTORE DATABASE
  command, 608
restore in progress table space
  state, 570
restore pending table space state,
  570
restoring. See recovery
REVOKE EXEMPTION ON
  RULE statement, 465
REVOKE statement
  database privileges, 438
  syntax diagrams for
    privileges
      indexes, 444
      packages, 446
      routines, 447
      schema, 439
      security labels, 449
      sequences, 448
    SETSESSIONUSER, 451
table spaces, 441
  tables/views, 443
right outer joins, 659
Role-Based Access Control. See
  RBAC
roles, 453-456
  assigning privileges examples
    with roles, 455
    without roles, 455
  granting/revoking, 454, 491
rolling forward
  progress table space state,
    570
  pending table space state, 570
databases, 589
table spaces, 616
root installation, 101
routines
  Database Routines Editor and
    Debugger, 173-174
  privileges, 446-447
  Row and Column Access
  Control. See RCAC
  ROWNNUMBER function
    (SELECT statement), 662-663
rows, 338
  compression
    advantages/disadvantages,
      366
dictionary, 367
  enabling, 368-369
types, 367-368
deleting, 664
lock modes
  compatibility chart, 524
  overview, 520-521
locking, 519
permissions
  creating, 467-468
  enforcing, 472
security labels, 460
subsets, selectively viewing,
  394
temporal tables, inserting,
  375
UNION/UNION ALL operators, 668-669
updating, 663
RRs (repeatable reads), 511
RS (read stability), 510
RUN scripts, 157
RUNSTATS utility, 580-582
SAMPLE database, 304
saving configuration files, 229
scalar functions, 651
schemas, 330-332
creating, 331, 407
dropping, 332
overview, 330-331
privileges, 438-440
granting, 438
implicitly creating, 440
revoking, 439
session, setting, 332
scripts
db2_install, 93, 113-114
installation, performing, 113
keywords, 114
parameters, 114
EDIT, 157-158
environment variables, setting, 191-194
GET, 157
RUN, 157
SECADM authority, 431
secondary log files, 592-594
Secured Socket Layer (SSL), 457
security, 415
administrative authorities, 431
case study, 486-490
database-level, 431
hierarchy, 432-434
instance-level, 431
managing with IBM Data Studio, 484
operating system/security facility groups, 435
reviewing, 484
system catalog views, 484-485
system groups, assigning, 435
authentication. See authentication
case study
administrative authorities/
privileges, 486-490
encryption, 491
ownership transfers, 491
RCAC, 492-493
roles, granting/revoking, 491
encryption/decryption, 456-458
built-in functions, 457
case study, 491
DATA_ENCRYPT authentication value, 457-458
SSL, 457
facility group authorities, 435
granting/revoking roles case study, 491
labels, 460-461
arrays, 460
components, choosing, 463
creating, 463
defining, 464
granting, 465
privileges, 449-450
sets, 460
trees, 461
types, 460
LBAC, 458-461
column-level security, 466
implementing, 462-465
overview, 458
security label components, 460-461
security policies, 460
table access examples, 465-466
views, 462
objects, enabling, 100-101
overview, 415-417
ownership transfers, 491
privileges
case study, 486-490
granting, 438
implicit, 452-453
indexes, 444-445
managing with IBM Data Studio, 486
ownership, transferring, 456
packages, 445
revoking, 438
roles. See privileges, roles
case study, 486-490
case scenario, 476-478
creating column masks, 471
creating row permissions, 469-470
defined, 467
RCAC
benefits, 479
built-in functions, 468
case scenario, 476-478
case study, 492-493
creating column masks, 471
creating row permissions, 469-470
defined, 467
DELETE operations, 473
enforcing column masks, 473
enforcing row permissions, 472
examples, 475-476
implementing, 473-475
INSERT operations, 473
overview, 467-468
session variables, 469
