HOW IT WORKS

COMPONENTS

ROBLOX.EXE (BOOTSTRAPPER)

This is the initial starting point for installing Roblox. It checks with the install website to see if the local version of Roblox is up-to-date. If not, it downloads, deploys and configures everything needed to run Roblox for the current user.

Roblox.exe uses a mutex `www.roblox.com/bootstrapper` to ensure that only one instance of Roblox.exe is installing/uninstalling at any one time.

Roblox.exe uses a shared memory file to determine how to start RobloxApp.exe. Upon launch, each instance or Roblox.exe registers its PID as the most recent one. After installation, each instance checks to see if it is the most recent one. If so, then it executes RobloxApp.exe. This ensures that the most recent Roblox.exe instance is the only one to start RobloxApp.exe.

COMMAND LINE ARGUMENTS

[NO ARGUMENTS]

Equivalent to "-browser"

-UNINSTALL

Remove Roblox

-INSTALL

Install/Update Roblox

-BROWSER

Install/Update Roblox and then start RobloxApp as a web browser

-IDE

Install/Update Roblox and then start RobloxApp as a the IDE

-FORCE

Force Roblox.exe to download and install all components, even if cached or already present.

-PLAY [SCRIPT] [TICKET]

Internal use only. Used by RobloxProxy.dll to start a game.

ROBLOXPROXY.DLL (COM SERVER)

Contains a Launcher COM object that can be called from javascript. Its main job is to marshal cookies from IE's protected mode to RobloxApp.exe, and then to start the Roblox game.

RobloxProxy doesn't call RobloxApp directly. Instead, it simply creates an instance of Roblox.exe using the "-play" argument. This way, Roblox.exe can make sure that the code is up-to-date.

RobloxProxy looks in the registry to find out which Roblox.exe is the most recent.

ROBLOXAPP.EXE (APPLICATION)

This is the actual application. It is registered as a COM server so that RobloxProxy.dll can start it.

Just before displaying the main window this component sets IPC event `www.roblox.com/robloxStartedEvent`. This will be used by Roblox.exe (if running) to signal it to hide its progress dialog and exit.

INTERACTIONS WITH WEBSITE

Web pages start Roblox using the RobloxProxy.Launcher COM object. After an update, a page might point to an older version of the COM object, or at least a DLL that resides in an older version folder. If the website *must* work with a new version of the COM object, then the javascript needs to point to a different version of the COM object. FUTURE: For compatibility with old code, it should accept the old one as well for updating.

INSTALLATION

All components necessary to install or update Roblox are uploaded to a setup domain. The production domain is http://setup.roblox.com.

Roblox.exe queries several URLs in this domain to determine if an update is needed. For example, the path "version.txt" returns the current version string.

PER-USER, MOSTLY

Per-user installs work on WinXP Limted User Accounts and don't require elevation prompts under Vista.

Roblox installs per-user rather than for all users with one exception: Vista with UAC turned off installs all-user. This is done because of a bug with Vista's SP1 and OLE automation (TODO: cite blog post)

NO MSI

Roblox does not use any MSI technology for installs or updates. However, it does register itself for user uninstall from the Control Panel.

DEPLOYMENT

All installation components are uploaded to a website – currently a domain on S3.

The order of component deployment is carefully orchestrated to work concurrently with client-side access of the website.

1) Upload bootstrapper to the version-specific url
2) Upload version-specific installation components, like libraries, content folders, etc.
3) Upload "version.txt", which points to the items uploaded in the previous steps. This URL is used by existing users to detect the need to update.
4) Upload the "Roblox.exe" bootstrapper and other installation files to the version-independent urls. See Uploader.Program.UploadClient() in the Installer/Uploader project for details.

STATS AND ERRORS

The Roblox.exe bootstrapper posts HTTP requests to the main website to log installation start, stop, error, etc.

If an error occurs, the bootstrapper uploads the error log as well.

DEPLOYMENT PROCEDURE

There are 3 VS Solutions involved in deployment:

1) Client/**Client.sln** – This solution builds RobloxApp.exe and RCCService.exe
2) Client/Installer/**Installer.sln** – Builds and uploads installation files to AmazonS3
3) IISReset/**IISReset.sln** – Instructs cloud machines to upgrade

BUILD THE PRODUCTS

First you need to build the projects in **Client/Client.sln**. The Roblox project builds **RobloxApp.exe** and the RCCService project builds **RCCService.exe**.

Before building you will usually want to increment version numbers. The Roblox project contains a version resource, where you can change the version number that is displayed to end users in Roblox Studio's About Box. The RCCService contains a version resource as well. The first 3 parts of this version number must match the version number in the grid service (See deployment below). Finally, if any network-related changes have happened, you'll need to change the Guid found in Client/Network/Api.cpp. This should have already been done by a developer when he made a breaking change to the Networking or Data Model.

