

rvsXP

Version 4.00

User Manual

The products listed in this manual are protected by copyright.

rvsXP
Version 4.01
User Manual

© 2005 by gedas deutschland GmbH
Pascalstraße 11
10587 Berlin

This manual is protected by copyright. All rights reserved. No part of this book may be used or reproduced in any form or by any means including photocopies, microfilm or any other means or stored in a database or retrieval system without obtaining prior permission from gedas. Rights are also reserved as far as lectures, radio and television is concerned.

We reserve the right to make changes to the content of this manual without giving prior notice. gedas is not liable for technical or printing errors or defects in this manual. Moreover, gedas shall not be liable for damage which is directly or indirectly caused by delivery, performance and use of this material.

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 7 |
| 1.1 | Short description of the system | 7 |
| 1.2 | History | 7 |
| 1.3 | Encryption and offline compression | 10 |
| 1.4 | Target group | 11 |
| 1.5 | Structure of the rvs® documentation | 12 |
| 1.6 | Representation means | 13 |
| 2 | Installation of rvsXP | 15 |
| 2.1 | System requirements | 15 |
| 2.2 | Connection type installation | 15 |
| 2.3 | Installing rvsXP | 17 |
| 2.4 | The rvsXP program environment | 21 |
| 2.5 | How to obtain a license | 22 |
| 3 | Function Tests | 25 |
| 3.1 | Sending a file to yourself | 25 |
| 3.2 | Testing the connection with the local and the partner stations | 27 |
| 4 | Configuration of rvsXP | 31 |
| 4.1 | Setting up rvsXP stations | 32 |
| 4.1.2 | Setting up ODETTE parameters | 37 |
| 4.1.3 | Setting up the neighboring station | 42 |
| 4.1.4 | Setting up the routing connection | 45 |
| 4.1.5 | Setting up TCP/IP communication | 47 |
| 4.1.6 | Setting up the ISDN parameters | 49 |
| 4.1.7 | Setting up X.25 parameters | 53 |
| 4.1.8 | Setting up a virtual station | 57 |
| 4.1.9 | Editing the partner station | 58 |
| 4.2 | TCP/IP reception | 60 |
| 5 | rvsXP Monitor | 63 |
| 5.1 | Starting rvsXP Monitor | 63 |
| 5.3 | Customizing rvsXP Monitor parameters | 66 |
| 6 | How to work interactively with rvsXP | 69 |
| 6.1 | Starting rvsXP-Administrator (rvsmgr.exe) | 69 |
| 6.2 | Starting rvsXP (rvsdiant.exe) | 70 |
| 6.3 | How to send and receive files with rvsXP | 73 |
| 6.3.1 | Creating send orders | 73 |
| 6.3.2 | Serialization | 77 |
| 6.3.3 | Displaying send requests and received files | 77 |
| 6.3.4 | Handling files to be sent and received | 80 |
| 6.3.5 | Selecting display criteria | 82 |
| 6.4 | Administration with rvsXP | 84 |
| 6.4.1 | Resident receive entries | 84 |
| 6.4.2 | Job start after send attempt | 88 |

| | | |
|-----------|---|------------|
| 6.4.3 | User administration | 91 |
| 7 | rvsXP database maintenance | 95 |
| 7.1 | Back-up: backing up rvsXP tables | 96 |
| 7.1.1 | Backing up the station table | 96 |
| 7.1.2 | Saving the user table and follow-on processing | 97 |
| 7.2 | Backing up: Backing up the rvsXP database | 97 |
| 7.3 | Delete and reinitialize rvsXP database | 98 |
| 7.4 | Restoring rvsXP data from a back-up | 100 |
| 7.4.1 | Restoring the station table | 100 |
| 7.4.2 | Restoring the user table and the follow-on processing | 100 |
| 7.4.3 | Restoring the rvsXP database | 101 |
| 7.5 | Cleaning up the rvsXP database (cleanup) | 101 |
| 8 | Encryption configuration: Key administration | 103 |
| 8.1 | Encryption basics | 103 |
| 8.1.1 | Encrypted transmission with rvsXP | 103 |
| 8.1.2 | rvsXP encryption principle and procedure | 104 |
| 8.2 | Overview of the rvsXP key administration | 107 |
| 8.3 | Creating private and public keys | 107 |
| 8.4 | Importing Keys (rvskeyimp) | 111 |
| 8.5 | Distribution of keys (rvskeydst) | 113 |
| 8.6 | Deleting imported keys (rvskeydel) | 114 |
| 8.8 | Configuration of offline compression | 117 |
| 9 | Code conversion | 119 |
| 9.1 | Automatic code conversion with integrated conversion tables | 120 |
| 9.2 | Code conversion with user code tables | 121 |
| 9.2.1 | Structure of the code conversion tables | 121 |
| 9.3 | How to perform a code conversion | 122 |
| 9.3.1 | Code conversion when sending files | 122 |
| 9.3.2 | Code conversion when receiving files | 124 |
| 10 | Backing Up and Recovering rvsXP Data | 127 |
| 10.1 | New parameters | 127 |
| 10.2 | Procedure | 127 |
| 10.3 | rvsredo Options | 129 |
| 11 | rvsXP HighPerformance | 131 |
| 11.1 | Configuration of ORACLE | 131 |
| 11.2 | Configuration of rvsXP | 133 |
| 11.3 | Microsoft SQL Server configuration | 134 |

Appendix

- Appendix 1: The rvs[®] parameters
- Appendix 2: Log files
- Appendix 3: Environment variables
- Appendix 4: Glossary

Index

Change History

The following changes of User Manual were made in the previous releases (including the current release):

Version 4.0:

- size limit of files to be transferred is now 10 GB.
- new parameters in the OP table: OFTP Version and User Separation (See chapter 4.1.2 "Setting up ODETTE parameters").
- new supported platform HP Itanium (See chapter 1.2 "History").
- new feature: Backup and Recovery (See chapter 10 "Backing Up and Recovering rvsXP Data").

1 Introduction

The present chapter gives a brief introduction to the rvs[®] system, describes the structure of the rvs[®] manuals and defines the target group. It explains the typographic conventions used in the present manual.

1.1 Short description of the system

What is rvs[®]?

rvs = Rechner-Verbund-System

The abbreviation rvs[®] stands for the German word Rechner-Verbund-System. The rvs[®] computer communication system is a well established base service for electronic data interchange, EDI.

rvs[®] serves to ensure transmission of electronic data between heterogeneous computer platforms using different network protocols. To do so, rvs[®] implements a universal network model, which you can configure in each network node.

rvs[®] provides an efficient and reliable transport service for both standardized EDI message types and files of any format or contents. You can receive only such files that are explicitly destined for rvs[®]. This means that rvs[®] does not allow any unauthorized access to remote or to own data files.

1.2 History

The system was originally developed by Volkswagen AG and has been used in the German and European automobile industries for a number of years but also by banks, insurances and industry worldwide.

rvs[®] uses the OFTP protocol. An extension to the OFTP standard was developed for Volkswagen AG: It has been enhanced by a line driver for SNA LU 6.2.

This "portable" version of rvs[®] has been developed in order to complement the classical rvs[®] product line, which is based on MVS and VSE mainframes, for use on midrange, mini systems and personal

computing. Though the design and make of this product line differs considerably from the mainframe versions, the functional spectrum is almost exactly the same as that of rvs[®] MVS.

Supported platforms

The following versions of portable rvs[®] are currently available:

Platforms
(continued)

- rvsX for SINIX using LU 6.2, X.25, ISDN and TCP/IP
- rvsX for AIX using LU 6.2, X.25, ISDN and TCP/IP
- rvsX for HP-UX using LU 6.2, X.25, ISDN and TCP/IP
- rvsX for HP Itanium using TCP/IP, ISDN and X.25
- rvsX for IRIX using ISDN and TCP/IP
- rvsX for Linux using ISDN and TCP/IP
- rvsX (Linux/zSeries) using ISDN and TCP/IP from rvsX 2.06 onwards
- rvsX for Solaris using ISDN and TCP/IP
- rvs400 for IBM OS/400 systems using LU 6.2, X.25, ISDN and TCP/IP
- rvsNT for MS Windows NT / 2000 using LU 6.2, X.25, ISDN and TCP/IP
- rvsXP for MS Windows 2000 / XP and WS 2003 using X.25, ISDN and TCP/IP.

What rvs[®] is not

rvs[®] is not an online system. It neither supports direct terminal-like access to other sites, nor does it provide a communication pipe from application to application on a data record level. You cannot use rvs[®] to directly execute transfers in your own application. You rather can place send orders from within you application to rvs[®] which will be handled asynchronously.

rvs[®] is not a job scheduling system.

rvs[®] does not care about the contents of the files it is transporting. It only acts as a transparent transport medium and performs no semantic interpretation of the data it carries.

rvs[®] is not an EDI converter. You can, however, purchase additional components for converting between specific message formats (e.g. VDA, ODETTE, EDIFACT, XML) using rvs[®] as transport service via gedas deutschland GmbH.

rvs[®] is not a network control or monitoring tool.

Basic rvs[®] functions

Reliable
transmission,
cross-platform

The main function of rvs[®] is the reliable transfer of files. rvs[®] is suitable for transferring both large files and many small files. rvs[®] supports communication between different networks with many stations based on different platforms. In other word: rvs[®] supports various data formats.

The key features of rvs[®] are:

-
- Basic functions
(continued)
- The Monitor, the central rvs[®] component, controls all work to be done. rvs[®] works asynchronously: You place a send order with rvs[®]. rvs[®] executes this send order as fast as possible. Benefit: You or your application need not wait for the transmission to terminate.
 - The connection to the communication partner is automatically established when necessary.
 - The automatic submission of jobs after reception of a file is controlled by means of resident receive entries (follow-on processing). Usage of file name, user IDs, and station IDs is supported.
 - The file transfer is possible for the following record formats:
 - U unstructured
 - T text,
 - F fixed record length,
 - V variable record length.
 - Extensive security and authorization checking is performed.
 - rvs[®] automatically repeats the transmission after an unsuccessful connection attempt or disruption of a connection. rvs[®] only transmits the parts of a file not yet transmitted when a disrupted transmission is resumed. Therefore it is suited particularly to transmit large quantities of data even under difficult conditions. The compression of data during transfer increases transmission line throughput.
 - Online or offline compression more effectively uses the transmission line capacity.
 - The rvs[®] encryption ensures data security during network data transmission.
 - The graphical user interface features menu control.
 - Besides a graphical, menu controlled interface the rvs[®] batch interface (rvsbat) is provided. That means that you can call rvs[®] from within batch or command files.
 - With a C-Call interface application programs can call functions of the C language to directly generate rvs[®] command entries.
 - The ODETTE File Transfer Protocol, OFTP, is used.
 - The ISDN, X.25, TCP/IP and SNA LU6.2 line protocols communication are supported; however, not all system environments support the complete range of protocols.
 - Message LOG files are provided for revision purposes.
 - rvs[®] supports multiple languages in messages, operator console and user interfaces (presently English and German).
 - The data conversion, if wanted, can be carried out. Text files are automatically converted to the code (EBCDIC or ASCII) used on the target partner system.

- The earliest date/time of a transmission can be defined for each file to be transferred.
- A serialization facility allows the transfer of files in a sequence which you can define.
- rvs[®] automatically generates receipt acknowledgments (EERP as per the ODETTE standard) and supports partner-related control mechanisms during acknowledgment transmission. The user can choose to have the acknowledgments immediately sent by default, when suitable, or after manual intervention only; he can also delete individual receipts without sending them.
- rvs[®] supports the integration of applications and users as “virtual stations”. These do not exist as real stations but as a representation can receive or send data.

1.3 Encryption and offline compression

Encryption even from V2.05 onwards

rvs[®] provides powerful, customizable functions that add useful features to the rvs[®] data transmission solution:

- rvs[®] is able to encrypt and decrypt files in order to ensure privacy of the data when being transmitted over a network. This is especially useful for none-private wide area networks, e.g. when using the internet.
- Data may be compressed offline before network connection is established, in this way network costs are significantly reduced.
- All formats and character sets are supported.
- You can also configure compulsory encryption.

Both – encryption and offline compression – work automatically after having been configured once on a station by station base. The described features work directly between sender and receiver. Routing stations conveniently forward the encrypted and compressed files.

Sender and receiver use rvs[®]

Compression and encryption are available if both partner stations (sender and receiver) use rvs[®]. OFTP routing is not affected, i.e. encryption does not affect the routing stations on the way.

On MVS systems the encryption and compression subsystem enables extended file format information which prevents loss of file format information (e.g. block size, blocked, ASA).

Product independent of rvs[®]

Com-Secure is an independent product for offline encryption and compression developed by rvs[®] that also runs with other OFTP products (other than rvs[®]).

Encryption features

Symmetric and asymmetric encryption The security of rvs[®] uses both symmetric (3DES) and public-/private-key encryption (RSA). Symmetric encryption uses the same key for both encryption and decryption. The public-/private-key manner uses two different keys each for encryption and decryption.

Symmetric encryption is stronger from a security standpoint, and the public/private-key encryption and decryption are more expensive than the corresponding operations in symmetric systems. The primary problem with symmetric encryption is not their security but with key exchange and the number of the keys. With a private and public key pair it is possible to distribute the public key by using insecure communication channels and use one key with different partner stations.

New key for each session rvs[®] performs symmetric encryption on the files to be sent. The actual file being sent is then encrypted using the session key and sent to the recipient. The sender sends this session key, secured by public/private encryption, to the receiver prior to file transmission. It is possible to create and use private and public keys with a size ranging from 768 to 2048 bit.

Digital signature Furthermore rvs[®] provides a digital document signature protecting the transmitted file against changes and verifying the ownership of the transmitted data. The implementation of this feature uses a hash function (SHA1). A hash function is a many-to-one function that maps its input to a value in a finite set. This value is encrypted with the private key of the sender to guarantee the identity of the file owner.

Offline compression features

Algorithm: GNU zip Compression is performed using the GNU zip algorithms¹. rvs[®] applies a compression compatible with gzip (rfc. 1952). The compression method reduces the data size in the best case down to a level of 30% and essentially never expands the data.

Only very small files grow in size due to the process information added. The compression is independent of the CPU type, operating system and character set and performs an integrity check of the decompressed data.

1.4 Target group

For rvsXP users and administrators This manual is meant for regular users of rvsXP as well as administrators. It provides an overview of the basic rvsXP functions and also background information on rvs[®].

The rvsXP concept generally does not require an administrator. There are, however, tasks that cannot do without manual intervention. There is, for example, the very important task of maintaining the database. This requires regular cleanups and backups.

¹ 1995-1998 Jean-Loup Gailly and Mark Adler; for more information see Glossary

To use rvsXP the following skills are required:

- good knowledge of the current operating system
- knowledge of the communications techniques in use
X.25 native communications and/or ISDN native communications
TCP/IP.

Before starting to work with rvsXP it is advisable to have read this book.

1.5 Structure of the rvs[®] documentation

The rvs[®] documentation consists of the following manuals:

| | |
|----------------------------------|---|
| User Manual | The User Manual contains all important steps about installing and configuring rvs [®] . It shows you how to transfer files and handle the rvs [®] database and describes integrated features such as rvs [®] encryption, off-line compression, and code conversion. For this purpose, the rvs [®] administrator has unrestricted access to all functions. The rvs [®] user has access to all functions dealing with the transmission and reception of files. |
| Reference Manual | The Reference Manual is the common manual for rvs [®] portable (rvsX, rvsNT, rvsXP, rvs400). It contains descriptions of the C-Cal interface (<code>rvscal</code>), the rvs [®] batch interface (<code>rvsbat</code>) and of the rvs [®] utilities (mostly for rvsX, rvsNT, and rvsXP). Furthermore, this manual contains information on the technical background of rvs [®] . |
| Messages and Return Codes Manual | This manual describes all messages and error codes which could be displayed on the rvs [®] Monitor and in the log file. |
| Media | The User Manual and the Reference Manual are available as printed and as electronic documents (PDF). The Messages and Return Codes Manual is only available as electronic document. |

1.6 Representation means

This chapter contains the description of the typographic conventions which are used in this manual and the explanation of the expressions which are marked.

Typographic conventions

- Instructions begin with a bullet.
- Other lists begin with the en dash.

| | | |
|------------------|------------------------------------|--|
| Character styles | <code>Courier</code> | Commands, menu commands, file names, path names, programs, examples, scripts, options, qualifiers, data sets, fields, modes, window names, dialog boxes and statuses |
| | BOLD and IN CAPITAL LETTERS | Parameters, environment variables, variables |
| | “Inverted commas” | Links to other manuals, sections and chapters, literature |
| | Bold | Important terms, names of operating systems, proper names, buttons, function keys. |

Terms and variables

| | |
|---------------|--|
| Program names | <p><code>rvsXP</code> is the synonym of <code>rvs</code>[®] for Windows XP systems.</p> <p><code>rvsNT</code> is the synonym of <code>rvs</code>[®] for Windows NT systems.</p> <p><code>rvsX</code> is the synonym of <code>rvs</code>[®] for UNIX systems.</p> <p><code>rvs400</code> is the synonym of <code>rvs</code>[®] for OS/400 systems.</p> <p>Here the “naming” operating system is mentioned only.</p> |
| Directories | <p>As user directories are found on different locations for the different operating systems and respective hardware/software environments we use the \$RVSPATH variable in this manual. Default values are:</p> <ul style="list-style-type: none"> – <code>C:\rvs</code> for Windows XP, 2000, WS 2003 and NT – <code>/home/rvs/</code> for AIX, Solaris, IRIX, Linux, and SCO – <code>/users/rvs/</code> for HP-UX – <code>/defpath/rvs/</code> for SINIX <p>Substitute the variable with your correct path.</p> |

2 Installation of rvsXP

This chapter describes the system requirements, installation of ISDN, X.25 and TCP/IP protocols and of rvsXP.

2.1 System requirements

Hardware and software

You need the following equipment in order to operate rvsXP:

- PC (i386 platform), 300 MHz clock rate, at least 128 MB RAM and 1.5 GB of free hard disk space
- Microsoft Windows 2000, WS 2003, XP Professional or later
- Communication line based on ISDN, TCP/IP, X.25
- Hard disk storage for data transmission.
Recommended value: at least twice the size of the files to be transmitted.

Internal communication is based on TCP/IP. For this purpose the TCP/IP protocol must be installed. Instead of a network adapter the MS Loopback Adapter can be used.

2.2 Connection type installation

ISDN connection

If you want to use rvsXP to exchange data via the **ISDN network** you need the following equipment:

- ISDN type telephone connection with s_0 bus having at least two data channels (B-channel) and one control channel (D-channel)
- ISDN adapter
- CAPI 2.0 driver software for operation of the ISDN card under Windows XP.

This is how you install the ISDN interface:

- Install the ISDN card into your computer and attach the ISDN card to the ISDN connection.
- Make sure that the ISDN card works properly.

Note: Many card manufacturers supply suitable software for a self-test, such as a call from one data channel to another. Restart your computer after installation and the test in order to make the CAPI 2.0 driver available for other applications.

External ISDN router

If you use an external ISDN router (e.g. BinTec Brick), you do not need an internal ISDN card. In this configuration, several applications can share the same Brick router. The Brick router supports the "Remote CAPI" interface. This means that every computer in your LAN uses the Brick router as if it were a local ISDN card in the computer.

Note: gedas deutschland GmbH has successfully tested the following ISDN devices for use with rvsXP:

| Device | Manufacturer | Remarks |
|-------------------------------------|---|-----------------------|
| EICON DIVA Server BRI-2M | Eicon, http://www.eicon.de | |
| EICON DIVA Pro 2.0 | Eicon, http://www.eicon.de | |
| EICON DIVA 2.0 | Eicon, http://www.eicon.de | |
| AVM B1 | AVM, http://www.avm.de | not suitable for X.31 |
| AVM Fritz Card | AVM, http://www.avm.de | only Classic and PCI |
| BIANCA/BRICK-XS, -XL, -XM and X4100 | BinTec, http://www.bintec.de | with remote CAPI |

rvs[®] Support will provide you with the current list of tested devices (phone: +49 30 / 399 71 777, telefax: +49 30 / 399 71 994, email: rvs-service@gedas.de).

X.25 connection If you want to use rvsXP to exchange data via an **X.25 type network** you need the following equipment:

- X.25 network connection
- Interface hardware for interfacing the X.25 network in your computer.
- Driver software for operation of the X.25 card under Windows XP.

If you already have this equipment then you install the X.25 interface as follows:

- Install the interface card in your computer and attach it to the X.25 network connection according to the installation instructions that came with the X.25 card.
- Make sure that the X.25 connection works properly. To do so use the X.25 diagnostic or operating software provided by the card manufacturer.

Note: gedas deutschland GmbH has successfully tested the following X.25 cards for use with rvsXP: EiconCard - EICON Technology Corporation; cards manufactured by TECCOM Ltda, Rua da Assembleia, 10Gr. Centro - Rio de Janeiro - RJ, Brasil Connectionto.

TCP/IP connection The **TCP/IP protocol** is required for exchanging data using rvsXP and also for internal rvs[®] communication. For the data exchange you have to do the following:

- Install the network hardware with which you want to connect your computer to the network. For more detailed information consult the documentation provided by your hardware supplier.

- Display Network Connections:
Start -> Control Panel -> Network Connections.

The right window section displays existing dial-in and network connections.

- Click the right mouse button on the desired connection and choose *Properties*.
- Choose the *Network* tab when you have chosen a dial-in connection.

Windows XP automatically installs the TCP/IP protocol. If the TCP/IP protocol is not contained in the list shown:

- Click the **Install** button.
Highlight the TCP/IP protocol in the window displayed and confirm your selection with **OK**.
- Highlight the TCP/IP protocol and click the **Properties** button.
- Set the IP addresses for your environment and confirm with the **OK** button.
Note: The network administrator will provide you with the IP addresses.
- Confirm the entries by pressing the **OK** button and close the *Network connections* window.

2.3 Installing rvsXP

If an installation is an update, please make a backup of your old rvsXP data. Make a copy of your licence key (file *rdkey.dat*), of your stations, users and Jobstarts (See chapter 7 "rvsXP database maintenance").

- Start installation
- Start Windows XP and log in as an XP user with administrator rights.
 - Insert the data medium labeled **rvsXP** in a drive.
 - Skip the next three instructions if the rvsXP Installer starts automatically.
 - Choose Start -> Run.
 - Enter *D:\setup.exe* to run the installation program (if your CD ROM drive is drive D:).
 - Confirm the entry by pressing **OK**.

The rvsXP installer now starts and guides you through the installation.

- Follow the installer instructions. You can confirm the *C:\rvs* target folder or select another.
Please note that file and path names must not contain any spaces.
- You may choose between German or English as user languages.

- We recommend LOC as the station ID for the local station. However, you may choose a different ID.

Important: The installer automatically enters the current XP user as rvsXP user with rvsXP administrator privileges in the rvsXP database. Heed the appropriate dialog during installation if you wish to use another user account for the rvsXP service.

The installation program creates the `rvs` folder in the All Programs start menu folder along with the required links (icons) to rvsXP applications.

Tip: Read the `Readme.txt` file before you start the rvsXP-Administrator.

- Start the rvsXP-Administrator.

After your acknowledgement the rvsXP-Administrator generates the rvs database. This database is required to operate rvsXP. The program `Init Database` initializes this database and saves it in the database folder (`C:\rvs\db`).

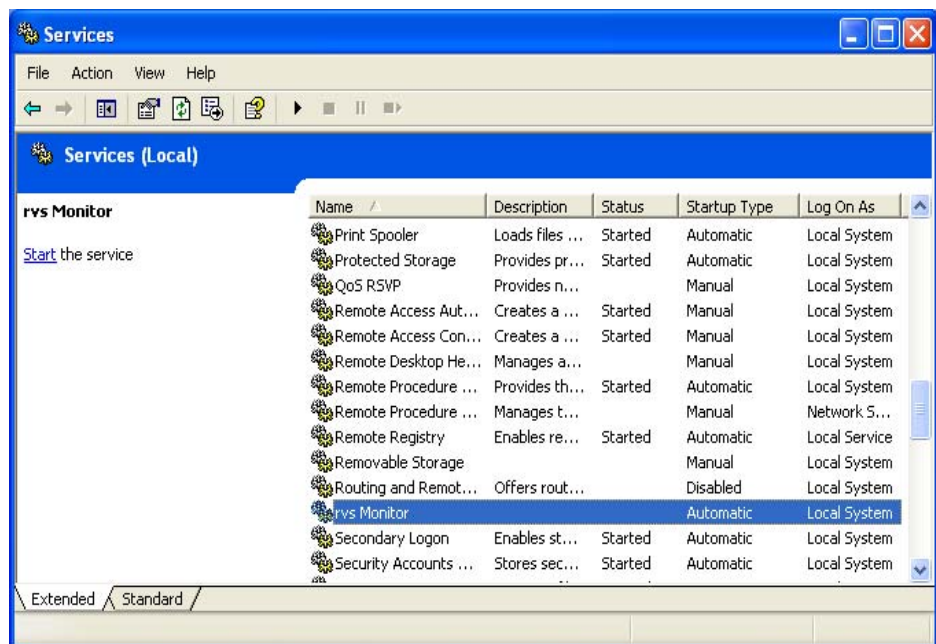
extern database rvsXP runs normally with a intern C-ISAM database. If you would like to install rvsXP with an extern database (e.g. Oracle or Microsoft SQL), please read the chapter 11 "rvsXP HighPerformance" for more information.

Specifying rvsXP Monitor / startup mode

The `rvs Monitor` service provides the applications with information on the current data exchange processes. **Note:** The term Service means a program that can be started from the operating system and works in the background.

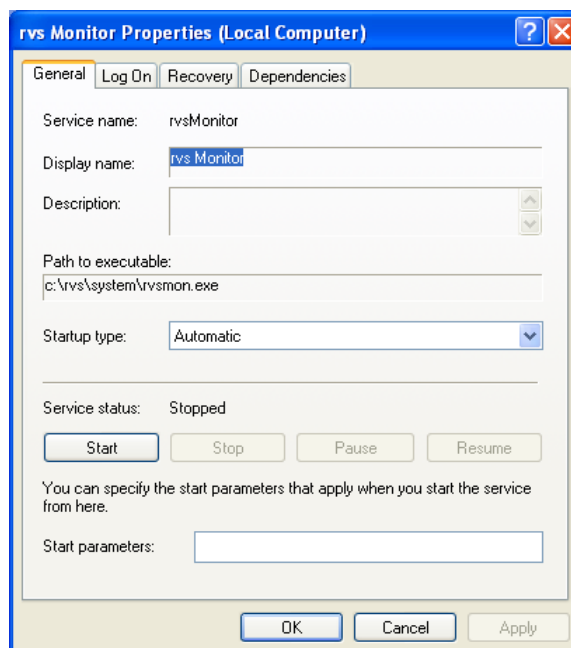
Automatic or manual service start Depending on your choice during the rvsXP installation, the startup type for this service is `Automatic` or `Manual`. The user must manually start the service if the `Manual` startup type was chosen. The rvsXP-Monitor is at first set to `stop` in this case. If you want to start rvsXP-Monitor every time the system starts, you can set the startup type to `Automatic` by choosing `Automatic` from the combo box in the **Startup type** area.

- Change startup type
- Choose `Start -> (right mouse) My Computer -> Change a setting`.
 - Launch the `Start -> Control Panel -> Administrative Tools -> Services` program.
 - Search the list of services for `rvs Monitor` and select the service by clicking the appropriate line.



- Double click on rvs Monitor.

The rvs Monitor Properties (Local Computer) window opens.



- Choose Automatic in the Startup type field and confirm with OK. The rvsXP Monitor startup type changes to Automatic.

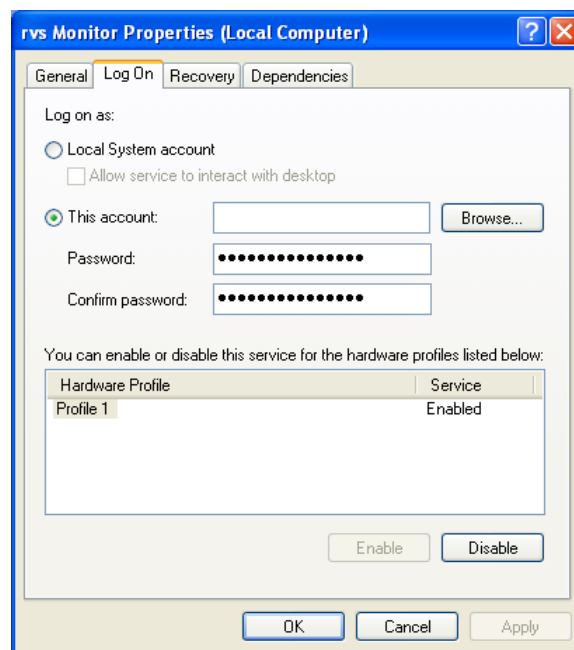
Assigning rvs[®] Monitor special user rights

Expanding access rights

By default the `rvs Monitor` service runs under the system user of the local machine. He has no access rights for network directories. If it is necessary for the `rvsXP Monitor` or the processes it controls (e.g. a batch job from a resident receive entry) to access network directories, the `rvs Monitor` service must work under a user authorized to do so.

To this end, proceed as follows:

- Double click on `rvs Monitor`.
The `rvs Monitor Properties (Local Computer)` window opens.
- Click the `Log On` tab.



- Enable `This Account` and then click the **Browse** button to choose a user. After selecting a user enter the password twice and acknowledge with **OK**. The service will now be assigned to this user and has the same rights.

Starting rvsXP-Monitor for the first time

- Double click on the `rvs Monitor` service.
- Click on the **Start** button.

Windows XP will now try to start the service. If the attempt is successful the `rvsXP Monitor` status will change to `Started`.

The entry is `rvs Monitor` `Started` `Automatic`.

2.4 The rvsXP program environment

After installation The rvsXP installation program creates a folder with all links required for the rvsXP program environment. The following table shows the program functions and the link instruction with the respective parameters. You can start the program by invoking the associated link, at the command prompt (Start -> All Programs -> Accessories-> Command Prompt) or by using the command Start -> Run.

| Program | Function / information | Link = command prompt command or Start / Run... |
|---------------------|---|---|
| Init Database | Initializes the rvsXP database. LOC is the standard station ID for your own (local) workstation. | C:\rvs\system\rvsiniNT.exe LOC |
| Get Machine ID | Reads a three- or four-character machine identification number and displays it on the screen. | C:\rvs\system\rvsgmid.exe |
| rvsXP administrator | Graphical configuration, administration and operating program for the rvsXP administrator. | C:\rvs\system\rvsmgr.exe |
| rvsXP | Graphical user interface for the rvsXP user. The tool for daily data exchange. | C:\rvs\system\rvsdiant.exe |
| Batch | Program for processing batch jobs when anticipated data arrive. | C:\rvs\system\rvsbat.exe |
| Delete Database | Deletes the rvsXP database. | C:\rvs\system\rvsdbdNT.exe |
| Monitor parameters | Opens the file with initialization commands. | notepad.exe C:\rvs\init\rdmini.dat |
| Readme file | Opens the readme file with the latest information that is not included in this manual. | notepad.exe C:\rvs\system\README.TXT |
| Stop rvs Monitor | Stops the rvsXP Monitor. | C:\rvs\system\rvsstop.exe |
| rvs Console | Command prompt window for displaying messages and entering operator commands. Has been replaced by the Monitor Log in the rvsXP Administrator MDI window. | C:\rvs\system\rvscns.exe |
| Dialog | rvsXP DOS prompt for creating and receiving send orders. Has been replaced by rvsXP. | C:\rvs\system\rvsdia.exe |

2.5 How to obtain a license

During installation, a license key file `rdkey.dat` is copied to the `$RVSPATH2\init\` folder.

Contact After initial installation rvsXP can be operated for test purposes without a license for a limited period of time. If you are satisfied with rvsXP and would like to continue working with it contact your sales representative or the rvs[®] support (phone: +49 30 / 399 71 777; telefax: +49 30 / 399 71 994, email: rvs-service@gedas.de). There you may obtain the desired license.

When you install a full rvsXP version you receive a license key file that does not contain an expiry date, i.e. there is no temporal restriction.

License key update

- Start the `Machine ID` application in the `rvs` folder. The program now displays a three- or four- character machine identification number on the screen.
- Inform your sales partner of the machine identification number and he will send you a license key by email.
- Make a backup copy of the old `C:\rvs\init\rdkey.dat` file in order to be able to restore the original file status.
- You should save the received license key as `rdkey.dat` file in the `Init` folder.

The license key has now been adapted.

² See Chapter 1.6 "Representation means" for more information.

Structure of the license key data

```

rdkey.dat file  0000000000      ! Customer Number      00000010
                NT          ! Product              00000020
                3.00.00    ! Release              00000030
                5LTX       ! Included Components  00000040
                TESTINSTALLATION ! Customer Name (Part  00000050
                                one)                          00000060
                                ! Customer Name (Part
                                two)
                                ! reserved                  00000070
                                ! reserved                  00000080
                                ! reserved                  00000090
                                ! reserved                  00000100
                                ! reserved                  00000110
                                ! Computer Model            00000120
                TESTVERSION ! PHYSICAL CPU        00000130
                                IDENTIFICATION
                94211      ! Expiration Date     00000140
                CA580BE5 20189735 612035F9 77AD6591 00000150
                162D0D7F 18FAB129 53B14EA4 79390F44 00000160
                ***** (C) gedas GmbH 1999 00000170
                *          make a success of rvs 00000180
                ***** 00000190

```

“Included Components” may contain two or more feature codes:

```

T    TCP/IP communications
X    X.25 native / ISDN communications

```

Not all features are available on all platforms at this time.

Expiration date The license expiration date is designated YYTTT where YY means the year, TTT the day-of-year of the respective year.

Note: Do not modify any of the fields in the license key file unless explicitly asked to do so by gedas GmbH.

3 Function Tests

| | |
|----------------------|--|
| Test without network | The present chapter describes two types of function tests. For one of the tests you send a file from your working environment to yourself, thus ensuring rvsXP correctly accepts and processes the jobs. For this test you do not yet need a functional network link because rvsXP-Monitor simply copies the data sets internally when you send to yourself. |
| Network line test | A second test is a network line test you can execute by activating the local station. |

3.1 Sending a file to yourself

To this end, proceed as follows:

- Start the `rvsXP Administrator` program in the `rvs` folder.
- Start `rvsXP-Monitor` if it has not already been started.
- Start the `rvs` program with the `Start Program Dia` command.

The `rvsXP rvsDia` window then opens and displays the list of files which have not yet been sent.

- Create a new send entry with **New Send Entry**.

The `Choose file to be sent` dialog box opens.

- Select any file and confirm your selection with **Open**.

Example: `C:\rvs\system\readme.txt`

The `New Send Entry` dialog opens and displays the file name selected in the `File` field.

- Enter the station code for your station in the `Station ID` field.

