OpenText™

OpenText™ Exceed™ TurboX

An Architecture Overview

Remote application access is increasingly important in today"s business environment. Solutions that provide remote access to applications and data can eliminate geographical and network boundaries, improve user productivity, enhance protection of intellectual properties, and help streamline IT operations while reducing costs.

OpenText Exceed TurboX is an advanced solution for desktop virtualization and remote access to enterprise applications and data, addressing the needs of modern enterprises, especially those with mixed UNIX®/Microsoft® Windows® hosting environments. This white paper describes the principles of its design, the architecture and key components of the solution, and their functions and interactions.

OPENTEXT™

Table of Contents

1.0 Introduction....3

2.0 The Architecture Diagram....4

3.0 The Building Blocks....4

4.0 Connectivity....7

5.0 ThinX Protocol....7

6.0 Client and Proxy Runtime....8

7.0 Authentication....9

8.0 Clustering....10

9.0 Licensing....11

10.0 Fail-Over Scenarios....12

11.0 Platforms....13

Closing....13

1.0 Introduction

Remote application access has been a solution category since computers were first linked into a network. Businesses have always sought a solution that would allow users to access applications running on a remote host. Along came globalization, followed by all forms of virtualization, which pushed applications further away from users, while increasing the need for users to be productive while accessing applications and data from a great distance.

OpenText has been in the business of providing remote application access solutions for more than 20 years. Beginning with OpenText™ Exceed™ in the 1990s and including OpenText™ Exceed onDemand™ in the 2000s, OpenText has pushed the performance envelope, bringing complex 2D and 3D applications to enterprise users all over the world. Exceed TurboX is the future of remote application access solutions, made available today. It is a culmination of years of experience with X Window and network programming, and the understanding of hundreds of real world cases using business-critical UNIX and Windows applications.

Six primary goals guided the conception and development of Exceed TurboX:

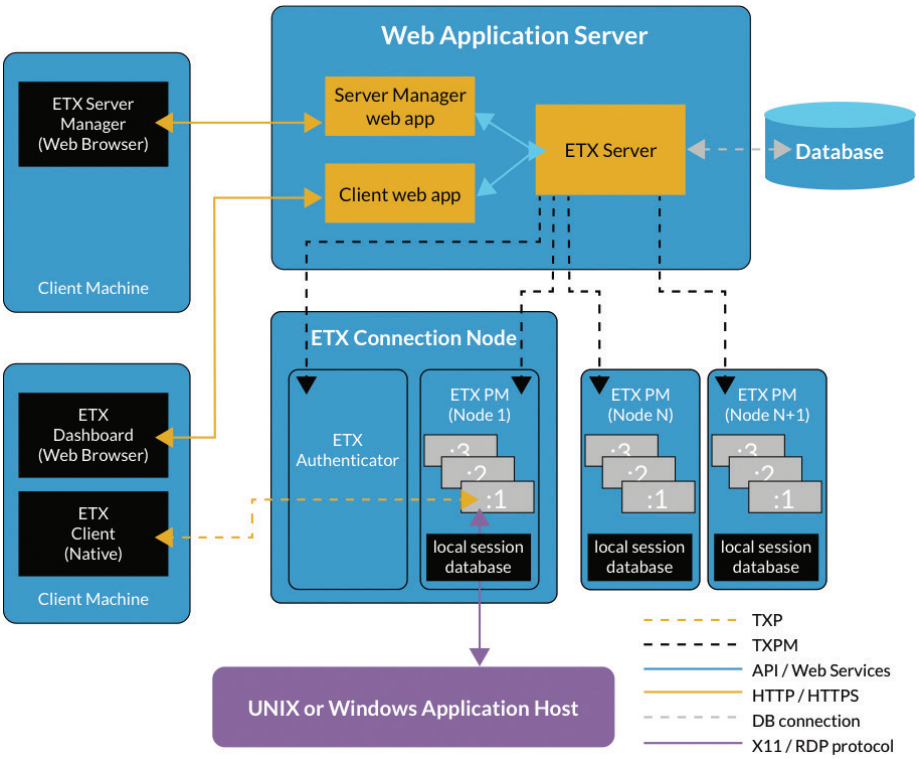
- Provide a single platform for IT and users to manage and access all of their remote infrastructure, regardless of platform
- Provide the fastest remote access solution on the market for accessing UNIX and Windows hosts
- Provide a consistent user experience across all supported client platforms
- Simplify the distribution and management of client software, profile, and settings to support large deployment scenarios
- Offer fully auditable events for better visibility, accountability, and problem resolution
- Work in tandem with any existing infrastructure and vendor, including mixed physical/virtual desktop environments

Exceed TurboX delivers results on all fronts. Its architecture is a departure from many traditional remote application-access solutions on the market, including our own Exceed onDemand. It relies on a hybrid of technologies to deliver high performance access with a great user experience and simplified administration.

