



## **xTrace Service and Host Installation Guide**

Version 2.5.7  
Copyright Meisner IT 2008-2017



# Installation Guide

---

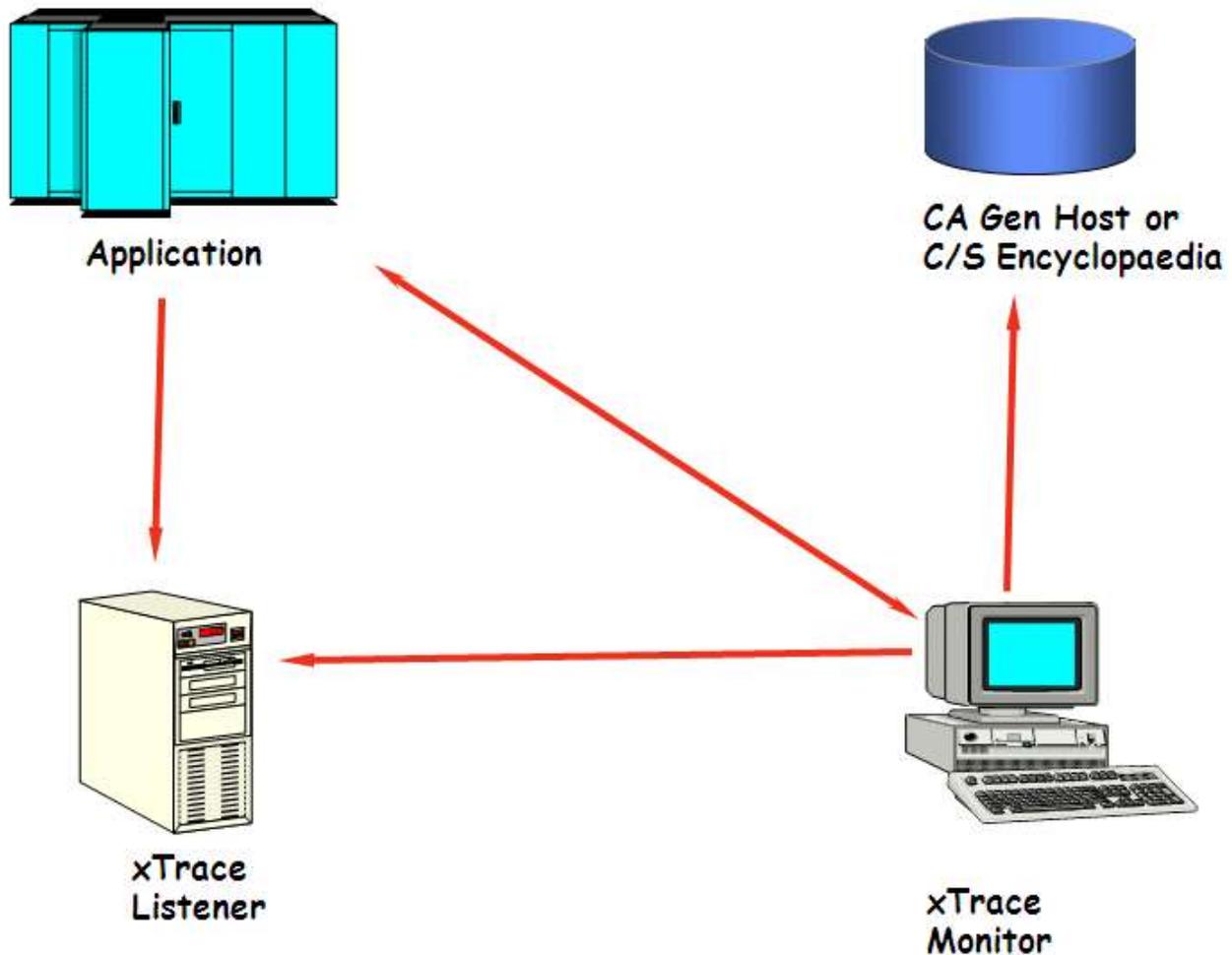
## Overview

This chapter gives an overview of the components of xTrace. Each component is described in detail in the next chapter.

xTrace is a replacement for CA Gen's trace facility for z/OS and differs in a number of ways:

- xTrace is written in C and uses TCP/IP sockets for communication. This makes tracing available for all environments supported by CA Gen on z/OS (CICS, IMS and Batch)
- In xTrace, the trace monitor is detached from the application which makes tracing seamless across platforms

The xTrace installation consists of four components as shown in the figure below. The arrows show the direction requests are sent between each component.





# Installation Guide

---

- The *xTrace listener* is a service running on a Windows server or z/OS Started Task. The listener controls all trace requests. The xTrace listener is also known as xTservice.
- The *xTrace monitor* is a GUI application running on the developer's Windows Workstation
- The *Encyclopedia* (Host or Client/Server) is accessed primarily for breakpoint definition. The Encyclopedia is accessed through ODBC direct from the developer's workstation. xTrace also uses a number of tables for storing user information. These are defined in the same database as the encyclopaedia.
- The xTrace part of the *Application program* is a replacement for the CA Gen trace runtime module TIRTRCE for z/OS generated applications.

## Component responsibility

### ***xTrace Listener***

The listener is the central controller of xTrace. It assigns tcp/ip port numbers for the monitor and the application and controls that each application is only traced by one monitor and that each monitor only traces one application at a time.

After accepting a connection from the monitor and the application, the listener will pass the tcp/ip address and port number of the application to the monitor and vice versa.

Parameters for the listener are defined in the xTservice.ini file on Windows or JCL(XTSVCINI) on the host.

### ***xTrace Monitor***

The monitor is used by the developer to control the application in trace mode and to define breakpoints.

### ***Application***

xTrace for z/OS is split into 2 modules:

- TIRTRCE is a replacement for the CA Gen TIRTRCE runtime module. It is statically linked into the application program.
- XTRACE performs all trace logic and is called dynamically from TIRTRCE

### ***Encyclopedia***

The encyclopedia is accessed through a series of database views giving the same "interface" on the CE and CSE.

Besides the encyclopedia views, xTrace uses a number of tables, primarily for storing breakpoint definitions.



# Installation Guide

## xTrace Installation

### Upgrading from a previous release

If you are upgrading from xTrace 2.1.2 or earlier, please note that new versions of all four components (Listener Service, Host Software, Database Views & Monitor) are required.