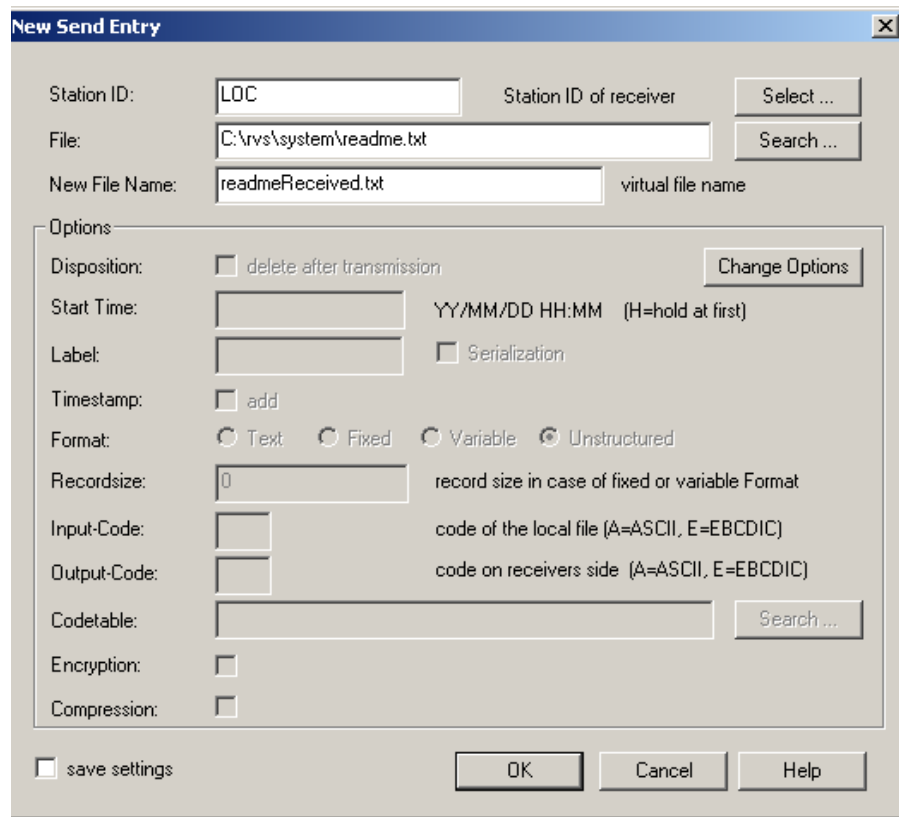
or:

- Choose `Select...` to open the `Select Station` dialog and select the ID.

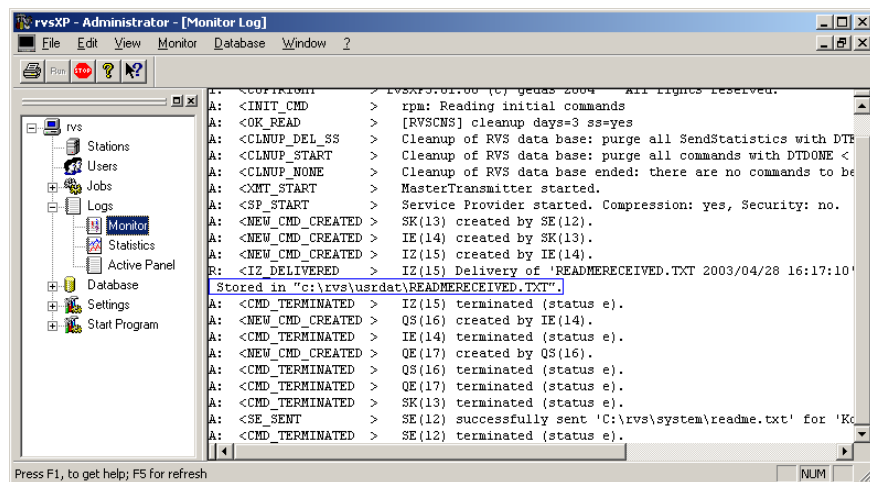
Example: `LOC`

`LOC` is the default setting.

- To send the file to the recipient with another name, type the new file name in the `New Name` text box.



- Confirm your entries with **OK**.
 This will save the send entry for transmission of the file in the rvsXP database. The job to send the `readme.txt` to your station has been created. rvsXP lists the job in the list of files waiting to be sent.
- Go to the `rvsXP-Administrator`.
 The `Monitor Log` window shows the rvsXP Monitor activities.
 The rvsXP-Monitor reads the send job and executes it: Using the LOC station ID rvsXP-Monitor recognizes the own station as the target and copies the file to the `C:\rvs\usrdat` folder.



- Check the output in Monitor Log for error messages.
- Eliminate any errors that occur in accordance with the advices in the "Messages and Return Codes" manual.

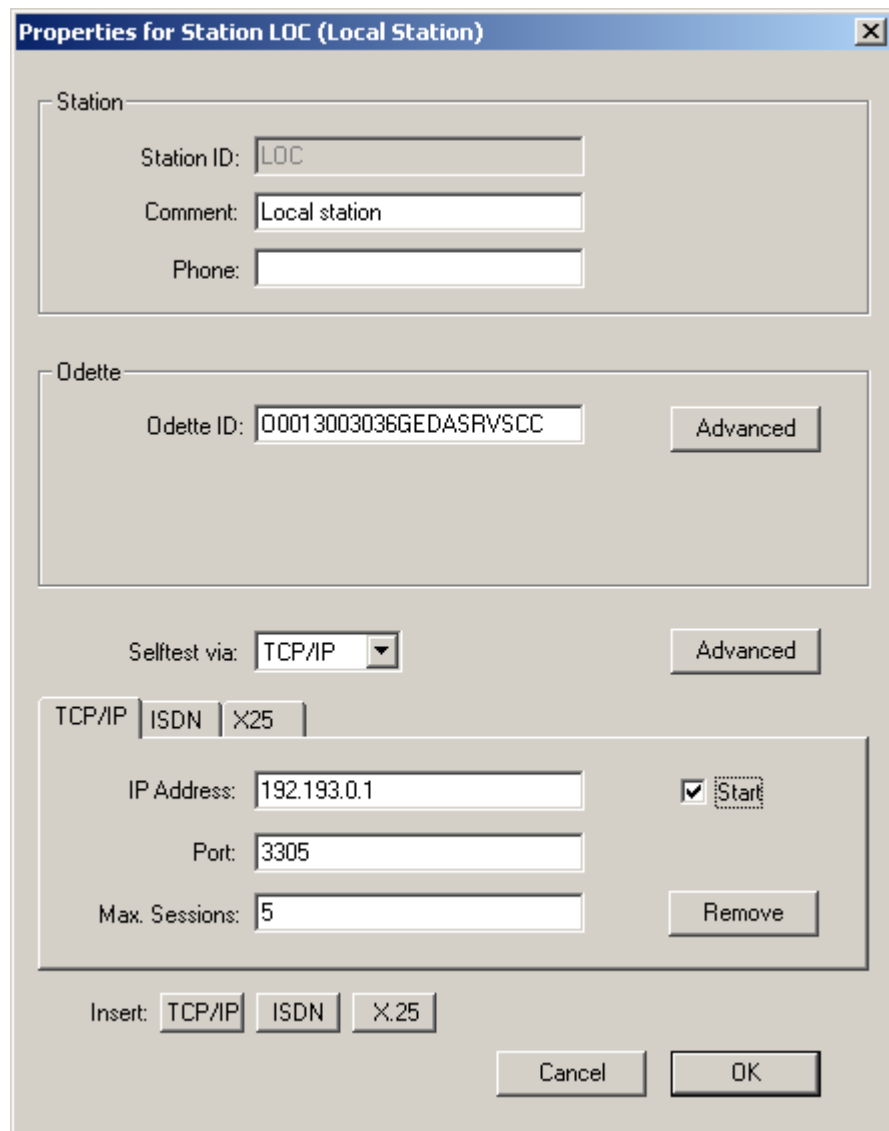
Note: Error messages start with **E** : and are written in red.

3.2 Testing the connection with the local and the partner stations

To ensure proper functioning of the network link you can use a self test to check the local station's connections. Perform the following steps, first for the local station. When the self test was successful you can perform a connection text with a particular partner station.

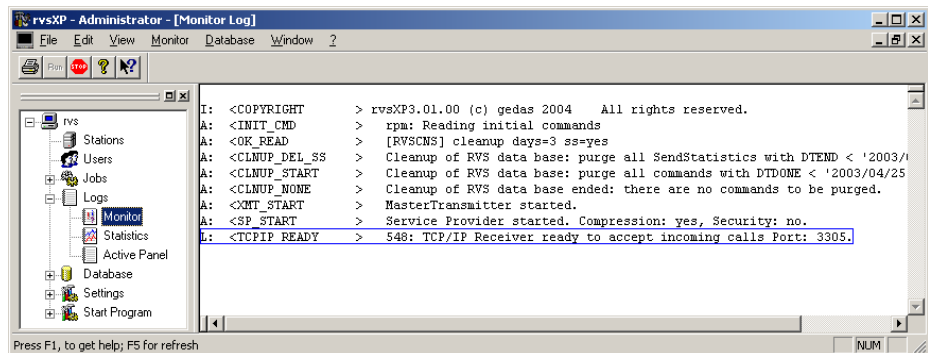
Preparing the local station

- Open the station table with the Stations menu command.
The window Station table for managing the partner stations opens.
- Select the local station.
- Double-click the station entry or choose Edit -> Update to edit the station.



- Click the tab at the bottom for the connection you wish to perform a line test with.
Note: Partner stations have only one tab here.
- Specify all line data.
 Heed the following
 - for TCP/IP: the IP address set for your network interface card
 - for ISDN: the correct phone number
 - for X.25: the correct X.25 address.
- Check the `Start` box.
- Confirm your entries with **OK**.
- Stop and restart rvsXP-Monitor.

- Check the output in Monitor Log. A message must indicate the chosen connection is ready to receive.



- Open the station table with the Stations menu command.

Link test between
local and remote
station

To test the link between local and remote station proceed as follows. Start with the local station first unless you are sure it functions correctly.

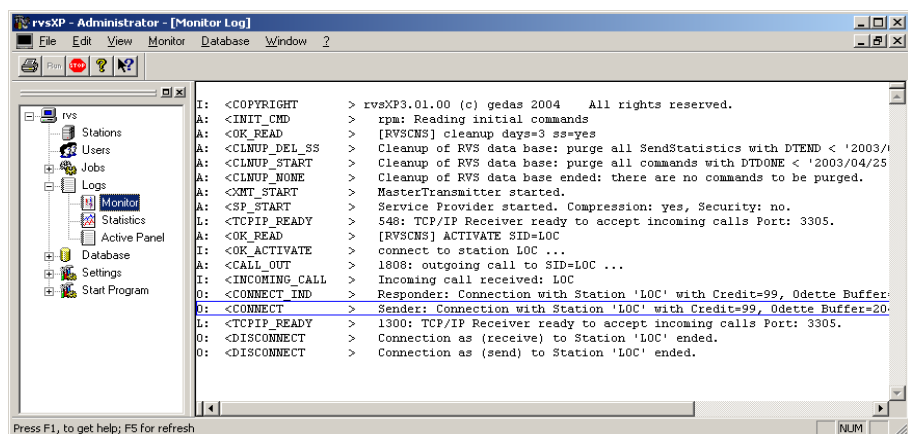
- Choose the local or partner station to be tested and choose Edit -> Activate.

rvsXP-Monitor reads the operator command and executes it.

The Monitor Log window shows the rvsXP Monitor activities. A message of the type

```
O: <CONNECT > Sender: Connection established
with station 'LOC' with credit=100, Odette
buffer=2047.
```

indicates a successful link test.



- Check the output in Monitor Log for error messages. Eliminate any errors that occur in accordance with the advices in the "Messages and Return Codes" manual.

4 Configuration of rvsXP

This chapter describes how to configure your rvsXP system, i.e. how to perform network and other basic configurations and how to assign rvsXP user rights.

Logging on as administrator

The `rvsXP-Administrator` is used to configure and manage rvsXP, and also for working with rvsXP. To be able to perform these tasks, the `rvsXP-Administrator` features appropriate detail windows. The rvsXP configuration requires that you have Administrator privileges at system level as well as in the rvsXP environment.

Program start with rvsTrayIcon

Tip: Using rvsTrayIcon

The installer sets up the `rvsTrayIcon` that facilitates launching the most frequently used rvsXP functions. After a Windows reboot you find this icon in the system tray.



To display the `rvsTrayIcon` right after installation,

- choose Start -> All Programs -> rvs -> rvs rvsTrayIcon.

Use the `rvsTrayIcon` to

- start the `rvsXP-Monitor`,
- stop the `rvsXP-Monitor`,
- start the `rvsXP-Administrator`, and
- open the `rvsXP` dialog where you can view and edit the `rvsXP` data transmission.



To start the `rvsXP` programs with the `rvsTrayIcon`

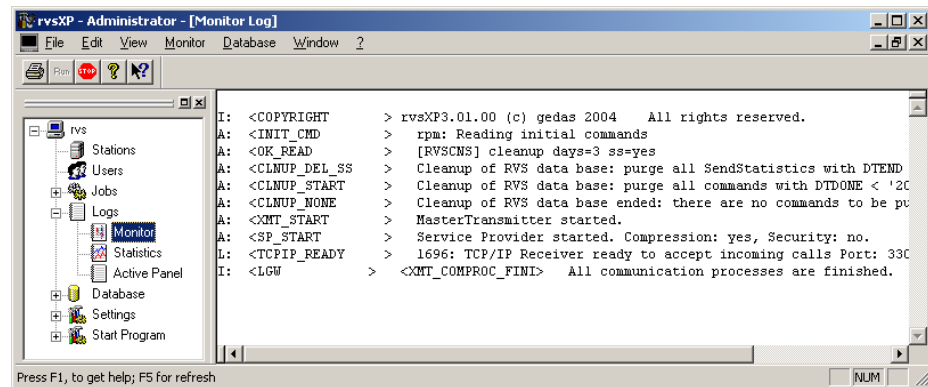
- click the `rvsTrayIcon` with the right mouse button and
- choose the desired menu item by clicking it with the left mouse button.

Starting the rvsXP-Administrator configuration program

- Start rvsXP-Administrator with the rvsTrayIcon command (right mouse button) rvsTrayIcon -> rvs Administrator or
- choose Start -> All Programs -> rvs -> rvs Administrator in the rvsXP folder.

Monitor log window

The rvsXP-Administrator then opens with the Monitor Log window displaying the current rvsXP-Monitor activities.



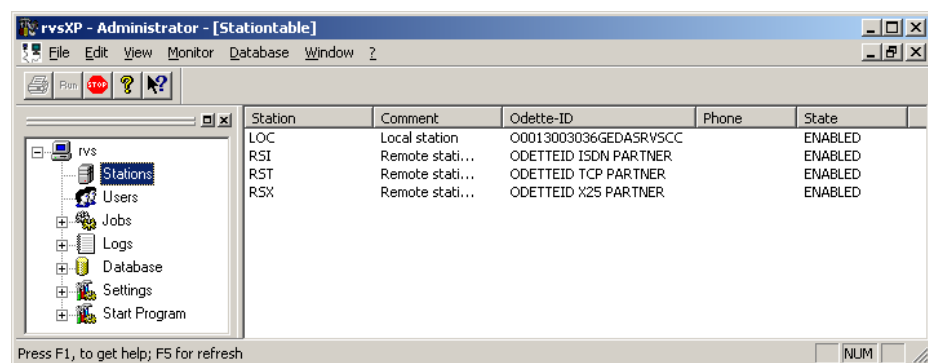
4.1 Setting up rvsXP stations

Before you can communicate with a partner station, you have to set up the partner station and your local station. The same also applies to your partner.

- Open the station table in the rvsXP-Administrator using the rvs -> Stations menu command.

Configuration example:

The window Station table for managing the partner stations opens. The table contains four sample stations after rvsXP has been installed:



LOC is the local station. If you have selected your own station ID for your local station during user-defined installation of rvsXP, then the station ID which you have selected will appear here.

Note: During default installation rvsXP sets up the following sample stations for the communication modes in the station table:

- RSI is a partner station for communication with ISDN.
- RST is a partner station for communication with TCP/IP and
- RSX is a partner station for communication with X.25.

Use the examples in order to set up your stations.

Updating a station To update station parameters,

- double-click the station entry or
- execute the menu command `Edit -> Update`.

Adding a station To set up a new station,

- click the right mouse button on the free area below the station list and choose `Add` or
- execute the menu command `Edit -> Add`.

The dialog window `Properties for Station` opens. It comprises three sections with station parameters.

The **Station** area shows the parameters that identify a station locally on your computer. **Odette** shows the parameters relevant for the ODETTE protocol. The bottom section of the dialog contains the parameters for the line and the pertaining line protocol.

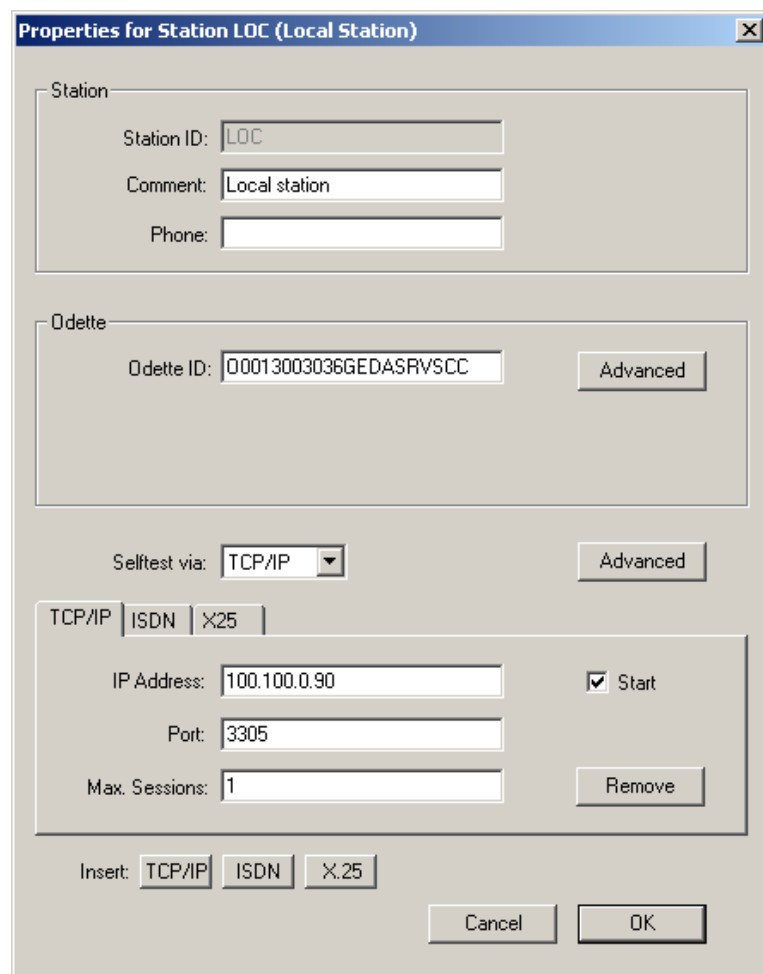
The screenshot shows the 'Properties for Station RST (Remote Station)' dialog box. It is organized into several sections:

- Station:** Contains text boxes for 'Station ID' (filled with 'RST'), 'Comment' (filled with 'Remote station via TCP/IP'), and 'Phone'. There is a 'Virtual' checkbox.
- Odette:** Contains text boxes for 'Odette ID' (filled with '00023003036GEDASRVSCC'), 'Send password' (filled with 'xxx'), and 'Receive password' (filled with 'xxx'). There is an 'Advanced' button.
- Connection:** A dropdown menu is set to 'TCP/IP', with an 'Advanced' button next to it.
- TCP/IP:** Contains text boxes for 'IP Address' (filled with '123.1.234.8') and 'Port' (filled with '3305').
- Buttons:** 'Cancel' and 'OK' buttons are located at the bottom of the dialog.

The dialog shows, depending on the feasibility of pertaining parameters, different property fields for your local station and the remote stations. The local station is always physically present and can never be a virtual station. Neither do you need a password to communicate with your own local station. This is why fields such as **virtual**, **Send password** and **Receive password** are available for remote stations only.

Simultaneous reception via several lines

While transfer to a remote station at a given time is possible via a particular data line only, several remote stations can send the data to your station via different data lines and line protocols. This is why connection parameters of the remote station always relate to one line type. Depending on the station's equipment, the parameters for the local station can comprise several lines and protocols.



Use this dialog to view and edit the station properties. Click the **Advanced** buttons in this window to open additional property dialogs for the Odette and Connection parameters areas.

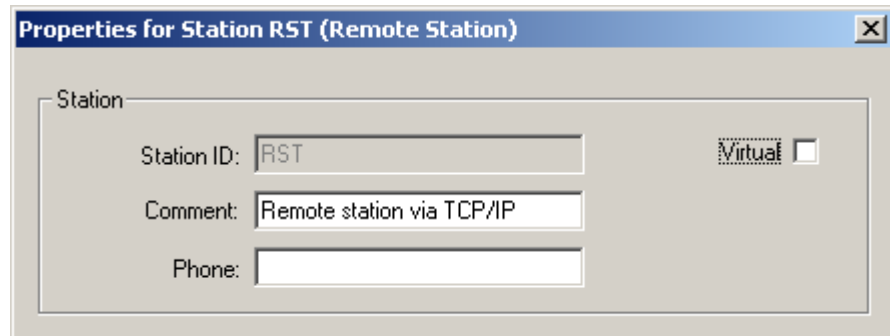
The abbreviations in this dialog stand for:

| | |
|------------------------|--|
| Station | The parameters that identify a station locally on your computer. |
| Odette | Parameters rvsXP needs for the ODETTE protocol. |
| Connection | For remote stations only: Connection type by which the station is reachable. |
| Self test via | Corresponds to the connection with remote stations: Path used to reach the local station during a self test. |
| X.25 ISDN TCP/IP | Details of the connection type chosen with <code>Connection</code> or <code>Self test</code> . |
| Routing | Parameters you need if you want to send files to an indirectly reachable station via your neighbor node. |

The rvsXP database saves the parameters in following tables

- **ST** **Station**table,
 - **OP** **O**dette **p**arameter,
 - **NK** line type (**N**achbar**K**noten = neighbor node),
 - **RT** **R**outingtable,
 - **XP** **X.25/ISDN-p**arameters and
 - **TC** **T**CP/IP networks .
- Enter the remaining parameters for the new station.
 - Confirm your entries with **OK**.

4.1.1 Setting up station parameters



The `Station` area uses three parameters to identify the station:

| Parameters | Meaning of station properties | Database parameter |
|--------------|--|--------------------|
| Station ID | Partner station identity marker consisting of up to 16 characters. | SID |
| Station name | Station name (optional). Enter a descriptive name for the station so that the station will be easily recognized by the user. | STATNAME |
| Phone | (Optional) telephone number for the contact person at the partner station. | PHONE |
| Virtual | Indicates whether or not the station is a virtual station. Additional information: See "Setting up a virtual station" on page 57. Possible values: L Automatic assignment for the local station, cannot be chosen as station property in the dialog V (virtual): = selected, R (remote): = not selected, remote non-virtual station Default: R (remote), not selected | SIDTYP |

Note: Do not confuse the optional phone number for the contact partner with the phone number used for data transmission. Type the latter in the connection parameters for X.25 or ISDN, depending on the connection type.

4.1.2 Setting up ODETTE parameters

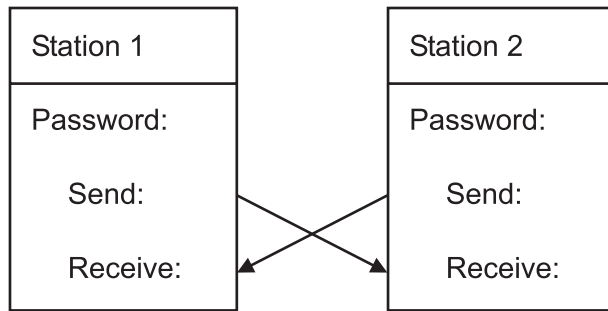
The Odette area contains information required by rvsXP for ODETTE file transfer.

For routing:
no passwords

If you access the partner station via a neighboring station (connection type = Routing) you do not have to enter the parameters for passwords. This information has already been entered at the neighboring station. In this case, rvsXP ignores the existing entries during the communication process.

| Parameters | Meaning of Odette parameters | Database parameter |
|------------|--|--------------------|
| Odette Id | In contrast to the station ID, which only has to be unique on your computer, the ODETTE ID is a worldwide, unequivocal identification of the partner or your own station, if ODETTE File Transfer Protocol is used. The ODETTE ID consists of 25 characters including the letter "O", the 18-character organization ID and the 6-character computer address within the respective organization. The computer address may be freely determined by the organization and must be unique. | ODETTEID |
| Send | Send password: rvsXP sends this password to the partner station as access authorization. | PSWTO |
| Receive | Receive password: rvsXP permits the partner station to have access with this password. | PSWFROM |

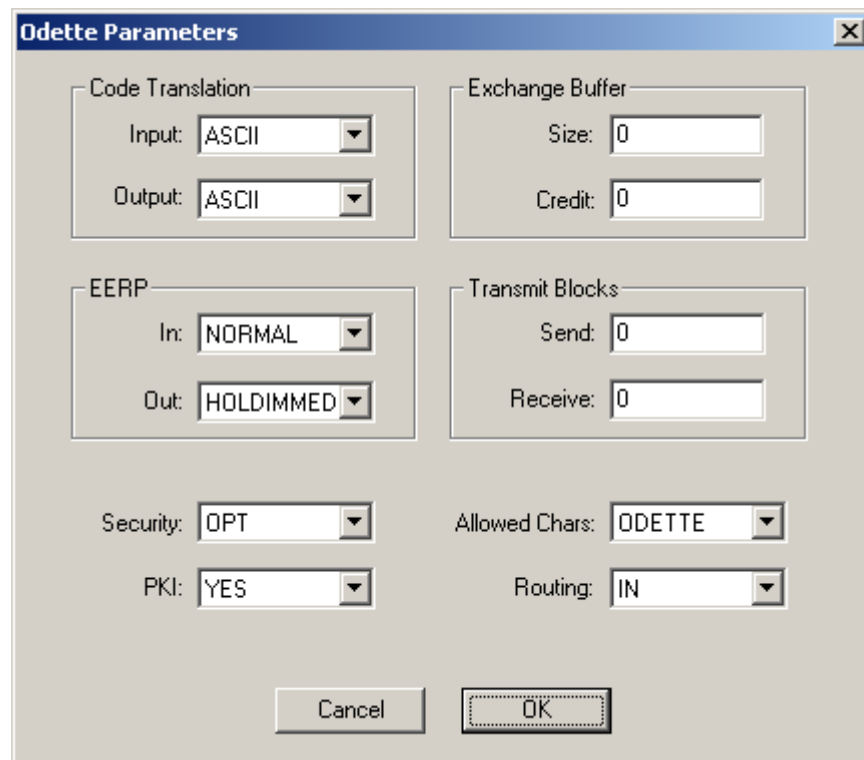
The local station does not display the password fields as no access authorization is required for communication with one-self. For a partner station you must agree on send and receive passwords with your partner. The partner must interchange the passwords during sending and receiving (see the illustration below). Communication aborts when transmission is attempted if the passwords do not match, and you receive an error message.



Note: You receive your personal ODETTE organization ID upon request from Verband der Automobilindustrie Abteilung Logistik (**VDA**), phone: +49-69/97507-283 Fax: +4969/97507-300.

rvsXP executes transfer with default values for the ODETTE file transfer protocol (OFTP).

- To view the default values or to change them, click the **Advanced** button. The `Odette-Parameter` window will open with advanced configuration options:



| Parameters | Meaning of advanced Odette parameters | Database parameter |
|-------------------------|---|--------------------|
| Exchange buffer size | Maximum transmission buffer size Default: 0, i.e. OEXBUF from parameter table applies. | OEXBUF |
| Exchange buffer credit | Maximum number of blocks sent without acknowledgement expected. Default: 0, i.e. OCREVAL from parameter table applies. | OCREVAL |
| Code translation input | Input format of the file which is to be sent, prior to conversion for the partner station. Values: ASCII : The files which are to be sent are in ASCII format EBCDIC : The files which are to be sent are in EBCDIC format Default: ASCII | CODEIN |
| Code translation output | Target format of the files for send transmission. Values: ASCII : During transmission rvsXP converts the files which are to be sent from input code format into ASCII format. EBCDIC : During transmission rvsXP converts the files which are to be sent from input code format into EBCDIC format. Default: ASCII | CODEOUT |
| Receive* | Number of blocks sent between two restarting positions for error correction. Default: 0, i.e. RECVBLOCKS from the parameter table applies. | RECVBLOCKS |
| Send | Number of blocks sent between two resuming positions for error correction. Default: 0 , i.e. SEENBLOCKS from the parameter table applies. | SEENBLOCKS |
| EERP IN | Send transmission (EERP = "End-to-End-Response") acknowledgement; Values: NEVER : Partner does not send EERP, file dispatch ends with successful transmission. Do not wait for acknowledgement. NORMAL : Wait for acknowledgement by partner after successful file transmission. File dispatch ends with receipt of EERP. Default: NORMAL | EERP_IN |

| Parameters | Meaning of advanced Odette parameters | Database parameter |
|--------------------|---|--------------------|
| EERP OUT | <p>Receive transmission (EERP) acknowledgement</p> <p>NORMAL: Prepare acknowledgement after successful reception of a file; send EERP only if a connection to the partner still exists, otherwise at next connection.</p> <p>IMMEDIATE: Prepare acknowledgement after successful file reception. Establish connection if there is no connection to the partner and send EERP to the partner.</p> <p>NEVER: Partner does not expect EERP. File reception ends with successful transmission. Do not acknowledge reception.</p> <p>HOLD: Prepare acknowledgement after successful file reception. However, do not send acknowledgement but wait for operator release. Send EERP after release only if a connection to the partner still exists, otherwise at next connection.</p> <p>HOLDIMMED: Prepare acknowledgement after successful file reception. However, do not send acknowledgement but wait for operator release. After the release: Establish connection if there is no connection to the partner and send EERP to the partner.</p> <p>Default: IMMEDIATE; Recommended: NORMAL</p> <p>Release acknowledgement again: See "Handling files to be sent and received" on page 80.</p> | EERP_OUT |
| Allowed characters | <p>Stipulates which characters are permitted for Odette-FTP:</p> <p>ALL: no limitations</p> <p>OFTPUNIX: all capital letters, digits and special characters ".-"</p> <p>UNIX: all letters and figures and the special characters "#_+."</p> <p>ODETTE: all capital letters, figures and special characters "()-./&" and blanks</p> <p>CHECK_RE: as with ALL; in addition, a resident receive entry RE must be present.</p> <p>Default: ODETTE</p> | VDSNCHAR |
| Compression | <p>Type of compression during the transfer</p> <p>NONE: the ODETTE blocks are not compressed</p> <p>ODETTE: the ODETTE blocks are compressed following the ODETTE standard</p> <p>RVS: no longer up-to-date; has no effects.</p> <p>Default: ODETTE</p> | FLAGCOMP |

| Parameters | Meaning of advanced Odette parameters | Database parameter |
|-----------------|---|--------------------|
| Encryption | <p>Defines whether or not encryption is to be used during file transmission. Make this setting either for all stations in <code>rdmini.dat</code> or for each single station in the ODETTE parameter configuration. The monitor parameter is applied to those stations for which the parameter has not been set.</p> <p>Values:</p> <p>SECURITY=NO Encryption is impossible. The job aborts with an error message if a send job requires encryption.</p> <p>SECURITY=OPT Encryption possible as an option and can be specified in the send job.</p> <p>SECURITY=FORCED Encryption is compulsory. A warning is issued and the send job is converted into an encrypted job if a send job is scheduled without encryption.</p> <p>Reception of the file is refused if the partner station sends an unencrypted file.</p> <p>A send job for station 'S' is processed according to the SECURITY entry for station 'S', regardless of whether 'S' is a neighboring station or is reached via routing.</p> <p>Default: SECURITY=OPT</p> | SECURITY |
| PKI | <p>rvsXP can obtain public keys from partners via LDAP Interface from a PKI (public key infrastructure). You can specify to use PKI instead of the rvsXP key memory for a particular station using the PKI parameter. Possible values:</p> <p>PKI= YES PKI= NO</p> <p>Default: NO</p> | USEPKI |
| Routing | <p>Determines the global rvsXP routing behavior.</p> <p>O(OUT): Routing is allowed for send jobs only. The own local station must not function as a router.</p> <p>B(BOTH): Standard OFTP routing, routing is allowed in send and receive direction.</p> <p>Default: B</p> | ROUTING |
| OFTP version | <p>This parameter is important, only if your partner has a OFTP product, which does not support OFTP 1.3. Normally must one OFTP product be able to negotiate a ODETTE protocol version during a ODETTE session. If your partner is not able to do so, you can set the protocol version with this parameter. Possible values: 1 (OFTP 1.2) or 2 (OFTP 1.3)</p> <p>Default: 2</p> | OFTPLEV |
| user separation | <p>This parameter is important only in communication with rvsMVS. It enables to send the userID of the user, who is sending a file in the ODETTE command SFID. Possible values: YES or NO.</p> <p>Default: NO.</p> | |

*** Tip:**

The greater the number, the more improbable the data transmission will be interrupted for entry of the restarting positions, but the greater will be the quantity of data to be transmitted again in the event of error. Choose:

Large values for reliable data lines in order to use the line quality for greater transmission speeds.

Small values for poor line quality in order to minimize the time required for repeated data transmission.

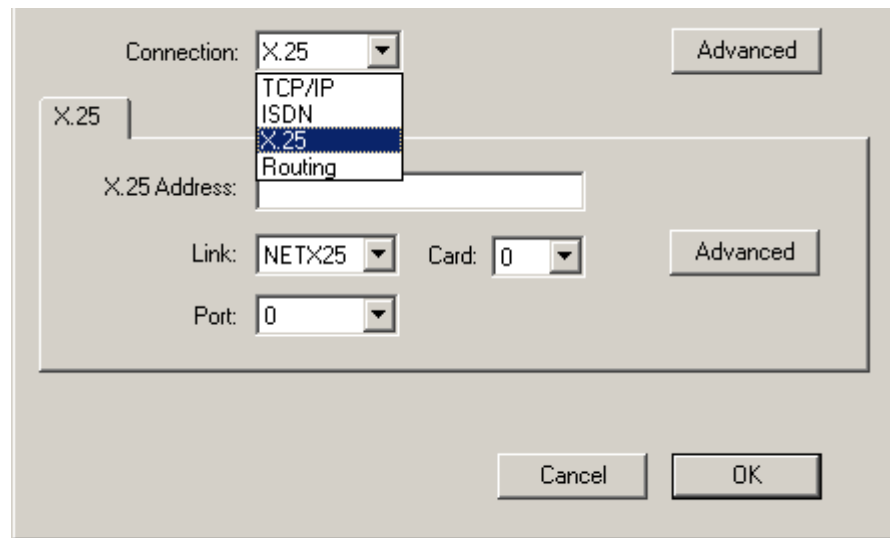
Tip to increase transmission speed

In order to increase the efficiency during the data exchange, you can use off-line compression (see 1.3 "Encryption and offline compression" and 6.3.1 "Creating send orders").

The size of compressed ODETTE blocks is set with parameter **block size (OEXBUF)**.

4.1.3 Setting up the neighboring station

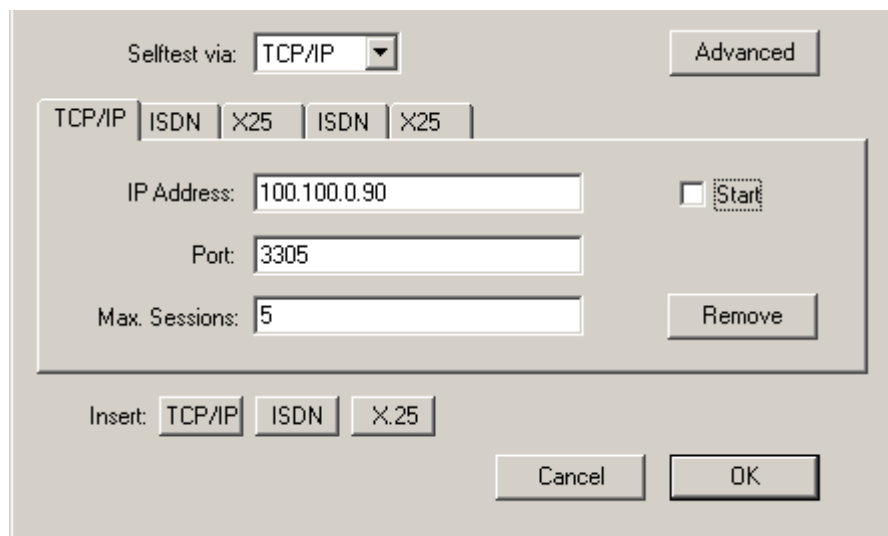
The `Connection type` tab contains information about the path via which the partner station is to be reached.



