Implementing Siebel *e*Business Applications with DB2 UDB on AIX/NT

Install and customize Siebel *e*Business Applications 2000, V6.2.1

Understand Siebel database characteristics

Performance tuning and planning tips

Seungrahn Hahn
Klemen Kobilica
Pak Lee
Natalia Tejero Stroher

Redbooks

ibm.com/redbooks

SG24-6211-00

International Technical Support Organization

Implementing Siebel *e*Business Applications with DB2 UDB on AIX/NT

January 2001

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix F, “Special notices” on page 205.

First Edition (January 2001)

This edition applies to Siebel *e*Business Applications 2000, V6.2.1, and DB2 UDB Version 6.1 for use with AIX 4.3.3. or Windows NT Version 4, Servicepack 5.

This document created or updated on January 29, 2001.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HYJ Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Preface

This IBM Redbook will help you install and customize Siebel *e*Business Applications 2000 using DB2 UDB V6 on AIX and NT platforms. It is based on the installation experiences gained while installing Siebel *e*Business Applications 2000 at the IBM ITSO Center in Poughkeepsie, and on customer support experiences from the IBM Toronto Laboratory, IBM DB2 UDB Data Management Consulting Services, Siebel Engineering, and Siebel Technical Services.

The redbook provides the following:

- An overview of Siebel Systems, including architecture, major components, and various types of clients
- Sizing information based on existing customer system environments
- A set of DB2 parameter values for high performance
- An extensive description of Siebel database characteristics on DB2 UDB
- Information on Siebel's Enterprise Integration Manager (EIM) - the Siebel importing/exporting tool
- Data migration considerations from DB2 on one platform to DB2 on another platform
- An examination of AIX and DB2 tools to help you administer Siebel system environments
- Backup/recovery scenarios

This redbook will be especially useful for those who are installing Siebel *e*Business Applications 2000 using DB2 UDB on NT or AIX platforms. It can also help systems programmers and DB2 DBAs to install Siebel *e*Business Applications 2000 for the first time, and to manage it effectively. A knowledge of DB2 UDB on NT or AIX platforms and NT/AIX system administration is assumed.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization Poughkeepsie Center.

Seungrahn Hahn is a Business Intelligence specialist at the International Technical Support Organization, Poughkeepsie Center. She writes and teaches IBM classes worldwide on all areas of BI solutions on the S/390 platform. Before joining the ITSO in 1999, she worked as a Certified IT specialist at IBM Korea in physical database design, performance tuning, the data sharing implementation with DB2 for OS/390, and BI application solutions.

Klemen Kobilica is a Data Management specialist in Ljubljana, Slovenia. He is responsible for technical support in Central and Eastern Europe, the Middle East, and Africa. He has three years of experience in data management, mainly in the Business Intelligence area. His areas of expertise include DB2, DataJoiner, warehousing and OLAP. He is an IBM Certified Solution Specialist for DB2 UDB Database Administration. He holds a degree in physics and is finishing his post-graduate study of Information Technology.

Pak Lee is a Data Management support specialist in Sydney, Australia. He has a total of eight years of experience in database administration in DB2 on NT, AIX, MVS and HP. He currently provides technical product support for DB2 UDB, the IBM Business Intelligence suite of products, and the IBM Content Management suite of products on workstation platforms. He is an IBM Certified Solution Specialist for DB2 UDB Database Administration, and Business Intelligence. He is also an IBM Certified Specialist for MQSeries.

Natalia Tejero Ströher is an IT Specialist and Siebel Certified Consultant with IBM Global Services in Madrid, Spain. Natalia has an Industrial Engineering degree, and majored in Industrial Electronics. She has participated in one of the biggest Siebel Projects in Spain.

Thanks to the following people for their invaluable contributions to this project:

Terry Barthel
Alison Chandler
Al Schwab
International Technical Support Organization, Poughkeepsie Center

Laurence Leong
IBM Data Management

Sudha Dev
Palmer Brisendine
Dennis Murphy
IBM Siebel International Competency Center

Kenneth Chen
Daniel Deaconu
Sandy Gregus

Serge Limoges
Karen Nolk
Berni Schiefer
IBM DB2 UDB Toronto Laboratory

Bob Harbus
IBM DB2 UDB Data Management Consulting Services - Toronto

Mike Alcorn
Susanne Englert
Tuong Truong
IBM Silicon Valley Lab

Tethuya Shirai
IBM International Technical Support Organization, San Jose Center

Byron Banks
Francisco Casas
Mark Farrier
Bronya Feldman
Eric Aronson
Larisa Medrish
Hasan Qureshi
Peter Samson
Jayesh Thakrar
Siebel Systems, Inc.

Comments welcome

Your comments are important to us!

We want our Redbooks to be as helpful as possible. Please send us your comments about this or other Redbooks in one of the following ways:

- Fax the evaluation form found in “IBM Redbooks review” on page 219 to the fax number shown on the form.
- Use the online evaluation form found at **ibm.com**/redbooks
- Send your comments in an Internet note to redbook@us.ibm.com

Chapter 1. Siebel system overview

Siebel Systems is an industry leader in Customer Relationship Management (CRM) solutions. Siebel provides an integrated suite of e-business application software enabling multichannel sales, marketing, and customer service systems to be deployed over the Web, call centers, field, reseller channels, retail, and dealer networks.

Siebel *e*Business Applications 2000 is an integrated suite of products that help manage customer relationships. It provides

- A set of horizontal solutions: This includes basic functions for all businesses or business partners such as Siebel Call Center, Siebel Sales, Siebel eChannels, Siebel Marketing, etc.
- A set of vertical solutions: This includes specific and unique solutions for particular industries such as Siebel eFinance, eInsurance or ePharma, etc.

However, in this redbook, we describe *horizontal* solutions only. With vertical solutions, you may need different considerations and different processes, and may get different results.

This chapter introduces Siebel architecture, Siebel's major components, and the different types of client supported by a Siebel Application.

1.1 Siebel product architecture

Siebel products are logically built on a multi-layered architecture as shown in Figure 1 on page 2. This multi-layered architecture provides the platform for common services to support the various Siebel applications. These services include:

- User Interface
- Object Manager
- Data Manager
- Data Exchange Layer

1.1.1 User Interface

User Interface provides the applets, views, charts and reports. A large variety of clients are supported, such as traditional connected clients, remote clients, thin clients over an intranet, Java thin clients or non-Windows-based operating systems, and HTML clients over the Internet.

1.1.2 Object Manager

Object Manager manages the enterprise's business rules, in the form of Siebel Business Objects, as highly configurable software representations of business concepts such as accounts, contacts, opportunities, and service requests. All Siebel Thin Clients connect to a Siebel Object Manager to access the application's business logic. It is hosted in Siebel Server environment, and delivers the following:

- Multi-user support
- Dynamic load balancing across multiple servers
- High resilience and high availability
- Full support for Siebel Business Objects
- Common administration framework

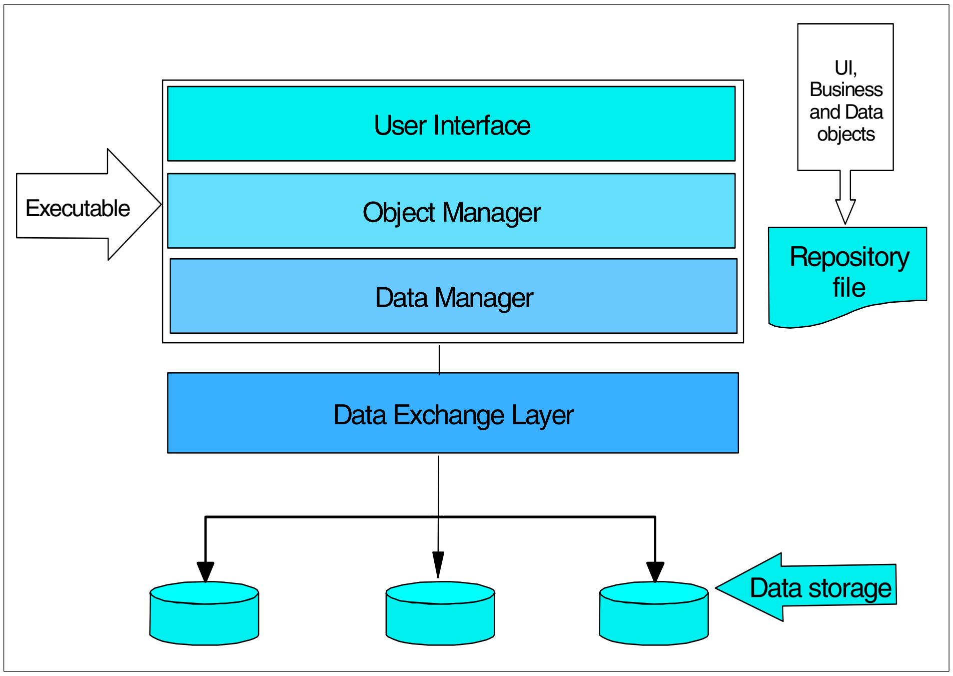


Figure 1. Siebel multi-layer architecture

1.1.3 Data Manager

Data Manager maintains a view of the data and allows the Object Manager to function independently of RDBMS. It also generates Structured Query Language (SQL) in real time to access the business data. It operates on object definitions in the repository file (.srf) that define the database schema.

Data storage services provide the data storage for the Siebel data model, which is a very comprehensive data model that is the basis of the Siebel applications.

The model supports the core Siebel products including call center, sales, services, and marketing, as well as the industry vertical products, such as eFinance, eInsurance, eCommunications, and eHealthcare, to name a few.

1.1.4 Data Exchange Layer

The Data Exchange Layer, normally implemented as a separate DLL, handles the interaction with a specific target database as follows:

- It understands the details of how the data is physically stored
- It delivers data from data sources upward to the Data Manager layer in a consistent and efficient manner
- It uses the efficient application programming interface (API) for the database
- It tunes the SQL statements for the database

1.2 Main Components of the Siebel's Architecture

Figure 2 on page 4 shows the major software components that make up the Siebel Server architecture that follows the multi-layered architecture. Multi-layered architecture allows managers to reside on different machines:

- User interface, Object manager, and Data manager all reside on the connected and mobile clients
- Object manager and Data manager reside on the Siebel Server for thin clients
- User interface resides on the Server for HTML clients

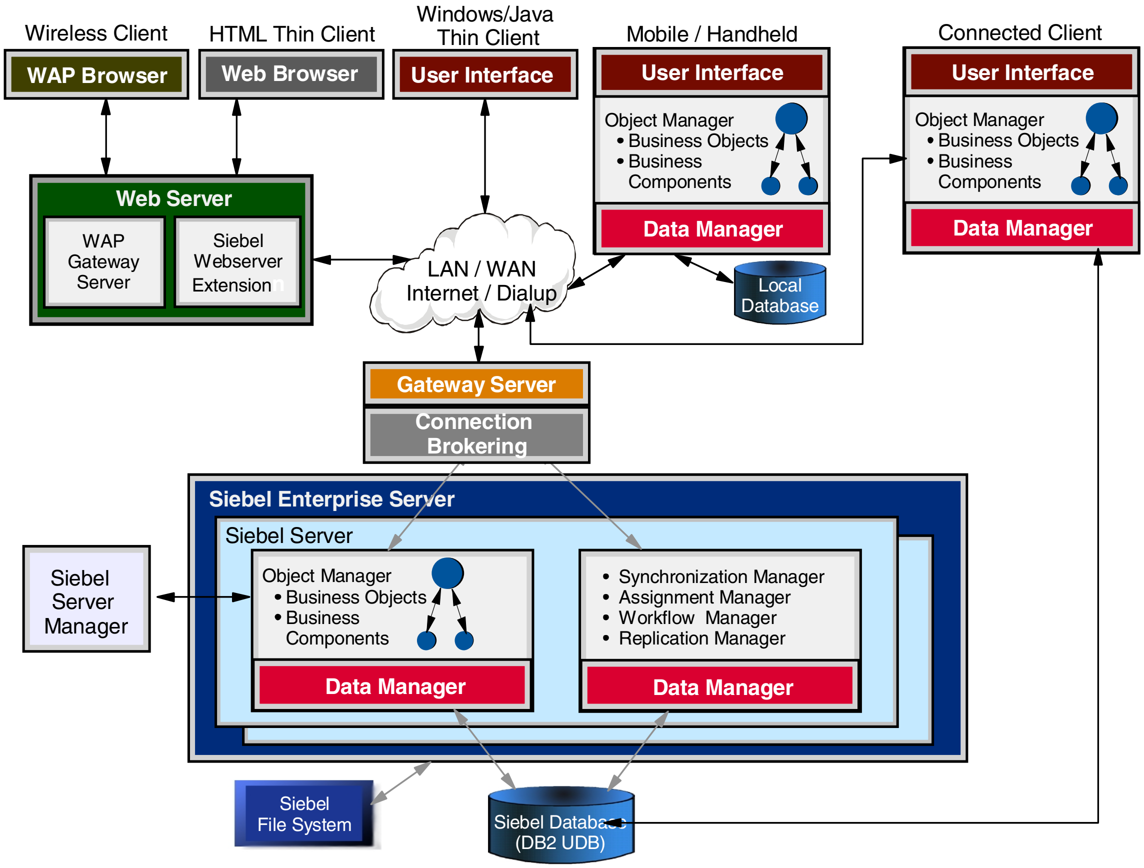


Figure 2. Siebel's architecture

When you deploy this Siebel architecture, you can consider:

- The Gateway Server, Siebel Server, Database, and Siebel File System can be implemented on one machine or spread across multiple machines (we strongly recommend a distributed network for better performance).
- The Siebel Servers should have a high speed LAN connection to the Siebel database.

1.2.1 Siebel database

In a Siebel Enterprise Server environment, there can be more than one database. The default is one, which is referred to as the *primary* or HQ database node. With Siebel Replication manager, you can have one or more child databases/Enterprise of the primary database/Enterprise. In addition, if you have mobile clients, you will have additional remote databases in a Siebel Enterprise. Unless defined as read-only databases, these mobile clients can create and update data locally that will be ultimately propagated up to the primary database when each mobile client synchronizes with the Siebel server.

The Siebel database stores predefined Siebel data schema (Table Objects, Index Objects, Constraints, and Triggers). It also holds Repository Objects (Business Objects and Components, view definitions and visibility rules) and the customer's data.

Siebel connected clients and Siebel Server components, including those that operate in conjunction with the Siebel Thin Client, connect directly to the Siebel database and make changes in real time. It supports Siebel Server access through ODBC and supports connected client access using a native driver.

There are three types of data in a Siebel database:

Seed data The initial data populated by the install.ksh script.

Repository data Predefined object definitions that specify the application look, behavior, and data. Populated by the imprep.ksh script.

User data Supplied by you, it includes transaction data, administrative data, interface/temporary data in the EIM tables, etc.

Siebel Mobile Clients download a subset of the server's data to use locally, and periodically synchronize with the server database through the Siebel Server to update both the local and the server database.

1.2.2 Siebel File System

The Siebel File System is a network-accessible directory located normally on an application server. It stores all the physical files created by Siebel applications, such as files created by third-party text editors, and other data that is not stored in the database.

All files in the Siebel File System are compressed and stored using a specialized naming convention. For this reason, users should never access files directly from the File System, but instead, only access through the Siebel client, which automates the compression and naming tasks. All files in the File System are stored in a single directory, and are distinguished by filename prefix.

Siebel Connected Clients read and write files directly to and from the Enterprise's File System. Siebel Mobile Clients have a local File System, which they synchronize with the server-based File System periodically. Siebel Thin Clients access the File System through the Siebel Server.

1.2.3 Siebel Enterprise Server

The Siebel Enterprise Server is a logical grouping of Siebel Servers that share a common table owner or a *single* primary database, point to a common Gateway Server, and can be administered as a group using Server Manager. The connection to the Gateway Server is via TCP/IP.

The Siebel Enterprise Server scales effectively by deploying multiple Siebel Servers in the Enterprise Server, providing a high degree of scalability in the middle tier of Siebel Enterprise Applications.

1.2.4 Siebel Server

The Siebel Server runs one or multiple Siebel Server Programs. It handles the incoming processing requests and monitors the state of all processes on the server. Siebel Server programs perform a specific function or job such as:

- Importing and exporting data
- Configuring the database to monitor for user-defined conditions
- Executing workflow and process automation
- Processing to support thin clients
- Processing to support mobile clients for data synchronization and replication
- Enforcing business rules

The Siebel Server itself is an NT Service (under Windows NT) or a daemon, (a background shell process) under UNIX. The Siebel Server supports both multi-process and multi-threaded components and can operate components in batch, service, and interactive modes.

1.2.5 Siebel Server Manager

The Siebel Server Manager is a utility that allows common control, administration and monitoring across disparate programs for the Siebel Servers and the Siebel Enterprise Server. Siebel Server Manager must be able to connect to both the Siebel Servers and Gateway. There are two interfaces to use the Server Manager: one way is by accessing via the Server Administration views from Screens in the Siebel application, and the other way is by using the command-line interface or `srvrmgr` program.

You can use the Server Manager to perform the following tasks:

- Start, stop, pause, and resume Siebel Servers, components, and tasks
- Monitor status and collect statistics for all multiple tasks, components, and Siebel Servers within an Enterprise
- Configure the Siebel Enterprise Server, individual Siebel Servers within the Enterprise, individual components, and tasks

1.2.6 Siebel Gateway Server

The Siebel Gateway Server is a logical entity that serves as a single entry point for accessing Siebel Servers. It provides enhanced scalability, load balancing and high availability across the Enterprise Server. It consists of the Name Server and the Connection Brokering component (this component is optional).

1.2.6.1 Name Server

The Name Server is responsible for keeping track of all Siebel Servers parameters. Their availability and connectivity information is stored in the Name Server’s volatile store. Enterprise components query the Name Server for this kind of information. In a Windows NT environment, this is run as a NT service controlled through the NT Service Control Panel. In a UNIX environment, the Name Server runs as a daemon process.

1.2.6.2 Connection Brokering

Connection Brokering directs client connection requests to the least-busy Siebel Server operating the desired component. It is an optional service of the Gateway Server that uses the Resonate Central Dispatch product (third-party software) to perform load balancing.

1.2.7 Siebel Clients

The Client types supported by Siebel are the following:

- Connected Clients
- Mobile Clients
- HTML Clients
- Wireless Clients
- Thin Clients for Windows
- Java Thin Clients

1.2.7.1 Connected Clients

Connected (or Dedicated) Clients are connected directly to a database server for all data access via a LAN or WAN connection. They don't store any data locally. They also access directly to the Siebel File System using a protocol supported by the Siebel File system.

Connected Clients have full functionality. The system's response to their requests is dynamic and immediate. If these clients need to access the interactive components (for example, the Workflow Manager and the Assignment Manager), there must be a connection via LAN, WAN, dial-up or the Internet to the Siebel Enterprise Server. For more information about these components of Siebel Server, refer to the Siebel Bookshelf.

1.2.7.2 Mobile Clients

Mobile Clients are designed for local data access, and therefore they have their own local database that contains the Siebel application tables that store user data, as well as a local Siebel File System.

Mobile Clients have the same functionality and system response as Connected Clients. The only difference is that there is no real time connection necessary. Mobile Clients interact with all other components through the Gateway Server. Through synchronization, the modifications from the two databases (local and Server database) are exchanged.

1.2.7.3 HTML Thin Clients

Siebel HTML Thin Clients are a set of components used for customizing and implementing the .COM Web applications. They are part of a large system that also has a Siebel Server, a Siebel database and a Web Server.

Following are the main elements of an HTML Thin Client system, as shown in Figure 3 on page 9:

- Siebel Web Application
- Siebel Web Server Extension
- Siebel Web Engine
- Siebel Web Templates

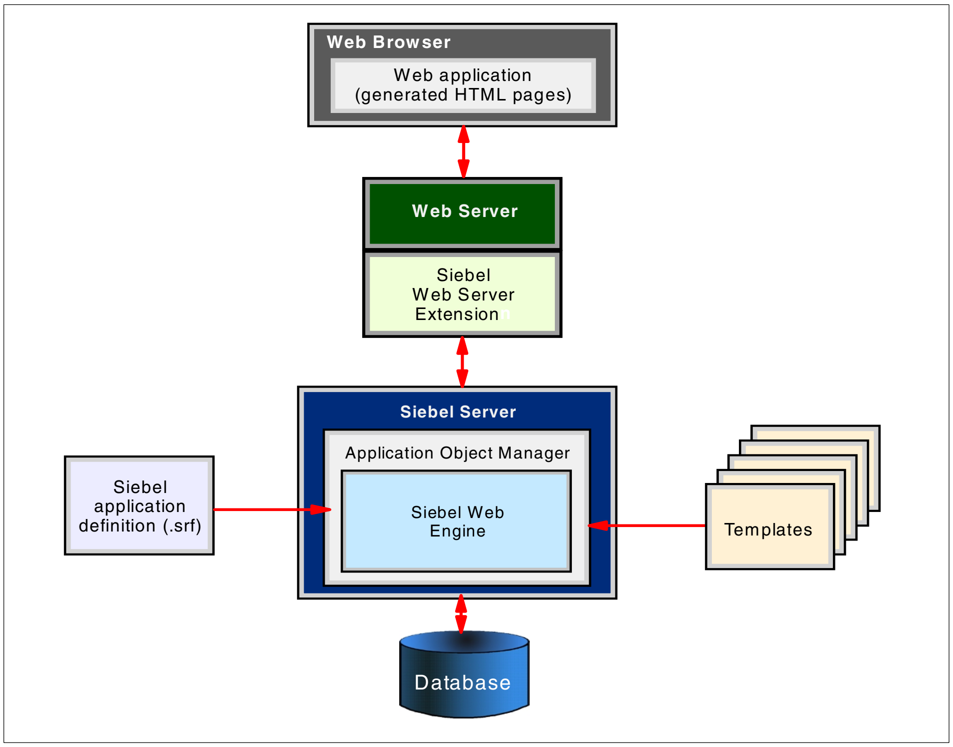

Web Browser
Web application (generated HTML pages)
Web Server
Siebel Web Server Extension
Siebel Server
Application Object Manager
Siebel Web Engine
Siebel application definition (.srf)
Templates
Database

Figure 3. Main components for HTML Thin Clients

What users can do using HTML Thin Clients:

- View dynamically generated Web pages that include data from a Siebel database
- Add to, delete, or modify data in a Siebel database
- Perform queries, sorts, and other operations using any of the other thin clients or a dedicated client.

For more information about these elements, refer to the *Configuring Siebel .COM Applications* in *Siebel Server Installation Guide*, 10PA1-IN00-06200.

1.2.7.4 Siebel Wireless Clients

Siebel Wireless Clients is essentially a thin client enabled on a wireless device. It uses a Wireless Application Protocol (WAP)-based user interface to deploy Siebel applications to wireless devices. Siebel Wireless enables field sales and service users of Siebel Applications to wirelessly access Siebel data directly.

1.2.7.5 Siebel Thin Clients for Windows and Java Thin Clients

With these types of clients, you can deploy the Siebel Enterprise Application over the intranet. They access the application dynamically over the network from any connected machine. These clients access the Gateway Server to determine the Siebel Server, and access Siebel database through the Object Manager Components in Siebel Server. Since Siebel Servers execute all the business logic, Thin Clients have more modest hardware requirements. These clients do not store any data locally. They have a completely Siebel look and feel. The functionality, the user's interaction, and performance are excellent.

- Siebel Thin Client for Windows: This client enables enterprises to deploy the user interface layer of Siebel *e*Business Applications in a standard Web browser or as a standalone Windows executable (even if the browser interface is used, it does not connect through a Web Server.)
- Siebel Java Thin Client: This client enables enterprises to deploy the user interface layer of Siebel *e*Business Applications as a standalone Java application.

1.2.8 Which type of client is the appropriate one

Table 1 lists a few points to take into consideration before deciding which type of client is the most appropriate for your business.

Table 1. Type of clients vs. business requirements

Type of client to use	Business requirements
Connected Client	Call Centers where there is no mobile requirement and immediate system response time is required
	Full set of Siebel's product functionality and features is needed
	Client-side Visual Basic scripting is needed

Type of client to use	Business requirements
Mobile Client	A field sales or service representative needs to operate Siebel applications locally without any real-time connection to any Siebel server.
	Periodic accesses to the Siebel server through a dial-up, WAN, or LAN connection to synchronize data changes with the Database Server and File System, are acceptable.
HTML Thin Client	Give the Siebel Application the look and feel of your company's Web site.
	You need to serve remote users, occasional users, channel partners or direct customers
Thin Client for Windows or Java	Deployment of Siebel Enterprise Applications over the Internet or intranet
	Less capable client machines are available
	Faster deployment of Siebel Enterprise applications is needed
	Real-time integration from the middle-tier servers is needed
Wireless Client	A field sales and service representative needs to operate Siebel applications to wirelessly access Siebel data directly from the corporate database.

1.3 Benefits of deploying Siebel *e*Business on DB2 UDB

As the foundation for e-business, DB2 UDB is the industry's first multimedia, Web-ready relational database management system (DBMS), powerful enough to meet the demands of large corporations and flexible enough to server medium-sized and small businesses.

DB2's ability to handle scalability, data integrity, manageability and security requirements makes it the best choice for Siebel Applications. The strengths DB2 brings to the Siebel applications environment include the following:

Scalability and reliability

The data in a Siebel system quickly becomes a corporate asset. To protect this asset, it is fundamental that the database be best of class in reliability and recoverability. It must meet the demands of a large number of concurrent users with response times that meet the dynamic needs found in an e-business system.

DB2 provides the cross-platform scalability required by most e-businesses today. Implemented in Windows NT and UNIX, DB2 for Siebel applications is scalable from small businesses to global enterprises, from a few hundred to tens of thousands concurrent users, and from uniprocessor servers to large symmetric multiprocessor (SMP) configurations.

Performance

With more than 1.3 million DB2 UDB server licenses and six million users worldwide, IBM DB2 Universal Database is the database market leader, thanks to its capability to manage these *mission-critical* tasks so well. Approximately 400 Web integrators work with DB2 Universal Database and a full 70 percent of the world's corporate data resides on DB2 – a testament to its outstanding performance.

With DB2, you can equip your company with a database for supporting customer-centered e-business initiatives which can in turn increase sales, boost productivity and heighten efficiency across your enterprise. Specific DB2 features that help to fine-tune Siebel application performance include multiple buffer pools and optimization levels, and earlyclose and lightoutput sqlda functions.

Availability

DB2's high availability features, which ensure maximum system availability and minimum downtime, are field-proven. It is a very stable database which provides online maintenance without service interruption.

Database and system management aids

DB2 is easy to manage, and its product quality helps lower maintenance. To help database administrators manage their database environments, DB2 offers an integrated set of tools and functions, including flexible security mechanisms, online reorg keeping the system up and running, an extensive set of logging and recovery utilities, trace facilities for tuning, and functions and tools to monitor and tune the DB2 system. For details about DB2 aids, refer to Appendix C, "DB2 utilities" on page 183.

Total flexibility

The DB2 family of databases provides great flexibility for implementing Siebel applications. With DB2, customers can maintain a standardized information technology architecture across UNIX and Windows NT servers, as well as

S/390 enterprise servers. Parallel query functionality is available independent of hardware platform and architecture.

Chapter 2. Planning for implementing Siebel solutions

In this chapter we review some of the factors that affect the planning and sizing of a Siebel DB2 UDB implementation. Skills, resources, and commitment have a major impact on the successful implementation of a Siebel system, as does collecting and documenting user requirements and user expectations.

If you look at a typical Siebel implementation, and assuming that you have already defined the requirements of Siebel components and applications, the focus will be on the following:

- Number and type of clients
- Hardware platform
- Disk space requirements
- LAN configurations
- Database environment and configurations
- Security requirements
- Backup and recovery requirements
- Rollout strategies
- Conversion/migration from the legacy system

However, the information you collect about these requirements and expectations only gives you a baseline to begin. The next question is how to *configure* the Siebel – DB2 UDB system so that it will be able to support the production deployment.

As with all planning and sizing efforts, there is no hard and fast rule; in most situations, the estimates are based on past experiences. But new users are always faced with the question of what to do with given parameters – should they be set high or low? To answer this question, it's useful to have a checklist or matrix to refer to in order to see what others have done. In the following section we provide such a matrix.

2.1 Existing customer configurations

In Table 2 on page 16, we list some of the configurations of existing Siebel DB2 UDB customers. This real-life information was collected from the field, and gives you an idea of what others are doing. Of course, some of the information might not be appropriate to you, as the requirements and expectations of these customers may be different from yours.

Note

This information should only be used as a guide. It simply shows what others are doing with regard to planning and sizing their Siebel environments, and gives a starting point for your Siebel DB2 UDB implementation.

Table 2. Example Siebel DB2 UDB customer configurations

Configuration	Customer 1	Customer 2
Siebel Server		
Number of Siebel Servers	7	2
Hardware platform	Intel-based PC	Intel-based PC
OS	Windows NT v4, SP6	Windows NT v4,SP5
Memory	2 GB	512 MB
Disks	50 GB	N/A
Database server		
Hardware platform	RS/6000 S80	RS/6000 S80
CPUs/machine	18 processors	12 processors
OS	AIX 4.3.3	AIX 4.3.3
Memory	16 GB	12 GB
Paging Space	16 GB	24 GB
Disk Space	1.5 TB	1.5 TB (80 18.2 GB SSA disks)
Database size	60 GB (120 GB in within a year)	200 GB
RAID configuration	RAID-5	RAID-1
Client		
Hardware platform	Various hardware platforms	Various types of PC with 300 Mhz or more

Configuration		Customer 1	Customer 2
	OS	Windows NT 4.0, SP5 and Windows 2000	Windows NT
	Memory	128 - 256 MB	128 MB ram
LAN configuration			
	Siebel Servers and Database server	FDDI	ATM
	Siebel Connected Clients and Siebel Servers	Ethernet with a handful of FDDI	ATM and WAN with 12 Mbits/sec of bandwidth
	Siebel Connected Clients and Database server	Ethernet with a handful of FDDI	ATM and WAN with 12 Mbits/sec of bandwidth
User connections			
	Maximum users	8100	2000
	Maximum concurrent users	2300	2000
	Number of Connected Clients	6600	2000
	Number of Mobile Clients	1400	0
	Number of HTML clients	100	0

Note

For details regarding the bufferpools, tablespaces, database manager configuration, database configuration, tables and indexes of these customers, see Appendix A, “Existing customer configurations” on page 159.

2.2 Planning and sizing your Siebel DB2 UDB environment

As previously mentioned, several factors will impact the planning and sizing of your Siebel DB2 UDB environment. In this section, we discuss several factors that will affect the planning.

2.2.1 Number and type of clients

The selection of client types used with your environment is dependent on your processing requirements. Some possible considerations are:

- Are there any mobile requirements?
- Does the client require a full set of Siebel's product functionality and features?
- Is there any client-side Visual Basic requirement?
- Is there any Internet or intranet requirement?
- What is the cost of deployment?

2.2.2 Hardware platforms

The type and number of servers and the CPU and memory resources required are directly dependent on the Siebel applications deployed and the type and number of clients to be supported.

Enabling of Compression for Thin Clients will reduce the network traffic between the server and the client, but this has a major impact on the hardware capacity requirements. Siebel estimates a 33% reduction in capacity on the Siebel Server (in terms of the number of concurrent Thin Clients) when compression is enabled for Thin Client activity.

Therefore, you should consider having a separate physical machine for Central Dispatch from Resonate, in order to distribute the load across multiple Siebel Servers.

You might also want to consider having dedicated Siebel Servers for Thin Clients for Java and Windows.

The supported hardware platforms for Siebel implementation are well-documented in the *Siebel System Requirements and Supported Platforms*, Siebel 2000, V6. We recommend that you always refer to the latest version of the Siebel Bookshelf for the Siebel technical publications that are relevant to the release. You can access support from:

```
http://www.siebel.com
```

2.2.3 Disk space requirements

Disk space requirements can be broken into the following:

- Disk space for Base tables
- Disk space for EIM tables
- Disk space for indexes
- Disk space for the Siebel File System
- Disk space for DB2 logs
- Disk space for DB2 temporary tablespaces
- Disk space for the EIM processing staging area
- Disk space for the backup image

The ideal solution is to ensure that all these reside on different disks and disk controllers in order to minimize the possibility of any I/O contention. You also have to have enough disk arms to handle concurrent random I/O. One rule of thumb is to have 6 to 10 disks per processor (for example, in the case of 12 CPUs, use 120 disks).

You might also consider disk mirroring, or using one of the RAID configurations, to reduce the possibility a disk failure.

Review Chapter 9, “Performance tuning tips” on page 143 for a detailed discussion on DB2 database design for optimal performance.

2.2.4 LAN configurations

Depending on the number and types of clients to be implemented, the network bandwidth between Siebel clients and the Siebel Servers and Web servers must be adequate to provide acceptable performance.

There will also be high network utilization between the Siebel Servers and the database server.

2.2.5 Database environment and configurations

The performance of applications running on DB2 UDB is dependent on the resources allocated to DB2 UDB. The areas of impact are:

- DB2 Bufferpools

  Bufferpools are cache areas for data and indexes. Data access and manipulation involve the use of the bufferpool. If you have separated tables into various table spaces for performance reasons, then you might want to consider having different bufferpools for different tablespaces.

- DB2 Database Manager configurations

  These configurations control the general behavior of the DB2 instance. Some of these configuration parameters affect the amount of system resources that will be allocated. Therefore, it is important to review and understand their impact on system resources.

- DB2 Database configurations

  These configurations control the behavior of the DB2 Database engine. As with the Database Manager configuration, we have provided you with a starting point, but you should consider doing a regular performance review.

Review 6.4, “DB2 parameters for a Siebel environment” on page 98 for a detailed discussion on DB2 Database and Database Manager configuration parameter tuning for the Siebel DB2 UDB environment.

2.2.6 Security requirements

Siebel applications enable you to control user access to information from within the application by defining responsibilities. Each responsibility provides access to a particular set of views, including the ability to see specific data and take specific actions.

The External Security Adapter is an interface that enables you to use an external system to authenticate users, rather than taking the traditional approach of using the database. With the External Security Adapter, you maintain user information in a Lightweight Data Access Protocol (LDAP) repository. After a user logs on to the Siebel application, and once the user ID is verified by LDAP, a credential is provided by LDAP to the user for accessing the Siebel applications.

You have the option of using either the traditional approach or LDAP to manage your user access.

2.2.7 Backup and recovery requirements

The type of backup (offline or online) to be taken for the database is dependent on your recovery requirements and the batch windows available to perform the backup.

For example, with a very large database (100 GB and above) and a 24 x 7 environment, you might not be able to afford a daily offline backup. Instead, you might consider doing a weekly offline backup with a daily online backup, or even a monthly offline backup with a daily online backup.

You may also want to look at the time required to recover from a failure. The recovery time from a weekly offline backup is definitely faster than the recovery time from a monthly offline backup, because DB2 has to process all the DB2 logs.

You should also consider the following regarding your backup/recovery strategy:

1. Should backup be on disk or on tape?
2. Should you use Tivoli Storage Manager for backup?
3. What is your expected throughput for backup?
4. How many tape drives should you have?
5. What about tablespace-level backup?

2.2.8 Rollout strategy

Siebel provides an e-Roadmap implementation methodology that can help you build an appropriate rollout strategy; we recommend that you contact Siebel if you need to develop such a strategy.

2.2.9 Conversion/migration from a legacy system

Converting or migrating from a legacy system is not an easy task; the method used depends on each customer’s data, the existing computer system environment, and so on, so there is no standard way to do these tasks. Contact Siebel if you have legacy system conversion/migration requirements.

2.3 High availability strategy

We recommend that you use a standard operating system failover software and hardware configuration for DB2 and Siebel.

HACMP and Microsoft Cluster Server (MSCS) can be used to provide a DB2 high availability solution on AIX and Windows NT, respectively.

For a complete understanding of HACMP concepts, installation, and configuration, refer to the following books:

- *HACMP for AIX, Version 4.2: Concepts and Facilities,* SC23-1938
- *HACMP for AIX, Version 4.2: Installation Guide,* SC23-1940

- *HACMP for AIX, Version 4.2: Planning Guide,* SC23-1939

2.3.1 DB2 high availability

Two types of configuration are available in a DB2 system:

- Hot Standby (asymmetric mode)

  In this configuration, one server is being actively used to run DB2, and the second is in standby mode, ready to take over if there is an operating system or hardware failure involving the first server.

- Mutual Takeover (symmetric mode)

  In this configuration, both servers participate in the database system (that is, each machine has at least one database server running on it). If one of the servers in the HACMP or MSCS cluster fails, the database server on the failing machine will be started to run on the other machine.

  In a mutual takeover configuration, a database server on one machine can fail independently of the database server on another machine. Any database server can be active on any machine at any given point in time.

Each of these configurations can be used to failover one or more partitions of a partitioned database.

For information about DB2 high availability on different platforms, refer to *DB2 Administration Guide: Design and Implementation V6*, Part 4 “Ensuring the High Availability of Your System”.

2.3.2 Siebel high availability

There are several different scenarios you can use to implement Siebel high availability. One of them is to divide clients by type, because clients access the Siebel database in different ways:

- Connected Clients

  You need to implement only the DB2 failover solution, because a Connected Client connects directly to the database server. Unless a Connected Client is performing actions such as administrative tasks, it does not connect to the Siebel server.

- Mobile Clients

  For a Mobile Client, connection is established through the Siebel Gateway Server and the Siebel Server. You need to implement a failover solution for both of them. If one of the servers is down, Mobile Clients can still work, but synchronization will fail. Users can synchronize later, when the working environment is established again.

If the Siebel Server cannot be restored, then the Mobile Client will need to be moved to a different Siebel Server supporting Mobile Clients, or the failed Siebel Server will have to be rebuilt. In both situations, a new Database Extract (a DBExtract task on the Siebel Server) will need to be created for each Mobile Client that was previously defined on the failed Siebel Server.

- Thin Clients

  For Thin Clients, connection is established through the Web server, the Siebel Gateway server, the Siebel Server, and Resonate Central Dispatch (if you are using it). You need to implement a failover solution for all these components. If one of the servers goes down, any uncommitted transactions (i.e., those in progress) are lost. Also, the Thin Client would need to reconnect and reauthenticate.

2.4 Summary

As you can see, planning and sizing is a non-trivial task. It requires extensive knowledge of both the Siebel and DB2 UDB systems. Therefore, we recommend that you get help with this process from the professionals at IBM Techline or Siebel Expert Services.

IBM Techline

IBM Techline provides hardware sizing support for CRM, ERP, and SCM software (for example, Siebel, SAP, PeopleSoft, etc.). By completing the questionnaire related to a future Siebel environment (such as the number of different types of clients, transaction requirements, your data profile, etc.), you can get a sizing result. You can contact IBM Techline at:

```
ibmerp@us.ibm.com
```

Siebel Expert Services

Siebel Expert Services provides consultants who ensure that each customer project takes full advantage of Siebel technology and products. The Siebel Expert Services team conducts configuration and architectural reviews throughout the project, and is prepared to resolve challenging technical and architectural issues related to a Siebel project. You can contact Siebel Global Services as the Siebel Expert Services at:

```
http://www.siebel.com/global/index.html
```

Chapter 3. Installing Siebel on NT

This chapter outlines considerations in installing the Siebel system using DB2 UDB as the RDBMS on NT platforms. We concentrate on a 3-tier installation, as this is the most common scenario.This chapter does not replace the Siebel and DB2 UDB documentation, but rather clarifies the steps and actions required.

Checklists

- Review *DB2 UDB V6 Release notes, Siebel 2000, V6.2.1*.
- Review *Siebel System Requirements and Supported Platforms, Siebel 2000, V6.2.1*, 10PA1-RP00-06210, to ensure you have the necessary hardware and software required for the environment.
- Review *Siebel Client Installation and Administration Guide, Siebel 2000, V6.2,* 10PA1-CA00-06200 for client installation information.
- Review *Siebel Server Installation Guide, Siebel 2000, V6.2*, 10PA1-IN00-06200 for Siebel Server component installation.
- Review *Siebel Server Administration Guide, Siebel 2000, V6.0, rev. A,* for Siebel Server administration.
- *Review DB2 Quick Beginning for Windows NT V6,* GC09-2835, for DB2 client and server installation.
- Review *Siebel Release Notes*, Siebel 2000, V6.2.
- Complete the Development Planning Worksheet A-1 of *Siebel Server Installation Guide Siebel 2000, V6.2,* 10PA1-IN00-06200.

Installation process summary

The general installation process consists of several steps:

1. Preparing for the installation.
2. Installing the Siebel File System.
3. Installing the Siebel Gateway Server.
   a. Installing the Resonate Central Dispatch product (optional).
   b. Installing the Gateway Server itself. (This includes the Name Server.)
4. Creating an Enterprise Server and installing one or more Siebel Servers.
5. Configuring and installing the Siebel Database Server with DB2 UDB.
6. Installing the CORBA Object Manager (optional).
7. Installing the Siebel Thin Client Applications (optional).

8. Installing the Siebel Report Server (optional).
9. Installing the Siebel Dedicated Client (described in the *Siebel Client Installation and Administration Guide*).

We did not install the optional components, so those installation steps are not described here. If you will use some of the optional components, refer to *Siebel Server Installation Guide*, Siebel 2000, V6.2, 10PA1-IN00-06200.

This chapter is divided into three parts:

1. Preparing for the installation
2. Performing the installation
    a. Installing the Siebel Database server
    b. Installing the Siebel Application Server
        1. Installing the Siebel Gateway Server component
        2. Installing the Siebel Server component
        3. Installing the Siebel Database component
3. Post-installation considerations

3.1 Pre-installation requirement

It is very important that you review the checklists before starting any installation.

Note

The first time you install any Siebel component on a Windows NT machine, the setup script will force a reboot of your machine before it starts the actual installation.

Your machines must be able to ping each other by hostname.

