

Entities (2016)

A description of various entities that shape the level.

- [Entities List \(Doom 2016\)](#)
- [idTarget_Command](#)
- [idTarget_Remove](#)
- [idTarget_Hide](#)
- [idTarget_Show](#)
- [idTarget_Spawn](#)
- [idTarget_Teleport](#)
- [idTrigger](#)
- [idResurrectionProxy](#)
- [idSummoningTemplate](#)
- [idVolume_PlayerEnvOverride](#)

Entities List (Doom 2016)

A complete list of every entity class in Doom 2016:

```
idAASObstacle
idAASReachability
idAbnormalPlayerMetrics
idActionNode
idActor
idActorAmbientFilter
idActorWorkNode
idAFEntity_Corpse
idAFEntity_Dummy
idAFEntity_Generic
idAI2
idAIAASHint
idAICombatHint
idAICombatHint_SuppressionFirePoint
idAIConductor
idAIConductor_Coop
idAIDeathVolume
idAIDebugCamera
idAIHint
idAIHintGroup
idAIInteraction
idAISearchHintAnimation
idAISearchHintAuto
idAISearchHintGroup
idAITest
idAITest_Rotation
idAlignedEntity
idAmbientTriggerModifier_SetAITypes
idAnimated
idAnimated_AnimWeb
idAnimated_AnimWeb_Coop
idAnimated_AnimWeb_Samuel
idAnimated_DemonicBait
idAnimated_DemonicEcho
idAnimated_SMM_Pillar
```

idAnimated_ThreatSensor
idAnimatedAttachment
idAnimatedAttachment_AF
idAnimatedEntity
idAnimatedInteractable
idAnimCamera
idAnimNode
idAnnouncementManager
idArcadeScoreBonus
idARTarget_Info
idAutoMapEntity
idAutoMapManager
idBarricade
idBasePropSpawner
idBecomeDemonPropSpawner
idBFG_Geyser
idBfgBurn
idBillboard
idBinaryAnimation
idBinaryModel
idBotActionEntity
idBotActionEntity_Info
idBotActionEntity_Info_POI
idBotActionEntity_Info_RoamGoal
idBotActionEntity_Usable
idBotPathNodeEntity
idBreakable
idBreakablePlatform
idBuildArea
idCamera
idCameraView
idChargeballComponentManager
idChargeballGoal
idChargeballGoalCamera
idChargeballPickup
idChargeballPlayerStart
idChargeballSpawner
idCinematicCamera
idClothEntity
idCollisionStreamArea

idCombatGrouping
idCombatVolume
idCoopStartingInventory
idCorpse
idCorpseManager
idCoverBlocker
idCyberdemonMissileShooter
idDamageableEntity
idDamageableTower
idDeathmatchPlayerStart
idDebrisEntity
idDecalEntity
idDemonicChainGrappleInfo
idDemonPlayer
idDemonPlayer_Bar on
idDemonPlayer_Cacodemon
idDemonPlayer_Harvester
idDemonPlayer_Mancubus
idDemonPlayer_Pinky
idDemonPlayer_Prowler
idDemonPlayer_Revenant
idDemonPlayer_Summoner
idDemonPlayerStart
idDesignSystems_AnimatedBlocker
idDesignSystems_Beacon
idDesignSystems_Checkpoint
idDesignSystems_ControlBox
idDesignSystems_Door
idDesignSystems_Elevator
idDesignSystems_ForceFieldGate
idDesignSystems_GuiDisplay
idDesignSystems_HackBox
idDesignSystems_HandScanner
idDesignSystems_LevelTransitionStation
idDesignSystems_LootBox
idDesignSystems_NarrativeTable
idDesignSystems_PowerCell
idDesignSystems_Receptacle
idDesignSystems_ResupplyStation
idDesignSystems_SaveStation

