

idProp2

An entity that will spawn a pickup or barrel.

Health

```
entity {
  entityDef pickups_pickup_health_large_1 {
    inherit = "pickup/health/large";
    class = "idProp2";
    expandInheritance = false;
    poolCount = 0;
    poolGranularity = 2;
    networkReplicated = true;
    disableAIPooling = false;
    edit = {
      renderModelInfo = {
        model = "art/pickups/health/health_pack_big_a.lwo";
        contributesToLightProbeGen = false;
        ignoreDesaturate = true;
        emissiveScale = 0.5;
        scale = {
          x = 1.75;
          y = 1.75;
          z = 1.75;
        }
      }
    }
    spawn_statIncreases = {
      num = 1;
      item[0] = {
        stat = "STAT_ITEMS_SPAWNED";
        increase = 1;
      }
    }
    equipOnPickup = true;
    lootStyle = "LOOT_TOUCH";
    triggerDef = "trigger/props/pickup";
```

```

    isStatic = true;
    canBePossessed = true;
    removeFlag = "RMV_CHECKPOINT_ALLOW_MS";
    difficultyScaleType = "DST_PICKUP";
    pickup_statIncreases = {
        num = 1;
        item[0] = {
            stat = "STAT_HEALTH_PICKUP";
            increase = 1;
        }
    }
    useableComponentDecl = "health/sp_health_large";
    spawnPosition = {
        x = 1;
        y = 1;
        z = 1;
    }
    spawnOrientation = {
        mat = {
            mat[0] = {
                x = 0;
                y = 1;
            }
            mat[1] = {
                x = -1;
                y = 0;
            }
        }
    }
    dormancy = {
        playerDistance = 20;
        playerRearwardDistance = 20;
    }
}

```

Armor

```

entity {
    entityDef pickups_pickup_armor_large_1 {
        inherit = "pickup/armor/large";
        class = "idProp2";
        expandInheritance = false;
        poolCount = 0;
        poolGranularity = 2;
        networkReplicated = true;
        disableAIPooling = false;
        edit = {
            renderModelInfo = {
                model = "art/pickups/armor/pickup_armor_01.lwo";
                contributesToLightProbeGen = false;
                ignoreDesaturate = true;
                scale = {
                    x = 2;
                    y = 2;
                    z = 2;
                }
                emissiveScale = 0.5;
            }
            spawn_statIncreases = {
                num = 1;
                item[0] = {
                    stat = "STAT_ITEMS_SPAWNED";
                    increase = 1;
                }
            }
            equipOnPickup = true;
            lootStyle = "LOOT_TOUCH";
            triggerDef = "trigger/props/armor_large";
            isStatic = true;
            canBePossessed = true;
            removeFlag = "RMV_CHECKPOINT_ALLOW_MS";
            difficultyScaleType = "DST_PICKUP";
            pickup_statIncreases = {
                num = 1;
                item[0] = {
                    stat = "STAT_ARMOR_PICKUP";

```

```

    []increase = 1;
  []
}

[]useableComponentDecl = "armor/sp_armor_50";
[]spawnPosition = {
  []x = 1;
  []y = 1;
  []z = 1;
}
[]spawnOrientation = {
  []mat = {
    []mat[0] = {
      []x = 0;
      []y = 1;
    }
    []mat[1] = {
      []x = -1;
      []y = 0;
    }
  }
}
[]dormancy = {
  []playerDistance = 20;
  []playerRearwardDistance = 20;
}
}
}

```

Ammo

```

entity {
  []entityDef pickups_pickup_ammo_shells_1 {
    []inherit = "pickup/ammo/shells";
    []class = "idProp2";
    []expandInheritance = false;
    []poolCount = 0;
    []poolGranularity = 2;
  }
}

```

```
[networkReplicated = false;
disableAIPooling = false;
edit = {
  renderModelInfo = {
    model = "art/pickups/ammo/ammo_shotgun_01.lwo";
    contributesToLightProbeGen = false;
    ignoreDesaturate = true;
    emissiveScale = 0.2;
    scale = {
      x = 1.39999998;
      y = 1.39999998;
      z = 1.39999998;
    }
  }
  spawn_statIncreases = {
    num = 1;
    item[0] = {
      stat = "STAT_ITEMS_SPAWNED";
      increase = 1;
    }
  }
  equipOnPickup = false;
  lootStyle = "LOOT_TOUCH";
  triggerDef = "trigger/props/pickup";
  isStatic = false;
  canBePossessed = true;
  removeFlag = "RMV_CHECKPOINT_ALLOW_MS";
  flags = {
    canBecomeDormant = true;
  }
  fxDecl = "pickups/ammo_shotgun";
  difficultyScaleType = "DST_PICKUP";
  updateFX = true;
  pickup_statIncreases = {
    num = 2;
    item[0] = {
      stat = "STAT_AMMO_PICKUP";
      increase = 1;
    }
  }
}
```

```

    item[1] = {
        stat = "STAT_PLACED_AMMO_PICKUP";
        increase = 1;
    }
}

useableComponentDecl = "propitem/ammo/shotgun_10";

spawnPosition = {
    x = 1;
    y = 1;
    z = 1;
}

spawnOrientation = {
    mat = {
        mat[0] = {
            x = 0;
            y = 1;
        }
        mat[1] = {
            x = -1;
            y = 0;
        }
    }
}

dormancy = {
    playerDistance = 20;
    playerRearwardDistance = 20;
}
}
}