3.1.1 Our Siebel DB2 environment

Figure 4 on page 27 illustrates our test environment. This figure may be helpful for you to compare different names of servers, Siebel components, users, etc. with yours and clarify the architecture and installation procedure.

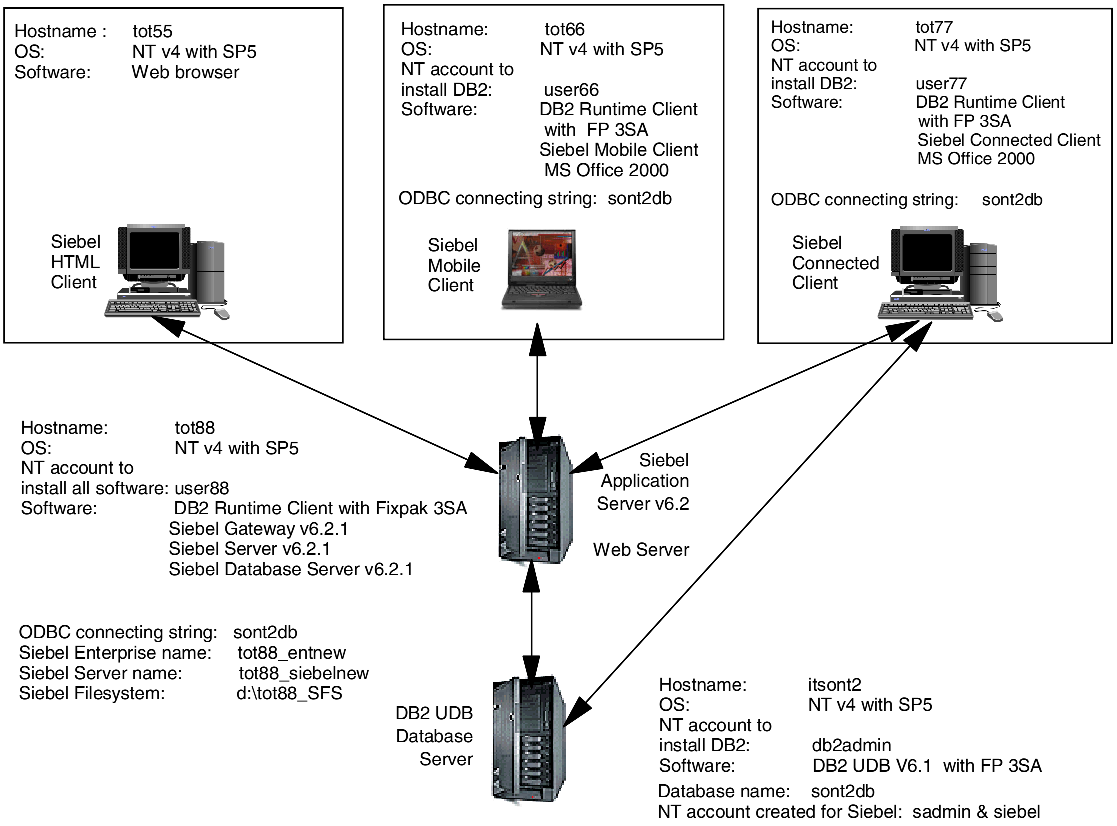

Hostname : tot55
OS: NT v4 with SP5
Software: Web browser
Siebel HTML Client
Hostname: tot66
OS: NT v4 with SP5
NT account to install DB2: user66
Software: DB2 Runtime Client with FP 3SA
Siebel Mobile Client
MS Office 2000
ODBC connecting string: sont2db
Siebel Mobile Client
Hostname: tot77
OS: NT v4 with SP5
NT account to install DB2: user77
Software: DB2 Runtime Client with FP 3SA
Siebel Connected Client
MS Office 2000
ODBC connecting string: sont2db
Siebel Connected Client
Hostname: tot88
OS: NT v4 with SP5
NT account to install all software: user88
Software: DB2 Runtime Client with Fixpak 3SA
Siebel Gateway v6.2.1
Siebel Server v6.2.1
Siebel Database Server v6.2.1
Siebel Application Server v6.2
Web Server
ODBC connecting string: sont2db
Siebel Enterprise name: tot88_entnew
Siebel Server name: tot88_siebelnew
Siebel Filesystem: d:\tot88_SFS
DB2 UDB Database Server
Hostname: itsont2
OS: NT v4 with SP5
NT account to install DB2: db2admin
Software: DB2 UDB V6.1 with FP 3SA
Database name: sont2db
NT account created for Siebel: sadmin & siebel

Figure 4. Environment of our DB2 and Siebel installation

3.2 Installing the DB2 UDB Server

This section provides an overview of the sequence of steps required to install a 3-tier Siebel environment. This configuration is the minimum required to have Siebel running. It is comprised of DB2 UDB, Siebel Gateway, Siebel Enterprise Server, Siebel Server, and Siebel clients.

For more detailed description, refer to *Configuring the DB2 UDB for UNIX and Windows NT*, and *Installing the Siebel Database Server with DB2 UDB for UNIX and Windows NT* in the *Siebel Server Installation Guide, Siebel 2000, V6.2,* 10PA1-IN00-06200*.*

1. Install Windows NT 4.0 Server with Service Pack 5.

2. Ensure the ODBC core components are at V3.5 or higher, using the ODBC Data Source Administrator.
3. Install DB2 UDB Enterprise Edition Server code as documented in *DB2 Quick Beginning for Windows NT V6,* GC09-2835.
4. Install the DB2 UDB fixpak for Siebel. The recommended DB2 fixpak (FP) 3SA can be downloaded from:

   `ftp://ftp.software.ibm.com/ps/products/db2/fixes/english-us/siebel/siebel2000/nt/`

   Note: While fixpack 3SA is the supported level of DB2 today, this may change over time as later software levels are certified. Check with Siebel for the latest supported version of DB2 UDB.
5. Verify the DB2 Installation as documented in *DB2 Quick Beginning for Windows NT V6*, GC09-2835.
6. Update DB2 Environment Parameters.
7. Update DB2 Database Manager Configuration.
8. Create a database for Siebel Enterprise Server.
   - DB2 data occupies around 1.2 GB of disk space after installation of Siebel servers. That includes Siebel tables, DB2 system tables and Siebel seed data. For details about DB2 objects, refer to Table 22 on page 89.
   - Prepare at least a 3.5 GB of disk for DB2 logs.
9. Update DB2 Database Configuration for the Siebel database.
10. Create DB2 buffer pools and tablespaces.
11. Create the following Windows NT account and groups:
    a. Siebel table owner account `siebel`.
    b. Siebel administrator account `sadmin`.
    c. Siebel local group `sse_role`.
    d. Add account `sadmin` and `siebel` to the `sse_role` group.
    e. Add the **NT Log on as a Service** and **Act as Part of the Operating System** rights to `sadmin` account.
    f. Configure **DB2-DB2** NT service to **Log On As** the account `sadmin`.

Note

User account `db2fenc1` is not needed in a DB2 UDB NT environment.

3.3 Installing the Siebel Application Server

This section outlines the tasks required to install and configure the Siebel mid-tier Application Server. The Siebel Gateway, Siebel Server and the Siebel Database Server components are installed on this machine. For a detailed description, refer to the *Siebel Server Installation Guide, Siebel 2000, V6.2,* 10PA1-IN00-06200.

For your reference, we have also provided tables indicating our input values for various Siebel installation screens.

3.3.1 DB2 Runtime Client installation

The DB2 Runtime Client has to be installed on the same machine where you run the Siebel Server.

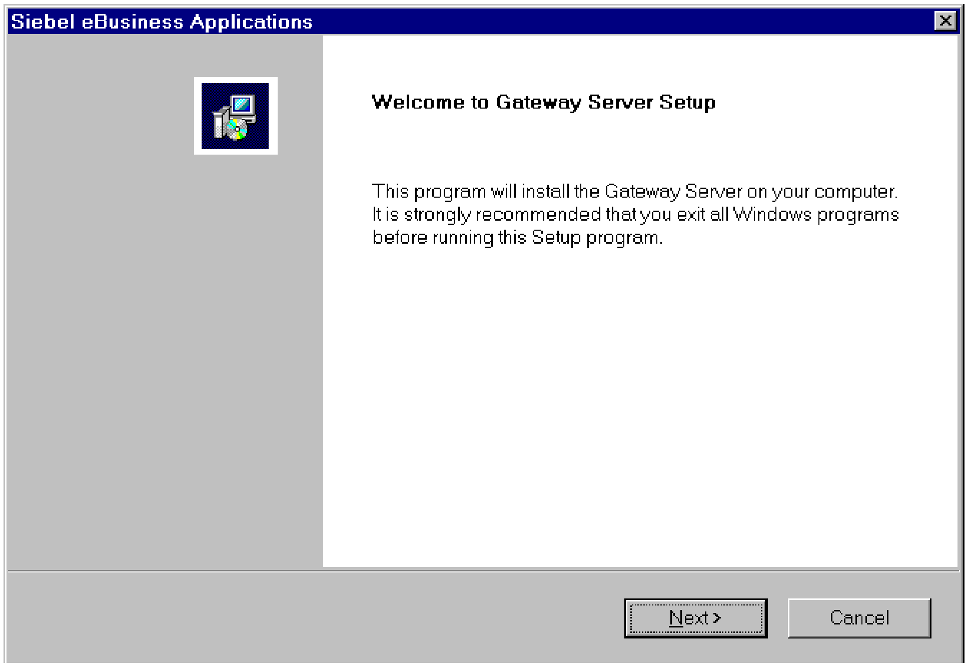
1. Install Windows NT 4.0 Server with Service Pack 5.
2. Ensure the ODBC core components are at V3.5 or higher using the ODBC Data Source Administrator.
3. Install DB2 Runtime Client as documented in *DB2 Quick Beginning for Windows NT V6,* GC09-2835.
4. Install DB2 UDB fixpak 3SA.
5. Create and test the ODBC data source pointing to the database in the Siebel Database Server machine, using the DB2 UDB Client Configuration Assistant.

3.3.2 Installing the Siebel Gateway Server

Install the Siebel Gateway component as documented in the *Siebel Server Installation Guide, Siebel 2000, V6.2,* 10PA1-IN00-06200.

1. Insert the Siebel *e*Business Applications CD-ROM.
2. In Windows NT Explorer, navigate to directory \gtwysrvr and double-click **setup.exe**; you will see a Welcome screen, as shown in Table 5 on page 30.

   Note: If no Siebel software has been previously installed, the Installation Wizard will reboot the machine and launch the setup.exe file again.
3. Select **Next** on the Welcome Screen.

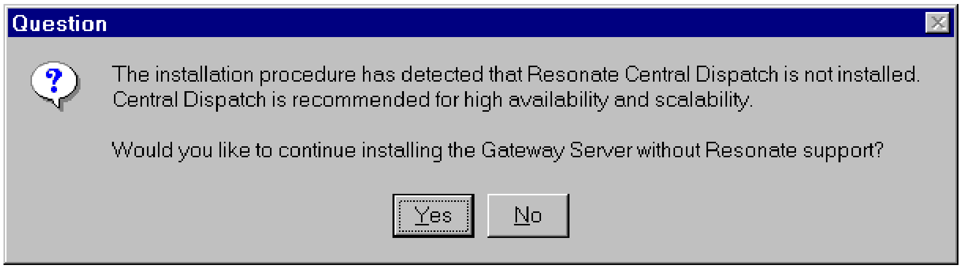


Welcome to Gateway Server Setup

This program will install the Gateway Server on your computer. It is strongly recommended that you exit all Windows programs before running this Setup program.

Figure 5. Welcome to Gateway Server Setup dialog box

If you have not installed Central Dispatch, you will see the following message:

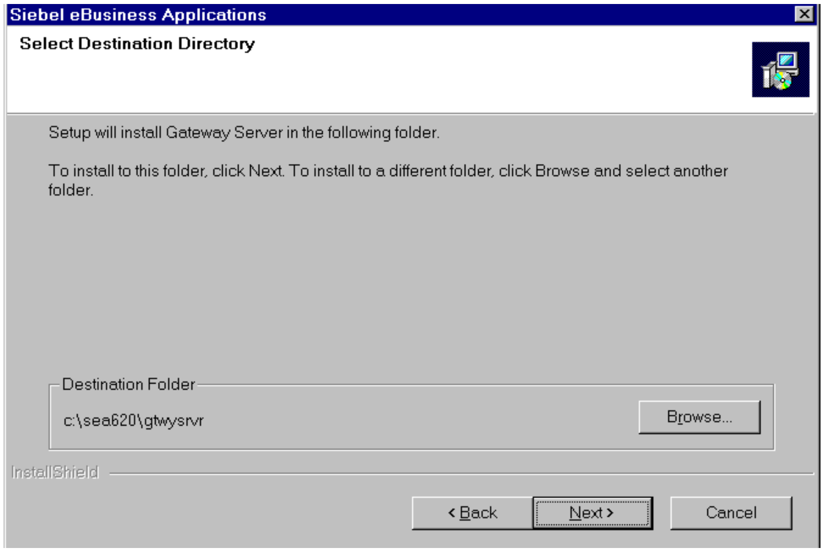


The installation procedure has detected that Resonate Central Dispatch is not installed. Central Dispatch is recommended for high availability and scalability.

Would you like to continue installing the Gateway Server without Resonate support?

Figure 6. Question dialog box

4. Click **Yes** to continue.
5. Select the Destination Directory. You may need to select **Browse** to change the path.


Siebel eBusiness Applications

Select Destination Directory

Setup will install Gateway Server in the following folder.

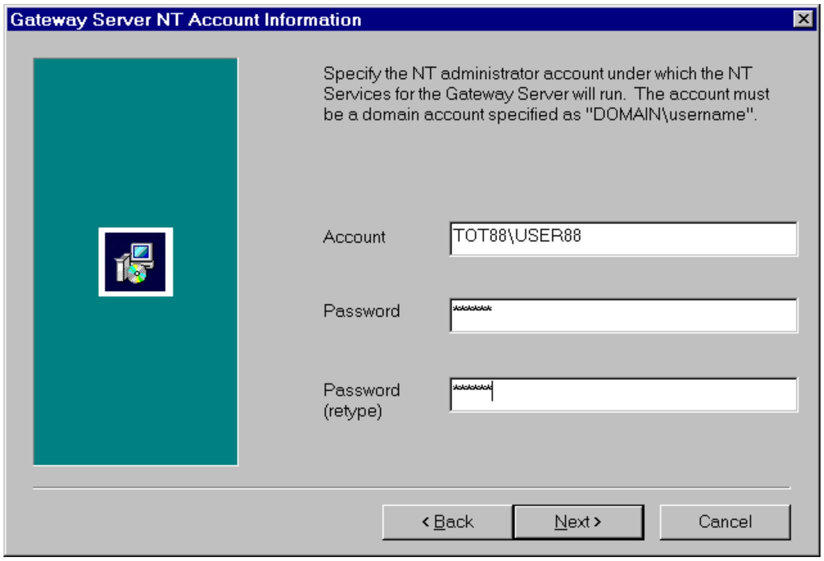
To install to this folder, click Next. To install to a different folder, click Browse and select another folder.

Destination Folder

c:\sea620\gtwysrvr

Figure 7. Select Destination Directory dialog box

6. In the Gateway Server NT Account Information dialog box, type the Siebel Service owner account and password.


Gateway Server NT Account Information

Specify the NT administrator account under which the NT Services for the Gateway Server will run. The account must be a domain account specified as "DOMAIN\username".

Account: TOT88\USER88

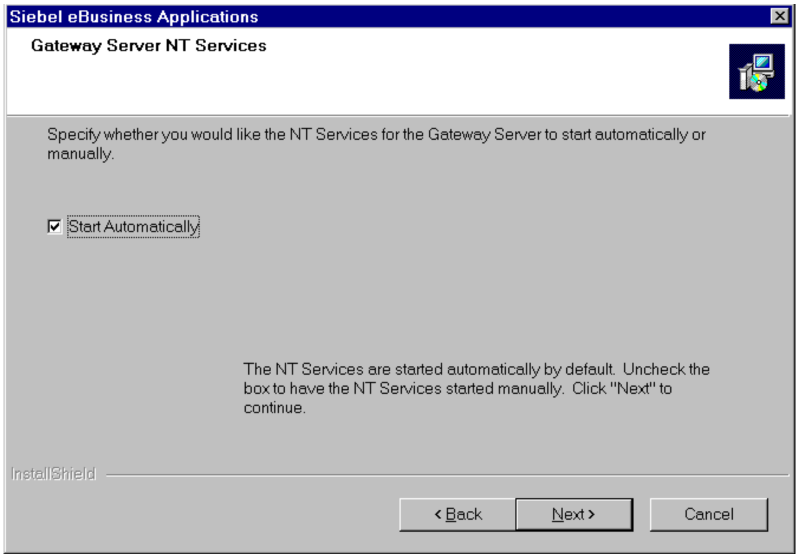
Password: ******

Password (retype): ******

Figure 8. Gateway Server NT Account information dialog box

Note: If you have more than one Siebel Server in the Siebel Enterprise, this must be an Windows NT Domain account (not a Windows NT account for that local server), and it must have Administrator privileges in each application server in which Central Dispatch, the Siebel Gateway, and the Siebel Servers will run.

7. In the Gateway Server NT Services dialog box, check the **Start Automatically** check box.



Siebel eBusiness Applications

Gateway Server NT Services

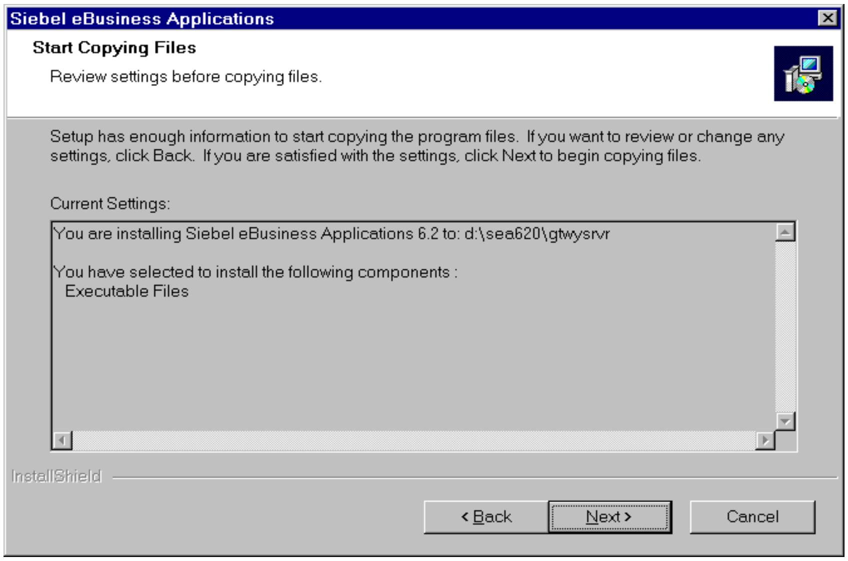
Specify whether you would like the NT Services for the Gateway Server to start automatically or manually.

☑ Start Automatically

The NT Services are started automatically by default. Uncheck the box to have the NT Services started manually. Click "Next" to continue.

Figure 9. Gateway Server NT Services dialog box

8. Click **Next** to start copying files.


Siebel eBusiness Applications
Start Copying Files
Review settings before copying files.

Setup has enough information to start copying the program files. If you want to review or change any settings, click Back. If you are satisfied with the settings, click Next to begin copying files.

Current Settings:

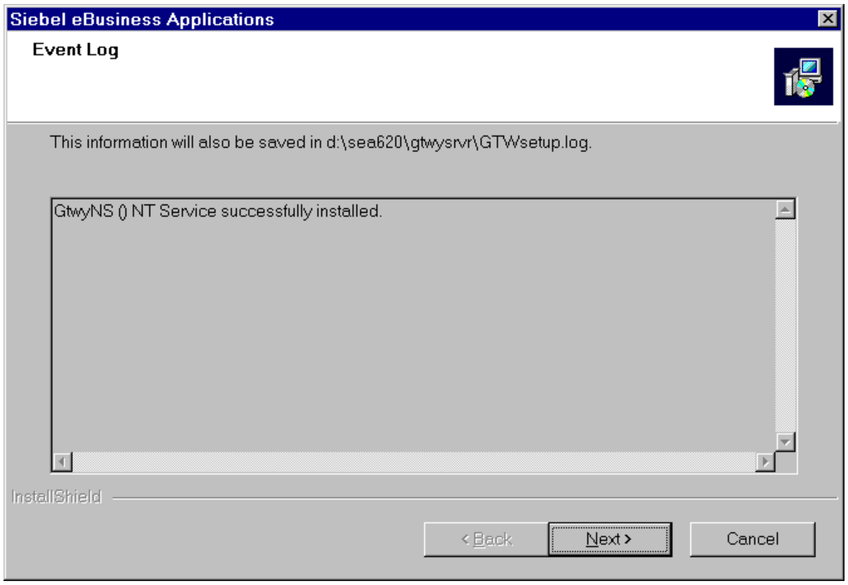
```
You are installing Siebel eBusiness Applications 6.2 to: d:\sea620\gtwysrvr

You have selected to install the following components :
  Executable Files
```

InstallShield
< Back
Next >
Cancel

Figure 10. Start Copying Files dialog box

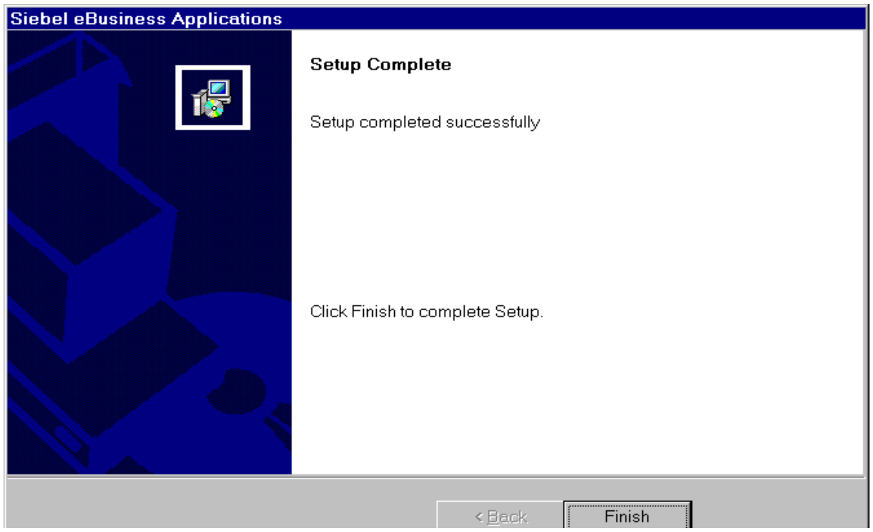
9. Click **Next** when the installation completes.



This information will also be saved in d:\sea620\gtwysrvr\GTWsetup.log.

```
GtwyNS () NT Service successfully installed.
```

Figure 11. Event Log dialog box



Siebel eBusiness Applications

Setup Complete

Setup completed successfully

Click Finish to complete Setup.

Figure 12. Setup Complete dialog box

10.Click **Finish**.

11.Start the Gateway Server.

To start the Gateway Server, navigate to the Windows NT Control Panel and double-click on the Service icon. Find and select the Siebel Name Server and click **Start**.

Table 3 lists our input values for Siebel Gateway Server installation.

Table 3. Siebel Gateway installation-specific information

Installation screen	Installation variable	Our input value
Gateway Server NT Account information	Account	tot88\user88
	Password	user88

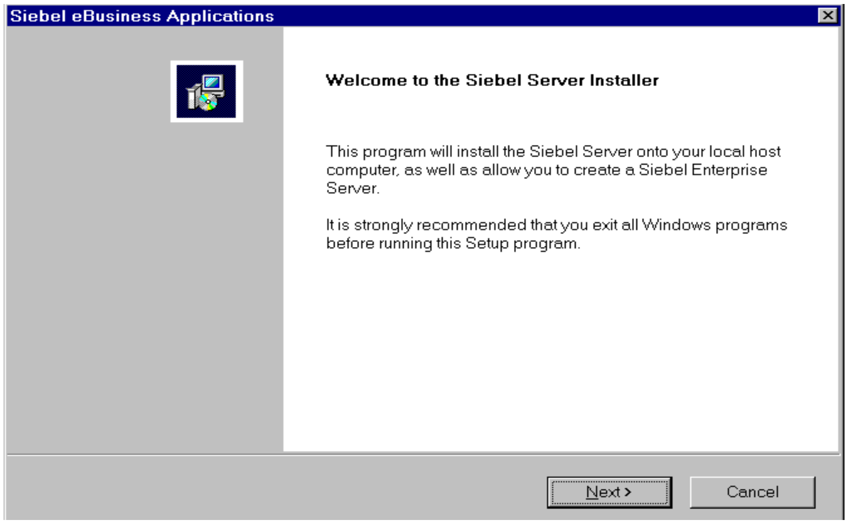
3.3.3 Installing the Siebel Server

Install the Siebel Server component as documented in *Siebel Server Installation Guide, Siebel 2000, V6.2,* 10PA1-IN00-06200.

1. Create the Siebel file system.
2. Shut down all open programs.
3. Start the server install program:

Insert the Siebel *e*Business Applications CD ROM, navigate to the \siebsrvr directory, and double-click **setup.exe**.

4. Select **Next** on the Welcome screen.



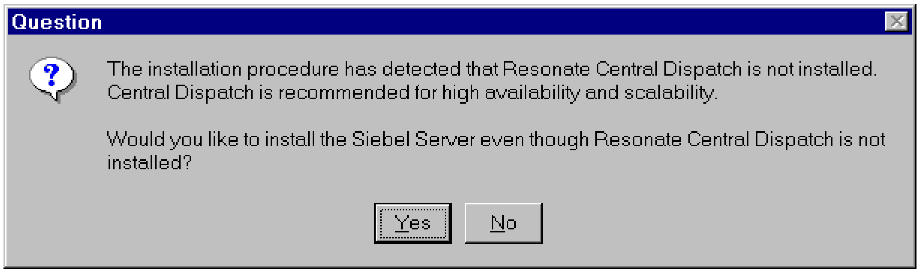
Welcome to the Siebel Server Installer

This program will install the Siebel Server onto your local host computer, as well as allow you to create a Siebel Enterprise Server.

It is strongly recommended that you exit all Windows programs before running this Setup program.

Figure 13. Welcome to the Siebel Server Installer dialog box

If you have not installed Central Dispatch, you will see the following message:



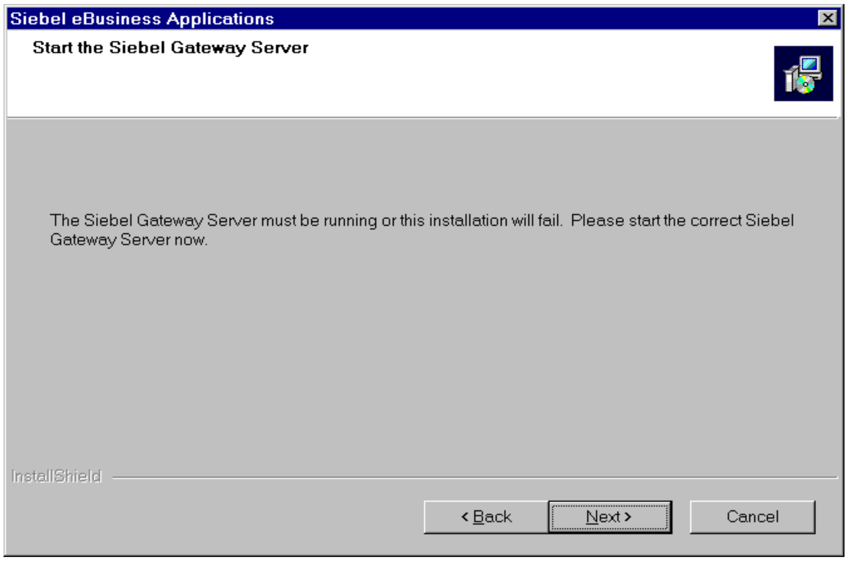
The installation procedure has detected that Resonate Central Dispatch is not installed. Central Dispatch is recommended for high availability and scalability.

Would you like to install the Siebel Server even though Resonate Central Dispatch is not installed?

Figure 14. Question dialog box

5. Click **Yes** to continue.

6. Insure that the Siebel Gateway Name Server Service is running and click **Next.**



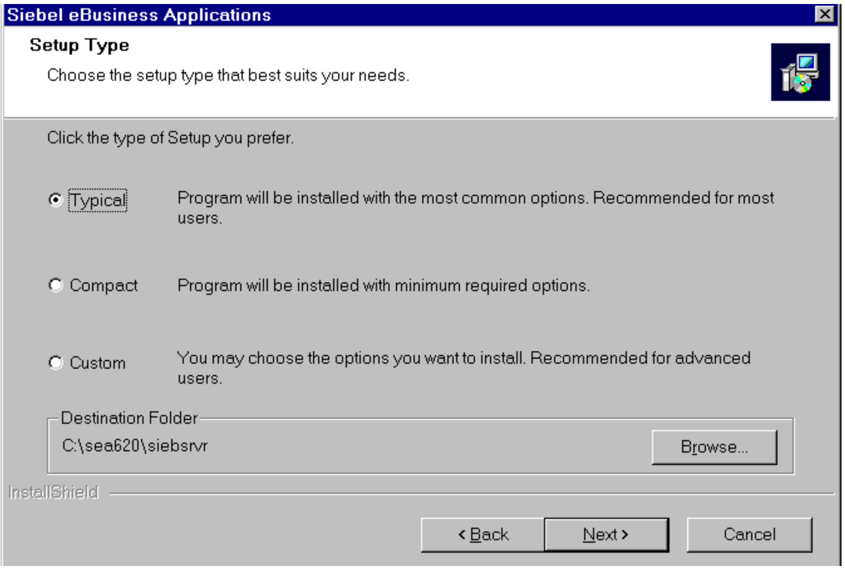
Siebel eBusiness Applications

Start the Siebel Gateway Server

The Siebel Gateway Server must be running or this installation will fail. Please start the correct Siebel Gateway Server now.

Figure 15. Start the Siebel Gateway Server dialog box

7. Select the **Typical** setup and verify the Destination Folder, and then click **Next**.



Siebel eBusiness Applications

Setup Type

Choose the setup type that best suits your needs.

Click the type of Setup you prefer.

● Typical Program will be installed with the most common options. Recommended for most users.

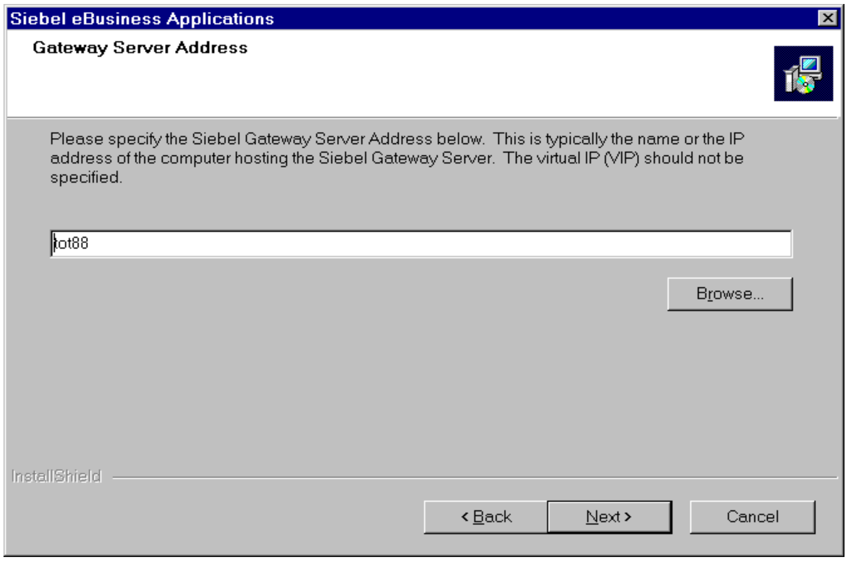
○ Compact Program will be installed with minimum required options.

○ Custom You may choose the options you want to install. Recommended for advanced users.

Destination Folder
C:\sea620\siebsrvr

Figure 16. Setup Type dialog box

8. Verify the Gateway Server Address. Click **Next**.



Siebel eBusiness Applications

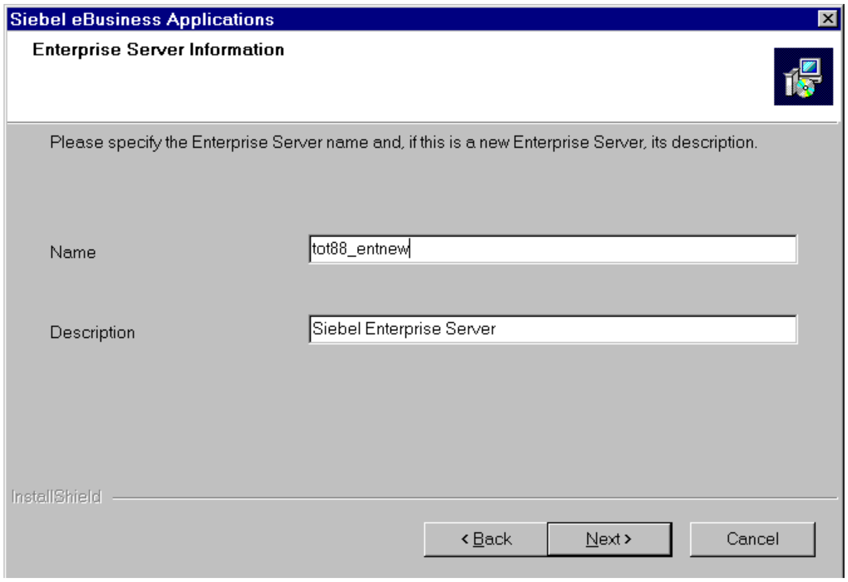
Gateway Server Address

Please specify the Siebel Gateway Server Address below. This is typically the name or the IP address of the computer hosting the Siebel Gateway Server. The virtual IP (VIP) should not be specified.

tot88

Figure 17. Gateway Server Address dialog box

9. Specify the Siebel Enterprise Server Information. Click **Next**.



Siebel eBusiness Applications

Enterprise Server Information

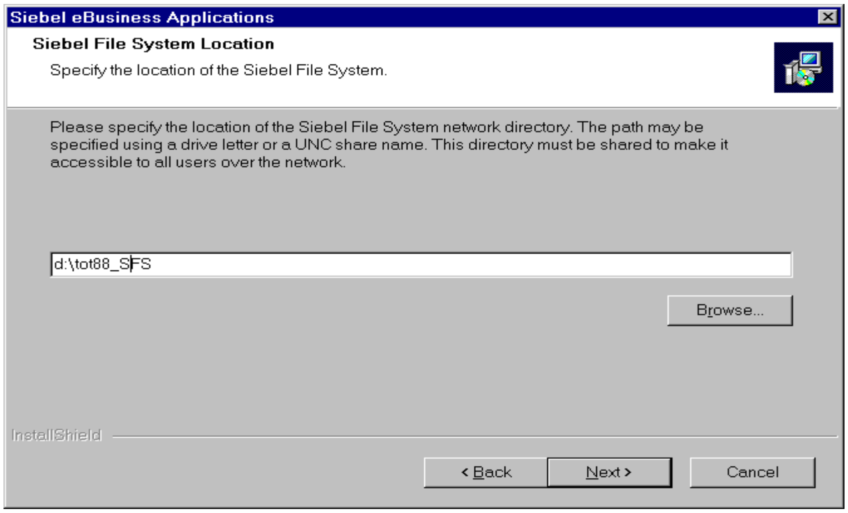
Please specify the Enterprise Server name and, if this is a new Enterprise Server, its description.

Name tot88_entnew

Description Siebel Enterprise Server

Figure 18. Enterprise Server Information dialog box

10.Specify the Siebel File System Location. Click **Next**.



Siebel eBusiness Applications

Siebel File System Location

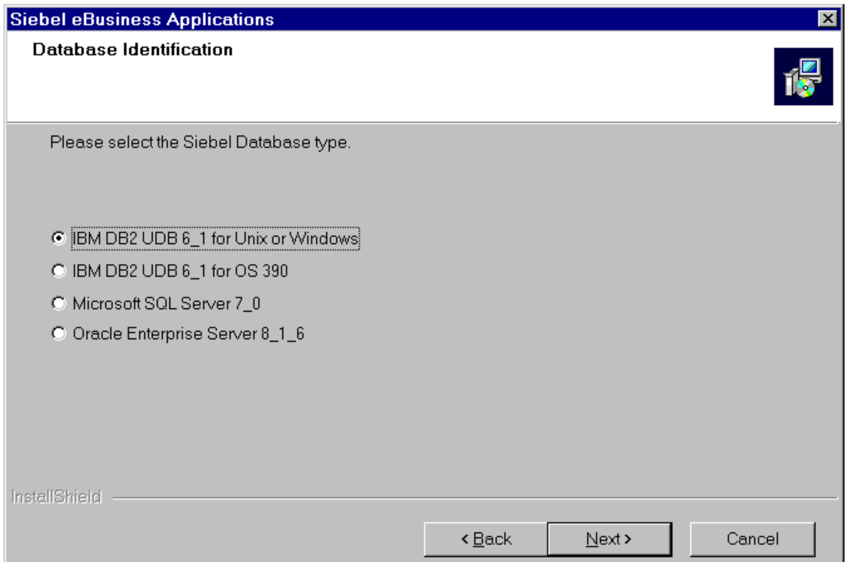
Specify the location of the Siebel File System.

Please specify the location of the Siebel File System network directory. The path may be specified using a drive letter or a UNC share name. This directory must be shared to make it accessible to all users over the network.

d:\tot88_SFS

Figure 19. Siebel File System Location dialog box

11.Select the Database Client and Server Version. Choose **IBM DB2 6_1 for Unix or Windows.**



Siebel eBusiness Applications

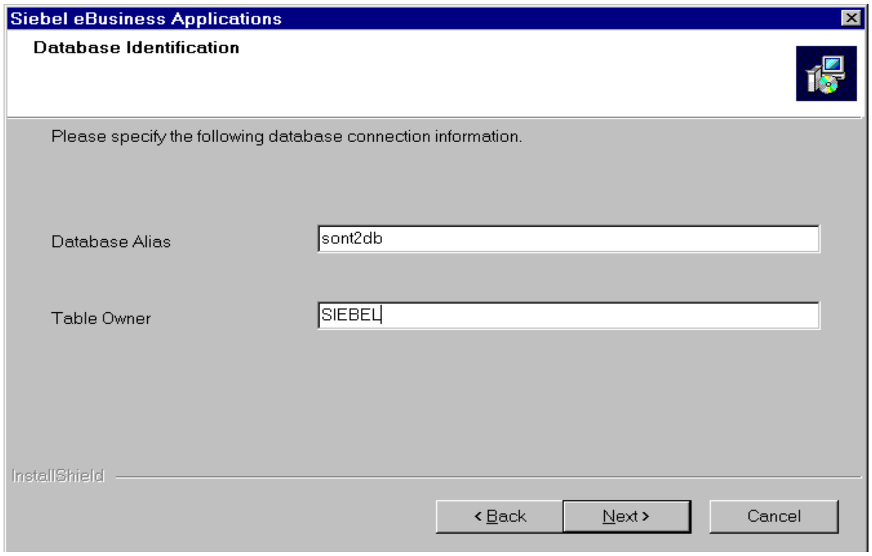
Database Identification

Please select the Siebel Database type.

● IBM DB2 UDB 6_1 for Unix or Windows
○ IBM DB2 UDB 6_1 for OS 390
○ Microsoft SQL Server 7_0
○ Oracle Enterprise Server 8_1_6

Figure 20. Database Identification dialog box

12.Specify the Database Identification - database alias and table owner, and click **Next**.


Siebel eBusiness Applications

Database Identification

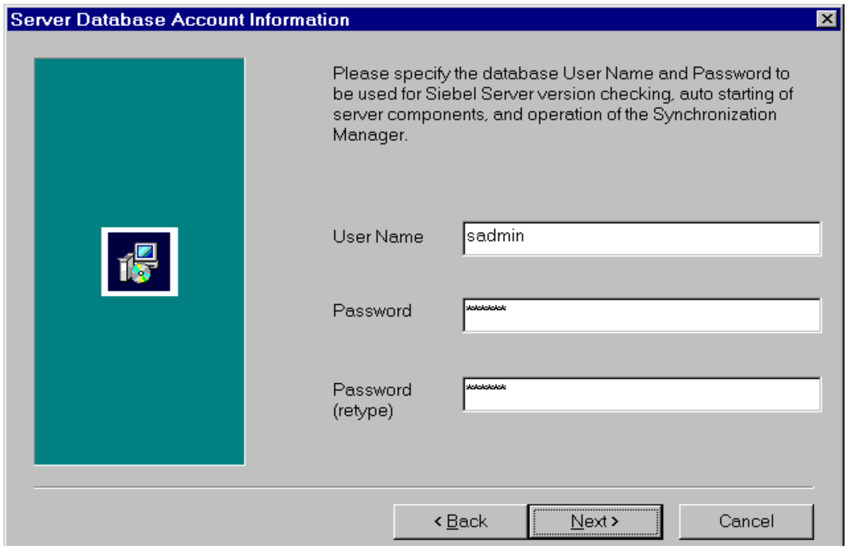
Please specify the following database connection information.

Database Alias: sont2db

Table Owner: SIEBEL

Figure 21. Database Identification dialog box

13.Specify the User Name and Password to be used for Siebel Administration and click **Next**.


Server Database Account Information

Please specify the database User Name and Password to be used for Siebel Server version checking, auto starting of server components, and operation of the Synchronization Manager.