If you are upgrading from xTrace 2.1.3, new versions of the Host Software, Database Views & Monitor are required but the Listener Service does not require updating.

If you are upgrading from xTrace 2.2.\* or above, new versions of the Host Software, Monitor and xTservice are required but the Database Views do not require updating.

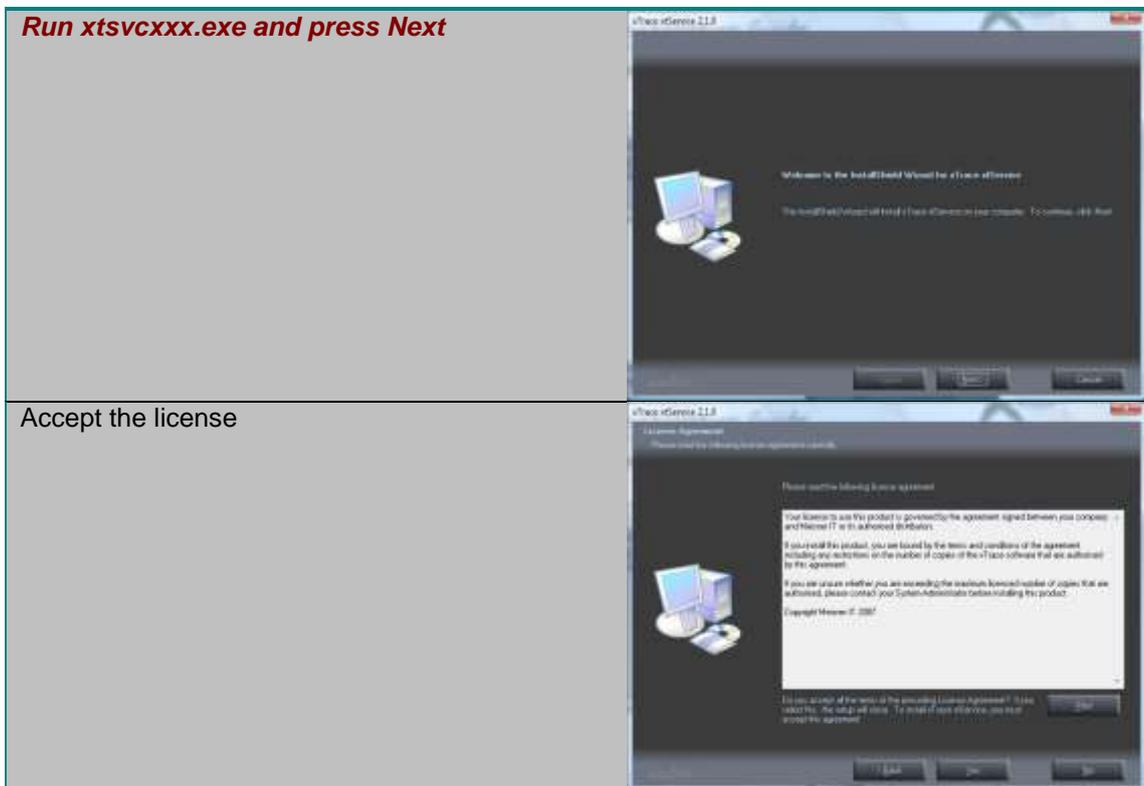
If you are upgrading from xTrace 2.5.2 or above, new versions of the Monitor, XTRACE host runtime and xTrace post processor only are required.

If you are using the Windows xTservice, it is recommended that you install the xTservice Listener software in the same directory as the previous version. This will require you to first stop the xTservice windows service. You do not need to re-register the service.

It is recommended that you install the host software into a separate set of libraries (i.e. XTRACE.V257.\*). Note that you will need to ensure the new version of the XTRACE load module is used by both on-line and batch applications. Since this is a dynamically called module, the applications do not require re-installation.

### Install xTrace Server Components

If you are installing the Windows xTservice, copy xtsvcxxx.exe to the server that is going to run xTservice (where xxx is the xTrace version number, e.g. xtsvc257.exe). Alternatively you can run xtsvcxxx on your workstation to extract the z/OS software components.



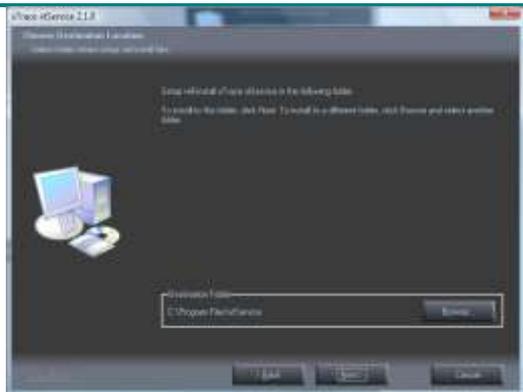


# Installation Guide

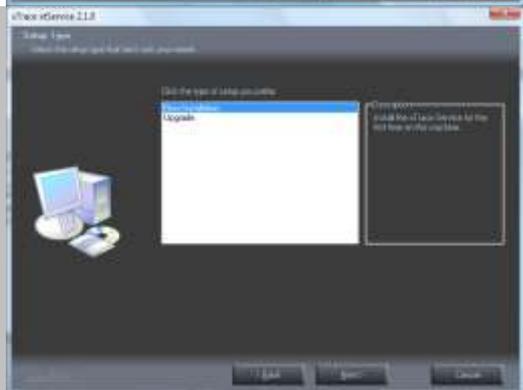
Select a destination folder where you want to install the software.

If you are installing an upgrade, it is recommended that you select the same location as the previous version.

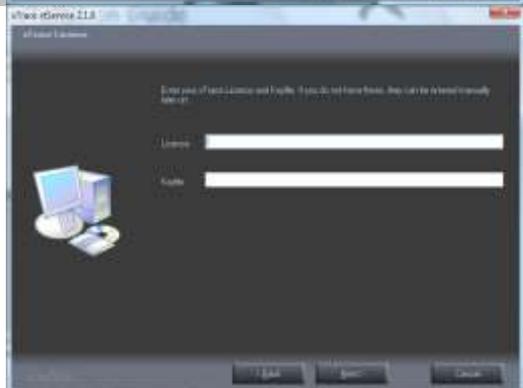
If you are using the Windows xTservice, ensure that you have first stopped the service



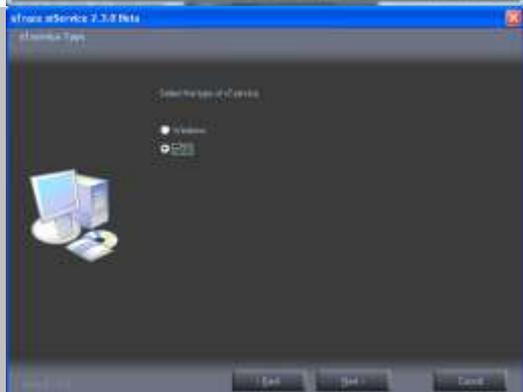
Select *New Installation* if this is the first time you are installing xTrace or *Upgrade* if you are upgrading to a new release



Enter the license and keyfile that you have received.



