

PowerGrid Guide for Plugin developers

Patrick Kramer (Chronio)
Vincent Wassenaar (Alaineman)

April 10, 2014

Abstract

This guide is intended for developers interested in developing plugins for PowerGrid. It contains overviews of various parts of the PowerGrid API, examples of how to use these APIs, and in-depth technical details on how PowerGrid works. It is not primarily intended for normal users of PowerGrid, although it may be interesting for those who wish to know more about PowerGrid and its inner workings.

1 General Overview

PowerGrid consists many individual components, ranging from UI components to C++ classes communicating with the Java Virtual Machine. All these various components are organized in a few subprojects. An overview with a brief description for each subproject follows:

Bridge The core of PowerGrid. This subproject provides functionality to interact with the Java Virtual Machine and as such contains some classes critical to PowerGrid's operations.

PluginFramework Features (pure virtual) classes used for implementing extensions to PowerGrid in the form of "Plugins". Qt's Plugin framework is used for the realization of this feature.

UserInterface contains PowerGrid's UI components. This subproject also is responsible for starting the PowerGrid loader (implemented in Java) and showing the main user interface.

Loader The only part of PowerGrid implemented in Java. It replicates the official Runescape loader in providing the right environment for the Runescape Applet to run. It also plays a role in acquiring the updater data and provide the C++ PowerGrid client with a way to load the classes of the Runescape client.

PowerGrid relies on Qt for the user interface components and some data structures (like QList and QMap). PowerGrid also makes use of Qt's meta-object system to handle objects in a type-safe way. Additionally, for performance and usability reasons, PowerGrid represents the Runescape world as a collection of Entities. Contrary to "normal" object-oriented design, this entity framework operates on properties of entities rather than on object types alone. This allows easy creation of generalized versions of tasks. For more information on the inner workings of the entity framework, read the documentation for the "entity" namespace or refer to chapter 2

2 The Bridge subproject

This subproject focuses on bridging the gap between Java and C++, but also between Runescape's obfuscated classes and a clear, accurate representation of the Runescape world. To achieve this, multiple steps are needed to represent Runescape in an easy and intuitive way. The components involved in each of these steps are highlighted below.

- 2.1 JACE: The Java-C++ bridge
- 2.2 Proxies: A low-level representation
- 2.3 Entities: A high-level representation
- 3 The PluginFramework subproject
- 4 The UserInterface subproject
- 5 The Java loader