



Troubleshooting Guide for Siebel e-Billing Manager

Version 4.5.3.1
Data Published: 5.26.2005

Siebel Systems, Inc., 2207 Bridgepointe Parkway, San Mateo, CA 94404

Copyright © 2005 Siebel Systems, Inc.

All rights reserved.

Printed in the United States of America

No part of this publication may be stored in a retrieval system, transmitted, or reproduced in any way, including but not limited to photocopy, photographic, magnetic, or other record, without the prior agreement and written permission of Siebel Systems, Inc.

Siebel, the Siebel logo, UAN, Universal Application Network, Siebel CRM OnDemand, TrickleSync, Universal Agent, and other Siebel names referenced herein are trademarks of Siebel Systems, Inc., and may be registered in certain jurisdictions.

Other product names, designations, logos, and symbols may be trademarks or registered trademarks of their respective owners.

PRODUCT MODULES AND OPTIONS. This guide contains descriptions of modules that are optional and for which you may not have purchased a license. Siebel's Sample Database also includes data related to these optional modules. As a result, your software implementation may differ from descriptions in this guide. To find out more about the modules your organization has purchased, see your corporate purchasing agent or your Siebel sales representative.

U.S. GOVERNMENT RESTRICTED RIGHTS. Programs, Ancillary Programs and Documentation, delivered subject to the Department of Defense Federal Acquisition Regulation Supplement, are "commercial computer software" as set forth in DFARS 227.7202, Commercial Computer Software and Commercial Computer Software Documentation, and as such, any use, duplication and disclosure of the Programs, Ancillary Programs and Documentation shall be subject to the restrictions contained in the applicable Siebel license agreement. All other use, duplication and disclosure of the Programs, Ancillary Programs and Documentation by the U.S. Government shall be subject to the applicable Siebel license agreement and the restrictions contained in subsection (c) of FAR 52.227-19, Commercial Computer Software - Restricted Rights (June 1987), or FAR 52.227-14, Rights in Data—General, including Alternate III (June 1987), as applicable. Contractor/licensor is Siebel Systems, Inc., 2207 Bridgepointe Parkway, San Mateo, CA 94404.

Proprietary Information

Siebel Systems, Inc. considers information included in this documentation and in Siebel eBusiness Applications Online Help to be Confidential Information. Your access to and use of this Confidential Information are subject to the terms and conditions of: (1) the applicable Siebel Systems software license agreement, which has been executed and with which you agree to comply; and (2) the proprietary and restricted rights notices included in this documentation.

Contents

1 Preface

- Siebel Customer Self-Service 5
- Siebel e-Billing Manager 6
- Key Benefits of e-Billing Manager 16
- The EPP Suite 22
- Conclusion 26
- About This Guide 27
- Related Documentation 28
- Obtaining Siebel Software and Documentation 28
- If You Need Help 29

2 General Troubleshooting Process

- How to Begin Troubleshooting 33
- Verifying Product Version 34
- Log Files 34
- Interpreting Exceptions 38
- Using Debug 40
- Reviewing Recent System Changes 43

3 Job Failure

- Multiple Job Failures 45
- When Jobs Hang 45
- When the System Hangs 46
- Indexer Job Failure 46
- EmailNotification Job Failure 50

4 DDF Validation Errors

Fixing Invalid DDF Content 57

5 Improving Application Performance

Summary List Response Time 61

Indexer Job Performance 61

EmailNotification Job Performance 62

Siebel Customer Self-Service

Siebel is the world's leading provider of customer self-service and e-billing software and services. Its solutions help service providers increase customer loyalty while reducing the overall cost to serve their customers. Service providers use the software to move customer service interactions from expensive paper and call center-based channels to lower-cost and more responsive self-service and assisted care channels like the web, e-mail and IVR.

Siebel software has been deployed in some of the most challenging business and technical environments in the world and has proven to be the industry's most scalable and reliable solution. The applications have been continuously developed and refined to deliver the most comprehensive functionality available. As a result, Siebel customers consistently gain the highest adoption and ROI in the communications industry – typically 5-10 times higher than companies using in-house or competing packaged solutions.

Self-Service for Consumers and Businesses

Many billing and customer care processes today are expensive, inefficient and unsatisfactory for both communication service providers (CSPs) and their customers. With Siebel customer self-service and e-billing solutions, carriers empower their business and consumer customers to serve themselves and address all of their account and service-related activities online, instead of going through a call center, account rep, or retail outlet. For end customers, this makes doing business with a service provider more convenient, more efficient, and more satisfying. For service providers, it means improved competitive differentiation, significantly reduced customer care costs, increased customer loyalty and streamlined billing/payment processing.

Contact centers provide the key to unlocking the potential business benefits of customer self-service. Contact center statistics show that 60-80 percent of customer service issues relate to a customer's account. For this reason, Siebel customer self-service solutions provide direct access to detailed account information, and interface with complementary front-and back-office systems to provide access to a variety of account-related services.

Siebel e-Billing Manager

Siebel e-Billing Manager enables organizations to deliver interactive access to electronic bills for both consumers and businesses with online payment. With Siebel e-billing platform, organizations can cost-efficiently present various bills online and provide users with multiple online payment options.

Designed to integrate with existing online applications and diverse core legacy systems, Siebel e-Billing Manager extends the convenience of online self-service to an organization's entire customer base, as well as to its call centers. Online account e-bills with payment enables key features such as drill-down to more detail, "paper turn-off" and access to advanced self-service capabilities, offering an online experience that provides superior service and value to customers, while decreasing the cost of servicing those customers.

- Multi-Bill Quick Navigation
- Account Aggregation
- Online Payment Wallet and Payment History
- Archiving via a "Virtual file cabinet with years worth of statements"
- Customer and Company Profile Management
- Notifications and Alerts
- Paper Turn-Off Support
- Targeted Marketing Promotions
- Print and Download Functionality

Siebel platform powers self-service solutions for leading organizations worldwide. The unmatched scalability and performance (1) of Siebel platform enables organizations to serve millions of end-customers with real-time access to years of statement data.

e-Billing Benefits

Improve Service Levels

- Allow customers to opt in for secure online e-bill delivery, providing valuable and convenient 24x7 access to e-bills for customers and service representatives.

Accelerate Cash Receipt

- Online payment accelerates cash receipt and reduces float from Days Sales Outstanding (DSO)

Generate Revenue

- Promote targeted offers within e-bills to increase product or service cross-sell.

Reduce Service, Print and Mail Costs

- Handle customer requests for billing reprints and account information online, replacing costly phone calls.
- e-Billing reduces paper bill production and mailing costs by saving \$924,000 annually per 100,000 customers enrolled.
- Online customers make 30% fewer calls to call centers.

Siebel solution has demonstrated an over 20% paperless enrollment rate among its deployments.

Key Features of e-Billing Manager

Siebel e-Billing Manager is designed to serve the diverse needs of the entire organization's business segments and constituents. Key features of Siebel e-Billing Manager include:

Interactive Statement Presentment - e-Bills can be presented in both a dynamic fashion, using HTML, and in a print-friendly format using PDF. When presented in HTML, e-bills provide an interactive Web experience with search, sorting, filtering, drill-down and download capabilities. Furthermore, dynamic HTML presentment allows embedded links so customers can click-through and drill-down to more detailed information and self-service functions.

Combined Accounts Statement - Siebel e-Billing Manager creates a single consolidated view of all of a customer's accounts by aggregating account information from multiple disparate legacy systems. Using the application's self-service features, accounts can be linked by customers themselves, or automatically, by the biller in the background.

Online Payments - e-Billing provides personal wallets for managing multiple payment vehicles (for example credit cards or checking accounts). Users or permitted customer service representatives (CSRs) may make one-time payments, establish scheduled recurring payments, view payment history, set thresholds and payment alerts.

Paperless Account Option (Print-Friendly Statements) - Customers have the ability to turn off paper via a managed consent model with both print-friendly PDF or HTML presentment alternatives online. Siebel solution complies with regulatory requirements for paperless delivery and has demonstrated an over 20% paperless enrollment rate among its deployments.

Multi-Year Statement History and Online Archive - A disk-efficient, high performance statement archiving system gives organizations the flexibility to store years of historical account statements and account-related communications without degrading performance or increasing storage costs.

Notifications and Alerts - E-mails and messaging alerts notify customers about account changes, payment events and updates. “Statement ready” notifications draw customers to the security of the online site and encourage ongoing use. In addition, customers can configure and manage personalized alerts such as when a checking balance is below certain level, a change was made to the account profile or that a Secure Message response to an inquiry is available for review. Customers can also designate their preferred notification channels including e-mail, PDA and mobile phone.

Contextual Self-Service and Customer Care - While customer self-service is often available via an organization’s website, integrating it into e-bills and making it contextual makes it easier for customers to find relevant information and act immediately. It also creates an opportunity for the organization to promote self-service operations that may otherwise be unknown or difficult to find on a site. Examples of self-service include changing an address, turning off paper, or initiating a dispute or an account inquiry.

Downloads - Siebel e-Billing Manager supports a variety of download formats that allow customers to conveniently manage their account information off line using personal financial management (e.g. Quicken, MS Money) or Microsoft Office (e.g. Excel, Access) applications. To meet most customers’ needs, Siebel supports OFX, QIF, CSV, XML, EDI and PDF download options.

Branch, Agency and Kiosk Access - Branch employees are granted access privileges to view and reprint customer’s e-bills through a standard web browser, expediting customer service requests. In addition, branch self-service kiosks can facilitate customer in-branch service needs.

CSR Call Center Access - Siebel e-Billing Manager provides customer service representatives (CSRs) with customer impersonation rights to access customer e-bills from a standard web browser. CSRs can easily facilitate both online and off line customer service requests since they can view the same statements that customers are inquiring about. Siebel customer service dashboard can incorporate additional operational data such as service/payment history and personalized marketing offers into CSRs’ views to give them a more comprehensive view of the organization’s relationships with customers.

Targeted Promotions and Messaging - Customized product offers or personalized service messages can be presented to customers within e-bills based on individual customer-specific account information. This creates additional revenue generation opportunities and improves marketing efficiency by reducing direct mail and telemarketing costs. Furthermore, since offers are so highly targeted, customers are more likely to respond to them.

e-Billing Manager Features Overview

e-Billing Manager is the leading electronic bill presentment and payment (EBPP) solution for communications service providers. e-Billing Manager provides the mission-critical application platform required for securely managing customer account information such as bills, statements, and other data. With e-Billing Manager, carriers can provide business and consumer customers with highly personalized online account information and self-service capabilities that can be served across multiple channels (Web, PDF, hand-held devices and e-mail). It serves as the foundation for managing recurring customer relationships.