Specify whether you want to use a Windows or z/OS xTservice



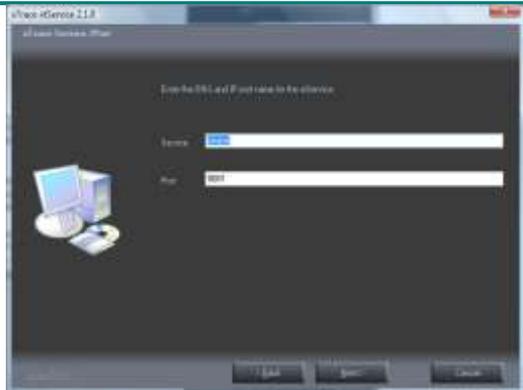


# Installation Guide

Enter the address of xTservice and the port number on which xTservice will listen for requests.

Service may be in the form of a DNS name that is resolved in your TCP/IP network or a TCP/IP address for the server that runs xTservice.

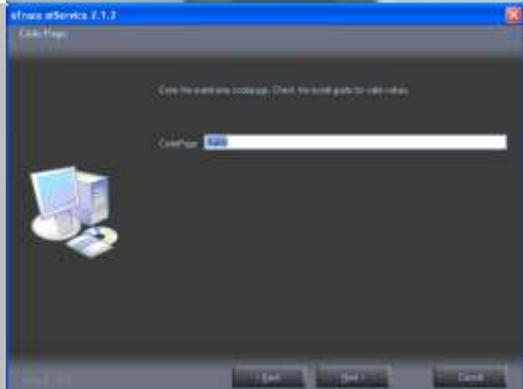
Remember this service and port number as you MUST specify the same values in the client and host installation.



Enter the host MVS codepage value. This should be in the form CPnnn where nnn is one of:

37,273,277,278,284,285,297,420,500,860,1047 or 1141 to 1149

If your codepage is not in the list, contact us since we can provide an alternate codepage for you.



The installation automatically uploads the installation files to your z/OS system.

You need to supply:

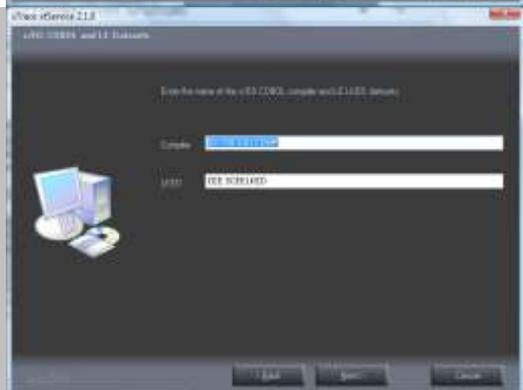
- IP address or DNS name of your z/OS system
- Userid that will be installing xTrace
- Dataset high level qualifier used for installation datasets



The installation process will ask for information to tailor the install jobs.

Enter dataset names for:

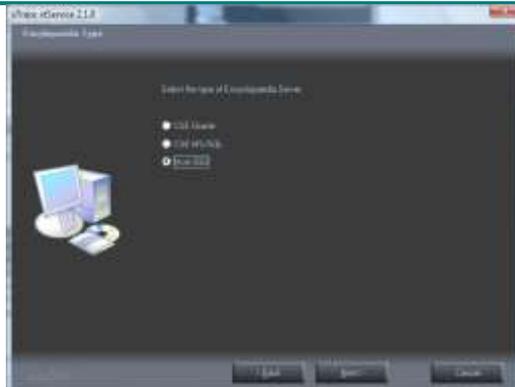
- COBOL compiler load library
- LE Link Edit library



# Installation Guide

Select your type of encyclopedia.

Depending on your selection, a number of screens with database selections will be displayed.



For Oracle CSE installation specify these variables:

- Oracle SID where the database is installed
- Userid and password for an account with the authority to install the database (default system/manager)
- Userid and password for the schema used to connect to the encyclopedia database (default (ency/ency))
- Enter the userid and password for the schema that will own the xTrace database (default xtrace/xtrace)
- Specify the name and location of the database files. These should be local to the database machine, so if the CSE database resides on a separate machine to xTservice, enter the value as it appears on this machine.



For MS SQL Server installation specify these variables:

- SQL server version used (7.0, 2000 or 2005)
- The SQL server name
- Database name where the xTrace database is to be installed (should be the same as the encyclopedia)
- Userid and password of an account with authority to install the database (default sa/xxxxx)



For a host encyclopedia installation specify these variables:

- DB2 subsystem
- SQLID owning the encyclopedia tables
- Name of the xTrace database
- SQLID for creating the xTrace database. See the section below on ODBC access to the xTrace tables and



# Installation Guide

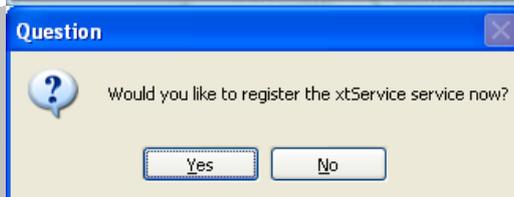
views on the Host Ency.

- Storage group used for creating table- and index spaces. This storage group must already exist since it will not be created by the installation jobs.

## Confirm your selections



If you are running xTservice on Windows, the xTservice may be started manually or be registered as a Windows service.



To register the Windows service select Yes. If you have already installed the xTservice into the same location, you do not need to re-register the service.

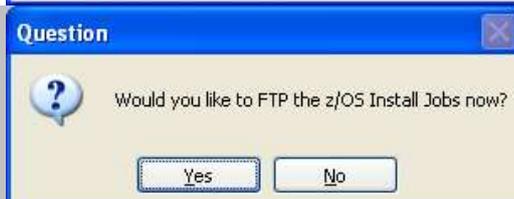
If you are installing or upgrading the database on this same machine as xTservice, select Yes. If the database is to be installed on a separate CSE machine, select No and see the section below on database install.