UPDATE operations, 473
trusted contexts, 479-481
Windows, 481
DB2_GRP_LOOKUP registry variable, 482-483
extended, 483-484
global groups, 481
local groups, 482
SELECT privilege, 442
SELECT statements
clauses
FROM, 653
DISTINCT, 648-649
FETCH FIRST n ROWS ONLY, 654
GROUP BY, 657
HAVING, 657
isolation, 517
ORDER BY, 656-657
WHERE, 653
DELETE statement
combination, 664
derived columns, 646-648
examples, 646
export command, 558
expressions
CASE, 661-662
CAST, 652-653
functions
column, 652
COUNT, 648
OVER, 662-663
scalar, 651
INSERT statement
combination, 665
joins, 657-659
full outer, 659
inner, 657-659
left outer, 659
right outer, 659
NULL values, 660-661
IN predicate, 655-656
predicates
BETWEEN, 655
LIKE, 654-655
ROWNUMBER function, 662-663
special registers, 649-651
UPDATE statement
combination, 665
semicolons (;), commands/statements, 70
sequences
creating, 401-402
defined, 287
dropping, 403
editing, 403
privileges, 448-449
tokens syntax, 25
values, retrieving, 403
SERVER authentication type, 419, 421
SERVER_ENCRYPT authentication type, 419, 421
servers
authentication types,
configuring, 417-418
client connections, accepting, 252-253, 259
configuration, verifying, 270-272
configuration parameters, 272
database existence, 271
DB2COMM registry variable, 271
connectivity, 235-236
DAS, 45
commands, listing of, 217
configuring, 99
creating, 217
dropping, 217
multiple DB2 copies, 126
overview, 216
resetting admin configuration file, 217
starting, 45, 217
stopping, 45, 217
updating admin configuration file, 217
users, creating, 216
viewing admin configuration file, 217
host
client connections, accepting, 259
Connect gateway, 262-263
TCP/IP connectivity, 259
interpartition communication, 62
local connectivity, 249-250
remote connectivity. See remote connections
users, authenticating, 422-423
session variables, 469
SESSION_USER register, 450, 650
set command, 189
SET CURRENT ISOLATION statement, 513
set current lock timeout command, 525
set integrity pending table state, 569
SET INTEGRITY statement, 359
SET SCHEMA command, 332
SET SESSION AUTHORIZATION statement, 450
SETSESSIONUSER privilege, 451
setup command, 112
setup tools, 42-43, 175
Configure DB2.NET Data Provider, 176
Default DB2 and Database Client Interface Selection Wizard, 177
First Steps, 176-177
Replication Center, 178
Setup Wizard, 92
das, configuring, 99
DB2 copy names, 99
DB2 instance, 99
ESE from scratch installation case study, 130-131
features to install, selecting, 98
installation, starting, 101
launching Linux/UNIX, 97
Windows, 96-97
object security, 100-101
response files, creating, 97, 109-111
settings, reviewing, 101
silent install, 93
single-byte character large objects (CLOBs), 336
slices (MDC tables), 385
SMALLINT (small integer data type), 333
SMS (system-managed space) table spaces, 310
creating, 319-320
absolute/relative paths, 319
containers, 319-320
DMS, compared, 323
object IDs, 320
overview, 318
Snapshot Monitor locking, troubleshooting, 532-536
application snapshots, 533-534
database lock snapshot, 534-536
database snapshots, 533
monitor switches, turning on, 532
snapshot table functions, 536
sourced functions, 400
special registers, 649-651
SQL (Structured Query Language)
; (semicolons), 70
Big SQL, 645
buffers, 156-158
compound statement example, 156-157
GET/RUN scripts, 157
output, 158
scripts, editing, 157-158
derived columns, 646-648
functions, 401
Oracle compatibility, 676-681
overview, 645
Query Builder, 172
recursive, 671-673
current table example, 671-672
common table expression syntax, 671
maximum number of recursive levels, 673
query example, 672-673
result, 673
Reference for Cross-Platform Development website, 27