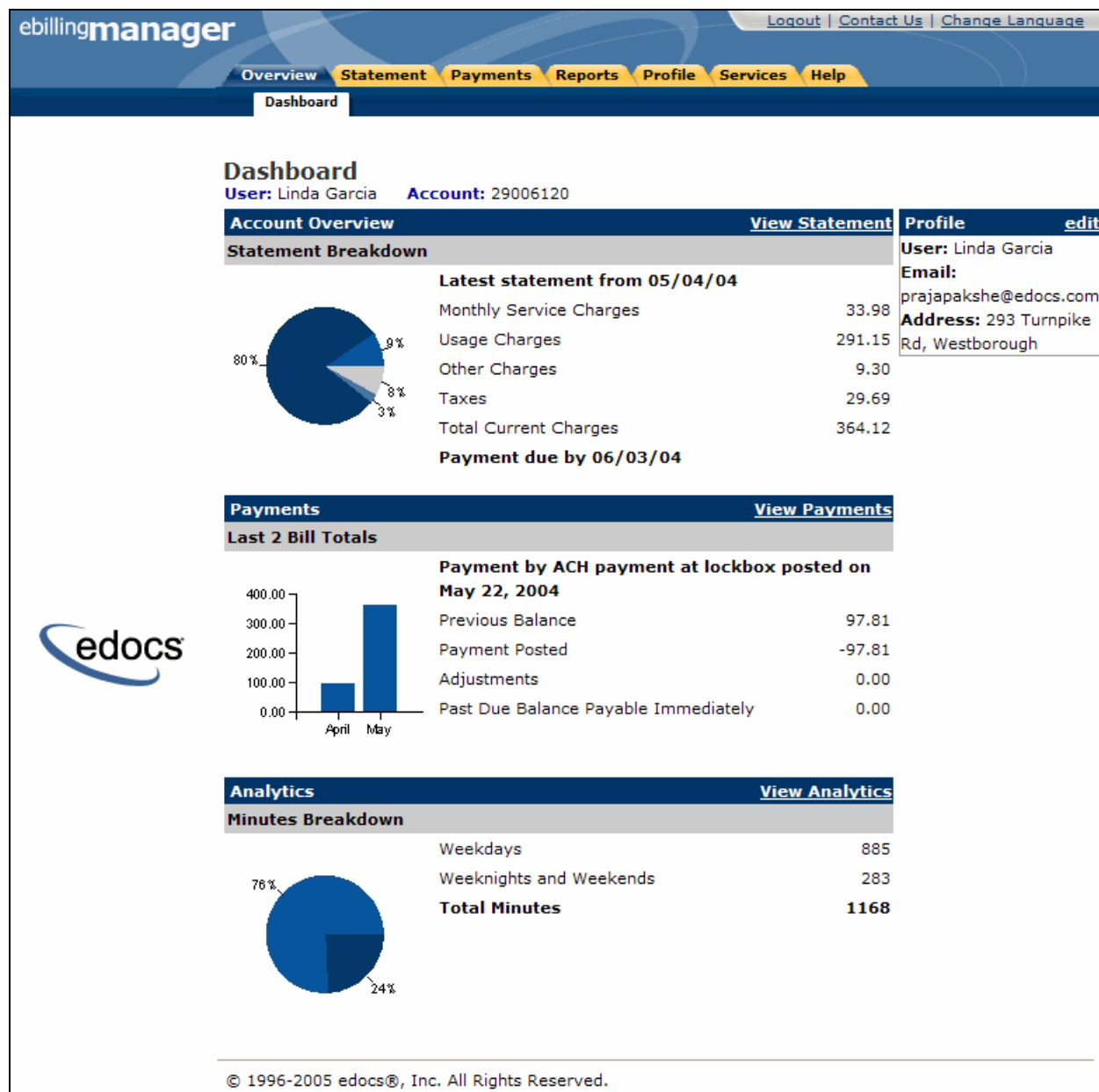
e-Billing Manager is specifically designed for organizations with large numbers of customers, high data volumes and extensive integration with systems and business processes across the enterprise. With its sophisticated data access layer, platform services and data stores, e-Billing Manager is uniquely capable of powering the most complex EBPP and Customer Self-Service solutions.

Realizing that consumer and business customers have very different service needs, e-Billing Manager is available in a consumer edition and business edition with specific features and functionality designed for each user base.

The core capabilities of the e-Billing Manager application editions are detailed below.

e-Billing Manager Consumer Edition

e-Billing Manager Consumer Edition dashboard



e-Billing Manager Consumer Edition is focused on the self-service needs of a large consumer base. The core features of the application are described below.

Customer Account Dashboard

Bills or statement data is dynamically presented within the application dashboard. Once properly authenticated, users can easily and conveniently navigate and view their billing statements, account summaries and bill details.

Electronic Bill Presentment and Consolidation

Consumer customers can access their bills or invoices online, including historical bills stored for as many years as the service provider prefers. Each monthly bill, for a single service or consolidated for multiple services, is presented in dynamic fashion using HTML and can include up-to-date account information such as current balance, bill details, unbilled-usage, last payment received, last payment date, etc. Customers can sort table columns, filter and drill-down into the call detail to further validate charges.

Electronic Bill Payment and Posting

Customers have the complete flexibility to control how and when payments are made. Payment is set up via bank and/or other credit accounts within e-Billing Manager to execute instant payments or to schedule future automatic payments. Customers have the ability to view the status and record of previously made payments. e-Billing Manager also provides an accounts-receivable settlement file for the service provider with the following information: payment amount, payment date, statement number, account number, payment account, payment type (one-time/recurring) and return status (yes or no with corresponding negative amount). Users can change their payment options and user preferences within the application. They can also establish multiple payment methods and payment notification preferences, make one-time payments, automate recurring payments, and set payment thresholds and reminders.

User Profile Management and Notifications

User information and settings are presented and modified within the application. Users may update their profile information, correct personal data, and manage password or notification options, email addresses, and other personal preferences.

Customer profiles may be updated on a real-time basis at anytime while the customer is properly authenticated. Profile information such as the customer's preference for account notifications can be set. e-Billing Manager manages and sends e-mail messages concerning individual accounts or transaction events. A variety of conditions or circumstances can trigger email notifications. Service providers may wish to enable some common notifications:

- Current statement available
- Payment reminders
- Payment confirmations
- Payment rejections
- Payment overdue

- Threshold exceeded on recurring payments
- Credit card expiration pending

Users may establish multiple notification or alert options and modify their settings at any time.

Unbilled Data Presentment

Unbilled account charges can be viewed online prior to bill cycle closing within the application. Users can view their charges as well as analyze to-date spending for the current billing period.

B2C Reporting

Specific reports come with the application whereby users may view summaries, analyze charts, and access their complete statement history using built-in reporting features. Furthermore, users may also drill-down through account information, bill or statement data within the application. Specifically, they can page through bill details, sort across pages, sub-total and filter account information viewing dynamic reports.

Print Friendly Views, PDFs and Downloads

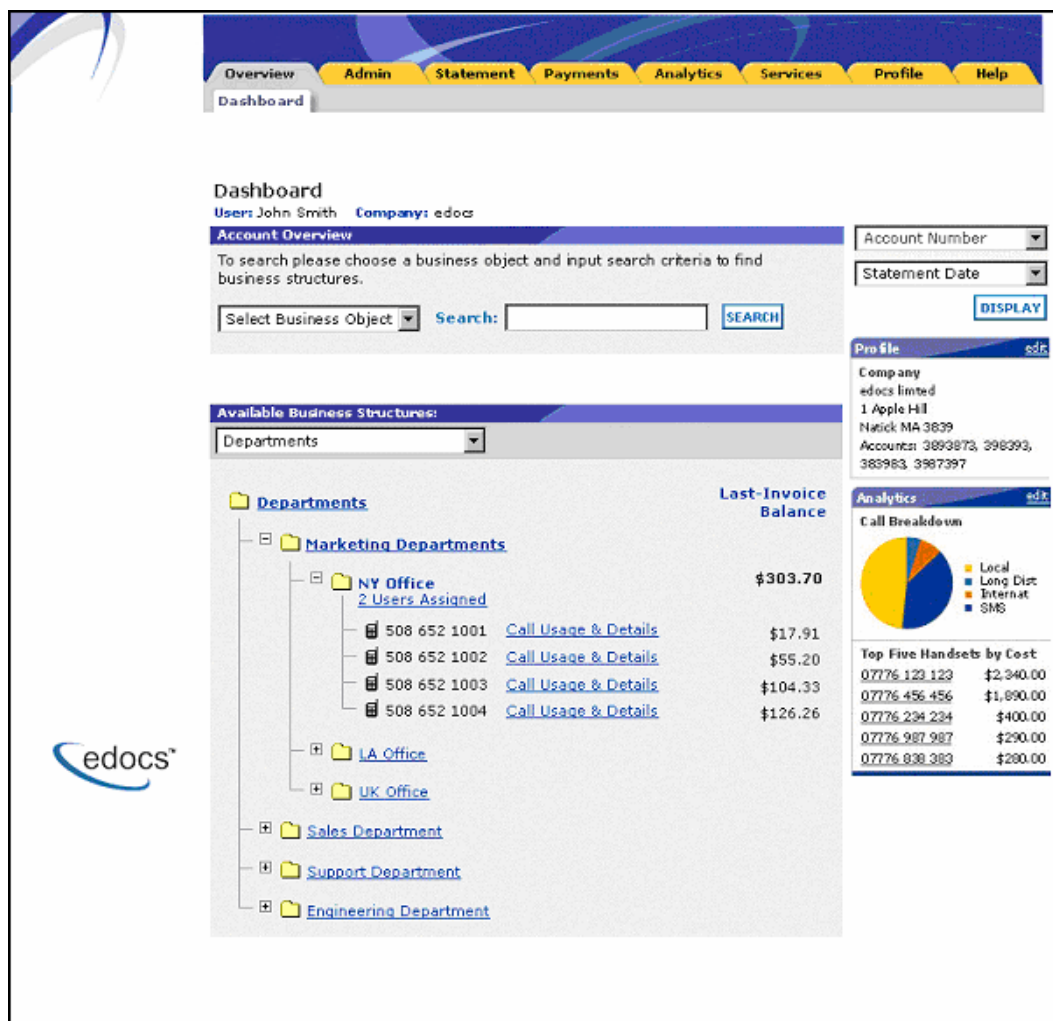
Users may access print friendly views, request a dynamic PDF of their statement, or select to download a dataset. These features are all available for key application views. e-Billing Manager's versioned, disk-efficient and high performance bill archiving feature along with its print-friendly views makes generating hard-copy reprints simple and efficient.

Customer Service Representative (CSR) Views

e-Billing Manager also provides a lightweight solution for customer service management. CSRs can also "impersonate" the user to facilitating the efficient servicing of online accounts. For example, representatives may have roles that provide application access to execute payments, reset passwords or simply search accounts to view the customer statement. The application may also be extended for complete case management with Siebel CSR Manager Application, providing additional case creation, routing and tracking, status views of cases, escalation workflows and queuing. Cases can either be completely automated without agent intervention based on predetermined business rules, or routed appropriately to engage an agent for resolution.

e-Billing Manager Business Edition

e-Billing Manager Business Edition Dashboard



e-Billing Manager Business Edition is focused on the self-service needs of an enterprise customer base. The core features of the application are described below.

Corporate Account Dashboard

Account information, bills or statement data is dynamically presented within the application by the business billing structure. Once properly authenticated, managers, administrators and users may view consolidated account summaries, sub-accounts, individual statements or all bill details based on their access permissions for the billing structure.

Hierarchical Billing and Business Structures

Billing structures are presented within the application as account hierarchies. Managers or administrators may search, view and update their billing structure by assigning other attributes to hierarchy nodes, such as meaningful names. Users are assigned within the billing structure, defining navigation within the account and sub-accounts of billing data. Administrator assignment in the billing structure may provide access to the entire enterprise or limit a single user to viewing their individual statement only. By creating separate business structures within the application, users may maintain multiple hierarchies of their own. Billing structures and user created business structures offer different views of account information for segmentation, payment, analysis and reporting.

Electronic Bill Presentment and Consolidation

Business customers can access their bills or invoices online including historical bills stored for as many years as the service provider prefers. Each billing period, for a single service or consolidated for multiple services, is presented in dynamic fashion using HTML and can include up-to-date account information such as current balance, bill details, last payment received, last payment date, etc. Furthermore, accounts can be consolidated across disparate system using either full data consolidation via an external data store or summary consolidation by dynamically linking the accounts. Customers can sort table columns, filter and drill-down into the call detail to further validate charges.

Cross Invoice/Sub-Invoice Payment

Payment options and enterprise preferences can be presented and modified within e-Billing Manager. Administrators may establish multiple payment methods and payment notification preferences, make one-time payments, schedule automated recurring payments, and set payment thresholds and reminders. Administrators may also make a single payment across multiple invoices, defining the payment allocation, even allocating payment to sub-invoice charges. Administrators have the complete flexibility to control how and when payments are made. Payment is set-up via bank and/or other credit accounts within e-Billing Manager to execute instant payments or to schedule future automatic payments. Administrators also have the ability to view the status and record of previously made payments. e-Billing Manager integration provides an accounts receivable settlement file to the service provider with the following information: payment amount, payment date, statement number, account number, payment account, payment type (one-time/recurring) and return status (yes or no with corresponding negative amount).

User Roles and Permissions

User roles and permissions can also be presented and modified within the application. Administrators may update user role or access, establishing view only permissions, accounts receivable roles, managerial access, etc. This feature may be configured based on the provider business requirements or back-office integration needs.

Corporate and User Profile Management and Notifications

Corporate profiles are managed by designated administrators, whereas users may update their own profile information, correct personal data, and manage password or notification options, email addresses, and other personal preferences.

Corporate and user profiles may be updated on a real-time basis at anytime while properly authenticated. Profile information such as the corporate contact preferences for account notifications can be set. e-Billing Manager manages and sends e-mail messages concerning accounts or transaction events. Notifications may be aggregated for account and sub-accounts events and limited to administrator or managerial roles. A variety of conditions or circumstances can trigger email notifications, so service providers may wish to enable some common notifications:

- Current corporate statements are available
- Payment reminders
- Payment confirmations
- Payment rejections
- Payment overdue
- Threshold exceeded on recurring payments
- Credit card expiration pending

Corporate administrators and users may establish multiple notification or alert options and modify their settings at any time.