You can use the mouse to fold out the `Connection` combo box and choose the desired connection type. Depending on the selection made the tab below changes and shows the pertaining parameters.

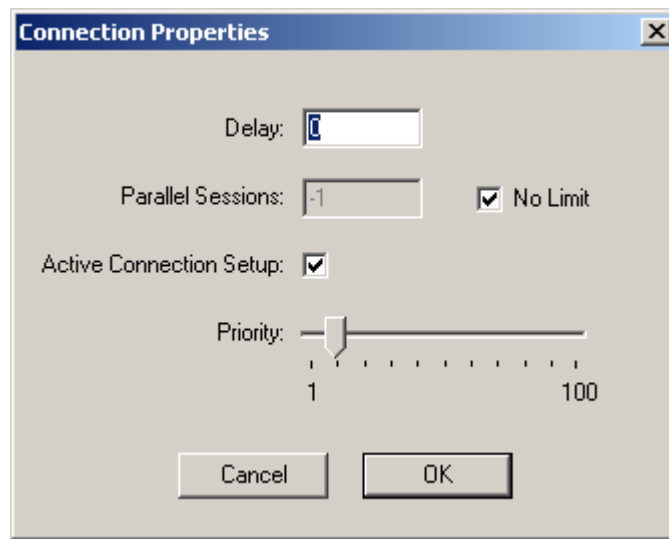
Multiple reception

The connection type defines the reception type for the local station. Just as your local station can feature reception devices you can set up several connection types. rvsXP supports multiple reception for TCP/IP, ISDN, and X.25.



| Parameters | Connection parameter meaning | Database parameter |
|--|---|---|
| Connection (this parameter is visible only if the partner station is configured) | The connection type with which the station is to be reached; Values: TCP/IP : The partner station is reached via a network with the TCP/IP protocol. ISDN : The partner station is reached via an ISDN line. X.25 : The partner station is reached via an X.25 line. Routing : The partner station uses a different partner station as a routing station. Enter the routing station ID in the RT-Routing tab. The connection type and parameters of the routing station then apply to the connection itself. For further information please refer to Chapter "Setting up the routing connection". Default : none | PROTOCOL Values: T = TCP/IP X = X.25/ISDN R = Routing |
| Self test via | May only be selected at the local station: Line by which transmission is to be made if the self test is performed when your own station is activated | PROTOCOL |
| Receiver | May only be selected at the local station: Receivers which have been configured and/or are still to be configured | No parameter |

Use the **Advanced** button to the right of where you choose the connection to open the Connection Properties dialog providing more parameters.



| Parameters | Connection parameter meaning | Database parameter |
|---------------------------------|---|--|
| Delay | Time (in seconds) to wait after receiving a file before a "Change Direction" (CD) is initiated | CDWAIT |
| Parallel sessions | Maximum number of active simultaneous connections via this line Default: -1 (no restrictions) | PSESSIONS |
| Active connection establishment | Specifies the send behavior for a pending send order; Values: ✓ :Sender should automatically start as soon as a file is ready for dispatch. □ :Pending send orders do not trigger a send action. Default: ✓ | AUTODIAL Values: Y = ✓ N = □ |
| Priority | The priority number is planned for rvsXP updates. It is reserved for future use. | PRIORITY |

Establishment of an active connection

Preparing files for fetching

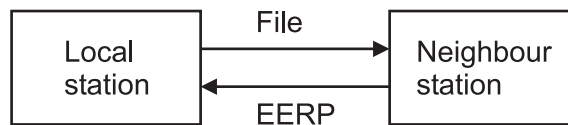
It is occasionally necessary for files to be made only available and not to be sent immediately. On the contrary, the partner station is to establish the connection and fetch the available files as and when required. The partner bears the costs for the connection. Active connection establishment must be switched off for this case.

4.1.4 Setting up the routing connection

What is routing? The forwarding of files throughout several OFTP systems is referred to as **routing**. If you would like to send a file via several systems, then the send order only has to be directed to the recipient. rvsXP then sends the file to the next OFTP system, the neighboring node which, in turn, then forwards the file. All stations must be defined in the station administrations of all stations involved.

Data flow with and/or without routing is depicted in the following illustrations:

Without routing



With routing



Acknowledgement: EERP When the file has arrived at the end system, it sends an acknowledgement **End-to-End-Response** back to the original sender. The send order is only completed as soon as the sender has received this acknowledgement.

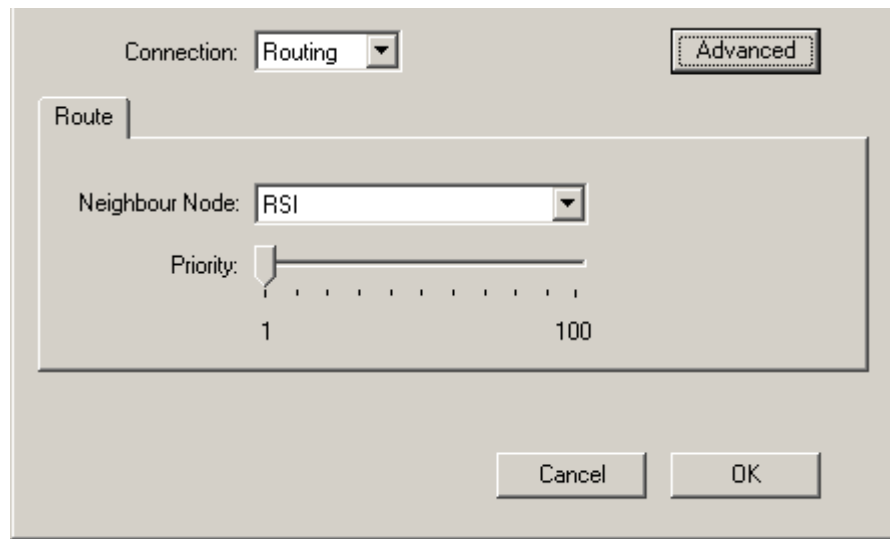
Prerequisites You have to define 2 stations in the `station table` for routing:

- the neighbor station (neighbor node) through which the end recipient is reached. It must have been set up prior to the target station.
- and the target station for the end recipient.

Select `Routing` as the connection type on the `Connection type` tab for this station, and the station ID for the neighbor station on the `Route` tab.

This is how you set the stations for routing

- Choose the station from the station table as usual and double-click the entry in the list or execute the menu command `Edit -> Update`.
- Choose `Routing` as connection type in the `Connection type` area. (Your partner station is not a direct neighbor station of yours.)



- Complete the `Route` tab.

The RT database table contains the neighbor station information by means of which the station selected (= target station) is to be reached.

Possible parameters are:

| Parameters | Connection parameter meaning | Database parameter |
|-------------------|--|--------------------|
| Neighboring nodes | Station ID of partner station (= neighbor station) which knows the path to the target station. The station ID has to be in the station table. | SIDNEIGHB |
| Priority | Priority number of neighbor station. It determines the data transmission route. The integer value ranges between 1 and 100 . The neighbor station with the lowest value comes into operation. The priority at the time when the send command is prepared is decisive. | PRIORITY |

Sample set-up of a target station

The destination MAR is reached with routing via RST.

- Set up station name: **MAR**
- Set up ODETTE parameters of target station **O001300...**
- Set up connection type: **Routing**
- Select neighbor station: **RST**

Properties for Station (Remote Station)

Station

Station ID: Virtual

Comment:

Phone:

Odette

Odette ID:

Send password:

Receive password:

Connection:

Route

Neighbour Node:

Priority: 1 100

4.1.5 Setting up TCP/IP communication

Connection:

TCP/IP

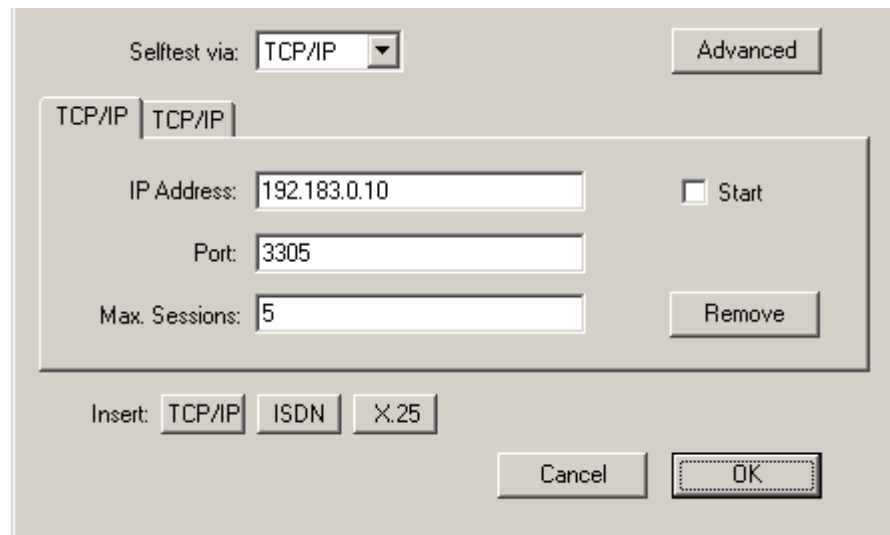
IP Address:

Port:

If you have selected the TCP/IP connection on the `Connection` type tab, you can edit the line parameters for the network connection on the `TCP/IP` tab with TCP/IP protocol.

Local station

rvsXP supports multiple TCP/IP reception under various Internet addresses at the same time. You can add additional TC parameter sets on your local station (LOC) using the **TCP/IP** button below the TCP/IP tab. Press the **Remove** button to delete the currently displayed set. rvsXP will automatically assign a number for the receiving channel (Port) when you add a new record.



Note: Enable the **Start** checkbox to activate the recipient. If the configured recipient is to be automatically active after the rvsXP Monitor has been started and is to wait for incoming calls, the parameter **TCPIPRCV** must be set to a value larger than **0**. For further details, read the Chapter "The rvs@ parameters".

Partner stations

The TCP/IP address has to be unique for a send transmission to a partner station. Thus you may only indicate one Internet address.

| Parameters | Meaning of the line properties TCP/IP | Database parameter |
|-----------------|--|-----------------------|
| Receiver number | Applicable and selectable at the local station only: Number for differentiating the various receiving channels through which the local station can be reached. Each number has a tab with a set of TCP/IP parameters. rvsXP automatically assigns and manages this number. Default: 0 | N |
| IP address | Internet addresses have the form "255.255.255.255" If you have not specified a value for the local station, you permit automatic definition of the Internet address. If your own station has only one Internet address, this field should be left empty. Default: " " | INADDR |

| Parameters | Meaning of the line properties TCP/IP | Database parameter |
|------------------|---|-----------------------|
| Port | Port address Default: 3305 | PORT |
| Maximum sessions | May only be selected at the local station: Maximum number of simultaneously running receiving processes on this channel. Default: 1 | MAX_IN |

4.1.6 Setting up the ISDN parameters

The screenshot shows a configuration window for ISDN. At the top, 'Connection:' is set to 'ISDN' with a dropdown arrow. To its right is an 'Advanced' button. Below this is a tabbed interface with the 'ISDN' tab selected. Inside the ISDN tab, there is a text field for 'ISDN Number' containing '1234567890'. Below that are two dropdown menus: 'Link' set to 'CAPI2A' and 'Karte' set to '0'. To the right of these is another 'Advanced' button. At the bottom of the ISDN tab is a 'Protocol' dropdown menu set to '1TR6'. At the very bottom of the main window are 'Cancel' and 'OK' buttons.

If you have selected the ISDN connection in the `Connection type` area, you can edit the line parameters for the network connection on the ISDN tab. You may be ready to receive for a partner station via several ISDN numbers at the same time.

- You must specify the parameters shown in this tab to make the ISDN connection work.

Local station

rvsXP supports multiple ISDN reception through various receiving channels (phone numbers) at the same time. You can add additional XP parameter sets for ISDN on your local station (LOC) using the **ISDN** button below the **ISDN** tab.

Press the **Remove** button to delete the currently displayed set. If you add a new record, then rvsXP will automatically assign a number for the recipient.

Note: Enable the **Start** checkbox to activate the receiver. If the configured recipient is to be automatically active after the rvs Monitor has been started and should wait for incoming calls, the parameter **MAXX25RCV** must be set. For further details, read Chapter “ISDN/X.25 reception”.

Partner stations

1 set of parameters only

The call number has to be unique for a send transmission to a partner station. Thus you may only indicate one set of parameters.

Click the **Advanced** button to the right of the parameter input boxes to open the **ISDN (Advanced)** dialog where you can edit these and other parameters.

- Edit the parameters and click the **OK** button to save the changes or click the **Cancel** button to discard the changes.

The table below explains the individual parameters.

| Parameters | Meaning of parameters ISDN | Database parameter |
|--------------|---|-----------------------|
| ISDN number | An ISDN number under which the partner station may be reached. For the 1TR6 ISDN standard the single-digit (EndSelectionNumber) (ESN) is assigned to the local station. Default: " " | ISDNNO |
| X.25 address | 15-digit X.25 DTE address of the partner station. This number is optional. It is however advisable for an ISDN connection to enter the ISDN number here, as some partners expect an X.25 address. | XADDRESS |

| Parameters | Meaning of parameters ISDN | Database parameter |
|---------------------|--|-----------------------------------|
| Link | <p>Link name; determines the type of connection within your computer. It is derived from the values for card type and card number.</p> <p>CAPI2mn Values for m and n:</p> <p>m: A = no Diehl/Eicon card 0 = Diehl/Eicon card</p> <p>n: Number of the X.25 card installed in the computer, beginning with "0"</p> <p>Example: CAPI200 = "Diehl/Eicon card" number "0" Default: CAPI200</p> | LINK |
| Protocol | <p>Specifies the ISDN standard used.</p> <p>1TR6 German national standard E-DSS1 EURO-ISDN Standard: E-DSS1</p> | ISDN_PROT |
| Packet size | Size of data packets during data transmission. | L2PACKETSZ |
| Alias | Alias name to be used in rvsXP-Monitor and in the LOG files. | ALIAS or RECV ALIAS |
| Timeout | <p>Cancellation time in seconds, after which the communication program aborts the connection if the partner station or CAPI does not reply.</p> <p>Default: 30</p> | TIMEOUT |
| X.25 Facilities | Special information or facilities for an X.25 transmission; see the information of the X.25 service in use. | FACILITIES |
| X.25 UserData | User data for an X.25 transmission; see the information of the X.25 service in use. | USERDATA |
| Terminal identifier | <p>Only required for X.31: Terminal End Identification.</p> <p>Default: 0 – no TEI assigned</p> | ISDN_TEI |
| ISDN Facilities | Special information or facilities of the ISDN transmission. | ISDN_FAC |
| ISDN UserData | User data for the ISDN transmission | ISDN_USRDT |
| Closed User Group | <p>ISDN and X.25 allow for the formation of a closed user group. All members of such a group can communicate with each other via the public telecommunication network. Connection requests to group members received from participants not being a member of the closed user group will be rejected by the switching exchange. The same applies to connection requests from group members to participants not being a member of the closed user group. This service attribute is called Closed User Group (CUG).</p> | CUG |

| Parameters | Meaning of parameters ISDN | Database parameter |
|---------------------|--|-----------------------|
| Rcv Timeout | Only valid for the local station: Cancellation time in seconds, after which the receiver interrupts the waiting for incoming calls and reconnects to the card driver. | RCVTIMEOUT |
| Parall. sessions | Maximum number of parallel active connections. The following applies for reception: As soon as a call has been accepted, the master transmitter starts a new receiver until the maximum number is reached. | SESSIONS |
| Receiver number | Applies to the remote station only: Number of the receiver accepting the call from the remote station. Each number of the local station has a tab with a set of XP parameters. Default: 1 | N |

4.1.7 Setting up X.25 parameters

The screenshot shows a configuration dialog box for X.25 parameters. At the top, there is a 'Connection:' dropdown menu currently set to 'X.25' and an 'Advanced' button. Below this is a tabbed interface with the 'X.25' tab selected. Inside the tab, there is an 'X.25 Address:' text box containing the value '1234567890'. Below the address field are three dropdown menus: 'Link:' set to 'EICON', 'Card:' set to '0', and 'Port:' set to '0'. To the right of these dropdowns is another 'Advanced' button. At the bottom of the dialog box are 'Cancel' and 'OK' buttons.

If you have selected the X.25 connection in the Connection type area, you can edit the line parameters for the network connection on the X.25 tab. You may be ready to receive for a partner station via several X.25 numbers at the same time.

- You must specify the parameters shown in this tab to make the X.25 connection work.

Local station

rvsXP supports multiple X.25 reception through various receiving channels (phone numbers) at the same time. You can add additional X.25 parameter sets on your local station (LOC) using the **X.25** button below the X . 25 tab.

Press the **Remove** button to delete the currently displayed set. rvsXP will automatically assign an internal consecutive number for the recipient when you add a new record.

Note: Enable the **Start** checkbox to activate the receiver. If the configured recipient is to be automatically active after rvsXP Monitor has been started and should wait for incoming calls, the parameter **MAXX25RCV** must be set. For further details, read Chapter “ISDN/X.25 reception”.

Partner stations

1 set of parameters only

The call number has to be unique for a send transmission to a partner station. Thus you may only indicate one set of parameters.

Click the **Advanced** button to the right of the parameter input boxes to open the X . 25 (Advanced) dialog where you can edit these and other parameters.

- Edit the parameters and click the **OK** button to save the changes or click the **Cancel** button to discard the changes.

The table below explains the individual parameters.

| Parameters | Meaning of parameters X.25 | Database parameter |
|--------------|--|-----------------------|
| X.25 address | 15-digit X.25 DTE address of the partner station. Specification is optional for an ISDN connection. It is however advisable for an ISDN connection to enter the ISDN number here, as many partners expect an X.25 address. | XADDRESS |

| Parameters | Meaning of parameters X.25 | Database parameter |
|-------------------|--|-----------------------|
| Link | <p>Link name; determines the type of connection within your computer. It is derived from the values for port, card type and card number.</p> <p>EICONn for X.25 Eicon cards</p> <p>n: Port number of the configured X.25 port, beginning with "1"</p> <p>Example: EICON1</p> <p>NETX25mn for TECCOM cards</p> <p>Values for m and n:</p> <p>m: Number of the X.25 card installed in the computer, beginning with "0"</p> <p>n: Port number of the configured X.25 port, beginning with "0"</p> <p>Example: NETX2500 = Card number. "0" with port number "0"</p> <p>Default: NETX2500</p> | LINK |
| Facilities | Special information or facilities for an X.25 transmission; see the information of the X.25 service in use. | FACILITIES |
| Userdata | User data for an X.25 transmission; see the information of the X.25 service in use. | USERDATA |
| Timeout | <p>Cancellation time in seconds, after which the communication program aborts the connection if the partner station or CAPI does not reply.</p> <p>Default: 30</p> | TIMEOUT |
| Parallel sessions | Maximum number of parallel active connections. The following applies for reception: As soon as a call has been accepted, the master transmitter starts a new receiver until the maximum number is reached. | SESSIONS |
| Timeout | Only valid for the local station: Cancellation time in seconds, after which the receiver interrupts the waiting for incoming calls and reconnects to the card driver. | RCVTIMEOUT |
| Receiver Number | <p>Applies to the remote station only: Number for differentiating the various recipients through which the local station can be reached. Each number has a tab with a set of XP parameters. rvsXP automatically assigns and manages this number.</p> <p>Default: 0</p> | N |

| Parameters | Meaning of parameters X.25 | Database parameter |
|-------------------|---|--------------------|
| Closed User Group | ISDN and X.25 allow for the formation of a closed user group. All members of such a group can communicate with each other via the public telecommunication network. Connection requests to group members received from participants not being a member of the closed user group will be rejected by the switching exchange. The same applies to connection requests from group members to participants not being a member of the closed user group. This service attribute is called Closed User Group (CUG). | CUG |
| Virtual Circuit | X.25 are referred to as channels and are assigned a channel ID each (virtual circuit). There are SVC (switched virtual circuits) and PVC (permanent virtual circuits for leased lines). You often need to specify the number of channels when configuring an X.25 layer. These are enabled by the X.25 access provider and often charged for according to their number. You need at least 2 SVC for parallel receiving and sending. | VC |

4.1.8 Setting up a virtual station

What are virtual stations?

Virtual stations are used to represent stations outside the OFTP network for the OFTP network. They allow files to be sent to destination stations outside the OFTP network.

ODETTE-IDs uniquely define the stations in an OFTP network. Each virtual station must also be assigned an own ODETTE ID. A virtual station can also send and receive.

These stations are specially flagged with the SIDTYP=V parameter value in the ST database table. To configure them you only need the OP database table in addition because virtual stations have their own ODETTE ID.

Receiving at virtual stations

Upon receipt in an OFTP system the file is not forwarded if the final file receiver's **SID (SIDDEST)** database parameter) matches the **SID** of the local station (**LID**). In this case the file is delivered locally, and EERP is created and a post-processing script starts if necessary.

In a routing case the final receiver's **SID (SIDDEST)** does not match the local station's **SID**. In this case a send entry is created that forwards the file to the actual receiver.

Upon receipt of an EERP a verification as to whether the receiver Odette ID in the EERP matches that of a virtual station is performed. In this case the file is delivered locally, and EERP is created and a post-processing script starts if necessary.

| | |
|------------------------------------|---|
| Sending from virtual stations | The send job is given the SIDORIGIN parameter for sending from virtual stations. This parameter must be filled with the station's SID when sending from a virtual station. When this has been done, the value of SIDORIGIN is written to the SFID as sending station for the file to be sent. |
| Setting up virtual stations | The process of creating and setting up virtual stations is basically identical with that of other stations. Refer to Chapter 4.1 "Setting up rvsXP stations". |
| Updating/removing virtual stations | The process of editing and removing virtual stations is basically identical with that of other stations. See also Chapter 4.1.9 "Editing the partner station". |

4.1.9 Editing the partner station

Editing properties

To edit the parameters of an existing station,

- In order to do so, open the station table in the `rvsXP-Administrator` using the `Stations` command.
- Double-click the station entry
or
- choose the station to be edited and execute the menu command `Edit -> Update`.

The dialog window `Properties for Station` opens. It comprises three sections with station parameters.

You may now edit all the station parameters with the exception of the station ID. Please refer to Chapter 4.1.1 "Setting up station parameters" for further information.

- Edit the station properties and confirm your entries with **OK**.

Renaming a station

Renaming To rename a station,

- In order to do so, open the station table in the `rvsXP-Administrator` using the `Stations` command.
- Choose the station to be renamed and execute the menu command `Edit -> Rename`.
or
- Click the station to be renamed with the right mouse button and choose `Rename`.

The `Rename station` dialog opens.

- Type a new name and confirm your entries with **OK**.

Setting the status

Setting the status You may activate, stop or set a station working. To change the status of a partner station:

- Select the station to be changed in the window `station table`.

Activate

- Activate
- In order to activate the station execute the menu command `Edit -> Activate`.

Note: The station must already be `Enabled` before you can activate it. `rvsNT` establishes the connection to the partner station and transmits the currently pending files.

Activating a station means establishing an OFTP connection to this station. As soon as the connection has been established, each side has the opportunity to send any files that may possibly be available. If no file is ready to be sent, the connection is closed.

Set working

- Default setting (set working)
- In order to set the station working execute the menu command `Edit -> Enable`.

`rvsXP` is now ready for communication with the partner station. The status is `Enabled`. This is the standard setting for a station. A connection is established to this station as soon as data is available to be sent to this station.

Stopping working

- Stopping
- In order to stop the station working execute the menu command `Edit -> Disable`.

`rvsXP` performs no send orders to the partner station but keeps them in the queue until you set the station working again. The status is `Disabled`.

Saving

- Saving settings
- When you set up or change a station, save the changed parameters by pressing **OK**.

When Windows XP is restarted, `rvsXP` continues to work with these parameters. However, if you delete the `rvsXP` database or reinstall, you also delete the newly set up stations. Subsequent initializing of the database only provides the specified standard configuration of the station table.

In order to avoid losing your entries, you may back up the station table in a separate file and restore it later. For further information please see the Chapter entitled 7.1.1 "Backing up the station table".

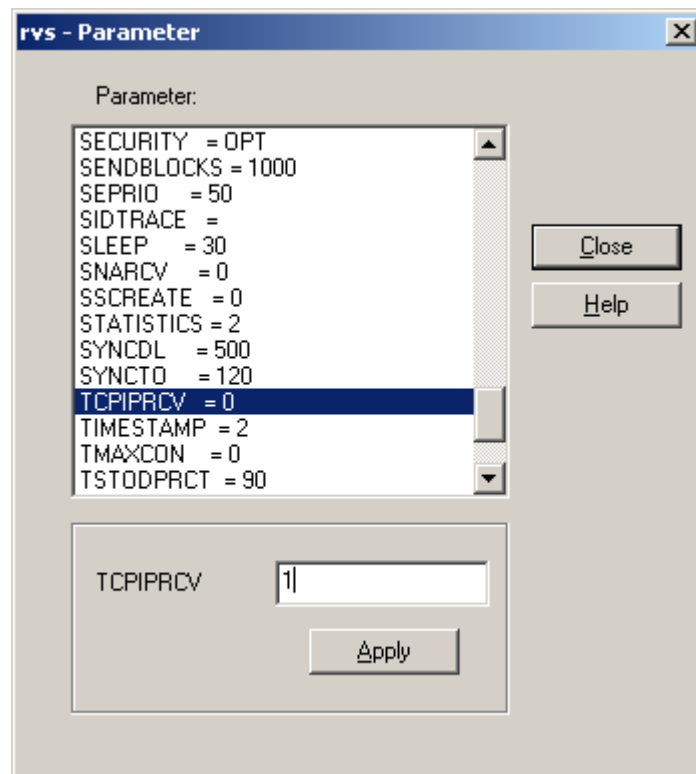
4.2 TCP/IP reception

Automatic reception of calls

If you want to be ready to receive via TCP/IP, rvsXP must start a TCP/IP receiver that waits for incoming calls. To set up rvsXP to receive calls automatically via TCP/IP:

- Execute the `Settings -> Parameters` menu command in the rvsXP Administrator.

The rvsXP Parameters dialog window for editing the parameters will open. In order for the TCP/IP receiver to start automatically start when the rvsXP Monitor starts, you have to set the value of the rvs **TCPIPRCV** parameter to 1:



- Double click **TCPIPRCV** in the parameter list.
- Enter the desired value **1** into the input field.
- Confirm your entry with **Apply**.
- Use **Close** to exit the parameter list.
- Activate the parameter by stopping and starting the rvsXP Monitor.

A message of the type

```
I: <TCPIP_READY> : TCP/IP Receiver ready to
accept incoming calls Port 3305
```

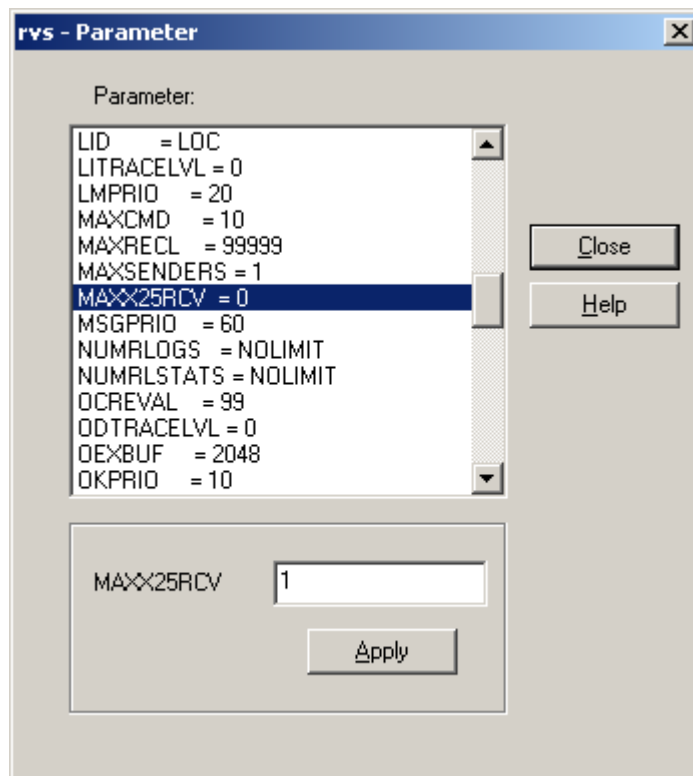
indicates your readiness to accept calls.

4.3 ISDN/X.25 reception

Automatic reception
of calls

If you want to be ready to receive via X.25/ISDN, rvsXP must start a receiver that waits for incoming calls on the X.25/ISDN line. To set up rvsXP to receive calls automatically via X.25/ISDN:

- Execute the `Settings -> Parameters` menu command in the rvsXP Administrator.
The rvsXP Parameters dialog window for editing the parameters will open. In order for X.25 and ISDN reception to start automatically when the rvsXP Monitor starts, you have to set the value of the `rvsXPMAXX25RCV` parameter to **1**:



- Double click **MAXX25RCV** in the parameter list.
- Enter the desired value **1** into the input field.
- Confirm your entry with **Apply**.
- Use **Close** to exit the parameter list.
- Activate the parameter by stopping and starting the rvsXP Monitor.

A message of the type

```
I: <ISDN_READY> : ISDN Receiver ready to accept
incoming calls No = 005361186751
```

indicates your readiness to accept calls.

5 rvsXP Monitor

This chapter describes how to start and stop rvsXP Monitor as well as how to customize the Monitor parameters for Windows XP.

rvsXP Monitor as
Windows service

Usually the rvsXP Monitor works as a system service. In order to install the rvsXP Monitor as a system service, it is necessary that you belong to the Windows XP Administrator group.

Windows offers various ways to open the list of `Services`. This list provides functions for managing and controlling services. Use the `Services` dialog to start, stop, pause or continue the rvsXP Monitor service.

Note: The term Service means a program that can be started from the operating system and works in the background.

5.1 Starting rvsXP Monitor

3 ways to start

You can start rvsXP Monitor from the operating system level, from `rvsXP-Administrator`, the tool for rvsXP operation, or directly via the `rvsTrayIcon`. Operation of rvsXP requires administrator user rights at the system level and in the rvsXP environment.

Depending on your choice during the rvsXP installation, the startup type for this service is `Automatic` or `Manual`.

Startup type Manual

The user must manually start the service if the `Manual` startup type was chosen. At first the status of the rvsXP Monitor service is `stopped` (represented by a blank entry in the status column).

To start the service:

- Highlight the `rvs Monitor` line.
- Click the **Start** button.

Setting to automatic

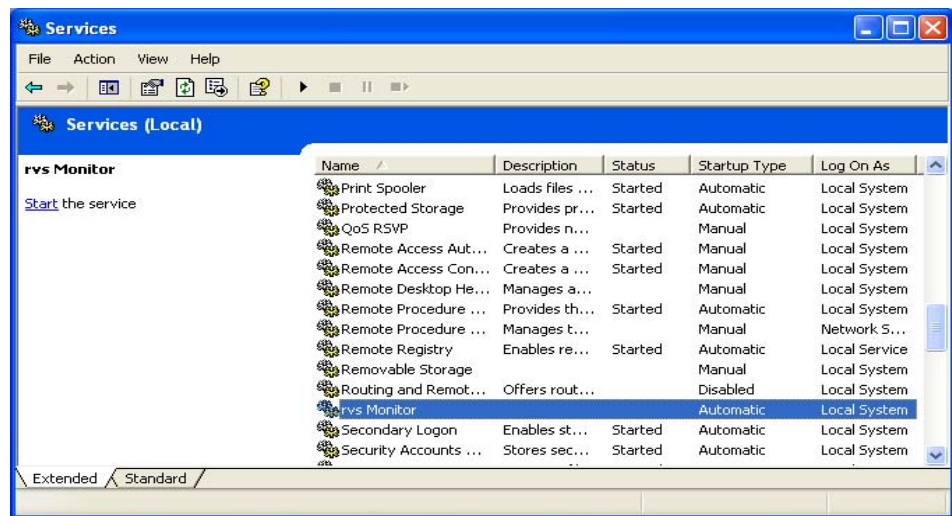
If you want to start rvsXP-Monitor every time the system starts, you can set the startup type to `Automatic`.

- Click the **Startup type** button and choose `Automatic`.

Starting
at system level

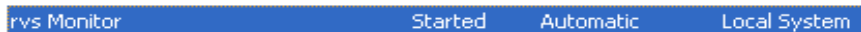
Carry out the following steps if you want to start rvsXP Monitor at the system level (XP 'classic Windows' style)

- Choose `Start -> Settings -> Control Panel`.
- `Start Administrative Tools -> Services`.
- Search the list of `Services` for `rvs Monitor` and select the service by clicking the appropriate line.



- Double click on the rvsXP Monitor entry. The rvs Monitor Properties (Local Computer) window opens.
- Click on the **Start** button. If the attempt is successful the rvsXP Monitor status will change to *Started*.

The entry is:



Starting with rvsTrayIcon

Carry out the following steps if you want to stop rvsXP Monitor with the rvsTrayIcon:

- Right-click the rvsTrayIcon.
- Execute the menu command `Start rvs Monitor`.

rvsXP Monitor starts. The rvsTrayIcon shows a green triangular arrow in a white circle if it is started.

Starting from the rvsXP environment

Carry out the following steps if you want to start rvsXP Monitor from within the rvsXP program environment while it is in operation:

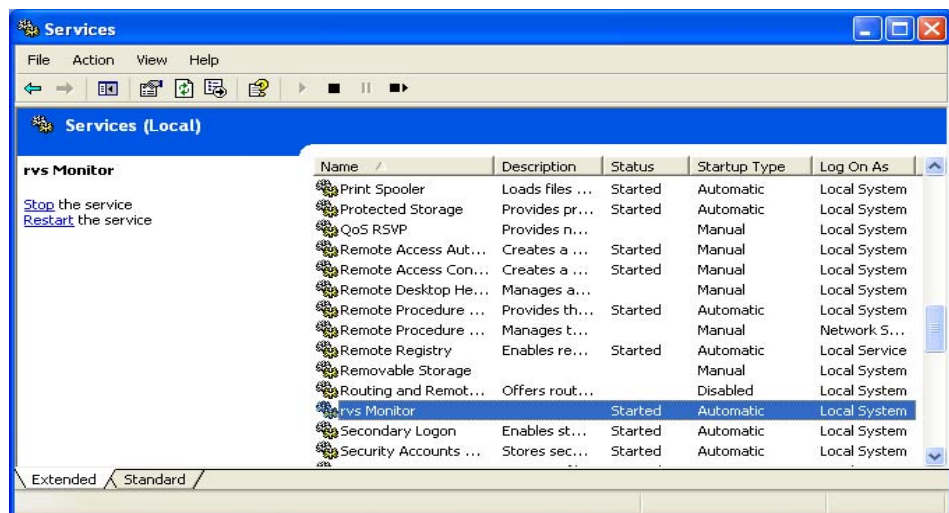
- Start the rvsXP Administrator program in the rvs folder.
- Execute the menu command `Monitor -> Start`. rvsXP Monitor starts. The rvsXP Monitor has been started if the stop symbol is active on the tool bar.

5.2 Stopping rvsXP Monitor

3 ways to stop You can stop rvsXP Monitor from the operating system level or from rvsXP-Administrator -, the tool for rvsXP operation. Operation of rvsXP requires administrator user rights at the system level and in the rvsXP environment.

Stopping at system level Carry out the following steps if you want to stop rvsXP Monitor at the system level (XP 'classic Windows' style)

- Choose Start -> Settings -> Control Panel.
- Start Administrative Tools -> Services.
- Search the list of Services for rvs Monitor and select the service by clicking the appropriate line.



- Click the **Stop** button. Windows XP will now try to end the service. If the attempt is successful the rvsXP Monitor status will change to " " (not started).

Stopping with rvsTrayIcon Carry out the following steps if you want to stop rvsXP Monitor with the rvsTrayIcon:

- Right-click the rvsTrayIcon.
- Execute the menu command Stop rvs Monitor.

rvsXP Monitor stops. The rvsTrayIcon shows a white cross in a red circle if rvsXP Monitor is stopped.