statements. See statements tables, modifying
EXCEPT/EXCEPT ALL operators, 670-671
INTERSECT/
INTERSECT ALL operators, 670
UNION/UNION ALL operators, 668-669
website, 646
SQL and XQuery editor, 171-172
SQLADM authority, 431
SRVCON_GSSPLUGIN_LIST parameter, 426
SRVCON_PW_PLUGIN parameter, 426
SRV_PLUGIN_MODE parameter, 426
SSL (Secured Socket Layer), 457
starting
CLPlus, 154
Command Window, 140
Configuration Advisor, 227
das, 45, 217
databases, 224
installation, 101
instances, 45, 205-207, 214
Linux/UNIX, 206-207
Windows, 205-206
statements, 33-34
; (semicolons), 70
ALTER
BUFFERPOOL, 329
DATABASE PARTITION GROUP, 68, 308
MASK, 473
MODULE, 402
PERMISSION, 472
PROCEDURE, 399
SEQUENCE, 403
STOGROUP, 314
TABLE, 358, 368, 519
TABLESPACE, 325
capturing for tuning, 168
case-sensitivity, 81
column masks, enforcing, 473
CONNECT, 518
CREATE
BUFFERPOOL, 68, 326
database objects, 288
DATABASE PARTITION GROUP, 307
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUNCTION</td>
<td>400</td>
</tr>
<tr>
<td>INDEX</td>
<td>379</td>
</tr>
<tr>
<td>MASK</td>
<td>471</td>
</tr>
<tr>
<td>MODULE</td>
<td>402</td>
</tr>
<tr>
<td>PERMISSION</td>
<td>470</td>
</tr>
<tr>
<td>PROCEDURE</td>
<td>398</td>
</tr>
<tr>
<td>SCHEMA</td>
<td>331</td>
</tr>
<tr>
<td>SECURITY LABEL COMPONENT</td>
<td>463</td>
</tr>
<tr>
<td>SEQUENCE</td>
<td>401</td>
</tr>
<tr>
<td>STOGROUP</td>
<td>313</td>
</tr>
<tr>
<td>TYPE</td>
<td>337</td>
</tr>
<tr>
<td>UNIQUE</td>
<td>352</td>
</tr>
<tr>
<td>USER MAPPING</td>
<td>80</td>
</tr>
<tr>
<td>USER TEMPORARY TABLESPACE</td>
<td>371</td>
</tr>
<tr>
<td>VIEW</td>
<td>390, 394</td>
</tr>
<tr>
<td>CREATE TABLE</td>
<td>341</td>
</tr>
<tr>
<td>check constraints</td>
<td>358</td>
</tr>
<tr>
<td>DATA INITIALLY</td>
<td></td>
</tr>
<tr>
<td>DEFERRED clause</td>
<td>370</td>
</tr>
<tr>
<td>informational constraints</td>
<td>360</td>
</tr>
<tr>
<td>NOT LOGGED</td>
<td></td>
</tr>
<tr>
<td>INITIALLY clause</td>
<td>362</td>
</tr>
<tr>
<td>ORGANIZE BY</td>
<td></td>
</tr>
<tr>
<td>COLUMN clause</td>
<td>378</td>
</tr>
<tr>
<td>partitions</td>
<td>363</td>
</tr>
<tr>
<td>REFERENCES clause</td>
<td>354</td>
</tr>
<tr>
<td>REFRESH DEFERRED clause</td>
<td>370</td>
</tr>
<tr>
<td>row compression</td>
<td>368</td>
</tr>
<tr>
<td>VALUE COMPRESSION clause</td>
<td>369</td>
</tr>
<tr>
<td>CREATE TABLESPACE</td>
<td>69, 317</td>
</tr>
<tr>
<td>automatic storage table spaces</td>
<td>322</td>
</tr>
<tr>
<td>DMS table spaces</td>
<td>320</td>
</tr>
<tr>
<td>SMS table spaces</td>
<td>319</td>
</tr>
<tr>
<td>DCL</td>
<td>34</td>
</tr>
<tr>
<td>DDL</td>
<td>33</td>
</tr>
<tr>
<td>DELETE</td>
<td>664</td>
</tr>
<tr>
<td>FROM</td>
<td>356-357</td>
</tr>
<tr>
<td>SELECT from</td>
<td>664</td>
</tr>
<tr>
<td>DML, 33</td>
<td></td>
</tr>
<tr>
<td>DROP</td>
<td></td>
</tr>
<tr>
<td>BUFFERPOOL</td>
<td>330</td>
</tr>
<tr>
<td>database objects</td>
<td>288</td>
</tr>
<tr>
<td>DATABASE</td>
<td></td>
</tr>
<tr>
<td>PARTITION, 68</td>
<td></td>
</tr>
<tr>
<td>INDEX</td>
<td>382</td>
</tr>
<tr>
<td>MASK</td>
<td>473</td>
</tr>
<tr>
<td>PERMISSION</td>
<td>472</td>
</tr>
<tr>
<td>PROCEDURE</td>
<td>399</td>
</tr>
<tr>
<td>SCHEMA</td>
<td>332</td>
</tr>
<tr>
<td>SEQUENCE</td>
<td>403</td>
</tr>
<tr>
<td>STOGROUP</td>
<td>314</td>
</tr>
<tr>
<td>TABLE</td>
<td>344</td>
</tr>
<tr>
<td>TABLESPACE</td>
<td>326</td>
</tr>
<tr>
<td>TYPE</td>
<td>338</td>
</tr>
<tr>
<td>executing in partitioned databases</td>
<td>69-70</td>
</tr>
<tr>
<td>GRANT</td>
<td>indexes, 444</td>
</tr>
<tr>
<td>packages, 445</td>
<td></td>
</tr>
<tr>
<td>routines, 446</td>
<td></td>
</tr>
<tr>
<td>security labels, 449</td>
<td></td>