2.0 The Architecture Diagram

The following diagram depicts the overall layout of all major Exceed TurboX components and their relationships. Firstly, you will notice that ETX relies on a Web Application Server to provide a web-based graphical front end. Secondly, ETX uses a database to store application data.

In addition to the ETX server, which provides the user interface, configuration and load balancing, ETX Connection Nodes are responsible for managing client sessions and compressing the remote display for transmission to users. A group of connection nodes along with the ETX server is called the ETX Site.



3.0 The Building Blocks

The following components form the foundation of Exceed TurboX.

3.1 ETX Server

This is the brain of the solution, functioning as concierge, gatekeeper, and traffic controller. It is a web application that runs on an Apache® Tomcat® web application server and is presented to users when they first connect to the ETX web portal. ETX Server displays either a Dashboard to users or a Server Manager to administrators. Users can use the Web Dashboard to launch new sessions, create profiles, or manage their existing sessions. Administrators use the Server Manager to monitor and manage every aspect of the solution, from user enrollment to generating license and resource usage reports.

3.2 ETX Connection Nodes

The ETX Connection Node is responsible for starting session proxies (an application which manages and compresses remote sessions), as well as interacting with the ETX server. With multiple nodes present in the infrastructure, ETX server can select from multiple connection nodes for load balancing and node failover.

ETX Connection Nodes include several sub-components, which each handle a different task. In addition to the Proxy Manager (which starts the session proxies), other important components include the ETX Authenticator, Xstart Launcher, and Application Scanner, which are explained in more detail below.

3.2.1 ETX Proxy Manager

The main function of the ETX Proxy Manager is to create remote session proxies. Proxies act as an intermediary between the client and application or desktop host, providing encryption and compression of the session display over the WAN. The proxy also handles passing user input to the host, as well as other data such as files and devices.

The Proxy Manager monitors the well-being of these session proxies and communicates the status of each session proxy to the ETX Server. The Proxy Manager also handles user

commands, such as session suspend and resume, session sharing (multicasting), file transfer, and other session-related features.

A Proxy is a fully functional X server/RDP client, and can therefore operate as a stand-alone server, handling display protocol from default installations of Windows or UNIX hosts without any local software being installed. The proxy includes an X server as well as an RDP client for rendering and compressing the display over the WAN. In the case of Windows Direct desktop connections, the proxy must be installed directly on the Windows desktop host for local rendering and compression. A compressed data stream is sent from the Proxy to the Client, which is installed on the user"s device. The client handles screen drawing and keyboard and mouse inputs. More information on the functions of Proxies and Clients is available in Section 6.0.

3.2.2 ETX Authenticator

As the name implies, this component handles authentication of users and administrators when they log in to the ETX web portal. The authenticator is required to support PAM or Native authentication in UNIX environments. LDAP-based environments (e.g. Microsoft Windows) do not require an authenticator because LDAP authentication is handled by ETX Server.

The ETX Authenticator provides seamless integration of ETX into UNIX environments that use a dedicated PAM or Native authentication server. This requires you to install an ETX Connection Node on the PAM/Native authentication server and register the Node as an Authenticator during installation. ETX supports multiple authentication nodes for load balancing, but each node must be configured with the same PAM or Native authentication settings.

3.2.3 Xstart Launcher

Xstart Launcher is the module responsible for launching individual X Window or Windows applications when a user launches an ETX profile. ETX profiles include all configuration settings for the remote session, including which host to select and which applications to run.

3.2.4 Application Scanner

The application scanner is an optional utility that scans the Connection Node for installed applications, making them available for publishing via the ETX Server Manager. Once an application is published to users, users can launch the application by creating a profile and selecting the application to launch.

The application scanner works on all ETX platforms, so you can publish applications running on both UNIX and Windows. If you have multiple Connection Nodes with the same application installed, ETX Server detects the overlap and groups applications into a single item. This provides load balancing of applications across multiple hosts.