Unbilled Data Presentment

With e-Billing Manager, managers, administrators and users may view unbilled account charges prior to bill cycle closing. All users may view their charges and analyze to-date spending for the current billing period. This feature may be configured based on the provider business requirements or back-office integration needs.

Advanced Reporting

Specific reports come with the application whereby users may view summaries, analyze charts, and access their complete statement history using built-in reporting features. Additional business reports are also provided whereby administrators may view a set library of summarized cost reports and charts. These “roll-up” reports provide for broad analysis of account activity for corporate managers. Furthermore, managers, administrators and users may drill-down through account information, bill or statement data within the application interface. Users can page through bill details, sort across pages, sub-total and filter account information viewing dynamic reports. More complex reporting can be performed with Siebel Analytics Manager including extensive call, cost, cost-center, and variance reporting.

Print Friendly, PDFs and Downloads

Users may access print friendly views, request a dynamic PDF of their invoice, or select to download a dataset. Managers and administrators have greater access to account summary PDFs and datasets for the accounts they supervise. e-Billing Manager's versioned, disk-efficient and high performance bill archiving feature along with its print-friendly views makes generating hardcopy reprints quick and easy.

Customer Service

CSRs may create corporate administrators for an organization, thereby enrolling a corporate account and establishing designated internal administrators. Service representatives and corporate administrators may create, manage and search the business organizations and accounts under their supervision. The designated administrators accessing corporate billing statements use e-Billing Manager's on-line self-service application features to manage their own organization and users.

Key Benefits of e-Billing Manager

Solving Business Problems

Many billing and customer care processes today are expensive, inefficient and unsatisfactory for both communication service providers and their customers. With Siebel e-Billing Manager, carriers empower their business and consumer customers to serve themselves, and address all their account and service-related activities online, instead of using expensive paper and call center based channels. For end customers, this makes doing business with a carrier more convenient, more efficient, and more satisfying. For carriers, it means improved competitive differentiation, significantly reduced customer care costs, increased customer loyalty, and streamlined billing/payment processing.

By combining comprehensive online account management functionality with the world's most scalable and reliable e-billing platform, e-Billing Manager enables service providers to gain outstanding adoption and ROI typically 5-10 times higher than companies using in-house, customer or competing packaged solutions. Some of the primary benefits e-Billing Manager enables include:

Business Benefits

Call Center Deflection

By enabling customers to serve themselves online, e-Billing Manager moves expensive call center interactions (globally these costs translate to \$5-\$10 USD per consumer interaction; \$10-\$30 USD per business customer interaction) to the web site where costs run at less than \$1 USD per interaction.

Reduced Paper, Printing and Postage Costs

In countries where paper turn-off is a legal option, providers are saving enormous printing, post-age, reprint and logistical costs. Even when rules do apply, to guarantee postage revenues, paper may be reduced for reprints services and logistical savings realized.

Streamlined Payment Processing

Online payments are proven to reduce payment time as well as errors associated with them. In addition, it is less expensive for the carrier to process an electronic payment when compared to all the manual handling required to process a paper-based payment. Online payment eliminates lockbox fees, minimizes exception processing, and replaces the costly and time-consuming procedure of processing paper checks.

Case Example		
Total customer base		19,700,000
× Online adoption	12%	
× Pct paying	30%	
× Savings per payment	\$0.14	
× Payments per year	<u>12</u>	
= Annual Savings		\$1,191,456

Some recent metrics:

- Adoption: 5%-60%
- Percentage paying: 30%-85%
- Savings per transaction: \$0.11 - \$1.08 USD

Given these numbers, an attractive ROI can be rapidly achieved.

Improved Days Sales Outstanding (DSO)

By making statements easier to access and read, and speeding approval cycles, e-Billing Manager reduces the time it takes for customers to pay their bills. This improves cash and reduces DSO.

Increased Customer Satisfaction (Reduce Churn)

By providing the option for 24/7/365 online service, carriers make it more convenient and more satisfying for their customers. Although somewhat more difficult to measure, increased customer satisfaction may ultimately be the most critical differentiator. This is particularly true in the B2B world of electronic invoice presentment and payment (EIPP). As evidenced by an independent AMR Research report, “The ROI exercise that generally launches EIPP projects cites process automation and cash flow as key benefits, but the biggest appeal of electronic billing, as it is used today, is its account management and customer support functionality. In reference to calls we conducted, Siebel users remarked that the main benefit of implementing EIPP is the dramatic improvement in customer service they can provide. In some cases, where the supplier took the extra step of helping customers incorporate EIPP into A/P workflows, satisfaction went through the roof. Reduction in DSO and paper-based billing costs – the quantifiable benefits of EIPP – paled as suppliers basked in the warm glow of happy customers.” In today’s economy of high customer acquisition costs, providing a robust online account management and electronic payment experience is critical to doing business and keeping your existing customers satisfied.

Application Benefits

e-Billing Manager is a packaged enterprise software application with support, training, and regularly scheduled product enhancements and upgrades. Standard deployments are predictable and deliver “quick wins” for the service provider.

Modularity

e-Billing Manager may be deployed and additional e-Billing Manager Suite applications added to the deployment, all using the complementary Siebel platform services.

Open Standards Based

e-Billing Manager is built on a Java 2 Enterprise Edition (J2EE™) architecture making it highly scalable and flexible. Based on leading Java standards like Enterprise Java Beans™ ((EJB™), Java Server Pages™ (JSP™), and Java Servlet application programming interfaces (APIs), users of e-Billing Manager can leverage all the sophisticated application management tools, enterprise-class performance, scalability, portability, and easy extensibility this environment provides. The Siebel applications presentation layer use struts and tiles standards and may be easily “re-skinned” for branding, internationalization, or customized look and feel.

Proven Scalability

e-Billing Manager has a linearly scalable architecture that supports millions of complex accounts and thousands of concurrent users. Its flexible data access layer provides for truly massive scalability in a way no competing architecture can claim. In fact, data volumes do not impact the real-time performance of the core Siebel solution at all. This means organizations can retrieve, process, and archive tens of millions of accounts, but only need to scale the application server tier to handle the growing user adoption. These services scale linearly, meaning the platform can easily be extended to accommodate growth in volumes, users, or additional application services. Independent scalability reports from Sun Microsystems and IBM that demonstrate this power can be provided upon request.

Ease of Deployment

The off line authoring components of the e-Billing Manager Suite visual development environment provide menus and wizards that dramatically reduce the time required to configure and deploy e-Billing Manager. Simulation capability is built-in allowing developers to test designs prior to deploying an application. Early testing lowers risks and costly late changes. Through the intuitive, easy-to-use graphical development environment, users can analyze the data sources, define the essential data, map the account data to presentation templates, and define business rules for one-to-one marketing and customer service messaging. Designers can create any aspect of account presentation for the Web, wireless, or other delivery devices.

Unmatched Data Access Flexibility

Siebel provides a flexible Business Services Layer (BSL) that can leverage a wide variety of data sources for account information/Tariff analysis. Extensible APIs provide a full suite of integration services for robust comprehensive functionality. Simply stated, e-Billing Manager transforms account data for Internet consumption. But e-Billing Manager is not a simple data translation application: it re-purposes available account data, dynamically retrieving only the information required for the new presentation media.

Simple Integration

e-Billing Manager preserves investment in and extends capabilities of existing infrastructure through enterprise wide integration. The Siebel design environment allows designers to develop and integrate custom features with existing system processes into the Customer Self-Service solution. Integration of custom functionality or external legacy systems can be accomplished quickly and easily using standard tools and technologies. Siebel eXtensible Modular Architecture (XMA) and Platform Service's APIs are Java-based with XML-interfaces detailed in the e-Billing Manager Software Developer's Kit (SDK). Integration is straightforward for linking existing enrollment and authentication systems, payment processing, order management processes, accounts receivable systems, customer service and CRM systems, middleware infrastructure, and third-party operational tools.

Operational Oversight

The Command Center console provides a single production management browser application for the e-Billing Manager Suite including scheduling of services, statement and account application configuration, customer statement notification configuration, auditing and event logging with general system administration and reporting.

Proven Performance

e-Billing Manager delivers superior performance without regard to data volumes or user loads.

Multiple Communication Channels

e-Billing Manager supports a multi-channel deployment paradigm for data delivery via, HTML, XML, IVR, PDF, Email, and devices.

Personalization

Any e-Billing Manager application view (or the entire template itself) can be replaced or modified based on business logic. By leveraging the account information contained in the data stream, e-Billing Manager can present highly personalized and dynamic views of a customer's account. Designers define logical expressions based on actual account data that will modify the presentation to the customer. This means that all account data can be used as variables in the dynamic of the rendered account view. The mapping of data elements to logical expressions is encapsulated in an XML file used at run-time.

Architecture Benefits

Reduced Storage Requirements

Competing approaches generally only offer transforming and storing statement data as XML or as normalized rows and columns in a database. The XML tags, which need to be stored as part of the data file, can increase file sizes by 30-40% in most cases. Siebel solution may store data in native file formats, and convert it to delivery formats (including XML) only on-demand. This dramatically reduces storage requirements. Data compression may also be optionally provided to further reduce storage by a ratio of greater than 3:1.

A Tunable Application

Billing information is the corner stone of any Customer Self Service solution, it is the data held within the bill that affects the bottom line of a customer's business or a consumer's likelihood to delay payment or switch providers. e-Billing Manager manages arriving bill data, processes it for storage, and archives the statement files into an extensible statement repository. A data store of account tables are also maintained in a relational database along with user management information and logging and tracking information. This data store may be "dialed down" using a lightweight metadata index to dynamically access statements, or "dialed up" creating a content repository of billing data. e-Billing Manager offers this option of loading bill detail or account summary information into a content repository for sophisticated customer reporting and analysis. The e-Billing Manager application and platform is therefore "tunable," providing a flexible approach to managing account-based communications for multi-channel delivery.

Reduced Database Licensing Requirements

Siebel data store generally comprises only 3% of the size of the raw statement data. Competing architectures, on the other hand, require a massive database deployment as the full content of each statement is stored as database rows. Database licensing implications can therefore add a significant cost load to competing solutions.

Enhanced Performance and Memory Management

Siebel utilizes dynamic data extraction and transformation for presentation of statement data. In addition, individual statements are decomposed into "views" such as summary, detail, sub-accounts, etc. Only the view being requested by the user is extracted and transformed. By carefully tuning view size, the developer retains complete control over the application's real time performance profile.

Improved Reliability and Data Quality

Competing architectures require batch data transformation to convert data formats. Any time data transformation occurs, there is a risk of user or system error where data will be "left on the cutting room floor," improperly truncated, or generally mis-mapped. Since transformation occurs in large-scale batch jobs, error recovery is very expensive. In the case of error, all of the errant data needs to be identified and purged from the database, the source data needs to be recreated, transformation rules corrected, and the batch jobs rerun. If the batch cycle is long, the correction may be impossible if several days have passed before the error was detected. Siebel utilizes "dynamic data transformation" whereby data is transformed on the fly by dynamically applying extraction and presentation rules. As such, if an error should occur, the rules can be fixed and re-published on the fly without having to rerun the batch process or drop data.

Reduced Batch Cycle

Siebel batch cycle consists of scanning and indexing arriving batch files. Depending on the file format, the batch process typically runs at 7.5 GB of source data per hour. The process may even be configured to run parallel across server clusters for greater throughput. Competing architectures are simply not able to achieve this load time, leading to consumption of additional system resources and higher stakes for error recovery.