</tr>
<tr>
<td>sequences, 448</td>
<td></td>
</tr>
<tr>
<td>SETSESSIONUSER</td>
<td>451</td>
</tr>
<tr>
<td>table spaces, 440</td>
<td></td>
</tr>
<tr>
<td>tables/views, 441</td>
<td></td>
</tr>
<tr>
<td>INSERT, 663</td>
<td></td>
</tr>
<tr>
<td>INTO</td>
<td>355-356</td>
</tr>
<tr>
<td>SELECT from</td>
<td>665</td>
</tr>
<tr>
<td>isolation levels</td>
<td>516-517</td>
</tr>
<tr>
<td>LIST DATABASE</td>
<td>PARTITION GROUP, 309</td>
</tr>
<tr>
<td>LOCK TABLE</td>
<td>519</td>
</tr>
<tr>
<td>MERGE</td>
<td>666-667</td>
</tr>
<tr>
<td>NEXT VALUE FOR</td>
<td>403</td>
</tr>
<tr>
<td>NEXTVAL FOR</td>
<td>403</td>
</tr>
<tr>
<td>PREVIOUS VALUE FOR</td>
<td>403</td>
</tr>
<tr>
<td>PREVVAL FOR</td>
<td>403</td>
</tr>
<tr>
<td>recursive, 671-673</td>
<td></td>
</tr>
<tr>
<td>children table example</td>
<td>671-672</td>
</tr>
<tr>
<td>common table expression syntax</td>
<td>671</td>
</tr>
<tr>
<td>maximum number of recursive levels</td>
<td>673</td>
</tr>
<tr>
<td>query example</td>
<td>672-673</td>
</tr>
<tr>
<td>result</td>
<td>673</td>
</tr>
<tr>
<td>referential integrity implications</td>
<td>355-358</td>
</tr>
<tr>
<td>DELETE FROM</td>
<td>356-357</td>
</tr>
<tr>
<td>INSERT INTO</td>
<td>355-356</td>
</tr>
<tr>
<td>UPDATE</td>
<td>357-358</td>
</tr>
<tr>
<td>REVOKE. See REVOKE statement</td>
<td></td>
</tr>
<tr>
<td>row permissions, enforcing</td>
<td>472</td>
</tr>
<tr>
<td>SELECT statements</td>
<td>648</td>
</tr>
<tr>
<td>SET CURRENT</td>
<td></td>
</tr>
<tr>
<td>ISOLATION</td>
<td>513</td>
</tr>
<tr>
<td>SET INTEGRITY</td>
<td>359</td>
</tr>
<tr>
<td>SET SESSION AUTHORIZATION</td>
<td>450</td>
</tr>
<tr>
<td>table data, modifying</td>
<td>663-664</td>
</tr>
<tr>
<td>DELETE</td>
<td>664</td>
</tr>
<tr>
<td>INSERT</td>
<td>663</td>
</tr>
<tr>
<td>MERGE</td>
<td>666-667</td>
</tr>
<tr>
<td>UNION/INTERSECT/EXCEPT operators</td>
<td>668</td>
</tr>
<tr>
<td>UPDATE</td>
<td>663-664</td>
</tr>
<tr>
<td>TRANSFER OWNERSHIP</td>
<td>456</td>
</tr>
<tr>
<td>TRIGGER</td>
<td>396</td>
</tr>
<tr>
<td>UPDATE</td>
<td>663-664</td>
</tr>
<tr>
<td>referential integrity implications</td>
<td>357-358</td>
</tr>
<tr>
<td>SELECT from</td>
<td>665</td>
</tr>
<tr>
<td>XQuery, 34-35</td>
<td></td>
</tr>
<tr>
<td>case-sensitivity</td>
<td>81</td>
</tr>
<tr>
<td>FLWOR expression</td>
<td>35</td>
</tr>
<tr>
<td>XPath</td>
<td>34-35</td>
</tr>
<tr>
<td>states</td>
<td>logs</td>
</tr>
</tbody>
</table>
restoring backups, 609-610
redirected restores, 610-614
rolling forward, 616
SMS
creating, 319-320
overview, 318
states, 569-570
status, checking, 572
storage groups, 312-314
SYSCATSPACE, 52, 67
SYSTOOLSPACE, 52
SYSTOOLTMPSPACE, 52
temporary, 304
TEMPSPACE1, 52
user tables, 343-344
USERSPACE1, 52
tables
bitemporal, 377
case study, 407-408
classification, 339-341
column-organized, 338, 377-379
creating, 378
populating, 379
row-organized
conversions, 379
compressing, 369
constraints, 350
check, 358-360, 406-407
data validation, 568
informational, 360-361
referential, 352
See also referential integrity
unique, 351-352
creating, 405
data, loading, 464, 474
data modifying SQL statements, 663-664
DELETE, 664
INSERT, 663
MERGE statement, 666-667
UNION/INTERSECT/EXCEPT operators, 668
UPDATE, 663-664
default values, 344-346
defined, 53, 287, 338
dropped, recovering, 616-618
EXCEPT/EXCEPT ALL operators, 670-671
exception, 565-567
identity columns, 347-350
inserting, 663
INTERSECT/INTERSECT ALL operators, 670
joins, 657-659
full outer, 659
inner, 657-659
left outer, 659
right outer, 659
LBAC, creating, 464
locking, 519
compatibility chart, 520
overview, 519-520
materialized query (MQTs), 370
MDC
benefits, 383-384
