Version 0.1a
22 Dec 2019

Quake 1 Dev Kit for Dynamic Map Logic

Authored by Qalten
(Qalten#6722)

Entity Reference

This document will give a brief description and available fields/values for all new entities present in the CPQ dev kit. All references are as of the version number listed on the front page of this document and may be superceded or deprecated in future versions.

There is a FGD file (currently for TrenchBroom) included with the dev kit. As of this version, all entity source code is currently found in **triggers.qc**.

NEW FIELDS

.operationtype	Used in most CPQ entities, defines behavior
.target_reg_1	Target register (primary)
.target_reg_2	Target register (secondary)
.target_reg_result	Target register (result of **trigger_operator**)
.target_reg_entity	Target entity to read/write w/register
.target_expr_true	Target trigger if evaluated to true (boolexpr only)
.target_expr_false	Target trigger if evaluated to false (boolexpr only)
.message_expr_true	Message shown if evaluated to true (boolexpr only)
.message_expr_false	Message shown if evaluated to false (boolexpr only)

FIELD UPDATES

Most CPQ triggers make use of the **spawnflags** field, using bitflag 2 for "Print Result as Message". Additionally, all CPQ entities with the **trigger_** prefix use the same base class as **trigger_multiple**. So any builtin fields (such as **message**, **target**, etc) will function the same.

ENTITY DEFINITIONS (with relevant fields)

info_register	(Point entity) Core entity of the CPQ dev kit. Stores values given in the **count** field.
trigger_setregister	(Brush entity) Sets the value of the register defined in **target_reg_1**. If **target_reg_entity** is provided it will use that, otherwise it defaults to **activator**. **operationvalue** behaviors: 0 : Target Health (any target **health**) 1 : Player Armor (player only **armorvalue**) 2 : Skill/Difficulty (global **skill**) 3 : Target Position (x) (any target **origin.x**)

	4 : Target Position (y) (any target **origin.y**) 5 : Target Position (z) (any target **origin.z**) 6 : Target Angle (x) (any target **v_angle.x**) 7 : Target Angle (y) (any target **v_angle.y**) 8 : Target Angle (z) (any target **v_angle.z**) 9 : Player Items Bitflags (player only **items**) 10 : **Count** value of this trigger
trigger_useregister	(Brush entity) Reads the value of the register defined in **target_reg_1**. If **target_reg_entity** is provided it will use that, otherwise it defaults to **activator**. **operationtype** behaviors: 0 : Apply Normal Damage (uses **T_Damage** method) 1 : Heal (no overheal) (uses **T_Heal** method) 2 : Heal (overheals) (uses **T_Heal** method) 3 : Set Current Health (**health**) 4 : Set Max Player Health (**max_health** & **health**) 5 : Set Player Armor (sets player **armorvalue,** defaults to **armortype** 0.8) 6 : Add Player Armor (adds to player **armorvalue**) 7 : Set Player Item Bitflags (sets player **items**) 8 : Add Player Ammo Shells (adds to **ammo_shells**) 20 : Set Door/Platform Lip (sets entity **lip** and recalculates **pos2**) 21 : Set Door/Platform Speed (sets entity **speed**) 22 : Set Door/Platform Wait (sets entity **wait**) 23 : Set Door/Platform Angle (sets entity **angle,** appropriately setting -1 and -2 values) 99 : Set trigger operationtype (sets target **operationtype,** can be used to dynamically alter how other CPQ triggers work with register values (ie: making a **trigger_useregister** do damage instead of healing).
trigger_operator	(Brush entity) Reads the value of **target_reg_1** and completes a mathematical operation on it (driven by **operationtype**) and stores the result in **target_reg_result**. If **target_reg_2** is defined, it will use that as the second term in the operation, otherwise it will use the trigger's **count**. **operationtype** behaviors:

	0 : Set Value (will use **count** if no **target_reg_1**) 1 : Value Build (Allows for a value entry similar to a calculator, where each number adds a place (ie: if the value was 4, and the next was 3, the result would be 43. Five digits max) 2 : Add 3 : Subtract 4 : Multiply 5 : Divide 90 : Debug - Print Current Value (no value set) 91 : Debug - Print Current Operation Type (no value set)
trigger_boolexpr	(Brush entity) Reads **target_reg_1** and evaluates it either against **target_reg_2** or the trigger's own **count**. **operationtype** defines evaluation. It can (but doesn't have to) trigger **target_expr_true** or **target_expr_false** based on result. Also can display **message_expr_true** or **message_expr_false** based on result. **operationtype** behavior: 0 : Equal 1 : Not Equal 2 : Greater Than 3 : Greater Than/Equal to 4 : Less Than 5 : Lesser Than/Equal to