The installation may file transfer the z/OS installation jobs.

Select Yes to do this.

If you wish to do this manually, this is described later

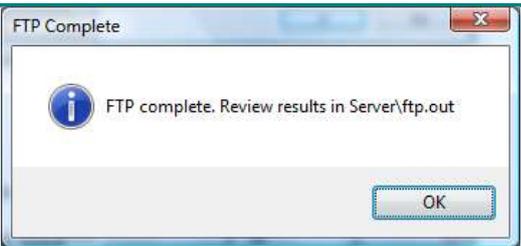


If you selected Yes to file transfer, you need to supply a password for FTP login.

Once the transfer is complete, review the output in the <install directory>\server\ftp.out file to verify that the files were transferred to the host.



# Installation Guide

	 <p>FTP Complete</p> <p>FTP complete. Review results in Server\ftp.out</p> <p>OK</p>
<p>If you are running a Windows xTservice, Setup will prompt you to install the Visual C 2008 runtime if this is not yet installed.</p>	
<p>You may need to restart your computer, if so, do this at a convenient point.</p>	 <p>Visual C++ 2008 Runtime</p> <p>Install Visual C++ 2008 Runtime</p> <p>The Visual C++ 2008 Runtime is required for many applications. Before you can run the program, you must install the runtime.</p> <p><input checked="" type="radio"/> Yes, I want to install the runtime now</p> <p><input type="radio"/> No, I will install by computer later</p> <p>Thank you. You can close this dialog and then click OK to complete setup.</p>



# Installation Guide

## Install z/OS Software

Installation files must be uploaded to your target z/OS server

### Upload installation files

During installation of the xTservice, you may have asked for the file transfer to be performed during the installation. If so, check that the file transfer completed successfully, otherwise perform the file transfer now:

In the installation directory of xTservice you will find a sub-directory called server.

- 1) Open a DOS prompt
- 2) Change to this directory
- 3) Type `xTraceinstall.cmd <password>` where `<password>` is the z/OS password of the installer userid specified when installing the software.

Check that you now have 13 datasets uploaded to z/OS:

```
<prefix>.JCL  
<prefix>.XMTBATL  
<prefix>.XMTBATLD  
<prefix>.XMTBATN  
<prefix>.XMTBATND  
<prefix>.XMTCICL  
<prefix>.XMTCICLD  
<prefix>.XMTCICN  
<prefix>.XMTCICND  
<prefix>.XMTIMSL  
<prefix>.XMTIMSLD  
<prefix>.XMTIMSN  
<prefix>.XMTIMSND
```

### Extract installation files

The JCL installation dataset contains one member: JOB01.

This job allocates the rest of the installation datasets and extracts the installation files.

Submit JOB01.

### Compile user exits

At runtime xTrace needs to know the TCP/IP socket address of where to find xTservice.

The COBOL source has been tailored by the installation process and should not need modification.

If you need to change the defaults you entered, modify the exits in the COBOL source dataset.

```
EXITCICS for CICS  
EXITIMS for IMS  
EXITBATC for batch
```

Then submit JCL(EXITx) to compile the exits where

- C is CICS



- I is IMS
- B is batch

You should always compile the batch exits and the CICS / IMS exits if using xTrace for on-line CICS and/or batch applications. This needs to be done even if you have not customised the source.

A description of the user exit parameters can be found in Appendix B.

## Install DDL

**Only run this step if you are installing xTrace for the first time on a host encyclopedia**

Submit JCL(HEDDL) to create the xTrace database and views to access the encyclopedia.

Notes:

- 1) The storage group must already exist.
- 2) If your installation does not use the default plan name for DSNTEP2, you need to change the plan name in HEDDL before submitting the job.
- 3) See the section below on ODBC access to the xTrace tables and views.

## Upgrade DDL

**Only run this step if you are upgrading xTrace from an earlier release on a host encyclopedia that requires a database upgrade**

Submit JCL(HEDDLUPG) to update the xTrace views.

Note:

If your installation does not use the default plan name for DSNTEP2, you need to change the plan name in HEDDLUPG before submitting the job.

## CICS definitions

If your site is not using CICS autoinstall, you need to run JCL(CICSCSD) which will define the CICS program definitions or define the xTrace programs to CICS using the CICS Resource Definition transactions.

If you have already installed xTrace, then this step is not required.

## Install modules

The xTrace runtime modules are available as both DLL and non-DLL versions. If you are using Gen 6.\*, then you should use the non-DLL version. If you are using Gen r7 or above, then you can either use the DLL or non-DLL versions. The DLL version is supplied to enable you to directly replace the TIRTRCE module in the Gen runtime DLL. See Appendix C for more details.

The DLL versions have the same dataset name as below except that there is a .DLL suffix.

Copy all members from <xtdshlq>.LOADC (.DLL) to a load



# Installation Guide

library that is available in the CICS region running your Gen applications. This must be a PDSE dataset.

Copy all members from <xtdshlq>.LOADI(.DLL) to a load library that is available in the IMS region running your Gen applications. This must be a PDSE dataset.

Copy all members from <xtdshlq>.LOADB(.DLL) to a load library that is available in your batch environment. This must be a PDSE dataset. The alternative is to include <xtdshlq>.LOADB in the STEPLIB for your batch applications and thus retain a single location for the software to make maintenance easier.

## Dataset Access

Verify that you have read-only access to 'TCPIP.ETC.SERVICES' and /etc/services

## Linking TIRTRCE

You must ensure that the xTrace version of TIRTRCE is linked into your applications instead of the default Gen version. See Appendix C for details.

## z/OS XTsvc

If you are using the z/OS xTservice, then the JCL to start the started task is in JCL(XTsvc). If you want to run this as a batch job, add a jobcard to the member.

If you did not enter your licence and keyfile when installing the software, enter these in JCL(XTsvcINI).

To stop the started task, use the MVS STOP command, for example, STOP XTsvc

## Client/Server Encyclopedia

xTrace requires the definition of several tables and views in the encyclopaedia database.

If you invoked the database install during the setup, review the setup.out file to verify correct installation of the database tables and views.

If you did not perform the database install during the initial setup, this needs to be done now.