block indexes, 385-387
block maps, 387
blocks, 384
cells, 385
creating, 384
database partitioning/table partitioning, combining, 388-389
dimensions, choosing, 388
slices, 385
not logged initially, 362-363
NULL values, 346-347
partitions, 363-366
attaching, 365
creating, 363-353
database partitioning/MDC, combining, 388-389
detaching, 366
range boundaries, 364-365
physical characteristics
statistics, updating, 581-582
privileges, 441-444
referential integrity, 352-353
case study example, 406
foreign keys, 353-354
primary keys, 353-354
relational columns
and XML document elements, 355
SQL operations
implications, 355-358
unique keys, 354
reorganizing, 582-584
rows. See rows
snapshot functions, 536
states, 569
system catalog, 340-341
temporary, 372-377
creating, 374-375
information, retrieving, 375-376
rows, inserting, 375
time travel queries, 376-377
versioning, enabling, 375
temporary, 370-372
CGTTs, 372
creating, 371
DGGTTs, 371
indexes, 372
types, 371
transition, 665
UNION/UNION ALL operations, 668-669
user, 341-344
based on another table
definition, creating, 343
creating, 341-342
dropping, 344
query result-based,
creating, 344
table spaces, 343-344
TARGET PRINCIPAL
authentication type, 421
Task Launcher (Data Studio), 161-162
TCP/IP
client-server connection
problems
client configuration, verifying, 272-274
server configuration, verifying, 270-272
keepalive timeouts, 269
listeners, enabling, 279
remote server connectivity, 253-256, 259
server interpartition
communication, 62
technical materials library, 704
technical support website, 704
temporal tables, 372-377
creating, 374-375
information, retrieving, 375-376
rows, inserting, 375
time travel queries, 376-377
versioning, enabling, 375
temporary table spaces, 311
temporary tables, 370-372
CGTTs
creating, 372
DGTTs, compared, 371
creating, 371
DGTTs
CGTTs, compared, 371
creating, 371
indexes, 372
types, 371
TEMPSPACE1 table space, 52, 311
text base installation. See db2_install script
Text Search commands, 37
time and date data types, 336
TIME data type, 336
time travel queries, 376-377
TIMESTAMP data type, 336
tokens (sequences), 25
tools. See also utilities
command-line. See
command-line tools
Configuration Advisor
configuration parameter
values, passing, 228
running automatically/
manually, 228
starting, 227
Configure Parameters Editor, 48
Control Center, 138
Database Routines Editor and
Debugger, 173-174
db2look, 555
db2relocatedb, 555
dp2pd, 181
Index Advisor, 382
information, 43, 179
Check for DB2 Updates
menu option, 43, 180
Information Center. See
Information Center
InfoSphere
Data Architect (IDA), 42
Optim portfolio tools, 42
locking, troubleshooting
Event Monitor, 536
force application
command, 530-532
list applications command, 529-531
Snapshot Monitor,
532-536
snapshot table functions,
536
overview, 137
set-up, 42-43, 175
Configure DB2.NET Data
Provider, 176
Default DB2 and Database
Client Interface
Selection Wizard, 177
First Steps, 176-177
Replication Center, 178
SQL and XQuery editor,
171-172
SQL Query Builder, 172
troubleshooting
administration notification
logs, 686, 690
core files, 687
db2ckbkp, 699
db2cos script, 695-696
db2dart, 694-695
db2diag.log, 686,
690-691, 692
db2fodc, 697-698
db2mtrk, 699
db2pdcfg, 697
db2support, 692-693
db2top, 699
db2val command, 692
diaglevel parameter,
689
diagpath parameter,
688
diagsize parameter,
689
dump files, 687
FODC, 686
help (? command,
684-685
INSPECT, 695
notifylevel
parameter, 687
summary, 698
trace utility, 693-694
trap files, 686-687
Tpm keyword (autoconfigure
command), 227
trace utility, 693-694
trail copies, 94
transaction logs, 589
active, 594-595
defined, 53
handling, 601-602