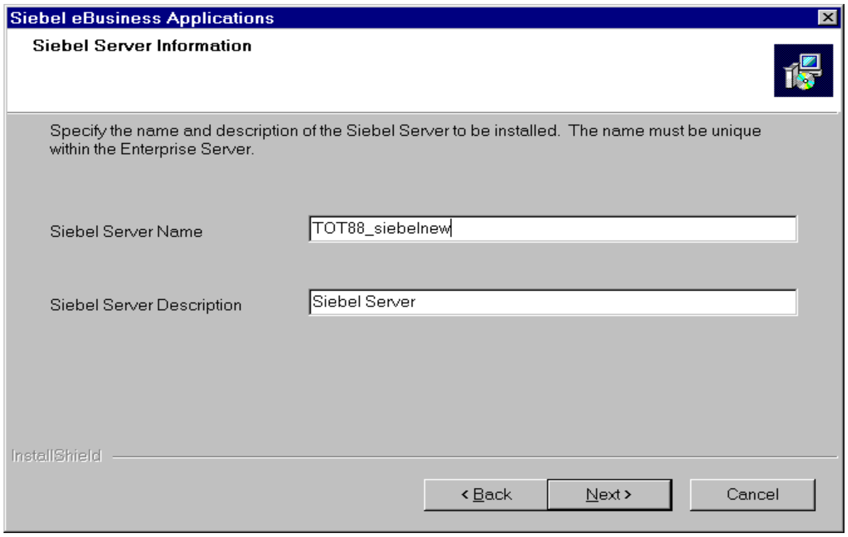
User Name: sadmin

Password: ******

Password (retype): ******

Figure 22. Server Database Account Information dialog box

14.Specify The Name and Description for the Siebel Server to be installed.



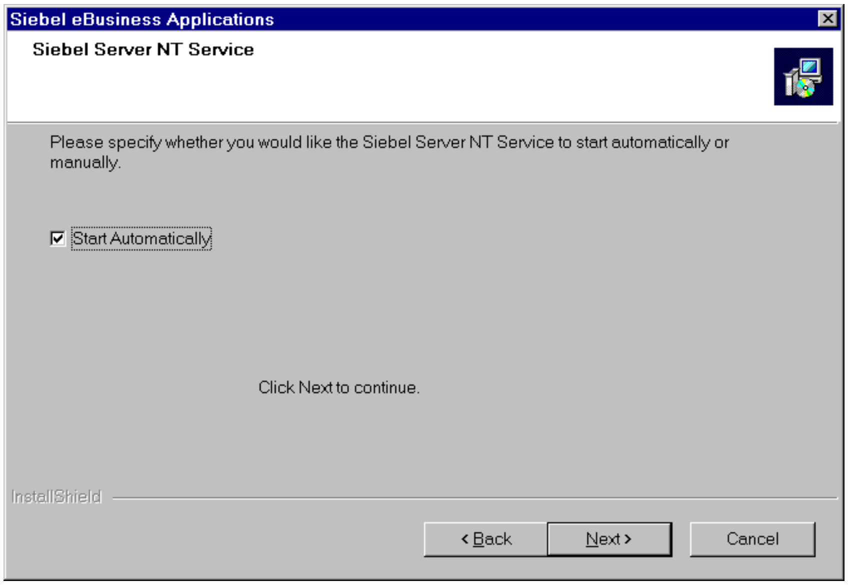
Specify the name and description of the Siebel Server to be installed. The name must be unique within the Enterprise Server.

Siebel Server Name: TOT88_siebelnew

Siebel Server Description: Siebel Server

Figure 23. Siebel Server Information dialog box

15.Specify **Start Automatically** for the Siebel Server NT Service.



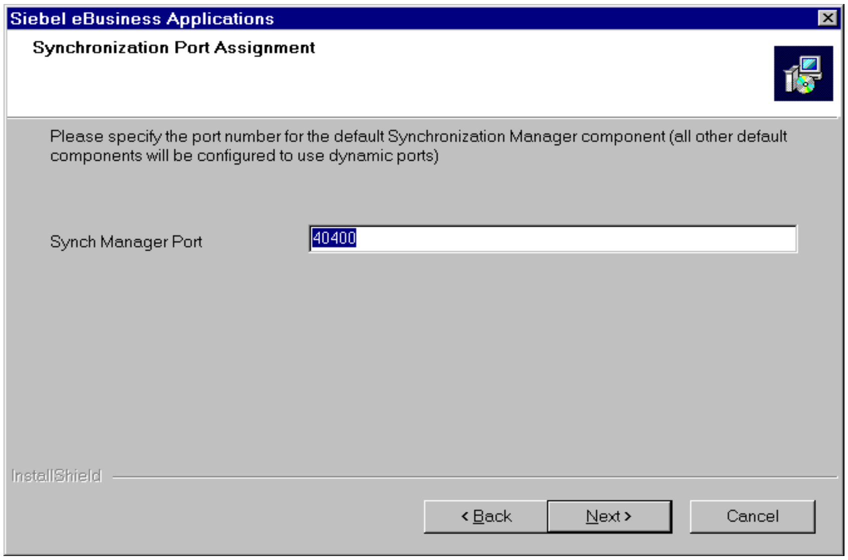
Please specify whether you would like the Siebel Server NT Service to start automatically or manually.

☑ Start Automatically

Click Next to continue.

Figure 24. Siebel Server NT Service dialog box

16. This window is used only if connection brokering is implemented. We did not use this function, so we simply clicked **Next**.



Siebel eBusiness Applications

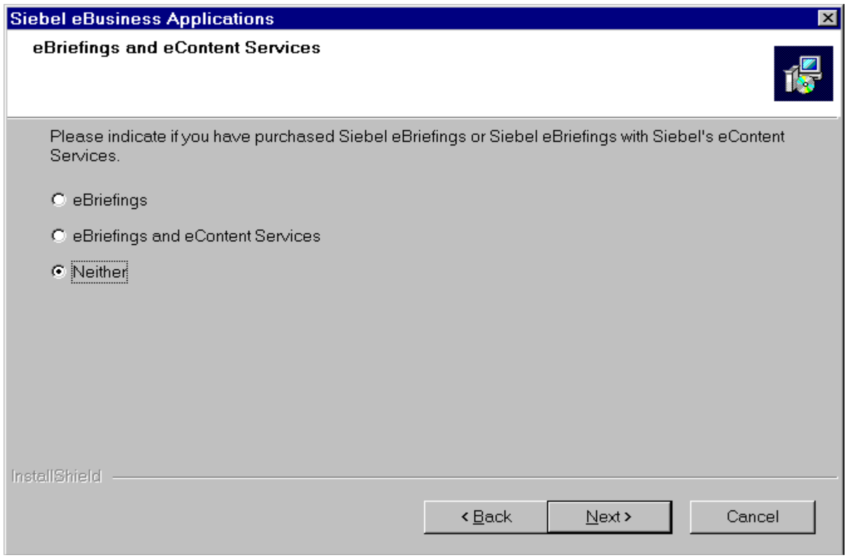
Synchronization Port Assignment

Please specify the port number for the default Synchronization Manager component (all other default components will be configured to use dynamic ports)

Synch Manager Port: 40400

Figure 25. Synchronization Port Assignment dialog box

17. We were not implementing eBriefings or eContent Services, so we selected Neither, and clicked **Next**.



Siebel eBusiness Applications

eBriefings and eContent Services

Please indicate if you have purchased Siebel eBriefings or Siebel eBriefings with Siebel's eContent Services.

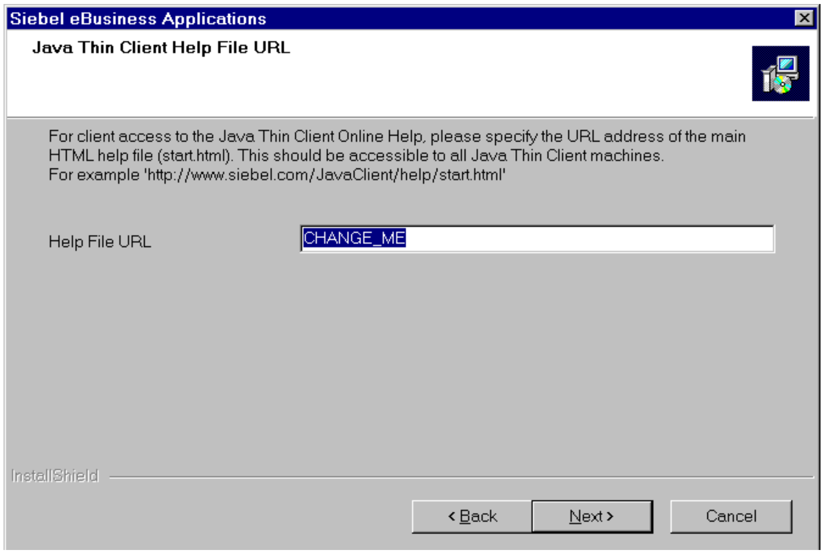
○ eBriefings

○ eBriefings and eContent Services

● Neither

Figure 26. eBriefing and eContent Services dialog box

18.Specify the Java Thin Client Help File URL. We did not implement Siebel Thin Client, so we simply clicked **Next**.



Siebel eBusiness Applications

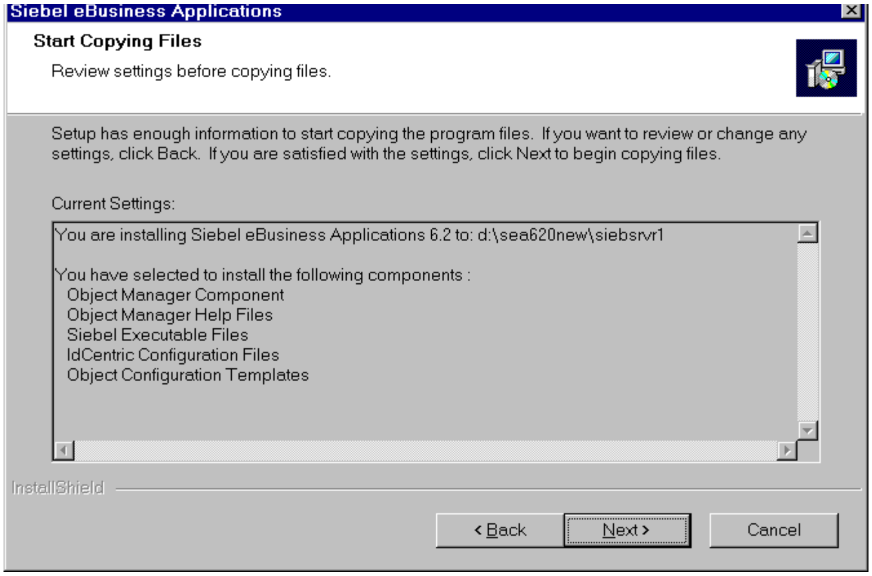
Java Thin Client Help File URL

For client access to the Java Thin Client Online Help, please specify the URL address of the main HTML help file (start.html). This should be accessible to all Java Thin Client machines.
For example 'http://www.siebel.com/JavaClient/help/start.html'

Help File URL: CHANGE_ME

Figure 27. Java Thin Client Help File URL dialog box

19.Click **Next** to start copying files.



Siebel eBusiness Applications

Start Copying Files

Review settings before copying files.

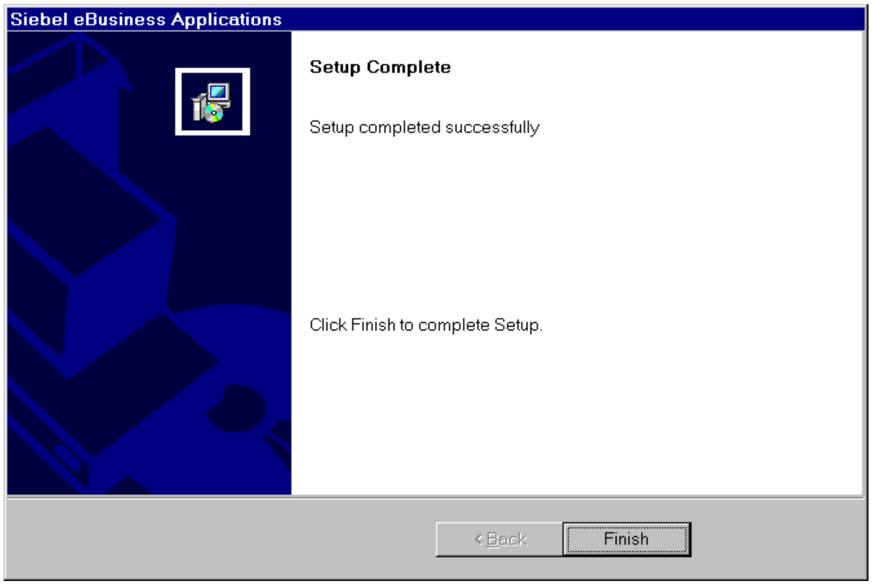
Setup has enough information to start copying the program files. If you want to review or change any settings, click Back. If you are satisfied with the settings, click Next to begin copying files.

Current Settings:

```
You are installing Siebel eBusiness Applications 6.2 to: d:\sea620new\siebsrvr1

You have selected to install the following components :
  Object Manager Component
  Object Manager Help Files
  Siebel Executable Files
  IdCentric Configuration Files
  Object Configuration Templates
```

Figure 28. Start Copying Files dialog box



Siebel eBusiness Applications

Setup Complete

Setup completed successfully

Click Finish to complete Setup.

Figure 29. Setup Complete dialog box

20.Click **Finish**.

Table 4 lists the values used at our installation.

Table 4. Siebel Server installation-specific information

Installation screen	Installation variable	Our input value
Enterprise Server Information	Name	tot88_entnew
	Description	Siebel Enterprise Server
Database Identification	Database alias	sont2db
	Table Owner	siebel
Server Database Account Login Information	User Name	sadmin
	Password	sadmin
Siebel Server Information	Siebel Server Name	tot88_siebelnew
	Siebel Server Description	Siebel Server

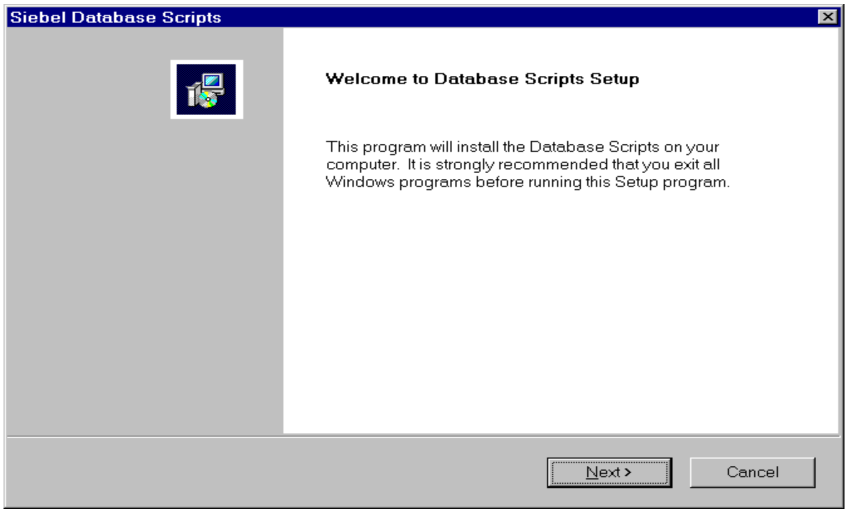
3.3.4 Installing the Siebel Database Server component

Install the Siebel Database Server component as documented in the *Siebel Server Installation Guide, Siebel 2000, V6.2,* 10PA1-IN00-06200.

1. Start the Siebel Database Server install programs:

   Insert the Siebel *e*Business Applications CD-ROM, navigate to the \dbsrvr directory, and double-click **setup.exe**.

2. Select **Next** on the Welcome screen.

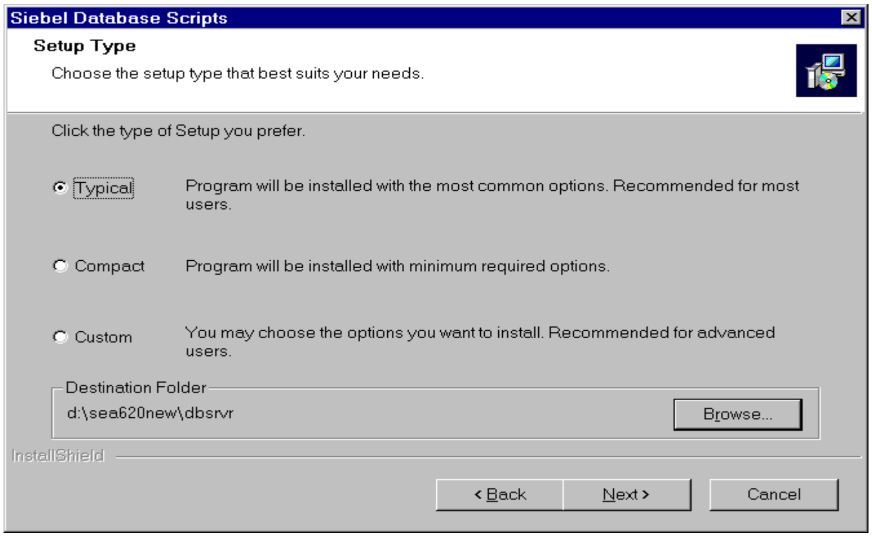


Welcome to Database Scripts Setup

This program will install the Database Scripts on your computer. It is strongly recommended that you exit all Windows programs before running this Setup program.

Figure 30. Welcome to Database Scripts Setup dialog box

3. Select **Typical** install and verify the Destination Folder.



Click the type of Setup you prefer.

● Typical — Program will be installed with the most common options. Recommended for most users.

○ Compact — Program will be installed with minimum required options.

○ Custom — You may choose the options you want to install. Recommended for advanced users.

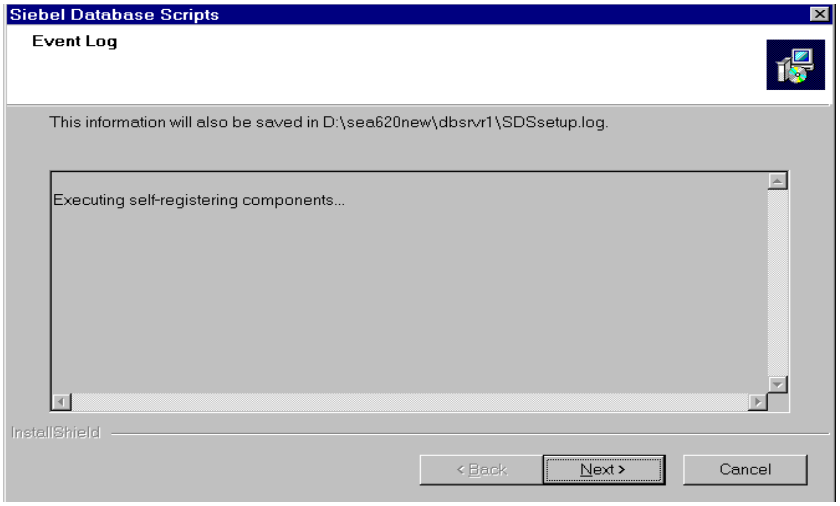
Destination Folder: d:\sea620new\dbsrvr

Figure 31. Setup Type dialog box

Note: When you select **Typical** as the type of setup, then scripts for all supported database platforms will be installed onto the target computer and drive/folder. This will consume a little more disk space.

When you select **Custom**, only scripts for DB2 UDB will be installed.

4. Review the Event Log and click **Next**.

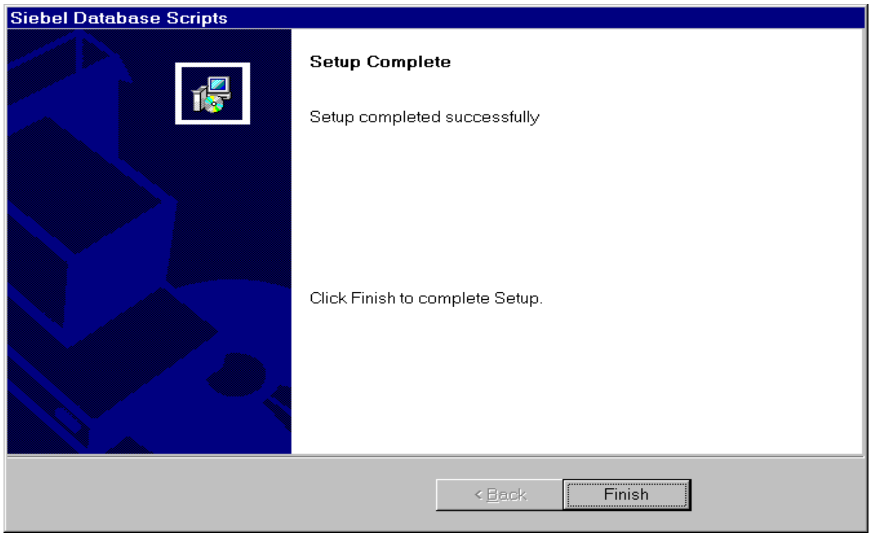


This information will also be saved in D:\sea620new\dbsrvr1\SDSsetup.log.

```
Executing self-registering components...
```

Figure 32. Event Log dialog box

5. Click **Finish** at the Setup Complete window.



Siebel Database Scripts

Setup Complete

Setup completed successfully

Click Finish to complete Setup.

Figure 33. Setup Complete dialog box

6. Update and run `grantusr.sql` to grant privileges to the Siebel account.
   a. Change all user IDs to lowercase.
   b. Comment out the line: `GRANT DBADM ON DATABASE TO USER DB2FENC1`.
7. Run `installsiebel` to copy the Siebel UDF and Stored Procedures to the db2 function directory.
8. Update and run install.ksh to create the Siebel Schema.
   This script will do the following:
   a. It creates Siebel tables and indexes.
   b. It imports Siebel Seed data.
   c. It creates Siebel Stored procedures and UDFs.
   d. It binds Siebel packages.
   e. It seeds Siebel control tables with DB2-specific information.
   f. It creates new Siebel views.
   g. It updates some Siebel table statistics.
   h. It performs runstats for most of the Siebel tables.

Table 5 lists the values used at our installation.

Table 5. Install.ksh installation-specific information

Variable	Our values
SRC_USR	sadmin
SRC_PSWD	sadmin
SRC_TBLO	siebel
SRC_TBLO_PSWD	siebel
DBSRVR_ROOT	d:\sea620\dbsrvr
DATA_AREA	“SIEBEL_4K”
INDX_AREA	“SIEBEL_IDX”
LONG_AREA	“SIEBEL_4KL”
DB16K_AREA	“SIEBEL_16K”
DB_LANG	enu
DB_UNICODE_FLG	N
DBSERVER_OS	winnt
SIEBEL_HOME	d:\sea620\siebsrvr

i. Update and run imprep.ksh to import the Siebel Repository.

Table 6 lists the values in imprep.ksh script at our installation.

Table 6. Imprep.ksh installation specific information

Variable	Our values
SRC_USR	sadmin
SRC_PSWD	sadmin
SRC_TBLO	siebel
ODBC	sont2db
REPOS_NAME	“Siebel Repository”
DBSRVR_ROOT	d:\sea620\dbsrvr
DATABASE_PLATFORM	DB2udb
DB_LANG	enu
SIEBEL_HOME	d:\sea620\siebsrvr

3.4 Post-installation

After successfully installing all Siebel components, ensure that a full DB2 database offline backup is taken so that you can restore the database if there is a need to.

We strongly recommend that you perform a RUNSTATS again on the tables in the Siebel database (either using updatestats.ksh, or by the DBA on individual tables), and then perform a RUNSTATS on the System Catalog tables in the Siebel database.

Note that there are a number of tables that you should *never* do runstats on; these tables are located in the loadstats.sql file from Siebel. Siebel sets some statistics as it needs, and if you run DB2 runstats later on these tables, you will overwrite this information and have performance issues. Refer to 9.4, “The runstats utility vs. updatestats” on page 156 for more information.

Important

If statistics are not updated, then Siebel Tools will perform poorly since DB2 statistics indicate zero records in most tables used to hold Repository information, but some tables (e.g. S_COLUMN) have 59,000 records after imprep.ksh.

Then, install a Siebel client and test it. Details are covered in Chapter 5, “Client installation” on page 67.

Chapter 4. Installing Siebel on AIX

In this chapter we describe the installation of Siebel Server components on AIX using DB2 UDB. We cover the 3-tier configuration, since this is the most common scenario. The intention of this chapter is to clarify and simplify some steps found in the Siebel documentation, rather than replace that information.

Checklists

- Review *DB2 UDB V6 Release notes, Siebel 2000, V6.2.1*.
- Review *Siebel System Requirements and Supported Platforms, Siebel 2000, V6.2.1*, 10PA1-RP00-06210, to ensure you have the necessary hardware and software required for the environment.
- Review *Siebel Client Installation and Administration Guide, Siebel 2000, V6.2,* 10PA1-CA00-06200 for client installation information.
- Review *Siebel Server Installation Guide, Siebel 2000, V6.2*, 10PA1-IN00-06200 for Siebel Server component installation.
- Review *Siebel Server Administration Guide, Siebel 2000, V6.0, rev. A,* for Siebel Server administration.
- Review *DB2 Quick Beginning for UNIX V6* GC09-2836 for DB2 client and server installation.
- Complete the Development Planning Worksheet A-1 of *Siebel Server Installation Guide*.

Installation process summary

The general installation process consists of several steps:

1. Preparing for the installation.
2. Installing the Siebel Gateway Server.
   a. Installing the Resonate Central Dispatch product (optional).
   b. Installing the Gateway Server itself. (This includes the Name Server.)
3. Creating an Enterprise Server and installing one or more Siebel Servers.
4. Configuring and installing the Siebel Database Server with DB2 UDB.
5. Installing the CORBA Object Manager (optional).
6. Installing the Siebel Thin Client Applications (optional).
7. Installing the Siebel Report Server (optional).
8. Installing the Siebel Dedicated Client (described in Chapter 5, “Client installation” on page 67).

We did not install the optional components, so those installation steps are not described here. If you will use some of the optional components, refer to *Siebel Server Installation Guide*, Siebel 2000, V6.2, 10PA1-IN00-06200.

This chapter is divided into four parts:

1. Preparing for the installation
2. Installing DB2
3. Installing and configuring Siebel Server components:
    a. Siebel Gateway Server
    b. Siebel Server
    c. Siebel Database Server with DB2 UDB
4. Post-installation considerations

4.1 Pre-installation steps

It is very important that you review the checklists before starting any installation.

Note

Your machines must be able to ping each other by hostname. If you are unable to ping each machine, you have to add IP addresses and their hostnames into the hosts file, which is located in the /etc directory on AIX.

4.1.1 Our Siebel DB2 environment

Our test environment is described in Figure 34 on page 51. This figure may be helpful for you to compare the different names of servers, Siebel components, users, etc. with yours, and clarify the architecture and installation procedure.

Our installation steps are described for our architecture. We have two AIX servers: the DB2 UDB database server and the Siebel Application server. First we describe how to finish the installation on the DB2 UDB database server, and then on the Siebel Application server.

If your environment is different from ours, the installation steps may also differ.

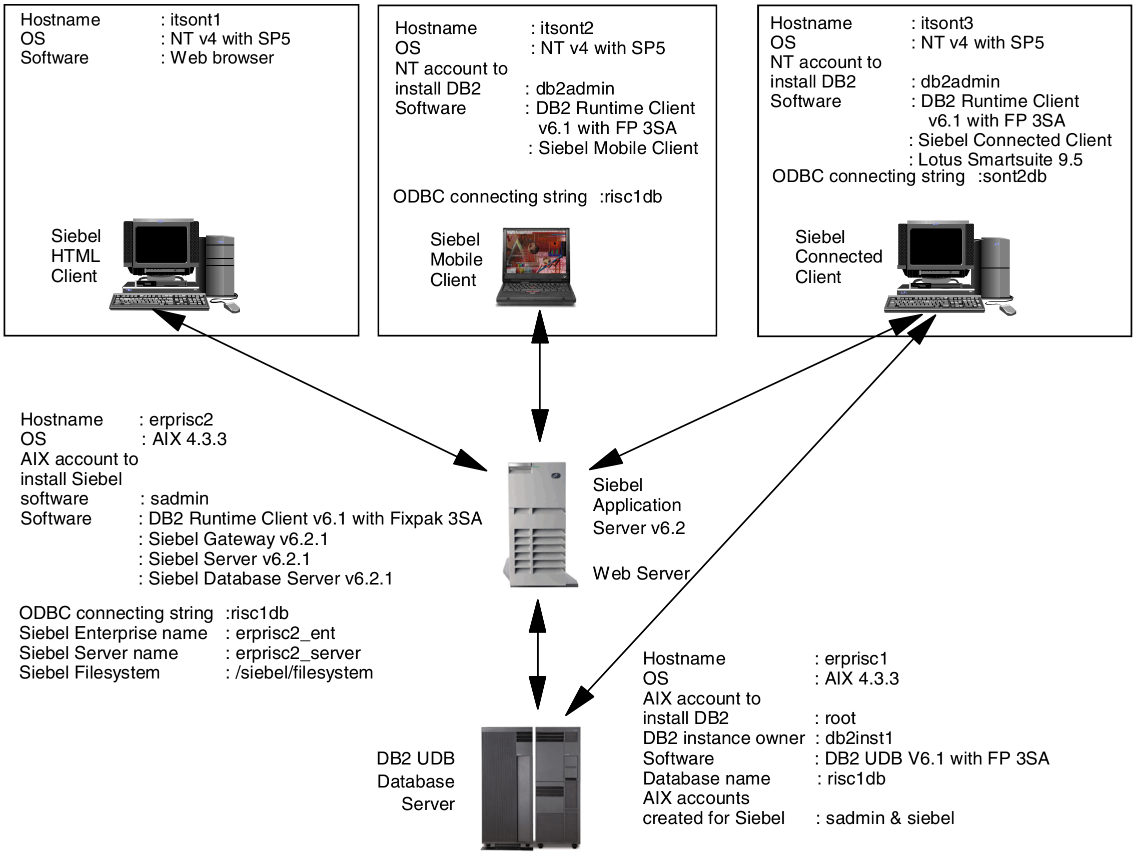

Hostname : itsont1
OS : NT v4 with SP5
Software : Web browser
Siebel HTML Client
Hostname : itsont2
OS : NT v4 with SP5
NT account to install DB2 : db2admin
Software : DB2 Runtime Client v6.1 with FP 3SA
: Siebel Mobile Client
ODBC connecting string :risc1db
Siebel Mobile Client
Hostname : itsont3
OS : NT v4 with SP5
NT account to install DB2 : db2admin
Software : DB2 Runtime Client v6.1 with FP 3SA
: Siebel Connected Client
: Lotus Smartsuite 9.5
ODBC connecting string :sont2db
Siebel Connected Client
Hostname : erprisc2
OS : AIX 4.3.3
AIX account to install Siebel software : sadmin
Software : DB2 Runtime Client v6.1 with Fixpak 3SA
: Siebel Gateway v6.2.1
: Siebel Server v6.2.1
: Siebel Database Server v6.2.1
ODBC connecting string :risc1db
Siebel Enterprise name : erprisc2_ent
Siebel Server name : erprisc2_server
Siebel Filesystem : /siebel/filesystem
Siebel Application Server v6.2
Web Server
Hostname : erprisc1
OS : AIX 4.3.3
AIX account to install DB2 : root
DB2 instance owner : db2inst1
Software : DB2 UDB V6.1 with FP 3SA
Database name : risc1db
AIX accounts created for Siebel : sadmin & siebel
DB2 UDB Database Server

Figure 34. Environment of our DB2 and Siebel installation

4.2 DB2 UDB database server installation

In this section we describe the DB2 installation checkpoints for the DB2 UDB database server only. *All steps have to be done on this machine*.

Follow the steps below to finish the installation:

1. Install the DB2 software. If you do not have experience with DB2 installation, follow *DB2 Quick Beginning for UNIX V6,* GC09-2836, Chapter 2 “Installing DB2 for AIX”.

   Siebel 2000 supports DB2 Enterprise Edition with FixPak 3SA.

2. After successful installation of DB2, you have to apply FixPak 3SA. FixPak can be downloaded from:

`ftp://ftp.software.ibm.com/ps/products/db2/fixes/english-us/siebel/siebel2000/aix/`

Select version 3SA. Follow the instructions in the Fixpak.readme file to successfully install the FixPak. (Note that all DB2-related processes have to be stopped before installing the FixPak.) After the installation, all existing instances have to be updated as described in the readme file. After that, DB2 is ready for Siebel installation.

Note: While fixpack 3SA is the supported level of DB2 today, this may change over time as later software levels are certified. Check with Siebel for the latest supported version of DB2 UDB.

3. Create a new Siebel database.

- DB2 data occupies around 1.2 GB of disk space after installation of Siebel servers. That includes Siebel tables, DB2 system tables and Siebel seed data. For details about DB2 objects, refer to Table 22 on page 89.
- Create separate file systems for DB2 data and DB2 logs. Be aware that at a minimum, a DB2 instance and DB2 Administration Server (DAS) owner should have access to these file systems. We suggest that you spread DB2 log files and data around on different disks.
- Prepare at least a 3.5 GB file system for DB2 logs. Consider using an even larger file system, as your Siebel environment will grow in the future.

4.2.1 Create users and groups

After the successful installation of the DB2 server, create the users and groups that will be used by the Siebel application. Users and groups can be created with root authority. We suggest that you use lowercase user and group names for DB2, because you can not connect to DB2 on AIX if using an uppercase user ID.

1. Create Siebel table owner user `siebel`.
2. Create Siebel administrator user `sadmin`.
3. Create group `sse_role`.
4. Put users `siebel`, `sadmin` and `db2fenc1` in the `sse_role` group (user `db2inst1` is created during DB2 installation).

4.2.2 Connecting user to DB2

User `sadmin` must be able to connect to DB2 after logging on AIX:

- The `.db2profile` has to be executed in the `.profile` of the user `sadmin`, so add `. /home/db2inst1/sqllib/db2profile` to the `.profile` file of user `sadmin`.
- After that, the default DB2 instance should be displayed when running `echo $DB2INSTANCE` (for example, db2inst1).
- You will be able to connect to DB2 with user sadmin after finishing the step called “run grantusr.sql” described in 4.3.5, “Installing the Siebel Database Server component” on page 60; this step grants DB2 access to user `sadmin` and group `sse_role`.

At this point, the DB2 server installation is complete. All further tasks will be performed on the Siebel Application Server.

4.3 Siebel Application Server installation

In this section we describe how to install the DB2 Runtime Client and Siebel Server components on the Siebel Application Server machine. We cover Siebel Gateway Server, Siebel Enterprise Server (Application server), and Siebel Database server component installation on the same machine. *All steps have to be done on this machine*.

4.3.1 DB2 Runtime Client installation

1. The DB2 Runtime Client has to be installed on the same machine where you run the Siebel Server. If you do not have experience with DB2 installation, follow *DB2 Quick Beginning for UNIX V6 GC09-2836,* Chapter 6 “Installing DB2 Clients on UNIX Operating Systems”.

   Note*:* If you will install Siebel components and DB2 on one machine (for example, on a test machine), this step is not required.

2. After successful installation of the DB2 Runtime Client, FixPak 3SA has to be applied. Follow the instructions in the Fixpak.Readme file to successfully install FixPak. (Note that all DB2-related processes have to be stopped before installing FixPak.)

3. You have to establish DB2 connectivity from your Siebel Server machine to the DB2 server. Catalog the node and database on the client side by using the following command:

   ```
   db2 catalog tcpip node <nodename> remote <hostname>
   ```

   ```
   db2 catalog database <database-name> as db-alias at node
   <nodename>
   ```

Your DB2 environment on AIX is now ready for the installation and configuration of Siebel Server components.

4.3.2 Installing the Siebel Gateway Server

In this section, we describe all installation steps for the Siebel Gateway Server. For details and explanations, see the *Siebel Server Installation Guide*.

During our installation, we noticed that it is useful to create all users, user groups, file systems, and directories in advance. Although this is not a requirement, it does expedite the installation procedure. Follow the instructions below.

On the Siebel Application Server machine, complete the following tasks as root:

1. Create Siebel table owner user `siebel`.
2. Create the Siebel administrator user `sadmin`; this user will be used for installation purposes.
3. Create group `sse_role`
4. Put users `siebel`, `sadmin` and `db2fenc1` in the `sse_role` group (user `db2inst1` is created during DB2 installation).

> **Note**
>
> As previously mentioned, we suggest that you use lowercase user and group names for DB2 because you cannot connect to DB2 on AIX if using an uppercase user ID.

5. Create the directory where the Siebel Server software will be installed.

   User sadmin has to own this directory; as root, use the following command:

   ```
   chown sadmin:sse_role siebel
   ```

   where sadmin is the new owner, sse_role is the group to which the user sadmin belongs, and siebel is the directory in which you will install the Siebel Server software.
6. Create the Siebel file system directory:

- User sadmin has to own this directory; as root use the following command:

  ```
  chown sadmin:sse_role siebelfs
  ```

  where sadmin is the new owner, sse_role is the group to which the user sadmin belongs, and siebelfs is the Siebel file system directory.

- The Siebel file system this directory is in should be around 500 MB.

  Note: You will need third-party software (for example, Samba) to access the Siebel file system on AIX from a Connected or Mobile Client on Windows NT; this software is not provided by IBM or Siebel.

- You can use an additional Windows NT server for the Siebel file system.

7. Verify Gateway Server Prerequisites.

- Verify that the application server computer that will host the Gateway Server meets hardware and operating system requirements.
- TCP/IP must be enabled.
- If you are using connection brokering, make sure that Central Dispatch is installed on the application server that will support the Gateway Server, as well as on all application servers that will support Siebel Servers.

Note

Install all Siebel components as user sadmin.

8. As user sadmin, install the Siebel Gateway Server.

   a. Run the `ksh` and `umask 027` commands.

   b. Run the following command:

   ```
   ./install_gateway -d /usr/tmp/gateway.log
   ```

   This will generate an installation log in the /usr/tmp directory so you can review the installation process.

   c. Choose the appropriate prompts during installation; see Table 7.

Table 7. Siebel Gateway installation-specific information

Installation screen	Installation default value	Our input value
Please specify the directory into which Siebel should be installed	/siebel	/siebel
Is Resonate installed for this Enterprise?	N	N
Would you like to accept these settings?	Y	Y
Would you like to configure this section? (Autostart setting)	Y	Y

Installation screen	Installation default value	Our input value
Should the Siebel Gateway Server have autostart enabled (Y or N)?	blank	Y
Do you want to apply these parameter values?	Y	Y

9. Post-installation tasks:
   a. Open an AIX shell window and enter `ksh` to run a Korn shell.
   b. Navigate to the SIEBEL_ROOT directory and run the following to execute the Siebel environment script:

```
. ./siebenv.sh
```

   c. Run the `set` command; you should then see SIEBEL_ROOT, SIEBEL_GATEWAY variables, and a new PATH and LIBPATH.
   d. To verify that the Gateway Server is running, enter: `./bin/list_ns`
   e. Configure the Siebel Gateway Server for Automatic Start.

   Note: If you encounter any errors in this step, contact your Siebel representative for support.

4.3.3 Installing the Siebel Server

1. Pre-installation tasks:
   a. Verify Siebel Server Prerequisites.
   b. Configure Database Connectivity Software.
   c. Verify Network Connectivity.
   d. Start the Gateway Server.
2. As user `sadmin`, install the Siebel Application Server:
   a. Run `ksh` and `umask 027` commands.
   b. Run the following command:

```
./install_server -d /usr/tmp/server.log
```

   This will generate an installation log in the /usr/tmp directory so you can review the installation process.

c. Choose the appropriate prompts and fill in required information; see Table 8.

Table 8. Siebel Server installation-specific information

Installation step	Installation default value	Our input value
Please specify the directory into which Siebel should be installed	/siebel	/siebel
Please enter the address of the Siebel Gateway server	erprisc2	erprisc2
Is Resonate installed for this Enterprise?	N	N
Would you like to accept these settings?	Y	Y
Enter a name for the new Siebel Enterprise	[blank]	erprisc2_ent
Provide a brief description of this Enterprise	Siebel Enterprise	Siebel Enterprise Server
Select the server database used by your organization	1	1
Enter the name of the database	[blank]	risc1db
Enter the database tableowner	siebel	siebel
Enter the name of the directory in which DB2 client is installed	[blank]	/home/db2inst1
database username	sadmin	sadmin
database user's password	sadmin	sadmin
Specify the Siebel File System directory for the Siebel Enterprise	/siebel/filesystem	/siebelfs
Specify the port to be used by the Synchronization Manager	40400	40400

Installation step	Installation default value	Our input value
Enter the root directory of your IdCentric installation	/usr/postware	/usr/postware
Do you want to apply these parameter values?	Y	Y
Enter a name for the new Siebel Server	erprisc2	erprisc2_siebel
Enter a description for this Siebel Server	Siebel Server erprisc2_siebel	Siebel Server erprisc2_siebel
Select the appropriate licensing option for eBriefing and eContent Services	1	3
Would you like to configure this section? (SynchMgr static port override)	Y	N
Would you like to configure this section? (Autostart setting)	Y	Y
Should this Siebel Server have autostart enabled	[blank]	Y
Do you want to apply these settings?	Y	Y
Do you want to start Siebel Server erprisc2_siebel?	Y	Y

d. You will get the following message:

```
Completed Siebel Server Configuration. Your DB2 environment is
not set. After this installation, you need to set your DB2
environment and run /siebel/bin/configdb2.
```

At this point, issue the following:

```
su - db2inst1
cd /siebel/bin
configdb2
```

3. Perform the following post-installation tasks:

Check the post-installation tasks in the S*iebel Server Installation Guide*.

a. Verify the Siebel Server ODBC Data Source.

b. Manage the Siebel Environment Variables.

c. Configure the Siebel Server for Automatic Start.

   Note: If you encounter any errors in this step, contact your Siebel Technical Support.

d. Establish Network Connectivity for Mobile Users.

e. Enable Siebel Server Component Groups.