- 1) Edit xtOraInst.bat (for Oracle) or xtSQLInst.bat (for SQL Server) and check that the variables entered are correct for this machine, especially the Oracle path for the data files.
- 2) If you will be running the database install from the xTservice machine (either to a local or remote database), then execute the xt<dbms>Inst.bat file.
- 3) If you cannot access the database from the xTservice machine, then transfer the xTraceOra.sql or xTraceSQL.sql file to the CSE machine and execute it using a database tool like sql\*plus or isql. If you are installing the Oracle database on UNIX, then first edit the XtraceORA.sql file and change the '\ ' to a '/' in the tablespace path name.

Note that for a database upgrade, use the xt<DDD>Upg.bat file and xTrace<DDD>-upg.sql files where <DDD> is Ora or SQL.



# Installation Guide

---

## ***ODBC Access to the xTrace Tables and Views***

It is recommended that each user logs onto ODBC using their own userid and then accesses the xTrace tables using the xTrace schema value defined in the Monitor setup. This requires that each user is granted access to the xTrace tables and views. Alternatively each user can logon to ODBC using an 'xTrace' userid.

If the users are not logging onto ODBC using a common xTrace userid, you will need to GRANT access to the xTrace tables and views, either to PUBLIC, to each user or to a group id. The DDL to accomplish this is DDL(GRANT) for the HE. By default these GRANTS are issued for Oracle when creating the xTrace database.

If you wish to create the XTRACE tables using one userid and access them via a secondary userid, then you can create synonyms for the tables using the XTSYN member in the DDL dataset. You will also need to execute the XTVIEWS DDL for the secondary userid to create the views and ensure that the user has the authority to execute SET CURRENT SQLID for the secondary userid.

## ***xTrace administration client***

The xTservice installation installs an administration client (xtadm.dll/xtadm.exe). Use the xtadmin.bat file to start the admin client.



# Installation Guide

---

## ***Multi-Home and NAT Support***

The xtservice.ini file can be customised to support multi-homed IP configurations and NAT (Network Address Translation) for both the application (z/OS) and Monitor.

To enable multi-home support, set the MULTIHOMED parameter to YES.

To enable NAT for the application, specify a table in the format below, with APP\_IPn and APP\_IP\_NATn taking values of n from 0 to 9.

```
APP_IP0 = 111.111.111.112
APP_IP_NAT0 = 555.555.555.552
APP_IP1 = 111.111.111.113
APP_IP_NAT1 = 555.555.555.553
APP_IP2 = 111.111.111.114
APP_IP_NAT2 = 555.555.555.554
```

To enable NAT for the Monitor, specify a table in the format below, with MON\_IPn and MON\_IP\_NATn taking values of n from 0 to 9.

```
MON_IP0 = 111.111.111.112
MON_IP_NAT0 = 555.555.555.552
MON_IP1 = 111.111.111.113
MON_IP_NAT1 = 555.555.555.553
MON_IP2 = 111.111.111.114
MON_IP_NAT2 = 555.555.555.554
```

You can also specify a \* wild card for the last octet, for example:

```
MON_IP0 = 111.111.111.*
MON_IP_NAT0 = 555.555.555.*
```



# Installation Guide

---

## Building and Running Applications Using xTrace

### **Gen 6.5**

xTrace provides a direct replacement for the TIRTRCE module. Therefore as long as you ensure that the xTrace version of TIRTRCE is linked into your application, you will be able to use xTrace.

### **Gen r7 and above**

With Gen r7 and above, the TIRTRCE module is included in a Gen supplied runtime DLL (TIRCRUNC for CICS and TIRTRCTZ for batch).

To ensure that the xTrace version of TIRTRCE is used instead of the Gen runtime module, you will need to amend the Gen supplied link-edit skeletons as detailed in Appendix C. You then build your applications as usual.

For IMS, the standard Gen runtime does not have its own version of TIRTRCE so you only need to ensure that the xTrace version of this module is available in the SYSLIB concatenation for the link-edit of the load modules.

### **Linking TIRTRCE**

If you are linking the xTrace version of TIRTRCE into your load module as opposed to the Gen runtime, ensure that the xTrace version of TIRTRCE is available in the SYSLIB concatenation ahead of the standard Gen version.

### **Verification**

To verify that you have correctly linked the xTrace version of TIRTRCE into your application, browse your load module and find "Meisner IT" and you should find "(C) Copyright 2006 Meisner IT".



# Installation Guide

## Appendix A – Installation variables

Name	Description	Site value
<b>General</b>		
XTUNIT	Disk unit for xTrace installation datasets E.g. SYSDA. This value is SET in JOB01. Amend the value in this job if you want to change the default UNIT value.	
XTDSDLQ	Dataset qualifier used for installation datasets. E.g. XTRACE.V200	
XTINUSR	Userid that runs the installation jobs	
XTSADDR	IP address or DNS name of xTservice E.g. "123.123.123.123" or "xtrace"	
XTSPORT	Port number that xTrace Listener is listening on. E.g. "8081"	
CEELKED	LE/370 link edit library E.g. SYS1.SCEELKED	
COBCMP	Cobol compiler library E.g. SYS1.IGYSCOMP	
CICSLOAD	CICS load library This variable is only used if your installation does not use CICS auto install	
CICSCSD	CICS system definition dataset This variable is only used if your installation does not use CICS auto install	
CICSGRP	The CICS group into which the xTrace modules should be defined This variable is only used if your installation does not use CICS auto install	
<b>Below only for CE</b>		
CESQLID	SQLid qualifier for the host ency tables	
XTSQLID	SQLid used to create the xTrace tables	
XTSTGRP	Storage group used to create the xTrace database	
XTRACDB	The name of the xTrace database.	
XTDB2SS	DB2 subsystem for the xTrace database / CE	



# Installation Guide

## Appendix B – User exits

### XTRACEUE

User exit for specifying installation specific values.

The user exit is found in the SOURCE dataset in a separate member per environment. EXITCICS for CICS, EXITBATC for batch and EXITIMS for IMS.