Stopping from the rvsXP environment Carry out the following steps if you want to stop rvsXP Monitor from within the rvsXP program environment while it is in operation:

- Start the `rvsXP Administrator` program in the `rvs` folder.
- Execute the menu command `Monitor -> Stop`.
or
- Launch the `das Stop rvs Monitor` program in the `rvs` folder.
rvsXP Monitor stops. It has been stopped if the stop symbol is not active on the tool bar.

5.3 Customizing rvsXP Monitor parameters

What can be customized? The Monitor parameters determine the operational mode of the rvsXP Monitor, the main component of the rvsXP system. By customizing the Monitor parameters you can modify the rvsXP Monitor in accordance with your system environment and your own requirements. You should customize the Monitor parameters in order to

- minimize the load on your system and in order to enhance the performance (parameters **OCREVAL**, **OEXBUF**, **RECVBLOCKS** and **SENDERBLOCKS**; for more information please see the Appendix "The rvs® parameters").
- control the Monitor Log Book display
- make optimal use of data line quality
- monitor communication processes with other stations
- manage the size and number of the log files.

The rvsXP database, which works as a central information source, provides the rvsXP Monitor with all the information required for the tasks to be executed.

When the rvsXP database is initialized the initialization program first creates a parameter table for the Monitor parameters with default values which determine the default behavior of rvsXP Monitor.

Start settings You can modify these values by editing the `C:\rvs\init\rdmini.dat` file, which is the source for modification of the parameter table in the rvsXP Monitor start phase. Changes will only take effect when the Monitor is started again if you modify the file while rvsXP Monitor is in operation. Dynamic modification of the monitor parameters cannot be achieved in this way.

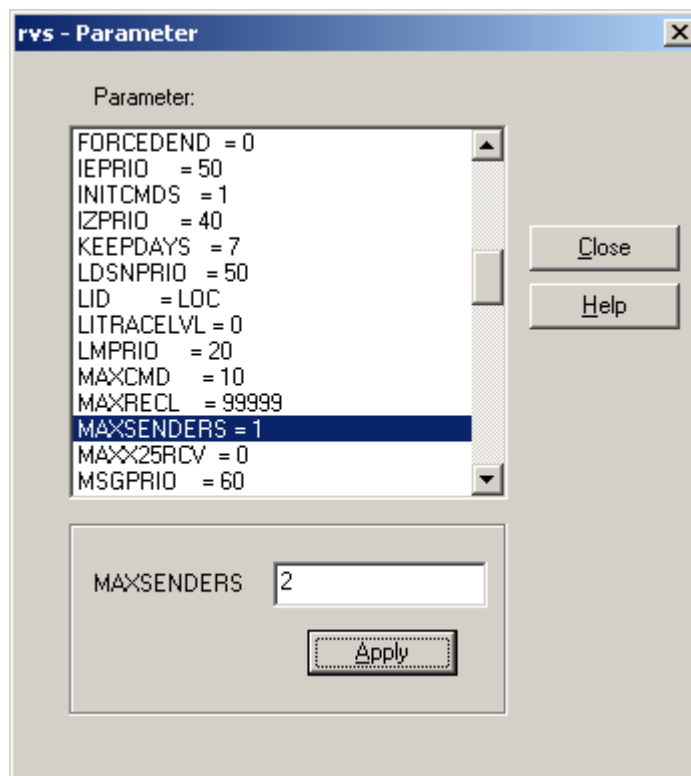
Dynamic modification

With rvsXP Administrator you can dynamically modify the parameters during Monitor operation. In order to do so you have to enter individual operator commands.

Changes during operation Carry out the following steps if you want to change a monitor parameter without stopping rvsXP Monitor:

- Click Settings -> Parameters in the rvsXP Administrator menu tree.

The rvsXP Parameters dialog window for editing the parameters will open.



- Double click the parameter that you want to modify.
- Enter the desired value parameter value into the input field and click **Apply**.
- Use **Close** to exit the parameter list.

A complete list and description of all the monitor parameters can be found in the Appendix "The rvs® parameters".

Example:

```
MAXSENDERS 2
```

With this example you indicate that a maximum of 2 senders transmit data at the same time.

- Confirm your entry with **Apply**.
- Use **Close** to exit the parameter list.

This operation transfers the command into the rvsXP database. When restarted, rvsXP Monitor will read this command and use the modified parameter.

Stop and start rvsXP Monitor to activate the parameter right away.

Static modification

Changes with restarting

You can also customize the monitor parameters by making entries in the `rdmini.dat` file. It is read every time rvsXP Monitor restarts and the orders and commands it contains are executed. Carry out the following steps if you want to permanently modify one or more monitor parameters:

- Open the `C:\rvs\init\rdmini.dat` file. To do so click the `Monitor Parameters` icon in the `rvs` folder.

The file contains predefined operator commands rvsXP Monitor executes each time it starts. Commands for modifying the monitor parameters begin with `setparm`, comment lines with `*`.

- Edit the file in accordance with your requirements.

A complete list and description of all the monitor parameters can be found in the Appendix "The rvs® parameters".

- Save the file and close the text editor.
- Display the `Monitor Log` window in the `Log -> Monitor of the rvs tree in the rvsXP Administrator`.
- Stop and start the rvsXP Monitor.
- Read the action messages: They begin with `A:`.

Each `setparm` command in the initialization file triggers a parameter modification when the rvsXP Monitor is restarted.

Note: Statically modify only those parameters which are to permanently differ from the rvsXP standard setting.

Example

You want to monitor a transmission line temporarily in order to trace errors. In order to do so, modify the **LITRACELEVEL** parameter to **3** using the `Settings -> Parameter` command and reset it again using this command after you are done with monitoring.

Please refer to the Appendix "The rvs® parameters" for more detailed information, in particular on additional operator commands and optimizing rvsXP.

6 How to work interactively with rvsXP

Function overview The following functions are available for the interactive usage of rvsXP:

- Create send entries for the transmission to one or more recipient at a definite time
- display the status of your transmissions.
- Delete send entries, as long as rvsXP did not begin to process them
- Create resident receive entries in order to define how to save or handle arriving files
- Create send job entries in order to trigger actions if files are successfully sent or could not be sent
- Delete or modify resident receive and job send entries.

6.1 Starting rvsXP-Administrator (rvsmgr.exe)

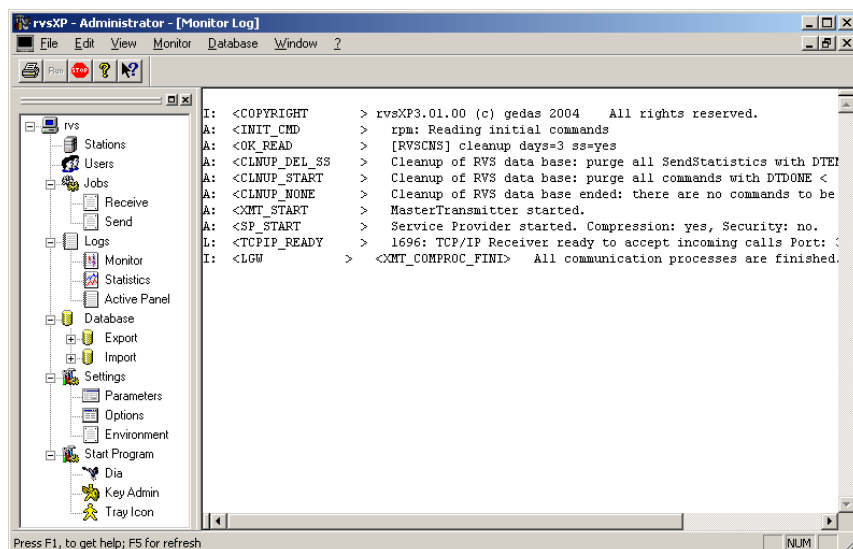
Logging on as administrator (system and rvsXP)

rvsXP Administrator is used for configuring, managing and operating rvsXP. It comprises several detail windows. The configuration requires administrator user rights at system level and in the rvsXP environment.

Start the rvsXP Administrator program

- by launching it in the rvs folder or
- with the rvsTrayIcon or
- use Start -> All Programs -> rvs -> rvsXP-Administrator.

The rvsXP-Administrator opens with the Monitor Log window displaying the current rvsXP Monitor activities.



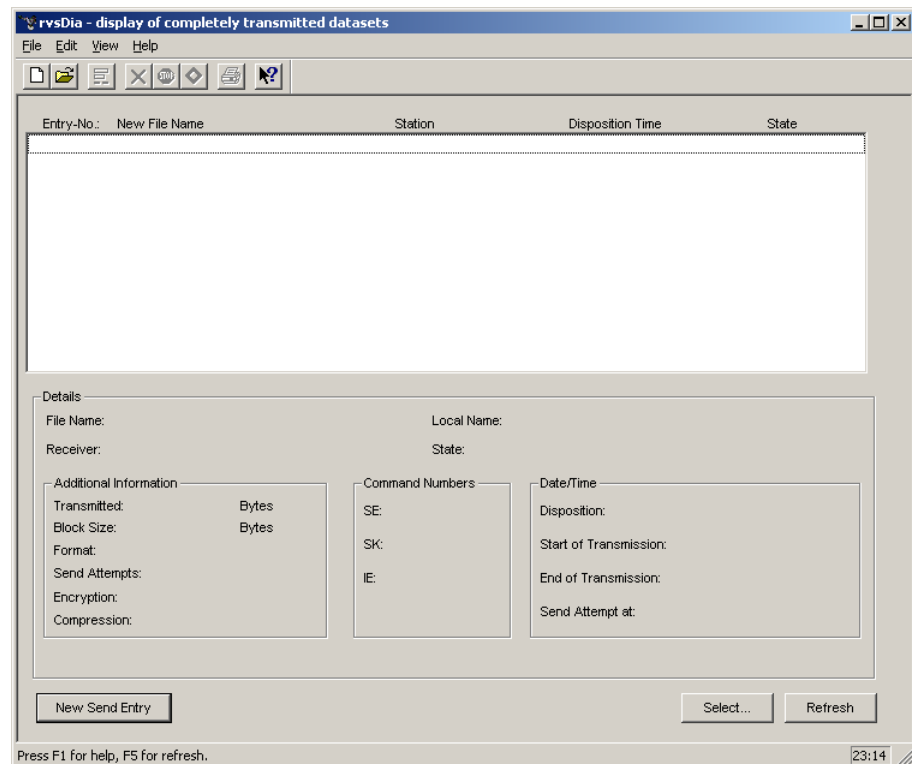
6.2 Starting rvsXP (rvsdiant.exe)

Requirements You need a system which has been set up for operating rvsXP and the required access rights.

Start the rvsXP program

- in the rvs folder
- in the rvsXP Administrator using Program -> Dia or
- with the rvsTrayIcon.

The rvsXP window opens with display of not transmitted datasets.



The window displays the transmission information in four different display modes.

Display modes

- Send
- Reception
- Completely sent
- Completely received

Window areas Depending on the rvsXP display mode, the **entry list** will display the list of files which have been sent or received.

The **Details** area provides you with more detailed information on the currently selected file: file name, sender or recipient, local name or original name and status of the file.

In the **additional information** you can see the currently transmitted data quantity, block size, file format and the number of dialing attempts (only for sending).

The respective successive **command numbers** for send entries (SE), send command (SK) and information entries (IE) identify the individual transmission steps. These numbers can also be found in the rvsXP Log Book (C:\rvs\db\rlog.log file).

The **Date/Time** area indicates the transmission times and attempts logged by rvsXP Monitor.

Status display

The **Status** field displays the progress of the send order selected or that of the order arriving.

Sending In the **Send** or **Completely sent** display modes the status field shows three events SE SK QE.

Receiving In the **Receive** or **Completely received** display modes the status field shows three events IE IZ QS.

Sample send order: Status (qq-)

SE = q; SK = q; QE = *-*

| Abbr. | Event status | Possible values = status chain |
|-------------------|--|--------------------------------|
| Send orders | | |
| SE | Send entry | q, f, p, e |
| SK | Send command | -, q, f, a, i, p, e |
| QE | Acknowledgement receipt, sent by recipient | -, e |
| Reception arrival | | |
| IE | Information entry (about incoming information) | q, f, p, e |
| IZ | Information delivery (to the recipient) | q, f, p, e |
| QS | Acknowledgement transmission back to sender | -, q, f, a, i, p, e |

Note: Each event normally runs through the status chain in the order of possible values indicated. Some statuses have such a short duration that they are only briefly visible.

The meaning of the values

- no event
- q queued, not yet processed by Monitor
- f forwardable, waiting for transmission process
- p pending, waiting for completion
- e ended
- a active
- i in transit, transmitter currently processing order

In addition, every event may also have these statuses:

- h held by the system or operator
- s traffic with the destination suspended at present

Exiting rvsXP

It is always possible to exit

You can exit rvsXP at any time, even if all of the send orders have not been completed or you have not yet saved several of the files which have been received. rvsXP Monitor continues to process your send orders and places arriving files into your user directory. The next time you start rvsXP you can display the progress of your send orders and any files which may have arrived.

- To terminate rvsXP choose `File -> Exit`.

6.3 How to send and receive files with rvsXP

Overview This chapter describes how to interactively send and receive files with rvsXP. First, it will be explained how you can create send requests. The next chapter describes how you can display sent and received transmissions. The last chapter shows how to edit send and receive data sets.

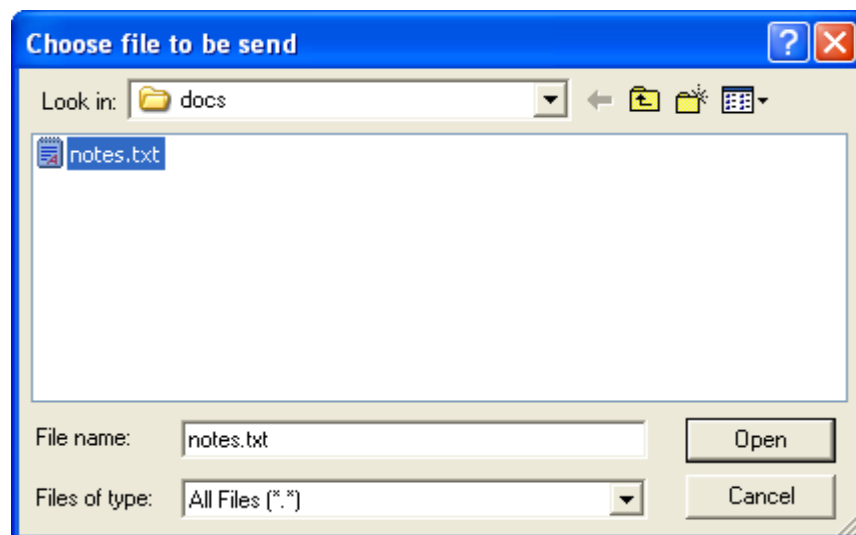
6.3.1 Creating send orders

To send a file to a business partner:

- Choose `Edit -> New Send Entry` from the menu.

You can also execute this command by using **F2** or by clicking **New Send Entry**.

The `Choose file to be sent` dialog box opens.



- Select the file which you would like to send to your partner and confirm your selection with **Open**.

Example `C:\docs\notes.txt`

The `New Send Entry` dialog opens and displays the file name selected in the `File` field.

- Manually enter the station code for the receiving station in the `Station ID` field.
or:
- Choose `Select...` to open the `Select Station` dialog and select the ID.

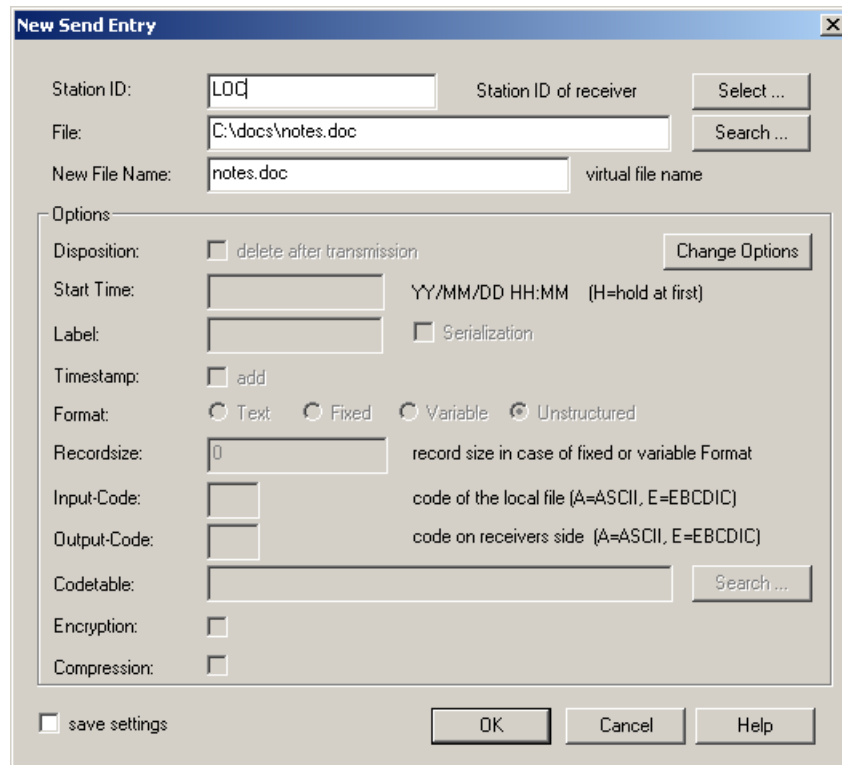
Example LOC

- To send the file to your partner under a different name enter the new file name into the New Name field.

It is the file name that is used for the ODETTE transfer and has to be agreed upon by both sides. The name length must not exceed 26 characters.

- Confirm your entries with **OK**.

The file size limit is at the moment 10 GB.



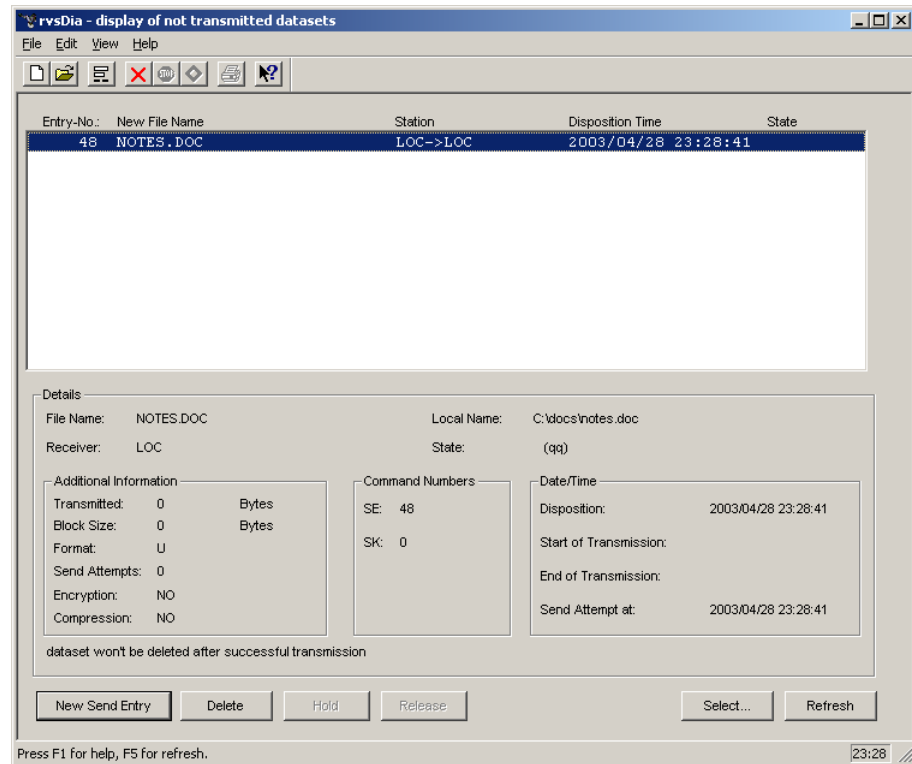
Options

At the bottom of the New Send Entry dialog you will find various options that can be useful when sending a file. Click the Change options button if you wish to change any of these options. (For details refer to the table below.)

| | |
|---------------------------------------|---|
| Options | Transmission settings and actions |
| Disposition | Click the <code>delete after transmission</code> checkbox to have files deleted on your local computer after successful transmission. Otherwise it would not be deleted. |
| Start Time | rvsXP sends files immediately. If this is not desired, you can define the send time. H: the file will be set to hold status until you release it. Release occurs in the rvsXP -> display of not transmitted datasets -> Edit -> Release window. |
| Start Time (continued) | Exact time: If you want that the file to be sent your partner at an exact time you should indicate the start time in the format YY/MM/DD HH:MM or YYYY/MM/DD HH:MM. Example: 2004/05/28 10:28 no entry (blank): the file will be sent immediately |
| Station ID + Serialization | If you activate these two options, the files would arrive at their destination in a specific order. All files sent in the same group must have the same ID (name). The Serialization checkbox must also be enabled. For more information on Serialization please refer to Chapter 6.3.2 "Serialization". |
| Format | Choose the format of the file to be transmitted from one of the following: Text: a string of ASCII characters Fixed: fixed record length Variable: variable record length Unstructured: binary file none: system format (i.e. unstructured for rvsXP) |
| Record size | Specify the fixed record size the receiver is to interpret the file with for files in the Fixed format. The record length is the number of characters before CR/LF (MS Windows systems) or LF (UNIX systems). Specify the maximum record size for files of the Variable format. |
| Record format | Intern: Windows XP does not support the fixed and variable mainframe formats; to send files in one of these formats you have to convert the files with the rvsut2fv utility before sending. Please, see the Reference manual, Part III, Utilities. This feature can also be performed by rvsXP, if you use the option Native Text or Native binary . Text: The file, which should be sent in format fixed or variable must already exist in this format. It means, that this file contents only ASCII characters. Each line must end with CR/LF. Each line will be converted to one record. In this case, you do not need to use the utility rvsut2fv. Native Text: The file in format text will be converted in fixed or variable format by rvsut2fv before sending. Native binary: The file in format binary will be converted in fixed or variable format by rvsut2fv before sending |
| Input, Output Code; Code Table | For detailed description of these options and different file formats please the read the Chapter 9 "Code conversion". |

| | |
|--------------------------------|--|
| Options | Transmission settings and actions |
| Encryption, compression | To send a file with encryption and/or compression you must enable the appropriate options. For further information please see the Chapter entitled 8 "Encryption configuration: Key administration". For more information on compression please refer to Chapter 8.8 "Configuration of offline compression". |

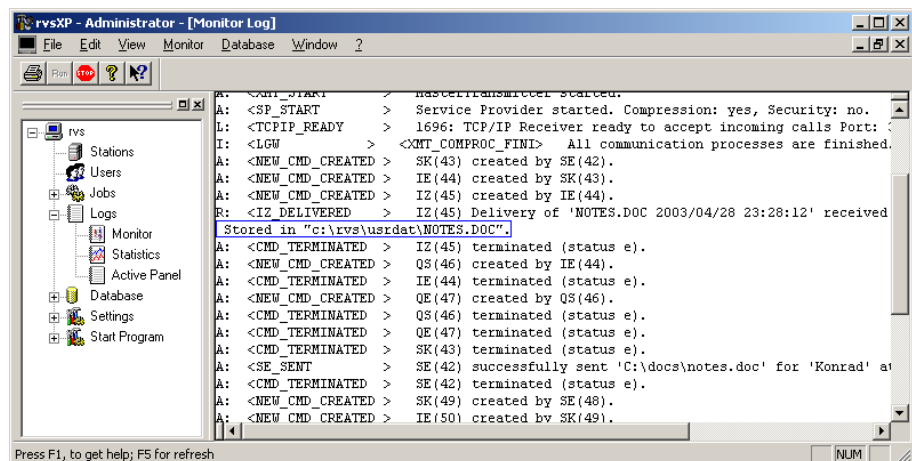
rvsXP then writes the send entry for transmission of the file to the rvsXP database. The send order has been completed. rvsXP lists the job in the list of files waiting to be sent.



rvsXP Monitor reads the send order at the next read cycle and executes it.

In the example rvsXP Monitor recognizes through the LOC station ID that the own station is the target and copies the file to the c : \rvs\usrdat folder.

Note: You can view the activities of rvsXP Monitor in the rvsXP Administrator in the Monitor LOG window:



6.3.2 Serialization

Purpose
of serialization

Sometimes it may be important that a number of files arrive in a specific order at their destination to make sure they will be processed in proper sequence.

Even though rvsXP will start processing send entries in the order in which they were submitted, there is no guarantee that the data sets will arrive in just this order (the line may break down while one file is being sent and another file may be transmitted before restart of the interrupted transmission is attempted).

To enforce the proper sequence,

- specify a **label** when creating the send entry for the first file in your series (display of not transmitted datasets, **New Send Entry** button)
- use the same label and enable the **Serialization** checkbox for all subsequent files
- create all send entries linked by the same label in the correct sequence.

rvsXP will not start a serialized transfer until it has received the acknowledgement (EERP) for the previous transmission. This will ensure transfer in the specified order.

6.3.3 Displaying send requests and received files

Displaying send orders

Two display choices

You can choose between two lists to display the send orders:

The list of send orders which have **not yet been completed** displays those files for which you have chosen to hold transmission or which are waiting in the queue of files to be transmitted.

- Choose **View -> Send** to display the send orders which have not yet been completed.

The list of **completed send orders** displays those files which have been successfully transmitted and acknowledged by the recipients.

Note: You can, however, view completely sent files only if the rvsXP **CMDDELETE** parameter has the value **0**. **CMDDELETE=0** means that old, successful transfers remain in the rvsXP database and can also be viewed. If this parameter in rvsXP has the value **1**, only uncompleted transfers are displayed. All successful transfers are deleted from the rvsXP database in order not to overload the database unnecessarily.

The **CLEANUP** parameter finally determines how long such finished jobs are to be retained in the database if **CMDDELETE = 0**. This does however place a burden on the rvsXP database and we therefore recommend that by default you set the **CMDDELETE** parameter to 1. If you want an overview of files sent and received, or if you require this information for archiving purposes, use the rvsXP `$RVSPATH3\db\rlstat.log` statistic file.

- Execute the menu command **View -> Completely sent files** in order to display the list of send orders which have been completed.

Refreshing the view

If you want to update the list display, then do the following:

- Click the **Refresh View** button or switch to a different display mode and then back again.

Displaying received files

Two display choices You can choose between two lists to display the received files:

The list of **files being received** displays those files which have not yet been completely transmitted. Your station and the partner station are presently transmitting the files. To display the files announced:

- Choose **View -> Receive**.

The list of **files which have been completely received** displays those files which have been successfully transmitted to your station. At this stage your station has acknowledged reception. To display completed send orders:

- Choose **View -> Completely received**.

³ For more detailed information on the **\$RVSPATH** parameter see Chapter 1.6 "Representation means".

Refreshing the view

If you want to update the list display, then do the following:

- Click the **Refresh View** button or switch to a different display mode and then back again.

6.3.4 Handling files to be sent and received

rvsXP and
interactive operation

This chapter describes how you can handle files with rvsXP if you are using an interactive application.

Holding a send order

Since rvsXP operates asynchronously there is always a certain time span between your send order and file transmission. This depends on the rvsXP Monitor load and settings. During this time you can still put a hold on a send order and modify it.

To hold one or more send orders:

- Use the mouse to click on a single send order.

Note: To find out how to limit the display to the send orders placed on hold or certain file types, please refer to Chapter “Selecting display criteria” on page 82.

- Click the **Hold** button or choose `Edit -> Hold` from the menu.

Modifying a send order

To modify a send order:

- Use the mouse to click the send order that you want to modify.
- Execute the menu command `Edit -> Update`.

The `Change Send Entry` dialog opens. In this dialog you can modify all the fields for the send order.

- Carry out the desired changes.
- Confirm your entries with **OK**.

The send order has been modified. rvsXP displays the modified order in the list of files which have not yet been sent.

Temporarily hold

If you want to create a send order and then release it later, you should carry out the following steps to put a temporary hold on it:

- Click **Change Options** in the `New Send Entry` dialog.
- Enter the letter **H** in the `Start Time` field.
- Confirm the entry by pressing **OK**.

Releasing a send order

If you have put a hold on a send order, you can also release it for subsequent execution. To release the execution:

- Use the mouse to click on a send order.

Note: To find out how to limit the display to the send orders placed on hold or certain file types, please refer to Chapter “Selecting display criteria” on page 82.

- Release the orders selected with **Release** or the `Edit -> Release` menu command.

Restarting a send order

Automatic repetition rvsXP repeats an unsuccessful transmission attempt in accordance with the time specified by means of the **DTCONNnn** parameter. You can avoid this waiting period and immediately start a new transmission attempt. rvsXP Monitor then treats the send order as though you had created it for the first time. To restart one or more send orders:

- Use the mouse to click on a send order.
- Click the **Hold** button or choose `Edit -> Hold` from the menu.
- Release the orders selected with **Release** or the `Edit -> Release` menu command.

Deleting a send order

Removing before transmission You can remove a send order from the database. If you remove a send order which has the status `forwardable` or `held`, then your partner will not receive the file which was intended for him.

No revoking after transmission A send order that has already been completed remains executed. It is not possible to revoke the order. You can only remove the entry from the rvsXP database. This may make sense in some cases in order to improve management of transmission activities.

Note: The rvsXP Monitor constantly purges the rvsXP database of send entries which have been completed since a particular period of time (**KEEPDAYS** parameter). Thus you do not have to remove the send orders in order to clean the database.

To remove a send order from the database:

- Double-click the send order you want to remove.

Note: To find out how to limit the display to the send orders placed on hold or certain file types, please refer to Chapter “Selecting display criteria” on page 82.

- Click the **Delete** button or execute menu command `Edit -> Delete`.

Releasing or deleting an acknowledgement

You must manually release automatically created acknowledgements if you have specified the **HOLD** or **HOLDDIMMED** parameters for processing **EERP OUT** acknowledgements for a partner station. Received files are considered as having been completely received only after you have released or deleted the acknowledgement.

- Releasing EERP To release an acknowledgement for a received file,
- choose the received file and execute the menu command `Edit -> Release EERP`.
- Deleting EERP To delete an acknowledgement for a received file
- choose the received file and execute the menu command `Edit -> Remove EERP`.

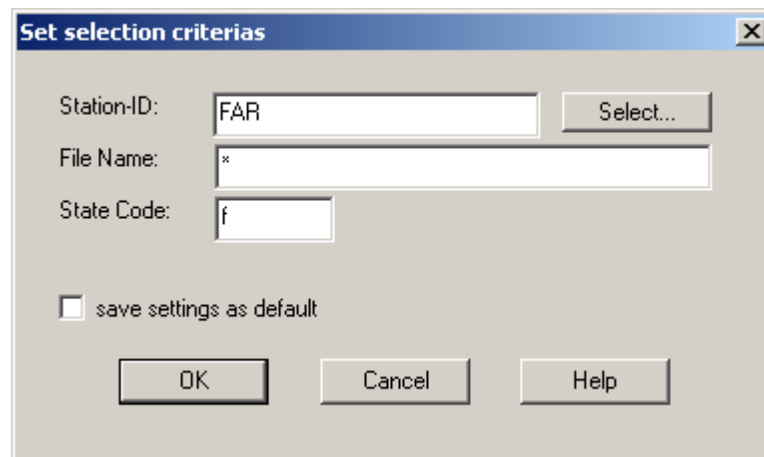
6.3.5 Selecting display criteria

You can select certain kinds of send orders for display in the entry list. You can use the station code, file name and status of the orders as selection criteria.

To set the display criteria for send orders:

- Execute the menu command `View -> Select`.

The `Set selection criteria` dialog opens.



- Enter the station ID for the station(s) to be selected.
or:
- Choose `Select...` to open the `Select Station` dialog and select the ID.
- Enter the criterion for the files to be selected in the `File Name` field.
- Enter the status code to be selected in the `State Code` field.

This entry is optional.

Note: You can use the wildcard characters * and ? or a combination of * and ? for entering selection criteria.

* stands for any - or even empty - character string

? stands for one character of any value

Use one of the following characters as the status code:

| | |
|---|--|
| y | no event |
| q | queued, not yet processed by rvsXP Monitor |
| f | forwardable, waiting for transmission process |
| p | pending, waiting for completion |
| y | ended |
| a | active |
| i | in transit, transmitter currently processing order |
| d | deleted |
| h | held by the system or operator |

- If you want to repeatedly use this setting then enable the `Save Setting as Standard` check box.
- Confirm the entry by pressing **OK**.

6.4 Administration with rvsXP

This chapter describes how to administrate resident receive entries, jobs in `rvsXP Administrator` after send attempts and according to user lists.

6.4.1 Resident receive entries

When a file is received the `rvsXP Monitor` stores it in the `$RVSPATH\usrdat` folder. Then it logs reception in the `rvsXP Administrator` and in the `$RVSPATH\db\rlog.log` `rvsXP` log book.

Action during reception You can trigger other actions when receiving files by generating resident receive entries `RE` in the `rvsXP` database. A `(RE)` specifies a program that is to start when a particular file or selection of files has arrived.

Example When a file with the extension `TXT` (*.TXT) arrives, the batch program `kopiere.bat` should start.

This batch program copies the received file from the `C:\rvs\usrdat` user data folder to the `D:\temp` folder.

```
copy ?DSN? d:\temp
```

DSN stands for `DataSetName` and states where the file was placed in the local file system. `?...?` are `rvsXP` replacement markers and `rvsXP` replaces the `?DSN?` character string by the local file name of the received file.

Example

```
copy c:\rvs\usrdat\konto1.txt d:\temp or
copy c:\rvs\usrdat\liste.txt d:\temp
```

Another example is to be found in the file:

```
C:\rvs\system\resentr.bat.
```

Note: You must not specify a binary executable file (such as `*.exe`) as the batch program. It should be a readable and parseable file. Ensure that your batch program is sensibly structured because it is not checked by `rvsXP`.

Troubleshooting tip Each time you execute an `RE` a copy of the batch file, in which the replacement markers `?...?` are replaced by your actual values, is saved in the `C:\rvs\temp` folder. The copies are named `reXXXXXX.bat`. `XXXXXX` stands for a consecutive number. If the behavior of a `RE` is incomprehensible to you, you should read in the last generated copy of the batch program (`reXXXXXX.bat` file) just which commands have been executed. The number of the jobs (`XXXXXX`) must not exceed 32 000 (Windows XP restriction).

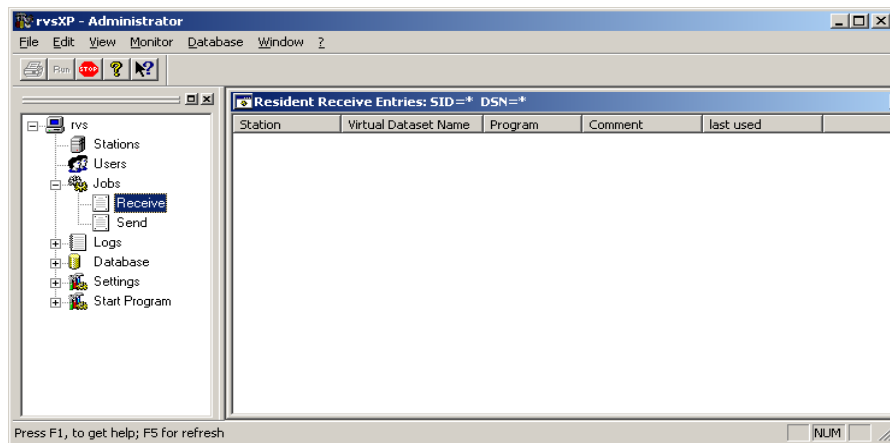
Note: If you want rvsXP not to save copies of the batch programs (reXXXXXX.bat file) in the C:\rvs\temp folder you should add the following command to your batch file: DELETE ?DSNTEMP?.

For more information about rvsXP replacement markers ?...? and how to define resident receive entries using the rvs® batch interface (rvsbat) please refer to the "rvs® Reference Manual".