Robust Version Management

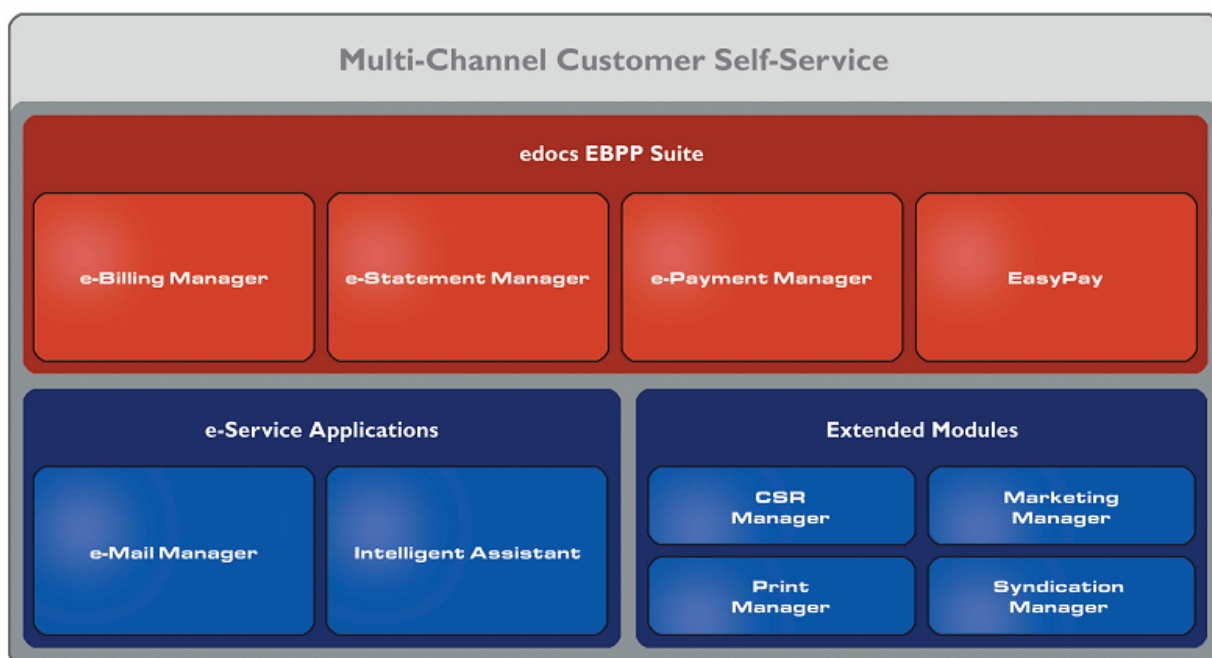
Siebel realizes the frequency with which businesses change: new products are introduced, statements are redesigned, and old products are discontinued. As such, Siebel makes extensive use of a template-based architecture and provides robust capabilities for versioning templates. Essentially, rules for transforming and presenting data are versioned by date and stored with the source files they correspond to. As such, if a new service offering or bill format is introduced, a developer need only publish new templates which accommodate the changes and need not worry about affecting historical statements. Competing approaches to changing fixed relational database schemas are far more complicated to manage. These advantages translate in to a lower total cost of ownership, which offers returns every day that the system is in production.

The EPP Suite

Siebel Electronic Presentment and Payment (EPP) Suite is a set of modular applications that include every capability organizations need to enable a complete online account management experience. The suite is comprised of four packaged applications:

- e-Billing Manager
- e-Payment Manager
- e-Statement Manager
- EasyPay

The suite combines the industry's most comprehensive e-billing, e-invoicing and e-payment functionality with Siebel extensive industry experience. These out-of-the-box applications have been deployed in some of the most challenging business and technical environments worldwide and are proven to be the industry's most scalable and reliable. As a result, Siebel customers consistently gain the highest customer adoption and ROI in their industries -- often five, ten, even twenty times higher than companies using in-house, custom or competing packaged solutions.



e-Billing Manager

Siebel e-Billing Manager gives consumer and business customers' valuable and convenient online access to their bills or invoices, along with the ability to easily make online payments. Bills and invoices encapsulate all of the activity and charges associated with the exchange of goods or services between a customer and an organization and as such, are at the foundation of the relationship.

- HTML Presentment
- Drill-down
- Consolidated View
- Bill/Invoice Archiving
- Paper Turn-off Support
- Printer-friendly Views
- Reporting
- Targeted Messaging
- Proactive Advanced e-Care
- Notifications
- Profile Management
- Electronic Payments
- Payment Scheduling

- Payment Status and History
- Accounts Receivable Reconciliation

e-Statement Manager

Siebel e-Statement Manager enables organizations to deliver interactive and personalized electronic account statements to both business and consumer customers. The application provides a more cost-efficient way to present current and archived statements online. Organizations, especially financial institutions and healthcare insurance carriers, are deploying e-statements as part of a comprehensive online customer care strategy.

Realizing that not all customers require paper statements and that many prefer to access their account information online in a more convenient and valuable way, organizations are switching to the online model and are reaping the benefits.

- HTML and PDF Presentment
- Statement Consolidation
- Drill-down
- Statement Archiving
- Printer-friendly Views
- Paper Turn-off Support
- Reporting
- Targeted Messaging
- Proactive Advanced e-Care
- Notifications
- Profile Management

e-Payment Manager

Siebel e-Payment Manager is a complete solution for electronic payments executed by customers enrolled in e-billing and e-invoicing. It plugs into existing applications and systems, preserving investments in, and extending the capabilities of existing infrastructure. With Siebel e-Payment Manager, organizations decrease payment-processing costs, accelerate receivables, and improve operational efficiency in the areas of billing and collections, customer service and treasury management.

EasyPay

Siebel EasyPay is a convenience payment solution that enables organizations to integrate web, response (IVR) and contact center payment channels to accept one-time, often last minute, using a wide range of payment instruments, eliminating the existence of multiple technology “silos” that are often used to collect payments from customers.

- Web, CSR, and IVR-based Payment
- Payment Scheduling
- Summary View
- Drill-down
- Error and Fraud-Prevention
- Multi-lingual Interfaces
- Reporting and Auditing
- Organizational Branding
- Hosted and In-house Deployment Models

e-Service Applications

Whether customers are visiting an organization's web site, communicating by email, or seeking to chat real-time with a CSR, Siebel e-Service Applications ensure the ability to deliver knowledgeable and exceptional customer service. Driven by sophisticated intelligence engines and automation technologies, these solutions replicate the knowledge of an organization's most experienced personnel, providing timely, accurate responses to customer inquiries.

e-Mail Manager is an automated e-mail response management system that determines the intent of the incoming e-mail messages and composes personalized answers that can be automatically dispatched to customers or routed to service agents for a single-click review.

Intelligent Assistant is an advanced natural language-based self-service application that empowers customers, prospects and customer service representatives (CSRs) to leverage all of an organization's knowledge assets-web pages, account data, documents, databases, existing legacy data sources, and knowledge bases to quickly and accurately find answers to their specific billing, account, product and service questions.

Extended Customer Service Modules

Siebel Extended Customer Service Modules augment Siebel core e-billing and online self-service capabilities and extend them to additional users, channels and endpoints. This enables organizations to provide more effective and efficient service regardless of what service channel customers choose.

Print Manager is a complete solution for data consolidation, visual statement formatting and design, and print output generation that significantly reduces the cost and complexity of producing paper bills, invoices and statements. Siebel combined electronic and print output solution handily solves the challenge of account consolidation avoiding the need to alter complex back-end legacy systems to present a consolidated account view online or on paper.

Syndication Manager is an account content distribution system that handles all the complexities of securely distributing summary account information to any endpoint, while also enabling customers to go back to the billing organization's website to take advantage of more comprehensive self-service capabilities.

CSR Manager enables customer service representatives (CSR) to access critical account data and service-related information and capabilities to effectively service both online and off line customers. CSR Manager is a browser-based application that couples traditional customer-facing online self-service capabilities with CSR-specific features including case management, facilitating better service at the point of customer contact.

Marketing Manager is a personalization, campaign and content management solution that weaves personalized marketing and customer service messages based on specific account information throughout the customer self-service and e-billing experience. The browser-based application facilitates collaboration between internal marketing and customer service departments as they create, deploy and track the performance of campaigns.

The Siebel Advantage

Siebel EPP Suite has the lowest total cost of ownership (TCO) of any self-service approach. It provides an optimal TCO because of the following key attributes:

User Scalability - Siebel platform demonstrated unparalleled scalability and performance in testing by IBM and Sun. The platform routinely supports millions of accounts and online users without performance degradation for Siebel individual global 2000 customers around the world. This translates into less hardware and lower IT costs for Siebel customers.

Standards-Based - Siebel J2EE standards-based platform and modular component architecture meets the demanding requirements of leading organizations.

Integration - Siebel EPP Suite integrates with existing infrastructure including web portal, call center, IVR and other enterprise applications to maximize the solution's ROI. The application uniquely accesses a variety of diverse data sources (e.g. legacy, print source, data warehouse) that most applications cannot, providing a consolidated view of customer accounts.

Time to Market - Siebel EPP Suite allows for rapid deployment because it comes as a complete packaged, out-of-the-box application that is tailored to solve distinct business problems and meet industry specific process requirements.

Conclusion

Siebel e-Billing Manager is the proven software platform for scalable, high-performance self-service and e-billing applications that enable organizations to manage account relationships with business and consumer customers online. e-Billing Manager application solutions:

- Reduce support and processing costs
- Improve operational effectiveness
- Increase customer satisfaction
- Enhance marketing effectiveness

e-Billing Manager's out of the box consumer and business applications for data access, online account composition, analytics, one-to-one messaging, and business hierarchy dramatically decrease time to market for deploying self-service solutions. Developed on J2EE technology, the e-Billing Manager platform is flexible, extensible, and easy to manage, and is the proven platform for online consumer and business applications.

e-Billing Manager leverages available data from transactional systems to dramatically reduce the costs associated with producing, delivering, and paying account communications, while simultaneously transforming accounts into dynamic, interactive self-service and marketing tools.

e-Billing Manager goes far beyond bill presentment solutions providing packaged applications with many e-billing features, including online account and statement composition, payment, business logic, one-to-one marketing, hierarchy, customer service access, and service messaging with application management.

e-Billing Manager's EJB platform architecture has been proven scalable in production and through independent testing. This is a key differentiator in comparing packaged solutions.

Finally, because of e-Billing Manager's flexibility and methodology, time to market with an e-Billing Manager Suite solution is extremely rapid, stable, and scalable resulting in a total cost of ownership lower than competing or home grown solutions.

About This Guide

This guide is intended for system administrators or other IT professionals responsible for setting up and running a live e-Billing Manager application in a J2EE environment. It describes the general process and specific procedures required to:

- Set up a new e-Billing Manager application and the associated jobs using the Siebel Command Center.
- Produce regular online statements electronically and manage the ongoing live production process.

This guide assumes you have:

- Installed e-Billing Manager in your J2EE environment.
- Designed and developed the necessary application files (using DefTool and Composer in a Windows 2000 environment).

This guide does *not* describe general UNIX or Windows system administration. See the appropriate UNIX or Windows user documentation.

Related Documentation

The following online Help is available in the Siebel Command Center:

Online Help	How to Access
Comprehensive Command Center Help	Click Help on the Command Center menu. Help contains additional information about administering your application in a live production environment.
Screen-level Command Center Help	Click the Help button on a screen for details about that particular screen. Click Help Contents there to access the complete administration Help.
A PDF version of this guide	A PDF version of this document is located on your product CD.

This guide is part of the e-Billing Manager documentation set. For more information about implementing your application, see one of the following guides:

Print Document	Description
e-Billing Manager <i>Installation Guide</i>	How to install e-Billing Manager and configure it in a distributed environment.
e-Billing Manager <i>Data Definition Guide</i>	How to create Data Definition Files (DDFs) for use in indexing your application and extracting data for live presentment.
e-Billing Manager <i>Presentation Design Guide</i>	How to create Application Logic Files (ALFs) to present statement data for dynamic online display.

Obtaining Siebel Software and Documentation

You can download Siebel software and documentation directly from Customer Central at <https://support.edocs.com>. After you log in, click on the Downloads button on the left. When the next page appears, you will see a table displaying all of the available downloads. To search for specific items, select the Version and/or Category and click the Search Downloads button. If you download software, an email from Siebel Technical Support will automatically be sent to you (the registered owner) with your license key information.

If you received an Siebel product installation CD, load it on your system and navigate from its root directory to the folder where the software installer resides for your operating system. You can run the installer from that location, or you can copy it to your file system and run it from there. The product documentation included with your CD is in the Documentation folder located in the root directory. The license key information for the products on the CD is included with the package materials shipped with the CD.

If You Need Help

Technical Support is available to customers who have an active maintenance and support contract with Siebel. Technical Support engineers can help you install, configure, and maintain your Siebel application.

This guide contains general troubleshooting guidelines intended to empower you to resolve problems on your own. If you are still unable to identify and correct an issue, contact Technical Support for assistance.