4.3.4 Configuring DB2

Configure DB2 as suggested in Chapter 6, “DB2 UDB characteristics with Siebel Applications” on page 89. You may change these parameters during deployment in your environment.

1. Set DB2 Database Manager configuration parameters as suggested in Table 26 on page 99.
2. Set DB2SET parameters as suggested in Table 28 on page 105.
3. Set DB2 Database configuration parameters as suggested in Table 27 on page 101.
4. Set DB2 log settings; note the following:
   - LOGFILSIZ: the size of the log file
   - LOGPRIMARY: the number of log files to preallocate and use
   - LOGSECOND: extra log files that are allocated only if they are needed for a large transaction
5. Create DB2 bufferpools.

   The total shared memory limitation on AIX is 1.75 GB. See 9.1.3, “DB2 bufferpool management” on page 147 for details.

Table 9. Suggested default bufferpools

Bufferpool	Suggested Bufferpool Size	Pagesize
IBMDEFAULTBP	50% of available memory up to a limit of the shared memory limit/2	4 KB
BUF4KTEMP	100 to 200 MB	4 KB
BUF32KTEMP	32 MB	32 KB
BUF16K	25% of available memory up to a limit of the shared memory limit/2	16 KB

6. Create DB2 tablespaces as suggested in Table 10 and Table 11 to finish the installation. This information can be also found in *Siebel Server Installation Guide*.

Table 10. Suggested default tablespaces

DB2 tablespace name	Bufferpool name	Disk space	Description
SIEBEL_4K	IBMDEFAULTBP	500 MB	Tablespace for tables with row sizes < 4005 bytes.
SIEBEL_16K	BUF16K	120 MB	Tablespace for tables with row sizes from 4006 bytes through 16,293 bytes.
SIEBEL_4KL	IBMDEFAULTBP	20 MB	Tablespace for data stored in LONG VARCHAR columns.
SIEBEL_IDX	IBMDEFAULTBP	400 MB	Tablespace for storing the index data for the Siebel tables.

Table 11 lists the suggested default temporary tablespaces.

Table 11. Suggested default temporary tablespaces

DB2 temporary tablespace name	Bufferpool name
TEMP4K	BUFF4KTEMP
TEMP16K	BUFF16K
TEMP32K	BUFF32KTEMP

7. After the creation of the TEMP4K tablespace, you should drop TEMPSPACE1. Both TEMP4K and TEMPSPACE1 are 4 KB temporary tablespaces, and *you should choose only one*.

8. You do not have to run updating statistics (the updatestats.ksh script in the /siebel/db2udb directory), because there are no tables in the database yet

4.3.5 Installing the Siebel Database Server component

First you have to install the Database Server scripts on an AIX Siebel server. After completing this, you will run database scripts to do the following in your DB2:

- Add appropriate access to the users/group to access Siebel Database objects.

- Create tables.
- Create indexes.
- Create triggers and stored procedures.
- Create seed data.

Follow the steps below.

1. Pre-installation steps:
   a. The Siebel Gateway Server and Siebel Application server must run successfully.
   b. DB2 has to be properly configured and running.
2. As user `sadmin`, install the Database Server scripts:
   a. Run the `ksh` and `umask 027` commands.
   b. Run the following command:

```
./install_dbsrvr -d /usr/tmp/dbsrvr.log
```

This will generate an installation log in the /usr/tmp directory so you can review the installation process.

Table 12. Siebel Database Server installation-specific information

Installation screen	Installation default value	Our input value
Please specify the directory into which Siebel should be installed	/siebel	/siebel
Would you like to accept these settings?	Y	Y

   c. Verify that the installation procedure created all directories and files. The directory structure should look as follows:

```
    bin
    files
    db2udb
       enu
       siebproc
          aix
          hpux
          solaris
          winnt
       ctiproc
          aix
          hpux
          solaris
          winnt
```

d. Edit the `grantusr.sql` file.

> **Note**
>
> All user names in grantusr.sql must be in lowercase.

e. Run grantusr.sql:

```
db2 connect to risc1db user db2inst1 using ibmdb2
db2 -vf /siebel/db2udb/grantusr.sql
```

f. Install the Stored Procedures and UDFs; for details, see *Siebel Server Installation Guide:*

- ftp `siebel.tar` from the /siebel/db2udb/siebproc/aix directory on the Siebel Application server to /home/db2inst1 on the DB2 UDB Siebel database server.
- Then untar the `siebel.tar` file by running the `tar -xf siebel.tar` command.
- From the /home/db2inst1/siebel directory on the DB2 UDB Siebel database server, run the `./installsiebel` command to install the Siebel stored procedures.

g. Test the ODBC Data Source. Run the following commands from the /siebel/bin directory:

- `./odbcsql`
- `set source siebsrvr_erprisc2_ent`

- `login sadmin/sadmin`
- `select name from sysibm.systables`

3. Execute the Database Server Installation Script:

   a. Edit the install.ksh script according to Table 13.

   Table 13. Install.ksh installation-specific information

Variable	Our values
SRC_USR	sadmin
SRC_PSWD	sadmin
SRC_TBLO	siebel
SRC_TBLO_PSWD	siebel
DBSRVR_ROOT	/siebel
DATA_AREA	“SIEBEL_4K”
INDX_AREA	“SIEBEL_IDX”
LONG_AREA	“SIEBEL_4KL”
DB16K_AREA	“SIEBEL_16K”
DB_LANG	enu
DB_UNICODE_FLG	N
DBSERVER_OS	aix
SIEBEL_HOME	/siebel

   b. Run install.ksh.

   c. Review error.rtf and log files for any errors.

   d. When calling the updatestats utility, you will see ODBC error messages like the following:

```
{call siebstat ('siebel','IDOC_CUST_KNB1H','SQL_STATS_ALL')}
ODBC error S1000 in SQLExecDirect:
[IBM][CLI Driver][DB2/6000] SQL2306N  The table or index
"IDOC_CUST_KNB1H" does not exist.
(native error -2306).
```

   Error messages like this, which have names starting with IDOC_, V_, and W_, are acceptable because they are not created by the default Siebel installation. See the Siebel Release Notes for more detailed information.

4. Import the Siebel Repository:

   a. Edit the *imprep.ksh* script according to Table 14.

   Table 14. Imprep.ksh installation-specific information

Variable	Our values
SRC_USR	sadmin
SRC_PSWD	sadmin
SRC_TBLO	siebel
ODBC	siebsrvr_erprisc2_ent
REPOS_NAME	“Siebel Repository”
DBSRVR_ROOT	/siebel/db2udb
DATABASE_PLATFORM	DB2udb
DB_LANG	enu
SIEBEL_HOME	/siebel

   b. Run *imprep.ksh*.

   c. Review the *error.rtf* and log files for any errors.

5. Populate the Siebel File System:

   a. Copy all files from the /siebel/files subdirectory to the Siebel file system directory /siebelfs.

   b. Verify that the files are where they need to be.

4.4 Post-installation steps

After successfully installing the above, ensure that a full DB2 database offline backup is taken so that you can restore the database if there is a need to.

We strongly recommend that you perform a RUNSTATS again on the tables in the Siebel database (either using updatestats.ksh or on individual tables by the DBA), and then perform a RUNSTATS on the System Catalog tables in the Siebel database. There are a number of tables that you should *never* do runstats on; these tables are located in the loadstats.sql file from Siebel. Siebel sets some statistics as it needs, and if you run DB2 runstats later on these tables, you will overwrite this information and have performance issues. Refer to 9.4, “The runstats utility vs. updatestats” on page 156 for more information.

Important

If statistics are not updated, then Siebel Tools will perform poorly since DB2 statistics indicate zero records in most tables used to hold Repository information, but some tables (e.g. S_COLUMN) have 59,000 records after imprep.ksh.

Then, install the Siebel client and test it. Details are covered in Chapter 5, “Client installation” on page 67.

Chapter 5. Client installation

This chapter provides an overview of the minimum hardware and software requirements for different types of clients, and also covers the installation and configuration of the three of the Siebel client types.

The client type installations described in this chapter are:

- Connected Clients
- Mobile Clients
- HTML Clients

5.1 Hardware and software requirements

Table 15 lists the minimum hardware and software requirements needed for various client types.

Table 15. Hardware and software requirements

Client	Hardware	Software
Connected Client	Intel-compatible PC with: 266 MHz CPU 48 MB of RAM for Windows 95/98/2000 64 MB for Win NT 4.0	Adobe Acrobat Reader; Microsoft Office 97 or 2000 or Lotus WordPro 9.5 for Windows; Microsoft Access and Text ODBC Drivers V3.5 or higher; Microsoft Explorer 4.01 or 5.01, or Netscape 4.6 or 4.7
Mobile Client	Intel-compatible PC with: 266 MHz CPU 48 MB of RAM for Windows 95/98/2000 80 MB for Win NT 4.0	
HTML Thin Client	Intel, Apple or UNIX-compatible PC with: 133 MHz CPU 24 MB of RAM	Web browser: Microsoft Internet Explorer 4.01, 5.01 or Netscape Communicator 4.6, 4.7
Thin Client for Windows	Intel-compatible PC with: 133 MHz CPU 32 MB of RAM for Windows 95/98/2000 48 MB for Win NT 4.0	
Java Thin Client	Sun SPARC-compatible PC with: 333 MHz CPU, 64 MB RAM	Sun Solaris 2.6

Client	Hardware	Software
Siebel WML Thin Client	Wireless-enabled client device	WAP 1.0 browser or above

5.2 Connected Clients and Mobile Clients

The default client for a Siebel application is the Siebel Connected Client.

5.2.1 Pre-installation steps

Before running the installation program, make sure the following tasks are complete:

- Choosing a directory name:

  The default value is c:\sea\client.

  Note: You cannot install different Siebel components in the same directory. Therefore, if you will be installing more than one component on the same machine, you should determine your directory-naming convention *before* beginning the installation process.

- Verifying Siebel client prerequisites:

  The client must have the appropriate prerequisites, as listed in Table 16.

  Table 16. Client prerequisites

Prerequisites	
Hardware	Intel-compatible PC with: 266 MHz CPU 48 MB of Ram for Win95/98/2000 64 MB for Win NT 4.0
Software	Microsoft Access and Text ODBC Drivers V3.5 or higher; Microsoft Explorer 4.01 or 5.01 or Netscape 4.6 or 4.7; Microsoft Office 97 or 2000 or Lotus WordPro 9.5 for Windows

- Installing the DB2 UDB Runtime Client:

  Install the DB2 UDB Runtime Client V6.1 with FixPak 3SA and configure it to connect to the database server that you have chosen to run the DB2 database.

- Installing included third-party software:

  - Adobe Acrobat Reader (required)

- Fulcrum SearchServer (optional)

- Installing Fulcrum SearchServer Desktop

  For more information about this topic, refer to the *Siebel Client Installation and Administration Guide, C*hapter 4.

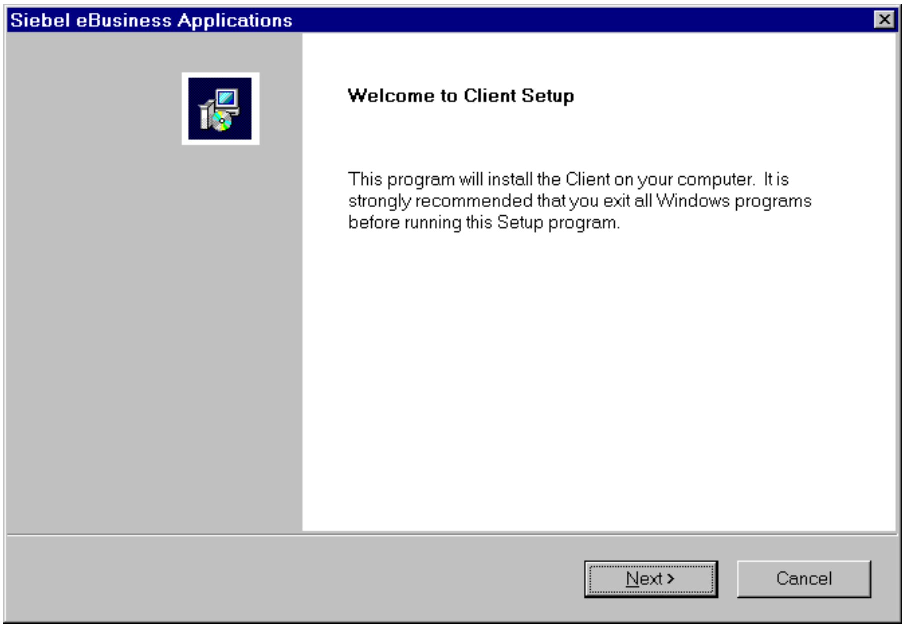
5.2.2 Installation process

The installation steps are as follows:

1. Run the \sea\setup.exe file.

> **Note**
>
> If no Siebel software has been previously installed, the Installation Wizard is going to reboot the machine. After rebooting, launch the setup.exe file again.

2. From the Welcome dialog box, click **Next**.



Welcome to Client Setup

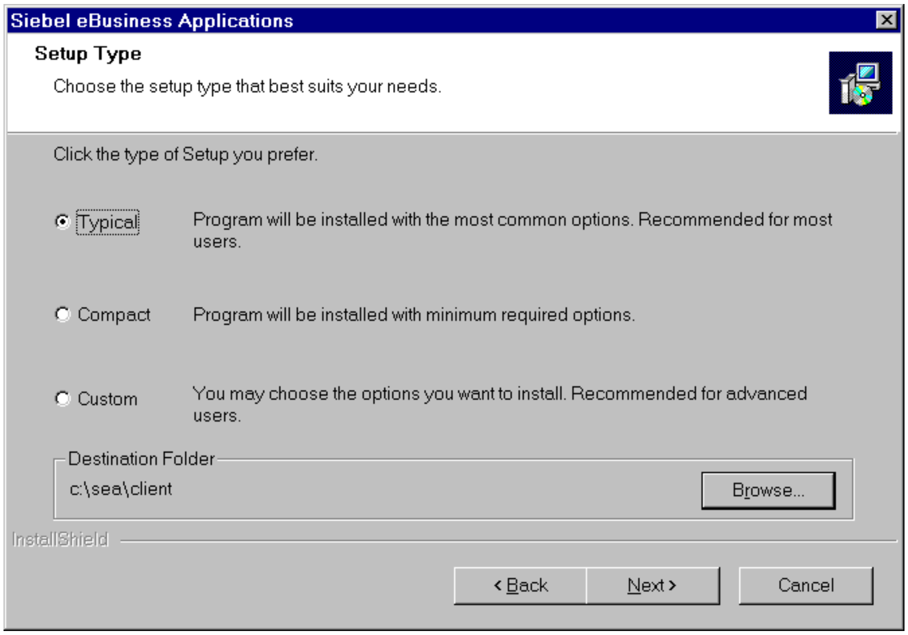
This program will install the Client on your computer. It is strongly recommended that you exit all Windows programs before running this Setup program.

Figure 35. Welcome dialog box

3. In the Setup Type dialog box, select the type of Siebel installation to perform.

   Note: In our case, we chose the Typical Setup. If you select the Compact Setup, the dialog boxes and the information you have to provide are the same. However, with Custom Setup, you are presented with one new box before the DB identification dialog box, where you have to check the components to install; the other steps will be the same.

   If you intend to use the Siebel client to access/administer the Siebel *e*Business Applications using the Server Administration Views, then you must select Custom as the type of setup and then explicitly select the Server Administration component.



Siebel eBusiness Applications

Setup Type

Choose the setup type that best suits your needs.

Click the type of Setup you prefer.

● Typical — Program will be installed with the most common options. Recommended for most users.

○ Compact — Program will be installed with minimum required options.

○ Custom — You may choose the options you want to install. Recommended for advanced users.

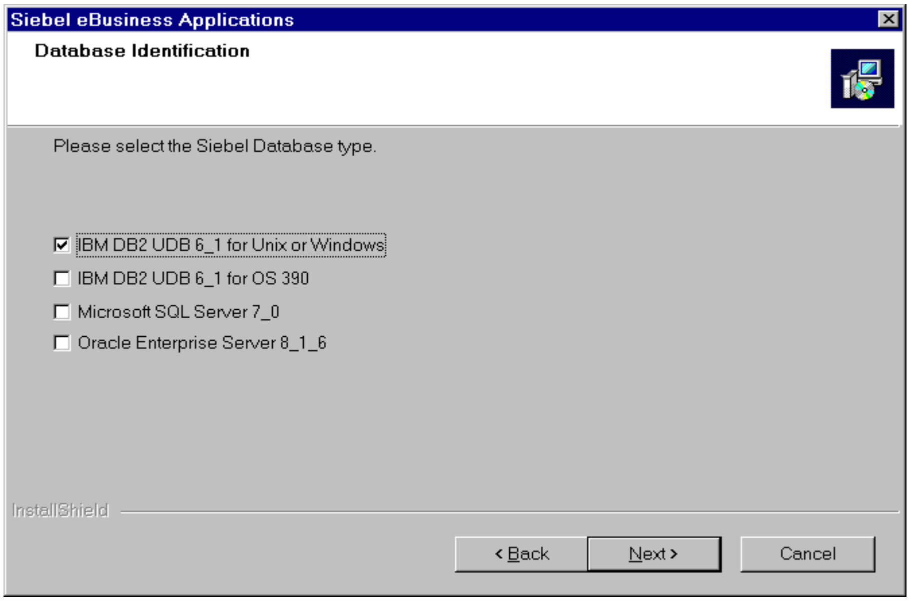
Destination Folder: c:\sea\client

Figure 36. Setup Type dialog box - choosing a type

Tip

To save substantial disk space, you can choose Custom Setup and deselect the sample database, if you do not need this data.

4. Verify that the installation directory listed is correct:
   - Click **Next** to select the default directory: c:\sea\client:
   - Click **Browse** to select a different destination directory.
5. In the DB Identification dialog box, select the type of database: **IBM DB2 UDB 6_1 for Unix or Windows**, then click **Next**.



Siebel eBusiness Applications

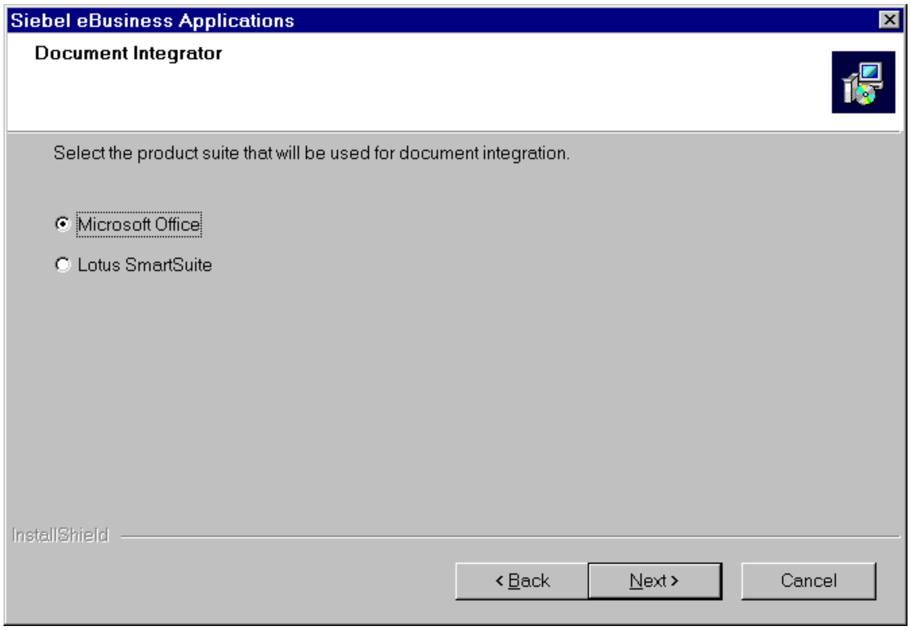
Database Identification

Please select the Siebel Database type.

☑ IBM DB2 UDB 6_1 for Unix or Windows
□ IBM DB2 UDB 6_1 for OS 390
□ Microsoft SQL Server 7_0
□ Oracle Enterprise Server 8_1_6

Figure 37. DB Identification dialog box

6. In the Document Integrator dialog box, select the product suite that you want to use for document integration and click **Next**. The options are:
   - Microsoft Office
   - Lotus SmartSuite


Siebel eBusiness Applications

Document Integrator

Select the product suite that will be used for document integration.

● Microsoft Office

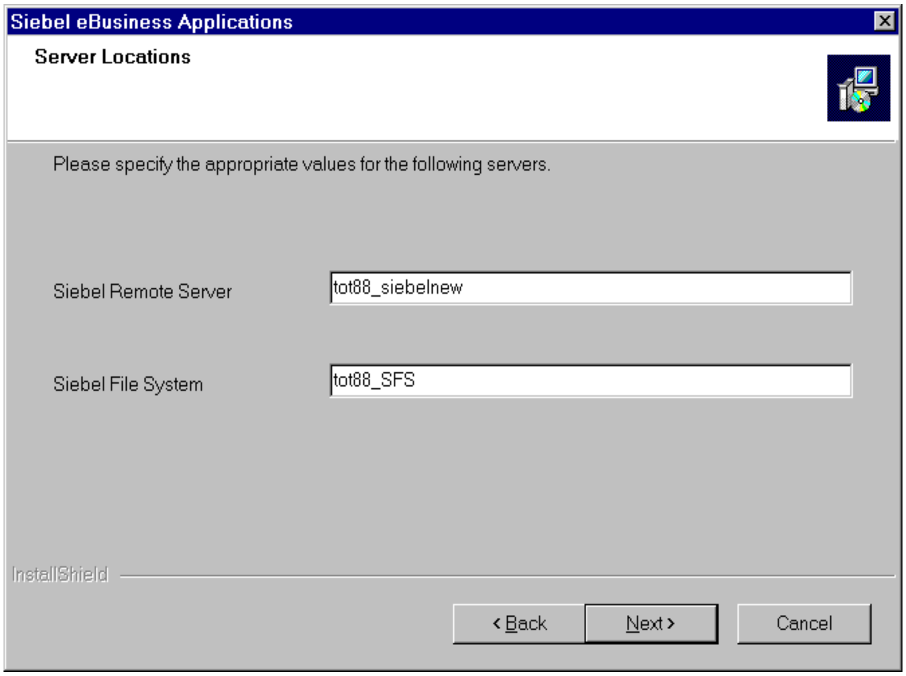
○ Lotus SmartSuite

Figure 38. Document Integrator dialog box

Important

If you are installing your client with Lotus SmartSuite, the ODBC Drivers are not going to be updated automatically. You then have to provide them.

7. In the Server Locations dialog box, enter your Siebel Remote Server and File System connectivity information.



Siebel eBusiness Applications

Server Locations

Please specify the appropriate values for the following servers.

Siebel Remote Server: tot88_siebelnew

Siebel File System: tot88_SFS

Figure 39. Server Locations dialog box

In our case, we entered the following information:

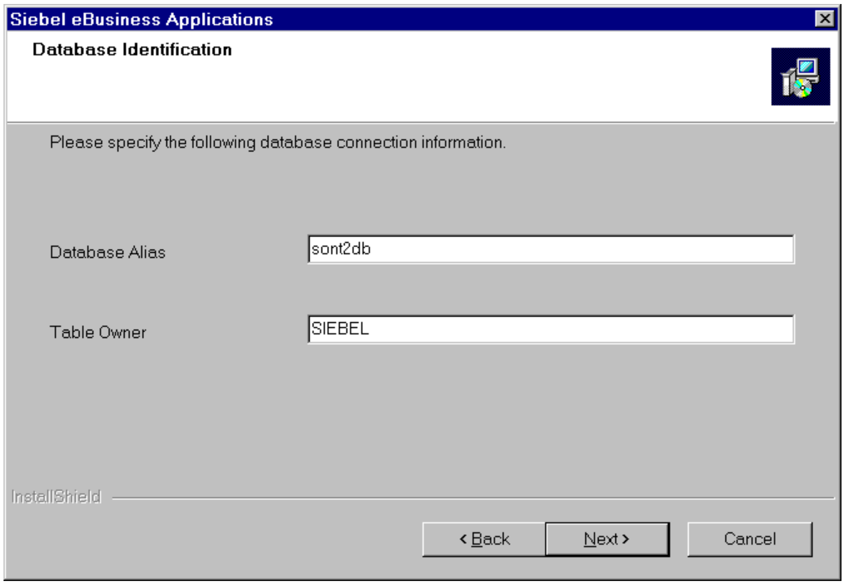
Table 17. Input values for Server Locations dialog box

Installation screen	Installation variable	Our input value
Server Locations dialog box	Siebel Remote Server	tot88_siebelnew
	Siebel File System	tot88_SFS

8. In the Database Identification dialog box, enter the following information:

   - Database Alias
   - Table owner

   Then click **Next**.



Please specify the following database connection information.

Database Alias: sont2db

Table Owner: SIEBEL

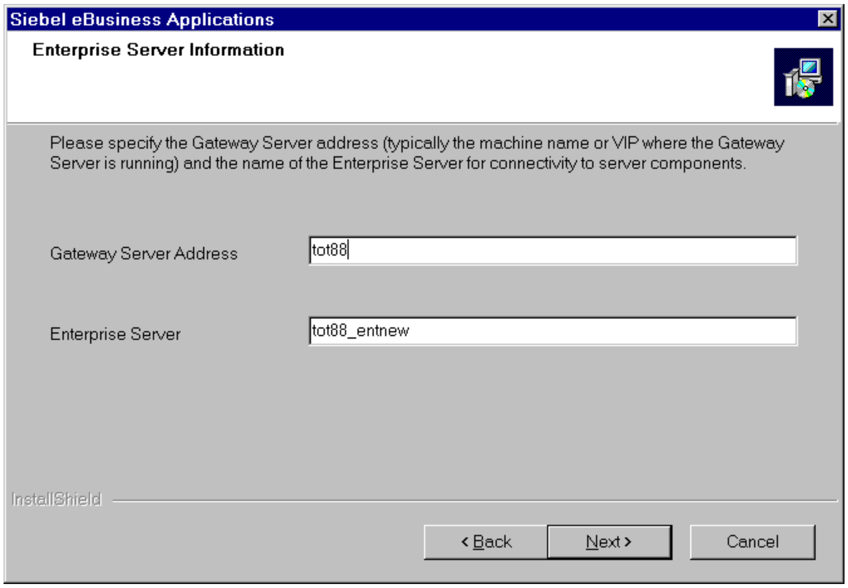
Figure 40. Database Identification dialog box - database connection

In our case, we entered the following information:

Table 18. Input values for the Database Identification dialog box

Installation screen	Variable	Our input value
Database Identification dialog box	Database Alias	sont2db
	Table owner	siebel

9. In the Enterprise Server Information dialog box, specify the Gateway Server Address and the Enterprise Server to which this client will connect for administration.



Please specify the Gateway Server address (typically the machine name or VIP where the Gateway Server is running) and the name of the Enterprise Server for connectivity to server components.

Gateway Server Address: tot88

Enterprise Server: tot88_entnew

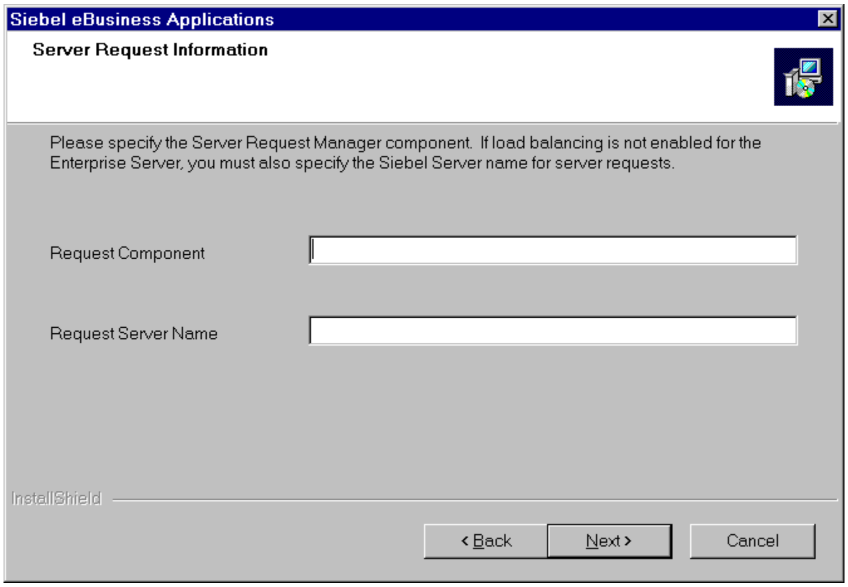
Figure 41. Enterprise Server Information dialog box - address

In our case, we entered the following information:

Table 19. Input values for Enterprise Server Information dialog box

Installation screen	**Variable**	**Our input value**
Enterprise Server Information dialog box	Gateway Server Address	tot88
	Enterprise Server	tot88_entnew

10.In the Server Request Information dialog box, enter the information that dedicated clients will use to invoke the Server Request Manager component for dynamic assignment and other interactive operations. Then click **Next**.


Siebel eBusiness Applications

Server Request Information

Please specify the Server Request Manager component. If load balancing is not enabled for the Enterprise Server, you must also specify the Siebel Server name for server requests.

Request Component

Request Server Name

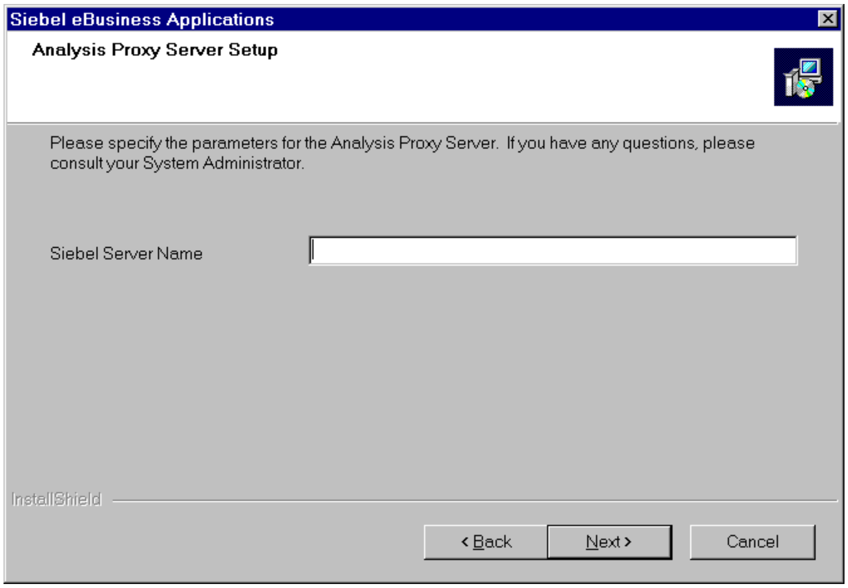
Figure 42. Server Request Information dialog box

We did not enter any information because we did not intend to use the Server Request Manager.

11. If you are using Siebel Marketing, do the following:

    In the Analysis Proxy Server Setup dialog box, enter the machine name or IP address of the Siebel Server on which the Marketing Analysis Proxy Manager server component is operating, then click **Next**.

    For more information, see the *Siebel Marketing Guide.*



Siebel eBusiness Applications

Analysis Proxy Server Setup

Please specify the parameters for the Analysis Proxy Server. If you have any questions, please consult your System Administrator.

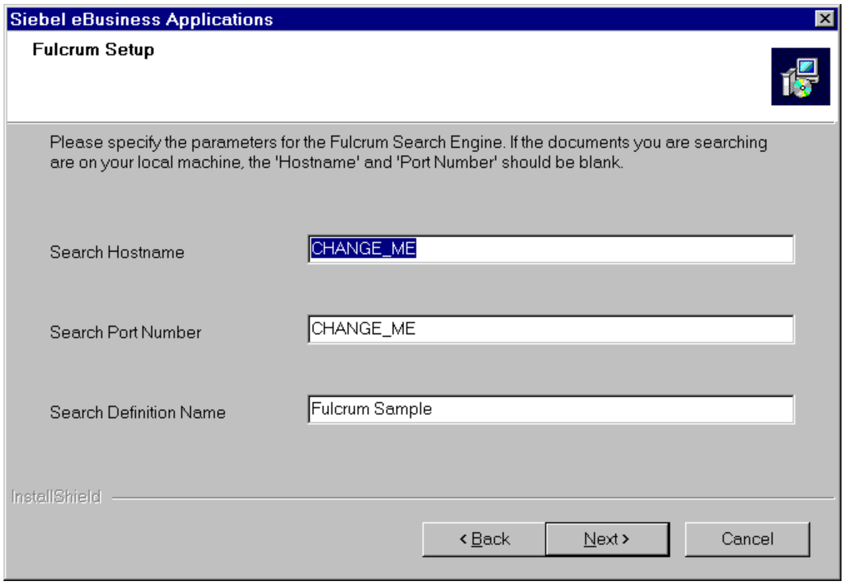
Siebel Server Name

Figure 43. Analysis Proxy Server Setup dialog box

We did not do this step.

12. If you are not using Fulcrum, accept the default values shown in the Fulcrum Setup dialog box.

    If you are using Fulcrum, enter the following information: Search Hostname, Search Port Number, and Search Definition Name.



Siebel eBusiness Applications

Fulcrum Setup

Please specify the parameters for the Fulcrum Search Engine. If the documents you are searching are on your local machine, the 'Hostname' and 'Port Number' should be blank.

Search Hostname: CHANGE_ME

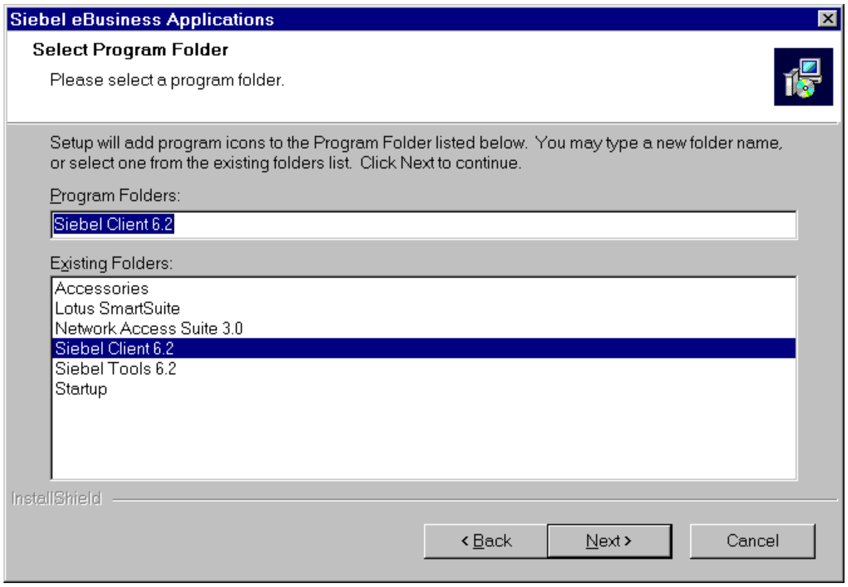
Search Port Number: CHANGE_ME

Search Definition Name: Fulcrum Sample

Figure 44. Fulcrum Setup dialog box

We did not do this step.

13.In the Select Program Folder dialog box, enter the name of the program folder that will contain your Siebel icons in the Select Program Folder dialog box.



Siebel eBusiness Applications

Select Program Folder

Please select a program folder.

Setup will add program icons to the Program Folder listed below. You may type a new folder name, or select one from the existing folders list. Click Next to continue.

Program Folders: Siebel Client 6.2

Existing Folders:

- Accessories
- Lotus SmartSuite
- Network Access Suite 3.0
- Siebel Client 6.2
- Siebel Tools 6.2
- Startup

Figure 45. Select Program Folder dialog box

In our case, the information was as follows:

Table 20. Input values for Select Program Folder dialog box

Installation screen	Variable	Our input value
Select Program Folder dialog box	program folder	Siebel Client 6.2 (by default)

14. Click **Next**, and a status bar will show the progress of the installation.
15. Click **Next** after checking the Event Log dialog box.
16. Click **Next** after checking the Registry Log dialog box.
17. Click **Finish** in the Setup Complete dialog box.

Note

For more detailed information, see the *Siebel Client Installation and Administration Guide.* Also see the *Siebel Release Notes* in case any problems have occurred during the installation. Some anomalies are detailed there.

5.2.3 Post-installation tasks

After completing the installation, perform these tasks:

1. Verify the Siebel Client directory structure.

   The structure should be as follows:

```
sea
   client
      actuate
         afc
         bin
         cache
      bin
      help
      idcentric
      local
         files
         inbox
         outbox
      locale
      log
      objects
      msgtempl
      packager
         install
      reports
         enu
      sample
         dnb
         files
         inbox
         outbox
         si
      search
         data
         index
         scripts
      temp
      upgrade
```

Figure 46. Structure of a Siebel client directory

2. Verify the Siebel Client icons.

   The Siebel installation creates icons in the Siebel program folder for the modules you have chosen. These include the following:

   - Siebel Packager
   - Siebel Call Center and Siebel Call Center Demo
   - Siebel *e*Business Applications Help
   - Siebel Marketing
   - Siebel Remote
   - Siebel Sales and Siebel Sales Demo
   - Siebel Service and Siebel Service Demo

3. Verify successful installation by connecting to the Sample Database.

   To do this, you must have chosen the Typical or Custom Siebel Application installation option. After one of the two options has been selected, then double-click on one of the Demo icons.

4. Verify Siebel Client ODBC Data Sources.

> **Note**
>
> In most cases, the ODBC Data Sources have to be created manually. Use the DB2 Command Line Processor to catalog an ODBC data source or the DB2 Client Configuration Assistant to accomplish this (refer to *DB2 for Windows NT Quick Beginnings* for information). Then, before continuing, verify that the creation was successful.

5.2.4 Configuring Connected Clients and Mobile Clients

This section describes how to configure Connected Clients and Mobile Clients after installing the client code.

5.2.4.1 Configuring the Connected Client

To log on for the first time to your Siebel application and connect to the server database, follow these steps:

1. Double-click the Siebel icon.
2. Click **Ok** to agree to the terms (the Terms dialog box only appears the first time you log on).

When the Siebel dialog box appears, do the following:

1. Specify `sadmin` for the User name and Password fields.

2. In the Connect to: field, choose a database from the picklist (generally, as an administrator, you will choose Server).
3. Click **Ok**.

> **Note**
>
> The first time you log on to the Server database, the system prompts you to enter a license key number. You can find the number in the letter provided with the product; enter it and press **Ok**.
>
> If there is an over allocation of memory due to DB2 configuration parameters and buffer pool requirements, you might encounter errors when adding a license key number. This can be remedied by reducing the memory requirements of the database.

Once you have entered the application, you can set up other users. Follow these steps:

1. Create the users at the operating system level.

   Make sure that these users are in the sse_role group.
2. Set these users up as Siebel application users:
   a. Choose **Screens -> Application Administration -> Employees -> Employees**.
   b. Choose **Edit** - **Add New Record**.
   c. Complete the fields.

   > **Note**
   >
   > The Password field is automatically filled out by the system with the user's operating system password. However, you must ensure that the logon name in both cases is the same.

   d. Repeat these steps for all users you need to add.
   e. Associate every user you have created with a responsibility and a position.

   To complete this association task, follow the steps described in the *Siebel Application Administration Guide,* Chapter 2.

   For detailed information, refer to the *Siebel Application Administration Guide.*

5.2.4.2 Configuring the Mobile Clients

Important

If the Siebel Application is installed for the first time, the Generate New Database component must be run first. To do that, follow the *Siebel Server Administration Guide*, Chapter 4*.*

To configure the Mobile Client, the following points must be verified:

- The client is created at the operating system level.
- The client has been set up as a Siebel Application user.
- The client has also been registered as a Mobile Client on the Siebel Server:
  a. Go to **Screens -> Siebel Remote Administration -> Mobile Clients.**
  b. Ensure that the client you are going to extract the database for is already recorded in this screen.

Once this has been checked, the database extract can be run. Follow these steps in the Siebel Server:

1. Verify that the Siebel Server Manager is running:

   Screens -> Server Administration -> Servers -> Server Tasks

2. Click **New** on the Server Tasks applet.
3. Click the drop-down arrow in the component field, and select **Database Extract** from the Pick Assigned Component dialog box.

Note

If this option of Database Extract does not appear, follow the steps described in the *Siebel Server Administration Guide*.

4. Click **Parameters**, and set the Client name parameter to the name of your client user.
5. Close the dialog box.
6. Click **Start**.

If more details are needed, refer to the *Siebel Client Installation and Administration Guide*, Chapter 5.

After extracting the database successfully, ensure that the Transaction Processor, the Transaction Router and the Transaction Merger are running. This can also be followed in the *Siebel Client Installation and Administration Guide*, Chapter 5.

After these steps are completed, the client can launch the Siebel application by launching Siebel Remote.

5.3 HTML Thin Clients

Siebel is focusing on HTML Thin Clients because of their facilities for users and because of the growing importance of the Internet.

5.3.1 Preinstallation steps

To have a Siebel HTML Thin Client, you must install the .COM application. Before installing, make sure these tasks are complete:

- Verifying Thin Client platforms: see *Siebel System Requirements and Supported Platforms.*
- Planning the installation.
- Verifying the Siebel Enterprise Server; see the *Siebel Server Installation Guide.*
- Verifying Web server requirements, shown in Table 21:

Table 21. Web server requirements

Web server	**Siebel Server**	**Hardware and operating system**
Microsoft IIS 4.0	Windows NT 4.0	Intel-compatible hardware running on Windows NT Server 4.0 with ServicePack 4, 5 or 6a
IBM HTTP Server 1.3.12	AIX 4.3.3	IBM RS/6000 hardware running on IBM AIX 4.3.3

These steps are described in *Siebel Client Installation and Administration Guide, C*hapter 15.