Displaying resident receive entry

Choose Job -> Receive in the rvsXP Administrator to display resident receive entries.

The Resident Receive Entries dialog opens. It displays a list of the previously created resident receive entries. The list is initially empty.

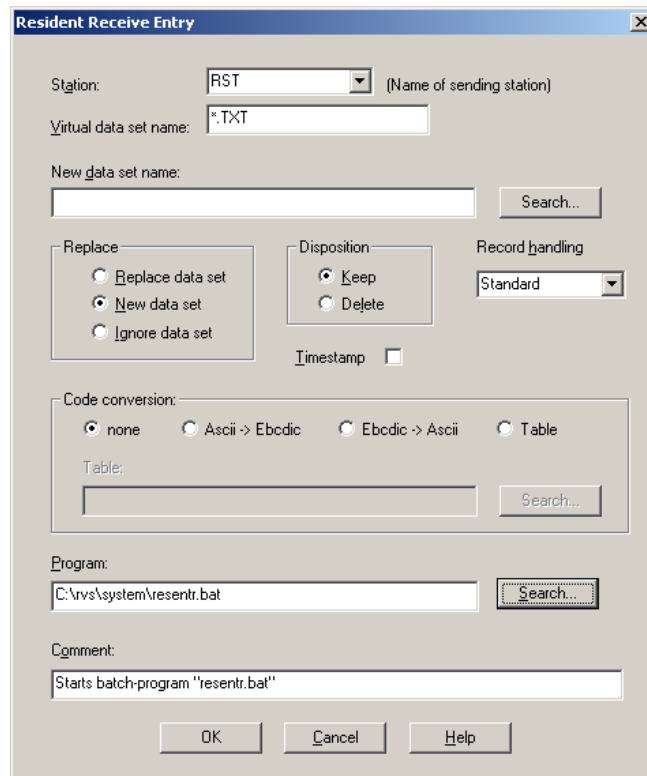


Creating resident receive entries

If you want to start a program when a file arrives, do the following:

- Choose Edit -> Add.

The Resident Receive Entries dialog opens.

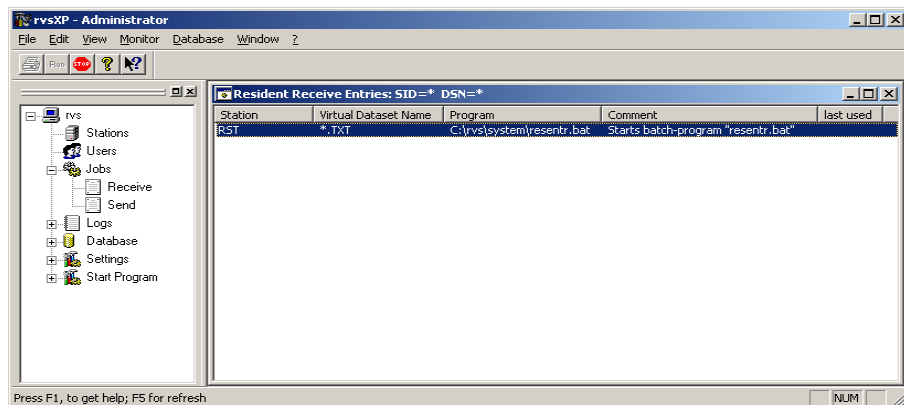


- Enter the station from which you expect to receive the file. (The Station contains * by default. In other words, RE applies to file reception from all stations.)
- Enter the file or the file specification for which the RE is to apply in the Virtual data set name field. Virtual data set name is the file name that is used for the ODETTE transfer and has to be agreed upon by both sides. The name length must not exceed 26 characters.
- If you want to change the file name after it has arrived, enter the new file name in the New data set name field.

Disposition: determines whether the received file is deleted or kept after reception and execution of the RE.

- When you decide upon **Disposition, keep** and the C:\rvs\usrdat folder contains a file with the same name, you can enable the **Replace** option to choose whether to
 - replace the existing file,
 - create a new file with a unique name (default) or
 - ignore the file being received.
- Specify in the Record handling field whether the received file is to be stored as a text file with a line feed after every record. This applies to files that are received in **Fixed** or **Variable** format only. Select **Text** here if the line feed is to be switched on.

- In the Code Conversion field you may specify whether a code conversion is to take place. Select a special conversion table by means of **Search**. For default conversion (ASCII:ISO 8859 Latin 1 and EBCDIC:CECP 037) click **ASCII-EBCDIC** or **EBCDIC->ASCII**.
 - Use **Search** to select the batch program that should start when the file arrives.
 - Confirm your entries with **OK**.
- rvsXP Administrator writes the new entry to the rvsXP database and displays it in the list of resident entries.



Editing resident receive entries

To edit a resident receive entry:

- Select the entry to be modified in the Resident Receive Entries window.
- Execute the menu command **Edit -> Update**.

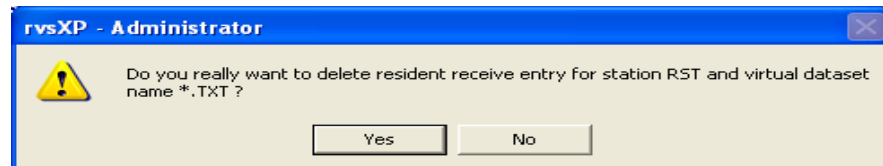
The Resident Receive Entries dialog opens and displays the properties of the entry.

- Edit the entry properties and confirm your entries with **OK**.

Removing a resident receive entry

To remove a resident receive entry:

- Select the entry to be deleted in the Resident Receive Entries window.
- Execute the menu command Edit -> Delete.
rvsXP will ask you to confirm your delete command.



If you are sure you want to delete the entry, confirm the command with **Yes**. With **No** you cancel the operation without deleting.

6.4.2 Job start after send attempt

You can trigger actions, after rvsXP has attempted (with or without success) to send files by generating resident job start entries (JS). A job start entry specifies the program that is to start after the first attempt to start a particular file or a selection of files.

Example: The batch program `sendjob.bat` is to start after the third unsuccessful attempt to send a file. This example is also to be found in the `C:\rvs\system` folder.

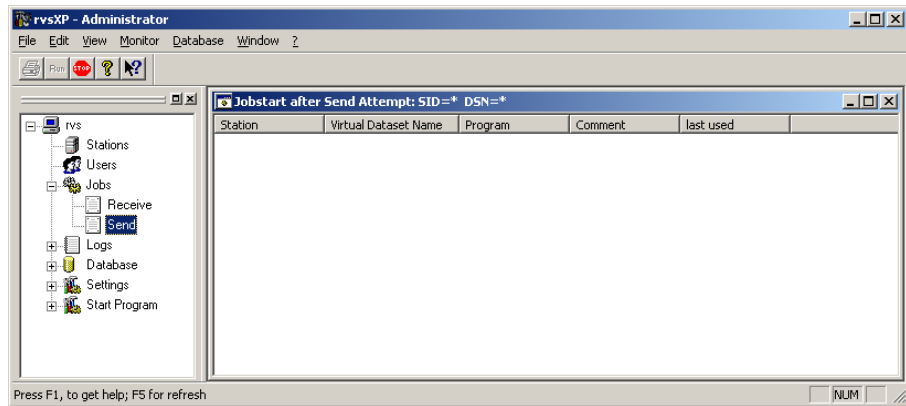
The modified copies of the batch program are also saved in the `$RVSPATH\temp` folder, exactly as is the case with RE. The copies are stored under the name `jsXXXXXX.bat`, where XXXXXX stands for a consecutive number. You can examine these files for correctly executed commands for troubleshooting purposes. The number of the jobs (XXXXXX) must not exceed 32 000 (Windows XP restriction).

Note: If you want rvsXP not to save copies of the batch programs (`jsXXXXXX.bat` file) in the rvsXP temp folder you should add the following command to your batch file: `DELETE ?DSNTEMP?`.

Displaying job start after send attempt

- Choose Job -> Send in the rvsXP Administrator to display resident job start entries (JS).

The Jobstart after Send Attempt window opens. It displays a list of the previously created JS's. The list is initially empty.

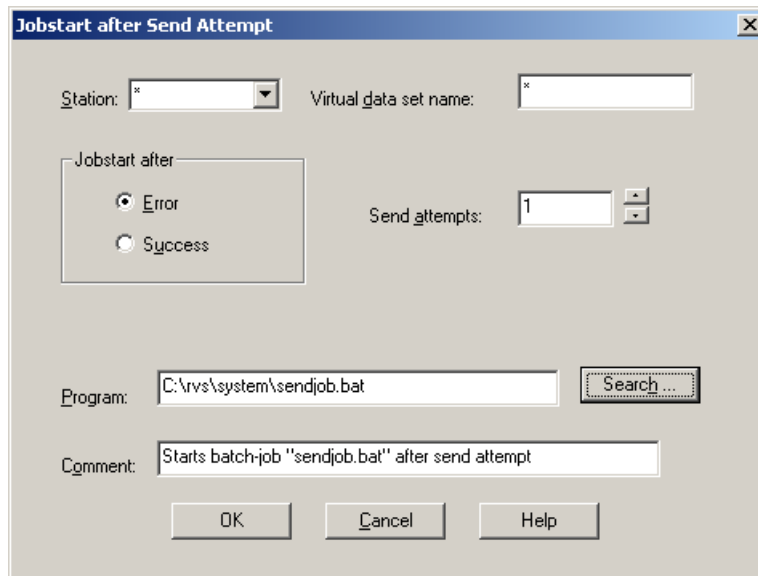


Creating a job start entry

If you want to start a program after an attempt to send a file, do the following:

- Choose Edit -> Add.

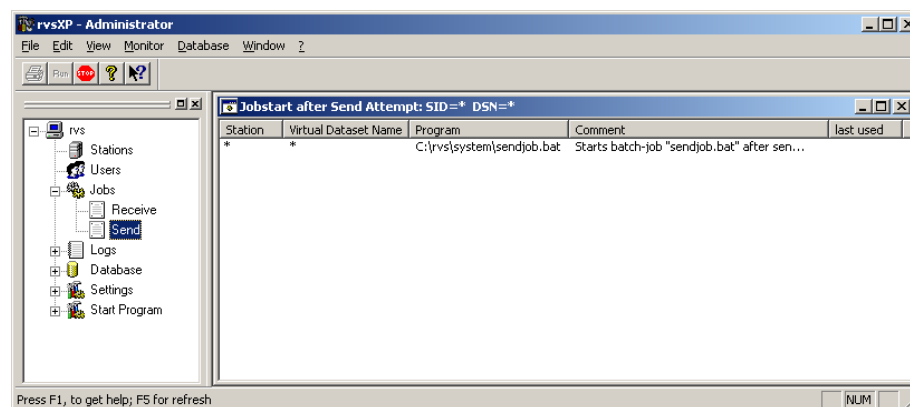
The Jobstart after Send Attempt dialog window then opens.



- Enter the station you want to send the file to.

- Enter the file or the file specification for which the JS is to apply in the **Virtual Data Set Name** field. Virtual Data Set Name is the file name that is used for the ODETTE transfer and has to be agreed upon by both sides. The name length must not exceed 26 characters.
- Use **Search** to select the batch program that should start after the attempt to send the file.
- If you want the program to start in the event, for example, five unsuccessful attempts, select **Error** and specify the number of unsuccessful attempts after which the program it to start under **Send Attempts**. The send job will remain in the rvsXP database, and rvsXP will continue attempting to send the file to the partner. Your script (program) will be launched after each fifth unsuccessful attempt, for example. rvsXP only calls your script and does not perform any plausibility check.
- Confirm your entries with **OK**.

rvsXP Administrator writes the new entry to the rvsXP database and displays it in the list of resident entries.



Editing a job start entry

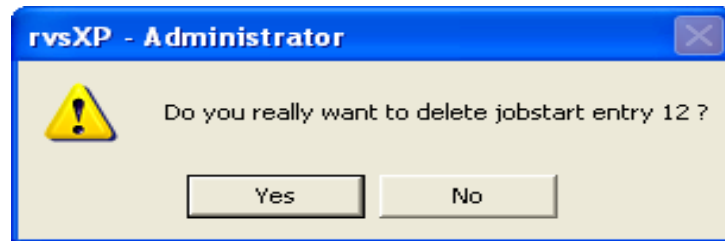
If you want to modify a job start entry:

- Select the entry to be modified in the Jobstart after send attempt window.
- Execute the menu command Edit -> Update.
The Jobstart after send attempt dialog opens and displays the properties of the entry.
- Edit the entry properties and confirm your entries with **OK**.

Deleting a job start entry

If you want to delete a resident send entry:

- Select the entry to be deleted in the `Jobstart` after `Send Attempt` window.
- Execute the menu command `Edit -> Remove`.
 rvsXP will ask you to confirm your delete command.



- If you are sure you want to delete the entry, confirm the command with **Yes**. With **No** you cancel the operation without deleting the entry.

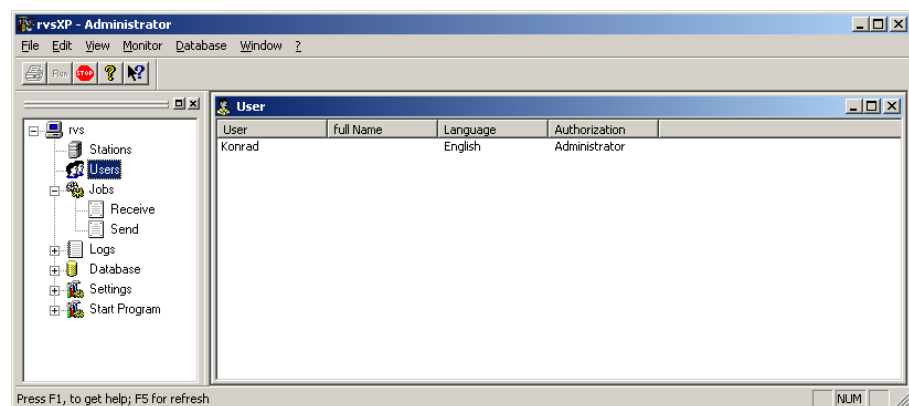
For more information about the rvs[®] batch interface (`rvsbat`) and how to define alternative resident job start entries using this interface please refer to the "rvs[®] Reference Manual".

6.4.3 User administration

Only those users may work with rvsXP who are registered in the rvsXP database. In order to display or modify user data:

- Open the user table in the `rvsXP-Administrator` with `User`.

The user table contains at least the table entry for your own data.

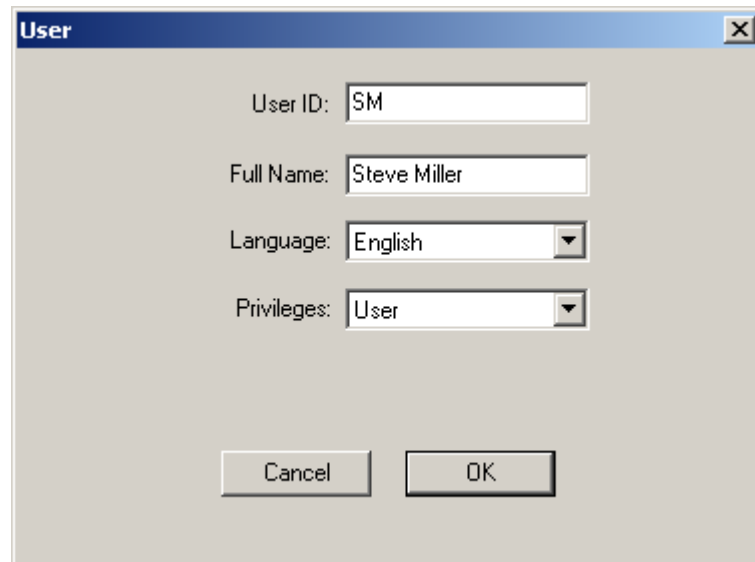


Sorting Use the mouse to click the column header to sort the table by any column.

New user (User dialog)

To set up a new user:

- Choose **Edit** -> **Add**. The **User** dialog opens and displays the User Table parameters.



- Enter the user identification (**UID**).
Note: Heed upper- and lowercase spelling. The spelling of your rvsXP user ID must match that of the Windows XP user name. To ensure this copy the exact spelling of your Windows XP user name from the Windows XP User Administration.
- Enter the complete user name.
- Select the language for screen dialog.
- Use the mouse to click on the user access authorization.
- Confirm your entries with **OK**.

Note: A basic **User** may use the rvsXP program to send and receive files. An **Operator** has user rights and may also execute operator commands. A user with **Administrator** rights may in addition to this use the rvsXP **Administrator** program to configure the rvsXP system.

Removing a user (User dialog)

If you want to remove a user:

- Execute the menu command `Edit -> Remove`. To remove the user, answer the query that appears with **Yes**. If not, answer with **No**.

Editing a user

To edit an existing user:

- Select the user to be edited in the `User` window.
- Execute the menu command `Edit -> Update`.

The `User` dialog opens.

You can now edit all of the user data parameters. You can select German or English in the language field. rvsXP will then display all the menus and messages in the selected language.

Note: If you want to modify your own user data then the entries will only take effect after restarting the `rvsXP Administrator`.

- Confirm your entries with **OK**.

Saving a user

If you set up or modify a user, use **OK** in order to save the modified parameters. rvsXP will use the modified parameters after a restart of rvsXP Monitor. If you delete the rvsXP database or reinstall rvsXP you will also delete all users. Subsequent initializing of the database only retains the administrator who has set up rvsXP.

In order to avoid losing your entries you can save the users and the job start entries or even the entire rvsXP database and restore them later. To do so, please refer to Chapter "Saving the user table and follow-on processing" on page 97.

7 rvsXP database maintenance

The rvsXP database is the memory of rvsXP both as to what happened in the past and what still has to be done. If the rvsXP database is damaged or contains inconsistent entries, files may be sent twice or not at all.

This chapter describes how to maintain the rvsXP database for Windows XP. You will be given recommendations on how to apply strategies for backing up and restoring the database environment and entries suitable for a respective application case.



2 ways to back up and restore


The following table shows the back-up and restore functions implemented in the rvsXP database along with their characteristic features.

Method 1 covers backing up of tables and comprises the following steps

- Saving the station table and
- Saving the user table and the tables for follow-on processing

Method 2 covers backing up the entire rvsXP database.

| To... | Back up | Restore | Recommendation, benefits, disadvantages |
|---|---|---|--|
| Backing up settings after configuration | Backing up the station table (Chapter 7.1.1) and user table and follow-on processing (Chapter 7.1.2). |  | Perform back-up immediately after database creation and setting up of stations, users, and follow-on processing. Repeat back-up each time changes are made to the station table, the user table, and the follow-on processing. |
| Database cleanup (Recommended procedure) |  Important! Verify whether or not current back-ups of station table, user table, and follow-on processing are available. | <p>Step 1 Delete and reinitialize rvsXP database (Chapter 7.3).</p> <p>Step 2 Restore the station table (Chapter 7.4.1).</p> <p>Step 3 Restore the user table and the follow-on processing (Chapter 7.4.2).</p> | <p>Benefit Database cleanup.</p> <p>Disadvantage Transmission data is deleted from the database.</p> <p>Recommendation To restore to specific start conditions and for regular database maintenance.</p> <p>Note: This method does not preserve the parameters. Restart TCPIPRCV or MAXX25RCV.</p> |

| To... | Back up | Restore | Recommendation, benefits, disadvantages |
|---|--|---|---|
| Back up all data and restore them when needed | Save the rvsXP database (Chapter 7.2)  | Restore the rvsXP using a database dump (Chapter 7.4.3) | All data and settings of the rvsXP environment are completely backed up/restored. Benefit One step only Disadvantage All processes, including unsuccessful ones, are restored (normally undesired "junk data" is backed up/restored as well) Recommendation Use for regular backing up transmission data and prior to software updates. We recommend that you also back up the cleaned-up rvsXP environment (station table, user table and follow-on processing). |

External database **Note:** rvsXP HighPerformance uses an external database instead of an internal rvsXP database. Perform maintenance according to the conditions of this database environment. For further information please refer to Chapter 11 describing the SQL server and Oracle interface.

7.1 Back-up: backing up rvsXP tables

7.1.1 Backing up the station table

We recommend that you back up the station table after rvsXP installation and prior to each setting up/editing station settings. You can revert to this back-up if you wish to restore the previous station settings after a re-initialization of the rvsXP database. You thus avoid setting up the stations again. A station table back-up is also ideal to establish a "clean" initial database state in conjunction with a back-up of the user table and the follow-on processing.

- Backing up the station table
- In order to back up the station table with all the parameters for partner stations:
- Start rvsXP Monitor.
 - Open the Station Table window with the Stations menu command.
 - Execute the menu command `Edit -> Export Stationtable`.
 The `Select Backup-File` dialog then opens and displays the files in the rvsXP system folder with the "DAT" extension.
 - Enter the file name under which you want to back up the station table.

- Confirm the entry by clicking **Open**.

rvsXP then backs up the station stable and displays any errors in the `rvs Dump Stationtable` dialog.

Back-up has been successfully completed if the message `Done with 0 errors` appears.

7.1.2 Saving the user table and follow-on processing

We recommend you back up the user table and follow-on processing each time you make changes to them. You can revert to this back-up if you wish to restore the previous settings for users and follow-on processing after a re-initialization of the rvsXP database. You thus avoid performing the set-up again. Such a back-up is also ideal to establish a "clean" initial database state in conjunction with a back-up of the station table.

Table back-up To save the user table and the follow-on processing:

- Stop rvsXP Monitor.
- Start `rvsXP-Administrator` with the `rvsTrayIcon` command `rvs Administrator` or choose `Start -> All Programs -> rvs -> rvs Administrator`.
- Execute the menu command `Database -> Export -> Jobstarts & Users`.

The `Select Backup-File` dialog opens and displays the files in the rvsXP system folder with the "LOG" extension.

- Enter the file name under which you want to save the tables.
- Confirm the entry by clicking **Open**.

rvsXP saves the tables and displays any errors in the `rvs - Dump Jobstart Entries and User` dialog.

Backing up has been successfully completed if the message `END` appears.

- Start rvsXP Monitor.

7.2 Backing up: Backing up the rvsXP database

This method backs up the entire rvsXP database and all rvsXP parameters.

The backup contains the complete current rvsXP environment state, i.e. the station table, the user table and the follow-on processing as well as logs of all transmissions including aborted and not completed processes.

You can revert to this back-up if you wish to restore the previous system state after a re-initialization of the rvsXP database. In most cases this will

not be a “clean” initial state but rather an “unmaintained” state of the user environment, which is disadvantageous and not desired in practice.

Backing up the complete database

To back up the entire rvsXP database:

- Stop rvsXP Monitor.
- Execute the menu command `Database -> Export`.
The `Select Backup-File` dialog opens and displays the files in the rvsXP system folder with the "LOG" extension.
- Enter the file name under which you want to save the rvsXP database.
- Confirm the entry by clicking **Open**.

rvsXP then saves the rvs database and displays any errors in the `rvs Dump Database` message window. Back-up has been successfully completed if the message `End` appears.

- Start rvsXP Monitor.

7.3 Delete and reinitialize rvsXP database

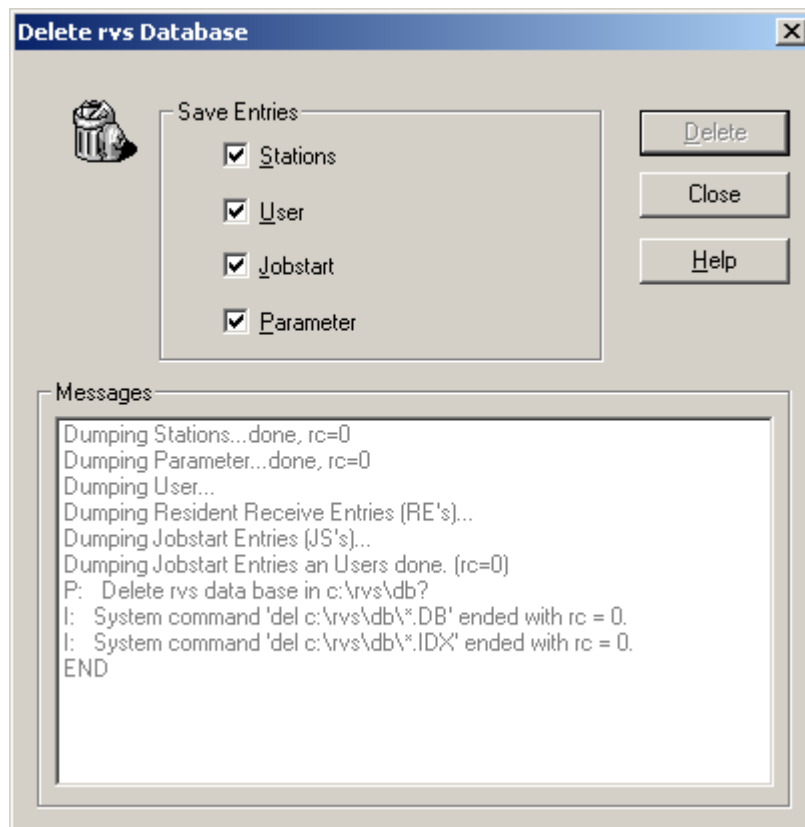
Delete rvs database

First you must delete the rvsXP database contents.

Attention: If you delete the rvsXP database you will irrevocably lose any incomplete data transmissions and all settings which you have made in the rvsXP Administrator.

If you are absolutely sure that you want to rvsXP reinstall your database carry out the following steps:

- Make sure that all send and receive entries have been completed. Pay attention to the send orders and receive entries which make use of the serialization function.
- Back up the tables which you would like to restore after initialization of the rvsXP database is complete. Further instructions are contained in the Chapter 7.1.
- Execute the menu command `Monitor -> Stop`.
- Use the preferred backup method to back up the station tables, users tables and the job start entries or the entire rvsXP database. For appropriate instructions please read Chapter 7.1.
- Delete the rvsXP database. To do so start the `Delete rvs Database` program in the `rvs` program folder.



Backing up selected entries prior to deleting

Prior to the delete operation you can save selected entries.

- Click the entries to be backed up in the field `Save entries`. In this case you will be asked under what name the entries are to be saved. Depending on the type of entries two different file types are created. When backing up `User` and `Program start` the specified file name is given the extension `_job`. The entries can be read-in again with the menu command `Database -> Import` and/or `Database -> Import of Jobstart Entries and User` after the `rvsXP` database has been created again.

Deletion has been successfully completed if the message ended with `RC = 0` appears.

Reinitializing the database

- Initialize the `rvsXP` database. To do so start the `Init Database` program in the `rvs` program folder. `rvsXP` will then generate a completely new `rvsXP` database.

After initialization all tables contain the default `rvsXP` settings.

- Restore the tables desired from the backup files which you made earlier. For appropriate instructions please read Chapter 7.4.
- Start `rvsXP` Monitor.

7.4 Restoring rvsXP data from a back-up

When you have performed a back-up with one of the back-up procedures you can restore the station table, the user table together with the tables for follow-on processing or the entire rvsXP database. See “rvsXP database maintenance” on page 95.

7.4.1 Restoring the station table

To restore the station table:

- Start rvsXP Monitor.
- Execute the menu command Database -> Import -> Stations.

The Select file with station definitions dialog opens and displays the files with the *.dat extension in the rvsXP system folder.

- Select the file which contains the station table which is to be restored.
- Confirm your selection by clicking **Open**.

Restoration of the data has been successfully completed if the Monitor LOG window contains the message

```
I: <ok cmd done> [RVSCNS] 'modst' terminated.
```

7.4.2 Restoring the user table and the follow-on processing

Carry out the following steps if you want restore data for the user table and follow-on processing:

- Stop rvsXP Monitor.
- Execute the menu command Database -> Import -> Jobstarts & Users.

The Select Backup-File dialog opens and displays the files with the *.LOG extension in the rvsXP system folder.

- Select the file which contains the user table and follow-on processing which are to be restored.
- Confirm your selection by clicking **Open**.

rvsXP then opens the rvs - Restore Jobstart Entries and User command prompt window and restores the saved entries for follow-on processing in the rvs database.

Restoration of the data has been successfully completed if the command prompt window closes without error.

Note: The command prompt window appears only briefly in case of a small number of users and entries.

- Start rvsXP Monitor.

7.4.3 Restoring the rvsXP database

To restore the entire rvsXP database using a complete back-up:

- Stop rvsXP Monitor.
- Execute the menu command Database -> Import.
The `Select Backup-File` dialog opens and displays the files with the *.LOG extension in the rvsXP system folder.
- Select the file containing the rvsXP database to be restored.
- Confirm your selection by clicking **Open**.
rvsXP then restores the rvs database and displays any errors in the `rvsXP - Restore Database` dialog.

Restoration of the data has been successfully completed if the message `0 error(s) occurred` appears.

- Start rvsXP Monitor.

7.5 Cleaning up the rvsXP database (cleanup)

Purpose of this function The larger the rvsXP database, the more time is required by rvsXP Monitor and for user access. In order to get the best possible performance rvsXP operates with an automatic cleanup function, which removes information on completed or deleted transmissions from the database after a certain retention period.

Default: Immediate deletion Information about completed actions, e.g. transmissions, is deleted immediately from the rvsXP database by default.

Parameter **CMDDELETE** (rvsXP Administrator -> Settings -> Parameter -> CMDDELETE) controls this function. It has the value **1** by default. If you want to keep information about completed actions for some time, set this parameter to value **0**. This does however place a burden on the rvsXP database and we therefore recommend that by default you set the **CMDDELETE** parameter to **1**. If you want an overview of files sent and received, or if you require this information for archiving purposes, use the rvsXP `$RVSPATH\db\r1stat.log` statistic file (see Appendix "Log files").

Retention time of completed jobs The **CLEANUP** parameter finally determines how long such finished jobs are to be retained in the database if **CMDDELETE = 0**. You can determine the information retention time by using the operator command `cleanup`. The general command is:

```
cleanup days=n ss=yes/no
```

`n` represents the period in units of 24 hours; with `yes` or `no` you specify whether you also want to delete the associated entries in the transmission statistics.

Possible parameter modification:

- Dynamic adaptation during operation of rvsXP Monitor using the rvsXP Administrator. See “Customizing rvsXP Monitor parameters” on page 66.
or:
- Interactively issue the command as a single operator command.
or:
- Static adaptation by integrating the command in the `C:\rvs\init\rdmini.dat` initialization file.

By default the file contains the `cleanup days=3 ss=yes` entry. That means that completed or deleted transmissions remain stored for 72 hours in the rvsXP database until rvsXP Monitor deletes them along with the transmission statistics.

Carry out the following steps if you want to modify the parameters for the restart:

- Open the `C:\rvs\init\rdmini.dat` file. To do so click the `Monitor Parameters` icon in the `rvs` folder.
- Search for the `cleanup` entry and modify it to meet your requirements.
- Save the file and close the text editor.
- Stop rvsXP Monitor and then start it again in order to activate the modified settings.

8 Encryption configuration: Key administration

The present chapter describes the key administration for secure data exchange. It describes how to generate, import and distribute keys as well as how to list and delete imported keys.

8.1 Encryption basics

This is encryption To make unauthorized reading of information impossible you modify the data such that user information is still contained but no longer readily readable. This method is called encryption. There are two types of encryption methods:

Symmetrical method **Symmetrical procedures** apply the same key for data encryption and decryption. Owners of the respective key can encrypt data and decrypt them at any time. This method ensures fast encryption but a low degree of security as the partners must agree on the key to be used for decryption and exchange it in an insecure environment.

Asymmetrical methods **Asymmetrical methods** use a complementary key pair. One key is used for encryption while the other is used for decrypting the data. You cannot use the key used for encryption for decryption. These methods are very safe but require a high degree of computing power during encryption and decryption and may increase the data volume by up to 100%. The nondisclosure of the key required for decryption is a security advantage of the asymmetrical method: Only the recipient is in possession of the key.

8.1.1 Encrypted transmission with rvsXP

rvsXP combines both benefits: The high speed of the symmetrical method and the security level offered by the asymmetrical method. rvsXP uses the following methods:

- **3DES** as symmetrical method (length 3x56 bit = 168 bit),
- **RSA** as asymmetrical method (length 768 to 2048 bit).

Electronic signature To enhance the degree of security rvsXP uses an electronic signature. The signature ensures the data does not suffer any unnoticed corruption during transmission. For this purpose rvsXP calculates the checksum of a given file and encrypts this checksum using the private key. The recipient station decrypts the encrypted checksum with the sender's public key and compares it with the checksum of the decrypted file. A matching checksum indicates the file was not corrupted during transmission.

8.1.2 rvsXP encryption principle and procedure

Creating a key pair Each participant in encrypted communication creates a key pair on his system, consisting of the **public key** and the **private key**.

Distributing public keys / safely storing private keys The recipient makes his public key available to all partners he expects to receive data transmissions from. The respective sender thus can encrypt the data for precisely the partner who sent him the public key in question. There is no security risk in openly transmitting the public key as it is not suited for decryption.


The partner does not share his private key and stores it in a safe place. He can use this key to decrypt all data encrypted with the public key. He can no longer read encrypted messages destined for him if he loses this key.

Importing an external public key Once a partner has received another partner's public key he must import it into the rvsXP database.

The following illustrations show how to exchange your encrypted data with a partner.


You have both successfully installed rvsXP and both have a key pair consisting of the private and the public key. Partner 1 has the PR1 / PU1 key pair. Partner 2 has the PR2 / PU2 key pair.



| | | |
|-----|--|-----|
| PR1 |  | PR2 |
| PU1 | | PU2 |


Keep the private key for yourself. Send your public key to your partner and he will send you his public key.



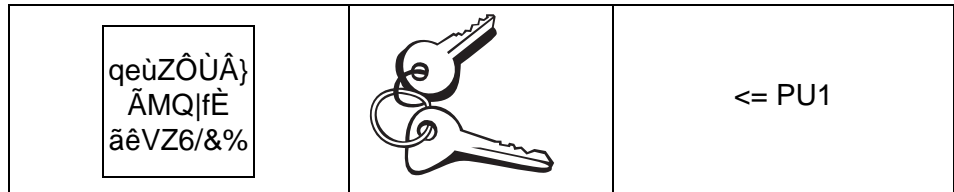
| | | |
|-----|--|-----|
| PR1 |  | PR2 |
| PU1 | => | |
| | <= | PU2 |

Each of you must import the partner's public key into the rvsXP database. This completes the preparations for a long time of securely exchanging any number of files.

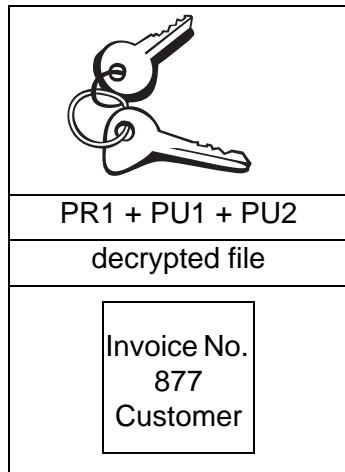


| | | |
|---------------|--|---------------|
| PR1 |  | PR2 |
| Import of PU2 | | Import of PU1 |

Your partner encrypts a file with your public key and sends it to you.



You decrypt the file you received. To do so you need your private key, your own public key and the partner's public key.



8.2 Overview of the rvsXP key administration

In order to use encryption with rvsXP please take these steps:

- Generating keys

 - Check the rvsXP license key. If the **Encryption** module was purchased the rvsXP license key (`$RVSPATH4\init\rdkey.dat`) contains the **Y** module.
 - Generate a key pair for your local station. To do so, use the **Key Admin** application in the rvsXP Administrator or the `genKey` tool; see Chapter "Creating private and public keys".
- Importing keys

 - Import your **private key** and your **own public key** for your local station in rvsXP. To do so, use the **Key Admin** application in the rvsXP Administrator or the `rvskeyimp` tool; see Chapter "Importing Keys (`rvskeyimp`)".

CAUTION: Please handle the private key file very carefully to ensure your privacy.