Information to provide

Before contacting Siebel Technical Support, try resolving the problem yourself using the information provided in this guide. If you cannot resolve the issue on your own, be sure to gather the following information and have it handy when you contact technical support. This will enable your Siebel support engineer to more quickly assess your problem and get you back up and running more quickly.

Please be prepared to provide Technical Support the following information:

Contact information:

- Your name and role in your organization.
- Your company's name
- Your phone number and best times to call you
- Your e-mail address

Product and platform:

- In which Siebel product did the problem occur?
- What version of the product do you have?
- What is your operating system version? RDBMS? Other platform information?

Specific details about your problem:

- Did your system crash or hang?
- What system activity was taking place when the problem occurred?

- Did the system generate a screen error message? If so, please send us that message. (Type the error text or press the Print Screen button and paste the screen into your email.)
- Did the system write information to a log? If so, please send us that file. For more information, see the *e-Billing Manager Troubleshooting Guide*.
- How did the system respond to the error?
- What steps have you taken to attempt to resolve the problem?
- What other information would we need to have (supporting data files, steps we'd need to take) to replicate the problem or error?
- **Problem severity:**
- Clearly communicate the impact of the case (Severity I, II, III, IV) as well as the Priority (Urgent, High, Medium, Low, No Rush).
- Specify whether the problem occurred in a production or test environment.

Contacting Siebel Technical Support

You can contact Technical Support online, by email, or by telephone.

Siebel provides global Technical Support services from the following Support Centers:

US Support Center

Natick, MA
Mon-Fri 8:30am – 8:00pm US EST
Telephone: 508-652-8400

Europe Support Center

London, United Kingdom
Mon-Fri 9:00am – 5:00 GMT
Telephone: +44 20 8956 2673

Asia Pac Rim Support Center

Melbourne, Australia
Mon-Fri 9:00am – 5:00pm AU
Telephone: +61 3 9909 7301

Customer Central

<https://support.edocs.com>

Email Support

<mailto:support@edocs.com>

Escalation process

Siebel managerial escalation ensures that critical problems are properly managed through resolution including aligning proper resources and providing notification and frequent status reports to the client.

Siebel escalation process has two tiers:

1. **Technical Escalation** - Siebel technical escalation chain ensures access to the right technical resources to determine the best course of action.
2. **Managerial Escalation** - All severity 1 cases are immediately brought to the attention of the Technical Support Manager, who can align the necessary resources for resolution. Our escalation process ensures that critical problems are properly managed to resolution, and that clients as well as Siebel executive management receive notification and frequent status reports.

By separating their tasks, the technical resources remain 100% focused on resolving the problem while the Support Manager handles communication and status.

To escalate your case, ask the Technical Support Engineer to:

1. Raise the severity level classification.
2. Put you in contact with the Technical Support Escalation Manager.
3. Request that the Director of Technical Support arrange a conference call with the Vice President of Services.
4. Contact VP of Services directly if you are still in need of more immediate assistance.

2 General Troubleshooting Process

How to Begin Troubleshooting

If you have a problem with your application, there are several steps you should take to get the troubleshooting process underway.

There are tools and resources available on your system and the Siebel Web site can help you gather information about what was happening in your application when the problem occurred and determine how to correct it.

Please open a Technical Support case if you are unable to resolve the problem using the guidelines in this book. Assist your support engineer by having important system information at hand to enable your Support engineer to more quickly resolve the problem. See “If You Need Help” on Page 29 for details.

Follow these general steps to troubleshoot a problem:

1. Determine the customer impact; if the problem is in production, is batch processing or statement presentment affected? How many customers are affected (100% or a few)? Are payments affected?
2. Verify the product version and patch level.
3. Check whether you have a customized build of the product and whether there have been any recent changes made.
4. Use the Command Center to verify job and task status and examine the error message.
5. Establish when the problem started and how often it occurs. Did it ever work? Did a dependent system fail, like db or an integration point? Does it happen in other environments? Has anything changed or been added since it last worked?
6. Look at application and system logs for error or exception detail. Are all customers seeing this error? How many exceptions are in the log? When did the first exception occur? Is there a pattern?
7. Use the information in this guide and the Customer Central Knowledge Base (KB) to interpret exceptions or errors and to find possible causes and solutions, in particular, see:
 - The release notes for your version to see known issues.
 - The appropriate section in this guide on troubleshooting a **job failure**.

- The KB to search for details about your problem.
8. Make any necessary corrections or changes.
 9. Try to rerun the job or resume normal processing.

Verifying Product Version

It is important to know the version of your Siebel product and the versions and patch levels of your platform to:

- Look up specific troubleshooting information about your product and version in the Siebel Customer Central Knowledge Base.
- Provide your Siebel Technical Support engineer if you open a support case.

Verify and provide Technical Support with version and patch levels of:

- WebLogic
- WebSphere
- Operating System
- Database Server
- Database Driver
- Middleware (MQ Series)
- Your Siebel product

Verify the patches required in the Siebel Release Notes for your product are correctly installed.

To determine which Siebel product version you have:

- View the content of the version.txt file in your root *EDX_HOME* installation directory.

(The Siebel version also indicates patch level.)

Log Files

Whenever a problem arises, review the e-Billing Manager application logs along with your system logs for details about any errors that occurred and what activity was taking place at the time.

e-Billing Manager

e-Billing Manager maintains logs of all activities that occur and messages generated during production. You can use the Command Center to create a report showing the following types of log messages generated over a select time period:

- **Error** – Error log
- **Information** – Activity log
- **Warning** – Warning log

It is a good idea to review these logs on a regular, ongoing basis to monitor jobs in your production environment.

Log reports display the following information:

Log Report Column	Description
Timestamp	The date and time the message was created in the log.
SourceHost	Name of the server that generated the error message or where the production activity occurred.
Message ID	A code identifying the e-Billing Manager task where the error occurred and the level of error.
Message	Message text.

To view e-Billing Manager production logs:

1. Click Reporting on the Command Center menu. The Reporting screen appears.
2. Click the **View Logs** tab to display the View Logs screen.
3. Select the type of message log to view.
4. Enter a start date and end date range to search. Click **Popup Calendar** to select dates quickly.
5. Enter a start time and end time to search.
6. Click **Submit Query**.
7. e-Billing Manager displays the log messages of the selected type generated during the selected date and time range.
8. To select different log information to view, click **Reselect Log View**.

You can also view this information directly in the e-Billing Manager system_activity database table.

WebSphere

WebSphere maintains a set of logs to record various activity, error, and output information.

WebSphere Log File Name	Type of Information
Default_Server_stderr	Exceptions
Default_Server_stdout	Debug information; inputs and normal auditing of operations
tracefile	Server startup problems (IBM-specific)

WebSphere uses “Default_Server” or whatever you named your server.

Location of WebSphere Logs

WebSphere generates the logs to %WAS_HOME%/logs, where %WAS_HOME% is the directory specified when the server instance was created in WebSphere (for example, C:\WebSphere\AppServer\logs).

WebLogic

WebLogic maintains a set of logs to record various system event data.

WebLogic Log File Name	Type of Information
weblogic.log	All console output for the server
access.log	Record of files accessed from the server.

Using weblogic.log and access.log

Timestamps indicate the start and end of a log entry.

In weblogic.log, look for entries with <error> or <exception>; you can ignore <notice> and <info> entries.

Access.log contains the IP of the person who accessed the page, the URL of the page they accessed, and HTTP errors. An error 200 appears if it was successful; other error numbers may indicate a problem.

For example, in the following access.log entry, the HTTP status code is 500, which is a server error:

```
10.xxx.xxx.x71 - - [21/Oct/2003:15:29:19 -0700] "GET
//Payment/Payment?app=Payment&ddn=acme&acctNum=8744&form=payCr
edit&amountDue=406.67&billId=ivn-3592/po-17334386/bc-2230/pc-
1/dd-20030912 HTTP/1.1" 500 0
```

If you got a 500 error there must be an exception in the weblogic.log. Look for an exception in weblogic.log from around this time, such as:

```
####<Oct 21, 2003 3:29:19 PM PDT> <Info> <EJB> <server1>
<myserver> <ExecuteThread: '12' for queue: 'default'> <> <>
<010051> <EJB Exception during invocation from home:
com.edocs.ps.acme.DBAccessBean_33lj8w_HomeImpl@7bdfd0 threw
exception: java.rmi.RemoteException: EJB Exception;; nested
exception is:

    java.rmi.RemoteException: ORA-01403: no data found
ORA-06512: at "EBPP.ACME_DBACCESS", line 133
ORA-06512: at line 1
>
```

For a description of HTTP error codes, see:

<http://www.w3.org/Protocols/HTTP/HTRESP.html>

You can also use access.log to generate a chronological list of actions taken by the user. Sorting the entries by IP yields the sequence of steps the user took that generated the error or problem.

Be sure to look at the access.log for the customer-facing server.

Location of WebLogic Logs

WebLogic generates the logs to %WL_HOME%/config/mydomain/logs, where %WL_HOME% is the directory specified when the instance was created in WebLogic (and mydomain is as you have configured).

Solaris

To create a log, you must add a nohup command to your WebLogic or WebSphere startup scripts. You specify the directory in the nohup command.

Solaris Log File Name	Type of Information
nohup.out (or other name you specify at startup)	Debug information

On WebLogic, stdout and stderr information goes to nohup.out if you start WebLogic via nohup (a process immune to hangups) with no redirection to an output file, for example (using Bourne shell syntax):

```
% nohup startWebLogic.sh &
```

Having a single nohup.out file isn't very practical, however, because it gets overwritten every time WebLogic restarts, and is also not a very functional name. To maintain a history of log files, redirect the output from nohup to a log file using an appropriate naming convention. For instance, using Bourne shell syntax, to launch WebLogic with stdout and stderr information going to console_weblogic_admin_server.log, use the command:

```
% nohup startWebLogic.sh > console_weblogic_admin_server.log  
2>&1 &
```

It is also a good idea to append the log file name with the timestamp of when WebLogic was started.

Interpreting Exceptions

An exception is an error that the application or system generates when it encounters a problem while running in a live environment. You must correct the problem before processing can resume normally. Exceptions can appear in:

- The Siebel error log
- J2EE server logs (for example, weblogic.log in WebLogic, and Default_Server_stderr.log for WebSphere). These paths are configurable in the J2EE console.
- The Nohup.out log for debugging Solaris
- The Windows console (when not running as a service for debugging)

It is possible for an Siebel exception to get captured in the server log only, so be sure to check all relevant logs. See “Log Files” on page 34 for details. When an exception occurs, it generates a Java call stack trace in one or more logs. A stack trace is a list of raw audit information about what was happening when the exception occurred. The most recently executed code is at the top and usually contains the error message of interest.

Sample stack trace:

```

Servlet failed with IOException
java.rmi.RemoteException: EJB Exception: ; nested exception is:
javax.ejb.EJBException: java.rmi.RemoteException:
java.sql.SQLException: ORA-01001: invalid cursor
ORA-06512: at "EDX_DBA.Acme", line 148
ORA-06512: at line 1
javax.ejb.EJBException: java.rmi.RemoteException:
java.sql.SQLException: ORA-01001: invalid cursor
ORA-06512: at "EDX_DBA.Acme", line 148
ORA-06512: at line 1
.at
com.edocs.services.application.AppIndexVolMgr.getHitListByIvn(AppIndex
VolMgr.java:850)
.at
com.edocs.services.application.AppIndexVolMgr_fcksga_EOImpl.getHitList
ByIvn(AppIndexVolMgr_fcksga_EOImpl.java:348)
at com.edocs.app.verify.Verify.getHitList(Verify.java:584)
.at com.acme.util.RejectProcessor.getAccounts(Unknown Source)
.at com.acme.util.RejectProcessor.processRejects(Unknown Source)
.at com.acme.util.RejectProcessor.<init>(Unknown Source)
.at
jsp_servlet.__accountnumberlist.__jspService(__accountnumberlist.java:1
32)
.at weblogic.servlet.jsp.JspBase.service(JspBase.java:27)
.at
weblogic.servlet.internal.ServletStubImpl.invokeServlet(ServletStubImp
l.java:265)
.at
weblogic.servlet.internal.ServletStubImpl.invokeServlet(ServletStubImp
l.java:304)
.at
weblogic.servlet.internal.ServletStubImpl.invokeServlet(ServletStubImp
l.java:200)
.at
weblogic.servlet.internal.WebAppServletContext.invokeServlet(WebAppSer
vletContext.java:2546)
.at
weblogic.servlet.internal.ServletRequestImpl.execute(ServletRequestImp
l.java:2260)
.at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:139)
.at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:120)

```

To troubleshoot an exception:

1. Look at the stack trace and identify some basic information. The top few lines of the call stack usually contain the most important information, such as:
 - **Whether it's an Oracle (SQL) or Java error.** In the sample exception above, it's an Oracle error, not Java error, so it originated in the database.
 - **Whose code is the problem.** If you see **com.edocs** the problem is related to Siebel code and you can look it up in the Knowledge Base on <https://support.edocs.com>.
 - **Description and other information.** You may see a brief description of the error. In the sample, you see "invalid cursor." (EDX_DBA.Acme is the package name.)