5.3.2 Installation process

The HTML client does not need any installation process, except for the browser and the network connectivity installation. Therefore, all the necessary components to run (such as client type) reside on the server side; see the *Siebel Server Installation Guide.*

These steps are as follows:

> **Note**
>
> Before starting the installation of the .COM application, the Web Server has to be installed on the machine where your application will reside. Follow the vendor's instructions to complete this task.

1. Run: \eappweb\setup.exe
2. In the Welcome screen, click **Next**.
3. Follow the on-screen instructions, then click **Next**.
4. Specify the destination directory for the Web Engine and .COM applications:
5. Click **Next** to accept the default values.
6. Click **Browse** to specify a different destination directory.
7. In the Load Balancing screen, indicate whether or not you use the Central Dispatch load balancing feature, then click **Next**.
8. In the Gateway and Enterprise server information screens, type the values according to your Deployment Planning Worksheet. Then click **Next**.
9. In the Siebel Server Information screen, type the logical name of your Siebel Server (not the machine name). Then click **Next**.
10. In the Connection Protocol screen, choose the type of protocol that Siebel should use to connect to the Application Object Manager:

    - TCP IP
    - HTTP

    Then click **Next**.
11. In the Encryption Type screen, choose whether or not you want to encrypt communications to the Application Object Manager. Then click **Next**.
12. In the Compression Methods screen, choose the data compression you prefer when the .COM applications communicate to the Applications Object Manager. Then click **Next**.
13. Type the login and password for employees. Then click **Next**.

    a. In the Anonymous User Account screen, type the login and password for contact. Then click **Next**.

b. In the Error Level for Logging screen, specify the level of operational error logging for the .COM application:

   - Errors only
   - All (errors and warnings)
   - None

   Click **Next**.

c. In the Event Log screen, review the results of the installation. Then click **Next**.

d. Click **Finish** in the Setup Complete screen.

Note

For further information about this procedure, consult the *Siebel Server Installation Guide.*

5.3.3 Post-installation tasks

Once the installation has been completed, perform the following tasks if your environment requires any changes to the default settings:

- Review the installation directories.
- Edit the .COM applications configuration file.
- Verify that these services are running by restarting them:
  - Microsoft IIS Admin Service
  - World Wide Web Publishing Service
  - Siebel Gateway Name Server
  - Siebel Server

Note

Consult the *Siebel Server Installation Guide* if needed.

5.3.4 Configuring the .COM application

This process has seven phases:

1. Create .COM applets and views by using the Dedicated Client.
2. Associate each applet and view with the correct template.
3. Modify templates as needed to create a corporate image.

4. Establish mapping between controls and templates.
5. Compile the repository changes into a .SRF file.
6. Test the application.
7. Deploy the application.

An overview of these steps is found in the *Configuring Siebel .COM Applications* guide.

5.3.5 Optimizing the performance

To optimize the performance of your application, follow these guidelines:

- In Siebel Tools, create a separate application definition for the Web application instead of using an existing dedicated-client application.
- In your HTML Thin Client application definition, follow these guidelines:
  - Reduce the number of objects.
  - Do not define any view to have a list and a form accessing the same business component.
- In the templates, avoid duplicating information on the same page.
- In the templates and other Web pages, minimize the use of multimedia objects (graphics, audios, etc).

Chapter 6. DB2 UDB characteristics with Siebel Applications

This chapter describes the characteristics of DB2 UDB components of Siebel *e*Business Applications 2000 running on AIX and NT platforms. In our environment we focused on Siebel horizontal applications such as Siebel Call Center, Siebel Sales, etc. If you are using Siebel vertical applications, the data may differ.

6.1 Siebel database characteristics

After the default installation of the database server component of the Siebel Server, many DB2 UDB objects are created as shown in Table 22.

Table 22. DB2 object statistics of Siebel Applications after default installation

DB2 Objects	No. of Occurrences	Remarks
Databases	1	Single DB2 database per Siebel Enterprise Server
Bufferpools	4	
Tablespaces	7	4 tablespaces for Siebel data, 3 tablespaces for temporary usage
Tables	1,377	
Columns	52,550	
Indexes	7,327	
Views	4	For Siebel internal processes
UDFs	1	Called NEXTSEQ
Stored Procedures	5	

After the default installation, DB2 data occupies about 1.2 GB of disk space, which is about 900 MB of disk for backup. For the DB2 log, you should prepare at least 4 GB of disk space.

Note: the Siebel Repository consumes approximately 150 MB of disk space after being installed/loaded into the database.

6.1.1 Tables

Tables for Siebel Applications are highly normalized (no repeating groups are permitted) in order to provide easily customizable solutions.

There are about 1400 tables in a database for Siebel Applications. The biggest tables or high activity tables are different at each installation, and the size or activity of the tables may be the barometer of importance. Refer to 9.1.1.2, “Tables” on page 146 for details about important Siebel tables in terms of activity.

As example, Table 23 lists the 20 biggest tables of a customer environment.

Table 23. An example of the 20 biggest Siebel tables

Table name	Tablespace name	Cardinality	Fpages
S_POSTN_CON	S_POSTN_CO	12893971	527559
S_ACCNT_POSTN	S_ACCNT_PO	4152221	219159
S_EVT_ACT	S_EVT_ACT_	3939168	162765
S_EVT_ACT_X	S_EVT_ACT_	2328152	291195
S_ORG_TERR	S_ORG_TERR	2090550	94539
S_OPTY_POSTN	S_OPTY_POS	1642705	109898
S_DNB_CONTACT	SIEBEL_4K	953311	45960
S_DNB_CON_MRC	SIEBEL_4K	948726	24977
S_DNB_ORG_SIC	SIEBEL_4K	872005	22609
S_CAMP_CON	SIEBEL_4K	765137	61985
S_OPTY_TERR	SIEBEL_4K	717986	27690
S_CONTACT	S_CONTACT_	654406	129572
S_EXP_ITEM	SIEBEL_4K	646247	109829
S_DNB_ORG	S_DNB_ORG_	540615	180383
S_PRSP_CONTACT	SIEBEL_4K	512191	70260
S_CALL_LST_CON	SIEBEL_4K	510121	16456
S_CONTACT_X	SIEBEL_4K	500945	69441
S_COMM_DTL	SIEBEL_4K	492481	32189
S_COMMUNICATION	SIEBEL_4K	485084	29851
S_ADDR_ORG	SIEBEL_4K	443537	31081

Note: Fpages are the total number of pages the table data occupies.

6.1.1.1 Types of Siebel tables

Siebel *e*Business Applications categorize their tables into these types:

Data tables

These tables store user business data, administrative data, seed data, and transaction data. They are populated and updated through Siebel Applications and Siebel Server processes such as EIM or Assignment Manager. Their names are prefixed with S_.

- Base tables

  These are tables containing columns that are defined and used by Siebel Applications. They provide the essential columns for a business component specified in the table property of that business component.

- Intersection tables

  These are tables in Siebel Applications that implement a many-to-many relationship between two business components. They may also hold intersection data columns, which store information pertaining to each association. Intersection tables provide the data structures for association applets.

Interface tables

These tables denormalize a group of base tables into a single table that external programs can interface to. They are a staging area for exporting and importing of data, and are used only by the Siebel Enterprise Integration Manager (EIM). EIM is used to import initial data for populating one or more base tables, and also for ongoing data import/export. Their names are prefixed with EIM_.

Repository tables

These tables contain the object definitions that specify one or more Siebel Applications regarding:

- The client application configuration
- The mapping used for importing and exporting data
- Rules for transferring data to mobile clients

6.1.1.2 Sizes of the tables

In this section, we show you how to size these tables. Your statistics should be updated by running the runstats utility in order to get the right information. You can get the cardinality of the tables by running:

```
db2 select card, name from sysibm.systables order by card desc
```

To get the size of a table, you can use the following methods:

1. The Estimate Size utility (accessible from the DB2 Control Center)

   You can estimate the amount of storage space required for new or existing tables or indexes by invoking the Estimate Size dialog. Invoke this dialog by selecting individual tables and indexes and clicking the right mouse button on them.

   The size is estimated on the definition of the particular table and its dependent indexes. The estimate is the projected amount of storage space that would be used when the table has a given number of rows. The minimum and maximum space is also estimated based on the smallest and largest size of variable length fields.

   When invoked on a table or an index, the Estimate Size dialog is prefilled with the specifications of the table, and contains numbers relating to the table and all of its dependent indexes. When you click the Refresh push button, the estimated size, minimum size, and maximum size are updated based on the number you enter in the New total number of rows and New average row length fields.

2. Simple calculation for user tables:

   ```
   number_of_records * maximum row size
   ```

   where `number_of_records` is equal to the cardinality (CARD) of the table obtained in the previous SELECT statement from SYSIBM.SYSTABLES.

   The maximum row size of each table can be obtained by using (we assume that the Siebel schema is called SIEBEL):

   ```
   select cast(a.tabname as char(35)) as table, sum(a.length) as rowlen,
   sum(a.length*b.card) as tablesize
   from syscat.columns a, sysibm.systables b
   where a.tabschema = 'SIEBEL' and a.tabname=b.name
   group by a.tabname order by a.tabname;
   ```

   This formula provides only an estimate. The accuracy of the estimate is lower if the length of records varies due to fragmentation and overflow records.

6.1.2 Columns

Each Siebel table has multiple columns to store user and system data. System data columns are maintained by Siebel Applications and tasks. Some tables have less than 10 columns, while others have more than 200 columns.

Primary key

The Primary key column uniquely identifies each row in a Siebel table. Its name, by convention, is ROW_ID for every Siebel database table.

Column ROW_ID contains a Siebel-generated identifier (maintained by Siebel Applications) that is unique across all tables and mobile users. It is the basis for Siebel Applications maintaining referential integrity.

Column type

In Siebel tables, several types of columns are used, such as:

- VARCHAR

  Many columns use the VARCHAR type; therefore, it is not easy to estimate the average row length at the beginning.

- CHAR

  Some values for this column type represent attribute, flag, etc.

- DECIMAL

  Most numeric values use DECIMAL type rather than INTEGER.

- TIMESTAMP

  The values are the timestamp of when records are created or updated, or when some activities are started or ended.

6.1.3 Indexes

An index is a list of the locations of rows, sorted by the contents of one or more specified columns. Indexes are typically used to speed up access to a table.

In the Siebel environment, there are on average five indexes per Siebel table, but some have many more (for example, the S_EVT_ACT table has 44 indexes). More than 35 tables have between 20 and 46 indexes.

There are some unique key indexes, many duplicate key indexes and *no* clustering indexes for Siebel *e*Business Applications.

6.1.4 Views

Four views are created during a default installation of Siebel Applications. They are used for Siebel internal processes, such as remote administration or synchronization of a remote client, etc. They are:

- S_PERSON_V
- S_TXN_LOG_ITEM_V

- S_TXN_LOG_NODE_V
- S_TXN_LOG_OPER_V

6.1.5 Tablespaces

Before the Siebel installation process, seven tablespaces should be created for Siebel tables and indexes as shown in Table 24. This information can be also found in *Siebel Server Installation Guide*.

Table 24. Default DB2 tablespaces for Siebel

Tablespace name	**Bufferpool**	**Description**
SIEBEL_4K	IBMDEFAULTBP	For Siebel tables with a row length less than 4K
SIEBEL_16K	BUF16K	For Siebel tables whose row length is greater than 4K (this does not have to be 16K)
SIEBEL_4KL	IBMDEFAULTBP	For Siebel tables with LONG VARCHAR columns
SIEBEL_IDX	IBMDEFAULTBP	For Siebel indexes
TEMP4K	BUFF4KTEMP	4K Temporary tablespace
TEMP16K	BUFF16K	16K Temporary tablespace
TEMP32K	BUFF32KTEMP	32K Temporary tablespace (It may not be required depending on your workload.)

A tablespace can be stored in either system-managed space (SMS) or database-managed space (DMS). For an SMS tablespace, each container is a directory in the file system of the operating system, and the operating system's file manager controls the storage space. For a DMS tablespace, each container is either a fixed-size preallocated file, or a physical device such as a disk, and the database manager controls the storage space.

Siebel Applications use DMS tablespaces for storing data and indexes, and SMS tablespaces for temporary tablespaces. Most of the tables use 4 KB tablespaces, and some long-length-row tables use 16 KB tablespaces. 16 KB and 32 KB temporary tablespaces are used for queries that are executing against tables in tablespaces that are defined with a page size of larger than the default 4 KB.

Refer to Table 10 on page 60 for a list of Siebel tablespaces and suggested sizes.

6.1.6 Referential integrity

Many Siebel tables are closely related to each other; however, Siebel tables do not have any primary key and foreign key constraints at the DBMS level. Siebel Applications manage their own referential integrity at the application level and do not rely on SQL constraints. Constraints are defined in the Siebel Repository, and enforced by Siebel Applications automatically.

6.1.7 User Defined Function (UDF)

Siebel creates a UDF called NEXTSEQ. It is used to get a unique identifier for generating Siebel's row IDs.

How to test the UDF function

You can test the NEXTSEQ UDF by following this procedure:

1. Create a table with an integer column:

   ```
   db2 create table siebel.t1 (c1 int);
   ```

2. Run the following command:

   ```
   db2 values siebel.nextseq ('SIEBEL','T1','C1','');
   ```

This command should return 1 at the first execution, and then be increased by 1 at any later execution.

6.1.8 Stored procedures

The following five stored procedures created after a default installation.

- SIEBFCST - for forecasting.
- ROUTE_GENESYS - for routing CTI requests to the Genesys switch.
- ROUTE_PROSPECT - for routing CTI requests.
- SIEBSTAT - for running update statistics.
- SIEBTRUN - for truncating a table.

6.2 DB2 logs

Database logs keep a record of database changes. If a database needs to be restored to a point beyond the last full online backup, then logs are required to roll the data forward to the point of failure.

We recommend that you prepare a file system of at least 4 GB for DB2 logs.

Crash recovery

If your database becomes inconsistent due to an external factor, such as a power failure, media problem, or application failure, it initiates an activity called *crash recovery*, when a new connection is made, or when the restart command is issued. Crash recovery consists of rolling back incomplete units of work from the time a failure took place, thus allowing the database to be operational again. DB2 uses the database log files for the database crash recovery process.

There are two types of DB2 logging: *circular* and *archive*, each providing a different level of recovery capability.

Circular logging

Circular logging is the default behavior when a new database is created. With this type of logging, only full, offline backups of the database are valid. As the name suggests, circular logging uses a ring of online logs to provide recovery from transaction failures and system crashes. The logs are used and retained only to the point of ensuring the integrity of current transactions. Circular logging does not allow you to roll a database forward through prior transactions from the last full backup. Recovery from media failures and disasters is done by restoring from a full, offline backup. All changes since the last backup are lost. The database must be offline (inaccessible to users) when a full backup is taken. Since this type of restore recovers your data to the specific point in time of the full backup, it is called *version recovery*.

Archival logging

In a production environment you should use archive logging so that you can exploit online backup, backup tablespaces and roll forward recovery features. For this you need more disk space because the old logs will remain in the log directory, and you have to take care of them. We recommend that you move old log files to a backup device. You can use user exit programs for archiving and retrieving logs.

The LOGRETAIN parameter must be set to RECOVERY in order to enable archive logging when you have USEREXIT OFF, so DB2 can do the backup with all users and applications connected. The backup can be taken for either the whole database, or only for some tablespaces.

When this backup is taken, the database administrator should always be aware that future archived logs (logs that contain all units of work that have been committed) must also be copied to a safe unit since they store the data needed to update the database to the last committed transaction in case a crash occurs. This is the recommended backup strategy for a DB2 system.

6.3 Bufferpools

A bufferpool is an area of storage into which database pages (containing table rows or index entries) are temporarily read and where they are changed. The purpose of the bufferpool is to improve database system performance. Data can be accessed much faster from memory than from a disk. Therefore, the fewer times the database manager needs to read from or write to a disk, the better the performance.

The storage associated with the buffer pool is allocated when a database is activated, or when the first application connects to the database.

Note

Each tablespace page with a specific page size (4 KB, 8 KB, 16 KB, 32 KB) must be mapped to a buffer pool with the same page size.

Table 25 lists default Siebel bufferpools and their suggested sizes. The total shared memory limitation on AIX is approximately 1.75 GB, and it is less than 3 GB on Windows NT. If you don't have memory constraint, we recommend you allocate up to the maximum for all DB2 bufferpools by allocating each bufferpool based on Table 25. See 9.1.3, "DB2 bufferpool management" on page 147 for details.

Table 25. Rule of thumb for default Siebel bufferpools

Bufferpool	Suggested bufferpool size	Page size
IBMDEFAULTBP	50% of available memory up to a limit of the shared memory limit/2	4 KB
BUF4KTEMP	100 to 200 MB	4 KB
BUF32KTEMP	32 MB	32 KB
BUF16K	25% of available memory up to a limit of the shared memory limit/2	16 KB

The actual size of the buffer pools will be dictated by the data being accessed. For instance, you may not use the IBMDEFAULTBP or you may allocate large Index BP, etc.

6.4 DB2 parameters for a Siebel environment

In this section we describe the suggested DB2 parameters for a Siebel environment. Refer to “Configuring DB2 UDB” of *DB2 UDB Administration Guide: Performance,* SC09-2840, for a detailed description of those parameters.

You may use one of the following methods to reset, update, and view the DB2 customization parameters:

- The DB2 Control Center
- The command line processor

6.4.1 Suggested database manager configuration parameters

Database manager configuration parameters are related to the entire instance. Most of them either affect the amount of system resources that will be allocated, or they configure the setup of the database manager and the different communication subsystems.

After changing any of the database manager configuration parameters, the database manager must be stopped (`db2stop`) and restarted (`db2start`) to make the changes available.

Table 26 on page 99 shows the suggested database manager parameters for Siebel Applications. You can use the following commands to get, update or reset these parameters:

```
db2 get database manager configuration

db2 update database manager configuration using parm1 value1 ..

db2 reset database manager configuration
```

Table 26. Suggested database manager configuration parameters for Siebel

Parameter	Explanation	Setting/Comment
UDF_MEM_SZ	UDF Shared memory set size (4 KB)	64 (should be <= 128)
SHEAPTHRES	Sort heap threshold (4 KB)	100000
RQRIOBLK	Maximum requestor I/O block size (bytes)	65535
QUERY_HEAP_SZ	Query heap size (4 KB)	16384
MAXAGENTS	Maximum number of existing agents	1000 or the number of users + a delta for the Siebel Application Server connections
INTRA_PARALLEL	Enable intra-partition parallelism	NO

6.4.1.1 UDF_MEM_SZ

For a fenced user-defined function (UDF), this parameter specifies the default allocation for memory to be shared between the database process and the UDF. This is very critical in a Siebel environment, since each Siebel connection invokes the nextseq fenced UDF.

We recommend that you use a value of less than 128. Up to this amount of memory is also allocated and persists till the connection is reset. This buffer should be large enough to handle the input parameters and output results passed to/from UDF (both are small) and some overhead. More than that is a waste.

6.4.1.2 Sort heap threshold (SHEAPTHRES)

The parameter SHEAPTHRES is used to control the total amount of memory allocated for sorting on the DB2 server. This parameter should be set to the maximum amount of memory for all sort heaps that should be allowed to be allocated at any given time.

6.4.1.3 Client I/O block size (RQRIOBLK)

This parameter specifies the size of the communication buffer between remote applications and their database agents on the database server. The client I/O block of memory is used to serve the request and replies from and to remote applications. When a database client requests a connection to a remote database, this communication buffer is allocated on the client.

On the database server, a communication buffer of 32767 bytes is initially allocated, until a connection is established and the server can determine the value of RQRIOBLK at the client. Once the server knows the RQRIOBLK value, it will reallocate its communication buffer if the value on the client is not 32767 bytes. It is independent of the protocol used to connect the client and server.

6.4.1.4 Query heap size (QUERY_HEAP_SZ)

The maximum size of the query heap is defined by the QUERY_HEAP_SZ parameter. QUERY_HEAP_SZ is the set of contiguous memory allocated by the database manager to receive the data sent by the local application. The query heap is used to store each query in the agent's private memory. The information for each query consists of:

- Input SQLDA
- Output SQLDA
- The statement text
- SQLCA
- The package name
- Creator
- Session number
- Consistency token

This query heap parameter is provided to ensure that an application does not consume unnecessarily large amounts of virtual memory within an agent.

6.4.1.5 Maximum number of agents (MAXAGENT)

The parameter MAXAGENTS defines the maximum number of DB2 agents, including coordinator agents and subagents, in the instance. If the value of this parameter increases, the number of the Application Global Memory segments and the Agent Private Memory segments may grow, depending on the number of connected applications and DB2 agents, respectively.

6.4.1.6 Intra-partition parallelism (INTRA_PARALLEL)

The parameter INTRA_PARALLEL controls whether or not intra-partition parallelism is enabled. We suggest that this should be enabled or disabled depending on:

- Typical workload
- The number of users

In the Siebel *e*Business Applications environment, we recommend that you do not enable intra-partition parallelism (set to NO).

6.4.2 Suggested database configuration parameters

These parameters reside only on a database server and are individually assigned to each database.

Updates to database configuration parameters do not take effect while applications or users are connected to the database. After termination of all applications and reconnection, the changes are available. If a database is in the active state, it must be deactivated, which can be achieved by executing the `DB2 DEACTIVATE DATABASE` command.

Table 27 shows the suggested database parameters for Siebel Applications. We describe some key parameters in the Siebel environment in the following section. For further information, refer to *DB2 UDB Administration Guide: Performance,* SC09-2840.

You can use the following commands to get, update or reset these parameters:

```
db2 get database configuration for database-alias
```

```
db2 update database configuration for database-alias using parm1
value1
```

```
db2 reset database configuration
```

Table 27. Suggested database configuration parameters for Siebel

Parameter	Explanation	Setting/Comment
SORTHEAP	Sort list heap (4 KB)	32 (Recommended size; this may increase or decrease depending on the amount of memory in the database server machine.)
NUM_IOSERVERS	Number of I/O servers	Number of disks
SEQDETECT	Sequential detect flag	YES
DBHEAP	Database heap (4 KB)	7429
CATALOGCACHE_SZ	Catalog cache size (4 KB)	5558
LOGBUFSZ	Log buffer size (4 KB) 512	for AIX, set this to 128
UTIL_HEAP_SZ	Utilities heap size (4 KB)	5000

Parameter	Explanation	Setting/Comment
LOCKLIST	Maximum storage for lock list (4 KB)	5000 (The setting should never be smaller than this, but may be increased. We recommend that you increase LOCKLIST and MAXLOCKS if lock escalation is observed.)
APP_CTL_HEAP_SZ	Maximum applications control heap size (4 KB)	304
STMTHEAP	SQL statement heap (4 KB)	8192 (This value represents the minimum setting; if you will use EIM to import a large amount of data, set the parameter to 10000 as a starting point.)
APPLHEAPSZ	Default application heap (4 KB)	2500
PCKCACHESZ	Package cache size (4 KB)	2048
STAT_HEAP_SZ	Statistics heap size (4 KB)	8000
MAXLOCKS	Percentage of lock lists per application	20
LOCKTIMEOUT	Lock time-out (sec.)	300 or higher
NUM_IOCLEANERS	Number of asynchronous page cleaners	Number of disks in the database
DFT_PREFETCH_SZ	Default prefetch size (4 KB)	128
LOGRETAIN	Sequential or circular log files	YES
MAXAPPLS	Maximum number of active applications	Based on the number of users plus at least 20 for Application Server connections
AVG_APPLS	Average number of active applications	Depends on the environment

Parameter	Explanation	Setting/Comment
MAXFILOP	Maximum DB files open per application	500
LOGFILSIZ	Log file size (4 KB)	8000
LOGPRIMARY	Number of primary log files	30
LOGSECOND	Number of secondary log files	98
SOFTMAX	Percent log file reclaimed before soft checkpoint	80

6.4.2.1 Sort heap size (SORTHEAP)

SORTHEAP is the amount of private memory allocated to each process connected to a database at the time of a sort. This memory is allocated only at the time of sort, and is deallocated after sorting has finished.

We recommend that you use 32 to 256 pages (i.e. 128 K to 1024 K) for Siebel Applications. With a high number of concurrent users, you may not be able to afford to use a big SORTHEAP.

This parameter should have a low value so that the optimizer evaluates sorting as expensive, and therefore chooses a plan that does not sort. The DBA may notice some sort overflows, which would affect query performance. Slow queries can be isolated and fixed with additional indexes by evaluating the query plan and adding appropriate indexes.

The `db2 get snapshot for database database_name` command provides two indicators that can be used for tuning the SORTHEAP parameter:

- Total sort time
- Total sorts

Before increasing the SORTHEAP setting, ensure that both of the following are true:

- You have real memory available.
- The value of the equation *total sort time/total sorts* is decreasing.

SHEAPTHRES is a related parameter. A good rule of thumb for determining SHEAPTHRES is SORTHEAP*MAXAGENTS, or even a higher number.

6.4.2.2 Number of I/O servers (num_ioservers)

I/O servers are used on behalf of the database agents to perform prefetch I/O and synchronous I/O by utilities such as backup and restore. This parameter specifies the number of I/O servers for a database. The number of I/Os for prefetching and utilities that can be in progress for a database at any time cannot exceed num_ioservers.

We recommend that you experiment with decreasing the number of I/O servers when you encounter excessive I/O waits that are not related to the shortage of the number of tablespace containers, and you find that DB2 prefetchers are busier than the agents.

6.4.2.3 Sequential detection (SEQDETECT)

The database manager monitors I/O. If sequential page reading is occurring, the database manager can activate I/O prefetching. This is known as *sequential detection*. Sequential detection is activated by setting SEQDETECT=YES at the db cfg level, and YES is the default value.

In a Siebel environment where the queries reference lots of rows, but most of the time only a small number of rows are actually fetched by the user, the default value YES may lead to unnecessary I/O for certain parts of the Siebel Application. However, this value is very useful for DBEXTRACT.

It is possible that SEQDETECT=NO will actually lead to DB2 reading only the needed rows, without trying to bring in the bufferpool pages that will *not* be used later on.

We recommend that you experiment with NUM_IOSERVERS and SQLDETECT until you reach a point where the server is not extremely busy due to prefetchers and the response time remains good for various activities that a regular Siebel user would execute during the day.

6.4.2.4 Database heap (DBHEAP)

There is one database heap per database, and the database manager uses it on behalf of all aplications connected to the database. It contains control block information for tables, indexes, table spaces, and bufferpools. It also contains space for the log buffer (LOGBUFSZ), and the catalog cache (CATALOGCACHE_SZ). Therefore, the size of the heap is dependent on the number of control blocks stored in the heap at a given time. The control block information is kept in the heap until all applications disconnect from the database.

6.4.2.5 Log buffer size (LOGBUFSZ)

This parameter allows you to specify the amount of the database heap (defined by the DBHEAP parameter) to use as a buffer for log records before writing these records to disk. The log records are written to disk when one of the following occurs:

- A transaction commits, or a group of transactions commits (MINCOMMIT).
- The log buffer is full.
- As a result of some other internal database manager event.

This parameter must also be less than or equal to the DBHEAP parameter. Buffering the log records will result in more efficient logging file I/O because the log records will be written to disk less frequently and more log records will be written each time.

6.4.3 Suggested DB2 registry and environment variables

Table 28 shows the suggested registry and environment variables for Siebel Applications.

You can view a list of all supported registry variables by using

```
db2set -lr
```

You can change the value for a variable for the current session by using

```
db2set registry_variable_name=new_value
```

The values for the changed registry variables must be set before the `db2start` command is issued.

Table 28. Suggested DB2 registry parameters for Siebel

Parameter	Explanation	Setting
DB2MEMMAXFREE	To support hundreds or thousands of users, we need to use as little memory as we can.	3000000 or more
DB2MEMDISCLAIM	When set to YES, DB2 disclaims some or all freed memory, depending on the value of DB2MEMMAXFREE.	YES
DB2_STRIPED_CONTAINERS	When set to ON, in RAID devices, the tablespaces are created with an extent size that is equal to, or a multiple of, the RAID stripe size with an extra extent for one-page tags.	ON

Parameter	Explanation	Setting
DB2_RR_TO_RS	Improves DB2 performance with Siebel. Note that you set it to YES only in production environment servers. Turning this parameter on disables all repeatable reads, causing an application to use “read stability”. This is only acceptable for a Siebel production environment.	YES

6.4.3.1 Memory management (DB2MEMMAXFREE)

This parameter specifies the amount of free memory that is retained by each DB2 agent. In a Siebel environment, we need to support hundreds or thousands of users, which translates into hundreds or thousands of DB2 agents. Therefore, it is very important to make sure that each of these agents uses as little memory as possible without losing performance.

We recommend that you use DB2MEMMAXFREE=3,000,000 or 4,000,000 (or more). This parameter will keep agent sizes in the neighborhood of this number, keeping them from varying wildly.

Make sure that DB2MEMDISCLAIM is set to YES in order to release memory when the agent size becomes bigger than the value of DB2MEMMAXFREE. In other words, DB2MEMMAXFREE has to be combined with DB2MEMDISCLAIM=YES. DB2MEMMAXFREE and DB2MEMDISCLAIM are valid only for AIX.

6.4.3.2 DB2_STRIPED_CONTAINERS

When creating a DMS tablespace container (device or file), a one-page tag is stored at the beginning of the container. The remaining pages are available for data storage by DB2 UDB and are grouped into extent-sized blocks. When using RAID devices for tablespace containers, it is suggested that the tablespace be created with an extent size that is equal to, or a multiple of, the RAID stripe size. However, because of the one-page container tag, the extents will not line up with the RAID stripes, and it may be necessary during an I/O request to access more physical disks than would be optimal. DMS table space containers can now be created in such a way that the tag exists in its own (full) extent. This avoids the problem described above, but it requires an extra extent of overhead within the container.

To create containers in this fashion if you use RAID or Enterprise Storage System (ESS) type devices, we recommend that you set the DB2 UDB registry variable DB2_STRIPED_CONTAINERS to ON by issuing:

```
db2set DB2_STRIPED_CONTAINERS=ON
```

Any DMS container that is created will have new containers with tags taking up a full extent. Existing containers will remain unchanged.

6.5 Queries

Queries generated by Siebel Applications typically join 10 or more tables as shown in Appendix D, “How to test your SQL” on page 197. They have the inner joins first, then the left outer joins (LOJ) to add details (because LOJ always provides an answer set that is bigger than the initial answer).

All queries are dynamic, with parameter markers. DB2 reuses dynamic statements from the package cache. This means that if one user already executed a certain SQL statement, it will be reused for the second user if he executes the same SQL.

Caching packages allows the database manager to reduce its internal overhead by eliminating the need to access the system catalogs when reloading a package, or in the case of dynamic SQL, eliminating the need for compilation.

The default optimization class for Siebel Client-generated queries is 0. However, the optimization class for individual Business Components (BusComp) can be changed to a value other than the default using Siebel Tools. Some Business Components might have their optimization class set to 3 upon installation. For Siebel Server-generated queries, the default database optimization class (DFT_QUERYOPT) setting is used. The optimization class has an influence on the DB2 Optimizer's choice of plan. For example, when the optimization class is set to 3, star join is disabled (N) and nested-loop join is enabled (Y) for optimization. See Table 29 on page 108 for details about different optimization classes typically used by Siebel.

If you intend to use Siebel Tools for changing the optimization class, you should first consult with Siebel Expert Services or IBM Global Services.

Table 29. DB2 features at different optimization classes used by Siebel

DB2 features	Optimization class		
	0	**3**	**5**
star join	N	N	Y
hash join	N	N	Y
nested loop	Y	Y	Y
merge scan	N	Y	Y
list prefetch	N	Y	Y
frequency distribution stats	N	Y	Y
quantile distribution stats	N	N	Y

Chapter 7. Siebel Enterprise Integration Manager (EIM)

Siebel Enterprise Integration Manager (EIM) is a major component of the Siebel environment. From Siebel's point of view, this is their major interface with other external corporate data sources. In this chapter, we highlight the importance of EIM and factors that might have an impact on its performance.

7.1 What is Enterprise Integration Manager

EIM manages the exchange of data between Siebel database tables and other corporate databases.

Use EIM to perform bulk imports, exports, merges, and deletes. Examples of these functions include:

- When initially implementing a Siebel application, you can load the Siebel database tables with data and file attachments created by external applications.
- For example, you might import information about product lines and products from an Inventory Control database into the Siebel database.
- As part of maintaining the database, you can update it with information created by external applications.
- When initially implementing a non-Siebel application, you can export data from the Siebel database tables for use by that application.
- As part of maintaining a non-Siebel database, you can update it with information from the Siebel database.
- In response to some external events such as corporate mergers, you can merge two or more database rows into a single row.
- As part of maintaining the Siebel database, you can identify rows to be deleted from a table and its associated child and intersection tables.

For a detailed discussion of EIM, see *Siebel Enterprise Integration Manager Administration Guide, Siebel 2000, v6.0,* 10PA1-EL01-06000.

Note

Loading data directly into Siebel base tables using DB2 Load, DB2 Import, SQL Insert, Update or Delete is not supported. Due to the complexity of table relationships, you *must* use EIM to import data into Siebel base tables. Do not attempt to modify data directly in the physical tables.

Note: There are other ways to update data in addition to Siebel client or EIM. You can also write programs (e.g. VB/COM) that communicate with the Siebel Object Manager (SOM layer). Such programs are allowed to manipulate data since they are effectively using the Business Components (BusComps). For more information, refer to *Siebel VB Language Reference,* 10PA1-VB00-06000.

7.2 EIM architecture

Siebel EIM architecture consists of two data components and two data managing processes. One data component is the non-Siebel data that resides in the external data source. The other data component is the Siebel database that consists of Siebel interface tables and base tables.

One data managing process is the non-Siebel utility/process to import into and export from the EIM interface table. The other data managing process is the Siebel EIM process. See Figure 47 for an illustration of this architecture.

The Siebel EIM process either transfers data from the interface table to the base table, or extracts from the base table and populates the interface table, depending on the EIM configuration file.

The EIM interface table acts as a staging area for the EIM process.

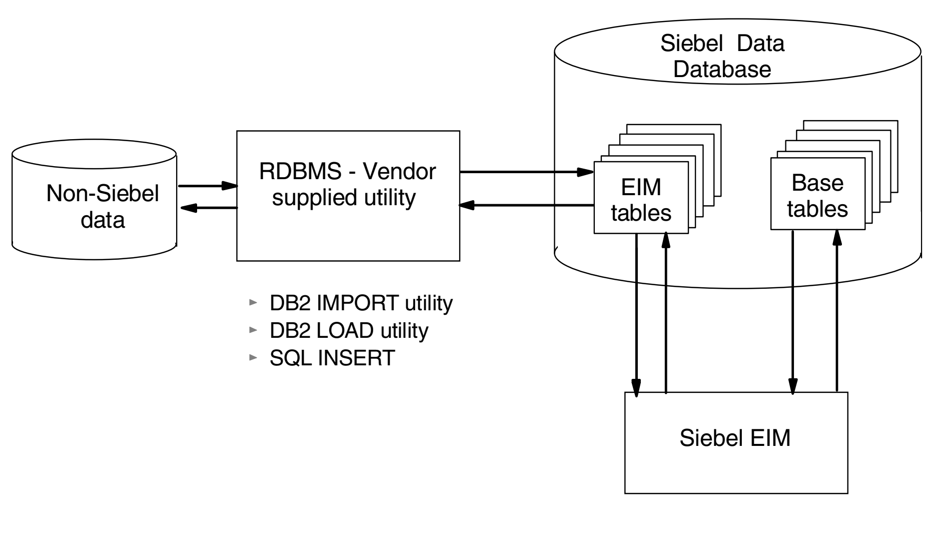


Figure 47. Siebel EIM architecture

7.3 Types of EIM interface tables

Siebel interface tables are intermediate database tables that act as a staging area between the Siebel database and other external databases.

As of Siebel 2000.x, there are two different types of interface tables: S_ type interface tables (with S_prefix and IF suffix) were released prior to Version 6.x, and EIM_ type interface tables (with EIM_ prefix), which are new in version 6.

You should use EIM_ type interface tables if:

- This is your first Siebel implementation.
- You have multiple organizations in your Siebel environment.
- You export, delete, or merge data using EIM.
- You set explicit primaries.

7.4 EIM processing

Primarily, the following steps are involved in an EIM process:

1. Preparing the interface table.
2. Preparing the EIM configuration file.
3. Running EIM.
4. Checking the results.

7.4.1 Preparing the interface table

Depending on the type of EIM processing (import, export, merge, delete or update), the selected interface table must be prepared.

For import, merge or delete processing, the interface tables must first be populated from the external data source. The combination of *rowid* and *batch number* for each row in the interface table must be unique.

For export, the interface table must not contain any row with a batch number identical to the batch number your EIM export process will be running.

7.4.2 The EIM configuration file

EIM reads the configuration file (.ifb) to determine where the data is stored and to where the data needs to be transferred. The interface tables store de-normalized data and EIM determines the relationship based on the EIM table-mapping definitions. As such, any base table extensions must be updated using Siebel Tools. Siebel Tools is a graphical user interface tool for use by developers to customize the Siebel applications.

Following are EIM parameters that can have an impact on the performance of EIM processing:

- ONLY BASE TABLES/IGNORE BASE TABLES
  Use these parameters to include tables you are interested in, or to exclude the tables that are not required. This will reduce the amount of unnecessary EIM processing.
- ONLY BASE COLUMNS/IGNORE BASE COLUMNS
  Use these parameters to include the columns you are interested in, or to exclude the columns that are not required. This will reduce the amount of unnecessary EIM processing.
- ONLY BASE COLUMNS/INSERT ROWS
  Use this parameter to control whether EIM will attempt to insert new records into the target base table(s). The default value is TRUE.
- ONLY BASE COLUMNS/UPDATE ROWS
  Use this parameter to control whether EIM will attempt to update existing records in the target base table(s). The default value is TRUE.
- COMMIT EACH PASS
  This should always be set to TRUE, to ensure the smallest unit of work within your EIM processing.
- COMMIT EACH TABLE
  You might want to use this in conjunction with COMMIT EACH PASS.
- COMMIT OPERATIONS (IMPORT only)
  This specifies the number of insert and update operations to be performed before a commit is invoked, and with the value greater than zero, it prevents the transaction rollback space from overflowing when large data sets are imported. The default for COMMIT OPERATIONS is not set; a commit is thus invoked only at the end of the import by default. Note that this parameter is useful only for row-by-row processing (with transaction logging on). It is not used for set-based processing operations.
- ROLLBACK ON ERROR
  With this set to TRUE, you should also consider using a smaller batch size, as this can have an impact on the size of the DB2 logs required.
- UPDATE STATISTICS
  To minimize the performance impact of running multiple concurrent EIM processes on the same EIM table, you should consider disabling this option.

We recommend that every section in an IFB include the parameters for ONLY/IGNORE BASE TABLES, ONLY/IGNORE BASE COLUMNS, INSERT ROWS, and UPDATE ROWS. These four parameters allow you to fine-tune EIM and have it perform the minimal amount of SQL statements.

We also recommend that (if possible) you segregate data in the EIM table so that insert-only transactions are processed separately from update-only transactions. This requires that you segregate the transactions by batch number in the EIM table, and that you include two Sections (for each previous Section) in the IFB as shown in Figure 48.

```
 <old IFB>
[Process Accounts]
TYPE = IMPORT
ONLY BASE TABLES = S_ORG_EXT
ONLY BASE COLUMNS = <list of columns>
BATCH = 1-100

<new IFB>
[Process New Accounts]
TYPE = IMPORT
ONLY BASE TABLES = S_ORG_EXT
ONLY BASE COLUMNS = <list of columns>
BATCH = 1-50
INSERT ROWS = TRUE
UPDATE ROWS = FALSE

[Process Existing Accounts]
TYPE = IMPORT
ONLY BASE TABLES = S_ORG_EXT
ONLY BASE COLUMNS = <list of columns>
BATCH = 51-100
INSERT ROWS = FALSE
UPDATE ROWS = TRUE
```

Figure 48. IFB example - segregate transaction by batch number

7.4.3 Running an EIM process

You can run an EIM process (import, export, delete, or merge) once you have done the following:

1. Identified the data for EIM processing
2. Prepared the related interface tables

3. Modified the EIM configuration file accordingly

To run an EIM process, use a graphical user interface. For example, from the Siebel Call Center screen, choose **Screens > Server Administration > Servers > Server Tasks** and follow the instructions outlined in *Siebel Enterprise Integration Manager Administration Guide, Siebel 2000, v6.0,* 10PA1-EL01-06000.

7.4.4 Checking the results

It is very important to check the results of your EIM processing. Monitor the status of the EIM process by:

- Reviewing the IF_ROW_STAT columns of the respective row in the interface table

  EIM updates the IF_ROW_STAT column of the respective row in the interface tables after each phase of EIM processing.

- Reviewing the Task Info log
  The Task Info log consists of three general sections:
  - Startup messages
  - Run-time messages
  - Row count summary of each process

- Enabling the trace facility

  EIM also provides a tracing and SQL tracing facility. The trace flags can be set during the EIM task definition.

For details, refer to *Siebel Enterprise Integration Manager Administration Guide, Siebel 2000, v6.0,* 10PA1-EL01-06000*.*

7.5 Tuning EIM processing

In this section we highlight the factors that can affect/improve the performance of EIM processing. They are:

1. Placement of EIM tables

   Because EIM tables are not used by any online user, they should be placed in separate tablespaces and bufferpools (if possible). This will minimize I/O and bufferpool contention. This is especially true when EIM processes are run during the day, when online users can be impacted.

2. Indexes of base tables

   For the initial loading of the base tables, you might want to consider dropping the indexes of base tables before running the EIM. Then, recreate the indexes after successful loading of the base tables.

3. Data-related considerations

   Separate and group rows based on different activities (insert/updates/delete/merge) into different batches.

   By separating and grouping rows based on activities, you can tune the configuration in the EIM ifb file. With the specified ifb file, you can either choose IGNORED or INCLUDE BASE TABLES or COLUMNS. This will eliminate unnecessary processing.