The other thing to configure is the installation domain. The Bootstrapper project has 2 string resources that need to be configured:

        IDS_INSTALLHOST   The Amazon S3 domain for installation. Examples are setup.roblox.com, setup-dev.roblox.com

        IDS_BASEHOST   The domain used by RobloxApp.exe for browsing. Examples are www.roblox.com, dev.roblox.com

BUILD AND UPLOAD INSTALLATION FILES

Once the products have been built, you need to build the installation files and upload them to Amazon S3. Build the entire **Client/Installer/Installer.sln** Solution. This will build all the installer/bootstrapper binaries and will also build an msi file used for cloud deployment later.

After building everything, it is time to run the **Uploader** project. Before running, make sure that the Working Directory for Uploader.exe is the Client/ folder. Also, you will need to place a configured version of Roblox.Amazon.Properties.Settings.config in the executable's bin directory. The website machines have this file. *Please note that this file contains password information for AmazonS3. It should not be stored in an unsecure location.*

Uploader.exe extracts the IDS_INSTALLHOST string from the boostrapper's resource fork (See how to configure this in the previous section). It will use this domain for uploading the installation files. The console will display this URL so that you can confirm it is correct.

Now you have a few options:

        C – Client upload

        R – RCCService upload

        B – Both

After choosing your option, Uploader.exe will push the required installation files to Amazon S3. If you picked "C" or "B" then end-users will instantly get updates the next time they play. If you are deploying to the cloud you need to perform the next step.

DEPLOY TO CLOUD

Open **IISReset/IISReset.sln**.

Open Addresses.txt and make sure it contains the addresses of the cloud machines you want to deploy to.

Start the IISReset project.

You will see a lot of options. To upgrade machines, pick option #3.

You will now be prompted for which Amazon S3 domain to use for deployment. Chose the domain that you configured earlier.

You can verify that installation worked by querying version numbers using command #18.

INSTALLER COMPARISON

		Old	New
User Experience	Clicks to Install & Play	4	3
	Clicks to Play	1	1
	Clicks to Update & Play	4 – Updates on Vista display an "unknown publisher" prompt	1
	Installer GUI	Sequence of 3 windows	1 window – Vista displays a standard Task Dialog
	Reboot required	Sometimes	Never
	Install Payload	13MB – all up-front	340KB up-front 8.6MB during install
	Update Payload	13MB	~2.5MB
Security	Vista Administrator	UAC elevation prompt	Yes
	Vista Standard User	UAC admin password prompt	Yes
	Vista Guest	UAC admin password prompt	Yes
	WinXP Administrator	Yes	Yes
	WinXP Limited User	Error 1925	Yes
	WinXP Guest	Error 1925	Yes
	Antivirus	?	?
Stats	Install/Update failure rate	?	<1%

TODO

~~Delete old files after successful install, not before~~

~~Don't allow cancel after installer point-of-no-return~~

Can we undo if install fails after installer point-of-no-return?

Cleanup all Roblox folders on uninstall

Cleanup Roblox content folder – within RobloxApp?

Fix WinXP folder error (for proper uninstall)

Can XP SP1 go per-user? If so, deal with legacy all-user installs

UNINSTALL CLEANUP

HKCU\Software\Roblox Corporation

REQUIRED

REVIEW/DIAGNOSE ANTIVIRUS ISSUES

SECURITY

SITELOCK

ROBLOXAPP.EXE

LATER

UAC OFF

If you first install per user (with UAC on) and then turn UAC off, the COM registration gets tricky. The installer deploys to HKLM, but Vista SP1 still uses HKCU to initiate COM (but then later goes to HKLM – this is a known Vista bug).

So, to work around this, when we do an all-user deploy, we unregister the HKCU COM stuff.

Here is what we should do: Uninstall any per-user instance of Roblox, and do it any time you log on with any user account.

CLIENTINSTALLER.ASCX SHOULD ALSO HANDLE UPDATES, NOT JUST RAW INSTALLS

This will prevent game requests from being initiated before the client is ready

PLACE ROBLOX.EXE IN A LESS-FREQUENTLY CHANGED FOLDER

If it keeps moving around then shortcuts to it will go out-of-date

It could go in a folder whose name is the version number of the exe, rather than the GUID folder used for the program files

STREAMLINE THE FULL BUILD PROCESS

SIGN ROBLOXAPP.EXE

VISTA "GAMES" INSTALL

TRY CAB FILE (ACTIVEX INSTALL)

UPLOAD LOGS AND DMP FILES IN THE EVENT OF A CRASH.

REVIEW BOOTSTRAPPER.CPP AND TRY TO MAKE IT LESS LIKE SPAGHETTI

CLEANUP

CLEAN UP OLD ELEVATION POLICY REGISTRY ITEMS

DELETE OLD FOLDERS IN VERSIONS DIRECTORY

UNINSTALL SHOULD DELETE ALL FILES, INCLUDING ROBLOX.EXE

Roblox.exe should spawn a copy in %temp% and run that instead.