Name	Description
XT-SERVICE-ADDRESS	IP address or DNS name of the server running the xTrace listener. Check that you can 'ping' the value entered here, for example, 'ping xtrace'.
XT-SERVICE-PORT	Port number on which the xTrace listener is listening for trace requests. IP address and port number must match the definitions in the client side exit.
XT-USER-ID	Specify the userid value when XT-USER is set to 0
XT-USER	<p>0 – If you specify your own userid (XT-USER-ID)            1 – Userid is obtained from the task authorisation id for batch and IMS and from <b>EXEC CICS ASSIGN USERID</b> in CICS.            2 – Use the Globdata userid            3 – Use the Globdata Client userid            4 – Use the values provided in the import views of the module. Specify the view name, entity type name and attribute name using:</p> <p>XT-IMPORT-VIEW            XT-IMPORT-ENTITY            XT-IMPORT-ATTRIBUTE</p> <p>If this view is not found in the import view, the value of the userid in GLOBDATA is used instead.</p>
XT-LOG-TYPE	<p>If you want xTrace to log information, the logtype specifies where log information is written to.</p> <p>0 – No logging is performed            1 – Log information is written to xTservice. This does have a performance overhead.            2 – Logging is done via print to the console (printf())            3 – Log information is written to a file</p>
XT-LOG-LEVEL	<p>0 – Error            1 – Warning            2 – Info            3 - Debug</p>
XT-IMPORT-VIEW	See XT-USER above
XT-IMPORT-ENTITY	See XT-USER above
XT-IMPORT-ATTRIBUTE	See XT-USER above
XT-LOCAL-ADDRESS	Specifies the local IP address to bind the socket to. Normally this should be left blank to pick the default local IP address. If you have multiple local IP addresses and want to use a specific one, then provide the IP address using this variable.



# Installation Guide

---

## ***Fatal Error Exit***

CA Gen applications trap a number of errors automatically and passes error information to a fatal error exit. This error exit allows your installation to direct error messages to your standard location for program errors. When this type of error occurs, control is not returned to the application which includes xTrace.

In the event of an error the xTrace monitor is not notified and will assume that the application is still active.

xTrace has its own fatal error exit that may be activated from the CA Gen fatal error exit. The xTrace error exit must be called statically.

The fatal error exits are:

TIRTERMA for online load modules

TIRTERMB for batch load modules

TIRELOG for server load modules

Edit the fatal error exit and add the following in the program logic after MAINLINE:

TIRTERMA/B:

```
CALL 'XTRACEFE' USING TERM-EXIT-PARM-LIST.
```

TIRELOG:

```
CALL 'XTRACEFE' USING ELOG-EXIT-PARM-LIST.
```

For Gen r7, you will need to re-build the runtime DLL exits. To ensure that XTRACEFE is included, add <xtrace>.NCAL to the SYSLIB library list or copy it to the Gen LOAD library.



# Installation Guide

---

## Appendix C – Linking TIRTRCE

### Gen 6.5

With Gen 6.5 the TIRTRCE module is included within the TIRCALCC Gen supplied runtime module which is included into online and server load modules.

To ensure that the xTrace version of TIRTRCE is used instead of the Gen runtime module, you will need to follow one of the following approaches:

- 1) Use a customised version of TIRCALCC that contains the xTrace version of the TIRTRCE module. If you wish to adopt this approach, link the xTrace version of TIRTRCE into TIRCALCC.
- 2) Amend the Gen supplied link-edit skeletons (located in the Gen SLIB dataset) to ensure that the xTrace version of TIRTRCE is used instead of the standard Gen runtime version. This will affect all applications that use this SLIB dataset.

The modifications are as follows:

#### TICLCICC

Insert the line INCLUDE SYSLIB(TIRTRCE) after line 1. The first three lines should therefore read:

```
INCLUDE SYSLIB(TIRMSGC)  
INCLUDE SYSLIB(TIRTRCE)  
INCLUDE SYSLIB(TIRCALCC)
```

#### TICLCICD

Insert the line INCLUDE SYSLIB(TIRTRCE) after line 1. The first three lines should therefore read:

```
INCLUDE SYSLIB(TIRDAT2C)  
INCLUDE SYSLIB(TIRTRCE)  
INCLUDE SYSLIB(TIRCALCC)
```



# Installation Guide

---

## Gen r7 and above

With Gen r7 and above, the TIRTRCE module is included in a Gen supplied runtime DLL (TIRCRUNC for CICS and TIRTRCTZ for batch). For IMS, no TIRTRCE is provided in the Gen runtime DLLs since Gen does not support tracing for the IMS environment.

To ensure that the xTrace version of TIRTRCE is used instead of the Gen runtime module, you will need to either:

1. Replace the standard Gen TIRTRCE module in the Gen runtime DLL with the xTrace supplied TIRTRCE (CICS or batch) only. For IMS because the Gen runtime does not contain TIRTRCE, you can simply place the xTrace version of TIRTRCE in a SYSLIB dataset and it will be picked up at link time using AUTOCALL.
2. Amend the Gen supplied link-edit skeletons (located in the Gen SLIB dataset) to ensure that the xTrace version of TIRTRCE is used instead of the standard Gen runtime version

Method (1) is recommended since some strange runtime issues have been experienced when using method (2). For IMS you will need to use method (2).

## Re-link Gen Runtime

An example job is show below to re-link the runtime modules to replace the standard Gen TIRTRCE with the xTrace version. You will also need to re-link TIRCRUNC.

When using this method, you should reference the DLL versions of the xTrace runtime for the link-edit and also for the runtime modules.

```
//LINKB EXEC PGM=IEWL, PARM='DYNAM(DLL) '
//SYSLIB DD DISP=SHR, DSN=CEE.SCEELKED
// DD DISP=SHR, DSN=XTRACE.V250.NCALB.DLL <- XTRACE NCAL DLL DATASET
// DD DISP=SHR, DSN=CA.GEN.CEHBPLD1 <- GEN RUNTIME LOAD
//SYSLMOD DD DISP=SHR, DSN=CA.GEN.CUSTOM.LOAD <- DATASET FOR CUSTOM RUNTIME
//SYSUT1 DD UNIT=SYSDA, SPACE=(TRK, (5,1)), DCB=BLKSIZE=4096
//SYSPRINT DD SYSOUT=*
//SYSDEFSD DD SYSOUT=*
//SYSTEM DD SYSOUT=*
//SYSLIN DD *
REPLACE TIRTRCE
INCLUDE SYSLIB(TIRTRCTZ)
INCLUDE SYSLIB(TIRTRCE)
NAME TIRTRCTZ(R)
/*
//LINKC EXEC PGM=IEWL, PARM='DYNAM(DLL) '
//SYSLIB DD DISP=SHR, DSN=CEE.SCEELKED
// DD DISP=SHR, DSN=XTRACE.V250.NCALC.DLL <- XTRACE NCAL DLL DATASET
// DD DISP=SHR, DSN=CA.GEN.CEHBPLD1 <- GEN RUNTIME LOAD
//SYSLMOD DD DISP=SHR, DSN=CA.GEN.CUSTOM.LOAD <- DATASET FOR CUSTOM RUNTIME
//SYSUT1 DD UNIT=SYSDA, SPACE=(TRK, (5,1)), DCB=BLKSIZE=4096
//SYSPRINT DD SYSOUT=*
//SYSDEFSD DD SYSOUT=*
//SYSTEM DD SYSOUT=*
//SYSLIN DD *
<see below for link-edit statements>
/*
```