4. Use concurrent EIM processing

   You might consider running the EIM tasks in parallel. Concurrent EIM processes can run against the same EIM interface table only if they are operating on different batch numbers.

   To minimize the possibility of a deadlock and lock contention when doing parallel EIM processing, choose UPDATE STATISTICS using a single EIM run. However, subsequently do the following:

   a. Run a parallel EIM with UPDATE STATISTICS turned off.
   b. Ensure that COMMIT EACH PASS and COMMIT EACH TABLE are turned on to free up locks as soon as possible.
   c. Use a small number of records in each batch number in the EIM table. We recommend approximately 2,000 to 3,000 records in each batch number.

5. DB2-related activities can be viewed from these perspectives:

- Batch size and DB2 logs

  Changes to DB2 data are logged into the DB2 log, which must be big enough to accommodate the complete unit of work. If there are insufficient DB2 logs available, DB2 will crash and force a crash recovery as that is the only way to ensure data integrity. During the crash recovery, DB2 will flush all committed transactions to disks and rollback all uncommitted transactions.

  DB2 has a limit of 128 logs file (primary and secondary) and a log space limit of 4 GB (max). If you have exceeded these limits, do the following:

  a. Reduce the number of concurrent EIM processes.
  a. Ensure that your COMMIT ON PASS is enabled.

- Tune SQL

  If, after tuning and sizing your database, your EIM is still performing poorly, we recommend you tune the SQL. For a detailed description of tuning an SQL, see 9.2, “How to improve SQL performance” on page 150.

- Avoid lock escalation

  Increase the LOCKLIST and MAXLOCKS DB2 parameters if lock escalation is observed in the db2diag.log file.

- Disable the updating of DB2 statistics

  Consider disabling UPDATE STATISTICS within the EIM processing. Instead, you might want to put in place a process to monitor the percentage of change of EIM tables and issue a run update statistics job outside the EIM processing.

  We recommend that you run EIM using UPDATE STATISTICS once for a given EIM interface table. Subsequent runs can have UPDATE STATISTICS turned off.

Note

Never update statistics on any EIM interface tables after you have updated them as part of EIM processing.

7.6 An EIM tuning challenge

In this section we highlight an interesting challenge related to tuning the EIM process.

7.6.1 The challenge

When importing, exporting, merging or deleting data, EIM performs a sequence of tasks. Each task involves multiple passes. At least one pass is required for each EIM interface table included in the process. In some cases, EIM may repeat several tasks. Some of these tasks update columns in the EIM tables. It is the updating process that presents the challenge.

To understand the problem, let’s look at some SQL statements in an EIM process in detail.

Assume that when processing EIM interface table S_ACCOUNT__IF, we have to perform two passes on this table. The sequences/observations are as follows:

1. The statistics before the start of the EIM process are as shown in Table 30. Note that the column T_ORG_EXT_RID is a column in the S_ACCOUNT_IF table.

   Table 30. Starting statistics

Variable	Value	Description
NUMNULLS (from SYSCAT.COLUMNS)	10000	Number of nulls value in column T_ORG_EXT__RID
Batch Size	10000	Number of rows for that batch

2. The S_ACCOUNT__IF table has about 23 indexes, but only two are related to this example; see Table 31.

   Table 31. Two indexes of S_ACCOUNT__IF

Index name	Index columns
S_ACCOUNT_IF_T02	IF_ROW_BATCH_NUM,T_ORG_EXT__RID
S_ACCOUNT_IF_T04	IF_ROW_BATCH_NUM,ACCNT_NAME,ACCNT _LOC,TORG_EXT_BU_ID

3. For the first pass, EIM updates column T_ORG_EXT__RID; see Figure 49.

```
UPDATE siebelS_ACCOUNT_IF T1
     SET        T_ORG_EXT__RID = '1+9ZY+' ||
rtrim(char(int(siebel.nextseq('siebel',
            'S_APP_VER', 'DCKING_NUM', '6') - 416311))),
        T_ORG_EXT__UNQ = 'Y'
  WHERE (T_ORG_EXT__EXS = 'N' AND
        ROW_ID =
        (  ...........) )
```

Figure 49. SQL to update column T_ORG_EXT__RID

4. In the second pass, EIM updates T_ORG_EXT__RID again; see Figure 50 on page 118.

   One of the predicates in the second pass is T2.T_ORG_EXT__RID IS NOT NULL. Based on the columns statistics, index S_ACCOUNT_IF_T02 is ideal for this SQL, because there should be no rows qualifying the predicates. But in reality, the column has been updated by the first EIM pass and there are no null values in the column.

Therefore, using index S_ACCOUNT_IF_T04 might be a better choice!

The problem is that the column statistics have not been updated before the second pass, thereby resulting in the wrong access path.

```
UPDATE siebel.S_ACCOUNT_IF T1
    SET        T_ORG_EXT__RID =
       (SELECT MIN(T_ORG_EXT__RID)
          FROM siebel.S_ACCOUNT_IF T2
        WHERE (T2.T_ORG_EXT__RID IS NOT NULL AND
               T2.ACCNT_NAME = T1.ACCNT_NAME AND
               (T2.ACCNT_LOC = T1.ACCNT_LOC OR
                (T2.ACCNT_LOC IS NULL AND T1.ACCNT_LOC IS NULL)) AND
               T2.IF_ROW_BATCH_NUM = 11))
```

Figure 50. Second SQL to update the T_ORG_EXT__RID

7.6.2 Solution

This is a tricky situation. The ideal solution is to have some dynamic way of collecting the statistics. Failing that, you might have to set numnulls to 0 initially.

7.7 EIM tuning scenario

In this section, we describe two EIM tuning scenarios that frequently happen in many existing customers, so that you can avoid a similiar problem in your installation. In these cases, the customer had created a Siebel Enterprise for their User Acceptance Test (UAT) environment.

7.7.1 Scenario 1

Background

The customer had created an EIM IFB with two sections - the first section would load Contact records into S_CONTACT, and the second section would load associated Contact Address records into S_ADDR_PER. No records existed in S_CONTACT or S_ADDR_PER, since this was a new Siebel implementation.

The customer loaded multiple batches of related data for Contacts and Contact Addresses. The customer observed that the throughput for Contacts remained approximately constant, but the throughput for Contact Addresses seemed to decline at an almost exponential rate.

Analysis

We verified that there was sufficient hardware capacity for the Siebel Server and the Database Server. We then verified that the EIM IFB had at least some of the parameters to help focus the EIM task (in this case, the customer was using ONLY BASE TABLES and ONLY BASE COLUMNS).

We set the parameters INSERT ROWS=TRUE and UPDATE ROWS=FALSE in each IFB Section since the customer was only inserting new records. Finally, we verified that the DB2 instance and database were configured as expected.

We ran the EIM task again, using the parameters ErrorFlags=1 and TraceFlags=1, and then reviewed the resulting trace file. The parameter TraceFlags=1 caused EIM to write a summary after each batch; the summary lists the elapsed execution time for each of the 11 steps for every batch number. We reviewed the trace file and noticed that the elapsed time for Step 11 of each Contact Address batch would increase as more Contacts and Contact Addresses were loaded into the base tables (S_CONTACT and S_ADDR_PER).

We ran the EIM task again, using the parameters ErrorFlags=1, TraceFlags=1, and SQLTrace=1. This produced a very large trace file since all SQL statements were now being captured in the trace file. We reviewed the trace file and noticed that a single SQL statement (UPDATE S_CONTACT SET...) in Step 11 was the source of the problem.

Results

We and the customer realized that the DB2 Optimizer thought there were no records in the table S_CONTACT. Statistics had been collected for both S_CONTACT and S_ADDR_PER prior to loading any records in either table, and had not been collected since loading thousands of records into both S_CONTACT and S_ADDR_PER. We and the customer deduced that the DB2 Optimizer thought there were no records in S_CONTACT and was therefore not using any indices when performing the UPDATE statement on S_CONTACT in Step 11.

To test the hypothesis, the customer rolled back the UAT database to the point where there were no records in the S_CONTACT and S_ADDR_PER tables.

We ran the EIM task again (with ErrorFlags=1 and TraceFlags=1) and allowed approximately 5,000 records to be loaded into both S_CONTACT and S_ADDR_PER. As expected, Step 11 for the Contact Addresses performed poorly. We ran the RUNSTATS utility for the S_CONTACT table, and then ran the EIM task again and loaded an additional 5,000 records into both S_CONTACT and S_ADDR_PER. The SQL statement (UPDATE S_CONTACT SET...) in Step 11 for the Contact Addresses now performed optimally.

Recommendation

We recommend that you collect new statistics (either using RUNSTATS or the script UPDATESTATS.KSH) after loading data into a table that previously had no records.

7.7.2 Scenario 2

Background

The customer had created a Siebel Enterprise Server for their User Acceptance Test (UAT) environment. They began to do the initial data loads using EIM and were upset to see EIM's throughput at approximately one to two rows per second.

Analysis

We verified that there was sufficient hardware capacity for the Siebel Server and the Database Server. We then verified that the EIM IFB had at least some of the parameters to help focus the EIM task (in this case, the customer was using ONLY BASE TABLES and ONLY BASE COLUMNS).

We then began to review the configuration of the DB2 UDB instance and database. It was discovered that the bufferpools had been left at the default value of 250 pages.

Results

The customer increased the number of pages allocated for the various bufferpools, stopped the DB2 instance, restarted the DB2 instance, and then ran the same EIM task again. The throughput for the EIM task increased to approximately 50 rows per second for Contacts (S_CONTACT) and 140 rows per second for Contact Addresses (S_ADDR_PER).

Chapter 8. Administering Siebel and DB2 UDB

In this chapter we highlight the activities that are usually performed by database and system administrators in a production environment, and apply them to an environment utilizing Siebel and DB2 UDB.

Database and system administrators must put in place procedures, utilities, and processes to monitor and manage the CPU usage, I/O usage and disks space usage. They have to ensure that the performance of the system is monitored and managed.

They must also ensure that the Siebel DB2 UDB system is fully recoverable after hardware, software or power failures. They might even develop a disaster recovery plan.

In this chapter we examine the following:

1. Monitoring system resources
2. Monitoring DB2 activities
3. Capacity planning
4. Backup/recovery
5. Using other DB2 tools
6. Using other Siebel tools

Note

In this chapter we do not fully describe in detail all the day-to-day activities of database and system administrators. Instead, we highlight the importance of these roles and the impact they have on the overall management of the Siebel DB2 UDB environment.

Database and system administrator may further develop the in-house activities involved in administering this environment, based on the discussion in this chapter.

8.1 A typical day of a database and system administrator

Let’s look at a typical day of database and system administrators. Assume they are attending a monthly operations meeting. In this meeting, issues and concerns regarding the operation and performance of the Siebel DB2 UDB environment are raised, representing feedback from the user community.

Two performance issues raised by the user community are:

1. On the second Tuesday of this month, between 10 am and 11 am, the user community experienced relatively slow response time when using the Siebel applications. It seemed to affect every Siebel user and application.
2. On the second Thursday of this month, users experienced relatively slow response time when using the Siebel application to access account information. This happened throughout the day.

Reactive environment response

In a *reactive* operating environment, the database and system administrator response will likely be:

1. We need more information regarding the problem.
2. Can you isolate the problem?
3. What were you doing?
4. Once you have isolated the problem, we will enable traces to capture the problem and maybe tune the system or SQL.

Proactive environment response

In a *proactive* operating environment, the database and system administrator response will likely be:

1. We need more information regarding the problem.
2. Can you isolate the problem?
3. What were you doing?
4. Once you have isolated the problem, we will enable traces to capture the problem and maybe tune the system or SQL.
5. In the meantime, we will review our statistics to see if we can find any possible causes of your performance problem.

Note that it is important to build an infrastructure to collect and maintain performance statistics on the system continuously so that you are prepared for unexpected performance issues.

In this case, the administrators reviewed three in-house-developed DB2 tables: the system resources utilization table; the Siebel tables statistics table; the Siebel tablespaces statistics table. These tables are shown on the following pages.

Table 32. System resources utilization table

Data collected	Description
Timestamp	Date and time of statistics collection
% CPU utilization	% CPU utilization
% memory utilization	% memory utilization
% I/O utilization	% I/O utilization
% network utilization	% network utilization

Table 33. Siebel table statistics table

Data collected	Description
Timestamp	Date and time statistics were collected
Table name	Name of Siebel table
No. of rows	Row count (cardinality)

Table 34. Siebel tablespace statistics table

Data collected	Description
Timestamp	Date and time of statistics collection
Tablespace name	Name of Siebel tablespace
Total allocated pages	Total allocated pages
Total usable pages	Total usable pages
Total used pages	Total used pages

Note

The columns used in these tables have been simplified for discussion purposes. The actual values to be captured are discussed in 8.2, "Monitoring system resources on AIX" on page 125.

The information in the system resources utilization table is collected hourly, while the information in the Siebel table statistics table and the Siebel tablespace statistics table are collected nightly.

In our scenario, this information has been collected for the last six months.

After analyzing these tables, three observations were made:

1. For the last two months, on every second Tuesday of the month between 10 am and 11am, there was a high percentage of network utilization.
2. For the last two months, on every second Thursday of the month, the average CPU utilization and I/O utilization of the day seems to be higher than the norm.
3. For the last two months, on every second Thursday of the month, the account table grow by 15%.

The administrators finally found the cause of the problem, which seemed to be related to a legacy system. Two months previously, the enterprise started to gradually merge account information from a legacy system, in a monthly partial merge, as follows:

1. Every second Tuesday of the month, at about 10 am, the data was FTPed from the legacy system to the database server machine, which took about 45 minutes.
2. The data was then loaded, using EIM, to the Siebel Base table on Wednesday night after the nightly runstats.

The solution to the performance issues was to FTP the file *after* office hours, and to run the EIM load/merge *before* the nightly runstats.

In this section, we highlighted the differences between a reactive and a proactive administration methodology. With the proactive methodology, you are able to get an overall picture of system resources utilization over a period of time, and possibly pinpoint the cause of the problem by analyzing the statistics collected. We hope you appreciate the importance of this methodology and apply this proactive technique in the administration of your Siebel DB2 UDB environment.

Now let's examine what you we can monitor and capture. We have also included an example detailing how to capture the data necessary for trend analysis. This example is based on the Siebel tablespace statistics table; see Appendix B, "An example of data collection for trend analysis" on page 173.

8.2 Monitoring system resources on AIX

In this section we highlight the AIX system resources you need to monitor. Tools to capture the information are also identified.

We assume that you schedule a series of jobs at predefined intervals to collect system performance statistics and transform them into insert statements. The system performance statistics are then inserted into special user-defined tables (non-Siebel tables).

These special user-defined tables are ultimately used for trend analysis. They provide a pretty good overall picture of your system resource utilization over a period of time.

8.2.1 System bottlenecks

System resource utilization/bottlenecks can be classified into three types:

- CPU utilization
  A system is considered CPU bound when the %user plus %sys is greater than 80%.

  A good rule of thumb is to keep the average CPU usage the below 80%, so that it can handle small, unpredictable bursts of the activities.

  While monitoring CPU, do not worry about small (1 to 3 seconds) of 100% CPU busy, but instead take a long-term view, as you are more concerned with the average CPU utilization.

- Memory and paging

  You should monitor the following thresholds:

  - Paging space is considered low when greater than 70% of paging space is active.
  - Paging is considered bound when:
    - The paging volume %busy is greater than 30% of the I/O.

        - The paging logical volume-to-paging volume activity is greater than 40%.

- Disk and I/O

  You should monitor the following thresholds:

    - A disk is I/O-bound when CPU %iowait is greater than 40%.
    - A disk is bound when a disk is regularly more than 45% busy.

- Process problems

    - You should monitor a process if it is getting more CPU than necessary.
    - While monitoring processes, do not worry about small (1 to 3 seconds) of 100% CPU busy, but instead, take a long-term view as you are more concerned with the average CPU utilization.

- Network problems

    - Monitor a network if it is transferring too much data or too many packets.
    - Keep an eye on the number of packets transferred per second, and on the number of bytes transferred per second.
    - While monitoring processes, do not worry about small (1 to 3 second) bursts of activities.

8.2.2 Tools available in AIX

In this section we highlight some of the tools available in AIX to monitor system resources.

1. Performance ToolKit/6000 (also known as PTX or xmperf).

   This is an excellent X-Window graphical performance monitoring tool. It can show historical data and is good for trend analysis of all system resources.

   This is an optional AIX component.

2. Online Monitor (nmon)

   This is a freely available and useful tool. It provides a snapshot of AIX statistics on a *dumb* screen.

   It is available to IBM and IBM Business Partners via PartnerInfo and is located at:

   `http://w3.aixncc.uk.ibm.com`

3. File I/O Monitor - filemon

This tool is used to monitor the performance of the File System and report the I/O activity on behalf of files, virtual memory segments, logical volumes, and physical volumes.

Following is an example:

a. Enter: `filemon -O all -o filemon.out`

b. Start the workload.

c. Enter: `trcstop`

4. Disk I/O statistics - iostat

   This tool reports CPU and I/O statistics for TTY devices, disks and CD-ROMs.

   Following is an example:

   ```
   iostat 10 20
   ```

5. List Paging Space - lsps

   This tool displays paging space characteristics such as physical volume, volume group name, size, percentage of paging space used and whether the space is active or inactive.

   Following is an example:

   ```
   lsps -a -s [paging space]
   ```

6. Network Monitor - netpmon

   This tool reports network activities and network-related CPU usage.

   Following is an example:

   ```
   netpmon -O all -o netpmon.out
   ```

7. Process State - ps

   This tool displays the status of currently active processes.

   Following is an example:

   ```
   ps -ef
   ```

8. System Activity Reporter - sar

   This command is a standard UNIX command, used to gather statistical data about the system.

   Following are examples:

   ```
   sar 10 100
   ```

   ```
   sar -P ALL 1 10
   ```

9. System Virtual Memory Monitor - svmon

   This command is used to capture and analyze a snapshot of virtual memory.

   Following is an example:

   ```
   svmon -G
   svmon -Ssu 10
   ```

10. Virtual Memory Management Statistics - vmstat

    This command is used to report statistics on kernel threads in the run and wait queues, memory, paging, disks, interrupts, system calls, context switches, and CPU activity.

    Following is an example:

    ```
    vmstat 10 20
    ```

Note

For more detailed information related to monitoring AIX system resources, refer to:

- *Understanding IBM RS/6000 Performance and Sizing*, SG24-4810
- *RS/6000 Performance Tools in Focus*, SG24-4989
- AIX standard documentation can be found at:
  `http://www.rs6000.ibm.com/doc_link/en_US/a_doc_lib/aixgen/`

8.2.3 Administrating system resources using AIX tools

Except for xmperf, all the AIX tools mentioned give snapshots of the system. When you have a problem, you would run these tools to find out where the bottlenecks are. However, these tools are not useful for system administration, trend analysis, or capacity planning.

In administrating and managing a AIX system, system administrators should consider putting in place scheduled jobs to run the AIX tools to collect information at predefined intervals. This information is extracted and consolidated into DB2 tables for trend analysis.

Depending on the interval and the data collected, you can get a pretty good picture of your system. You will be able to determine:

- Peak CPU usage, I/O usage, memory usage by hour, by day, by month
- The top ten processes by CPU usage, I/O usage, or memory usage
- The growth rate of the system for capacity planning purposes
- And many more trends

8.3 Monitoring system resources on NT

In Windows NT, you would probably monitor the same kind of information as in an AIX environment:

- Memory bottlenecks
- Processor bottlenecks
- Disk bottlenecks
- Network bottlenecks

8.3.1 Monitoring tools available in NT

The following tools are available for monitoring NT system resources:

- NT Performance Monitor (perfmon.exe)
- NT Data Log Server (datalog.exe)
- NT Performance Data Log Server (pdlsvc.exe)
- Microsoft SMS
- Tivoli IT Director

You can exploit these tools to capture information at regular intervals, collect the information, and then store it in user-defined DB2 tables for trend analysis.

8.4 Monitoring DB2 activities

Primarily, DB2 provides two types of performance monitor: the Event Monitor and the Snapshot Monitor.

The Event Monitor is useful for monitoring a particular event, especially in the problem determination or performance tuning areas, where you are looking for occurrences of a situation. For example, if you are interested in deadlocks and want to see what processes are holding the locks, or what locks each process is waiting on and how the deadlock was resolved, the Event Monitor will provide these answers.

On the other hand, the Snapshot monitor gives you an overall picture of a point in time. It will capture the number of locks held, bufferpool usage, number of lock escalations and so on, at a particular time.

In the context of administrating and managing a Siebel DB2 UDB environment, the Snapshot monitor is very useful. But the information must be captured at regular intervals and stored in user-defined DB2 tables, so that trend analysis can be performed.

Information that should be monitored and collected can be classified into these types:

- Sorts-related information
  - Total Sort Heap Allocated
  - Post Threshold Sorts
  - Piped Sorts Requested
  - Piped Sorts Accepted
  - Total Sorts
  - Total Sort Time
  - Sort Overflows
  - Active Sorts
- Hash Join-related information
  - Total Hash Joins
  - Hash Join Threshold
  - Total Hash Loops
  - Hash Join Overflows
- Bufferpool-related information
  - Buffer Pool Data Logical Reads
  - Buffer Pool Data Physical Reads
  - Buffer Pool Data Writes
  - Buffer Pool Index Logical Reads
  - Buffer Pool Index Physical Reads
  - Buffer Pool Index Writes
  - Total Buffer Pool Physical Read Time
  - Total Buffer Pool Physical Write Time3
  - Database Files Closed
  - Buffer Pool Asynchronous Data Reads
  - Buffer Pool Asynchronous Data Writes6
  - Buffer Pool Asynchronous Index Writes
  - Buffer Pool Asynchronous Index Reads
  - Buffer Pool Asynchronous Read Time
  - Buffer Pool Asynchronous Write Time

  - Buffer Pool Asynchronous Read Requests
  - Buffer Pool Log Space Cleaners Triggered
  - Buffer Pool Victim Page Cleaners Triggered
  - Buffer Pool Threshold Cleaners Triggered
  - Buffer Pool Information
  - Buffer pool Name
  - Time Waited for Prefetch
  - Buffer pool Hit Ratio (calculated):

    ```
    (1 - ((Buffer Pool Data Physical Reads + Buffer Pool Index
    Physical Reads)/(Buffer Pool Data Logical Reads + Buffer
    Pool Index Logical Reads))) * 100%
    ```

  - Index Buffer pool Hit Ratio (calculated):

    ```
    (1 - ((Buffer Pool Index Physical Reads)/(Buffer Pool Index
    Logical Reads))) * 100%
    ```

  - Data Page Hit Ratio for Buffer pool (calculated):

    ```
    1 - (Buffer Pool Data Physical Reads/Buffer Pool Data
    Logical Reads)
    ```

  - Overall Buffer pool Hit Ratio (calculated):

    ```
    1 - ((Buffer Pool Data Physical Reads + Buffer Pool Index
    Physical Reads)/(Buffer Pool Data Logical Reads + Buffer
    Pool Index Logical Reads))
    ```

- Catalog Cache information
  - Catalog Cache Lookups
  - Catalog Cache Inserts
  - Catalog Cache Overflows
  - Catalog Cache Heap Full
  - Catalog Cache Hit Ratio (Calculated):

    ```
    (1 - (Catalog Cache Inserts/Catalog Cache Lookups))
    ```

- Package Cache information
  - Package Cache Lookups
  - Package Cache Inserts
  - Package Cache Overflows
  - Maximum Package Cache Size
  - Section Lookups
  - Section Inserts

    ```
    (1 - (Package Cache Inserts/Package Cache Lookups))
    ```

- Logging information-circular logging only
  - Maximum Secondary Log Space Used
  - Maximum Total Log Space Used

    - Secondary Logs Allocated Currently

- Logging information for both Circular and Archival logging:
    - Number of Log Pages Read
    - Number of Log Pages Written
    - Unit of Work Log Space Used
    - Total Log Space Used
    - Total Log Available

- Locks and Deadlocks information
    - Locks Held
    - Total Lock List Memory In Use
    - Deadlocks Detected
    - Number of Lock Escalations
    - Exclusive Lock Escalations
    - Lock Mode
    - Lock Status
    - Lock Object Type Waited On
    - Lock Object Name
    - Number of Lock Timeouts
    - Maximum Number of Locks Held
    - Connections Involved in Deadlock
    - Lock Escalation
    - Lock Mode Requested

There are more performance variables available for monitoring; we have simply highlighted the most important ones. For a detailed description of this subject, see *DB2 System Monitor Guide and Reference* V6, SC09-2849.

8.5 Monitoring for capacity planning

Capacity planning is the ability to predict and estimate the growth of system resources such that you can put in place some planning and budgeting to cater for future growth. (In this context, we mean resource requirements in the next two to five years.) Factors that affect growth are:

- Implementation of new applications into the production environment
- Rolling out the application in phases to different departments
- Rolling out new features in phases to the user community
- Unexpected growth of tables
- Changes in user expectation and uses for the applications
- And many others

As you can see, some activities are planned, and some are not. The demand for more capacity for those planned activities is accounted for. But activities that are unexpected can have a major impact on the demand on your system resources.

To be able to cater for unexpected growth, you must put a strategy in place to collect information and perform trend analysis. In the following sections, we examine what information can be collected for the purposes of capacity planning.

8.5.1 Monitoring disk space requirements

There are two DB2 objects that you can monitor and collect information about for capacity planning purposes: DB2 tables and tablespaces. You will need to know the growth rate of each Siebel table and tablespace, because this information will give you an insight into the expected growth rate of your Siebel database.

8.5.1.1 DB2 tables

You should schedule weekly runstats for most of the Siebel and system tables. After the runstats, a script is run to extract table-related information from the SYSCAT.TABLES and format it into insert statements.

Information to be extracted from DB2 tables is as follows:

- TABNAME
- COLCOUNT
- CARD
- NPAGES
- FPAGES
- OVERFLOW
- Current timestamp

These insert statements are then used to populate a special user-defined DB2 table.

At this point, you can perform trend analysis on this special user-defined table. It will give you a useful view of the growth pattern of your Siebel tables.

If you have another strategy for EIM tables, you might not want to collect this information for EIM tables. You may run runstats and the script before or after an EIM process.

Note

You can use the information collected on the Siebel tables to schedule your runstats.

8.5.1.2 DB2 tablespaces

The same technique applies to DB2 tablespaces. Tablespace information is collected from the `db2 list tablespaces show detail` command.

Information to be extracted from DB2 tablespaces is as follows:

- Total pages
- Usable pages
- Used pages
- Free pages
- Current timestamp

8.5.2 Monitoring CPU utilization requirements

The information to be monitored and collected is discussed in 8.2, “Monitoring system resources on AIX” on page 125.

8.5.3 Monitoring memory utilization requirements

The information to be monitored and collected is discussed in 8.2, “Monitoring system resources on AIX” on page 125.

8.6 Backup/recovery

This section covers the backup/restore requirements of a Siebel DB2 UDB environment to ensure that you are able to perform full recovery of the Siebel DB2 UDB environment.

The successful factors for a DB2 database recovery are:

- Availability of the DB2 database backup image.
- Backups of the Siebel DB2 database are scheduled regularly.
- Availability of DB2 logs for any recovery purposes.
- DB2 logs are archived.

Note

Some data used by the Siebel environment is externalized and managed by Siebel outside of the DB2 database.

Therefore, to ensure full recovery of the Siebel and DB2 UDB environment, you must also look at the backup strategy for Siebel's files.

8.6.1 Backing up the DB2 database

To ensure full recovery of the DB2 database, we recommend the following:

- At a minimum, schedule a full weekly offline DB2 backup and a daily online DB2 backup.
- Either:
  - Enable Archival Logging (LOGRETAIN=ON) for your Siebel production database by entering:

    ```
    db2 update db cfg for [database name] using LOGRETAIN ON
    ```

  Or:

  - Consider enabling DB2 USEREXIT to archive DB2 logs to disk or to the Tivoli Storage Manager by entering:

    ```
    db2 update db cfg for [database name] using USEREXIT ON
    ```

    Note: If you activate DB@ USEREXIT, you do not need to activate LOGRETAIN as it is implicit, although it will not show in the database configuration.

For a system that uses DB2 Circular Logging, you should schedule full offline DB2 backups as regularly as possible, because your only recovery option is to restore from an offline DB2 backup image copy.

8.6.2 Backing up Siebel files

Using the standard operating system backup command, when you back up your database, you should also back up the following Siebel objects:

- The Siebel install directory for all Siebel components
- The Siebel File System

8.6.3 Recovery scenarios

In this section, we review the various recovery scenarios and the action to be taken to restore the DB2 database to an acceptable state.

The five possible recovery scenarios are:

Scenario 1: DB2 or the system crashed without any disk failure.
This is when, after a DB2 or system crash, the disk holding DB2 data and DB2 logs are intact.

Scenario 2: DB2 or the system crashed with data disk failure.
This is when the disks holding the DB2 data are lost, but the DB2 logs are intact.

Scenario 3: DB2 or the system crashed with DB2 log disk failure.
This is when the disks holding the DB2 logs are lost.

Scenario 4: The total system crashed and system rebuild/restore are required.
This is in the event of a major disaster and a full system restore/rebuild is necessary.

Scenario 5: Point-in-time recovery.
This is when there is no system or DB2 crash, but a user would like to roll back the database to a past point in time.

8.6.3.1 With DB2 Circular Logging enabled - recovery actions

For scenario 1:

1. DB2 should automatically perform a crash recovery on the first DB2 connection after the crash when auto restart is enabled.

   ```
   db2 update db cfg for [database name] using AUTORESTRAT ON
   ```

2. No further action is required from the DB2 perspective.
3. Run the Siebel File System Utility (SFSUTL) to ensure no orphan attachments exist. See 8.8.7, “SFSUTL” on page 141 for more details.
4. Consider re-extracting all Mobile Clients.

For scenario 2, 3 and 4:

1. Ensure that the DB2 file systems for data and logs that are affected by the disk failure are available/online.
2. Restore the DB2 database from the latest DB2 offline backup image.
3. Restore the corresponding Siebel files.
4. Run the Siebel File System Utility (SFSUTL) to ensure no orphan attachments exist. See 8.8.7, “SFSUTL” on page 141 for more details.
5. Consider re-extracting all Mobile cCients.

For Scenario 5:

1. Identify the appropriate DB2 offline backup image.
2. Restore the DB2 database from the designated DB2 offline backup image.
3. Restore the corresponding Siebel files.
4. Run the Siebel File System Utility (SFSUTL) to ensure no orphan attachments exist. See 8.8.7, “SFSUTL” on page 141 for more details.
5. Consider re-extracting all Mobile Clients.

8.6.3.2 With DB2 Archival Logging enabled - recovery actions

With this option enabled, each scenario is handled differently.

For scenario 1:

1. DB2 should automatically perform a crash recovery on the first DB2 connection after the crash when auto restart is enabled.

   ```
   db2 update db cfg for [database name] using AUTORESTRAT ON
   ```

2. No further action is required from the DB2 perspective.
3. Run the Siebel File System Utility (SFSUTL) to ensure no orphan attachments exist. See 8.8.7, “SFSUTL” on page 141 for more details.
4. Consider re-extracting all Mobile Clients.

For scenario 2: A DB2 restore from an offline backup followed by a DB2 rollforward to end of log is required to fully recover from the situation.

After the data disks are back online:

1. Ensure that the DB2 file systems for data and logs that are affected by disk failure are available/online.
2. Run `db2 list history backup all for [database name]` to identify the latest DB2 offline backup image to restore to and take note of the Earliest Log that is associated with this backup image.
3. Make a copy of all the DB2 logs, starting from the earliest log in step 1 to the latest logs currently in the DB2 log directory. This is required as a precaution in case you need to re-restore the database due to unforeseen circumstances, because at the end of the DB2 rollforward process, DB2 will truncate the last log.
4. Restore DB2 from the designated offline image copy.
5. Roll forward to the end of logs and stop.
6. Restore the corresponding Siebel files.

7. Run the Siebel File System Utility (SFSUTL) to ensure no orphan attachments exist. See 8.8.7, “SFSUTL” on page 141 for more details.
8. Consider re-extracting all MobileClients.

For scenario 3 and 4: Since you have lost all your current DB2 logs, your only option is to do a DB2 restore from an offline backup, followed by a DB2 rollforward to the last available logs that have been archived.

After the data and log disk are back online:

1. Ensure that the DB2 file systems for data and logs that are affected by disk failure are available/online.
2. Run `db2 list history backup all for [name of database]` to identify the latest DB2 offline backup image to restore to, and take note of the earliest log that is associated with this backup image.
3. Make a copy of all the DB2 logs, starting from the earliest log in step 1 to the latest archived logs. This is required in case you need to re-restore the database due to unforeseen circumstances, because at the end of the DB2 rollforward process, DB2 will truncate the last log.

Note

If you do not archive your DB2 logs, or if your archive log is in the same physical disk as your active log, you would lose your active, inactive and archived logs if the physical disk is crashed. This is one good reason why you should archive your DB2 logs using Tivoli Storage Manager, or place your archived logs in a different physical disk from the active DB2 log directory.

4. Restore DB2 from the designated offline image copy.
5. Roll forward to the end of logs and stop.
6. Restore the corresponding Siebel files.
7. Run the Siebel File System Utility (SFSUTL) to ensure no orphan attachments exist. See 8.8.7, “SFSUTL” on page 141 for more details.
8. Consider re-extracting all Mobile Clients.

For scenario 5: In this scenario, there is no system or DB2 crash, but due to user errors, they have requested that you roll the database backward to undo the change. To accomplish this, you have to restore the database from an offline backup taken before the point in time requested by the user, and roll forward the database to the point in time requested by the customer. The action to be taken is as follows:

1. Run `db2 list history backup all for [name of database]` to identify the latest DB2 offline backup image to restore to, and take note of the earliest log that is associated with this backup image.
2. Make a copy of all the DB2 logs starting from the earliest log in step 1 to the latest logs currently in the DB2 log directory. This is required as a precaution in case you need to re-restore the database due to unforeseen circumstances, because at the end of the DB2 rollforward process, DB2 will truncate the last log.
3. Restore DB2 from the designated offline image copy.
4. Roll forward to the desired point in time and stop.

Note

You can initiate point-in-time recovery using either the DB2 Control Center or via the DB2 command line, but keep in mind this difference:

- DB2 Control Center uses *local time* for point-in-time recovery
- If you are initiating point-in-time recovery via the DB2 command line, you have to use Coordinated Universal Time instead of local time.

5. Restore the corresponding Siebel files.
6. Run the Siebel File System Utility (SFSUTL) to ensure no orphan attachments exist. See 8.8.7, “SFSUTL” on page 141 for more details.
7. Consider re-extracting all Mobile Clients.

8.6.3.3 Backing up the primary Development Enterprise

Backing up Development Enterprise is important, since the Development Enterprise contains the *master* Siebel Repository. We recommend that you do *daily* backups of the development database, enable roll forward recovery (archiving logging), and also export the Repository using the EXPREP.KSH script.

8.7 Other DB2 tools/utilities

Various DB2 tools or utilities can be used in a Siebel environment. A list follows. For detailed information, see Appendix C, “DB2 utilities” on page 183.

- DB2 Control Center

- Visual Explain
- Index Advisor utility
- DB2 Governor
- DB2 Runstats
- DB2 Reorg
- DB2 Reorgchk
- Online index reorg
- db2move
- db2dart
- db2look

8.8 Other Siebel tools/utilities

In this section, we briefly describe various Siebel utilities and scripts. For more information, refer to *Siebel Tools Guide*, 10PA1-ST00-06001.

8.8.1 Ddldict

This utility exports the Siebel repository schema to a text file. This is useful in a database migration process.

Following is an example:

```
ddldict /U sadmin /P [password] /C [ODBC Data Source] /D dbo /F
schema.ddl /E Y /L ddldict.log
```

8.8.2 Ddlimp

This utility builds the Siebel repository from a text file generated by ddldict.

Following is an example:

```
ddlimp /U siebel /P [passowrd] /C [ODBC Data Source] /D SIEBEL /F
schema.ddl /G sse_role /ddlimp.log
```

8.8.3 Dataexp

This utility exports data at the row and column level from source data to a output file.

Following is an example:

```
dataexp /U siebel /P [password] /C [ODBC Data Source] /D siebel
/F data.dat /I param.ctl /L dataexp.log
```

8.8.4 Dataimp

This utility imports data at the row and column level from an input file generated by dataexp.

Following is an example:

```
dataimp /U siebel /P [password] /C [ODBC Data Source] /D siebel /F
data.dat /I param.ctl /L dataimp.log
```

8.8.5 EXPREP.KSH

This script is used to export a Siebel Repository (using REPIMEXP.EXE) from the Siebel Enterprise database to a file. It can be used as a method to back up the Repository in a Siebel Enterprise. For example, run EXPREP.KSH and then save the output file to a different file server.

8.8.6 DEV2PROD.KSH

This script is used to migrate a Siebel Repository from one Siebel Enterprise (source) to another Siebel Enterprise (target), and then synchronize the target's physical schema with the tables and indices defined in the new or updated Siebel Repository. For example, migrate the Repository from Development to Test, Test to Production, etc.

8.8.7 SFSUTL

The utility SFSUTL.EXE in siebsrvr\bin can be used to verify the integrity between database records and files in the Siebel File System, or remove files from the Siebel File System that no longer have a corresponding database record. Note the difference between the two modes of operation: verify/report and move/delete.

Verify/report mode usage

To use the utility in verify/report mode, enter a command string similar to the following:

```
sfsutl /u sadmin /p sadmin /c siebsrvr_siebel /d dbo /f e:\siebfile /x
c:\sfsutl_output.txt
```

There are three key statuses to note:

- Keeping - The file corresponds to a database record.

- Discard - The file does not correspond to a database record, and would be moved/deleted if SFSUTL was run in that mode.
- Missing - An expected file does not exist for a database record.

Move/delete mode usage

To use the utility in move/delete mode:

- Create a separate/new folder (this will be used by SFSUTL when moving the discarded files).

We recommend that you do not create this new folder as a subfolder of the existing Siebel File System. For example, if the Siebel File System is e:\siebfile, then perhaps create the new folder as e:\siebfilediscards.

Enter a command string similar to the following:

```
sfsutl /u sadmin /p sadmin /c siebsrvr_siebel /d dbo /f e:\siebfile /m
e:\siebfilediscards /x c:\sfsutl_output.txt
```

Note that the key difference between the two modes is the `/m` switch. If you do not include it, SFSUTL will run in the verify mode. If you do include it, then SFSUTL will actually move any files with a Discard status to the folder identified in the /m switch.

Also note that SFSUTL will never actually delete any files, it only moves them. However, we recommend that you perform a backup of any production Siebel File System before using the SFSUTL utility.

Chapter 9. Performance tuning tips

In this chapter we describe some DB2 performance tips for running Siebel Applications efficiently.

9.1 DB2 database design for optimal performance

In this section we describe the placement of tablespaces and DB2 logs, the design of bufferpools, and the setting of DB2 parameters.

9.1.1 Physical database design

The physical database design is concerned with the mapping of tables to tablespaces, the layout of the tablespace containers, the storage parameters for tables, tablespaces and containers, etc.

Get the following hardware information for the physical database design:

- Number of physical disks and logical disks available for DB2 UDB
- Number of file systems and/or raw volumes available for DB2 containers
- I/O fetch size of the operating system, disk controllers and the disks/disk subsystem
- Table sizes for key/critical/large tables
- Backup strategy

  This will determine the number of log files that need to be kept on the disk, and therefore will determine storage requirements.

9.1.1.1 Tablespaces

If possible, distribute tablespaces across multiple disks, and try not to use the same disk for multiple tablespaces.

Use multiple tablespaces to separate tables from indexes and to also separate the database objects by their size, growth, concurrency and high amount of inserts/updates/deletes. For example, use separate table and index tablespaces for key/large tables (for example, the top 20 tables) and their indexes.

Also use separate tablespaces for the EIM interface tables, as they have a high number of inserts, deletes and truncations. This allows having tablespaces with default storage characteristics (for example, extent size) that are suitable for the tables that they would contain (for example, large extent size for tablespaces with big tables, and smaller extent size for tablespaces that will contain all the remaining smaller tables).

We recommend that you have at least the following tablespaces for Siebel:

- A tablespace for large tables
- A tablespace for indexes of the above large tables
- A tablespace for EIM tables
- A tablespace for EIM indexes
- A tablespace for all other remaining tables
- A tablespace for indexes of all the remaining tables

The easiest time to put the tables in a different tablespace is immediately after finishing the installation; create new tablespaces and use export/import utilities or the Copy Table tool in the Control Center.

Note that if RAID disk subsystems are utilized for tablespace containers, then the DB2 registry parameter DB2_STRIPED_CONTAINERS should be set to ON, so that the container tag page is actually an extent and not a single page.

Another DB2 registry parameter, DB2_PARALLEL_IO, turns on parallel I/O (when necessary) for tablespaces with even one container. This is under the assumption that the container is on multiple disk spindles and therefore can sustain parallel I/Os. The DB2_PARALLEL_IO registry variable should be set to specify the tablespaces (either all, or a specific one) that have containers on RAID devices.

Select a page size and extent size for tablespaces that take advantage of the stripe depth/size for RAID storage. This makes I/O operation efficient. For example, if the RAID system is designed so that a single disk fetch results in a fetch operation to 4 disks and the fetch size for each disk is 2 KB, then each I/O operation results in an 8 KB data transfer. In this case, an 8 KB page size is more efficient than 4 KB. For very large tables that get heavily accessed, an even larger page size (in multiples of the RAID strip size) may also be suitable. If the necessary information is not available, then a recommended page size is 8 KB or 16 KB.

DB2 distributes extents for a table across all the available containers for a tablespace in a round-robin manner. Also, it should be noted that if a container is added to a tablespace, then DB2 would rebalance the data among all the containers (i.e., among the existing containers and the new container) immediately. This can adversely affect performance if it is done at peak database activity.