- Distributing and importing public keys

 - Send the own **public key file** to the partner stations that shall participate in encrypted file transfer. You can do this rvsXP: Use the `rvskeydst` tool, see Chapter "Distribution of keys (`rvskeydst`)".
 - Get a public key file from each of your partner stations that may participate in encrypted file transfer. Import the keys in rvsXP database. To do so, use the **Key Admin** application or the `rvskeyimp` tool; see Chapter "Importing Keys (`rvskeyimp`)".
 - Check rvsXP key configuration to ensure availability of the local private key and of the partner public keys. To do so, use the **Key Admin** application or the `rvskeylst` rvsXP tool; see Chapter "Listing of imported keys (`rvskeylst`)".

Encryption may be enabled for each send job separately. rvsXP automatically decrypts the files received.

8.3 Creating private and public keys

To use the rvsXP encryption function you must have created the key pairs comprising the private and the public key with the `genKey` tool. This is an easy to use program, which randomly creates a new key pair at any call.

You can launch the `genKey` program in two ways: via the command prompt or the graphic interface.

⁴ For more detailed information on the `$RVSPATH` parameter see Chapter 1.6 "Representation means".

Launching via the command prompt

Syntax

```
genKey
    [-chefmopstv] [--768] [--1024] [--2048]
    [--owner] [--creator] [--from] [--to]
    [--help] [--size] private_key public_key
```

The `private_key` (in the dialog: **Private Key File**) and `public_key` (in the dialog: **Public Key File**) parameters stand for the files where the generated keys will be saved. Only these two parameters are required to produce a new key pair. The other parameters are optional. The table below gives an overview of all optional parameters with explanations and default values.

The | character in the **Parameter** column separates the two possible variants of the same optional parameter.

Examples `genKey -h`
`genKey --h`
 (These two calls display the `genKey` help.)

Options

| Parameters | Description | Default value |
|---------------------------------------|--|---------------|
| <code>-h --help</code> | Print this message | |
| <code>-c --creator</code> string | Creator of the files Example: <code>-c gha</code> <code>--creator gha</code> In the dialog: Creator | 'unknown' |
| <code>-o --owner</code> string | Owner of the files In the dialog: Owner | 'unknown' |
| <code>-f --from</code> string | First day of validity period (TT.MM.JJJJ YYYY-MM-DD) You can choose between the German (TT.MM.JJJJ) and the international (YYYY-MM-DD) date version. In the dialog: begin | today |
| <code>-t --to</code> string | Last day of validity period: (TT.MM.JJJJ YYYY-MM-DD). In the dialog: end | |

| Parameters | Description | Default value |
|--------------------------------|--|---------------|
| <code>-p --per string</code> | Validity period d D t T=day, m M=month, j J y Y=year, h H=hours, i I=minutes, s S=seconds You must decide upon one unit of time. Example: 1m (1 month) or 5D (5 days) In the dialog: period | 3 months |
| <code>-s --size value</code> | Key size in bit (maximum 2048 bits; must be divisible by 8) In the dialog: KeySize | 1024 |
| <code>--768</code> | Creates a key pair with 768 bits In the dialog: 768 bit | |
| <code>--1024</code> | Creates a key pair with 1024 bits In the dialog: 1024 bit | |
| <code>--2048</code> | Creates a key pair with 2048 bits In the dialog: 2048 bit | |
| <code>-e</code> | Use the exact current time (normally the validity period starts at 00:00:00 UTC) In the dialog: start today | |
| <code>-m</code> | Print process information (not reasonable on batch systems) | |
| <code>-q</code> | be quiet | |
| <code>-v</code> | be verbose | |

Example 1 `genKey TC2private TC2public`
(Generates a private key named `TC2private` and a public key named `TC2public`. The length of the keys will be 1024 bit and they will be valid three months beginning with the actual day.)

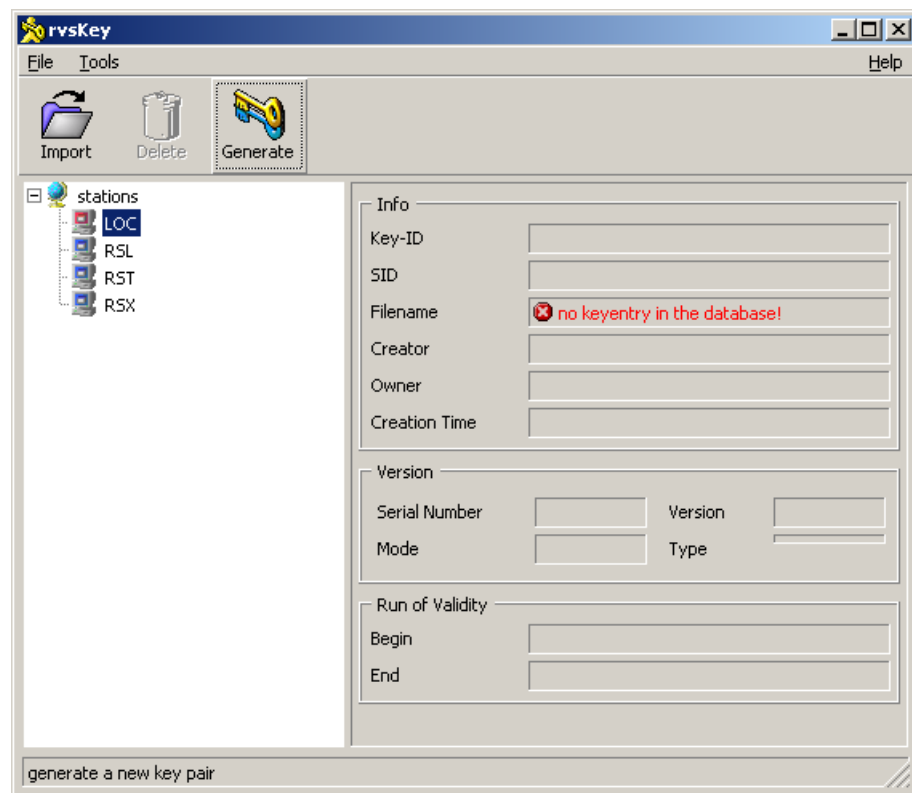
Example 2 `genKey -c Fischer -f 01.06.2001 -t 01.12.2001 -s 2048 WO1.pri WO1.pub`
(Generates a key pair with the creator `Fischer` and with a size of 2048 bit. The valid period starts on June, 1 2001 and ends on December, 1 2001.)

Graphic interface

Start rvsKey with

- Start -> Run
and type rvskey or
- Start the program rvsKey in the rvs folder (Start -> Programs -> rvs ->Encryption Key Administration)
or
- Start the program in the rvsXP Administrator with
Start Program -> Key Admin

In the rvsKey dialog you can call the genKey program by choosing the Generate symbol to generate a new key pair.



The Generate New Keypair... dialog opens.

- After setting all parameters click the **Generate** button to generate a new key.

Note: An explanation of the parameters has been given in the previous chapter (Creating private and public keys).

8.4 Importing Keys (`rvskeyimp`)

The `rvskeyimp` tool imports a key into the `rvsXP` database.

To import a key it is necessary to provide the name of the file containing the key with full path and the ID of the station (SID) for which the key shall be used. The tool `rvskeyimp` reads the file containing the key, copies it into the directory containing all imported keys (see `rvsenv.dat` parameter) and writes all necessary information into the `rvsXP` database (Table SV: S=Keys, V=Administration).

You can launch the `rvskeyimp` program via the command prompt or the graphic interface.

Launching via the command prompt

Syntax

```
rvskeyimp [-?] -i <FILE> -s <SID>
```

Options

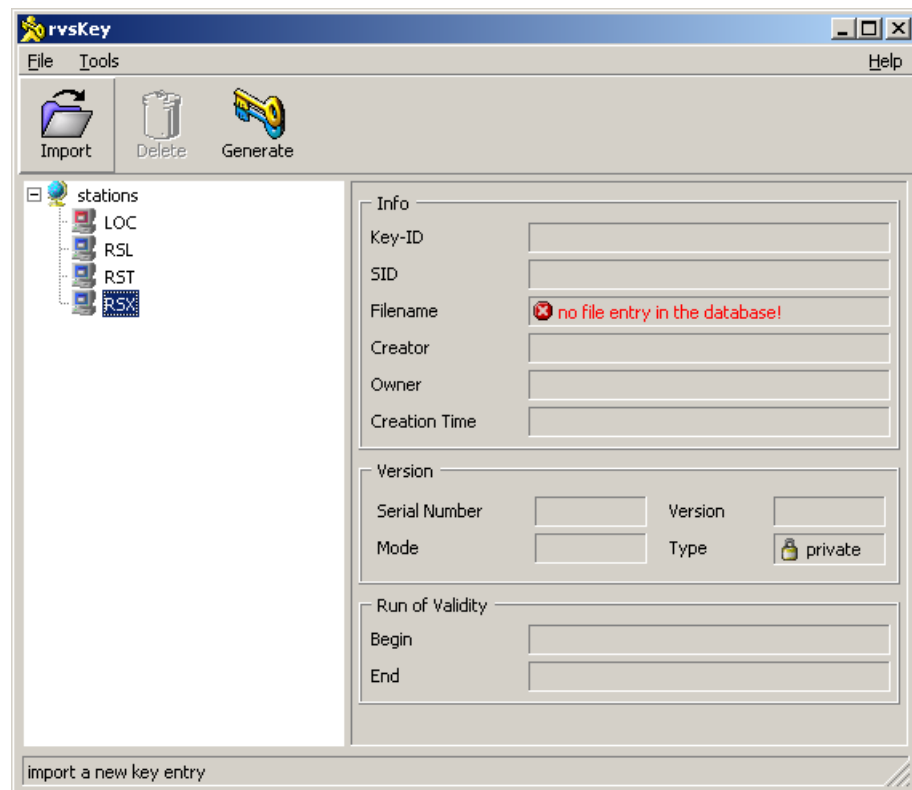
- i <FILE> Import the key from file FILE where FILE must be specified with its full path
- s <SID> The station ID for which the key will be imported

Example

```
rvskeyimp -i C:\rvs\usrdat\A15A13.pri -s LOC
rvskeyimp -i C:\rvs\usrdat\A15A13.pub -s RSL
```

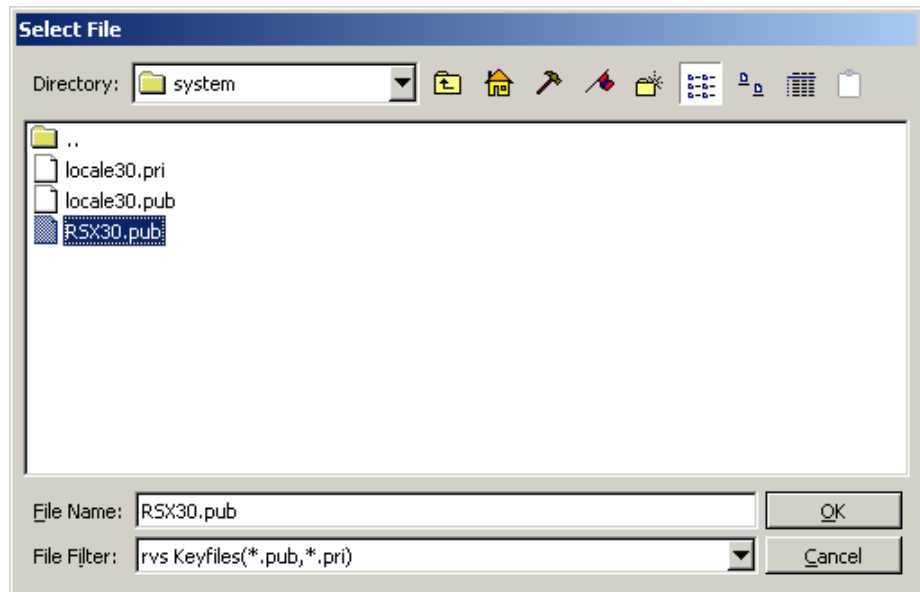
Graphic interface

To import the keys into the rvsXP database choose the `Import` symbol in the `rvsKey` dialog to launch the `rvskeyimp` tool:



- Select the station for which the import should take place and then click the `Import` symbol.

The `Select File` dialog appears:



- Select the file containing the key or write the file name in the `File Name` field of the `Select File` dialog.
- Click the **OK** button to import the key.

8.5 Distribution of keys (`rvskeydst`)

To enable the communication partner to receive and decrypt data intended for him the data must have been encrypted with his public key. This why the sender needs the recipient's public key. Exchange of public keys can be done by e-mail, mail, fax, disk or as well with `rvs`[®]. The `rvskeydst` tool sends a key file to a communication partner.

You must specify the name and the full path of the file containing the key to send a key file to a communication partner. The station IDs (SIDs) can be either handed over as parameter or in the form of an input file (one line for each station ID).

You can launch the `rvskeydst` program via the command prompt only.

Launching via the command prompt

Syntax

```
rvskeydst [-?] -f <FILE> -s <SID> | -l <FILE>
```

Options

`-f <FILE>` The file containing the key where the full path must be specified with `FILE`

- s <SID> The ID of the station to which the key will be sent (repeatable)
- l <FILE> The file containing a list of station IDs

Example

```
rvskeydst -f C:\rvs\keycreate\C45PUB.pub -s r11 -s a34 -s gedas  
rvskeydst -f C:\rvs\keycreate\C45PUB.pub -l C:\rvs\list\sendlist1.lst
```

8.6 Deleting imported keys (*rvskeydel*)

The *rvskeydel* tool deletes keys that previously were imported into the rvsXP database. It deletes the key from the database and the respective key file in the key folder. The *-l* option creates a list containing the delete statements for all keys in the database. The output comprises two lines for each key, one with all key data, one with the appropriate delete statement.

To use this list for deleting keys you must pipe the output into a file. All delete statement lines have been commented out with REM at the line start.

- Start *rvskeydel* at the command prompt and the *-l* parameter to pipe the output into a file. Example:
C:\rvs\system\rvskeydel.exe -l DelFile.txt
- Remove the comment indicator (REM) from those keys you wish to delete.
- Assign the file executable rights.
- Start the file from the command prompt:

You can launch the *rvskeydel* tool via the command prompt or the graphic interface.

Launching via the command prompt

Syntax

```
rvskeydel [-?ldur]
```

Options

- l Generate a list of all key entries
- d KEYID Delete key with the ID KEYID
- u Remove the public key
- r Remove the private key

Example of a generated key list:

```
REM SID=LOC DATEBEGIN=2001/01/25 00:00:00
DATEEND=2001/04/25 02:00:00 ...
REM rvskeydel -d 1074002581 -r
REM SID=LOC DATEBEGIN=2001/01/25 00:00:00
DATEEND=2001/04/25 02:00:00 ...
REM rvskeydel -d 1074002581 -u
```

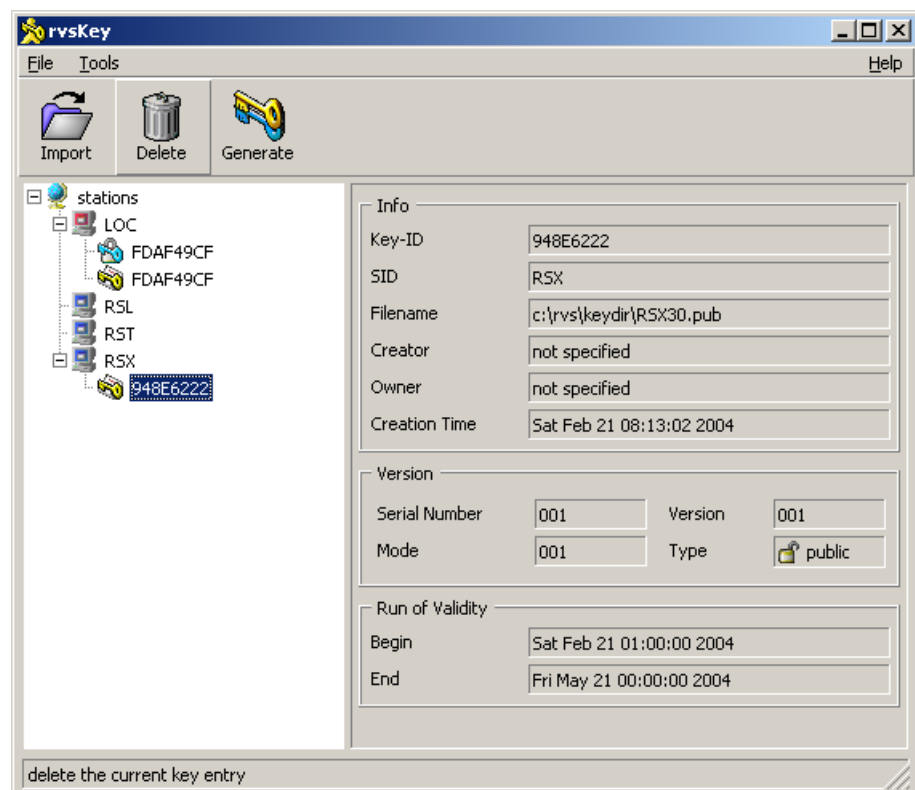
Example

```
rvskeydel -l
(Lists all key entries.)
```

```
rvskeydel -d 103456734 -u
(Removes the public key with the ID 103456734.)
```

Graphic interface

- To delete the keys from the rvsXP database choose the Delete symbol in the rvsKey dialog to launch the rvskeydel tool:



- Choose the private or public key (highlighted in the `rvsKey` dialog above) in the `rvsKey` dialog.
- Click the `Delete` button.
- Answering the following query with **Yes** causes the key to be deleted in the `rvsXP` database and from the list in the `rvsKey` dialog. The key will not be deleted if you choose **No**.

8.7 Listing of imported keys (`rvskeylst`)

The `rvskeylst` tool lists all keys that have been imported into the database with their complete data. This command does not support any options.

You can launch the `rvskeylst` tool via the command prompt or the graphic interface.

Launching via the command prompt

Syntax

```
rvskeylst
```

Example of result

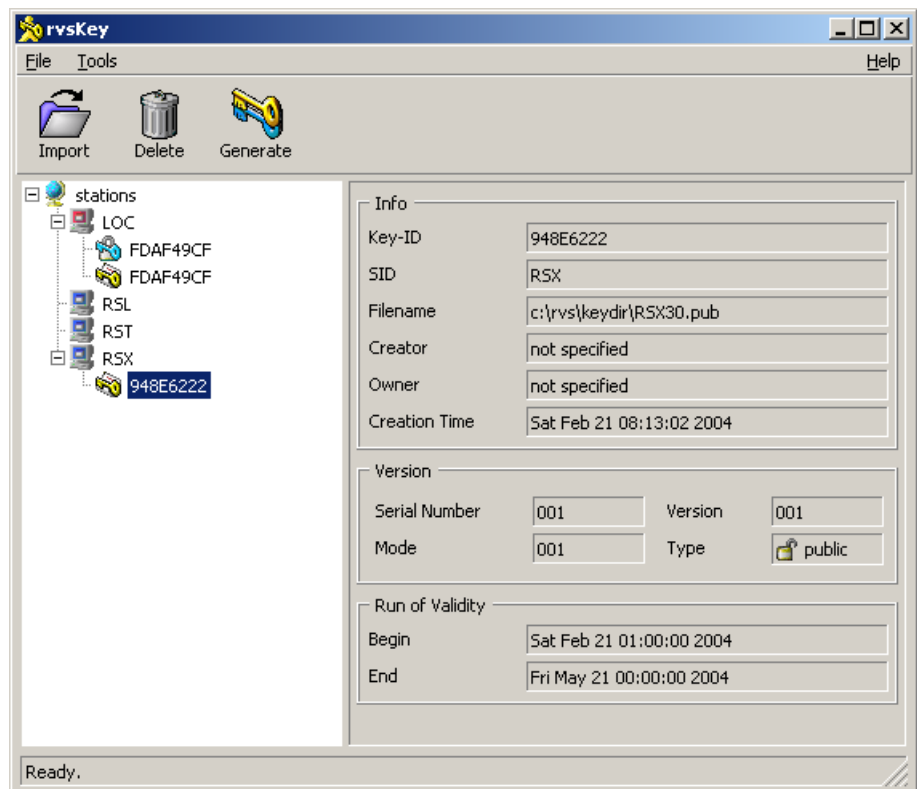
```
SID=WO1 BEGIN=2001/02/22 00:00:00 END=2001/05/22  
01:00:00 FILE= ...  
SID=LOC BEGIN=2001/02/22 00:00:00 END=2001/05/22  
02:00:00 FILE= ...
```

Launching via the graphic interface

After launching `rvskeylst` the key entries are shown in the graphic `rvsKey`.

Start `rvsKey` with

- Start -> Run
and type `rvskey` or
- Start the program `rvsKey` in the `rvs` folder (Start ->
Programs -> `rvs` -> Encryption Key Administration)
or
- Start the program in the `rvsXP` Administrator with
Start Program -> Key Admin



- If you want to refresh the list, select **Reload** in the **File** menu.

8.8 Configuration of offline compression

If the **offline compression** module was purchased it is enabled by the **rvsXP** license key. Offline compression is enabled when the key file **rdkey.dat** (located in the **rvsXP init** folder) contains the **Z** module.

There are none additional configuration steps required for offline compression. Offline compression may be switched on for each send job separately. It is handled automatically when receiving files.

9 Code conversion

ASCII and EBCDIC You are probably aware that text files are stored on most systems in one of two computer codes, namely ASCII (American National Standard Code for Information Interchange) or EBCDIC (Extended Binary Coded Decimal Interchange Code). ASCII is the standard code for UNIX and DOS/Windows Systems, EBCDIC for OS/400 and OS/390 operating systems. While the assignment of digits and letters of the Latin alphabet is standardized within each of these two code families, special characters (like square brackets []) or national language characters (like accented letters or umlauts) may be assigned to different codes in different code pages⁵ within a family.

The Odette FTP standard and so rvs[®] distinguish between **text** files (format **T**) and **non-text** files (**F**, **V** and **U** formats).

Text files are always transmitted in ASCII (stipulated by the ODETTE protocol) and automatically delivered in the local system-specific code of the target system (where required).

Non-text files are transmitted as is without conversion, unless you specifically request code conversion on the sender and/or recipient side.

Example 1 Text files are coded in ASCII on a Windows XP computer and in EBCDIC (text file): on an OS/390 system.

Direction: rvsXP => rvsMVS

A rvsXP station is to send a text file to a rvsMVS station. To be recognized as a text file, this file must be sent in the text format (right mouse button `rvsTrayIcon -> rvs -> New Send Entry -> Change Options -> Format=Text`). This file will be transmitted as an ASCII file (stipulated by the ODETTE protocol) and will be converted into EBCDIC by the rvsMVS station.

Direction: rvsMVS => rvsXP

When sending a text file from an OS/390 system you must define the format of the file (**T** format), so this EBCDIC text file will be converted to ASCII code before sending. Text files are transmitted only in ASCII format (**Odette**). The rvsXP system will receive the text file in ASCII.

Example 2 Direction: rvsXP => rvsMVS (Non text file):

Normally an OS/390 system requires (rvsMVS) files in the **F** (fixed) or **V** (variable) format to be able to store and process them without problems. As a Windows XP system is able to work only with files in the **T** (Text) or **U** (unstructured=binary) format you have to convert them to the pseudo fixed or pseudo variable format before sending.

For this purpose rvsXP offers the `rvsut2fv` tool. For more details on this tool and its command syntax please refer to the rvs[®] Reference

⁵ The code pages vary as they depend on the respective operating system.

manual, Part III, Utilities. The rvsXP station is then to send the converted file in the **F** or **V** format and also set the parameters for the input and output code (Start -> Programs -> rvs -> rvsXP -> Edit -> New Send Entry -> Options -> Format=Fixed/Variable -> Input-Code=A; Output-Code=E). The input code is **A** for ASCII, the output code is **E** for EBCDIC.

rvsXP supports code conversion both during sending and receiving of files. You can force a code conversion of received files by setting up an appropriate resident receive entry.

For the code conversion, you can use two sorts of code conversion tables:

- those integrated in rvsXP, or
- your own conversion tables.

9.1 Automatic code conversion with integrated conversion tables

The rvsXP code conversion tables meet the following standards:

ASCII:ISO 8859 Latin 1 and
EBCDIC:CECP 037.

The code conversion tables are located as `rtcae.dat` and `rtcea.dat` files in the `$RVSPATH6\system` folder.

The meaning of the letters in those file names is:

- r rvs[®]
- t translate (conversion)
- c code
- e EBCDIC
- a ASCII

The `$RVSPATH\system\rtcae.dat` file contains the conversion table for ASCII => EBCDIC conversion, and the `$RVSPATH\system\rtcea.dat` file the respective conversion table for EBCDIC => ASCII conversion.

To have a conversion performed according to these conversion tables you must copy the following files to the `$RVSPATH\init\` folder:

```
$RVSPATH\system\rtcae.dat,
$RVSPATH\system\rtcea.dat.
```

⁶ For more detailed information on the **\$RVSPATH** parameter see Chapter 1.6 "Representation means".

9.2 Code conversion with user code tables

The rvsXP user can use his own code tables instead of using the system code tables for code conversion. The input code (CODEIN), output code (CODEOUT) parameters as well as the code table (CODETABLE) are of importance for code conversion (Start -> Programs -> rvs -> rvsXP -> Edit -> New Send Entry -> Options). The Input Code parameter defines the code of the file before conversion and the Output Code parameter is the code after the conversion. The code table parameter allows you to use your own table for code conversion.

To use the user specific code tables you have two possibilities:

1. You define all three parameters: input code, output code, and code table. See Chapter 9.3 "How to perform a code conversion" for more detailed explanations on code conversion during receiving and sending.
2. You define Input Code and Output Code only and copy your own conversion table to the \$RVSPATH\init folder. Your own conversion table must be named `rtcae.dat` or `rtcea.dat`. rvsXP will only recognize files named according to this convention as conversion tables.

9.2.1 Structure of the code conversion tables

If you intend to create your own code conversion tables, these should have the same structure as rvs[®] system code conversion tables (\$RVSPATH\system\rtcae.dat and \$RVSPATH\system\rtcea.dat).

These code conversion tables have 256 decimal numbers (8 bit). The position in the table (0th position in the table is the number 0) defines the input code, and the number located on this position is the output code for the same character. The following example will illustrate this:

Example To understand it, you need an ASCII code table, an EBCDIC code table and the rvs[®] conversion tables (`rtcae.dat`, `rtcea.dat`).

We assume you wish to perform an ASCII => EBCDIC code conversion for the small letter **r**. This character is assigned the decimal value **114** in the ASCII code table. The number **114** defines the position of the decimal value for the small letter **r** in the `rtcae.dat` code conversion table in the EBCDIC table. So, you can find at the **114th** position (number 0 is 0th position) in the `rtcae.dat` file the number **153** representing the value for the small letter **r** in the EBCDIC table.

One more example: **A**: On the **65th** position in the \$RVSPATH\system\rtcae.dat file is the number **193**, being the EBCDIC decimal value for the capital letter **A**. In the opposite direction you can find ASCII code **65** for **A** on the **193th** position in the \$RVSPATH\system\rtcea.dat file.

9.3 How to perform a code conversion

The following examples show how to perform a code conversion when receiving and sending files.

9.3.1 Code conversion when sending files

This chapter describes how to send files via `rvsXP` and `rvsbat` with code conversion.

Sending files with
code conversion via
`rvsXP`

Automatic ASCII to EBCDIC code conversion

- Start the `rvsXP` program using
right mouse button `rvsTrayIcon` -> `rvs`
or
Start -> Programs -> `rvs` -> `rvsXP` .
- Choose Edit -> New Send Entry.

The Choose file to be sent dialog opens.

- Select the desired file and confirm your selection with **Open**.
- Specify other send parameters
Station ID and New Name.
- Specify the format of the file to be sent in the Options area (see Chapter 6.3.1 "Creating send orders");
 - Specify **A** for ASCII as Input-Code
 - Specify **E** for EBCDIC as Output-Code

Result: An ASCII file will be sent as an EBCDIC file.

Note: If your partner station expects a file in the F (fixed) or V (variable) format, you should convert it with the `rvsut2fv` `rvsXP` utility. For more details on this tool and its command syntax please refer to the Reference manual, Part III, Utilities.

Code conversion with user specific code conversion tables

- Start the `rvsXP` program using
right mouse button `rvsTrayIcon` -> `rvs`
or
Start -> Programs -> `rvs` -> `rvsXP` .
- Choose Edit -> New Send Entry.

The Choose file to be sent dialog opens.

- Select the desired file and confirm your selection with **Open**.

- Specify other send parameters
Station ID and New Name.
- Specify the other send parameters Station ID and New Name.
- Under Options specify the format of the file to be sent (see Chapter 6.3.1 "Creating send orders").
- Click the **Search** button in the Codetable area to specify the path to your own conversion table (e.g. `$RVSPATH7\arcdire\rtcusr.dat`).

Result: A file will be sent and converted as specified in the user code table.

Sending with
code conversion via
rvsbat

Automatic ASCII to EBCDIC code conversion

- Create a job file with ASCII to EBCDIC code conversion.

Example

Create the job file (e.g. job) containing

```
send /c dsn="<file to be sent>" format=U
codein=A (sid="<recipient>" codeout=E).
```

- Start the job with

```
rvsbat /ijob or rvsbat -ijob.
```

Result: The ASCII file <file to send> will be sent as an EBCDIC file.

Code conversion with user specific code conversion tables

- Create a job file in which you explicitly specify the conversion tables.

Example

Create a job file (e.g. job) containing

```
send /c dsn="<file to be sent>" format=U
sid="<recipient>" codetrans=t codetable="<path
and name of user code table>".
```

- Start the job with

```
rvsbat /ijob or rvsbat -ijob.
```

Result: A file <file to be sent> in the U format will be sent and converted as specified in the user code table. `codetrans=t(able)` indicates you are using your own conversion table.

⁷ For more detailed information on the **\$RVSPATH** parameter see Chapter 1.6 "Representation means".

9.3.2 Code conversion when receiving files

This chapter describes how to use the code conversion for receiving files.

9.3.2.1 Using resident receive entries created with rvsXP

Creating a resident receive entry with ASCII to EBCDIC code conversion:

- Start the rvsXP Administrator with right mouse button rvsTrayIcon -> rvs Administrator
or
Start -> Programs -> rvs -> rvs Administrator.

- Choose Job -> Receive.
- The Resident Receive Entries window appears.
- Choose Edit -> Add to add a new resident receive entry.
- Enter your resident receive entry parameters.

In the Code Conversion field you can define the direction of the code conversion (none, ASCII => EBCDIC or EBCDIC => ASCII), the path and name of the rvsXP system code conversion table from ASCII to EBCDIC (\$RVSPATH⁸\init\rtcae.dat or \$RVSPATH\system\rtcae.dat).

- Press the <Enter> key to create the new resident receive entry.

Result: An ASCII file received via the defined resident receive entry will be saved as an EBCDIC file.

Creating a resident receive entry with EBCDIC to ASCII code conversion

- Start the rvsXP Administrator with right mouse button rvsTrayIcon -> rvs Administrator
or
Start -> Programs -> rvs -> rvs Administrator.

- Choose Job -> Receive.
- The Resident Receive Entries window appears.
- Choose Edit -> Add to add a new resident receive entry.

⁸ For more detailed information on the \$RVSPATH parameter see Chapter 1.6 "Representation means".

- Enter your resident receive entry parameter.

In the Code Table field you can define the path and name of the rvsXP system code conversion table from EBCDIC to ASCII (\$RVSPATH⁹\init\rtcea.dat or \$RVSPATH\system\rtcae.dat).

- Press the <Enter> key to create the new resident receive entry.

Result: An EBCDIC file received via the defined resident receive entry will be saved as an ASCII file.

Create a resident receive entry with user specific code conversion table:

- Start the rvsXP Administrator with
right mouse button rvstrayIcon -> rvs
Administrator
or
Start -> Programs -> rvs -> rvs Administrator.

- Choose Job -> Receive.

- The Resident Receive Entries window appears.

- Choose Edit -> Add to add a new resident receive entry.

- Enter your resident receive entry parameter.

Specify path and name of the user-defined conversion table (e.g. \$RVSPATH\arcdire\rtcusr.dat) in Code Table.

- Press the <Enter> key to create the new resident receive entry.

Result: A file received via the defined resident receive entry will be processed converted with the user defined code table.

⁹ For more detailed information on the \$RVSPATH parameter see Chapter 1.6 "Representation means".

9.3.2.2 Using resident receive entries created with rvsbat

Creating a resident receive entry with ASCII to EBCDIC code conversion:

- Create a job file that creates a resident receive entry.

Example

Create a job file (e.g. job) containing

```
resentr /c dsn="<received ASCII file >"  
codetrans=e sid="<Sender>".
```

- Start the job with

```
rvsbat /ijob or rvsbat -ijob.
```

Result: An ASCII file received via the defined resident receive entry will be saved as an EBCDIC (codetrans=e) file.

Create a resident receive entry with EBCDIC to ASCII code conversion:

- Create a job file that creates a resident receive entry.

Example

Create a job file (e.g. job) containing

```
resentr /c dsn="<received EBCDIC file>"  
codetrans=a sid="<Sender>".
```

- Start the job with

```
rvsbat /ijob or rvsbat -ijob.
```

Result: An EBCDIC file received via the defined resident receive entry will be saved as an ASCII (codetrans=a) file.

Creating a resident receive entry with user specific code conversion table:

- Create a job file that creates a resident receive entry.

Example

Create a job file (e.g. job) containing

```
resentr /c dsn="<received EBCDIC file>"  
codetrans=t codetable="<user code table, e.g.  
$RVSPATH10\arcdire\rtcusrdat>" sid="<Sender>".
```

- Start the job with

```
rvsbat /ijob or rvsbat -ijob.
```

Result: A file received via the defined resident receive entry will be converted with the user defined code table.

¹⁰ For more detailed information on the **\$RVSPATH** parameter see Chapter 1.6 "Representation means".

10 Backing Up and Recovering rvsXP Data

rvsXP lets you back up all relevant (dynamic and static) data and recover them if necessary. This is particularly important when an error has occurred in rvsXP and the user wishes to revert to the old status prior to the error.

For backing up, a distinction is made between dynamic and static data.

Static data are: station data, jobstarts (REs and JSs), users and rvs[®] parameters.

“Dynamic data” refers to information on send and receive jobs. During back-up, this data is written to a Redolog file to be able to recover incomplete transmission jobs at a later time. Completed jobs are logged but are not relevant for recovery (see 10.2 "Procedure").

10.1 New parameters

To write the data to a Redolog file, the BACKUP parameter in the \$RVSPATH/rvsenv.dat environment file must be set to ON (rvsXP Administrator -> Edit -> Environment).

Example:

```
BACKUP = 'ON'
```

As an option, you can also use the REDOLOGFILE parameter to specify the path to the Redolog file. The default is: \$RVS_HOME/arcdiredo.log.

Example:

```
REDOLOGFILE = 'D:\rvsrecover\redo.log'
```

Note: Make sure to save the back-up data on a disk where rvs[®] is not installed in order to be able to recover data when a hard disk failure has occurred.

You must restart rvs[®] to have these parameters take effect. Any dynamic data will be written (saved) to the Redolog file from this moment onwards.

10.2 Procedure

The present chapter describes the general procedure during rvs[®] data back-up and recovery.

The new rvs[®] program for backing up and recovering data is called rvsredo.exe and is located in the \$RVSPATH/system directory.

Syntax:

```
rvsredo [-latype] [-tfbsdvr]
```

More about the rvsredo options, please read in chapter 10.3 "rvsredo Options".

Back-up

The following steps are necessary:

- Set the `BACKUP` parameter in the `$RVSPATH/rvsenv.dat` file to `ON`. As an option you can also set the `REDOLOGFILE` parameter.
- Stop and restart `rvs`[®] so that the parameter configuration change can take effect. Any dynamic `rvs`[®] data will be logged from this moment onwards.
- Execute the

```
rvsredo -c
```

command. This will back up the static `rvs`[®] data (stations and parameters) to the `$RVSPATH/arcdire/static.dat` file.
- Backup of the static data (users, REs and JSs) with the command

```
rvsdru
```

This program generates the text file `rlrudump.log` to the `$RVSPATH/arcdire` directory.
- Backup of the dynamic data with a command

```
rvsredo -y
```

This command stores all dynamic data in the file `$RVSPATH/arcdire/dynamic.dat`.