# Installation Guide

---

## Link-edit statements for Gen r7\*

```
REPLACE TIRTRCE
INCLUDE SYSLIB(TIRCRUNC)
INCLUDE SYSLIB(TIRTRCE)
NAME TIRCRUNC(R)
```

## Link-edit statements for Gen r8\*

```
REPLACE TIRTRCE
INCLUDE SYSLIB(TIRCRUNC)
INCLUDE SYSLIB(TIRTRCE)
IMPORT CODE, 'TIRXINFZ', 'TIRXINFO'
IMPORT CODE, 'TIRMDLL', 'TIRMMFN'
ENTRY CEESTART
MODE AMODE(31), RMODE(ANY)
NAME TIRCRUNC(R)
```

## Skeleton Modifications

To modify the Gen skeletons to ensure that the xTrace runtime module is used, edit each of the skeletons referenced below and comment the line containing the `IMPORT CODE, 'Tlxxxxx', 'TIRTRCE'` then insert the line `INCLUDE SYSLIB(TIRTRCE)` immediately afterwards:

### TICLBTC, TICLDLI, TICLDSN, TICLAED:

```
)CM IMPORT CODE, 'TIRTRCTZ', 'TIRTRCE'
INCLUDE SYSLIB(TIRTRCE)
```

### TI\$BTC, TI\$LDLI, TI\$LDSN, TI\$LAED

Insert the following line: `INCLUDE SYSLIB(TIRTRCE)`  
Before: `INCLUDE SYSLIB(TI$TRCTZ)`

### TICLCICD, TICLCICM

```
)CM IMPORT CODE, 'TIRCRUNC', 'TIRTRCE'
INCLUDE SYSLIB(TIRTRCE)
```

### TICLIMSD, TI\$LIMSD

Insert the following line: `INCLUDE SYSLIB(TIRTRCE)`  
Before: `IMPORT CODE, 'TIRCRUNI', 'TIRCTRC'`

### TI\$LCICD, TI\$LCICM if Gen r7.6 Pty Enhancement installed

```
INCLUDE SYSLIB(TI$CRUNC)
IMPORT CODE, 'TIRCRUNC', 'RTATRCE'
```

must be changed to

```
INCLUDE SYSLIB(TIRTRCE)
INCLUDE SYSLIB(TI$CRUNC)
IMPORT CODE, 'TIRCRUNC', 'RTATRCE'
```



# Installation Guide

## Appendix D – xTrace Source Code Post-Processor

The xTrace Source Code Post-Processor is a utility which enhances generated code. It enhances the generated code in the following areas:

- Current Subscript values for Group Views are passed to xTrace and displayed in the View windows
- Performance enhancements when executing code generated with trace, but without an active trace monitor
- Extra information about generation options passed to xTrace
- More reliable CPU figures in Call Stack

To determine if an Action Block has been post-processed, open the Action Block Detail window in the xTrace Monitor.

Sample JCL for executing the postprocessor is available in the installation JCL dataset.

### ***COBOL Examples and Details of Post-Processed Source Code***

#### **Current subscript**

The generated COBOL source keeps the current subscript for group views in a local structure:

```
*****  
* REPEATING GROUP VIEW STATUS FIELDS  
*****  
  
01 FILLER.  
  03 EXPORT-GROUP-0002FL PIC X(1).  
  03 EXPORT-GROUP-0002PS PIC S9(4) COMP.  
  03 EXPORT-GROUP-0002RF PIC X(1).  
  03 EXPORT-GROUP-0002MM PIC S9(9) VALUE 100 COMP.  
  03 EXPORT-GROUP-0001FL PIC X(1).  
  03 EXPORT-GROUP-0001PS PIC S9(4) COMP.  
  03 EXPORT-GROUP-0001RF PIC X(1).
```

The xTrace source code post-processor renames this structure:

```
*****  
* REPEATING GROUP VIEW STATUS FIELDS  
*****  
  
01 XTRACE-RGV.  
  03 EXPORT-GROUP-0002FL PIC X(1).  
  03 EXPORT-GROUP-0002PS PIC S9(4) COMP.  
  03 EXPORT-GROUP-0002RF PIC X(1).  
  03 EXPORT-GROUP-0002MM PIC S9(9) VALUE 100 COMP.  
  03 EXPORT-GROUP-0001FL PIC X(1).  
  03 EXPORT-GROUP-0001PS PIC S9(4) COMP.  
  03 EXPORT-GROUP-0001RF PIC X(1).
```

And passes it to the xTrace runtime:



# Installation Guide

---

```
CALL 'TIRTRCE' USING GLOBDATA
  PADTEXT
  LAST-STATEMENT-NUM
  W-IA
  W-OA
  W-LA
  GLOBDATA
  PADVIEW
  TRACE-RET-CD
  XTRACE-INFO
  XTRACE-RGV
```

## Performance enhancements

The xTrace runtime module is designed for high performance, but it is dependent upon debug information in the generated COBOL code. A member that is generated with debug support calls the trace runtime (xTrace or the standard Gen trace module) once for each statement.

Before each call to the trace runtime, all Import and Export View data is moved to a temporary storage area, and on return from trace, this View data is moved back. If the Action Block has large Import / Export Views there is a considerable overhead and is totally unnecessary unless someone is actually tracing that statement.