3.2.4 Local Session Database

The local session database keeps track of session status and reports this back to ETX Server on a regular basis. If the connection between ETX Server and the Proxy manager fails, the local session database enables the Connection Node to continue operating normally, including suspending or terminating sessions. When the connection to ETX Server is re-established, the local session database is synced with ETX Server. This makes each node fault-tolerant in case of a network failure.

3.3 Database

The RDBMS is the bookkeeper of the solution. It is used to store all non-transient information, including server settings, user settings, profiles, event logs, license usage statistics, etc. ETX supports IBM® DB2®, Microsoft SQL Server®, and Apache® Derby™, and comes bundled with Apache Derby for quick and easy installation. For load-balancing, fail-over, and high-availability capabilities, please refer to the ETX High Availability White Paper, as well as vendor documentation for setting up database replication.

3.4 ETX License Server

Multiple ETX sites can share a single pool of licenses and have licenses dynamically or statically distributed to each site based on rules defined by administrators. The ETX License Server functionality is embedded in every ETX Server and can be enabled during setup. Once enabled, other ETX sites can connect to the License Server and use licenses from the pool, rather than using locally installed license keys. Special care has been given to the design of this solution to ensure that if the License Server is out of service or out of reach, administrators will be notified via email, and sites will remain operational for 72 hours so that the problem can be resolved.

3.5 ETX Client Runtime

ETX relies on a native client Runtime to handle remote communications and interactions. It is the counterpart to the ETX Proxy and communicates with the proxy continuously whenever a session is established. Like a Proxy, the client Runtime is a native application, which takes advantage of the best possible performance, reliability, and compatibility. Deployment of the ETX client Runtime is automatic; the correct Runtime is downloaded silently in the background whenever a user launches an application. Client Runtimes are available for Windows, Linux® and Mac® users.

3.6 Java Runtime Environment (JRE)

While it is not necessarily a building block on its own, JRE is a system requirement for the ETX Server. Naturally, JRE is required on the machine running the ETX Server and Apache Tomcat. Java can also be used to launch ETX remote sessions from a browser, although this is not the default option. By default, launching ETX sessions requires the Native Client Launcher component to be installed on the user"s device (see section 3.7 below).

3.7 Native Client Launcher

The Native Client Launcher must be installed on the user"s machine before the user can launch a profile in Exceed TurboX (unless they have selected the Java client launcher; see section 3.6 above). The Native Client Launcher will download and execute the Native Client Runtime in the background when the user launches a profile. The Launcher does not require administrator privileges to install, and it is provided for Windows, Linux, and Mac clients.

4.0 Network Configuration

Communication between ETX components occurs on several different ports. It is important to ensure that your firewall is configured properly to ensure proper operation of the ETX Site. The table below summarizes the required network channels, including the purpose of the channel, the components involved, the protocols used, and whether the channel is secured.

DESCRIPTION	FIRST COMPONENT	SECOND COMPONENT	PROTOCOL(S)	DEFAULT PORT(S)	SECURITY
Accessing ETX Dashboard and Server Manager in a browser	Web browser	ETX Server (Tomcat)	HTTP/HTTPS	8080/8443	None/SSL
ETX Server contacts IBM DB2 database	ETX Server	IBM DB2 Database	JDBC	50000	None (SSL optional)
ETX Server contacts Microsoft SQL database	ETX Server	Microsoft SQL Database	JDBC	1433	None (SSL optional)
ETX Server contacts Apache DerbyDB database	ETX Server	Apache Derby Database	JDBC	1527	None (SSL optional)
ETX Server sends a message to the Connection Node	ETX Server	ETX Connection Node	TXPM	5510	SSL
ETX Client launches any profile	ETX Client	ETX Connection Node	ThinX	5510	None (SSL optional)
User launches an X application	ETX Connection Node	UNIX Host	X11	6000+	SSH
User launches a Windows application or RDP desktop	ETX Connection Node	Windows Host	RDP	3389	SSL
ETX Server contacts the license server	ETX Server	ETX License Server	HTTP/HTTPS	8080/8443	None/SSL
Accessing ETX Server REST APIs	Any application or script	ETX Server	HTTP/HTTPS	8080/8443	None/SSL

5.0 ThinX Protocol

ThinX Protocol is the digital blood coursing through the veins of ETX, and it is what makes ETX work so well, so fast. ThinX Protocol, or TXP, provides exceptional performance over a wide range of network bandwidths and latencies. Results of in-house analysis have shown that the amount of network traffic generated by TXP is between one and 10 percent of the traditional X protocol. It is also more efficient than competitive remote access solutions.