idDesignSystems_SecurityUnlockStation
idDesignSystems_Turret
idDesignSystems_UpgradeStation
idDesignSystems_VirtualGUI
idDesignSystemsEntity
idDestroyableProp
idDynamicEntity
idDynamicEntity_Damageable
idDynamicSnapMapEntity
idDynamicStampEntity
idDynamicStampEntity_Coop
idDynamicVisibilityBlocker
idDynamicWaterEntity
idEditorModelEntity
idElectricBoltEmitter
idEncounterAmbientAudio
idEncounterGroupMgr
idEncounterManager
idEncounterModifier_Reset
idEncounterModifier_SetEnabledState
idEncounterModifier_SetNextScriptIndex
idEncounterTrigger_AmbientAIAudio
idEncounterTrigger_Commit
idEncounterTrigger_Exit
idEncounterTrigger_OverrideCombatGrouping
idEncounterTrigger_PreCombat
idEncounterTrigger_RaiseUserFlag
idEncounterVolume_ValidCover
idEntity_Safehouse_Location
idEntityCamera
idEntityFog
idEntityFx
idEntityFxRandom
idEntityImposter
idEnvArea
idEnvironmentalDamage_Hurt_Trigger
idEnvironmentalDamage_Point
idEnvironmentalDamage_PointManager_Trigger
idFlightVolumeTraversalSpline
idFoliage

idFreeCamera
idFuncCloth
idFuncRotate
idFuncShadowCaster
idFuncSwing
idGameChallenge
idGameChallenge_Bloodthirst
idGameChallenge_CampaignSinglePlayer
idGameChallenge_Chargeball
idGameChallenge_ClanArena
idGameChallenge_Conveyor
idGameChallenge_Coop
idGameChallenge_Coop_HudInfo
idGameChallenge_Coop_Variables
idGameChallenge_Domination
idGameChallenge_DoubleHelix
idGameChallenge_Elimination
idGameChallenge_FFA
idGameChallenge_FlagMode
idGameChallenge_FreezeTag
idGameChallenge_Possession
idGameChallenge_PVP
idGameChallenge_SafeHouse
idGameChallenge_Sector
idGameChallenge_TDM
idGameSystems_Manager_ContainerEntity
idGlass
idGlobalDataComponent
idGodRays
idGoreEntity
idGuiEntity
idGuiEntityCinematic
idGuiEntityCinematicWide
idGuiEntityCounter
idGuiEntityStatic
idHologram
idInfluenceSpawnSettings
idInfo
idInfo_BounceDestination
idInfo_DoorTraversalChain

idInfo_GatedTraversalChain
idInfo_GatedTraversalSet
idInfo_SpawnConditionProxy
idInfo_TeleportDestination
idInfo_TraversalChain
idInfo_TraversalPoint
idInfo_Trigger_Facing_Target
idInfo_UniversalTraversal
idInfoAAS
idInfoAmbient
idInfoBounds
idInfoCloudShot
idInfoCover
idInfoCoverExposed
idInfoDebugMapName
idInfoDebugMarker
idInfoDebugText
idInfoExportHint
idInfoFlightVolumeEntrance
idInfoFocus
idInfoGoalPosition
idInfoLevelFadeIn
idInfoLookTarget
idInfoLookTargetGroup
idInfoOrbit
idInfoPath
idInfoPlayerHud
idInformationEntity
idInfoRoam
idInfoRuler
idInfoSpawnPoint
idInfoSplineChild
idInfoTexLod
idInfoTraversal
idInfoTraversalEndPoint
idInfoWaitForButtonAfterLoad
idInteractable
idInteractable_EliteGuard
idInteractable_EliteGuard_Coop
idInteractable_GiveItems