log space parameters, 592
offline archive, 596
online archive, 595
overview, 590-592
primary log files, 592-594
secondary log files, 592-594
transactions
defined, 502
recovery, 587
TRANSFER OWNERSHIP
statement, 456
transition tables, 665
trap files, 686-687
trees (security labels), 461
trial copies, 94
triggers, 396-397
BEFORE, 397
AFTER, 397
classifications, 396
creating, 396
defined, 287
INSTEAD OF, 397
troubleshooting
connectivity, 269
client configuration,
verifying, 272-274
client-host diagnostic
flowchart, 276
client-server diagnostic
flowchart, 275
database connection
flowchart, 269
diagnosis aids, 277-278
server configuration,
verifying, 270-272
error messages, 684
help (?) command, 684-685
information, collecting
administration notification
logs, 686, 690
core files, 687
db2diag.log, 685, 690-691
DIAGLEVEL parameter, 689
DIAGPATH parameter, 688
DIAGSIZE parameter, 689
dump files, 687
FODC, 686
NOTIFYLEVEL parameter, 687
trap files, 686-687
locking
Event Monitor, 536
force application
command, 530-532
list applications command, 529-531
Snapshot Monitor, 532-536
snapshot table functions, 536
overview, 683-684
searching for known
problems, 699
tools
db2ckbkp, 699
db2cos script, 695-696
db2dart, 694-695
db2diag.log, 692
db2fodc, 697-698
db2mtrk, 699
db2pdcfg, 697
db2support, 692-693
db2top, 699
db2val command, 692
INSPECT, 695
summary, 698
trace utility, 693-694
TRUST_ALLCLNTS parameter, 427
NO value, 429
YES value, 428
TRUST_CLNAUTH parameter, 429
TRUST_CLNTAUTH parameter, 428
tuned contexts, 479-481
TSM:[management class name]
value (LOGARCHMETH1 parameter), 599
tuning
query workload, 42
SQL statements, 168

U
UDFs (user-defined functions)
classifications, 400-401
creating, 400
defined, 287
UDTs (user-defined data types), 337-338
classifications, 337
creating, 337
dropping, 338
unavailable table state, 569
uncatalog db command, 237
uncatalog DCS database
command, 237
uncatalog node command, 237
concurrent data access, 504
testing, 502
UNION ALL operator, 668-669
UNION operators, 668-669
unique constraints, 351-352
unique keys, 354
UNIX
CLP script file, 146
core files, 687
DAS
creating, 217
dropping, 217
users, creating, 216
DB2 installation
  db2_install script, 113-114
  non-root, 92
payload files manual,
  115-116
response files, 112-113
root, 90
sample db2iprune input
  file, 118
DB2 Setup Wizard
  DAS, configuring, 99
  DB2 copy names, 99
  DB2 instance, 99
features to install,
  selecting, 98
installation, starting, 101
launching, 97
response files, generating,
  97
settings, reviewing, 101
db2nodes.cfg columns, 292
db2profile script file, 191-194
directory locations
local database, 241
node, 242
  system database, 239
instances
  creating, 45, 201
  starting, 206-207
license certificates, installing,
  116
non-root installations
  fix packs, 128
  limitations, 102-103
  performing, 102
root, compared, 101
root-based features, enabling, 102
root installations
  non-root, compared, 101
user IDs/groups required,
  106-107
unknown table state, 569
unlocking big data, 8
updatable views, 393
update admin cfg command, 217
update db cfg command, 226
update db cfg for database_name
  command, 221
update dbm cfg command, 213, 216
UPDATE privilege, 442
UPDATE statement, 663-664
referential integrity
  implications, 357-358
SELECT from, 665
updates
  checking for, 180
  configuration parameters
    database level, 48
    instance level, 48
  connection profiles (Data
    Studio), 164
DAS configuration file, 217
DBM configuration file
  parameters, 212-213
tables, 663
upgrading DB2 editions, 14, 128
URs (uncommitted reads)
  concurrent data access, 504
  overview, 502