TXP is designed to accomplish the following:

1. Reduce bandwidth requirements for remote users
2. Adapt to changing network conditions
3. Reduce round-trip requests for X11 traffic
4. Strengthen security

Note: TXP is also used between the Client and Proxy for connections to Windows hosts. In the case of RDP connections, the ETX Proxy converts the RDP protocol to ThinX for fast transmission over the WAN.

5.1 Reduce Bandwidth Requirements

TXP has a superior ability to compress, cache, and optimize requests. The goal is to minimize the amount of data that has to travel across the network between Proxy and Client, which is characteristically high in network latency and low in network bandwidth. Naturally, the work required to optimize and reduce network traffic comes at the expense of more CPU cycles and computing resources on both end-points of the connection. However, the abundant computing power that PCs and servers now harness can easily handle the increased workload without slowing down the remote server, enabling Exceed TurboX to achieve bandwidth reduction without sacrificing overall performance.

5.2 Adapt to Changing Network Conditions

The level of compression and optimization on the TXP data stream can change automatically and dynamically depending on real-time changes to bandwidth availability, network latency, and even the contents of the remote display. Exceed TurboX will adjust and choose the right type of compression, data type, and update frequency to maintain maximum performance and usability. These adjustments happen on the fly without any need to customize protocol settings. ETX tends to consume available bandwidth in order to improve display quality and responsiveness, but users and administrators can also limit the bandwidth usage of a particular session by adjusting ETX profile settings.

5.3 Reduce Round-Trip Requests (X11)

The X Window protocol was designed for communication in a LAN environment where network bandwidth is abundant. Connecting to X applications over the WAN without a compression layer will result in very slow performance of the remote X applications or desktops. The ETX Proxy can be installed either on the X application host itself – thus eliminating X Window protocol from the LAN entirely – or on a separate machine on the same LAN as the X application host. The Proxy will either render on the server and send pictures over the WAN, or will compress, batch and send drawing instructions using TXP (ThinX Protocol) based on screen contents. With the intelligence built in to the Proxy, fewer round-trip requests will need to reach the Client, eliminating inefficiency and delays in the remote communication. The reduction in the number of round-trip requests over the WAN or internet connection is significant, giving Exceed TurboX an unparalleled performance advantage.

5.4 Strengthen Security

TXP is secure in design. It can be easily encrypted using the latest TLS 1.2 protocol for heightened security. In fact, TLS encryption is used by default between ETX server components to ensure business critical information is protected. Connections from users outside of the corporate LAN may be encrypted either by a VPN or by enabling TLS encryption in the ETX Server Manager.

6.0 Automatic Updates

ETX is designed to automatically apply updates to both the client and server components, eliminating the need to manually roll out updates to the system. ETX also supports the ability to deploy multiple versions of the client and server runtimes so that you can apply patches and different versions to different groups of users, all from a central management interface built into ETX Server.

6.1 No Client Installation

When users launch a profile for the first time, ETX will automatically detect the presence of the client-side launcher and prompt users to download and install the launcher if it is not found. This eliminates the need to manually install the software on client machines.

6.2 Automatic Client and Proxy Updates via Runtime Packages

The lightweight client and proxy runtimes are pushed to the user"s desktop, as well as the ETX proxy, when a user launches a session. The client and proxy runtime are developed in pairs, ensuring that every user receives a compatible version of the client and server components every time they launch a new session.

Once a runtime is downloaded, it will be cached on the machine and will not have to be downloaded again. This ensures that subsequent connections are launched as quickly as possible.

6.3 Automatic Connection Node updates

Starting in ETX 11.5, patching the ETX Server will automatically and silently push all necessary updates to the ETX Connection Nodes that are registered with the server. In VDI rollouts with thousands of virtual machines, this eliminates the need to manually update the software on thousands of nodes. Note: the Connection Node software is used to manage the session proxies on the node and is separate from the proxy runtime.