2. Search the Siebel Knowledge Base (KB) for the exception number. The KB is the most up-to-date repository of specific troubleshooting information. The Siebel Technical Support staff maintains the KB on a regular basis with resolutions to issues.

If the exception occurred due to a Java problem, it indicates the name of the class file. The top Java method shown is the one that threw the exception. If necessary, your Siebel Technical Support engineer can trace a stack trace back through the methods to the related code to determine the cause of the problem. Be sure to have the exception and stack trace information at hand if you contacts Siebel Technical Support. It is best to attach this to the case or, if brief, paste it into the case body.

Using Debug

About debugging

Setting debug keys for various system features enables debugging, letting you trace and expose specific code-related information about what is occurring in your Siebel software, including input parameter values used to call a method, results, etc.

Each debug flag is designed to debug a particular area of functionality where you're having a problem, such as EmailNotification, user enrollment, or presentment, and starts debugging whenever that feature is executed.

For example, if a stack trace for a crash mentions libformatter, setting a debug flag can show which view it was trying to display. This information could help determine whether an ALF was deleted, the ALF path is incorrect, if there is a problem with read permissions, or some other related issue.



Running with debugging enabled can slow system performance. Some debug flags are too low-level to be useful for debugging. Debug message content varies in format and may be of limited use to non-programmers or for general troubleshooting purposes.

Using Java flags

To set debug keys using Java flags:

On WebSphere:

1. Add debug keys to the system properties in the JVM Settings tab for each server in the WebSphere console. For example:

```
Name = com.edocs.payment.debug  
Value = true
```
2. Restart the servers.

On WebLogic:

Windows

Debugging does not produce an output in Windows when e-Billing Manager is running as a service. You must run e-Billing Manager from the command line while you are in debug mode. This “logs” the output to the console on your desktop. You can also redirect that output to a file.

- It is best to pass the debug Java flag in edx.config.bat. For example:
Set JAVA_OPTIONS="\$JAVA_OPTIONS -DMAIL_DEBUG_KEY=1"

Unix

- In edx.config:
JAVA_OPTIONS="\$JAVA_OPTIONS -DMAIL_DEBUG_KEY=1"
EXPORT JAVA_OPTIONS

Debug keys you can set as Java flags

e-Billing Manager Debug Key	Use to debug...
-DJAVA_ONLY	Bypass core classes to compose test HTML statements (to establish whether the problem is in Java or core classes).
-Dcom.edocs.fs.logging.System.out=true	Sends all log messages to System.out rather than JMS.
-Dcom.edocs.app.chart.debug=true	e-Billing Manager Charting
-Dcom.edocs.tasks.mns.debug=true	EmailNotification Job
-Dcom.edocs.mns.db.debug=true	EmailNotification Job
-Dmail.debug=true	EmailNotification Job
-Dcom.edocs.core.debug=true	JNI
-Dcom.edocs.services.application.debug=true	Application (DDN related)
-Dcom.edocs.services.mailer.debug=true	Email issues
-Dcom.edocs.services.merger.debug=true	Bill presentment
-Dcom.edocs.services.reporting.debug=true	Reports
-Dcom.edocs.services.versioning.debug=true	Version set/views
-Dcom.edocs.pwc.debug=true	Command Center
-Dcom.edocs.pwc.tasks.CommonTask.debug=true	Command Center
-Dcom.edocs.enrollment.user.debug=true	User enrollment
-Dcom.edocs.enrollment.user.edocs.debug=true	User enrollment

e-Billing Manager Debug Key	Use to debug...
-Dcom.edocs.enrollment.access.debug=true	User enrollment
-Dcom.edocs.app.publisher.debug=true	Publisher
-Dcom.edocs.app.user.debug=true	Presentment
-Dcom.edocs.services.session.debug=true	ISession
-Dcom.edocs.pwc.db.debug=true	Scheduler
-Dcom.edocs.tasks.staticHtmlFormatter.debug=true	StaticHTMLFormatter task
-Dcom.edocs.tasks.ixloader.debug=true	IXLoader task
-Dcom.edocs.tasks.indexer.debug=true	Indexer task
-Dcom.edocs.tasks.ivnscanner.debug=true	IVNScanner task
-Dcom.edocs.tasks.shellcmd.debug=true	Shell commands
-Dcom.edocs.tasks.scanner.debug=true	Scanner task
-Dcom.edocs.payment.debug=true	General Payment functionality
-Dcom.edocs.payment.encrypt.debug=true	Payment for encryption-related issues
-Dcom.edocs.eapost.debug=true	General Syndication Manager functionality

Setting as an environmental variable

To set debug keys as an environment variable:

For all platforms, it is a good idea to set debug flags environment variables in the e-Billing Manager configuration file, `edx.config`.

To set the debug flag as an environmental variable in Unix:

- Set the following environmental system variable, as:

```
EDOCS_DUMP_JNI_ARGS=1
export EDOCS_DUMP_JNI_ARGS
```

To optionally set the debug flag as an environmental variable in Windows:

Add the debug key to `edx.config.bat`:

- Add a new line at the top right, after the “Main:” line, and add the key, as in:

```
SET EDOCS_DUMP_JNI_ARGS=1
```

- Restart the server.

Debug keys you can set in the environment

e-Billing Manager Debug Key	Use to debug...
EDOCS_DUMP_JNI_ARGS=1	Input and output of the libformatter.so

Using Log4j

We recommend Log4j, an open source tool that can simplify logging. It inserts log statements into your code, a low-tech method for debugging. Log4j lets you enable logging at runtime without modifying the application binary. The log4j package is designed to include log statements in shipped code without incurring a heavy performance cost. See logging.apache.org for more information.

Reviewing Recent System Changes

Find out what changes, if any, have been made to your system recently, including:

- Upgrades or point-releases
- Any new features
- Any changes to related software or hardware
- Customizations to your application made in-house or by Siebel Professional Services
- Using Customer Central

Customer Central is the primary repository of information about Siebel products.

Day or night you can use Customer Central resources to troubleshoot your system, learn more about how to use your application, and download product documentation.

Use Customer Central to access the following Siebel resources:

Knowledge Base – The Siebel Knowledge Base (KB) is the most up-to-date repository of specific troubleshooting information about Siebel products. Our extensive Knowledge Base contains thousands of articles and is continually updated by the Siebel Technical Support staff. The KB is a great self-help tool to find answers and helpful information or troubleshoot problems. Knowledge Base Articles (KBA's) have been arranged logically so that you can find information browsing by category, viewing frequently asked questions, or using keyword, key-phrase searches. There are several ways to access information in the KB:

- Browse Knowledge Base – Lets you browse KBA's based on specific categories. Articles displayed in each category have been created or modified within the time frame you specify in a drop down box. Lets you quickly view recently posted KBA's.

- **Frequently Asked Questions** – Lists KBA's most frequently referenced by customers and the Siebel Technical Support team.
- **Search Knowledge Base** - Use the search interface to quickly perform a keyword search across all Siebel KBA's.
- **My Notifications** – Subscribe to receive notification if a KBA you subscribe to is updated.
- **Top 25 Articles** – View KBA's that have been read most by others. This list is dynamically compiled based on the number of hits each KBA receives.

Downloads – Download Siebel software products and documentation. You can access documentation at any time, including installation, Tools, Command Center, SDK, and release notes, for any combination of products, versions, categories, and platforms. (Note that online help contains supplemental information about Siebel products and is not accessible from the Customer Central.)

Customer Support – Open a support case with Siebel Technical Support. See “If You Need Help” on Page 29 for details. Real-time case management tools allow you log and track support cases quickly and easily, allowing you to always know the status of your case.

Education and Training – View up-to-the-minute product training course offerings and open enrollment sessions available from Siebel Education & Training Services.

Customer Profile – View up-to-date information Siebel has online about you and your company. Update your email address; change your password, view product history and more.

To access Customer Central:

- Customers can access Customer Central from <https://support.edocs.com> (or the Support link on the edocs.com site).

New users must enroll and register with Customer Central. There is a 2-day turnaround for your registration to be accepted. You will receive an email containing your Customer Central username and password.

3 Job Failure

Multiple Job Failures

If multiple jobs fail at once:

1. Find out whether the database is down. This is the most likely cause.
2. Look for a possible job “collision.” If you are running multiple Indexer or EmailNotification jobs, turn on debug.

We do not recommend scheduling more than one of these jobs to run at one time. It’s a better idea to configure Indexer to run multiple job instances concurrently; see the Command Center help for more information.

When Jobs Hang

If one or more jobs hang (versus fail), and the entire server is not hung, try the following steps to recover:

1. Stop scheduler using:

```
wl_scheduler -stop url t3://audi:7001
```

Where *audi:7001* is the name and port number of your production server. When scheduler stops, the jobs that were running appear as Failed.
2. Shell command tasks launched by Command Center have their own process ID, and therefore may continue to run. You need to kill these processes separately.
3. When scheduler restarts, all jobs set to run in the past run at once. This can cause conflicts. Change the schedule before restarting.
4. Restart scheduler using:

```
wl_scheduler -start url t3://audi:7001
```

When the System Hangs

If the server hangs, follow these steps to help identify the problem:

1. Look at the CPU usage. Are you at %100?
2. Create a series of thread dumps to see what the CPU is working or waiting on. You can capture the thread dump in Unix with:

```
$ kill -3 pid.
```

where *pid* is the Java process ID. This sends the thread dump to std.out. Do four of these in a row, 30 seconds apart, and send the logs to Siebel Technical Support.

Indexer Job Failure

General steps to troubleshoot Indexer job failure

An Indexer job fails when its status is Failed or No operation.

Follow these general steps to gather basic information about the failure and resolve it:

1. View task status to find which task failed; click the status of the job (**status** column) on the Main Console.
2. View the e-Billing Manager Error log to see if e-Billing Manager generated an error or exception. Also see “Interpreting Exceptions” on Page 38. (The error number prefix indicates which task was processing when the error occurred: SCN, IDX, IXL, IAC, etc.)
3. View your server and other system logs to see if any errors were generated to these files.
4. Look for specific solutions:
 - Check the resolutions in this chapter to problems related to the particular task where the Indexer job failed.

For further information, the Siebel Command Center Help or the e-Billing Manager *Administration Guide* describe the function of each task in detail, including input, output, and configuration parameters.
 - Search the Siebel Customer Central Knowledge Base on a specific error, exception, or descriptive text that appears in a log. You can always make a more general search on the Knowledge Base, searching on “Indexer” for example.

5. Retry the failed Indexer job after correcting the problem (unless you're using Payment).

When to retry vs. cancel an Indexer job

When an Indexer job fails it is always safe to retry the job (unless you're using Payment). If the problem is resolved and the job can proceed, Indexer resumes where it left off. For example, if the job failed during IXLoader, the Indexer job continues with that task since it doesn't need to scan the data file again.

You can cancel an indexer job, if necessary.