DMS and SMS considerations for tablespaces

We recommend that you use DMS tablespaces, because they reduce dynamic space allocation at the operating system level, allow you to add more containers to tablespaces in the future and, by using raw devices, can bypass the overhead (if any) of the operating system's file system buffers.

In general, DMS storage is recommended over SMS for tablespaces because:

- SMS does not allow adding containers (directories) to tablespaces after they have been created. The only option then is to expand the file system of the container.
- SMS stores table and index objects in the same container (directory).
- In an SMS tablespace, if any container becomes full, then the entire tablespace is considered full and the only option is to expand the file system for the directory, which is not the case with DMS tablespaces.
- In an SMS tablespace, there is one file in the container (directory) per object. Also, the object grows one page at a time, unless multipage file allocation has been enabled using the db2empfa utility. Even using db2empfa has other considerations when you turn it off, such as a reorg required or strongly suggested.

DMS tablespace considerations

- For DMS table containers, do not define more than one container per RAID unit.
- Make the extent size a multiple of the RAID stripe size, if possible.
- Make the container size and prefetch size a multiple of the extent size.
- Use the DB2 registry variable DB2_STRIPED_CONTAINERS to align extents to the RAID stripe size.
- Use the DB2 registry variable DB2_PARALLEL_IO to enable parallel disk I/O.
- For tables and indexes in DMS tablespaces, choose extent sizes based on the table size estimates given in the database sizing earlier. This will reduce dynamic extensions.

Temporary tablespace considerations

Temporary tablespaces should be SMS. Note that DB2 allocates temporary space in a round-robin fashion if there are two or more tablespaces. Even if one tablespace is full, it can cause a problem, as DB2 cannot do a round-robin allocation and no more temporary objects can be created. For this reason, using a single, large temporary tablespace for each page size is recommended.

If queries are executing against tables in tablespaces that are defined with a page size of larger than the default 4 KB, some of them may fail because of the lack of a temporary tablespace defined with a larger page size (for example, an ORDER BY on 1012 columns). You may need to create a temporary tablespace with a larger page size (8 KB, 16 KB, or 32 KB). In fact, any Data Manipulation Language (DML) statement could fail unless there exists a temporary tablespace with the same page size as the largest page size of user data.

9.1.1.2 Tables

We recommend that you place important Siebel tables in terms of activity on their own tablespaces and, if it is possible, spread them on several separate disk spindles. The following is a list of high-activity tables that require separate tablespaces and disks.

- S_DOCK_TXN
- S_EVT_ACT
- S_POSTN_CON
- S_DNB_ORG
- S_EVT_ACT_X
- S_CONTACT
- S_PROD_DEFECT
- S_ORG_TERR
- S_COLUMN
- S_ACCNT_POSTN
- S_ESCL_REQ
- S_OPTY_POSTN

This list of tables depends on the functional modules that you purchased from Siebel, and also on your customizations for your Siebel implementation.

In addition to the base and intersection tables, the following are other Siebel tables that have high activity and/or grow significantly:

- Siebel transaction log tables (S_TXN_*)

- Activities and events tables (S_EVT_ACT)
- Workflow-related tables (S_ESCL_REQ, S_ESCL_STATE, S_ESCL_ACTN_REQ)

9.1.2 DB2 log files

We recommend that you allocate the DB2 log files on a physical disk that does not have high I/O activity. For instance, avoid putting the logs on the same disk as the operating system or high volume databases. This will allow for efficient logging activity with a minimum of overhead (such as waiting for I/O). You can track the number of I/Os related to database logging by using a database system monitor. You can change the log path to different disks by using Configure... from the Control Center menu, or by running the following command:

```
db2 update db cfg for <database name> using newlogpath <new directory>
```

Consider having more primary logs than the default setting of 10:

```
db2 update db cfg for <database name> using logprimary <new logfile number>
```

Primary log files are preallocated, while secondary log files are created if primary logs are not sufficient. Therefore, you will reduce the overhead of creating secondary logs during high DB2 activity if you have sufficient primary logs. Notice that the maximum number of logs (primary and secondary) is 128. We recommend that you have around 4 GB of DB2 log files for Siebel Applications.

9.1.3 DB2 bufferpool management

The bufferpool is one of the most important tuning areas in the DB2 environment. This is the area of memory into which database pages (table rows or indexes) are temporarily read, and where they are manipulated.

All bufferpools are available to all applications using the database. The purpose of the bufferpool is to improve database performance. Data can be accessed much faster from memory than from disk. Therefore, the more data (rows and indexes) the database manager is able to read from or write to memory, the better the database performance.

Bufferpool hit ratio

A measure of the effectiveness of the bufferpool is the bufferpool hit ratio. The more data found in the bufferpool, the better it is for performance since this avoids unnecessary I/O operations. The bufferpool hit ratio should be as close to 100% as possible for all bufferpools. A good ratio is above 95%.

You can measure the bufferpool hit ratio (and many other parameters) with the DB2 Performance Monitor (available from the Control Center), or the SNAPSHOT command from the command line:

```
db2 get snapshot for bufferpools on <database-alias>
```

It is calculated as:

```
1-(data + index physical pages read)/(data + index logical pages read)
```

Bufferpool size

DB2 does not use all of its memory just for bufferpools. Bufferpool space is allocated from the same shared memory segment as the lock list, database heap, and utility heap (determined by configuration parameters locklist, dbheap, and util_heap_sz, respectively).

Because of this, you should try to get as optimal a size as possible for your bufferpool. If you achieve a bufferpool hit ratio of close to 100%, you may consider reducing the size of the bufferpool and trying to get the same result with a smaller size. The optimal situation is the point when the bufferpool is as small as possible, and the bufferpool hit ratio is still close to 100%.

If your tables are too big to fit into bufferpools, try at least to put indexes with heavy hit ratios in separate bufferpools.

IBM DB2 Universal Database Administration Guide Version 6 provides more information on this subject. See Part 7, called “Tuning and Configuring Your System”, and refer to Chapter 23, “Operational Performance - How DB2 Uses Memory”.

If your usage of EIM processing is high, consider having separate bufferpools for EIM tables. You will limit the impact of EIM processing on online users by flushing the bufferpools of EIM processes.

9.1.3.1 Shared memory considerations on UNIX

On a UNIX-based system, the database shared memory (database global memory) that can be allocated is constrained due to addressing schemes in which shared memory segments start at specific addresses. The database shared memory is memory that are allocated during the first connect to the database.

Shared memory is allocated both at instance activation time and database activation time. At instance activation time, memory is allocated in Global control blocks. At database activation time, shared memory (like bufferpool, locklist, database heap) is allocated.

For AIX, the shared memory limitation is approximately 1.75 GB.

The memory model is different on Windows NT. You can have 2 to 3 GB for a user, and 1 to 2 GB for a kernel, for a maximum of 4 GB.

Very often, we want to maximize the database bufferpools in order to avoid excessive I/O costs, but we also must be aware of the inherent shared memory limitations; otherwise, we will likely get an SQL10003C error: `Not enough system resources`.

To estimate the shared memory used, do the following:

1. Calculate the bufferpool usage, with:

   ```
   db2 select bpname, npages, pagesize from syscat.bufferpools
   ```

   Then sum up all the bufferpools.

2. From the output of the command:

   ```
   db2 get db cfg for <database alias>
   ```

   Sum up the following heaps:

   - Locklist
   - Utility heap
   - Database heap
   - Package heap
   - Logbufsz
   - Catalog cache size

The cumulative sum should be within the shared memory constraints. Use suggested values from 6.4, “DB2 parameters for a Siebel environment” on page 98 for these parameters.

9.1.4 Performance problems and possible solutions

In this section, we list possible problems that a DBA may encounter, their symptoms and possible solutions.

Problem 1: The system is paging

If there is no obvious cause, such as trying to have 1000 agents when the machine has only 4 GB of memory, a probable cause is the size of the agents and the sorts they do.

In this case, try decreasing DB2MEMMAXFREE and SORTHEAP to one of the values suggested in 6.4.3.1, “Memory management

(DB2MEMMAXFREE)” on page 106 and 6.4.2.1, “Sort heap size (SORTHEAP)” on page 103.

Problem 2: Huge I/O wait
Make sure that it is not the case that everybody waits on one disk. Use the iostat command to isolate the busiest disks. You need to try to map them to tablespaces (through tablespace containers), and if it turns out that there is a direct relationship between one of the tablespaces and the busiest disks, then add more containers to that tablespace.

If this is not the case, try to see if DB2 prefetchers are very busy (again, using the ps command), and if they are busier than the other agents, you may want to experiment with decreasing the number of prefetchers (even to the extreme, which is NUM_IOSERVERS=1 in db cfg).

If neither of the above is the case, it is possible that some bad queries are causing this I/O wait by doing tablescans, so try to isolate those queries, fix them, and see if the problem goes away.

9.2 How to improve SQL performance

In many cases, one bad SQL query can downgrade the performance of the whole system. In that case you should monitor SQL queries executed by Siebel clients on DB2.

We will consider two cases:

- A Siebel Connected client is executing SQL directly on DB2.
- A Siebel Thin client is passing requests to the Siebel server, which then executes SQL against DB2.

9.2.1 Siebel Connected clients

While using Siebel Connected clients, a particular view, query, or user action might not provide acceptable performance. The performance of a particular action may vary or degrade over time. This section describes a process by which you can isolate the components that contribute to the total response time to the user. Figure 51 on page 151 lists the steps involved from the time a user begins waiting until control is returned to him.

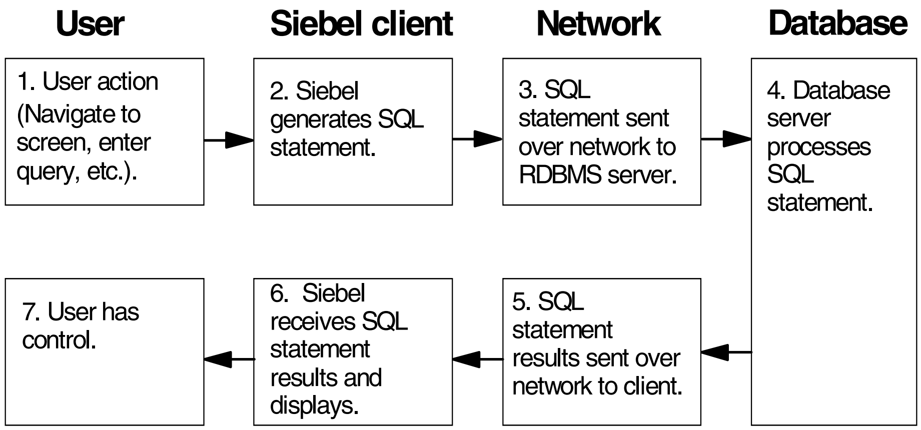

User
Siebel client
Network
Database
1. User action (Navigate to screen, enter query, etc.).
2. Siebel generates SQL statement.
3. SQL statement sent over network to RDBMS server.
4. Database server processes SQL statement.
5. SQL statement results sent over network to client.
6. Siebel receives SQL statement results and displays.
7. User has control.

Figure 51. Components that contribute to user response time

The user wait time is the time from the completion of step 1 to the completion of step 6. Note that this process applies to all supported database vendors; however, the methods for diagnosing problems with the database engine and SQL statement processing will vary by vendor.

9.2.1.1 Processing breakdown

- Steps 1 and 7 represent time spent by the user not waiting.
- Steps 2 and 6 represent time spent processing by the Siebel client application; this time is generally negligible and, with rare exceptions, not tunable through configuration.
- Steps 3 and 5 represent time spent to transfer data over the network.
- Step 4 represents time spent processing at the database.

Determining Siebel client processing time

Generate the SQL for the action in question (if you are unfamiliar with generating SQL, see 9.2.1.3, “Generating SQL output from the Siebel client application” on page 154). Use the db2batch utility on the same client machine that demonstrates the poor performance, and execute the SQL statement that was generated. Refer to C.7, “db2batch” on page 194 to see how to use this tool.

For SQL queries, measure the time it takes for the first record to be displayed. For other types of SQL statements (such as inserts, updates etc.), measure how long it takes for the statement to complete processing.

Compare this time to the user wait time. The difference is the *time spent processing by the Siebel client application*. The SQL generated from this step will be used in the subsequent steps also.

Determining network time

Transfer the same SQL file used in the previous step to the DB2 server. Run the statement through db2batch on the DB2 server and measure the processing time.

For SQL queries, measure the time it takes for the first record to be displayed. For other types of SQL statements (such as inserts, updates etc.), measure how long it takes for the statement to complete processing.

Compare the times between this step and the prior step. The difference is the *amount of time introduced by the network*. If the difference is large, you should investigate the network between the DB2 server and the affected clients. If the total processing time for the SQL statement running on the DB2 server is short, you most likely do not need to investigate the database processing time.

Determining database processing time

If the network time is short, you can do the work for this step from either the client machine or the DB2 server, whichever you prefer. There are several likely causes of poor performance in database processing:

- The SQL statement.
- The database server is not tuned, or is overloaded.
- The hardware on which the database server resides is not tuned, or is overloaded.

Keep in mind the following important issues regarding the tuning of individual SQL:

- The correct optimization class must be determined and set accordingly, in building the DB2 plan.
- The presence of the OPTIMIZE FOR 1 ROWS clause must also be determined and used accordingly, in building the DB2 plan.
- Sometimes the performance problem is not access path-related, but is due instead to the values used in the query. If the plan (access path) seems optimal, make sure the values substituted for the parameters markers are reasonable.

  For example, we have observed Siebel-generated queries that have predicates like NAME >= ? and at runtime use 0x01 for the ?, which makes the predicate basically useless since the predicate qualifies every single row. In such cases, it may be possible to work with Siebel experts to remove the predicate from the query.

The following section describes what is likely to indicate one cause or another.

9.2.1.2 Diagnosing poor performance - database processing

These indicators are not absolute; they are simply likely pointers to the possible problem. Some potential causes are described, and steps to diagnose and resolve the potential problems are listed.

SQL Statement is not tuned

Indications	Poor performance is observed for a particular view or particular query.
Causes	Configured views result in SQL that does not perform well. Data content in the Siebel database results in poor index selection.
Diagnosis	Generate the plan that the DB2 is using to execute the query. Use Visual Explain with the poorly performing SQL generated in the prior steps. The *DB2 Performance Tuning Guide* can help you in the interpretation of the plan output. If you believe that changes to indexes are necessary, review the output with a Siebel Enterprise Architect prior to recommending that an index change be made.

Database server is not tuned, or is overloaded

Indications	Poor performance is observed across all Siebel views, as well as other DB2 applications running on the same DB2 server.
Causes	Various.
Diagnosis	Review the *DB2 Performance Tuning Guide* and start with the Siebel DBA checklist items.

Hardware is not tuned or is overloaded

Indications	Poor performance is observed across all Siebel views, as well as across all applications running on the same physical server.
Causes	Lack of hardware resources (memory, I/O, CPU, etc.). Hardware not configured optimally for database.
Diagnosis	Review DB2 documentation for AIX to determine the appropriate hardware configuration.

9.2.1.3 Generating SQL output from the Siebel client application

In the command line for your Siebel application startup, use the /s filename option. This will begin spooling all of the SQL that Siebel generates to an output file named filename. If filename does not include an absolute path, it is placed in the Siebel bin directory. If you are not familiar with the SQL output, you may want to run Siebel once, up to the point of executing action with poor performance, but not actually performing the action. Then exit and save the output file. Restart the Siebel client and follow the same process, except that you should execute the poor performance action this time and then exit Siebel.

The difference between the output files is the SQL with poor performance. Often this will consist of multiple SQL statements, although frequently it is only one of the statements actually causing the poor performance. If several statements are included for the action with poor performance, perform your tests using all of them until you are able to isolate the ones causing the problem.

In order to run the generated SQL through db2batch, you will need to edit the output file to select just the desired portion of code.

In addition, you can use DB2 Event Monitor to get a summary of activity at the completion of events such as statement execution, transaction completion, or when an application disconnects. For detailed information about Event Monitor, refer to *DB2 UDB V6 System Monitor Guide and Reference*, SC09-2849.

9.2.2 Siebel Thin clients

Siebel Thin clients do not execute SQL directly on DB2. You cannot collect SQL on the client side. In this case, use the Event Monitor to monitor SQL statements. Issue the following commands to get the Event Monitor for statement information:

```
db2 connect to <database alias> user <userid> using <password>

db2 create event monitor <name> for statements write to file '<directory>'

db2 set event monitor <name> state 1

db2 flush event monitor <name>

db2evmon -db <database alias> -evm <name>
```

You can also use the graphical tools in Event Monitor and Event Analyzer available in the Control Center for the same purpose.

9.2.3 Analyzing the SQL

After you get your troubled SQL statements, you can use the program written in C provided in Appendix D, "How to test your SQL" on page 197 to do the following:

1. Create the Explain table by executing EXPLAIN.DDL in the sqllib\misc directory.
2. Change the database name, user name and password in the db2 connect section of the program.
3. Change your SQL in the *strcpy* section of the program.
4. Run the program.
5. Review the Explain tables.

9.3 Database reorganization

Run reorgchk on a regular basis to determine which tables require reorganization:

```
db2 reorgchk update statistics on table all
```

Run reorg on tables that require reorganization. Reorganizing large tables may be time-consuming and require too much temporary space. Reorg should be run on a weekly or monthly basis, depending on table activity.

```
db2 reorg table siebel.s_evt_act index siebel.s_evt_act_f1
```

After reorganizing a table, we recommend that you use the runstats command to provide the optimizer with current statistics.

9.4 The runstats utility vs. updatestats

DB2 catalog statistics should be updated in a timely manner so that the DB2 optimizer uses this information when determining access paths to the data. The statistics include the number of records, number of pages, average record length, and number of unique values, etc. In a Siebel on DB2 UDB environment, there are two methods to update these statistics:

- By running the DB2 runstats utility
- By running the updatestats.ksh script provided by Siebel

However, the choice of which utility to use and when to run it is up to you.

The updatestats script

The updatestats.ksh script invokes a C stored procedure called SIEBSTAT via the ODBC interface, and updates statistics for those tables listed in the corresponding updatestats.sql file.

It scans the tables and indexes to gather statistics about each DB2 object, such as the number of rows in the table, the number of unique key values in the index, etc. You can create multiple updatestats scripts to update statistics on groups of tables and indexes by editing the updatestats.sql file.

We recommend that you use this script if you are not experienced in DB2 database administration; however, there is no functionality difference between DB2 runstats and the updatestats.ksh script.

The DB2 runstats utility

In most DB2 installations, the DB2 runstats utility has been used when a table has had many updates, or after reorganizing a table. With Siebel Applications, we recommend that you use the following options:

- The `with distribution` option specifies that distribution statistics are requested.
- The `and indexes all` option specifies that statistics will be updated on both the table and its indexes.
- The `shrlevel change` option specifies that other users can read from and write to the table while statistics are calculated.

The following example shows the runstats update statistics about the physical characteristics of a table and the associated indexes:

```
db2 runstats on table siebel.s_evt_act with distribution and indexes all
shrlevel change
```

An experienced DB2 DBA may tend to prefer the runstats utility over the updatestats script.

Note

Do not use the runstats utility on tables S_DOCK_INIT_ITEM and S_ESCL_LOG. For these tables, Siebel preloads some statistics for performance at installation time. The Siebel dbxtract process will run much slower if you run runstats on this table.

If the DBA accidentally updates the statistics for these two tables, then you can run the SQL in the file loadstats.sql to reestablish the predicated statistics. This file is in the folder <dbsrvr>\db2udb.

9.5 The DB2 Governor

The DB2 Governor can be used to decrease the priority of the EIM applications and the Workflow applications. End-user tasks and queries could have a more consistent rate of response.

Using the DB2 Governor, different applications can be placed into different classes, and the Governor will ensure that an equal amount of resource is given to each class, based on the amount of CPU and I/O resource consumed and how quickly each query is expected to complete. Therefore, a query which is estimated to complete quickly will be given a larger amount of resource.

See the *DB2 Command Reference* and the *DB2 Administration Guide* for details about how to run the DB2 Governor and understand the output.

9.6 The Index Advisor utility

The Index Advisor utility can be used in cases where you receive a troubled SQL statement. If you already know which SQL is causing trouble, use Smart Guide to find out if the right indexes were used.

Appendix A. Existing customer configurations

In this appendix we describe the existing system configurations of some of our Siebel DB2 UDB customers. This information was collected from the field, and shows you what other Siebel DB2 UDB customers are doing with their parameters.

A.1 DB2 parameters

In this section we list the DB2 parameters of two different customers.

Table 35. DB2 Database Manager configuration parameters

Database Manager config parameters	Customer 1	Customer 2
UDF_MEM_SZ	1024	1024
SHEAPTHRES	100000	160000
DIR_CACHE	YES	YES
ASLHEAPSZ	15	15
RQRIOBLK	65535	65535
QUERY_HEAP_SZ	16384	8192
MAXAGENTS	1000	2000
NUM_POOLAGENTS	360 (Calculated)	240 (calculated)
NUM_INITAGENTS	0	0
MAX_COORDAGENTS	MAXAGENTS	MAXAGENTS
MAXCAGENTS	MAX_COORDAGENTS	MAX_COORDAGENTS
INDEXREC	RESTART	RESTART
MAX_QUERYDEGREE	1	1
INTRA_PARALLEL	NO	NO

Table 36. DB2 environment variables

DB2 environment variables	Customer 1	Customer 2
DB2_STRIPED_CONTAINERS	ON	ON
DB2_PARALLEL_IO	*	*
DB2MEMMAXFREE	3000000	3000000
DB2MEMDISCLAIM	YES	YES
DB2_HASH_JOIN	NO	NO
DB2_RR_TO_RS	YES	YES
DB2_MMAP_WRITE	OFF	OFF
DB2_MMAP_READ	OFF	OFF
DB2_CORRELATED_PREDICATES	YES	YES

Table 37. DB2 database configuration parameters

Database config parameters	Customer 1	Customer 2
DFT_DEGREE	1	1
DFT_QUERYOPT	5	5
DBHEAP	7429	8600
CATALOGCACHE_SZ	5558	5558
LOGBUFSZ	128	512
UTIL_HEAP_SZ	5000	35000
BUFFPAGE	1000	500000
LOCKLIST	37000	20000
APP_CTL_HEAP_SZ	152	152
SORTHEAP	32	32
STMTHEAP	8192	8192
APPHEAPSZ	2500	2500
PCKCACHESZ	2048	2048
STAT_HEAP_SZ	8000	8000

Database config parameters	Customer 1	Customer 2
DLCHKTIME	10000	100000
MAXLOCKS	70	20
LOCKTIMEOUT	300	300
CHNGPGS_THRESH	60	60
NUM_IOCLEANERS	18	35
NUM_IOSERVERS	42	32
INDEXSORT	YES	YES
SEQDETECT	YES	YES
DFT_PREFETCH_SZ	128	128
MAXAPPLS	2000	2000
AVG_APPLS	10	30
MAXFILOP	500	500
LOGFILSIZ	8000	20000
LOGPRIMARY	10	50
LOGSECOND	100	0
MINCOMMIT	1	1
SOFTMAX	80	60
LOGRETAIN	ON	OFF

A.2 Customer 1 - detailed information

Table 38. Customer 1 - bufferpool configuration

Bufferpool name	Page size	Number of pages
IBMDEFAULTBP	4 KB	10000
BUF4K	4 KB	90000
BUFIDX4K	4 KB	90000
BUF32K	32 KB	6000

Table 39. Customer 1 - tablespace configuration

Tablespace name	Page size	Extent size	Prefetch size	Bufferpool name	Storage type	Tablespace size
TEMP4K	4 KB	8	64	BUF4K	SMS	
SIEBEL_4K	4 KB	8	64	BUF4K	DMS	4 containers 2621440 pages
S_DNB_ORG_TBS	4 KB	8	64	BUF4K	DMS	4 containers 153600 pages
S_DOCK_TXN_TBS	4 KB	8	64	BUF4K	DMS	4 containers 2621440 pages
S_EVT_ACT_X_TBS	4 KB	8	64	BUF4K	DMS	4 containers 131072 pages
S_CONTACT_TBS	4 KB	8	64	BUF4K	DMS	4 containers 51200 pages
S_ORG_TERR_TBS	4 KB	8	64	BUF4K	DMS	4 containers 76800 pages
S_POSTN_CON_TBS	4 KB	8	64	BUF4K	DMS	4 containers 327680 pages
S_COLUMN_TBS	4 KB	8	64	BUF4K	DMS	4 containers 25600 pages
S_ACCNT_POSTN_TBS	4 KB	8	64	BUF4K	DMS	4 containers 153600 pages

Tablespace name	Page size	Extent size	Prefetch size	Bufferpool name	Storage type	Tablespace size
S_ESCL_REQ_TBS	4 KB	8	64	BUF4K	DMS	4 containers 12800 pages
S_OPTY_POSTN_TBS	4 KB	8	64	BUF4K	DMS	4 containers 6400 pages
SIEBEL_4KL	4 KB	8	64	IBMDEFAULTBP	DMS	4 containers 655360 pages
SIEBEL_IDX	4 KB	8	64	BUFIDX4K	DMS	4 containers 2621440 pages
TEMP32K	32 KB	2	8	BUF32K	SMS	
SIEBEL_32K	32 KB	8	64	BUF32K	DMS	4 containers 81920 pages
S_EVT_ACT_TBS	32 KB	8	64	BUF32K	DMS	4 containers 65536 pages
S_PROD_DEFECT_TBS	32 KB	8	64	BUF32K	DMS	4 containers 12800 pages

Table 40. Customer 1 - top ten tables

Table name	Number of rows
S_POSTN_CON	12893971
S_ACCNT_POSTN	4152221
S_EVT_ACT	3939168
S_EVT_ACT_X	2328152
S_ORG_TERR	2090550
S_OPTY_POSTN	1642705
S_DNB_CONTACT	95331
S_DNB_CON_MRC	948726
S_DNB_ORG_SIC	872005
S_CAMP_CON	765137

Table 41. Customer 1 - top twenty indexes

Index name	Fullkeycard	Firstkeycard
S_POSTN_CON_P1	12893971	12893971
S_POSTN_CON_U1	12893971	645438
S_POSTN_CON_M1	11169040	4589
S_POSTN_CON_M2	6877060	2
S_ACCNT_POSTN_P1	4152221	4152221
S_ACCNT_POSTN_U1	4152221	353291
S_ACCNT_POSTN_M1	4062526	3656
S_EVT_ACT_P1	3939168	3939168
S_EVT_ACT_U1	3939168	3939168
S_ACCNT_POSTN_M2	3230990	2
S_EVT_ACT_X_P1	2328152	2328152
S_EVT_ACT_X_U1	2328152	2328152
S_ORG_TERR_P1	2090550	2090550
S_ORG_TERR_U1	2090550	312627
S_OPTY_POSTN_P1	1642705	1642705
S_OPTY_POSTN_U1	1642705	71667
S_EVT_ACT_M6	1456176	6604
S_EVT_ACT_M2	1437190	6370
S_DNB_CONTACT_P1	953311	953311
S_DNB_CONTACT_U1	953311	322645

Table 42. Customer 1 - tablespaces/table mapping

Tablespace name	Table name
S_ACCNT_POSTN_TBS	S_ACCNT_POSTN
S_COLUMN_TBS	S_COLUMN
S_CONTACT_TBS	S_CONTACT
S_DNB_ORG_TBS	S_DNB_ORG
S_DOCK_TXN_TBS	S_DOCK_TXN_LOG
S_ESCL_REQ_TBS	S_ESCL_REQ
S_EVT_ACT_TBS	S_EVT_ACT
S_EVT_ACT_X_TBS	S_EVT_ACT_X
S_OPTY_POSTN_TBS	S_OPTY_POSTN
S_ORG_TERR_TBS	S_ORG_TERR
S_POSTN_CON_TBS	S_POSTN_CON
S_PROD_DEFECT_TBS	S_PROD_DEFECT
SIEBEL_32K	EIM_ACCNT_DTL, EIM_ACCNT_SRC
SIEBEL_32K	EIM_ACCOUNT, EIM_ACCOUNT1
SIEBEL_32K	EIM_ACT_DTL, EIM_ACTIVITY
SIEBEL_32K	EIM_ACTIVITY2, EIM_AGREEMENT
SIEBEL_32K	EIM_AGREEMENT1, EIM_ASGN_GRP
SIEBEL_32K	EIM_ASSET, EIM_ASSET1
SIEBEL_32K	EIM_BASELINE, EIM_CALL_LST
SIEBEL_32K	EIM_COMM, EIM_CON_DTL
SIEBEL_32K	EIM_CON_PRDINT, EIM_CONSUME
SIEBEL_32K	EIM_CONTACT, EIM_CONTACT1
SIEBEL_32K	EIM_CONTACT2, EIM_CS_QUEST
SIEBEL_32K	EIM_CS_RUN, EIM_CT_ACT
SIEBEL_32K	EIM_CT_APLT, EIM_CT_RULE
SIEBEL_32K	EIM_CTLG_CAT, EIM_DEFECT

Tablespace name	Table name
SIEBEL_32K	EIM_DMNDCRTPRG, EIM_EMP_DTL
SIEBEL_32K	EIM_EMPLOYEE, EIM_EMPLOYEE1
SIEBEL_32K	EIM_ENTLMNT, EIM_EXP_ITEM
SIEBEL_32K	EIM_EXP_RPT, EIM_FN_ASTLB
SIEBEL_32K	EIM_FN_NEED, EIM_FUL
SIEBEL_32K	EIM_INV_TXN, EIM_INVC_ITEM
SIEBEL_32K	EIM_INVLOC, EIM_INVOICE
SIEBEL_32K	EIM_ISS, EIM_LITERATURE
SIEBEL_32K	EIM_MSG, EIM_OPTY
SIEBEL_32K	EIM_OPTY_DTL, EIM_OPTY1
SIEBEL_32K	EIM_ORDER, EIM_ORDER_ITEM
SIEBEL_32K	EIM_ORG_INT_DTL, EIM_PART_RPR
SIEBEL_32K	EIM_PRDINT_DTL, EIM_PRI_LST
SIEBEL_32K	EIM_PROD_INT, EIM_PROD_INT1
SIEBEL_32K	EIM_PROD_SHIP, EIM_PROJECT
SIEBEL_32K	EIM_PROJECT1, EIM_PROJECTDTL
SIEBEL_32K	EIM_PROJITEM, EIM_PROPOSAL
SIEBEL_32K	EIM_PRSP_CON, EIM_QUOTE
SIEBEL_32K	EIM_QUOTE_IT, EIM_QUOTE_TERM
SIEBEL_32K	EIM_RMA, EIM_SR_DTL
SIEBEL_32K	EIM_SRC, EIM_SRC_DTL
SIEBEL_32K	EIM_SRV_REQ, EIM_SRV_REQ1
SIEBEL_32K	EIM_TARGET, EIM_TIMESHEET
SIEBEL_32K	S_ACCNT_SRC_IF, S_ACCOUNT_DTLIF
SIEBEL_32K	S_ACCOUNT_IF, S_ACT_DTLIF
SIEBEL_32K	S_ACTIVITY_IF, S_ACTIVITY1_IF
SIEBEL_32K	S_ACTIVITY2_IF, S_AGREEMENT_IF

Tablespace name	Table name
SIEBEL_32K	S_ASGN_GRP_IF, S_ASSET_IF
SIEBEL_32K	S_CALL_LST, S_CALL_LST_IF
SIEBEL_32K	S_CHART, S_CON_PRDINT_IF
SIEBEL_32K	S_CONTACT_DTLIF, S_CONTACT_IF
SIEBEL_32K	S_CONTACT_T, S_CONTACT1_IF
SIEBEL_32K	S_CONTACT2_IF, S_CS_RUN_IF
SIEBEL_32K	S_CT_ACTION, S_CT_RULE
SIEBEL_32K	S_CUST_SURVEY, S_DEFECT_IF
SIEBEL_32K	S_DOCK_VIS_RULE, S_EMP_DTLIF
SIEBEL_32K	S_EMPLOYEE, S_EMPLOYEE_IF
SIEBEL_32K	S_EXPENSE_IF, S_FIELD
SIEBEL_32K	S_INT_COMPMAP, S_LIT
SIEBEL_32K	S_LITERATURE_IF, S_MDF_IF
SIEBEL_32K	S_MSG, S_OPTORG_CMPT_T
SIEBEL_32K	S_OPTY, S_OPTY_CON
SIEBEL_32K	S_OPTY_DTLIF, S_OPTY_IF
SIEBEL_32K	S_OPTY1_IF, S_ORDER
SIEBEL_32K	S_ORDER_IF, S_ORDER_ITEM_IF
SIEBEL_32K	S_PART_RPR_IF, S_POSITION_IF
SIEBEL_32K	S_PPSL_SECT, S_PROD_DEFECT_OLD
SIEBEL_32K	S_PROD_INT_IF, S_PROD_INT1_IF
SIEBEL_32K	S_PROJECT_DTLIF, S_PROJECT_IF
SIEBEL_32K	S_PROPOSAL_IF, S_PRSP_CON_IF
SIEBEL_32K	S_QUOTE_IF, S_QUOTE_ITEM_IF
SIEBEL_32K	S_QUOTE_TERM_IF, S_RESITEM
SIEBEL_32K	S_SR_DTLIF, S_SRC
SIEBEL_32K	S_SRC_DTLIF, S_SRC_IF

Tablespace name	Table name
SIEBEL_32K	S_SRV_REQ_IF

Note: All other tables are stored in tablespace SIEBEL_4K.

A.3 Customer 2 - detailed information

Table 43. Customer 2 - bufferpool configuration

Bufferpool name	Page size	Number of pages
IBMDEFAULTBP	4 KB	2000
BUF4K	4 KB	40000
BUFTEMP4K	4 KB	2000
BUFBAS4KX	4 KB	30000
BUF8K	8 KB	29000
BUFTEMP8K	8 KB	2000
BUFFIF8K	8 KB	20000
BUFBAS8KX	8 KB	20000

Table 44. Customer 2 - tablespace configuration

Tablespace name	Page size	Extent size	Prefetch size	Bufferpool name	Storage type	Tablespace size
TEMPORARY_4K	4 KB	32	128	BUFTEMP4K	SMS	N/A
SIEBEL_4K	4 KB	32	224	BUF4K	DMS	7 containers 424288 pages
SIEBEL_4KL	4 KB	32	32	BUF4K	DMS	1 container 253000 pages
TBSGRC	4 KB	32	32	BUF4K	DMS	1 container 524288 pages
TBSGRC4KL	4 KB	32	32	BUF4K	DMS	1 container 73000 pages
IDXGRC	4 KB	32	32	BUFBAS4KX	DMS	1 container 532480 pages

Tablespace name	Page size	Extent size	Prefetch size	Bufferpool name	Storage type	Tablespace size
GRC_ORGEXT_IDX	4 KB	32	96	BUFBAS4KX	DMS	3 containers 524288 pages
GRC_INDEX_X	4 KB	32	288	BUFBAS4KX	DMS	9 containers 524288 pages
TEMPORARY_8K	8 KB	32	128	BUFTEMP8K	SMS	N/A
GRC_ASSET	8 KB	3 2	224	BUF8K	DMS	7 containers 1310720 pages
GRC_ASSET_X	8 KB	32	224	BUF8K	DMS	7 containers 819200 pages
GRC_SRVREQ	8 KB	32	224	BUF8K	DMS	3 containers 466944 pages
GRC_SRVREQ_X	8 KB	32	224	BUF8K	DMS	7 containers 270336 pages
GRC_EVTACT	8 KB	32	224	BUF8K	DMS	7 containers 819200 pages
GRC_EVTACT_X	8 KB	32	224	BUF8K	DMS	7 containers 819200 pages
SIEBEL_8K	8 KB	32	224	BUFFIF8K	DMS	7 containers 671744 pages
TBSGRC8K	8 KB	32	96	BUFFIF8K	DMS	3 containerS 270336 pages
SIEBEL_IDX	8 KB	32	224	BUFBAS8KX	DMS	7 containers 499712 pages
GRC_ASSET_IDX	8 KB	32	224	BUFBAS8KX	DMS	7 containers 901120 pages
GRC_SRVREQ_IDX	8 KB	32	224	BUFBAS8KX	DMS	7 containers 376832 pages
GRC_EVTACT_IDX	8 KB	32	160	BUFBAS8KX	DMS	5 containers 688128 pages

Table 45. Customer 2 - top ten tables

Table name	Number of rows
S_ASSET	80888577
S_ASSET_X	51262205
S_EVT_ACT	30111076
S_EVT_ACT_X	30108573
S_SRV_REQ_X	23217182
S_SRV_REQ	23217129
S_ASSET_ACCNT	7642661
S_ASSET_XM	7550047
S_ORG_REL	6168039
S_ADDR_ORG	4068542

Table 46. Customer 2 - top twenty indexes

Index Name	Fullkeycard	Firstkeycard
IBM_ASSET_5	80888577	5
IBM_ASSET_4	80888516	80888487
S_ASSET_U1	80887512	80887483
S_ASSET_P1	80885164	80885164
IBM_ASSET_X_3	51262205	51262205
S_ASSET_X_U1	51262204	51262204
S_ASSET_X_P1	51262201	51262201
IBM_EVT_ACT_8	30111076	1508
S_EVT_ACT_U1	30110998	30110998
S_EVT_ACT_P1	30110974	30110974
IBM_EVT_ACT_X_1	30108573	3
S_EVT_ACT_X_U1	30108572	30108572
S_EVT_ACT_X_P1	30108557	30108557

Index Name	Fullkeycard	Firstkeycard
IBM_EVT_ACT_2B	27165215	62433
IBM_EVT_ACT_1	27165206	62430
S_SRV_REQ_X_P1	23217182	23217182
S_SRV_REQ_X_U1	23217182	23217182
IBM_SRV_REQ_X_1	23217182	3
IBM_SRV_REQ_5	23217129	107
IBM_SRV_REQ_3	23217126	6

Table 47. Customer 2 - table spaces/table mapping

Tablespace name	Table name
GRC_ORGEXT	S_ORG_EXT
SIEBEL_8K	S_ACCOUNT_DTLIF,S_ACCOUNT_IF
SIEBEL_8K	S_ACT_DTLIF,S_ACTIVITY_IF
SIEBEL_8K	S_ACTIVITY2_IF,S_ASGN_GRP_IF
SIEBEL_8K	S_ASSET_IF,S_CONTACT_DTLIF
SIEBEL_8K	S_CONTACT_IF,S_CONTACT_T
SIEBEL_8K	S_CONTACT1_IF,S_CONTACT2_IF
SIEBEL_8K	S_DEFECT_IF,S_EMP_DTLIF
SIEBEL_8K	S_EMPLOYEE_IF,S_EXPENSE_IF
SIEBEL_8K	S_FUL_IF,S_INVLOC_IF
SIEBEL_8K	S_LITERATURE_IF,S_MDF_IF
SIEBEL_8K	S_OPTY_DTLIF,S_OPTY_IF
SIEBEL_8K	S_OPTY1_IF,S_ORDER_IF
SIEBEL_8K	S_PART_RPR_IF,S_POSITION_IF
SIEBEL_8K	S_PROD_INT1_IF,S_PROJECT_IF
SIEBEL_8K	S_PRSP_CON_IF,S_QUOTE_TERM_IF
SIEBEL_8K	S_SRC_DTLIF,S_SRC_PAY_IF

Appendix B. An example of data collection for trend analysis

In this appendix we provide with some scripts, DB2 commands, and SQL statements to analyze the growth trend of the tables and tablespaces in an AIX environment. We also present sample outputs for gathering table statistics, tablespace utilization, and tablespace growth patterns of a Siebel environment. You may develop your own scripts based on these samples.

The information about tablespaces can be collected from the `db2 list tablespaces show detail` script, as shown in Figure 52.

```
 Tablespaces for Current Database
Tablespace ID                        = 3
Name                                 = SIEBEL_16K
Type                                 = Database managed space
Contents                             = Any data
State                                = 0x0000
Detailed explanation:
Normal
Total pages                          = 9664
Useable pages                        = 9616
Used pages                           = 6054
Free pages                           = 3562
High water mark (pages)              = 6054
Page size (bytes)                    = 4096
Extent size (pages)                  = 32
Prefetch size (pages)                = 16
Number of containers                 = 1
```

Figure 52. DB2 list tablespaces show detail output

The steps required to load and analyze the Siebel tablespace statistics are as follows:

1. Create the Siebel tablespace statistics table.

   The data definition language (DDL) for creating the table is shown in Figure 53 on page 174.

```
create table tbspace_stats
(tbspace    char(18)  not null,
  time_capture timestamp not null,
  total_pages integer,
  useable_pages integer,
  used_pages   integer,
primary key ( tbspace,time_capture ));
```

Figure 53. DDL for creating the Siebel tablespace statistics table

2. Create the script to extract information from the `db2 list table spaces show detail` output and format appropriate insert statements.