  testing, 547
USAGE privilege, 448
USE privilege, 440
USER register, 650
user-defined data types. See
  UDTs
user-defined functions. See
  UDFs
USEREXIT value
  (LOGARCHMETH1
  parameter), 599
users
  authentication. See
    authentication
    DAS, creating, 216
    group website,
IDs, required
  Linux/UNIX, 105-107
  Windows, 105
instance owners, 200
security labels, 460
tables, 341-344
  creating, 341-344
dropping, 344
USERSPACE1 table space, 52,
  312
utilities. See also tools
  binding, 265-266
data movement, 553-555
case study, 622-623
db2move, 577-579
EXPER, 557-559
file formats, 555-557
IMPORT, 559-562
ingest, 573-576
LOAD. See LOAD utility
loading data utilities
  comparison, 576
db2dart, 694-695
db2iprune
  fix packs, 120
  input file example,
    118-120
  Linux/UNIX sample, 118
  overview, 118
  pureScale support, 120
  Windows sample, 118
db2support, 692-693
Global Switcher, 122
IBM Data Studio. See Data
    Studio
maintenance, 580
REBIND, 584
REORG, 582-584
REORGCHECK, 582-584
RUNSTATS, 581-582
response file generator,
  111-112
trace, 693-694
| **VALUE COMPRESSION clause** (CREATE TABLE statement), 369 |
| **VARCHAR columns**, 335 |
| **variables** |
| environment, 46 |
| contents, viewing, 189 |
| creating/editing, 46 |
| DB2INSTANCE, 46, 189, 204 |
| DB2NODE, 70-71 |
| permanently setting, 190 |
| setting, 187-188, 191-194 |
| temporarily setting, 189 |
| registry, 46-47 |
| available, viewing, 47 |
| case-sensitivity, 81 |
| changes taking effect, 47 |
| current, viewing, 47 |
| DB2COMM, 271 |
| DB2EVALUNCOMMITTED, 527 |
| DB2_GRP_LOOKUP, 482-483 |
| DB2_SKIPDELETED, 528 |
| DB2_SKIPINSERTED, 528 |
| instance-level **versus** global-level, 47 |
| level indicators, 196 |
| listing, 230 |
| search order, 196-197 |
| setting, 196, 230 |
| starting/stopping instances for changes to take effect, 197 |
| unsetting, 230 |
| value, setting, 47 |
| session, 469 |
| variety (big data), 4 |
| velocity (big data), 4 |
| **VENDOR**: library value |
| (LOGARCHMETH1 parameter), 599 |
| veracity (big data), 5 |
| **VERIFY_GROUP_FOR_USER** function, 468 |
| verifying configurations |
| clients, 272-274 |
| database names, 274 |
| host name, pinging, 273 |
| node directory, 272-273 |
| port accessibility, testing, 273 |
| servers, 270-272 |
| configuration parameters, 272 |
| database existence, 271 |
| DB2COMM registry variable, 271 |
| **VERIFY_ROLE_FOR_USER** function, 468 |
| **VERIFY_TRUSTED_CONTEXT_ROLE_FOR_USER** function, 468 |
| versions. See editions viewing |
| configuration parameters |
| database level, 48 |
| instance level, 48 |
| DAS configuration file, 217 |
| database configuration files, 84, 217-221, 222-223 |
| databases, 300 |
| DBM configuration parameter current values, 213-214 |
| directories |
| DCS, 51, 244 |
| local database, 50, 242 |
| node, 50, 243 |
| system database, 49, 239-241 |
| environment variable contents, 189 |
| instances, 203, 230 |
| current, 81 |
| Data Studio, 214 |
| defined values, 149 |
| listing of, 81 |
| package isolation levels, 514-515 |
| partition groups, 68, 309 |
| products/features installed, 125-126 |
| registry variables, 230 |
| available, 47 |
| current, 47 |
| row subsets, selectively, 394 |
| table spaces, 324-325 |
| views, 390 |
| classifications, 391-392 |
| deletable, 392 |
| insertable, 393-394 |
| read-only, 394 |
| updatable, 393 |
| creating, 390 |
| defined, 287, 389 |
| definitions, storing, 391 |
| dropping, 391 |
| LBAC, 462 |
| multiple tables, creating, 390-391 |
| nested, 395 |
| overview, 389 |
| privileges, 441-444 |
| row subsets, selectively displaying, 394 |
| viewing, 390 |
| VM/VSE edition, 13 |