Scanner task failure

If the Indexer job fails at the Scanner task (status is Failed or No operation), here are some possible causes:

Type of Failure	Probable Causes	What to Do
Task fails (with No Operation status)	Data file missing or not accessible.	Check the input directory for a valid data file.
	Wrong input directory specified.	Check the Indexer job configuration for the specified Input directory and edit the pathname if necessary. If you get error SCN0014 Scanner.process.Task on Solaris, see KB article #Q103007. (Solaris Data directory for the indexer job may be incorrect and mounted under /var/crash or there may be symbolic links associated with it. Remove the symbolic links and create a new data directory under /opt/Siebel/EBM, and ensure it is referenced correctly in the Indexer job.)
	Wrong output directory specified.	Check to make sure all directories exist, have the correct permissions and are mounted. See KB article # Q102770 for "Invalid task config" message in the Syndication Manager.
	Permissions do not allow read access.	Check the directory and data file permissions, correcting permissions if necessary.
	Two servers (Scheduler) are accessing the same database.	Don't run Scheduler on two servers that are both accessing the same database for job configuration information (such as a production application on one server and a test application on another server). Run one Scheduler at a time or replicate the database and file structure.
	Scheduler not running.	Verify whether Scheduler (Process Workflow Controller, PWC) is running and restart it if it is

Type of Failure	Probable Causes	What to Do
		<p>not.</p> <p>Scheduler is a java process; verify that this process is running.</p> <p>Solaris/AIX: Check <code>ps -ef grep xboot</code> (You can check the java processes, but it's tough to tell which java process is the <code>wl_scheduler</code> ... especially on a server with multiple WebLogic servers running.) <code>% cd \$EDX_HOME/bin</code> <code>% ls -la</code> look for a file named <code>.scheduler.localhost:8010.pid</code> where <code><localhost:8010></code> is the <code>-url</code> argument passed when launching the <code>wl_scheduler</code> and either "cat" or "more" this file to reveal the PID for the <code>wl_scheduler</code> process. Then verify if this PID is running via "<code>ps -ef grep <PID></code>"</p> <p>Windows: Check command prompt window or in the list of services (if you ran it as a service). Check the pwc log file for messages. When you start Scheduler, the system creates a log file in <code>EDX_HOME%\Logs</code>.</p> <p>Note that if the Server, Scheduler, and Logger processes are not running, the server may be down.</p> <p>See your e-Billing Manager <i>Installation Guide</i> for details about starting/restarting Scheduler (search Product Docs for "scheduler" in the Customer Central Knowledge Base).</p>
	<p>The Oracle database is down.</p> <p>Oracle Listener is stopped.</p>	<p>Check whether the database is up and restart the <code>edx_db</code> database instance. Stop and start WebLogic and retry the Indexer job.</p> <p>Make sure the Listener process is running.</p>
Task fails (with Failed status)	Out of disk space.	Check disk space and free up if necessary.
	File has 0 bytes.	<p>Check the size of the file data file in the Input directory. If it is 0 bytes, this file is the problem; Indexer will not process zero byte files. Scanner assumes a zero byte file is corrupt and the task/job fails.</p> <p>See KB article #Q102270.</p>

Indexer task failure

If the Indexer job fails at the Indexer task, check for any of the following potential causes:

Type of Failure	Probable Causes	What to Do
Task fails (with Failed status)	Corrupt DDF or data file.	Check against known good files if possible. Replace the corrupted file. To verify the data in an input file can be read, open the file in DefTool and simulate using the application DDF. If the files are being sent to the server in binary mode and you receive error message GE0629 on AIX or Oracle, see KB article #Q102381.
	Permissions do not allow read access.	Check the directory and data file permissions, correcting permissions if necessary. See KB article #Q102667 for information about error IDX0011 on AIX.
	Path to DDF not found.	Check the Indexer task configuration. If the DDF path says “Not found,” go to Publisher and verify that the version set was published properly (with the correct DDF and type). Check that EDX_HOME is properly set in startup files and that EDX.CONFIG is sourced
	Out of disk space.	Check disk space and free up if necessary. See KB article #Q102478 if using WebSphere on AIX. Check swap space using df-K. Add more space if necessary.

IXLoader task (also DXLoader) failure

If the Indexer job fails at the IXLoader task, check for any of the following potential causes:

Type of Failure	Probable Causes	What to Do
Task failed (Failed status). No *.ctl or *.log files were created in the Data directory. It appears as if SQLloader didn't run.	Database communication problem.	A “SQLloader.processTask” error may indicate a problem with the configuration of the database connection. For Solaris, see KB article #Q100202 for more details. Check the Oracle tnsnames.ora file on the application server, and try to manually connect to the database via sqlplus, TOAD, etc. Is Oracle Listener Started on the database

Type of Failure	Probable Causes	What to Do
		server? Check Listener.ora. Is SID correct? Check dbmap, dbaccess.properties (the files are from the 1.9 days – no longer around in 2.x and greater.)
Task failed, a partial .ir file was created, but no *.log or *.ctl files.	Out of memory.	Verify min and max heap sizes for java in \$EDX_HOME/config/edx.config. Also verify min and max heap sizes for java, which are specified as JAVA_OPTIONS in the scripts that launch the WebLogic servers: startWebLogic.sh and startManagedWebLogic.sh. Check swap space using df-K. If swap space is filled add more then retry the job.
Task failed, but *.log and *.ctl files were created.	Bad *.ir file (or *.ix file)	Check *.log and *.ctl for SQLloader error messages.
	Out of disk space.	If you receive an error such as IXL0015, verify that the application server has sufficient swap space and increase as necessary. See KBA #Q100807.
	Pointing to wrong database.	If a valid data file exists and the permissions on the folder allow read access, this is usually a mistake in pointing to the incorrect database. Make sure the JDBC and SQLLoader are both pointing to the same database.

AutoIndexVolAccept task failure

The AutoIndexVolAccept task does not typically fail.

EmailNotification Job Failure

General steps to troubleshoot EmailNotification job failure

An EmailNotification job fails if an email message fails to send to the SMTP mail server (after retrying the job the number of times specified in the “Max Number of Retries” field in the job configuration).

When an EmailNotification job fails, some emails have typically already been successfully sent to the server. But the failed email and subsequent email remain unsent, and remain unsent until you resolve the problem email and rerun the job. Until you cancel a failed EmailNotification job, no other files can be processed by the EmailNotification job. (Any sent mail does NOT get sent again when you retry a job.)

When the problem is resolved, you must retry the job to send the unsent email.

Follow these general steps to troubleshoot EmailNotification job failure:

1. Check the task status to verify which task failed; click the status of the job (**Status** column) on the Main Console.
2. View the Error log to see if e-Billing Manager generated an error or exception. See “Interpreting Exceptions” on Page 38.
3. View your server and other system logs to see if any errors were generated to these files.
4. Check the configuration parameters set for the EmailNotification job in the Siebel Command Center.
5. Look for specific reasons for email failure:
 - **When EmailNotification fails it generally means troubleshooting the failure of individual email. For reasons why email may fail and how to correct it, see the MailNotification Task Failure section below.**
 For further information, the Siebel Command Center Help or the e-Billing Manager *Administration Guide* describe the function of the IVNScanner and MailNotification tasks in detail, including input, output, and configuration parameters.
 - Search the Siebel Customer Central Knowledge Base on a specific error, exception, or descriptive text that appears in a log. You can always make a more general search on the Knowledge Base, searching on “email” for example.
6. Once you’ve corrected the problem, retry the failed EmailNotification job. If it fails again on other problem email, you must resolve the specific problem with that email before you can retry the job. You must resolve each email failure since the rest of the unsent email won’t get sent otherwise.

When to retry vs. cancel an EmailNotification job

If you have corrected the problem with a failed email, connection to the server, or other problem that caused an EmailNotification job to fail, you can always retry the job.

Canceling an EmailNotification job prohibits you from retrying the job, and any unsent messages cannot be sent for that data file.

There is a workaround for restarting a cancelled job, however, it requires you to update the database directly. Knowing how to perform this workaround enables you to cancel the failed job so that you can go ahead and start another mail job while researching and correcting the problem with individual email failures, then later restart the cancelled job so that the unsent email can be sent.

How to update the database to force a retry of a cancelled EmailNotification job:

1. Set the status of the failed email to “address error” in *DDN_mail* table for the accounts and IVN that failed.
2. In Customer Central, go to KB Article #Q102778. This article contains a link to the *eNotificationReset.sql* script, which you must run to reset the IVN for mail notification. This script deletes the row from *ddn_volume_types* where *z_ivn = your ivn* AND *volume_type = 'IVN Scanner email-Email Notification.'* This will allow the IVN to be picked up again next time the email job runs.

Edit the following section of the script to replace “ddn” with the application name, and “&2” with the number of the cancelled IVN you want to rerun:

```
delete
from   ddn_volume_types
where  ddn_vol_num = &2
and    volume_type = 'IVN Scanner email-Email Notification'
```

3. Run the EmailNotification job.

Additional information is available in KB Article #Q102744.

IVNScanner task (for EmailNotification) failure

IVNScanner does not typically fail.

MailNotification task failure

Type of Failure	Probable Causes	What to Do
Task failed (Failed status)	Individual email fails due to a problem with the address or other information.	<p>Find out which email failed and why; look at the contents of the email table, called <i>DDN_mail</i>, which populates when the job runs. (Siebel Technical Support uses Toad for viewing DB content.) The Status column in the mail table indicates which email failed. Status can be:</p> <p>Sent – The SMTP server accepted the email for delivery. It does not necessarily mean that the user has received the email.</p> <p>Unsent – The SMTP server did not accept the email. This often means an exception was thrown. Review your error logs.</p> <p>Server Error – The SMTP server</p>

Type of Failure	Probable Causes	What to Do
		<p>could not be contacted. The network or server may be down.</p> <p>Address Error – There is a problem with the email address on file for the recipient. The domain name may be bad (versus an invalid email format). Check the address for the failed email. If it contains a bad domain name, correct it.</p> <p>Failed – No message from the SMTP server; may not be able to access SMTP server or other failure.</p>
The task fails and the administrator gets no message back	Incorrect SMTP host.	Check the EmailNotification job configuration.
The task fails and generates an error or exception	Network or mail server connections not working or specifications incorrect.	<p>Posting Service Error: PS5002 Email notification not being sent out to customers. Failed to send email.</p> <p>Make sure the mail server is running, check the network connectivity to the mail server, and make sure mail server specifications are correct in the EmailNotification job configuration..</p>
The task appears to run forever.	Retry period is too large.	The default retry period is 60 minutes. Check configuration and adjust the retry period.

Using Mail Auditor

The Mail Auditor is an optional feature you can use to:

- Check the integrity of each email before the EmailNotification job sends it to the SMTP mail server.
- Maintain logs of which email was sent and which email failed the integrity checks for each EmailNotification job. Mail Auditor saves the entire text of the email and other data in logs called *.failed and *.sent in your data output directory (as you define in your Indexer job configuration).

You can use Mail Auditor with HTML or plain text mail.

The Mail Auditor performs the following checks on each email:

1. Ensures that the account number, the email address list, and the message body are not null. If the email address list is null, the account is logged to EDX_HOME/Data/ddn/*.failed with a status of “Skipped.” Here is a sample of the log format:

```
Fri May 24 18:26:55 EDT 2002 Status:Skipped DDN:NatlWireless
Account Number:0407200
```

```
Email Addresses:Message Text:
```

```
THE JOB WILL NOT FAIL FOR NULL EMAIL ADDRESS
```

To speed processing, no email message body is created and no attempt is made to send to the SMTP server for this account. No entry is made to the `ddn_mail` table to allow easier resending of emails by the administrator.

You can rerun the mail notification task for a particular index volume number by removing 'IVN Scanner email-Email Notification' from `ddn_volume_types` where `z_ivn` corresponds to your data file. Then rerun the EmailNotification job.

2. Verifies that there is at least one email address. See above.
3. Verifies that HTML start and end tags are correct for specific strings. If a start of email message text is provided as an environmental entry in the `ejb-jar.xml`, ensures that the message body starts with that text. Default is `<!DOCTYPE` which would be found on HTML email templates.
4. If an end of mail message text is provided as an environmental entry in the `ejb-jar.xml`, ensures that the message body ends with that text. Default is `</html>` which would be found at the end of the HTML email template.
5. If the environmental entry for account number validation is set to true, then there should be at least one occurrence of the account number within the message body.

Mail Auditor logs the following information:

- **If any of validation steps 3-5 above fail**, Mail Auditor logs this information to `*.failed` and fails the job to alert the administrator and allow them to easily determine the problem and then retry the job. It logs the following information:

```
Mon May 06 17:23:28 EDT 2002 Status:Aborted DDN:ddn Account
Number:accountNum Email Addresses:list of emailAddresses
separated by a semicolon Message Text:Message Body
```

- **If the email passes the validation steps listed above and the MailNotification job sends all email successfully**, Mail Auditor logs this information to `*.sent` with this information:

```
Log timestamp in the format Mon May 06 17:23:28 EDT 2002
Status:Sent DDN:ddn Account Number:accountNum Email
Addresses:list of emailAddresses separated by a semicolon
Message Text:Message Body
```

- **If the email passes all validation steps listed above and the email is unsuccessfully sent due to some exception**, Mail Auditor logs the following information to `*.failed`:

```
Log timestamp in the format Mon May 06 17:23:28 EDT 2002
Status:Failed DDN:ddn Account Number:accountNum Email
Addresses:list of emailAddresses separated by a semicolon
Message Text:Message Body
```

To troubleshoot a failed EmailNotification job using Mail Auditor log files:

1. When an EmailNotification job fails, check the Data directory for the ddn and look for the *.failed files. Open this file and determine why the email failed from the logged data.
2. Correct the problem email.
3. Retry the email job in Command Center. Previously sent emails will NOT be resent.
4. Mail Auditor appends successful emails during the retry to the *.sent file.