idInteractable_LootCrate
idInteractable_LootDrop
idInteractable_Minigame
idInteractable_Moveable
idInteractable_NightmareMarker
idInteractable_Obstacle
idInteractable_Obstacle_SnapDoor
idInteractable_PowerCoreReceptacle
idInteractable_Tutorial
idInteractable_VegaTraining
idInteractable_WeaponModBot
idInteractable_WorldCache
idInteractionCamera
idInventoryStorage
idItemPropSpawner
idJostleAnimated
idJostleSpring
idJostleSwivel
idLaserHazard
idLensFlare
idLight
idLoot_Blocker
idLoot_ItemCase
idLowGravityMover
idMapSpecificUnlockables
idMover
idMoverModifier
idMultiplayerTrigger
idNarrative_Interactable
idNarrativeDirector
idNavSplinePath
idNetworkedEntityFx
idNetworkedParticleEmitter
idObjective_Relay
idOverTheShoulderCamera
idParticleEmitter
idPathCorner
idPerceptionVolume
idPieceEmitter
idPlaceableSnapAIEncounter

idPlayer
idPlayerStart
idPortalSurface
idPortalWorld
idPowerLine
idProjectile
idProjectile_AutoTurret
idProjectile_BfgArc
idProjectile_BfgBurn
idProjectile_Boomerang
idProjectile_CacoDemonRocket
idProjectile_Chargeball
idProjectile_CyberdemonSwarmMissile
idProjectile_DamageOverTime
idProjectile_DeployedLaser
idProjectile_Dispenser
idProjectile_Drone
idProjectile_EMP
idProjectile_GBF
idProjectile_Grenade
idProjectile_Grenade_DemonControl
idProjectile_GroundDeploy
idProjectile_GuardBoulder
idProjectile_HarvesterDetonateBall
idProjectile_Nanobore
idProjectile_PersonalTeleporter
idProjectile_PlasmaPuddle
idProjectile_Proximity_Mine
idProjectile_Rocket
idProjectile_RocketBounce
idProjectile_SiphonGrenade
idProjectile_Slicer
idProjectile_StealthField
idProjectile_StrikerGrenade
idProjectile_SwarmMissile
idProjectile_SwarmMissile_V2
idProjectile_ThreatSensor
idProp
idProp_AIArmor
idProp_ArmorPickup

idProp_Breakable
idProp_BreakableLoot
idProp_ContinuallyUsed
idProp_Conveyor_ControlPoint
idProp_Coop
idProp_Coop_Billboard
idProp_DemonCircle
idProp_Domination_ControlPoint
idProp_DoubleHelix_CapturePoint
idProp_DoubleHelix_Flag
idProp_ElectricArmor
idProp_Explosive
idProp_FlagMode_CapturePoint
idProp_FlagMode_Flag
idProp_HealthPickup
idProp_Loot
idProp_Moveable
idProp_OnlineCollectible
idProp_Pickup
idProp_PlayableDemonCircle
idProp_PowerCell
idProp_Sector_ControlPoint
idProp_Spawnable
idProp_Static
idProp_Usable
idProp_WeaponPickup
idProp_WeaponStatic
idProp2
idQuestNavSplinePath
idRailSplinePath
idRailTurret
idRandomLoot
idReferenceMap
idRegenArea
idResurrectionProxy
idRibbonEmitter
idRiftBuildPosition
idRotatableCamera
idSlowMotionCamera
idSnapAmmoPickup

