



WELCOME TO THE RAGE TOOL KIT

Thank you for checking out the RAGE Tool Kit—your key to modding the first game using our premiere tech, the idTech 5 engine.

Game design brings together a number of creative disciplines, so it is impossible to detail everything you can create with these tools, but to get you started this guide offers an overview of the Kit—and a few examples of how to use it.

Keep in mind that this is our internal tech used to create RAGE and should not be interpreted as consumer-ready. This download is provided on an “as is” basis only for the technically sophisticated and adventurous!

If you haven’t already, please read the RAGE Tool Kit Terms and Conditions (“Editor EULA”) carefully. The Editor EULA is attached as Attachment A to the RAGE Tool Kit FAQs. If you do not accept the Editor EULA, do not download or use the RAGE Tool Kit. By downloading and/or using the RAGE Tool Kit, you agree to the terms and conditions set forth in the Editor EULA.

WHAT’S INCLUDED

IDSTUDIO

idStudio is your main world building tool -- it includes a multitude of components to get your new RAGE mod up and running. Use it to get an overview of the editing environment, or to learn about the various browsers and windows you will use while creating your masterpiece. You can also check out some of the toolbars you will encounter, or view the master list of key commands. Once you are familiar with idStudio and its features, take a look at the tutorials for a few step-by-step lessons on building environments.

TUTORIALS

After mastering the basics of the World Editor, explore the tutorial section and take your project to the next level. Here you will learn the nuts and bolts of constructing a map, using the particle system, adding lights to your map, and stamping a map for a completely unique look.

RAGE MAPS

An invaluable way to understand the tech is to see how some of our designers at id Software constructed playable game maps. We have included three original RAGE maps for this purpose - the Dusty 8 race track, the town of Wellspring, and The Well.

GAME ASSETS

You have access to virtually every weapon, enemy, animation, particle, and gadget we had when designing RAGE and "The Scorchers" DLC. From models and animations, to weapons and pickups, and stamps and textures -- we've provided all the assets necessary to build gorgeous new RAGE maps.

INITIAL SETUP

BUILDING ASSETS

When getting started, it's important to remember that you just downloaded many of the raw assets for creating RAGE. The first time you load a map or asset, the engine will create the compressed versions of the files that will be used by the game. For example, most of the space in the Tool Kit is used by .TGA's -- these get compressed into a .BIMAGE for the engine to use.

Note: the first time you load up the tools, this process may take a while.

Once you have these compressed versions, the engine begins tracking your changes and only updates these "digested" files when needed, which is the reason we cannot give you prebuilt digested files. We did, however, include a configuration file you can run to build everything we included. This process can take hours to complete, but we recommend running it before working in the tech.

Instructions for pre-building compressed assets:

1. Launch the RAGE Tool Kit through Steam
2. From the main menu, drop the console with the tilde key (~) and type 'exec buildAssets.cfg' (without the single quotes)
3. Press the ENTER key

Note: This will initially launch idStudio -- if this is the first time idStudio has launched on your machine, it may popup a dialog box letting you know a preferences file was not found and one will be created. You will need to accept this dialog before the process will continue!

The game will begin loading each of the provided maps. Once the process is complete, the game will shut itself down. You are ready now to re-launch the Tool Kit and your load time will be significantly reduced.

IMPORTANT FILES AND DIRECTORIES

- The digested files (the compressed versions of the raw assets) end up under your 'rage tool kit/base/generated' directory. If you notice an asset you changed not updating correctly, try wiping its generated output to force the engine to recompress.
- The .VMTR (Virtual Material) file is located in the 'rage tool kit/virtualtextures' directory. Whereas the generated output contains the raw data the engine needs (things like vertices and normals), the VMTR file contains the texture data for all dynamic media in the game (like the player's hands and weapons).
- The megatextures you build (and that we provided for the included maps) are also located in the 'virtualtextures' directory. Megatextures (.page files) contain the texture data for all static media baked into the map. Maps that have a terrain (like Dusty 8) will also have a terrain megatexture.
- Another important folder is the 'rage tool kit/SAVES' directory. This will contain the log files created every time you run the Tool Kit. Log files are extremely important and are the first stop when figuring out why the game crashed.

WHEN THINGS GO WRONG

Working with software can be meticulous, and things need to happen a certain way. When the Tool Kit crashes, it will fail with the error and callstack. It is important to investigate these so you can fix the issue.

While running the Tool Kit, the log file 'qconsole.log' is located under your 'SAVES/base' directory. Every time you launch the game, a new log generates. You can view this information in any text editor.

When building mods, a unique 'mods/[modName]' folder appears under your 'SAVES' directory. This will contain the output of the build process used to create the final package. It is rare that the build process will fail if the map loads in idStudio and the engine, but it can happen.

After opening the log file, scroll to the bottom of the file to find the error (usually written with 'FatalError:') and the callstack.

WHAT TO EXPECT

BUILD TIMES AND HARDWARE

Whether you are loading a map, baking a megatexture, or building a mod -- these are intensive processes that utilize your hardware to its full capabilities. The more cores and RAM you have available, the faster things will run. A dual-core machine with 4 GB of RAM will work, but it will be substantially slower than a quad-core with 12 GB of RAM. Not only are these processes time and cpu intensive, but total time required to complete these operations depends on how much content your map, or mod, contains. Here are some benchmarks.

TYPICAL BUILD TIMES:

- Initial map loads: 20–60 minutes
- Load times after initial load: 1–3 minutes
- Build an asset mod (does not include any maps): 10–30 minutes
- Build a mod that includes one map about the size of Wellspring: 20–40 minutes

The time to bake a map is dependent on the size of the map and number of entities included. Fully built maps like The Well, Wellspring, and Dusty 8 can take several hours to build on a local system.

DISK SPACE

The Tool Kit is initially 35 GB. If you run the 'buildAssets.cfg' like we recommend it will generate another 6 GB of data, bringing you up to 43 GB. From there, the additional disk space is completely dependent on what you build and do with the Tool Kit.

When building mods, the size they occupy will depend on what you are building. A mod that does not include any maps and only modifies existing game content will generate about 200 MB of data. A mod of Wellspring may only be about 500 MB, depending on how much content you alter/add.