6.4 Manage multiple runtime versions concurrently

ETX Server allows you to install and use multiple runtime versions, which means that you can apply patches to specific users without affecting other users. This can be used to apply hot fixes for specific applications or perform staging experiments with ETX service packs and hot fixes. Each ETX profile has a customizable runtime version, which can be set by the ETX administrator or the ETX user (if they have permissions to edit profiles).

7.0 Authentication

ETX supports the following authentication methods:

- Lightweight Directory Access Protocol (LDAP)
- Microsoft® Active Directory® (AD)
- Kerberos-based Single Sign On (SSO)
- Pluggable Authentication Module (PAM)
- Native User Credentials (UNIX only)

ETX will use different components in the architecture to authenticate users, based on your authentication method.

AUTHENTICATION TYPE	COMPONENT THAT HANDLES AUTHENTICATION
LDAP, MICROSOFT ACTIVE DIRECTORY	ETX Server
PAM, NATIVE USER CREDENTIALS	ETX Connection Nodes

LDAP and Microsoft Active Directory are centralized identity management systems, which can be accessed from any computer. It is functionally simpler and more efficient to let ETX Server take care of those authentication types. To implement LDAP-based authentication, simply configure the appropriate LDAP settings during installation of ETX Server. For Kerberos-based, single sign-on environments, refer to the Exceed TurboX Installation and Administration Guide, which is part of the core product documentation.

Additionally, for UNIX environments, you may use PAM (Pluggable Authentication Module) or Native authentication methods. In these authentication modes, a dedicated PAM or Native authentication server is configured to handle ETX login requests. PAM or Native authentication servers must have the ETX Connection Node software installed, and be assigned the 'Authenticator" role during installation. Once the node is registered with ETX server, ETX server will direct all login requests to the Authenticator node. ETX Server allows more than one PAM/Native authenticator node to be registered, so that users won"t be locked out if the authentication server goes offline. However, you must make sure the PAM configuration or user database is identical across all Authenticators.

In addition to the 'Authenticator" role, nodes can be assigned to the 'Proxy Manager" role. The Proxy Manager role means that the node will accept remote desktop and applica-

tion sessions, and start a local session proxy to compress and manage the remote user session. If your authentication server(s) are shared by many applications, you may want to disable the proxy manager role to avoid taxing these systems with ETX sessions.

AUTHENTICATION TYPE	COMPONENT THAT HANDLES AUTHENTICATION
AUTHENTICATOR	Responsible for handling PAM and Native User Credentials authentication requests only
PROXY MANAGER	Responsible for creating and managing Proxies
AUTHENTICATOR AND PROXY MANAGER	Both of the above

7.1 Multiple Authenticators

For organizations that choose to use native authentication or PAM, it is always a good idea to have more than one Authenticator in the system in case one of the ETX Connection Nodes is disconnected or otherwise unavailable. ETX Server will choose one of the available ETX Authenticators at random. Therefore, it is of utmost importance that the same set of users and their credentials are available or accessible by all ETX Authenticators in the system; otherwise some users may not be able to log in to ETX at any given moment.

In most cases, having two Authenticators in the system provides the necessary load balancing and protection against a single point of failure. More than two authenticators may be redundant and create unnecessary complication and workload involved in synchronizing native user credentials across multiple machines.

8.0 Load Balancing

8.1 What is Load Balancing?

Load Balancing refers to the capability of ETX Server to distribute remote sessions to a “cluster” of Connection Nodes. Load balancing is a key element of providing a highly available environment for users to access their remote desktops and application servers.

8.2 How Does Load Balancing Work?

When a user launches a remote application or desktop, one of the ETX connection nodes creates a process (called a “proxy”) to act as intermediary between the user and the remote host. The proxy process is responsible for interacting with the host (as the session client) and compressing the remote display for transmission over the WAN to the end user.

If ETX were configured with only a single node to handle all user sessions, failure of that node (due to hardware, network, or other failure) would be crippling. ETX would not be able to handle any new sessions until the node is restored and reconnected.

With two or more nodes present, failure of a single node does not cripple the system. The other nodes can take over until the first node comes back online.

All ETX Connection Nodes are registered with ETX Server, and their presence is recorded and monitored so ETX Server knows the exact number of ETX Connection Nodes available, what each node is doing, and the resources available on each node. From the user”s perspective, there is no obvious difference between different Connection Nodes, and in most cases, users should not care which node handles the user”s session. Just like when you access Amazon.com, you don”t know, nor do you care, which one of a thousand different web servers is responding to your request.