The xTrace source code post-processor adds a condition to the trace call and if the xTrace runtime determines that it is not necessary to inspect import or export views for the current statement, the PASS-DATA-TO-XTRACE flag is set to 'N'.

```
PARA-0000000000-TRACE.
  MOVE 'Y' TO SOURCE-CODE-ENHANCED
  IF PASS-DATA-TO-XTRACE NOT = 'N'
    MOVE IMPORT-0027EV TO IA-V-000
    MOVE IMPORT-0028EV TO IA-V-001
    MOVE EXPORT-GROUP-0003RG TO OA-V-000
    MOVE EXPORT-0030EV TO OA-V-001
    MOVE EXPORT-0031EV TO OA-V-002
    MOVE SPACES TO PASS-DATA-TO-XTRACE
  END-IF
  CALL 'TIRTRCE' USING GLOBDATA
    PADTEXT
    LAST-STATEMENT-NUM
    W-IA
    W-OA
    W-LA
    GLOBDATA
    PADVIEW
    TRACE-RET-CD
    XTRACE-INFO
    XTRACE-RGV
  MOVE 'N' TO SOURCE-CODE-ENHANCED
  IF PASS-DATA-TO-XTRACE = 'Y'
    GO TO PARA-0000000000-TRACE
  END-IF
  IF PASS-DATA-TO-XTRACE NOT = 'N'
    MOVE 'N' TO SOURCE-CODE-ENHANCED
```



# Installation Guide

---

```
MOVE IA-V-000 TO IMPORT-0027EV
MOVE IA-V-001 TO IMPORT-0028EV
MOVE OA-V-000 TO EXPORT-GROUP-0003RG
MOVE OA-V-001 TO EXPORT-0030EV
MOVE OA-V-002 TO EXPORT-0031EV
MOVE 'N' TO PASS-DATA-TO-XTRACE
END-IF
```

## Additional information

The generated source code holds a number of items of information about the generated Action Block. The xTrace source code post-processor passes this information to xTrace runtime for use by the xTrace Monitor:

```
01 XTRACE-INFO.
03 XT-POSTPROCESS-VERSION PIC X(5) VALUE '2.0'.
03 XT-DEBUG-TRACE-OPTION  PIC X(1).
03 XT-DATA-MODELING-CONSTRAINT PIC X(1).
03 XT-OPTIMIZED-IMPORT-VIEW  PIC X(1).
03 XT-HIGH-PERFORMANCE-VIEW  PIC X(1).
03 XT-LAST-STATEMENT-NUM    PIC X(1).
03 XT-ENFORCE-DEFAULT-VALUES PIC X(1).
03 XT-INIT-UNSPECIFIED-OPTIONAL PIC X(1).
03 XT-INIT-DYNAMIC-LINK-PROPERTY PIC X(1).
03 XT-GEN-VERSION          PIC X(32) VALUE
  'AllFusion(R) Gen r7.6'.
03 FILLER                  PIC X(58) VALUE
  'COPYRIGHT (C) 2008 CA. ALL RIGHTS RESERVED.'.
03 IEF-CGEN-DATE          PIC X(8) VALUE '20080515'.
03 IEF-CGEN-TIME         PIC X(8) VALUE '17:41:27'.
03 FILLER                 PIC X(8) VALUE '9.1.A5 '.
03 XT-GEN-USER           PIC X(8) VALUE 'HXM  '.
03 XT-GEN-MODEL         PIC X(32)
  VALUE 'HXM VIDEO STORE DEVELOPMENT  '.
03 FILLER                PIC X(32)
  VALUE 'ALL  '.
03 FILLER                PIC X(32)
  VALUE 'MXTR1011_CUSTOMER_MAINTAIN  '.
03 TODAYS-DATE          PIC 9(8) VALUE 20000000.
```

## CPU figures in call stack

The CPU figures in the Call Stack are captured by the xTrace runtime and stored in the Call Stack. In some situations, especially when calling between non-DLL and DLL implementations, it is not possible to derive correct CPU figures.

The xTrace source code post-processor adds calls to xTrace to get more reliable CPU figures.



# Installation Guide

---

## Appendix E – Pre-requisites

xTrace has been tested with CA Gen 6.5 and above.

xTrace is a C application which uses tcp/ip sockets for communication. This adds the following requirements.

Prerequisite	Description	Reference
CICS	xTrace uses TCP/IP sockets in CICS. The CICS region must support this.	Following IBM books: IP CICS Sockets Guide SC31-8807-03
IMS	xTrace uses TCP/IP sockets in IMS. The IMS region must be configured to allow this.	
DB2	Sufficient authority to create DB2 tablespaces, tables and views.	



## Appendix F – Troubleshooting

Question	Solution
Is the correct version of TIRTRCE being linked into my application	Browse your load module and do a find “Meisner IT” and you should find “(C) Copyright 2006 Meisner IT”.
I execute my transaction/batch job, but nothing shows in the monitor.	<p>The monitor will only display action diagrams when a breakpoint is reached if you have ‘Initial Breakpoint’ disabled. In the Tools/Options menu, check “Initial breakpoint”.</p> <p>The next thing to check is the userid. xTrace and the monitor must use the same userid. Set the xTrace log level to INFO in EXITCICS/EXITBATC and install the exits.</p> <p>In xTservice.log on the xTservice server look for:            QUERY(xxxxxxx) – the user that the monitor is using            TQUERY(xxxxxxx) – the user that the application is using.            These two userids must match to create a trace session.</p> <p>The next thing to check is that the IP port used for communication between the host and the monitor is not being blocked. The range of the trace session ports is sequential starting with the value defined in the XTservice.ini file:</p> <pre>mon_port = 9020</pre> <p>You therefore need to open 9020 (or whatever you have used for this variable) plus the next n sequential ports where n is the number of concurrent xtrace sessions that you might expect.</p>
I get “EDC8128I Connection refused. RC01128, Failed to connect() to <ip-address>:<port>”	<p>Connection refused can happen for 2 reasons:</p> <p>The port number is being used by another application on the workstation running the xTrace monitor. Close the monitor, open a DOS prompt and type “netstat -a” and make sure that the port does not appear in the list. If it does you can change mon_port in xTservice.ini</p> <p>The second reason is a firewall issue. If your windows firewall is set up to not accept connections on this port, you will receive this error.</p>
When I start the Monitor, I get an error message “Cannot access your xTrace licence”	<ol style="list-style-type: none"> <li>1) Check that the xTservice service is started</li> <li>2) Check that the value entered for the XT_SERVICE_IP variable in xTrace.bat is</li> </ol>



# Installation Guide

---

the correct IP address or DNS name for the server running xTservice.

- 3) Check that the port defined in `XT_SERVICE_PORT` is not being blocked by a firewall.