Data recovery

You need an existing back-up to restore data.

To recover data, perform the following steps:

- Stop `rvs`[®]
- Delete and Reinitialize the database (See chapter 7.3 "Delete and reinitialize `rvsXP` database")
- Restore static data. To do so, use the `rvswdb` and `rvsbat` program.
Syntax: `rvswdb /i<path of the input file>`
Syntax: `rvsbat /i<$RVSPATH\arcdire\rlrudump.log>`
Example: `rvswdb /iC:\rvs\arcdire\static.dat`
Example: `rvsbat /iC:\rvs\arcdire\rlrudump.log`
- Resore dynamic data
Generate the list of send/receive entries not yet fully processed using the `rvsredo -l` command to pipe it into a file (e.g. `restore.bat`).

```
rvsredo -l -bC:\rvs\arcdire\dynamic.log > job.bat
```
- At the command prompt, launch the batch file (e.g. `restore.bat`) generated with the `rvsredo -l` command. This will complete any

rvs[®] commands not yet processed (files will be sent, EERP will be sent or received, ...)

- Alternatively, you can generate the list of send/receive entries already completed and not yet fully completed using the `rvsredo -a` command to pipe it into a file (e.g. `restore.bat`).

```
rvsredo -a -bC:\rvs\arcdire\dynamic.log > job.bat
```

- Call the batch file (e.g. `job.bat`)

The example batch files `rvsbackup.bat` and `rvsrecovery.bat` you can find in the directory `$RVSPATH/system`.

10.3 rvsredo Options

The program `rvsredo.exe` offers the following options:

Syntax:

```
rvsredo [-latype] [-tfbsdsv]
```

| PARAMETER | DESCRIPTION |
|--|--|
| -l | Displays a list on the screen, comprising all send/receive orders not yet completed. You can use this option with the <code>-t <time limit></code> , <code>-f <redo file></code> and <code>-b <basis file></code> options. |
| -t <time limit> | Date and time up to which send/receive orders are displayed. |
| -f <redo file> | Name of the redo file. |
| -b <basis file> | Name of the dynamic file |
| -c | Generates the <code>\$RVSPATH/arcdire/static.dat</code> file with static data (station list,...). You can use the <code>-f</code> option when you wish to save your data to another file. |
| -f <file name of static data> | Name of the file where you wish to save your static data. You can use this option in conjunction with the <code>-c</code> option only. |
| -p | Changes the current status to "pending" (waiting for EERP) for an already existing send order. With this option the options (<code>-s</code> , <code>-d</code> , <code>-v</code> and <code>-t</code>) from the table can be used. |

| PARAMETER | DESCRIPTION |
|-----------|--|
| -e | Changes the status to "ended" for an already existing send order. With this option the options (<code>-s</code> , <code>-d</code> , <code>-v</code> and <code>-t</code>) from the table can be used. |
| -s | Sender StationID for which the <code>-p</code> and <code>-e</code> option is to be used. |
| -d | Receiver StationID for which the <code>-p</code> and <code>-e</code> option is to be used. |
| -v | Virtual file name for which the <code>-p</code> and <code>-e</code> option is to be used. |
| -t | Date and time of send order provision for which the <code>-p</code> and <code>-e</code> option is to be used Example: 2005/10/26 11:20:34 |

11 rvsXP HighPerformance

- SQL server and Oracle link In order to efficiently master the continuously growing flow of data and to increase the performance level of rvs[®] the internal rvs[®] C-ISAM database can be replaced with a high-performance database:
- On Windows NT/2000/WS 2003/XP, AIX and LINUX systems of version 2.05 and later there is the option to link to an Oracle database.
 - Windows NT/2000/WS 2003/XP systems of version 2.11 and later can be linked to a Microsoft SQL Server.

rvsXP license:

- License key The rvsXP license key file `$RVSPATH11\init\rdkey.dat` must include the letter **O** in the **Included Components** line to start rvsXP with Oracle link.

To start the SQL Server link you must specify **Q**.

11.1 Configuration of ORACLE

- Requirements To install rvsXP High-Performance you must have a fully functioning Oracle database, at least version 8.1.7. At the moment we support the link to the Oracle database version 8, 9 and 10. Normally we recommend to have the same version of Oracle client and Oracle server. Exception: On Windows systems we recommend Oracle Client 9 with the Oracle Server 8.1.7.

Note: For the Oracle client (on Windows) must Oracle version 9 with a patch 9.2.0.4 be installed.

To run the Oracle database on a UNIX system your configuration file (e.g. `.profile`) has to be expanded with Oracle-compatible definitions of environment variables.

- Configuration file Example: LINUX Here are two examples of the configuration file for LINUX and AIX systems with Oracle environment variables.

LINUX

```
ORACLE_OWNER=oracle
export ORACLE_OWNER
ORACLE_HOME=/opt/oracle/product/8.1.7
export ORACLE_HOME
ORACLE_SID=rvslnx4
export ORACLE_SID
export DBID=ORA
PATH=.:$PATH:$ORACLE_HOME/bin
```

¹¹ For more detailed information on the **\$RVSPATH** parameter see Chapter 1.6 "Representation means".

Configuration file
Example: AIX

```
export PATH
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/home/bwa/rvs/
system:
/home/bwa/rvs/lib:/opt/oracle/product/8.1.7/lib
export LD_LIBRARY_PATH
```

AIX

```
ORACLE_SID=rvsora
export ORACLE_SID
export DBID=ORA
PATH=$PATH:$ORACLE_HOME/bin:
export PATH
```

Two configuration options are offered:

1. rvs[®] and Oracle run on the same computer
2. rvs[®] and Oracle run on different computers

For option 2.: When rvs[®] and Oracle are running on different computers, the communication between both systems takes place via a LAN supported by both systems. Here the normal Oracle Client/Server solution would be used.

Server: The Oracle Server software must be installed and configured on the database server. The so called LISTNER must also be configured on this server.

Client: The rvs[®] OFTP server and Oracle client must be installed and configured on the client.

Note: If Oracle is running on an external computer, this computer must be defined in the file:

```
$ORACLEHOME\network\admin\tsnames.ora.
```

Example (tnsnames.ora):

```
#TNSNAMES.ORA Network Configuration File: /opt/
oracle/product/8.1.7/network/admin/tnsnames.ora
# Generated by Oracle configuration tools.
```

```
RVSORA.GEDAS.DE =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST =
rvsaix3) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = rvsora.gedas.de)
    )
  )
```

11.2 Configuration of rvsXP

The following variable definitions will be asked during the rvsXP installation.

These variable definitions will be added to the
`$RVSPATH\rvsenv.dat` rvsXP environment file by the installation
script:

```

DBNAME = '<ORACLE System ID>'
DBUSER  = '<ORACLE user@ORACLE database>'
DBPSW   = '<ORACLE user password>'

```

These variable values have been defined when installing and configuring Oracle.

Example (excerpt from \$RVSPATH¹²/rvsenv.dat):

```

*rvsXP- environment file (rvsenv.dat)
DBNAME      = 'rvsora'
DBUSER      = 'skk@RVSORA.GEDAS.DE'
DBPSW      = 'skk'

```

11.3 Microsoft SQL Server configuration

Requirements The SQL Server database, version 8.0 and later must have been set up prior to the installation of rvsXP (SQL Enterprise Manager).

The installer queries the link parameters to the Microsoft SQL Server and saves them to the rvsenv.dat file:

```

DBNAME      = '<host>.<database name>'
DBUSER      = '<database user>'
DBPSW      = '<database user password>'

```

Before rvsXP can access the database a database and a user with 'database owner' privileges for this database must have been set up on the database host.

Make sure to set 'ANSI NULL default' in the database properties.

The SQL server must allow SQL server authentication.

¹² For more detailed information on the \$RVSPATH parameter see Chapter 1.6 "Representation means".

Appendix

rvsXP User Manual

Appendix 1: The rvs® parameters

Purpose: The function of rvsXP Monitor and its related components may be influenced by changing parameter values. To edit the rvsXP parameters from the rvsXP-Administrator:
Controlling the rvsXP Monitor function

- rvsXP-Administrator -> Settings -> Parameter.

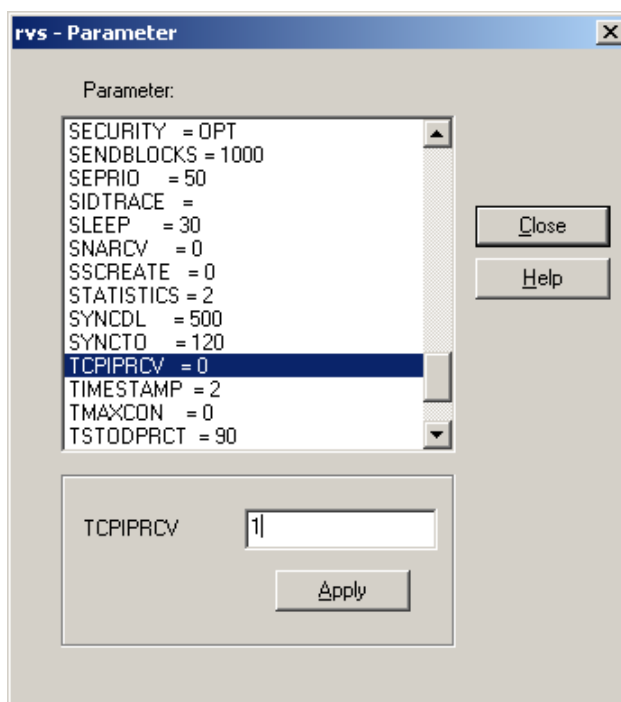
1.1 How to use rvs® parameters

Where do I find what? You can use a magnitude of parameters to customize rvs® processes. Chapter “rvs® parameter overview” provides a detailed description of all parameters. Tips how to choose rvs® parameter values are given in Chapter “Description of selected rvs® parameters”.

To modify a parameter:

- Execute the `Settings -> Parameters` command in the rvsXP Administrator.

The `rvs Parameters` dialog for editing the parameters will open. In order for the TCP/IP receiver to start automatically start when rvsXP Monitor starts, you have to set the value of the rvsXP parameter to **1**:



- Double click the parameter in the parameter list.
- Enter the desired value into the input field.
- Confirm your entry with **Apply**.
- Use **Close** to exit the parameter list.
- Activate the parameter by stopping and starting the rvsXP Monitor.

Alternative: You can also use the rvsXP-Administrator -> Edit -> Command level OperatorCommand command to display or edit a parameter.

Use `listparm NAME` in this case to list one or several parameter values.

NAME can be:

- the name of a parameter to list this one value,
- a wildcard to list all parameters whose names match this wildcard (wildcards are described in the following Chapter "Wildcards"),
or
- **ALL** to list all parameters.

You can modify the **NAME** parameter using the following command:

`setparm name=value`

ATTENTION No plausibility check is made on the value you specify. Thus, the results are unpredictable if you specify invalid values or wrong data types.

1.2 Wildcards

Multiple selection Some commands support wildcards. This means that the values you specify for these parameters may include wildcards asterisk (*) and/or question mark (?) to select more than one value at a time:

| | |
|---|---|
| * | Matches any number of arbitrary characters, |
| ? | Matches exactly one arbitrary character |

Wildcards must be enclosed in single or double quotation marks. For example, to list the execution priorities of all commands, enter

`listparm "*prio"`

and rvsXP Monitor will list the values of **BBPRIO**, **IEPRIO**, **IZPRIO**, etc.

When you specify the command as

`listparm "q?prio"`

you are returned the values **QEPRIO** and **QZPRIO**.

1.3 Command descriptions

Tip: Tracing in the Monitor log

You can type commands in `rvsXP-Administrator` and watch their execution in the Monitor log. To do so, choose `Edit -> OperatorCommand`.

Square brackets ([]) in the command descriptions indicate optional parameters. Curled brackets and pipes ({ | }) list alternatives.

| | |
|-----------------------|--|
| <code>Activate</code> | <code>SID=sid</code> |
| <code>Act</code> | is an alias of <code>activate</code> . Activates a partner station. An <code>rvsXP</code> communication program starts and establishes the connection. Queued data sets will be transmitted. |
| <code>Cleanup</code> | [<code>DAYS=n</code>] [<code>SS=YES</code>] Physically delete all commands from the database that have been executed (or logically deleted) at least <code>n</code> days ago, exactly <code>n x 24</code> hours. <code>cleanup DAYS=0</code> deletes all executed and deleted commands. If <code>SS=YES</code> is specified, all old <code>SendStatistics</code> records are deleted. |
| <code>Delcmd</code> | <code>CN=cn</code> Delete command with command number <code>cn</code> logically from database. |
| <code>Delst</code> | <code>SID=sid</code> Delete station ID (<code>sid</code>) from all station-table related database tables. |
| <code>Freecmd</code> | [<code>CN=cn</code>] [<code>SID=sid</code>] Free command with command number <code>cn</code> from hold status or release all suspended transfers to neighbor station <code>sid</code> . |
| <code>Holdcmd</code> | [<code>CN=cn</code>] [<code>SID=sid</code>] Put command with command number <code>cn</code> in hold status or suspend all transfers to neighbor station <code>sid</code> . |
| <code>Listdbv</code> | List version and creation date of the <code>rvsXP</code> database. |
| <code>Listcmd</code> | [<code>CN=n</code>] [<code>STATUS=x</code>] |

| | |
|----------|---|
| Lc | <p>Is an alias of <code>listcmd</code>.</p> <p>List details of command with number <code>n</code> or type and number of all commands whose status is <code>x</code>:</p> <ul style="list-style-type: none"> a active: The command is being processed d deleted: This entry was (logically) deleted e executed: Processing of the command ended f forwardable: ready to be sent (SK or QS) h held: The command was set to hold; it cannot be processed until released (i.e. set to status <code>q</code>). i in transit: The command is currently being sent (SK or QS) p pending: The command was partially completed. The command is waiting for some event (e.g. SE is in this status, after SKs have been created until all receipts are received; RE is always in this status, waiting for information that matches it to come in.) q queued: Ready to be processed s suspended: Ready to be sent but suspended, because all traffic to neighbor stations has been stopped (SK or QS). |
| Listparm | {name "pattern" ALL} |
| Lp | Is an alias of <code>listparm</code> ; lists one, more, or all rvsXP parameters. |
| Listst | SID= <code>sid</code> |
| Ls | <p>Is an alias of <code>listst</code></p> <p>Lists all station table entries involving station ID (<code>sid</code>).</p> |
| Opcmd | <p>[DSN=<code>dsn</code>] [CMD=<code>cmd</code>] [TIME=<code>hh</code> [:<code>mm</code>] [:<code>ss</code>]]]]</p> <p>[REPEAT=<code>hh</code> [:<code>mm</code>] [:<code>ss</code>]]]]</p> <p>Reads operator commands from external data set DSN which must be fully qualified; or executes an operator command specified in <code>cmd</code>; if TIME is specified, the operation will be rescheduled for the specified time of day; if REPEAT is specified, the command will be executed immediately and repeated (indefinitely) after the given interval has expired.</p> |
| Setparm | name= <code>value</code> |
| Sp | Is an alias of <code>setparm</code> ; modifies one rvsXP parameter. |
| Start | <p>[XMT] [CID=<code>console-id</code>]</p> <p>Start MasterTransmitter or one console.</p> |
| Stop | <p>[XMT] [RVS={END FORCE}] XMT=END CID={<code>console-id</code> <code>pattern</code>}]</p> <p>Stop rvsXP Monitor (normally or with FORCEDEND), MasterTransmitter, or one or more consoles.</p> <p><code>stop xmt</code> and <code>stop XMT=END</code> are equivalent commands.</p> <p><code>stop</code> without parameters; stops rvsXP-Monitor.</p> |
| System | <p>CMD="<code>cmd</code>"</p> <p>Pass command <code>cmd</code> to operating system for execution.</p> |

1.4 rvs[®] parameter overview

| | |
|---|---|
| ACTPCOUNT (parameter for ActivePanel only) | The interval after which the statistical information about the active lines will be updated. The unit is the percentage of the actual file size. Default: 10 |
| AECHECK | Checks authority to execute a command (integrated in rvsXP Monitor). Default: 0 (disabled) |
| BACKUPINT | Back-up interval in minutes Default: 720 |
| BBCREATE | Creation of user notifications (BB command) Default: 0 (disabled) |
| BBPRIO | Priority of user notifications (BB command) Default: 90 |
| BRICKOFTPTI | Time to wait for data by BRICK ISDN Adapter Default: 20 |
| CDWAIT | Time in seconds before executing an OFTP change direction after receiving a file. Default: 0 (disabled) |
| CMDDELETE | Removes each command and its related entries from rvsXP database as soon as command ends or is being deleted. Default: 1 (enabled) |
| CMDTRACE | Default: 0 |
| CNSMSGs | IDs of LOG messages to be sent to operator console. The following message codes are defined: A Action B Security E Error I Information L Line driver O ODETTE R Report S Severe error W Warning + Long messages Default: ABEILORSW+ |
| CODEIN | Default input code (local file) when creating a send job. Default: A |
| CODEOUT | Default output code (file at recipient) when creating a send job. Default: X |

- DTCONNnn** Wait periods until an unsuccessful connection attempt is repeated.
nn is the number of unsuccessful attempts (**CNTRETRY** in SK). There need not be a parameter for each value of **nn**; if a particular one is not defined, the next smaller one that is found will be used.
Format: MM/DD/YY HH:MM:SS
Defaults: Increasing time intervals, so that rvsXP will not be kept busy trying to reach a station that may be having hardware problems. For longer wait periods, minutes have been added to the defaults, so that retries will not occur at precisely the same time:
DTCONN01 "00/00/00 00:01:00"
DTCONN02 "00/00/00 00:02:00"
DTCONN03 "00/00/00 00:03:00"
DTCONN05 "00/00/00 00:05:00"
DTCONN07 "00/00/00 00:07:00"
DTCONN10 "00/00/00 00:10:00"
DTCONN15 "00/00/00 00:15:00"
DTCONN20 "00/00/00 00:20:00"
In addition, **DTCONN01** is the wait period for all other commands.
- EERP_IN** Send transmission (EERP = "End-to-End-Response") acknowledgement;
Values: **NEVER**: Partner does not send EERP, file dispatch ends with successful transmission. Do not wait for acknowledgement.
NORMAL: Wait for acknowledgement by partner after successful file transmission. File dispatch ends with receipt of EERP.
Default: **NORMAL**
- EERP_OUT** Receive transmission (EERP) acknowledgement
NORMAL: Prepare acknowledgement after successful reception of a file; send EERP only if a connection to the partner still exists, otherwise at next connection.
IMMEDIATE: Prepare acknowledgement after successful file reception. Establish connection if there is no connection to the partner and send EERP to the partner.
NEVER: Partner does not expect EERP. File reception ends with successful transmission. Do not acknowledge reception.
HOLD: Prepare acknowledgement after successful file reception. However, do not send acknowledgement but wait for operator release. Send EERP after release only if a connection to the partner still exists, otherwise at next connection.
HOLDIMMED: Prepare acknowledgement after successful file reception. However, do not send acknowledgement but wait for operator release. After the release: Establish connection if there is no connection to the partner and send EERP to the partner.
SYNC: Maintains the connection until the EERP has been generated (after successful delivery).
Default: **NORMAL**
- FORCEDEND** Halts rvsXP Monitor: Immediate cancellation, even if the transmitter and receiver are active.
Note: The Monitor will immediately stop if the parameter is set to "1".
Default: **0**
- IEPRIO** Priority of IE commands
Default: **50**

| | |
|-------------------|--|
| INITCMDS | Executes initialization commands Default: 1 (turned on) |
| IZPRIO | Priority of IZ commands Default: 40 |
| KEEPDAYS | Number of days, after which deleted and ended commands and their related information may be discarded during database cleanup Default: 7 |
| LDSNPRI | Send priority for long data sets Default: 50 |
| LID | Local station ID Default: LOC , supplied during rvsXP database initialization |
| LITRACELVL | Request line tracing (between OFTP and network): 0 No tracing 1 Minimum tracing (line driver events etc.) for station specified in parameter SIDTRACE 2 Detailed tracing (incl. hex dump of data) for station specified in parameter SIDTRACE 3 Detailed tracing for all stations Default: 0 |
| LMPRIO | Priority for LOG messages as external LM commands Default: 20 |
| MAXCMD | Maximum number of external commands read simultaneously Default: 10 |
| MAXRECL | Maximum record length for data sets with record format F or V to be received Default: 99999 |
| MAXSENDERS | Maximum number of concurrent senders. No sender will start if MAXSENDERS=0 Default: 1 |
| MAXX25RCV | Maximum number of concurrently active or prestarted <code>listening</code> receiver processes for X.25 communication Default: 0 |
| MSGPRIO | Send priority for operator to operator messages Default: 60 |
| NUMRLOGS | Number of <code>rlog.log</code> files that can be generated Default: NOLIMIT |
| NUMRLSTAT | Number of <code>rlstat.log</code> files that can be generated Default: NOLIMIT |
| OCREVAL | ODETTE <code>credit value</code> = window size of OFTP: Maximum number of sent blocks without confirmation Default: 99 |

| | |
|-------------------|--|
| ODTRACELVL | Request line tracing (between sender and OFTP): 0 No tracing 1 Minimum tracing (request names, only) for station specified in SIDTRACE . 2 Detailed tracing (parameter values etc.) for station specified in SIDTRACE . 3 Detailed tracing for all stations. Default: 0 |
| OEXBUF | ODETTE exchange buffer size in bytes (1019 for UNIX and 2048 for OS/400 , the largest ODETTE cmd (SFID)); max. size is 99999 bytes; Default: 2048 |
| OKPRIO | Priority for operator commands Default: 10 |
| ORETRY | Indicates the ODETTE error group for which a retry will be initiated after a request has been interrupted. Indication occurs using a bit field with each bit position assigned to a particular error code. 1 means: Restarting is performed 0 means: Restarting is suppressed The bits stand for the following errors: 1 Transmission is interrupted 2 File not found or cannot be opened 3 File cannot be read 4 "File size is too big" error code in SFNA permitted with repetition 5 "Unspecified reason" error code in SFNA permitted with repetition 6 "File size is too big" error code in SFNA not permitted with repetition 7 "Unspecified reason" error code in SFNA not permitted with repetition 8 "File size is too big" error code in EFNA 9 "Invalid record count" error code in EFNA 10 "Invalid byte count" error code in EFNA 11 "Access method failure" error code in EFNA 12 "Unspecified reason" error code in EFNA Default: 10111000111011111111 |
| OTIMEOUT | ODETTE time-out value (in seconds) Default: 600 |
| QEPRIO | Priority of QE commands Default: 30 |
| QSPRIO | Priority of QS commands should lie between MSGPRIO and SDSNPRIO Default: 30 |
| RECVBLOCKS | Number of buffers or records the receiver writes before closing temporary data set. Default: 1000 |
| RETRY | Parameter is no longer used in rvsXP. (Transmission repetition after unsuccessful attempt. Default: 00/00/00 00:01:00) |

| | |
|---------------------|--|
| RLCOMAXSIZE | Maximum file size for console messages <code>rlco.log</code> Default: 200000 |
| RLDBMAXSIZE | Maximum file size for logging of database actions <code>rldb.log</code> Default: 1000000 |
| RLOGMAXSIZE | Maximum file size for log messages <code>rlog.log</code> Default: 2000000 |
| ROUTING | Determines the global rvsXP routing behavior. O (OUT): Routing is allowed for send jobs only. The own local station must not function as a router. B (BOTH): Standard OFTP routing, routing is allowed in send and receive direction. Default: B |
| RSTATMAXSIZE | Maximum file size for statistical logs <code>rlstat.log</code> Default: 2000000 |
| SCPRIO | This parameter refers to the cooperation between rvsXP-Monitor and service provider. Change after consultation with rvs [®] Support only! Default: 10 |
| SDSNMAX | Maximum size for a data set to be considered short in Kbytes = units of 1024 bytes Default: 100 |
| SDSNPRIO | Send priority for short data sets Default: 40 |
| SECURITY | Use of encryption OPT : Optional. Use of encryption if NO : No encryption FORCED : Compulsory Default: OPT : |
| SENBLOCKS | Number of buffers or records that the Sender sends before looking at FORCEDEND again. Default: 1000 |
| SEPRIO | The priority for new SEs should be at least as high as the highest priority of valid for SE. Default: 50 |
| SIDTRACE | ID of station that shall be traced (if LITRACELVL or ODTRACELVL are at least set to 1 or 2 . Default is " " (3 blanks). If you need to trace incoming data, SIDTRACE must be set equal to the local station ID (LID). |
| SLEEP | The delay in seconds rvsXP Monitor waits during idling until it searches the database for new commands to be executed again. Default: 30 |

| | |
|-------------------|--|
| SNARCV | Enables start of SNA Transaction Program automatically on incoming calls: 0 No transaction program will start 1 The transaction program starts Default: 0 |
| SSCREATE | Creates a send statistics record for each transfer attempt Default: 0 (disabled) |
| STATISTICS | Creates a send statistics record in the statistics log file (<code>rlstat.log</code> for UNIX and Windows XP) 0 No statistics log file 1 Short form 2 Detailed form of statistics 3 Short form of statistics inclusive routed transfers 4 Detailed form of statistics inclusive routed transfers 5 New parameters such as file format, state of transmission, number of dial-in attempts 6 Statistics about deleted entries (also by the user) 7 Statistics about deleted entries and routed transmissions Default: 2 (detailed statistics enabled) |
| SYNCDL | When EERP_OUT=SYNC : Delay in ms until rvsXP checks whether EERP is available for dispatch. Default: 500 |
| SYNCTO | When EERP_OUT=SYNC : Number of wait procedures. Example: SYNCDL=400 and SYNCTO=5 . rvsXP waits for a maximum of 5 times 400 ms until the connection is closed. An EERP available for dispatch during this time will be sent and the connection closed thereafter. Default: 120 |
| TCPIPCV | Maximum number of (concurrently) prestarted listening processes for TCP/IP communication: 0 No TCP/IP receiver will be started 1 A TCP/IP receiver will be started Default: 1 |
| TIMESTAMP | Creation of a timestamp to distinguish data sets with the same data set name 0 000-999 (three-digit counter for MS DOS file names) 1 000000-999999 (counter) 2 Thhmmss (time) 3 Dyyymmdd.Thhmmss (date and time) 4 Thhmmsssmsms (date and time in milliseconds) Default: 2 (only time) |
| TMAXCON | Maximum number of simultaneously running operator consoles 0 no limit Default: 0 |
| TSTODPRCT | Percentage of non-error returns from ODETTE simulation program when rvsXP runs in test mode; -1 requests prompting for return values. Default: 90 |

| | |
|------------------|---|
| USEPKI | <p>rvsXP can obtain public keys from partners via LDAP Interface from a PKI (public key infrastructure). To use PKI instead of the rvsXP key memory for a particular station use the rvsXP parameter USEPKI.</p> <p>(Note: USEPKI is the global parameter. You can specify the usage of PKI with the PKI parameter for each station.)</p> <p>Possible values: USEPKI = YES USEPKI = NO Default: NO</p> |
| USERFIELD | <p>Reserved for future application.</p> |
| VDSNCHAR | <p>Range of allowable characters to be transferred within an ODETTE transmission:</p> <p>ALL: no restrictions</p> <p>OFTPUNIXS: All capital letters, digits and the following special characters: . -</p> <p>UNIX: all letters, digits and the following special characters: # _ - +.</p> <p>ODETTE: All capital letters, digits and the following special characters: () - . / &</p> <p>CHECK_RE: same as ALL but it is necessary that a RE exists</p> <p>Default: ODETTE</p> |

VFTYP

The way how files (with a fixed or variable format) will be converted prior to transmission.

- V** rvs[®] internal format, only useful for rvs[®] for variable and fixed formats
- S** Format of ft-SINIX, useful also for ft-SINIX
- T** Text format, each line is terminated by carriage return and line feed. Each line is converted into one output record. The record length is defined in **MAXRECL**.

Files in the **fixed** or **variable** format can be stored as text files. For this purpose you should use the **VFTYP** parameter. **VFTYP=T** means received files in format **F** or **V** will be stored as text files carrying a CR/LF at the end of the each line. You can also control this in the resident receive entry by setting the **VFTYP** parameter to **T**.

Files can be sent in the **fixed** or **variable** format without prior conversion by the `rvsut2fv.exe` utility. To do this, you can additionally specify the record length in the `New Send Entry` dialog. The record length for files with the **fixed** format is the length of each record before linefeed (CR/LF for MS Windows systems or LF for UNIX systems). This is the maximum record length for files of the **Variable** format. For this purpose the files must be available in the appropriate format; otherwise transmission errors will occur. Empty lines will be replaced with a record containing exactly one space.

The following optional parameters are available for sending files in the **fixed** or **variable** format with the rvs[®] batch interface:

VFTYP defines whether files in the **F** or **V** format are to be sent using the method mentioned before:

- T** The file is a text file and will be handled in line with the method described before

MAXRECL Maximum record length if **VFTYP=T** is set;

- V** rvs[®] internal format. Files in the **F** or **V** format have been formatted with `rvsut2fv.exe`.

Default: **V** (rvs[®] internal format)

XMCREATE

Creation of LOG messages with detailed information about what was transferred from and to whom after each successful data set send or receive process.

Default: **1** (enabled)

1.5 Description of selected rvs[®] parameters

rvsXP contains a number of optional and security related features which you may not need (all the time) at your installation. Each active function consumes resources (processor time and hard disk access). The sum of all functions can significantly affect the performance of rvsXP components.

Data transmission mechanism As an example, consider transmission of a large data set. To be able to resume transmission after a line failure without having to start at the beginning of the file all over again, the Receiver periodically closes the incoming data set. Both Sender and Receiver store the number of transmitted bytes or records in the database. The frequency of these actions is determined by parameters **SEENDBLOCKS** and **RECVBLOCKS**.

Reopening and positioning a large data set involves quite a number of disk access operations and therefore is very time consuming¹. So, if most of your communication lines are very stable, you may want to set these parameters to very large values effectively disabling the restart feature of rvsXP.

SEENDBLOCKS If, on the other hands, most of your lines tend to break down every few minutes, you will want to make sure that whatever has been transmitted once, will not have to be transmitted again. Note, that a large value of **SEENDBLOCKS** may also increase the time before senders terminate after Monitor has been stopped. Defaults have been chosen so that rvsXP will work securely and with most options enabled.

1.6 Safety, resource consumption and performance

RECVBLOCKS Besides **RECVBLOCKS** und **SEENDBLOCKS** which have been discussed above, there are several other parameters that influence the balance between safety, resource consumption, and performance.

OCREVAL **OCREVAL** (recommended window size 99) and **OEXBUF** (recommended size 4096 bytes) influence the overhead incurred by the ODETTE protocol. The higher these values the less overhead. They will, however, also increase the memory requirements for sender and receiver. These values may be negotiated down at the start of each transmission, so that unilateral changes may have no effect. What you really determine is the maximum amount of memory you are willing to allocate to ODETTE.

KEEPDAYS Searching in a large database generally takes longer than looking for something in a small one. A larger database, however, retains more information on completed transmissions. **KEEPDAYS** determines the

¹ Transmission time for a 4.5 MB data set between two OS/2 nodes was reduced by about a factor of 10 (from more than an hour to a few minutes) by changing the values of these parameters from **10** to **10000**.

number of days you want to keep information about ended or deleted transmissions (unless you use `cleanup days=n`, explicitly specifying the retention period in the command itself).

- CMDDELETE** and **XMCREATE** For **CMDDELETE = 1** all related entries will be removed physically from the database when a command ends or when it is (logically) deleted. This keeps the size of the database as small as possible. If you choose this option, you should leave **XMCREATE** at its default value (**1**). In this way, detailed LOG messages will be created after sending or receiving a data set. All users should have access to the log data set (`$RVSPATH2\db\rlog.log`) to be able to look at these messages, because the dialog interface will be unable to display any information about completed transfers. Consider using this option for continuous unattended operations.
- SLEEP** The rvsXP Monitor reaction time to new events is determined by **SLEEP**. This may influence for example, how long it takes before the Monitor starts acting on an operator command. **SLEEP** is the period of time (in seconds) that rvsXP Monitor is suspended when there is nothing to do for it. The longer you choose this period, the less it will interfere with your other applications, but the longer you may have to wait, before rvsXP Monitor starts processing your requests. The shorter you choose this period the higher is the unproductive overhead produced by scanning the rvsXP database when there is nothing to do.
- DTCONNxx** The time until rvsXP Monitor restarts an unsuccessful or aborted transmission is determined by the **DTCONNxx** parameters. The smaller these values, the sooner the transmission will start after the line is up again. However, the more computer time may have been wasted on unsuccessful attempts until the line is restored.

1.6.1 Limiting the number of concurrent senders

You may want to limit the number of senders that rvsXP is allowed to execute at the same time if your system is very busy or when you know that one or more of your neighbors cannot accept more than a few incoming calls at the same time.

- MAXSENDERS** **MAXSENDERS** tells MasterTransmitter `rvsxmt` how many Senders may run concurrently. When this number has been reached it waits until a Sender terminates before starting the next one. No Sender will be started at all if **MAXSENDERS** is set to **0**. This is useful if only the partner station is to establish the connection and fetch the queued data sets. Use the `activate` command to send data to a specific station even if **MAXSENDERS** is set to **0**.

² For more detailed information on the **\$RVSPATH** parameter see Chapter 1.6 "Representation means".

1.6.2 Limiting the number of concurrent X.25 or ISDN receivers

You must specify the number of concurrently active X.25 and ISDN receivers. A small number is adequate for low traffic, a higher number is required if you must be able to receive data on several connections in parallel. However, there cannot be more X.25 receivers than virtual channels are available on your X.25 multichannel. In the case of ISDN, there cannot be more receivers than B-channels are available. Because senders also occupy virtual channels or B channels in ISDN respectively, the number of concurrent receivers should be limited to half the total number of channels.

MAXX25RCV MasterTransmitter `rvsxmt` uses the **MAXX25RCV** parameter to determine how many receivers must run concurrently. It prestarts as many X.25 (ISDN) receivers as specified by the parameter. If a receiver terminates, MasterTransmitter will start a new receiver, which in turn will wait for incoming calls. **MAXX25RCV**, if set to **0**, prevents any incoming X.25 or ISDN traffic. It must be set to **0** if only TCP/IP communication is used.

You must define additional entries in the X.25 routing table on high-performance systems if **MAXX25RCV** is greater than **1**. For further details, read 4.1.4 "Setting up the routing connection".

1.6.3 TCP/IP receiver

TCPIPRCV and **MAX_IN** If you want to communicate via TCP/IP, rvsXP has to start a receiver waiting for incoming calls. You must set the value of the **TCPIPRCV** parameter. Set **TCPIPRCV** to **0** if you use TCP/IP only. If a TCP/IP receiver accepts an incoming call, MasterTransmitter will start a new receiver on the same port, which in turn will wait for incoming calls. The number of calls you can accept on each port corresponds to the **MAX_IN** values defined for your local station in your station table.

1.6.4 Optional features

Providing these optional services takes time and uses up disk space. We therefore recommend that you enable them as soon as they are no longer needed.

AECHECK **AECHECK** is a flag telling rvsXP Monitor to check whether the originator of the command currently being processed has the authority to issue this particular command. In a (future) multi-console environment, this could be used to prevent certain consoles from stopping the Monitor, for example. Currently, this feature is not fully supported, so **AECHECK** should remain at **0** (disabled).

BBCREATE User notification messages are generated and passed to the `rvsums` user exit if **BBCREATE** is enabled (**BBCREATE=1**). The default action is to send this message as UNIX mail (type `mail` to read the message). **?** prevents the creation of user notifications.