8.3 Load Balancing

Because the ETX Server keeps track of all connection nodes and the available resources on each node, the server can assign sessions to connection nodes by selecting the node with the most available resources. This ensures that users achieve the fastest possible connections for the available server hardware.

The following load balancing criteria are supported by ETX Server:

LOAD BALANCING RULE	FUNCTION
SESSIONS	The node with the fewest active or suspended sessions is selected for the next session launch.
CPU	The node with the most CPU resources is selected. This does not account for CPU speed or architecture, only percentage of CPU consumption.
MEMORY	The node with the most available memory is selected. This type of load balancing can be used when running memory-intensive applications.

9.0 Licensing

9.1 Type of Licenses

ETX Licenses are concurrent in nature, meaning they allow a large number of users to share a smaller set of licenses. For example, if you purchase 50 licenses for an ETX site, 100 users might share those licenses, so long as no more than 50 users have active sessions at the same time.

9.2 Rules of License Usage

A single user can never consume more than one license. This means that a user can launch multiple sessions on multiple devices and will never consume more than a single concurrent user license. However, if a user shares their session with multiple users, each user joining the shared session will consume one license (provided the user isn"t already consuming a license).

9.3 Sharing Licenses Across Multiple ETX Sites

Multiple ETX sites can connect to an ETX License Server in order to share a single pool of licenses. The ETX License Server functionality is embedded in every ETX Server, and can be turned on during setup. The License Server should be configured first, so that other ETX Servers can specify the license server during setup. Special care has been given to the design of this solution to ensure that if the License Server is out of service or out of reach, all sites will remain operational for up to 72 hours, and administrators will be notified as long as the license server has been configured to send email notifications.

9.4 License Distribution Methods

ETX License Server supports two types of license distribution: Static and Dynamic.

9.4.1 Static License Distribution

As the name implies, licenses are statically distributed to each ETX site so you can tightly control license usage and costs on a per-site basis. Once licenses are statically distributed to an ETX Site, the site will manage the license usage internally as if those licenses were locally installed.

9.4.2 Dynamic License Distribution

Any licenses that have not been assigned statically to an ETX site are available for any connected site to consume on a first-come, first serve basis. Dynamic licenses can flow freely from one ETX site to another seamlessly.

Note: If a user launches sessions on two independent ETX sites that share a common license server, the user will consume one license from each ETX site. This is because each ETX site manages its licenses independently.

10.0 Failure Scenarios

The ETX solution is designed to be resilient to various types of failures and offer continuous services. Disasters or failures can happen to computers hosting different components, and there are various degrees of failure. The table below outlines how each component handles each type of failure.

COMPONENT	NETWORK DISCONNECTION	COMPONENT SHUTDOWN
CLIENT	If the machine where ETX Client is running loses its connection to the network, the user"s session will be automatically suspended.	If the machine where ETX Client is running crashes or otherwise is powered off, the user"s session will be automatically suspended.
CONNECTION NODE	If the machine hosting the Connection Node is disconnected from the network, X Window sessions will be permanently lost. Windows RDP sessions will become disconnected but may be resumed by launching a new session connecting to the same host on a different node. Other ETX Connection Nodes will continue to accept new connections.	If the machine hosting the Connection Node crashes or is powered down unexpectedly, X Window sessions will be permanently lost. Windows RDP sessions will become disconnected but may be resumed by launching a new session connecting to the same host on a different node. Other ETX Connection Nodes will continue to accept new connections.
AUTHENTICATOR	When using Native or PAM authentication for UNIX environments, if the Authenticator node is disconnected from the network, ETX Server will not be able to authenticate users. In this situation, no one will be allowed to log in to the ETX web site. However, if the administrator created a maintenance user (by default 'etxinstall"), they can log in with this maintenance user to fix configuration problems with the authentication node. If the authenticator fails, it will not affect existing sessions, unless those sessions are running on the authentication node (requires the authenticator node to be configured as a Proxy Manager).	
APACHE TOMCAT/ETX SERVER	If Apache Tomcat or the ETX Server web app is unavailable – whether due to a broken network connection or server crash, users will be unable to log into the ETX web interface or perform basic functions such as launching or resuming sessions. Administrators will be unable to log into the web interface to administer the ETX environment. However, all active sessions will continue to operate without any limitations. If a user suspends an active session while Apache Tomcat or the ETX Server is down, he or she will have to wait until those systems are revived and online before the session can be resumed. Note: ETX supports high availability for the web server. This means that you can run ETX servers in parallel. If the primary web server fails, the secondary will take over. For information on setup, see the Exceed TurboX High Availability Configuration Guide.	
DATABASE	If the database is unavailable because of a network failure, the ETX Web Dashboard and Web Server Manager will be inaccessible. However, all active sessions will continue to operate without limitations. When the database connection is restored, all functions will be resumed.	If the machine running the database crashes or is powered down unexpectedly, or the database is corrupted, ETX Web Dashboard and Web Server Manager will be inaccessible. However, all active sessions will continue to operate without limitations. When the database machine is restored all functions will be resumed.
LICENSE SERVER	**STATIC LICENSE DISTRIBUTION** Upon disconnection from a License Server, an ETX Server will continue to manage the allotted licenses as if they were locally installed, for up to 72 hours. This provides time for IT to find and fix any failures. Impact to the ETX Server and its users is minimal unless this condition is left untreated. **DYNAMIC LICENSE DISTRIBUTION** Upon disconnection from a License Server, each ETX Server will assume that any unallocated licenses are available for local sessions for up to 72 hours after the disconnection. When the connection to the License Server is restored, each ETX Server will report its current license usage to the License Server, and the License Server will recalculate the total number of licenses that are in use. Users who have active sessions will not be penalized if the number of licenses in use is greater than the number of licenses installed. In this case, this event will be logged and any user activities that require additional licenses will not be permitted until the total number of licenses in use is fewer than the number of licenses installed.	