```

Extra Life

```

entity {
    layers {
        "game/sp/extralives/extralives_many" // the layer prevents it from spawning in Ultra
        Nightmare difficulty
    }
}

```

```
EntityDef pickups_pickup_extra_life_extra_life_1 {
    inherit = "pickup/extra_life/extra_life_1";
    class = "idProp2";
    expandInheritance = false;
    poolCount = 0;
    poolGranularity = 2;
    networkReplicated = true;
    disableAIPooling = false;
    edit = {
        whenToSave = "SGT_NORMAL";
        saveType = "SGS_GAME_DATA";
        removeFlag = "RMV_IMMEDIATE_ALLOW_MS";
        automapPropertiesDecl = "extra_life";
        renderModelInfo = {
            model = "art/pickups/extraLife.lwo";
            contributesToLightProbeGen = false;
            ignoreDesaturate = true;
            scale = {
                x = 1.75;
                y = 1.75;
                z = 1.75;
            }
            emissiveColor = {
                r = 0;
                b = 0.68235302;
            }
            emissiveScale = 1;
        }
        clipModelInfo = {
            type = "CLIPMODEL_BOX";
        }
        fxDecl = "pickups/ammo_extralife";
        physicsAttributes = "lootdrop/health";
        useableComponentDecl = "sp_extra_life_1";
        thinkComponentDecl = "bobthink";
        triggerDef = "trigger/props/pickup_large";
        updateFX = true;
        canBePossessed = true;
        sendNotableItemTelemetryEvent = true;
    }
}
```

```

    spawnPosition = {
        x = 1;
        y = 1;
        z = 1;
    }
    spawnOrientation = {
        mat = {
            mat[0] = {
                x = 0;
                y = 1;
            }
            mat[1] = {
                x = -1;
                y = 0;
            }
        }
    }
    flags = {
        canBecomeDormant = true;
        forcePhysicsUpdate = true;
        forceFXThink = true;
    }
    bindInfo = {
        bindParent = "movers_func_mover_1"; // bind it to an idMover
        bindOriented = true;
        lockLocalOffset = true;
    }
    dormancy = {
        playerDistance = 30;
        playerRearwardDistance = 30;
    }
}
}

```

Barrel

```

entity {
    entityDef game_envhazard_barrels_red_1 {
        inherit = "envhazard/barrels/red";
        class = "idProp2";
        expandInheritance = false;
        poolCount = 0;
        poolGranularity = 2;
        networkReplicated = true;
        disableAIPooling = false;
        edit = {
            removeFlag = "RMV_CHECKPOINT_ALLOW_MS";
            flags = {
                forcePhysicsUpdate = true;
                takedamage = true;
            }
            renderModelInfo = {
                model = "art/breakable/barrel/barrel_explode_a.lwo";
            }
            clipModelInfo = {
                type = "CLIPMODEL_CYLINDER";
                contentsFilter = {
                    playerClip = false;
                }
                numSides = 10;
            }
            networkSerializeTransforms = true;
            killerNames = {
                num = 1;
                item[0] = "#str_decl_damage_hazard_explosive_barrel_GHOST53200";
            }
            fxDecl = "breakable/barrel_01";
            physicsAttributes = "barrels";
            isStatic = false;
            damageComponentDecl = "barrel/default";
            lootDrop = {
                lootDropDataDecl = "barrels/rune";
            }
            canBePossessed = true;
            spawnPosition = {

```

```

    x = 1;
    y = 1;
    z = 1;
}
}
}
}

```

Key Card

```

entity {
    entityDef pickups_keycard_uac_red_1 {
        inherit = "pickup/keycard/uac_red";
        class = "idProp2";
        expandInheritance = false;
        poolCount = 0;
        poolGranularity = 2;
        networkReplicated = true;
        disableAIPooling = false;
        edit = {
            renderModelInfo = {
                model = "art/pickups/keycard_pickup_r.lwo";
                contributesToLightProbeGen = false;
                materialRemap = {
                    num = 1;
                    item[0] = {
                        from = "art/pickups/keycard_r";
                        to = "art/pickups/keycard_r_dlc";
                    }
                }
            }
            emissiveColor = {
                g = 0.0333333015;
                b = 0;
            }
        }
        spawn_statIncreases = {
            num = 1;
            item[0] = {
                stat = "STAT_ITEMS SPAWNED";
                increase = 1;
            }
        }
    }
}

```

```

    }
}

equipOnPickup = true;
lootStyle = "LOOT_TOUCH";
triggerDef = "trigger/props/progression_small";
isStatic = true;
canBePossessed = true;
whenToSave = "SGT_CHECKPOINT";
removeFlag = "RMV_CHECKPOINT_ALLOW_MS";
flags = {
    canBecomeDormant = true;
}
clipModelInfo = {
    type = "CLIPMODEL_BOX";
}
useableComponentDecl = "propitem/keycard/uac_red";
thinkComponentDecl = "bob_rotate";
sendNotableItemTelemetryEvent = true;
spawnPosition = {
    x = 1;
    y = 1;
    z = 1;
}
spawnOrientation = {
    mat = {
        mat[0] = {
            x = 0;
            y = 1;
        }
        mat[1] = {
            x = -1;
            y = 0;
        }
    }
}
fxDecl = "dlc1/e4m1_rig/red_keycard";
}
}

```