- SSCREATE** Statistics records will be created for every attempted transfer when flag **SSCREATE** is enabled (**SSCREATE=1**). These records contain the station ID of the neighboring station as well as time and completion code of the attempted (or completed) transfer. **SSCREATE=0** prevents generation of these records. Currently, no utility to analyze these records is provided.
- XMCREATE** **XMCREATE** (create xfer message) controls generation of detailed information about successful transfers in the `RVSPATH\db\rlog.log` system log file. **XMCREATE=1** (default) writes a log message in the following cases: Whenever a data set is successfully sent to a neighboring station (even before an acknowledgement has been received), whenever a send entry completes (after receiving acknowledgements from all recipients), and whenever a data set has been delivered to a local user. **XMCREATE=0** suppresses generation of these LOG messages.
- LITRACELVL** and **ODTRACELVL** When communication errors occur, helpful trace information can be found in the trace data sets, if the values of the **LITRACELVL** and **ODTRACELVL** parameters are larger than **0**. Tracing can dramatically reduce performance because a lot of data has to be analyzed, formatted and written into the trace file. For normal operations, tracing should be disabled, i.e. both parameters should be set to **0**.
- CNSMSG** **CNSMSG** controls, which LOG messages are echoed to the operator console. All messages, whose code letter is included in the character string value of **CNSMSG** are written to the console (all messages are always logged, independent of the value of **CNSMSG**). The additional message types **O** (ODETTE), **L** (line driver) and **(+)** (for long messages) can also be used.
- STATISTICS** **STATISTICS** controls the creation of the statistic log file. **STATISTICS=1** creates the (`RVSPATH3\db\r1stat.log`). It contains a line for each sent or received file with name, date, time and sender/receiver ID. **STATISTICS=2** creates the same file, but with extended information (e.g. the file name for the transmission (virtual file name), the file size and command numbers for SE, SK or IE, IZ). **STATISTICS=3** is the same as **STATISTICS=1**, but routed file transfer will also be logged (e.g. SID of destination station and SID of source station). **STATISTICS=4** is the same as **STATISTICS=2**, but routed file transfer will also be logged simultaneously. **STATISTICS=5** means a detailed output in (`RVSPATH\db\r1stat.log`) with new parameters such as file format, state of transmission and number of dial-in attempts. **STATISTICS=6** produces a more detailed output about deleted entries (by the user) with the cause of deletion (if specified with `delcmd`). **STATISTICS=7** has the same effect as **STATISTICS=6** but includes routing. **STATISTICS=0** prevents the creation of the statistics log file.

³ For more detailed information on the **RVSPATH** parameter see Chapter 1.6 "Representation means".

Appendix 2: Log files

Several log files are written during rvsXP operation. Their size and number can be controlled by means of rvsXP Monitor parameters (rvsXP- Administrator -> Settings -> Parameter).

2.1 Brief description of LOG files

The Monitor Log file `rlco.log`

System start messages In the Monitor Log view the `C:\rvs\temp\rlco.log` file is displayed. It is created from scratch each time rvsXP Monitor starts. Its maximum size is restricted by the **RLCOMAXSIZE** Monitor parameter. Only those message classes specified in the **CNSMSGs** Monitor parameter are written to this file.

The permanent `rlog.log` Monitor log file

rvsXP log messages All rvsXP log messages are written to the `C:\rvs\db\rlog.log` file, regardless of the **CNSMSGs** parameter. It will not be deleted when rvsXP Monitor restarts. Its maximum size is restricted by the **RLOGMAXSIZE** parameter. When the file reaches this size, it is renamed by replacing the file extension by a consecutive number. Then a new `C:\rvs\db\rlog.log` file is generated and updated. The maximum number of generations can be restricted by the **NUMRLOGs** monitor parameter.

The `rlstat.log` statistics file

For each sent/received file rvsXP writes statistics information to the `C:\rvs\db\rlstat.log` file. This file contains a line for each sent or received file with name, send/receive date and time and sender/receiver station ID. Other information such as transmission file name, file size, and command number can be logged for archiving purposes. The rvsXP **STATISTICS** parameter (rvsXP Administrator -> Settings -> Parameter -> STATISTICS) with values ranging from **1** to **7** controls the degree of detail in the statistics file.

Choose Settings -> Statistics to view the file contents. Its maximum size is restricted by the **RLSTATMAXSI** parameter. The file is renamed by assigning a consecutive number as extension and creating a new `C:\rvs\db\rlstat.log` file when this size is reached. You can restrict the maximum number of files generated with the **NUMRLSTATs** parameter.

2.2 Troubleshooting

Log file for
troubleshooting

The `rvssp` program, being a new addition to the `rvs`® set of programs, provides the compression and encryption features. This program saves protocol files in the **\$SPOUTDIR** folder and deletes them after successful job completion. In case of an error the program will not delete the protocol file, which is then available for troubleshooting.

The `rvsenv.dat` file defines the **\$SPOUTDIR** environment variable. You can also use the `rvsXP-Administrator` to set the value for this environment variable by choosing `Settings -> Environment`. For further information please refer to "Environment variables".

Appendix 3: Environment variables

You can adapt the rvs environment to meet your requirements, e.g. modification of the folder in which the Monitor stores the received files. To modify the rvsXP system environment:

- Choose Edit -> Environment.

The rvs - Environment dialog opens and displays the current environment settings:

The screenshot shows the 'rvs - Environment' dialog box with the following settings:

| | |
|----------------------|--------------------------|
| PATH: | c:\rvs |
| SYSTEM: | c:\rvs\system |
| DB: | c:\rvs\db |
| USRDAT: | c:\rvs\usrdat |
| IEMP: | c:\rvs\temp |
| INIT: | c:\rvs\init |
| SAMPLES: | c:\rvs\samples |
| KEYDIR: | c:\rvs\keydir |
| SPINDIR: | c:\rvs\temp\in |
| SPOUTDIR: | c:\rvs\temp\out |
| SPEFILESDIR: | c:\rvs\temp |
| SERVER: | butech-xpide |
| PORT: | 2956 |
| ARCDIR: | c:\rvs\arcdir |
| AP_DISPLAY_INTERVAL: | 1000 |
| DBNAME: | |
| DBUSER: | |
| DBPSW: | |
| DBLOG: | <input type="checkbox"/> |
| Language: | English |

- Confirm your changes / entries by pressing **OK**.

Note: Any changes will only take effect after rvsXP Monitor is restarted.

Meaning of the environment variables

| Designation | Meaning of the environment variable | Default value |
|----------------------------|--|----------------------------|
| PATH | Main rvsXP folder and location of <code>rvsenv.dat</code> environment file | C:\rvs |
| SYSTEM | System folder with all program, help, batch files and libraries | C:\rvs \system |
| DB | Folder of the rvsXP database | C:\rvs\db |
| USRDAT | Folder for user files received | C:\rvs \usrdat |
| TEMP | Temporary folder for the log book | C:\rvs \temp |
| INIT | Folder for the initialization files, e.g. <code>rdmini.dat</code> for rvsXP monitor parameters | C:\rvs \init |
| SAMPLES | Folder for files for the programming interface with examples | C:\rvs \samples |
| KEYDIR | Folder containing the private and public key files of local and partner stations | C:\rvs \keydir |
| SPINDIR | Internal Service Provider (SP) working folder containing the job files to be processed (compression, encryption, and code conversion). | C:\rvs \temp\in |
| SPOUTDIR | Internal Service Provider (SP) working folder containing the job processing result files (log files). You can use the log files for troubleshooting (see also Log files, Chapter "Troubleshooting"). | C:\rvs \temp\out |
| SPFILESDIR | Internal Service Provider (SP) working folder containing the job files used during job processing. | C:\rvs \temp |
| SERVER | URL (or name) of local server | IP address of the local PC |
| PORT | Server port number for line status messages | 2956 |
| ARCDIR | Folder for back-up files | C:\rvs \arcdir |
| AP_DISPLAY_INTERVAL | Interval in milliseconds, in which the display for active lines is updated | 1000 |

| Designation | Meaning of the environment variable | Default value |
|-----------------|--|---------------|
| DBLOG | Writes entries to the rvsXP database: Yes/No. rvsXP logs all database operation in the C:\rvs\db\rlstat.log file when you enable this checkmark. This file helps you restore the database entries that were created after the last database back-up. Attention: This will influence the rvsXP performance and resource requirements extremely | No |
| Language | Language for message display. | German |
| DBNAME | Name of database when using an SQL database. | |
| DBUSER | Name of database user when using an SQL database. | |
| DBPSW | Password for database access when using an SQL database. | |

Appendix 4: Glossary

ASCII

American **S**tandard **C**ode for Information **I**nterchange

Batch interface (rvsbat)

The rvs[®] batch interface provides functions for background processing.

CMX

Communications **M**anager **S**INIX

Dialog interface (rvsdia)

The rvs[®] dialog interface provides interactive user functionality.

EBCDIC

Extended **B**inary **C**oded **D**ecimal **I**nterchange **C**ode

EDI

Electronic **D**ata **I**nterchange

EDIFACT

Electronic **D**ata **I**nterchange for **A**dministration, **C**ommerce and **T**ransport

EERP

End-to-**E**nd-**R**esponse. ODETTE term for an acknowledgement at the end of a send transmission.

ETSI

European **T**elecommunications **S**tandardization **I**nstitute

FSS

Forwarding **S**upport **S**ervice

GNU zip algorithm

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software. Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

If you use the zlib library in a product, we would appreciate *not* receiving lengthy legal documents to sign. The sources are provided for free but without warranty of any kind. The library has been entirely written by Jean-loup Gailly and Mark Adler; it does not include third-party code.

If you redistribute modified sources, we would appreciate that you include in the file ChangeLog history information documenting your changes.

Communication module (rvscom)

The rvs[®] communication module connects different stations and sends or receives files.

MasterTransmitter (rvsxmt)

The MasterTransmitter of the rvs[®] system coordinates send and receive processes to ensure the optimal use of the net capacity.

Monitor (rvsmon)

rvs[®] monitor is the main component of the rvs[®] system. It controls transmissions sent and received and initiates automatic follow up jobs if necessary.

ODETTE

Organization for **D**ata **E**xchange by **T**ele **T**ransmission in **E**urope

Get the complete description of OFTP from

<http://www.odette.org/>

OFTP

ODETTE File Transfer Protocol

The ODETTE File Transfer Protocol is the definition of a file transfer protocol by the ODETTE Group IV for OSI Layers 4 to 7.

International Protocol used in many business fields (Industry, Commerce, Finance).

OSI

Open System Interconnection

PDF

Portable Document Format

Protocol

To connect two different computers they have to follow the same protocol. This protocol defines actions and reactions as well as the “language” spoken.

A

Acknowledgement
 During reception 40, A1-6
 During sending 39, A1-6
 Act (command) A1-3
 Activate
 Local station 29
 Partner station 29
 Station 59
 Activate (command) A1-3
 Activate (program) A1-14
 Active connection establishment (parameter) 44
 ACTPCOUNT (parameter) A1-5
 Adaptation
 License key 22
 Adapting
 Monitor parameters 67, 68
 AECHECK (parameter) A1-5, A1-15
 Allowed characters (parameter) 40
 AP_DISPLAY_INTERVAL (environment variable)
 A3-2
 Appendix A14-135
 ARCDIR (environment variable) A3-2
 ASCII 39, A4-1
 ASCII-EBCDIC conversion 119
 AUTODIAL (parameter) 44

B

Backing up
 Station 59
 Station table 96
 User table and follow-on processing 97
 Back-up
 Backing up rvsXP tables 96
 Backing up the station table 96
 Database, complete 97
 Overview 95
 Recommended procedure 95
 Restoring 100
 Restoring the rvsXP database 101
 Restoring the station table 100
 Restoring user table and follow-on processing
 100
 rvsXP 95
 User table and follow-on processing 97
 Write to log A3-3
 BACKUP (parameter) A1-5
 Basic functions 8
 BBCREATE (parameter) A1-5, A1-15
 BBPRIO (parameter) A1-2, A1-5
 Block size (parameter) 39, 42
 Blocks
 During reception 39

 During sending 39
 BRICKOFTPTI (parameter) A1-5

C

C-Cal interface 9
 CDWAIT (parameter) 44
 CDWAIT (parameters) A1-5
 Cleaning up see Cleaning up the rvsXP database
 Cleanup (command) 101, A1-3, A1-14
 Closed User Group (parameter) 52, 57
 CMDDELETE (parameter) 101, A1-5, A1-14
 CMDTRACE (parameter) A1-5
 CMX A4-1
 CNSMSGs (monitor parameters) A2-1
 CNSMSGs (parameter) A1-5, A1-16
 Code conversion 119
 During reception 124
 During sending 122
 Performing 122
 Code translation (parameter) 39
 CODEIN (parameter) 39, A1-5
 CODEOUT (parameter) 39, A1-5
 Command
 Cleanup A1-3
 Listparm A1-2, A1-4
 Command descriptions A1-3
 Commands
 Act A1-3
 Activate A1-3
 Cleanup 101, A1-14
 Delcmd A1-3
 Delst A1-3
 Freecmd A1-3
 Holdcmd A1-3
 Lc A1-4
 Listcmd A1-3
 Listdbv A1-3
 Listparm A1-2
 Listst A1-4
 Lp A1-4
 Ls A1-4
 opcmd A1-4
 Setparm 68, A1-2, A1-4
 Sp A1-4
 Start A1-4
 Stop A1-4
 System A1-4
 Communication module A4-2
 Communications Manager SINIX A4-1
 Configuration
 Key administration 103
 Configuration of rvsXP 31
 Connection (parameter) 43

Connection establishment 9

Active 44

Connection type 43

ISDN 15

TCP/IP 16

X.25 16

Creating

genKey 107

Jobstart entry 89

Resident receive entry 85

rvsXP database 98

Send order 73

User entry 92

CUG (parameter) 52, 57

D

Data administration see Managing the rvsXP database

Data conversion 9

Database

Backing up the complete database 98

Cleaning up 101

Reinitializing 98

Removing 98

Retention time of completed jobs 101

DB (environment variable) A3-2

DBLOG (environment variable) A3-3

DBNAME (environment variable) A3-3

DBPSW (environment variable) A3-3

DBUSER (environment variable) A3-3

Definition

Routing 45

Delay (parameter) 44

Delcmd (command) A1-3

Deleting

Key 114

Delst (command) A1-3

Dialog

Choosing a file to be sent 73

Display of not transmitted datasets 70

Program start after send attempt 89

Resident receive entries 85

rvsXP administrator 69

User 92, 93

Dialog interface A4-1

Starting 69

Directories 13

Display mode 70

Reception 71

Sending 71

Displaying

Jobstart entry 89

Received files 78

Refreshing 79

Resident receive entry 85

Send order 77

Transmissions 77

User entry 91

Displays

Refreshing 78

DTCONNnn (parameter) A1-6

DTCONNxx (parameter) A1-14

E

EBCDIC 39, A4-1

EBCDIC-ASCII conversion 119

EDI A4-1

EDIFACT A4-1

Editing

Jobstart entry 90

Resident receive entry 87

Send order 80

User 93

User entry 91

Editing (parameters) A1-1

EERP A4-1

EERP see also End-to-End-Response

EERP_OUT (parameter) 40, A1-6

EERP_IN (parameter) 39, A1-6

Electronic Data Interchange A4-1

Encryption 9, 10

and electronic signature 103

Asymmetrical method 103

Basics 103

Features 11

Key administration 107

rvsXP principle and procedure 104

Symmetrical method 103

End-to-End-Response 39, 45, A1-6, A4-1

Environment variable

AP_DISPLAY_INTERVAL A3-2

ARCDIR A3-2

DB A3-2

DBLOG A3-3

DBNAME A3-3

DBPSW A3-3

DBUSER A3-3

INIT A3-2

Input dialog A3-1

KEYDIR A3-2

Language A3-3

PATH A3-2

PORT A3-2

SAMPLES A3-2

SERVER A3-2

SPOUTDIR A2-2

SYSTEM A3-2
 TEMP A3-2
 USRDAT A3-2
 Environment variables A3-2
 Setting A3-1
 SPFILESDIR A3-2
 SPINDIR A3-2
 SPOUTDIR A3-2
 Equipment: See System requirements
 Establishment of an active connection 44
 ETSI A4-1
 Exit
 rvsXP 72
 Explanation of special terms A4-1

F

FACILITIES (parameter) 52, 56
 Features
 Encryption 11
 Offline compression 11
 Files
 rdkey.dat 22
 rdmini.dat 66, 68, 102
 rlco.log A2-1
 rlog.log A1-14, A1-16, A2-1
 rlstat.log A1-16, A2-1
 Flag
 SSCREATE A1-16
 Follow-on processing
 Backing up 97
 Restoring 100
 FORCEDEND (parameter) A1-6, A1-9
 Forwarding see Routing
 Forwarding Support Service A4-1
 Freecmd (command) A1-3
 FSS A4-1
 Function test rvsXP 25

G

genKey (program) 107
 Glossary A4-1
 GNU A4-2
 GNU zip algorithm 11, A4-2

H

Holdcmd (command) A1-3
 Holding
 Send order 80
 Holding (temporarily)
 Send order 80

I

IEPRIO (parameter) A1-2, A1-6
 Importing
 Key 111
 IN (parameter) 39
 INADDR (parameter) 48
 INIT (environment variable) A3-2
 INITCMDSD (parameter) A1-7
 Initializing
 rvsXP database 98
 Input code (parameter) 39
 Installation
 rvsXP 17
 Interface
 C-Cal 9
 Command prompt 9
 User 9
 IP address (parameter) 48
 ISDN (tab) 49
 ISDN connection 15, 43
 ISDN Facilities (parameter) 52
 ISDN number (parameter) 51
 ISDN protocol (parameter) 52
 ISDN reception 61
 ISDN UserData (parameter) 52
 ISDN_FAC (parameter) 52
 ISDN_PROT (parameter) 52
 ISDN_TEI (parameter) 52
 ISDN_USRDT (parameter) 52
 ISDNNO (parameter) 51
 IZPRIO (parameter) A1-2, A1-7

J

Job
 rvskeyimp 111
 Jobstart entry
 Creating 89
 Displaying 89
 Editing 90
 Removing 91

K

KEEPDAYS (parameter) A1-7, A1-14
 Key
 Creating 107
 Deleting 114
 Importing 111
 Listing 116
 Key administration 107
 Configuration 103
 Key features 8
 Key file

Sending 113
KEYDIR (environment variable) A3-2

L

Language (environment variable) A3-3
Launching
 rvsdiant.exe 70
 rvsmgr.exe 69
Lc (command) A1-4
LDSNPRIO (parameter) A1-7
License key
 Adapting 22
 NT 22
LID (parameter) A1-7
Line driver A1-5
Line type (neighboring station) 42
LINK (parameter) 52, 56
Link test 27
Listcmd (command) A1-3
Listdbv (command) A1-3
Listing
 Key 116
 User entry 91
listparm (command) A1-2, A1-4
listst (command) A1-4
LITRACELEVL (parameter) 68
LITRACELVL (parameter) A1-7, A1-9, A1-16
LMPRIO (parameter) A1-7
LOC see Local station
Local station 32
 Activate 29
 Managing 32
 Setting up 32
Log file for troubleshooting A2-2
Log files 9, A2-1
 Brief description A2-1
 rlco.log A2-1
 rlog.log A2-1
 rlstat.log A2-1
 Troubleshooting A2-2
Lp (command) A1-4
Ls (command) A1-4

M

Managing
 Local Station 32
 Partner station 32
MasterTransmitter A4-2
MAX_IN (parameter) 49
MAXCMD (parameter) A1-7
Maximum sessions (parameter) 49
MAXRECL (parameter) A1-7, A1-12

MAXSENDERS (parameter) A1-7, A1-14
MAXX25RCV (parameter) 50, 54, 61, A1-7, A1-15
Monitor 2, A4-2
 Starting 63
 Stopping 65
Monitor parameters 66
 Adapting 67, 68
 CNSMSGs A2-1
 NUMRLOGS A2-1
 NUMRLSTATS A2-1
 RLCOMAXSIZE A2-1
 RLOGMAXSIZE A2-1
 RLSTATMAXSI A2-1
MSGPRIO (parameter) A1-7
Multiple languages 9

N

N (parameter) 48, 53, 56
NAME (parameter) A1-2
Neighboring nodes (parameter) 46
Network link test 27
New user 92
NK line type (table) 35
Number
 Senders A1-14
 X.25 or ISDN receivers A1-15
NUMRLOGS (Monitor parameter) A2-1
NUMRLOGS (parameter) A1-7
NUMRLSTAT (parameter) A1-7
NUMRLSTATS (Monitor parameter) A2-1

O

OCREVAL (parameter) 39, A1-7, A1-13
ODETTE
 Explanation of terms A4-2
Odette A4-2
 File Transfer Protocol 37, A4-3
 Identification 37
ODETTE Id (parameter) 37
ODETTE parameters 37
ODETTE parameters (table) 35
ODETTEID (parameter) 37
ODTRACELVL (parameter) A1-8, A1-9, A1-16
OEXBUF (parameter) 39, 42, A1-8, A1-13
Offline compression 10, 117
 Configuration 117
 Features 11
OFTP A4-3
OKPRIO (parameter) A1-8
Online compression 9
Opcmd (command) A1-4
Open System Interconnection A4-3

Oracle link 131
 Orders See also Command
 ORETRY (parameter) A1-8
 OSI A4-3
 OTIMEOUT (parameter) A1-8
 OUT (parameter) 40
 Output code (parameter) 39

P

Parallel sessions (parameter) 44, 53, 56

Parameters

ACTPCOUNT A1-5
 AECHECK A1-5, A1-15
 Allowed characters 40
 AUTODIAL 44
 BACKUP A1-5
 BBCREATE A1-5, A1-15
 BBPRIO A1-2
 Block size 39, 42
 BRICKOFTPTI A1-5
 CDWAIT 44, A1-5
 Closed User Group 52, 57
 CMDDELETE 101, A1-5, A1-14
 CMDTRACE A1-5
 CNSMSGs A1-5, A1-16
 Code translation 39
 CODEIN 39, A1-5
 CODEOUT 39, A1-5
 Compression 40
 Connection 43
 Connection type 42
 CUG 52, 57
 Delay 44
 DTCONNnn A1-6
 DTCONNxx A1-14
 Editing A1-1
 EERP_IN 39, A1-6
 EERP_OUT 40, A1-6
 Establishment of an active connection 44
 FACILITIES 52, 56
 FORCEDEND A1-6, A1-9
 genKey 108
 IEPRIO A1-2, A1-6
 IN 39
 INADDR 48
 INTCMDS A1-7
 Input, code translation 39
 IP address 48
 ISDN facilities 52
 ISDN number 51
 ISDN protocol 52
 ISDN user data 52
 ISDN_FAC 52

ISDN_PROT 52
 ISDN_TEI 52
 ISDN_USRDT 52
 ISDNNO 51
 IZPRIO A1-2, A1-7
 KEEP_DAYS A1-7, A1-14
 LDSNPRIO A1-7
 LID A1-7
 LINK 52, 56
 LITRACELEVEL 68
 LITRACELVL A1-7, A1-9, A1-16
 LMPRIO A1-7
 MAX_IN 49
 MAXCMD A1-7
 Maximum sessions 49
 MAXRECL A1-7, A1-12
 MAXSENDERS A1-7, A1-14
 MAXX25RCV 50, 54, 61, A1-7, A1-15
 MSGPRIO A1-7
 N 48, 53, 56
 NAME A1-2
 Neighboring nodes 46
 NUMRLOGS A1-7
 NUMRLSTAT A1-7
 OCREVAL 39, A1-7, A1-13
 Odette 37
 Odette Id 37
 ODETTEID 37
 ODTRACELVL A1-8, A1-9, A1-16
 OEXBUF 39, 42, A1-8, A1-13
 OKPRIO A1-8
 ORETRY A1-8
 OTIMEOUT A1-8
 OUT 40
 Output, code translation 39
 Parallel sessions 44, 53, 56
 PHONE 36
 Phone 36
 PKI 41, A1-11
 PORT 49
 Port 49
 PRIORITY 44, 46
 Priority 44, 46
 PROTOCOL 43
 PSESSIONS 44
 PSWFROM 37
 PSWTO 37
 QEPRIOR A1-2, A1-8
 QSPRIOR A1-8
 QZPRIOR A1-2
 Rcv Timeout 53, 56
 RCVTIMEOUT 53, 56
 RECBLOCKS 39
 Receive 37, 39

Receiver 43
 Receiver number 48, 53, 56
 RECVBLOCKS A1-8, A1-13
 Remote 36
 RLCOMAXSIZE A1-9
 RLDBMAXSIZE A1-9
 RLOGMAXSIZE A1-9
 ROUTING A1-9
 RSTATMAXSIZE A1-9
 rvskeydel 114
 rvskeydst 113
 rvskeyimp 112
 SCPRIO 9
 SDSNMAX A1-9
 SDSNPRIO A1-9
 SECURITY A1-9
 Self test via 43
 Send 37, 39
 SENDBLOCKS 39, A1-9, A1-13
 SEPRIOR A1-9
 SESSIONS 53, 56
 SID 36
 SIDNEIGHB 46
 SIDTRACE A1-7, A1-9
 SIDTYP 36
 SLEEP A1-9, A1-14
 SNARCV A1-10
 Station 36
 Station ID 36
 Station name 36
 STATISTICS A1-10, A1-16
 STATNAME 36
 SYNCDL A1-10
 SYNCTO A1-10
 TCPIPRCV 48, 60, A1-10, A1-15
 TEI 52
 TIMEOUT 52, 56
 Timeout 52, 56
 TIMESTAMP A1-10
 TMAXCON A1-10
 TSTODPRCT A1-10
 USEPKI A1-11
 USERDATA 52, 56
 VC 57
 VDSNCHAR 40, A1-11
 VFTYP A1-12
 Virtual 36
 Virtual circuit 57
 Window size 39
 X.25 address 51, 55
 X.25 Facilities 52
 X.25 facilities 56
 X.25 UserData 52
 X.25 UserData (rvsNT) 56

XADDRESS 51, 55
 XMCREATE A1-12, A1-14, A1-16
 Parameters, obsolete
 RETRY A1-8
 Parameters, reserved
 USERFIELD A1-11
 Partner station
 Activate 29
 Managing 32
 Setting up 32
 Password administration 38
 PATH (environment variable) A3-2
 PDF A4-3
 Performing
 Code conversion 122, 124
 PHONE (parameter) 36
 Phone (parameter) 36
 PKI (parameter) 41, A1-11
 Platforms 8
 PORT (parameter) 49
 Portable Document Format A4-3
 PRIORITY (parameter) 44, 46
 Priority (parameter) 44, 46
 Program
 Activate A1-14
 Key 107
 rvsbat A4-1
 rvscom A4-2
 rvsdia A4-1
 rvskeydel 114
 rvskeydst 113
 rvskeylst 116
 rvsmon A4-2
 rvssp A2-2
 rvsxmt A1-14, A4-2
 Properties
 Station 33, 36, 58
 Protocol A4-3
 PROTOCOL (parameter) 43
 Protocol files 9
 PSESSIONS (parameter) 44
 PSWFROM (parameter) 37
 PSWTO (parameter) 37

Q

QEPRIO (parameter) A1-2, A1-8
 QSPRIO (parameter) A1-8
 QZPRIO (parameter) A1-2

R

RCVTIMEOUT (parameter) 53, 56
 rdkey.dat (file) 22

- rdmini.dat (file) 66, 68, 102
- RECBLOCKS (parameter) 39
- Receive (parameter) 37, 39
- Received files
 - Displaying 78
- Receiver (parameter) 43
- Receiver number (parameter) 48, 53, 56
- Reception
 - Resident receive entry 84
 - With code conversion 124
 - with ISDN 61
 - with TCP/IP 60
 - with X.25 61
- Record formats 9
- Recreating
 - rvsXP database 98
- RECVBLOCKS (parameter) A1-8, A1-13
- Refreshing
 - Display 79
 - Displaying 78
- Release
 - Station 59
- Releasing
 - Send order 81
- Remote (parameter) 36
- Removing
 - Jobstart entry 91
 - Resident receive entry 88
 - Send order 81, 82
 - User 93
 - User entry 93
- Renaming
 - Station 58
- Replacement markers 84
- Requirements 12
- Resident receive entry
 - Creating 85
 - Displaying 85
 - Editing 87
 - Receiving 84
 - Removing 88
- Restarting
 - Send order 81
- Restoring
 - Database 101
 - rvsXP database 101
 - Station table 100
 - User table and follow-on processing 100
- Resumed transmission
 - Automatic 9
- RETRY (parameter, obsolete) A1-8
- RLCOMAXSIZE (monitor parameter) A2-1
- RLCOMAXSIZE (parameter) A1-9
- RLDBMAXSIZE (parameter) A1-9
- rlog.log (file) A1-14, A1-16, A2-1
- RLOGMAXSIZE (Monitor parameter) A2-1
- RLOGMAXSIZE (parameter) A1-9
- rlstat.log (file) A1-16, A2-1
- Route (tab) 46
- Routing 45
 - Set-up 45
- ROUTING (parameter) A1-9
- RSTATMAXSIZE (parameter) A1-9
- RT routing (table) 35
- rvs® batch interface 9, A4-1
- rvs® Monitor 9, A4-2
 - Specify 18
 - Starting 20
 - Startup mode 18
- rvs® parameter
 - Working with A1-1
- rvs® parameters A1-13
- rvsbat A4-1
- rvsbat (program) A4-1
- rvscom A4-2
- rvscom (program) A4-2
- rvsdia A4-1
- rvsdia (program) A4-1
- rvsdiant.exe
 - Launching 70
 - Starting 70
- rvskeydel (program) 114
- rvskeydst (program) 113
- rvskeyimp (program) 111
- rvskeylst (program) 116
- rvsmgr.exe
 - Launching 69
 - Starting 69
- rvsmon A4-2
- rvsmon (program) A4-2
- rvssp (program) A2-2
- rvsTraylcon 31
- rvsxmt A4-2
- rvsxmt (program) A1-14, A4-2
- rvsXP
 - Configuration 31
 - Display mode 70
 - Exiting 72
 - Installation 17
 - Sending a file 25
 - Starting 70
 - Window areas 71
- rvsXP administrator
 - Starting 32, 69
- rvsXP back-up 95
- rvsXP database
 - Cleaning up 101
 - Managing 95

rvsXP database see Database
rvsXP function test 25
rvsXP HighPerformance 131
rvsXP Monitor A1-4
 Changing startup type 18
 Starting 63
 Startup type 18, 19
 Stopping 65
rvsXP Monitor parameters 66
 Adapting 67, 68
rvsXP test 25
rvsXP Traylcon 32

S

SAMPLES (environment variable) A3-2
Saving
 User 93
SCPRIO (parameter) A1-9
SDSNMAX (parameter) A1-9
SDSNPRIO (parameter) A1-9
SECURITY (parameter) A1-9
Selecting display criteria 82
Self test via (parameter) 43
Send
 File 25
Send (parameter) 37, 39
Send order
 Creating 73
 Displaying 77
 Editing 80
 Holding 80
 Releasing 81
 Removing 81, 82
 Restarting 81
 Temporarily hold 80
SENDERBLOCKS (Monitor parameter) 66
SENDERBLOCKS (parameter) 39, A1-9, A1-13
Senders
 Number A1-14
Sending
 File 73
 Key file 113
 With code conversion 122
Sending a file 25, 73
SEPRIO (parameter) A1-9
Serialization 10, 77
SERVER (environment variable) A3-2
Services
 rvs® Monitor 18
 Starting 20
 Startup type 19
SESSIONS (parameter) 53, 56
setparm (command) 68, A1-2, A1-4

Setting
 Environment variables A3-1
Set-up
 Local station 32
 Partner station 32
 Routing 45
SID A1-4
SID (parameter) 36
SIDNEIGHB (parameter) 46
SIDTRACE (parameter) A1-7, A1-9
SIDTYP (parameter) 36
SLEEP (parameter) A1-9, A1-14
SNARCV (parameter) A1-10
Sorting
 User entry 91
Sp (command) A1-4
SPFILESDIR (environment variable) A3-2
SPINDIR (environment variable) A3-2
SPOUTDIR (environment variable) A2-2, A3-2
SQL server link 131
 Requirements 134
SSCREATE (flag) A1-16
SSCREATE (Parameter) A1-10
ST station (table) 35
Start
 rvsTraylcon 31
 rvsXP administrator 32
Start (command) A1-4
Starting
 Dialog interface 69
 rvsdiant.exe 70
 rvsmgr.exe 69
 rvsXP 70
 rvsXP administrator 69
 rvsXP Monitor 63
State
 of jobs 71
Station
 Activate 59
 Backing up 59
 Properties 33, 58
 Release 59
 Renaming 58
 Setting the status 59
 Status change 59
 Stopping 59
Station (parameter) 36
Station (properties) 36
Station ID (parameter) 36
Station list 73
Station name (parameter) 36
Station parameters
 Virtual 36
Station status

- Editing 59
- Setting 59
- Station table
 - Backing up 96
 - Restoring 100
- StationsConfigFile 129
- STATISTICS (parameter) A1-10, A1-16
- STATNAME (parameter) 36
- Status display 71
- Stop
 - rvsXP processes A1-4
- Stop (command) A1-4
- Stopping
 - Monitor 65
 - Station 59
- SYNCDL (parameter) A1-10
- SYNCTO (parameter) A1-10
- System (command) A1-4
- SYSTEM (environment variable) A3-2
- System requirements 15

T

- Tab
 - ISDN 49
 - Route 46
 - TCP/IP 47
 - X.25 53
- Table
 - NK line type 35
 - ODETTE parameters OP 35
 - Routing table RT 35
 - ST station 35
 - TC-TCP/IP 35
 - XP-X.25/ISDN 35
- TCP/IP (tab) 47
- TCP/IP connection 16
- TCP/IP reception 60
- TCIPRCV (parameter) 48, 60, A1-10, A1-15
- TC-TCP/IP (table) 35
- TEI (parameter) 52
- TEMP (environment variable) A3-2
- Test rvsXP 25
- TIMEOUT (parameter) 52, 56
- TIMESTAMP (parameter) A1-10
- TMAXCON (parameter) A1-10
- Transmission buffer 39
- Transmission time 10
- Transmissions
 - Displaying 77
- TrayIcon see rvsTrayIcon
- Troubleshooting A2-2
- TSTODPRCT (parameter) A1-10
- Typographic conventions 13

U

- USEPKI (parameter) A1-11
- User
 - Backing up 97
 - Editing 93
 - Removing 93
 - Saving 93
- User entry
 - Creating 92
 - Displaying 91
 - Editing 91
 - Listing 91
 - Removing 93
 - Sorting 91
- User interface 9
- User rights 20
- User table
 - Backing up 97
 - Restoring 100
- USERDATA (parameter) 52, 56
- USERFIELD (parameter, reserved) A1-11
- USRDAT (environment variable) A3-2

V

- VC (parameter) 57
- VDA see "Verband der Deutschen Automobilindustrie"
- VDSNCHAR (parameter) 40, A1-11
- Verband der Automobilindustrie 38
- VFTYP (parameter) A1-12
- Virtual (parameter) 36
- Virtual Circuit (parameter) 57
- Virtual station
 - Editing 58
 - Interactive operation 86
 - Receiving at 57
 - Removing 58
 - Sending from 58
 - Setting up 36, 57
 - Statistics 16
 - Virtual channels 15
 - Virtual dataset name 86, 90
 - What is...? 57

W

- What is rvs® 7
- What rvs® is not 8
- Wildcards A1-2
- Window areas 71
- Window size (parameter) 39
- Working with rvs® parameters A1-1

X

X.25 (tab) 53
X.25 address (parameter) 51, 55
X.25 communication 16
X.25 connection 16, 43
X.25 Facilities (parameter) 52, 56
X.25 or ISDN receivers
 Number A1-15
X.25 reception 61

X.25 UserData (parameter) 52, 56
XADDRESS (parameter) 51, 55
XMCREATE (parameter) A1-12, A1-14, A1-16
XP-X.25/ISDN (table) 35

Z

ZIP algorithm A4-2