   This example is written in awk as shown in Figure 54 on page 175. Awk is one of the standard programming languages that is widely available in UNIX platforms.

```
begin {FS = ",|[ \t]+"
   first = " insert into tbspace values \( "
   middle = " ,current timestamp"
   last   = " \) \; "}
{  if ( $1 == "Name" )
      {  Name = "\'"$3"\'" }
}
{  if ( $1 == "Type" &&  $3 == "Database"  )
         { DMS = "Yes" }
}
{  if ( $1 == "Type" &&  $3 == "System"  )
         { DMS = "No" }
}
{  if ( $1 == "Total" )
      {  Total = "," $4 }
}
{  if ( $1 == "Useable" )
      {  Useable = "," $4 }
}
{  if ( $1 == "Used" )
      {  Used = "," $4 }
}
{  if ( $1 == "Number" && DMS == "Yes" )
      {  print "insert into tbspace_stats values \(", Name, ",current
timestamp", Total, Useable , Used  " \) \; "  }
}

END
```

Figure 54. Awk script to extract and generate insert statement (tablespace.awk)

3. Create another AIX script to call the awk script and insert the output generated by tablespace.awk into the table, as shown in Figure 55 on page 176.

```
. ~db2inst1/sqllib/db2profile
db2 connect to risc1db
db2 list tablespaces show detail | awk -f tablespace.awk >
tablespace.sql
db2 -vtf tablespace.sql
```

Figure 55. AIX script to populate the Siebel table space statistics table

4. Schedule the script to run at certain intervals using crontab.

   Crontab is a facility in AIX to schedule jobs. Here we schedule it to run tablespace.script every day at 10:30pm:

   ```
   crontab -e
   ```

   ```
   30 22 * * * ~db2inst1/jobs/tablespace.script
   ```

5. Create SQLs to extract the following information.

   We have included three sample SQLs and their outputs:

   a. Current tablespace utilization; see Figure 56 on page 177 and Table 48 on page 177.

   b. Growth pattern of tablespaces for the last four weeks; see Figure 57 on page 178 and Table 49 on page 179.

   c. Weekly percentage growth of tablespaces for the last four weeks; see Figure 58 on page 180 and Table 50 on page 181.

```
with
   lastest ( tbspace, time_capture ) as
      ( select t1.tbspace, max(time_capture)
         from tbspace_stats t1
            group by t1.tbspace
      )

select t2.tbspace as tablespace,
   t2.useable_pages - t2.used_pages  as available_free_pages,
   100*(t2.useable_pages - t2.used_pages )/ t2.useable_pages  as
percent_free_pages,
   t2.used_pages as total_pages_used,
   t2.useable_pages as total_useable_pages
   from tbspace_stats as t2, lastest as t3
   where t2.tbspace = t3.tbspace
   and   t2.time_capture = t3.time_capture
   order by 2 , 1
;
```

Figure 56. SQL to generate the current tablespace utilization

Table 48. Current table space utilization

Tablespace	Available free pages	% free pages	Total used pages	Total usable pages
SIEBEL_16K	3562	37	6054	9616
SIEBEL_4KL	7018	98	118	7136
SIEBEL_IDX	19520	24	60336	79856
SIEBEL_4K	23428	24	71788	95216

```
with
   lastest ( tbspace, time_capture ) as
      ( select t1.tbspace, t1.time_capture
         from tbspace_stats t1
                        where date(time_capture) > (current date - 30
days )
         and
         (   date(time_capture) = ( current date - 1 days )
         or  date(time_capture) = ( current date - 8 days )
         or  date(time_capture) = ( current date - 15 days )
         or  date(time_capture) = ( current date - 22 days )
         )
      )
select t2.tbspace as tablespace, date (t2.time_capture)  ,
   t2.useable_pages - t2.used_pages  as available_free_pages,
   100*(t2.useable_pages - t2.used_pages )/ t2.useable_pages  as
percent_free_pages,
   t2.used_pages as total_pages_used,
   t2.useable_pages as total_useable_pages
   from tbspace_stats as t2, lastest as t3
   where t2.tbspace = t3.tbspace
   and   t2.time_capture = t3.time_capture
   order by 1, 2 asc
;
```

Figure 57. SQL to generate growth pattern of tablespaces for last 4 weeks

Table 49. Growth pattern of tablespaces for last four weeks

Tablespace	Time snapshot taken	Available free pages	% free pages	Total used pages	Total usable pages
SIEBEL_16K	10/17/2000	3672	38	5944	9616
SIEBEL_16K	10/24/2000	3652	37	5964	9616
SIEBEL_16K	10/31/2000	3632	37	5984	9616
SIEBEL_16K	11/07/2000	3612	37	6004	9616
SIEBEL_4K	10/17/2000	25728	27	69488	95216
SIEBEL_4K	10/24/2000	25028	26	70188	95216
SIEBEL_4K	10/31/2000	24328	25	70888	95216
SIEBEL_4K	11/07/2000	24628	25	70588	95216
SIEBEL_4KL	10/17/2000	7058	98	78	7136
SIEBEL_4KL	10/24/2000	7048	98	88	7136
SIEBEL_4KL	10/31/2000	7038	98	98	7136
SIEBEL_4KL	11/07/2000	7028	98	108	7136
SIEBEL_IDX	10/17/2000	19800	24	60056	79856
SIEBEL_IDX	10/24/2000	19730	24	60126	79856
SIEBEL_IDX	10/31/2000	19660	24	60196	79856
SIEBEL_IDX	11/07/2000	19770	24	60086	79856

```
with
   first ( tbspace, time_capture,used_pages ) as
      ( select t1.tbspace, t1.time_capture, dec(t1.used_pages)
         from tbspace_stats t1
         where date(time_capture) = ( current date - 1 days )
         ),
   second ( tbspace, time_capture,used_pages ) as
      ( select t2.tbspace, t2.time_capture, dec(t2.used_pages)
         from tbspace_stats t2
         where date(time_capture) = ( current date - 8 days )
         ),
   third ( tbspace, time_capture,used_pages ) as
      ( select t3.tbspace, t3.time_capture, dec(t3.used_pages)
         from tbspace_stats t3
         where date(time_capture) = ( current date - 15 days )
         ),
   fourth ( tbspace, time_capture,used_pages ) as
      ( select t4.tbspace, t4.time_capture, dec(t4.used_pages)
         from tbspace_stats t4
         where date(time_capture) = ( current date - 22 days )
         ),
   fifth ( tbspace, time_capture,used_pages ) as
      ( select t5.tbspace, t5.time_capture, dec(t5.used_pages)
         from tbspace_stats t5
         where date(time_capture) = ( current date - 29 days )
         )
select t6.tbspace as tablespace_growth,
   dec(((100*(t9.used_pages - t10.used_pages))/t10.used_pages),2) as
four_weeks_ago,
   dec(((100*(t8.used_pages - t9.used_pages))/t9.used_pages),2) as
three_weeks_ago,
   dec(((100*(t7.used_pages - t8.used_pages))/t8.used_pages),2) as
two_weeks_ago,
   dec(((100*(t6.used_pages - t7.used_pages))/t7.used_pages),2) as
last_week
   from  first t6, second t7, third t8, fourth t9, fifth t10
   where t6.tbspace = t7.tbspace
   and   t6.tbspace = t8.tbspace
   and   t6.tbspace = t9.tbspace
   and   t6.tbspace = t10.tbspace
   order by 1 ;
```

Figure 58. SQL to generate weekly growth rate of tablespaces for last 4 weeks

Table 50. Weekly growth rate of tablespaces for last four weeks

Tablespace	Weekly growth rate (%)			
	Four weeks ago	Three weeks ago	Two weeks ago	Last week
SIEBEL_16K	0.	0.	0.	0.
SIEBEL_4K	0.	1.	0.	0.
SIEBEL_4KL	14.	12.	11.	10.
SIEBEL_IDX	0.	0.	0.	0.

Appendix C. DB2 utilities

In this appendix, we describe DB2 utilities that can be used in a Siebel Application environment. We cover DB2 utilities for DB2 and database administration, monitoring, backup/recovery, getting access path information, and so on.

C.1 DB2 administration tools

DB2 administration tools enable you to perform the same functions from a graphical user interface as you could from a command cine processor. These functions include the entering of DB2 commands, SQL statements, or system commands. In this section we cover the most popular tools in the DB2 UDB and Siebel environments.

C.1.1 DB2 Control Center

The Control Center is a graphical tool used to administer DB2 databases. From the Control Center, you can perform the tasks provided by the DBA Utility, Visual Explain, and Performance Monitor tools.

Through this graphical tool, it is possible to create and manage objects such as databases, tablespaces, bufferpools, tables, indexes, and triggers. It also allows you to launch other administration tools provided by DB2 UDB, such as the following:

- Script Center

  The Script Center enables you to create, run, and schedule operating system-level commands and DB2 command scripts.

- Alert Center

  The Alert Center notifies you when thresholds that you have set have been exceeded, or when a node in a multinode environment is no longer responding.

- Journal

  The Journal allows you to view the status of jobs and to view the recovery history log and messages log.

- Information Center

  The Information Center gives you quick access to the information in the DB2 product manuals and sample programs, and provides access to other sources of DB2 information on the Web.

- License Center

  The License Center displays the status of your license, and allows you to configure your system for proper license monitoring.

For database administration beginners, DB2 UDB provides Smart Guides. Smart Guides are GUI windows that help database administrators to understand more clearly the tasks that they are performing through a step-by-step panel that, based on the information received, is able to create and execute commands and also recommend changes in order to increase the database performance.

The Smart Guides are available for the following tasks:

- Back up database
- Create database
- Create table
- Create tablespace
- Index SmartGuide
- Performance configuration
- Restore database
- Configure multisite update SmartGuide

There are other GUI tools that are not available directly from the Control Center toolbar, but which have the same administration importance as the tools available on the Control Center. The following is a list of these tools and a brief description of their tasks:

- Performance Monitor

  This tool is used to monitor DB2 UDB objects such as databases, tables, and tablespaces, thus allowing the database administrator to easily tune the database.

- Event Monitor

  This tool collects monitoring information for a specified database activity during a period of time.

- Event Analyzer

  This tool is used to analyze the output generated by the Event Monitor.
- Visual Explain

  This tool allows the database administrator to investigate how DB2 UDB accesses and retrieves the data.
- Client Configuration Assistant

  This tool configures access to other servers by cataloging them on the current system.

C.1.2 DB2 runstats

The runstats utility updates statistics in the system catalog tables to help with the query optimization process. Without these statistics, the database manager could make a decision that would adversely affect the performance of an SQL statement. The runstats utility allows you to collect statistics on the data contained in the tables, indexes, or both tables and indexes.

Use the runstats utility to collect statistics, based on both the table and the index data, to provide accurate information to the access plan selection process in the following situations:

- When a table has been loaded with data, and the appropriate indexes have been created.
- When a table has been reorganized with the REORG utility.
- When there have been extensive updates, deletions, and insertions that affect a table and its indexes. (“Extensive” in this case may mean that 10 to 20 percent of the table and index data has been affected.)
- Before binding application programs whose performance is critical.
- When comparison with previous statistics is desired. Running statistics on a periodic basis permits the discovery of performance problems at an early stage, as described below.
- When the prefetch quantity is changed.

Following is an example:

```
db2 runstats on table [schema].[name of table] with distribution
and indexes all shrlevel change
```

C.1.3 DB2 reorg

The performance of SQL statements that use indexes can be impaired after many updates, deletes, or inserts have been made. Generally, newly inserted rows cannot be placed in a physical sequence that is the same as the logical sequence defined by the index (unless you use clustered indexes). This means that the Database Manager must perform additional read operations to access the data, because logically sequential data may be on different physical data pages that are not sequential.

The reorg utility optionally rearranges data into a physical sequence according to a specified index. The reorg utility has an option to specify the order of rows in a table with an index, thereby clustering the table data according to the index and improving the CLUSTERRATIO or CLUSTERFACTOR statistics collected by the runstats utility. As a result, SQL statements requiring rows in the indexed order can be processed more efficiently. The reorg utility also stores the tables more compactly by removing unused, empty space (though if you specified PCTFREE when you used ALTER TABLE, that space remains unused).

The reorg utility requires that all other applications that would normally be working against the affected table data and indexes be offline. You may have a work environment where you wish to limit the amount of time your applications cannot work against the data. In this environment, you might consider using the online index reorganization utility.

Following is an example:

```
db2 reorg table [schema].[name of table] index [name of index]
```

You may wish to consider the following factors to determine when to reorganize your table data:

- The volume of insert, update, and delete activity
- Any significant change to the performance of queries which use an index with a high cluster ratio
- Running statistics (runstats) does not improve the performance of queries
- The reorgchk command indicates a need to reorganize your table
- The cost of reorganizing your table, including the CPU time, the elapsed time, and the reduced concurrency resulting from the REORG utility locking the table until the reorganization is complete

C.1.4 DB2 reorgchk

The reorgchk command returns information about the physical characteristics of a table, and whether or not it would be beneficial to reorganize that table. This command can be used through the command line processor. Refer to the *Command Reference* for more information, including how to interpret the command output.

Following is an example:

```
db2 reorgchk
```

C.1.5 Online index reorganization

DB2 UDB V6.1 can perform online index reorganization, to use system resources efficiently.

Without online reorganization, you can only reclaim space through an offline reorganization of the data and index. Use online index reorganization to enable DB2 to dynamically reorganize the pages when database activity creates excess space in the pages. To enable DB2 to perform online reorganization for a specific index, specify the MINPCTUSED option when you create the index with the CREATE INDEX statement.

C.1.6 db2dart

This tool verifies that the architectural integrity of a database is correct. For example, this tool confirms that:

- The control information is correct.
- There are no discrepancies in the format of the data.
- The data pages are the correct size and contain the correct column types.
- Indexes are valid.

If db2dart reports a problem with an index, use the tool's /MI option to mark the index as invalid. The index is rebuilt based on the value of the indexrec database and database manager configuration parameters (see the *DB2 Administration Guide* for details).

You must run this tool on the DB2 server where the database resides. Also ensure that there are no active connections to the database. (Use the LIST APPLICATIONS FOR DATABASE database-alias command and disconnect any applications that are listed.)

Following is an example:

```
db2dart [name of database]
```

For information on db2dart options, type `db2dart` without any options.

C.1.7 db2move

This tool facilitates the movement of large numbers of tables between DB2 databases located on workstations. The tool queries the system catalog tables for a particular database and compiles a list of all user tables. It then exports these tables in PC/IXF format. The PC/IXF files can be imported or loaded to another local DB2 database on the same system, or can be transferred to another workstation platform and imported or loaded to a DB2 database on that platform.

Note: Tables with structured type columns are not moved when this tool is used.

This tool calls the DB2 export, import, and load APIs, depending on the action requested by the user. Therefore, the requesting user ID must have the correct authorization required by those APIs, or the request will fail. In general, user has to be sysadm or dbadm for all actions. Refer to the *Command Reference,* Chapter 3, “CLP Commands (DB2MOVE)” for extensive information about getting the right authority.

Following is an example:

```
db2move [name of database] export -tc [table owner]
```

C.1.8 db2look

This tool extracts the required DDL statements to reproduce the database objects of a production database on a test database. This tool can also generate the required UPDATE statements used to replicate the statistics on the objects in a test database, as well as the update database configuration and update database manager configuration parameters and the db2set statements so that the registry variables and configuration parameter settings on the test database match those of the production database.

It is often advantageous to have a test system contain a subset of the production system’s data. However, access plans selected for such a test system are not necessarily the same as those that would be selected for the production system. Both the catalog statistics and the configuration parameters for the test system must be updated to match those of the production system. Using this tool makes it possible to create a test database where access plans are similar to those that would be used on the production system.

The SELECT privilege on the system catalogs is required for using db2look. This command establishes a database connection.

Note: Only configuration parameters and registry variables that affect the DB2 query optimizer are extracted.

Following is an example:

```
db2look -d [name of database] -u [table owner] -l -m
```

C.2 DB2 monitoring tools

The DB2 Database Manager maintains data about its operation, its performance, and the applications using it. This data can provide important performance and troubleshooting information. In this section, we discuss a file and several tools, so that you are able you to monitor the DB2 activities by Siebel applications.

C.2.1 DB2 Performance Monitor

The Performance Monitor provides information about the state of the DB2 Universal Database and the data that it controls. It is a graphical utility that can be customized for your database environment. You can define thresholds or zones that trigger warnings or alarms when the values being collected by the Performance Monitor are not within acceptable ranges.

You can monitor DB2 objects such as instances, databases, tables, table spaces, and connections by selecting the object in the Object Tree pane or in the Contents pane and clicking mouse button 2. From there, you can choose to start monitoring activity.

C.2.2 db2diag.log

The db2diag.log file is an ASCII file that contains information logged by DB2. It is located in the directory specified by the DIAGPATH database manager configuration parameter. Use a text editor to view the file on the machine where you suspect a problem to have occurred.

The information in the db2diag.log file includes the following:

- The location reporting an error. Application identifiers allow you to match up entries pertaining to an application on the db2diag.log files of servers and clients.
- A diagnostic message (beginning with DIA) explaining the reason for the error.
- Any available supporting data, such as SQLCA data structures and pointers to the location of any extra dump or trap files.

There are two types of entries in the db2diag.log file:

- Administrative events

  These entries are valuable because they indicate whether events such as backups and restores started and finished.

- Error information

  This information is only useful if you are trying to diagnose an external symptom, or if you have already determined what the error is and are looking for more information. For example, when an application receives an unexpected SQL code, or when a database crashes, the file can contain error information, including pointers to dump files.

If the database is behaving normally, this type of information is not important and can be ignored.

Notes:

- Check the end of the file for the most recent data, because new information is always appended to the bottom of the file.
- Entries always have a timestamp. If you know when an error occurred, look for the first entry in the file marked with this time.
- The db2diag.log file grows continuously. When it gets too large, back it up and then erase the file. A new db2diag.log file is generated automatically the next time it is required by the system.
- Set database manager configuration parameter DIAGLEVEL to 4, which will give you additional information about errors, warnings and informational messages. The default value is 3.

C.2.3 DB2 Event Monitor

An event monitor tracks specific data as the result of an event. For example, starting the database might be an event that causes an event monitor to track the number of users on the system by taking an hourly snapshot of authorization IDs using the database.

In contrast to taking a point-in-time snapshot, an event monitor writes out database system monitor data to either a file or a named pipe, when one of the following events occurs:

- End of a transaction
- End of a statement
- A deadlock
- Start of a connection
- End of a connection
- Database activation
- Database deactivation
- End of a statement's subsection (when a database is partitioned)
- Flush event monitor statement is issued

An event monitor effectively provides the ability to obtain a trace of the activity on a database.

C.2.4 DB2 snapshot monitor

Taking a snapshot gives you information for a specific point in time. A *snapshot* is a picture of the current state of activity in the database manager for a particular object or group of objects.

C.3 Backup/Restore commands

There are two options in a backup/restore scenario:

1. Offline backup
2. Online backup

Version recovery using the BACKUP command in conjunction with the RESTORE command puts the database in a state that has been previously saved. You use this recovery method with non-recoverable databases (that is, databases for which you do not have archived logs). You can also use this method with recoverable databases by using the WITHOUT ROLLING FORWARD option.

Roll-forward recovery using the BACKUP command in conjunction with the RESTORE and ROLLFORWARD commands allows the database or tablespace to be recovered to its state at a specified point in time.

When a new database is created, the default backup option is: offline. If you would like to enable online backup option, you have to set the *logretain* parameter to *recovery*, to indicate that the logs are retained and can be used for forward recovery.

C.4 Visual Explain

DB2 provides the most comprehensive explain facility in the industry, with detailed optimizer information on the access plan chosen for an explained SQL statement. Several methods are provided to give you the flexibility you need to capture and access explain information.

Detailed optimizer information that allows for in-depth analysis of an access plan is kept in explain tables separate from the actual access plan itself. There are three ways to get information from the explain tables; however, in this appendix we only look into Visual Explain in detail:

- Write your own queries (based on the explain table descriptions)
- Use the db2exfmt tool
- Use Visual Explain (to view explain snapshot information)

Visual Explain

Visual Explain can be used to study queries in more detail when compared to the other methods, especially those that contain more complex sequences of operations.

Visual Explain lets you view the access plan for explained SQL statements as a graph. You can use the information available from the graph to tune your SQL queries for better performance. Visual Explain also lets you dynamically explain a SQL statement and view the resulting access plan graph.

The optimizer chooses an access plan and Visual Explain displays the information as an access plan graph in which tables and indexes, and each operation on them, are represented as nodes, and the flow of data is represented by the links between the nodes.

To display an access plan graph, you must have created an explain snapshot. From an access plan graph, you can view the details for:

- Tables and indexes (and their associated columns)

- Operators (such as table scans, sorts, and joins)
- Tablespaces and functions.

You can also use Visual Explain to:

- View the statistics that were used at the time of optimization. You can then compare these statistics to the current catalog statistics to help you determine whether rebinding the package might improve performance.
- Determine whether or not an index was used to access a table. If an index was not used, Visual Explain can help you determine which columns might benefit from being indexed.
- View the effects of performing various tuning techniques by comparing the before and after versions of the access plan graph for a query.
- Obtain information about each operation in the access plan, including the total estimated cost and number of rows retrieved (cardinality).

For additional detail on Visual Explain, refer to the online information available through the Control Center.

C.5 Index Advisor utility

DB2 UDB v6.1 introduces the Index Advisor utility, which helps you improve the performance of your database by suggesting and creating an optimal set of indexes. Use the Index Advisor to help you:

- Find the best indexes for a problem query
- Determine the best indexes for a set of queries (a workload), subject to resource limits which you can optionally apply
- Test an index on a workload without having to create the index

You can invoke the Index Advisor using any of the following:

- A SmartGuide, called the Index Smart Guide, available from the Control Center (**Indexes -> Create -> Index using SmartGuide**)
- A command-line program: db2advis
- DB2 database maintenance

C.6 DB2 Governor

The DB2 Governor is an application that uses snapshot monitoring to supervise the load and usage of the database manager. It provides the functions to force or change the run-time priority of applications exceeding certain limits. These limits are specified by the DBA in the db2gov configuration file. Application limits and privileges can be expressed using several different parameters, for example, maximum amount of CPU.

Following is an example:

```
db2gov start [database name] db2gov1.cfg db2gov1.log
```

See the *DB2 Command Reference* and the *DB2 Administration Guide* for more information.

C.7 db2batch

This tool will read SQL statements from either a flat file or standard input, dynamically describe and prepare the statements, and return an answer set. It also provides the added flexibility of allowing you to control the size of the answer set, as well as the number of rows that should be sent from this answer set to an output device.

You can also specify the level of performance-related information supplied, including the elapsed time, CPU and buffer pool usage, locking, and other statistics collected from the database monitor. If you are timing a set of SQL statements, db2batch will also summarize the performance results and provide both arithmetic and geometric means. For more information on invocation syntax, and options, type: `db2batch -h` on a command line.

A benchmark tool (db2batch) is provided in the misc subdirectory of your instance sqllib directory.

```
The following is an example of how db2batch could be used with an input file
db2batch.sql:

db2batch -d sample-f db2batch.sql
```

where `sample` is the name of the database and `db2batch.sql` is input file.

One of the output screens from db2batch tool looks like this:

```
Summary of Results
==================
              Elapsed   Agent CPU        Rows      Rows
Statement #   Time (s)  Time (s)         Fetched   Printed
1             0.074     0.020            5         5
2             0.037     Not Collected    8         5
Arith. mean   0.055
Geom.  mean   0.052
```

See the *DB2 Command reference* and *DB2 Administration Guide* for more information about using db2batch.

Appendix D. How to test your SQL

In this appendix, we provide a sample program that you can use to get access path information for your SQL. For detailed information about how to use this program, refer to 9.2.3, “Analyzing the SQL” on page 155.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "util.h"

#ifdef DB268K
#include <LibraryManager.h>
#endif

EXEC SQL INCLUDE SQLCA;

#define CHECKERR(CE_STR) if (check_error (CE_STR,&sqlca) != 0) return 1;

int main(int argc,char *argv[]) {

   EXEC SQL BEGIN DECLARE SECTION;
      char  st[4096];
      char  parm_var1[30];
      char  parm_var2[30];
      char  parm_var3[30];
      char  oparm_var1[30];
      char  oparm_var2[30];
      char  oparm_var3[30];
   EXEC SQL END DECLARE SECTION;
```

Figure 59. Sample program (1 of 3)

```
 printf( "Sample C program: Siebel\n" );


   EXEC SQL CONNECT TO siebeldb user siebel using siebel ;


   EXEC SQL SET CURRENT QUERY OPTIMIZATION 5;

   EXEC SQL set current explain mode  yes;


   strcpy(st,"SELECT T3.POSTN_ID, T1.PR_POSTN_ID, T1.LAST_UPD,
T1.PR_REP_SYS_FLG, T4.ROW_ID, T1.MID_NAME, T6.FST_NAME, T3.ROW_ID,
T8.ADDR, T1.DISA_CLEANSE_FLG, T1.EMAIL_ADDR, T1.CREATED_BY, T1.EMP_ID,
T1.PR_REP_DNRM_FLG, T2.PR_BL_ADDR_ID, T1.LAST_NAME, T1.CONFLICT_ID,
T3.CON_FST_NAME, T1.PR_OU_ADDR_ID, T1.PR_REP_MANL_FLG,
T1.PR_PER_ADDR_ID, T1.CREATED, T4.PR_EMP_ID, T2.PR_SRV_AGREE_ID,
T2.PR_SHIP_ADDR_ID, T1.ROW_ID, T5.LOGIN, T1.DEDUP_TOKEN, T1.FST_NAME,
T1.TMZONE_CD, T2.NAME, T1.BU_ID, T3.CON_MID_NAME, T2.LOC, T9.LOGIN,
T3.ROW_STATUS, T2.PR_BL_PER_ID, T1.COMMENTS, T1.HOME_PH_NUM, T8.COUNTRY,
T1.ASST_PH_NUM, T1.MODIFICATION_NUM, T6.LAST_NAME, T1.PR_OPTY_ID,
T2.ROW_ID, T7.NAME, T8.CITY, T2.INTEGRATION_ID, T1.SUPPRESS_CALL_FLG,
T1.SUPPRESS_MAIL_FLG, T3.CON_LAST_NAME, T4.NAME, T2.PR_SHIP_PER_ID,
T1.OWNER_PER_ID, T1.OU_MAIL_STOP, T1.WORK_PH_NUM, T1.LAST_UPD_BY,
T1.PREF_COMM_MEDIA_CD, T8.ZIPCODE, T1.CELL_PH_NUM, T8.STATE,
T1.CON_ASST_NAME, T1.PR_DEPT_OU_ID, T1.CALL_FLG, T1.EMAIL_SR_UPD_FLG,
T1.SEX_MF, T8.ROW_ID, T1.FAX_PH_NUM, T1.CON_MANAGER_PER_ID,
T1.PER_TITLE, T1.JOB_TITLE, T1.ASGN_USR_EXCLD_FLG

FROM siebel.S_CONTACT T1 INNER JOIN siebel.S_ORG_EXT T2 ON
T1.PR_DEPT_OU_ID = T2.ROW_ID INNER JOIN siebel.S_POSTN_CON T3 ON
T3.POSTN_ID = ? AND T1.ROW_ID = T3.CON_ID INNER JOIN siebel.S_POSTN T4
ON T4.ROW_ID = T3.POSTN_ID LEFT OUTER JOIN siebel.S_EMPLOYEE T5 ON
T1.EMP_ID = T5.ROW_ID LEFT OUTER JOIN siebel.S_CONTACT T6 ON
T1.CON_MANAGER_PER_ID = T6.ROW_ID LEFT OUTER JOIN siebel.S_ORG_INT T7 ON
T1.BU_ID = T7.ROW_ID LEFT OUTER JOIN siebel.S_ADDR_ORG T8 ON
T1.PR_OU_ADDR_ID = T8.ROW_ID LEFT OUTER JOIN siebel.S_EMPLOYEE T9 ON
T4.PR_EMP_ID = T9.ROW_ID

WHERE (T1.PRIV_FLG = ?) AND (T2.NAME LIKE ?)

ORDER BY T3.POSTN_ID, T3.CON_LAST_NAME, T3.CON_FST_NAME FOR FETCH ONLY
OPTIMIZE FOR 1 ROW ");
```

Figure 60. Sample program (2 of 3)

```
   EXEC SQL PREPARE s1 FROM :st;
   CHECKERR ("PREPARE");

   EXEC SQL DECLARE c1 CURSOR FOR s1;
   .......
   .......

   EXEC SQL OPEN c1 USING :parm_var1,:parm_var2,:parm_var3;
   CHECKERR ("OPEN");

   do {

      EXEC SQL FETCH c1 INTO :oparm_var1,:oparm_var2,:oparm_var3;

      if (SQLCODE != 0) break;

   } while ( 1 );

   EXEC SQL CLOSE c1;
   CHECKERR ("CLOSE");

   EXEC SQL COMMIT;
   CHECKERR ("COMMIT");

   EXEC SQL CONNECT RESET;
   CHECKERR ("CONNECT RESET");
   return 0;
}
```

Figure 61. Sample program (3 of 3)

Appendix E. Data migration considerations: from DB2 to DB2

In this Appendix we provide an overview of migration from DB2 on one platform to DB2 on a different platform. In the case of migration from other RDBMS to DB2, we suggest you use EIM. When you move data between DB2s on different platforms, use db2move and db2look tools. This is true only for initial migration. For all further updates of Siebel data, use EIM. See Chapter 7, “Siebel Enterprise Integration Manager (EIM)” on page 109 for details about EIM.

E.1 Moving DB2 databases between platforms

We describe how to copy database objects (for example, tables, tablespaces) between DB2 on the same or a different operating system. For this purpose you can use db2move (database movement) and db2look (DB2 statistics and DDL extraction) tools. See C.1.7, “db2move” on page 188 and C.1.8, “db2look” on page 188 for more information.

Procedures described in the following sections are applicable to different versions of DB2 on operating systems where DB2 UDB for UNIX, Windows, and OS/2 are available.

E.2 Moving database objects: examples

In this section we describe some examples of moving database objects using DB2 tools.

DB2MOVE EXPORT, DB2MOVE IMPORT

If you only need to copy tables between two DB2s, or if you want to import tables into a new database (the other option is to load), follow the steps described below. This approach is usable if your tables are not already created. Import is generally slower than load, so consider using db2move load, described in the next section.

1. Logon to the platform from which you want to export tables as a user, with the required authorization for export.
2. Use the db2move tool to export all tables and their contents:

   ```
   db2move <database-alias> export
   ```

   To export a subset of tables, use the options described in *DB2 UDB Command Reference*, SC09-2844.

   You will get the following files generated by EXPORT:

  - tabnnn.ixf

    The exported PC/IXF file of a specific table that is needed for creation of tables.

  - db2move.lst

    The list of all files can be used as input to the db2move IMPORT or LOAD action.

  - tabnnn.msg files

    The export message file of the corresponding table.

  - export.out

    The summarized result of the export action.

  - tabnnnc.yyy

    The exported LOB files of a specific table when your tables contain LOBs.

  - system.msg

    The message file containing system messages when your table contains LOBs .

3. Transfer db2move.lst and all tabnnn.ixf files (also tabnnnc.yyy, if your tables contain LOBs) to your target server by using, for example, FTP. You do not need to copy tabnnn.msg files, because they include only messages produced by export.
4. Logon to the target operating system as a user, with the required authorization for import.
5. Create a new database:

   ```
   create db <database-alias>
   ```

6. Use db2move to import tables into an existing database:

   ```
   db2move <database-alias> import
   ```

DB2MOVE EXPORT, DB2LOOK, DB2MOVE LOAD

If you need to load the contents of the tables into a database, or if you want to copy objects other than just tables, you have to use db2look and db2move. In order to use the db2move load command, the target tables must already exist.

Use this approach if you have a larger amount of data, since load is faster than import. For creating tables and other database objects, use the output created from db2look.

1. Logon as a user, with the required authorization for export.
2. As in the previous example, use db2move to export all tables and their contents:

   ```
   db2move <database-alias> export
   ```

3. Use db2look to get DDL statements for creation of all required objects:

   ```
   db2look -d <database-alias> -a -e -l -x -o <outfile-name>
   ```

   You should have the information about all objects created by all users in the outfile:

   - DDL statement for database objects such as tables, views, automatic summary tables, indexes, triggers, user defined functions, etc.
   - Authorization DDL (GRANT statement, for example)
   - DDL statements for user-defined tablespaces, user-defined bufferpools, and user-defined nodegroups

4. Transfer <file name>.sql, db2move.lst, and all tabnnn.ixf files (as well as tabnnnc.yyy, if your tables contain LOBs) to your target operating system by using, for example, FTP. Be careful to upload the <file name>.sql file in text mode, not binary mode. You do not need to copy tabnnn.msg files, because they include only messages produced by export.
5. Logon onto the target operating system as a user, with the required authorization for load.
6. Create a new database:

   ```
   create db <database-alias>
   ```

7. Edit <outfile-name>.sql and correct the CONNECT TO <database name> statement to match your new database name.
8. Run the <outfile-name>.sql file containing DDL to create all database objects. The file is created by db2look (see step 3):

   ```
   db2 -tvf <outfile-name>.sql
   ```

9. Use db2move to load tables into an existing database:

   ```
   db2move <database-alias> load
   ```

Note: User-defined functions are not copied by using db2move and db2look. You have to reinstall UDFs manually.

Appendix F. Special notices

This publication is intended to help systems programmers and DB2 DBAs to install Siebel *e*Business Applications 2000 on NT or AIX platforms. The information in this publication is not intended as the specification of any programming interfaces that are provided by AIX V4.3.3 and DB2 UDB V6. See the PUBLICATIONS section of the IBM Programming Announcement for AIX and DB2 for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

IBM ®
AIX
DB2 RETAIN
S/390
ServicePac
SP
System/390
Wizard
XT
NetView
Redbooks
Redbooks Logo
RS/6000
400
Lotus
Lotus SmartSuite
SmartSuite
Notes
TME

The following terms are trademarks of other companies:

Tivoli, Manage. Anything. Anywhere.,The Power To Manage., Anything. Anywhere.,TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli Systems Inc., an IBM company, in the United States, other countries, or both. In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S.

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

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

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

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

Appendix G. Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

G.1 IBM Redbooks

For information on ordering these publications see “How to get IBM Redbooks” on page 211.

- *Database Performance on AIX in DB2 UDB and Oracle Environments,* SG24-5511
- *Siebel 2000 Database Implementation on OS/390 Using NT Siebel Servers,* SG24-5953
- *Understanding IBM RS/6000 Performance and Sizing*, SG24-4810
- *RS/6000 Performance Tools in Focus*, SG24-4989

G.2 IBM Redbooks collections

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at **ibm.com**/redbooks for information about all the CD-ROMs offered, updates and formats.

CD-ROM Title	**Collection Kit Number**
IBM System/390 Redbooks Collection	SK2T-2177
IBM Networking Redbooks Collection	SK2T-6022
IBM Transaction Processing and Data Management Redbooks Collection	SK2T-8038
IBM Lotus Redbooks Collection	SK2T-8039
Tivoli Redbooks Collection	SK2T-8044
IBM AS/400 Redbooks Collection	SK2T-2849
IBM Netfinity Hardware and Software Redbooks Collection	SK2T-8046
IBM RS/6000 Redbooks Collection	SK2T-8043
IBM Application Development Redbooks Collection	SK2T-8037
IBM Enterprise Storage and Systems Management Solutions	SK3T-3694

G.3 Siebel documentation

These Siebel publications (shipped with the product) are useful information sources:

- *Siebel V6.2.1 Release Notes*
- *Siebel Server Installation Guide*

- *Siebel Client Installation and Administration Guide*
- *Siebel Server Administration Guide*
- *Siebel Enterprise Integration Manager Administration Guide*
- *Siebel System Requirements and Supported Platforms*
- *Siebel .COM Application Guide*

G.4 Other resources

- *DB2 UDB V6 Release notes*
- *DB2 Quick Beginning for Windows NT V6*, GC09-2835
- *DB2 Quick Beginning for UNIX V6*, GC09-2836
- *DB2 Administration Guide: Design and Implementation V6,* SC09-2839
- *HACMP for AIX, Version 4.2: Concepts and Facilities,* SC23-1938
- *HACMP for AIX, Version 4.2: Installation Guide,* SC23-1940
- *HACMP for AIX, Version 4.2: Planning Guide,* SC23-1939

G.5 Referenced Web sites

These Web sites are also relevant as further information sources:

- `http://www.siebel.com/` Siebel home page
- `http://www.siebel.com/global/index.html` Siebel Global Services
- `http://supportweb.siebel.com/` Siebel SupportWeb page
- `http://www.ibm.com/solutions/erp/` IBM ERP Web page
- `http://www.software.ibm.com/data/db2/` DB2 Product Family Web page
- `ftp://ftp.software.ibm.com/ps/products/db2/fixes/english-us/siebel/siebe l2000/` DB2 FixPack Web page (NT, AIX)
- `http://www.rs6000.ibm.com/doc_link/en_US/a_doc_lib/aixgen/` AIX standard documentation Web site.

How to get IBM Redbooks

This section explains how both customers and IBM employees can find out about IBM Redbooks, redpieces, and CD-ROMs. A form for ordering books and CD-ROMs by fax or e-mail is also provided.

- **Redbooks Web Site** **ibm.com**/redbooks

  Search for, view, download, or order hardcopy/CD-ROM Redbooks from the Redbooks Web site. Also read redpieces and download additional materials (code samples or diskette/CD-ROM images) from this Redbooks site.

  Redpieces are Redbooks in progress; not all Redbooks become redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

- **E-mail Orders**

  Send orders by e-mail including information from the IBM Redbooks fax order form to:

	e-mail address
In United States or Canada	pubscan@us.ibm.com
Outside North America	Contact information is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Telephone Orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	Country coordinator phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Fax Orders**

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	Fax phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

This information was current at the time of publication, but is continually subject to change. The latest information may be found at the Redbooks Web site.

IBM Intranet for Employees

IBM employees may register for information on workshops, residencies, and Redbooks by accessing the IBM Intranet Web site at http://w3.itso.ibm.com/ and clicking the ITSO Mailing List button. Look in the Materials repository for workshops, presentations, papers, and Web pages developed and written by the ITSO technical professionals; click the Additional Materials button. Employees may access MyNews at http://w3.ibm.com/ for redbook, residency, and workshop announcements.

IBM Redbooks fax order form

Please send me the following:

Title	**Order Number**	**Quantity**

First name Last name

Company

Address

City Postal code Country

Telephone number Telefax number VAT number

□ Invoice to customer number

□ Credit card number

Credit card expiration date Card issued to Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

Glossary

activity A specific task or event to be completed.

applet In Siebel Applications, a visual application unit that appears on screen as part of a view.

application In Siebel Applications, a subset of the applications that can be invoked from an icon on the desktop or Start menu.

base table A table containing columns that are defined and used by Siebel Applications.

data manager See data object layer.

Data Objects layer The architectural layer in the repository that consists of data object definitions. These include object definitions that directly represent tables, columns, and indexes in the underlying DBMS.

Enterprise Integration Manager (EIM) Manages the exchange of data between Siebel database tables and other corporate databases. It is used to perform bulk imports, exports, merges, and deletes.

extension column A column that is not used by standard Siebel applications, but is used by custom-configured applications.

extension table A table in Siebel Applications that provides additional columns to a data table that cannot be directly added to the original table.

Interface tables A table that denormalizes a group of base tables into a single table that external programs can interface to. Used by EIM to import initial data for populating one or more base tables.

mobile client Typically, refers to a portable computer used by a field representative who does not have continuous access to a network.

opportunity A potential revenue-generating event that is associated to an account, potential revenue, probability of completion, sales methodology, etc.

repository The complete set of object definition defining a Siebel application or suite of applications. It is part of the Siebel client, compiled into the Siebel repository file (the .srf file).

responsibility A class of users requiring access to the same set of application features. A user's responsibility identifies the views that he or she can navigate to in Siebel Applications.

screen In Siebel Applications, a screen is a logical collection of views. A list of the views available for the screen appears when you click a tab in the tab bar or choose a menu option in the screen menu.

seed data Required data created in the Siebel database during a Siebel Applications installation process that is needed for the application to start up.

view A display panel consisting of a particular arrangement of applets. The Siebel application window displays one view at any one time. You can change the currently active view by selecting a different view from the view bar or from a submenu of the screen menu.

Index

IBM Redbooks review

Your feedback is valued by the Redbook authors. In particular we are interested in situations where a Redbook "made the difference" in a task or problem you encountered. Using one of the following methods, **please review the Redbook, addressing value, subject matter, structure, depth and quality as appropriate.**

- Use the online **Contact us** review redbook form found at **ibm.com**/redbooks
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@us.ibm.com

Document Number **Redbook Title**	SG24-6211-00 Implementing Siebel eBusiness Applications with DB2 UDB on AIX/NT
Review	
What other subjects would you like to see IBM Redbooks address?	
Please rate your overall satisfaction:	O Very Good O Good O Average O Poor
Please identify yourself as belonging to one of the following groups:	O Customer O Business Partner O Solution Developer O IBM, Lotus or Tivoli Employee O None of the above
Your email address: The data you provide here may be used to provide you with information from IBM or our business partners about our products, services or activities.	 O Please do not use the information collected here for future marketing or promotional contacts or other communications beyond the scope of this transaction.
Questions about IBM's privacy policy?	The following link explains how we protect your personal information. **ibm.com**/privacy/yourprivacy/

Implementing Siebel *e*Business Applications with DB2 UDB on AIX/NT

(0.2”spine)
0.17”<->0.473”
90<->249 pages

Implementing Siebel *e*Business Applications with DB2 UDB on AIX/NT

Install and customize Siebel *e*Business Applications 2000, V6.2.1

Understand Siebel database characteristics

Performance tuning and planning tips

This IBM Redbook will help you install and customize Siebel *e*Business Applications 2000 using DB2 UDB V6 on AIX and NT platforms. It is based on the installation experiences gained while installing Siebel *e*Business Applications 2000 at the IBM ITSO Center in Poughkeepsie, and on customer support experiences from the IBM Toronto Laboratory, IBM DB2 UDB Data Management Consulting Services, Siebel Engineering, and Siebel Technical Services.

The redbook provides the following:

-An overview of Siebel Systems, including architecture, major components, and various types of clients

-Sizing information based on existing customer system environments

-A set of DB2 parameter values for high performance

-An extensive description of Siebel database characteristics on DB2 UDB

-Information on Siebel's Enterprise Integration Manager (EIM), the Siebel importing/exporting tool

-Data migration considerations from DB2 on one platform to DB2 on another platform

-An examination of AIX and DB2 tools to help you administer Siebel system environments

-Backup/recovery scenarios

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-6211-00 ISBN 0738421103