To manage log files:

- The administrator can safely delete the sent and failed logs at any time to reclaim the drive space.

To turn Mail Auditor on/off:

You can configure each email job to use a custom Mail Auditor bean. Several beans can be in the Enterprise Archive file. All email jobs that use the same bean will use the validation parameters contained in the deployment descriptor for that bean. You can change the deployment descriptors from the WebLogic or WebSphere console.

- In the Siebel Command Center, edit the EmailNotification job configuration (MailNotification task). To turn Mail Auditor on, specify the appropriate JNDI name of your system's custom Mail Auditor bean in the Auditor Model configuration field. To turn Mail Auditor off, leave the Auditor Model field blank.

**Caution**

Note that running Mail Auditor slows down EmailNotification job performance.

To change the parameters Mail Auditor looks for at the beginning and end of an email:

Edit the deployment descriptor for your MailAuditor bean. The environment entries in the ejb-jar.xml looks like:

```
<env-entry>
  <description>Check For Account Number</description>
  <env-entry-name>CheckForAccountNumber</env-entry-name>
  <env-entry-type>java.lang.Boolean</env-entry-type>
  <env-entry-value>true</env-entry-value>
</env-entry>
<env-entry>
  <description>Email Message Body Start</description>
  <env-entry-name>EmailStartText</env-entry-name>
  <env-entry-type>java.lang.String</env-entry-type>
  <env-entry-value><![CDATA[<!--DOCTYPE-->]]></env-entry-value>
</env-entry>
<env-entry>
  <description>Email Message Body end</description>
  <env-entry-name>EmailEndText</env-entry-name>
  <env-entry-type>java.lang.String</env-entry-type>
  <env-entry-value><![CDATA[</HTML>]]></env-entry-value>
</env-entry>
```

Where text in bold are the environmental entry names and cannot be changed and text in italics are the values for the environment entries that can be configured. Note that some values are enclosed in `<![CDATA[...value...]]>` because of the special characters like ‘!’, ‘<’ and ‘>’.

These entries are optional; if they do not occur in the deployment descriptor, then Mail Auditor does not perform the associated check.

4 DDF Validation Errors

Fixing Invalid DDF Content

If you receive errors opening or saving a DDF in DefTool or while trying to publish a DDF in Publisher, your DDF contains invalid content. DefTool also validates XML DDF content when you open or save an XML DDF. When you attempt to save a serialized DDF in e-Billing Manager, DefTool performs additional validations on the XML, sometimes revealing problems that previously went undetected.

DefTool displays the message “Failed during internal validation” when the schema validation fails and logs the error.

When the internal validation fails in Publisher, the message “Unable to create version set due to an invalid DDF” appears but the error is not logged.

To resolve DDF errors:

1. If the error appeared in DefTool, check your DDF error log *DDF_error.log*, where DDF is the application name, in the folder where the DDF is located for details. If you received an internal validation error in Publisher, open the DDF in DefTool and try to determine where the problem content is.
2. See the table below for resolutions to some common DDF errors. Although you can fix most problems manually, a few require you to contact Siebel Technical Support for resolution.
3. If you have saved the DDF in XML format, follow the resolution procedure for *XML* format. If you have installed 4.0 or higher but have not yet saved the serialized DDF in XML format, follow the instructions for the *serialized* DDF.

DDF Error	Resolution for XML or Serialized DDF
<Field name> field is not associated to a table in the DDF	<p><i>Serialized:</i></p> <ol style="list-style-type: none"> 1. Open the DDF in DefTool. DefTool automatically removes the field when you open the DDF. 2. Save the DDF. <p><i>XML:</i></p> <ol style="list-style-type: none"> 1. Open the XML DDF in a text editor. 2. Search for the unassociated field name. 3. Remove all the element information. 4. Save the DDF in the text editor. 5. Open the DDF in DefTool and save the file (to verify and stamp the integrity key into the DDF). <p>Repeat this process for all unassociated elements; DefTool shows only one error at a time.</p>
Error in reading field location for <field name>	<i>Serialized or XML:</i> Contact Technical Support.
<p>Invalid page style reference <pg style name> for a pure/group field location</p> <p>Invalid page style reference <pg style name> for a marker location</p>	<p><i>Serialized:</i></p> <ol style="list-style-type: none"> 1. In DefTool, remove the invalid page style reference from the pure/group field element or marker location and add the valid page style reference. 2. Save the DDF. <p><i>XML:</i></p> <ol style="list-style-type: none"> 1. Open the XML DDF using a text editor. 2. Remove the invalid page style reference value from 'page-type' attribute in the corresponding field (pure/group) or marker location and add the valid page style. 3. Save the DDF. 4. Open the DDF in DefTool and save the file (to verify and stamp the integrity key into the DDF).
Invalid 'type' value for field <field name>	<p><i>Serialized:</i></p> <ol style="list-style-type: none"> 1. In DefTool, edit the field properties and add the correct field type. 2. Save the DDF. <p><i>XML:</i></p> <ol style="list-style-type: none"> 1. Open the XML DDF using a text editor. 2. Add the correct field type value (STRING/NUMERIC/CURRENCY) in the 'data-type' attribute in the field element. 3. Save the DDF. 4. Open the DDF in DefTool and save the file (to verify and stamp the integrity key into the DDF).
Error in reading	<i>Serialized or XML:</i> Contact Technical Support.

DDF Error	Resolution for XML or Serialized DDF
locations in the marker <marker name>	
Contains an invalid dynamic field <field name> in the pattern	<p><i>Serialized:</i></p> <p>Remove the invalid dynamic field reference:</p> <ol style="list-style-type: none"> 1. In DefTool, edit the element properties and delete dynamic field reference. 2. If the 'Use Dynamic Pattern' option is set, unset it. 3. Save the DDF. <p><i>XML:</i></p> <ol style="list-style-type: none"> 1. Open the XML DDF in a text editor. 2. Search for the element with the invalid dynamic reference. 3. Set the value of 'dynamic-pattern' attribute to "false". 4. Open the DDF in DefTool and remove the dynamic field reference as described for the serialized DDF. 5. Save the DDF.
Pattern does not contain dynamic field	<p><i>Serialized:</i></p> <ol style="list-style-type: none"> 1. In DefTool, edit the element properties and uncheck the 'Use Dynamic Pattern' option. 2. Save the DDF. <p><i>XML:</i></p> <ol style="list-style-type: none"> 1. Open the XML DDF in a text editor. 2. Set the value of 'dynamic-pattern' attribute to "false" for the element. 3. Save the DDF. 4. Open the DDF in DefTool and save the file (to verify and stamp the integrity key into the DDF).
Page Style Name <pg style name> is invalid: Page Style, Group, Table, Marker, Column and Field names should contain alpha characters, "_" or numbers without spaces and should start with an alpha character.	<p><i>Serialized:</i></p> <ol style="list-style-type: none"> 1. In DefTool, remove the special characters. 2. Save the DDF. <p><i>XML:</i></p> <ol style="list-style-type: none"> 1. Open the XML DDF using a text editor 2. Remove the special characters. 3. Save the DDF. 4. Open the DDF in DefTool and save the file (to verify and stamp the integrity key into the DDF).
Element 'child' is not valid for content model: '(table-anchor,child*)'	<i>Serialized or XML:</i> Contact Technical Support.
Error in reading location information for the table field <tbl field name>	<i>Serialized or XML:</i> Contact Technical Support.

DDF Error	Resolution for XML or Serialized DDF
Value <code><string></code> does not match regular expression facet <code>'[^]+([]*[^]+)*'</code> - (schema error)	<p><i>Serialized:</i></p> <ol style="list-style-type: none"> 1. In DefTool, remove leading and trailing spaces of the element. 2. Save the DDF. <p><i>XML:</i></p> <ol style="list-style-type: none"> 1. Open the XML DDF in a text editor. 2. Remove leading and trailing spaces of element. 3. Save the DDF. 4. Open the DDF in DefTool and save the file (to verify and stamp the integrity key into the DDF).
Group <code><group name></code> has a invalid child table <code><table name></code>	<p><i>Serialized:</i></p> <ol style="list-style-type: none"> 1. In DefTool, delete the invalid table references from the group. If you cannot see it, contact Technical Support. 2. Save the DDF. <p><i>XML:</i></p> <ol style="list-style-type: none"> 1. Open the XML DDF in a text editor. 2. Remove the invalid child information from the group. 3. Save the DDF. 4. Open the DDF in DefTool and save the file (to verify and stamp the integrity key into the DDF).
Unable to read the DDF file	<p><i>Serialized or XML:</i></p> <ul style="list-style-type: none"> • Grant write permission for the DDF. If this does not work, contact Technical Support.
Multiple PRIMARY Keys are defined in the DDF	<p><i>Serialized:</i></p> <ol style="list-style-type: none"> 1. In DefTool, remove additional primary key fields or edit the field and unselect the field type "Primary Key" (to make it a regular field). 2. Save the DDF. <p><i>XML:</i></p> <ol style="list-style-type: none"> 1. Open the XML DDF using a text editor. 2. Delete the additional primary key fields or change the value of 'type' attribute in field tag to "SECONDARY". 3. Save the DDF. 4. Open the DDF in DefTool and save the file (to verify and stamp the integrity key into the DDF).
Field location contains an empty pattern	<i>Serialized or XML:</i> Contact Technical Support.

5

Improving Application Performance

Summary List Response Time

The summary list, sometimes called the “hit list,” is the application view that shows indexed data for the current statement plus any historical ones available.

To determine whether your summary list response time is a problem:

1. Using a Web monitoring tool, evaluate the .jsp response times.
2. Put timestamps around the call to getSummary to see how long the call took in elapsed time.

If the hit list getSummary response time is the primary cause of delay:

1. Narrow the range of months you return, for example, display 3 months instead of 12. This reduces the number of tables that need to be searched. You could add a link if you still want users to optionally be able to access all 12 months of historical statements.
2. Improve the speed of the Index table. On DB2, run Runstats to improve the performance of database queries on the indexed data; on Oracle use Compute Statistics. You can also modify table sizes, and for Oracle, enabling native table partitioning.

Indexer Job Performance

If an application’s Indexer job is slow, here are some ways to improve performance:

- Look at the Command Center logs and determine how many minutes the Indexer and IXLoader tasks take. (Scanner and AutoIndexVolAccept tasks are not going to be the problem.) If the Indexer task is slow:
 - The Indexing DDF may not be optimized for extraction. Try making long, narrow coordinates in DefTool and simplify RegEx patterns.
 - Compression on IndexerTask might be slow.

If the IXLoader task is slow:

- Review the load method you are using. Direct load is usually faster than Conventional load. Direct load stores data directly to the database and locks the Index table. A Conventional load uses Insert statements, one row at a time, performs multiple Selects and Inserts on the table at once, but does not lock up the table. Siebel recommends using Direct load for Oracle and SQL Server, and Conventional load for DB2. See the e-Billing Manager *Administration Guide* for details on the IXLoader task configuration options.
- There is an approximate 20-second overhead for processing individual data files. Using many small files can significantly impact the Indexer job. If possible, use fewer input files.

EmailNotification Job Performance

The Siebel email solution platform can scale to move more than one million emails/day. We are capable of 100,000/hr. If you get drastically slower performance, we have found that the majority of problems can be fixed by plug-in or by optimizing the SMTP server.

If an application's EmailNotification job is slow, here are some ways to improve performance:

- Review the response time of your email account resolver. Check how long it takes per lookup. Add elapsed time logging to your plug in so you know the moment it came in to the moment it returned the email address (the start and end of the method).
- Your SMTP server may be slow. To test, add multiple IP addresses for SMTP servers, comma-separated, to the "SMTP Hosts" field of your EmailNotification job configuration. This randomly selects SMTP servers to send email to. If the problem was your SMTP server, you should see a dramatic improvement in response time. Optimize or tune up the SMTP server if it is slow.