| volume (big data), 4 |

***W***

warehouses, simplification, 10-11

web console (Data Studio), 161, 174-175
Index

overview, 174
tasks, 175
websites
  Big Data University, 705
blogs, 699, 706
book, 43, 701
conferences, 705
Data Server Client, 701
database on the Cloud, 705
DB2
documentation, 704
Education, 703-704
Express, 16
Express-C, 701
technical support, 702
DB2 Essentials:
  Understanding DB2 in a
  Big Data World, 701
DB2 for Linux, UNIX, and
Windows
  DeveloperWorks forum, 699
  Technical support, 699
DCW (Database Conversion
  Workbench), 682
DeveloperWorks, 704
fix packs, downloading, 127
IBM
  Academic Initiative
  program, 26
  Business Partners, 706
certification program and
tutorials, 703
Data Magazine, 705
data server clients/driver
packages, 19
Data Studio, 161, 702
DB2 for LUW forum, 704
DB2 support website, 702
Information Management
  Best Practices, 704
  passport advantage, 701
Redbooks, 704
software, buying, 701
software support, 704
  support, 699
Information Center, 23, 179,
  699, 702
installation images, 94
Jaql, 12
language identifiers, 96
news group, 704
redbook Oracle to DB2
  Conversion Guide:
  Compatibility Made Easy,
  675
SQL, 646
SQL Reference for
Cross-Platform
Development, 27
  system requirements
  summary lists, 94
technical materials library,
  704
trial copies, 94
user groups, 705
WHERE clause (SELECT
statement), 653
Windows
  administration notification
  log example, 696
  backup file, 607
  CLP script file, 146
Command Window, 139-140
  invoking commands, 140
  starting, 140
  supported commands, 140
DAS
  creating, 217
  dropping, 217
DB2 installation overview, 90
DB2 Setup Wizard
copy names, 99
DAS, configuring, 99
features to install,
  selecting, 98
installation, starting, 101
  instances, 99
  launching, 96-97
  object security, 100-101
  response files, generating,
  97
  settings, reviewing, 101
db2nodes.cfg columns, 293
Default DB2 and Database
  Client Interface Selection
  Wizard, 177
directory locations
  local database, 241
  node, 242
  system database, 239
directory structure, 53-57
  configuration files,
  backing up, 57
database data directories,
  expanding, 55-53
  instance directory,
  expanding, 54
  instances, dropping, 56
  old databases from
  reinstallation, accessing,
  56
  instances
    creating, 200-201
    starting, 205-206
  license certificates, installing,
  117
  multiple DB2 versions/fix
  packs, 121-123
  copies, selecting, 121-122
  default copy selection,
  122-123
  response files
    creating with response
    file generator utility,
    111-112
DB2 installation,
  performing, 112
  example, 110-111
sample db2iprune input file, 118
security, 481
   DB2_GRP_LOOKUP
   registry variable, 482-483
   extended, 483-484
   global groups, 481
   local groups, 482
user IDs/groups required, 106
WITH CHECK OPTION clause
   (CREATE VIEW statement), 394
wizards
   Default DB2 and Database
   Client Interface Selection, 177
Setup, 92
   DAS, configuring, 99
   DB2 copy names, 99
   DB2 instance, 99
   ESE from scratch
   installation case study, 130-131
features to install,
   selecting, 98
installation, starting, 101
Linux/UNIX, launching, 97
object security, 100-101
response files, creating, 97, 109-111
settings, reviewing, 101
Windows, launching, 96-97
WLMADM authority, 431
Workload_type keyword
   (autoconfigure command), 227
workspace (Data Studio), 161
wrappers, 80
WSE (Workgroup Server
   Edition), 16
WSF file format, 557

X
   XML (Extensible Markup
   Language) data type, 337
   XPath, 34-35
   XQuery statements, 34-35
case-sensitivity, 81
   FLWOR expression, 35
   XPath, 34-35

Z
   Z/OS edition, 13