11.0 Platforms

The supported platforms for each ETX component is provided in the table below.

LOAD BALANCING RULE	FUNCTION	
CLIENT	**WINDOWS PLATFORMS** Windows 10 Windows 8.1 Windows 7 SP1 or later Windows Server 2008 R2 SP1 or later Windows Server 2012 R2 or later **LINUX PLATFORMS** Red Hat Enterprise Linux 6.5 or later, 64-bit SuSE Linux Enterprise Linux 11 & 12, 64-bit	**MAC OS X PLATFORMS** MacOS 10.12 (Sierra) Mac OS X 10.11 (El Capitan) **IOS® PLATFORMS** iOS 9.0 or later, on iPad® 2 or later, iPad Air®, iPad Air 2, iPad mini™ 2 or later, iPad Pro® **JAVA™ CLIENT**
ETX SERVER	Red Hat® Enterprise Linux® 6.5 or later, 64-bit Oracle® Solaris® SPARC® 10 & 11, 64-bit Oracle Solaris x86-64 10 & 11, 64-bit	
ETX CONNECTION NODE	Red Hat Enterprise Linux 6.5 or later, 64-bit SuSE® Linux Enterprise 11 & 12, 64-bit Oracle Solaris SPARC 10 & 11, 64-bit Oracle Solaris x86-64 10 & 11, 64-bit IBM AIX® 6.1 or 7.1	Windows 10, 64-bit Windows 8.1, 64-bit Windows 7 SP1, 64-bit Windows Server 2008 R2 SP1, 64-bit Windows Server 2012 R2
DATABASE	IBM DB2 Express-C 10.5 with Red Hat Enterprise Linux 6 or later, 64-bit only IBM DB2 Enterprise Server 10.5 with Red Hat Enterprise Linux 6 or later, 64- bit only Microsoft SQL 2012 SP1 Standard Edition or later on Windows Server 2012 R2 or later Microsoft SQL 2008 R2 SP2 Standard Edition or later on Windows Server 2008 R2 SP1 or later Apache® Derby™ 10 or later, on all supported ETX Server platforms	
WEB BROWSERS	Google Chrome™ Mozilla® Firefox® Microsoft® Internet Explorer® 11 Microsoft Edge™ Apple® Safari® 8 or later (applicable to Mac OS X only)	

Closing

Thank you for your interest in OpenText Exceed TurboX. To learn more about Exceed TurboX, visit our website at: connectivity.opentext.com/etx.

About OpenText

OpenText enables the digital world, creating a better way for organizations to work with information, on premises or in the cloud. For more information about OpenText (NASDAQ: OTEX, TSX: OTC) visit opentext.com.

Connect with us:

- OpenText CEO Mark Barrenechea"s blog
- Twitter | LinkedIn | Facebook

www.opentext.com/contact