idSnapBackpackPickup
idSnapDroppable
idSnapEvent
idSnapEvent_MiniBoss
idSnapEvent_Survive
idSnapEvent_WaveBattle
idSnapFilter
idSnapFilter_AI
idSnapFilter_Bool
idSnapFilter_CachedEntity
idSnapFilter_Droppable
idSnapFilter_Equipment
idSnapFilter_Module
idSnapFilter_Player
idSnapFilter_Race
idSnapFilter_Team
idSnapFilter_UserFilter
idSnapFilter_Volume
idSnapFilter_Weapon
idSnapGuiEntity
idSnapInspectCameraHint
idSnapItemHint
idSnapMapAction
idSnapMapAction_AI_SetMod
idSnapMapAction_AI_SetStaggerEnabled
idSnapMapAction_AIConductor_ClearSpawnQueue
idSnapMapAction_AIDifficulty
idSnapMapAction_AssignToTeam
idSnapMapAction_BoolVar_Set
idSnapMapAction_BoolVar_Test
idSnapMapAction_BoolVar_Toggle
idSnapMapAction_CachedEntityVar_Operation
idSnapMapAction_Camera_ActivateCamera
idSnapMapAction_Camera_DeactivateCamera
idSnapMapAction_Camera_FadeCamera
idSnapMapAction_Camera_FadeFromToCamera
idSnapMapAction_Camera_SetEnvironment
idSnapMapAction_Camera_Shake
idSnapMapAction_ChangeRace
idSnapMapAction_CheckTriggerContents

idSnapMapAction_ColorVar_Channel_Operation
idSnapMapAction_ColorVar_Operation
idSnapMapAction_ComboStart_SetEnable
idSnapMapAction_CompleteMainObjective
idSnapMapAction_CompleteSubObjective
idSnapMapAction_Counter_Modify
idSnapMapAction_Counter_Reset
idSnapMapAction_Counter_Set
idSnapMapAction_CustomEventFire
idSnapMapAction_Damage
idSnapMapAction_Door_EndLockdown
idSnapMapAction_Door_StartLockdown
idSnapMapAction_Droppable_Remove
idSnapMapAction_Droppable_Reset
idSnapMapAction_Echo_Glitch
idSnapMapAction_Echo_Opacity
idSnapMapAction_Echo_Play
idSnapMapAction_Encounter_ApplyBuff
idSnapMapAction_Encounter_AssignTeam
idSnapMapAction_EndGame
idSnapMapAction_EndGame_Conditional
idSnapMapAction_Explodable_Explode
idSnapMapAction_FinishCountDown
idSnapMapAction_FinishEncounter
idSnapMapAction_FinishSnapEvent
idSnapMapAction_FollowPath
idSnapMapAction_Gate_SetA
idSnapMapAction_Gate_SetB
idSnapMapAction_Gate_Test
idSnapMapAction_GivePlayerScore
idSnapMapAction_GiveTeamScore
idSnapMapAction_Hazard_SetDamage
idSnapMapAction_Heal
idSnapMapAction_HealAI
idSnapMapAction_HideMainObjective
idSnapMapAction_HideObjective
idSnapMapAction_Incapacitate
idSnapMapAction_Interactable_SetEnable
idSnapMapAction_Interaction
idSnapMapAction_IntTest

idSnapMapAction_IntVar_Operation
idSnapMapAction_IntVar_Random
idSnapMapAction_Kill
idSnapMapAction_KillAI
idSnapMapAction_LaserHazard_SetOn
idSnapMapAction_Light_SetIntensity
idSnapMapAction_Light_SetOn
idSnapMapAction_LoadSubObjectiveProgress
idSnapMapAction_Logic_Fire
idSnapMapAction_Logic_ResetRandomRelayUsedList
idSnapMapAction_Logic_SetDelay
idSnapMapAction_Logic_SetRandomRelayActivateCount
idSnapMapAction_Logic_SwitchFire
idSnapMapAction_LootDropRestriction
idSnapMapAction_ModifyAmmo
idSnapMapAction_Module_AllowAicSpawn
idSnapMapAction_Module_EndLockDown
idSnapMapAction_Module_LockDown
idSnapMapAction_Module_SetEnvironment
idSnapMapAction_Mover_Start
idSnapMapAction_NumTest
idSnapMapAction_NumVar_Operation
idSnapMapAction_NumVar_Random
idSnapMapAction_PauseCountDown
idSnapMapAction_PlayCallout
idSnapMapAction_Player_DropCurrentWeapon
idSnapMapAction_Player_GiveDroppable
idSnapMapAction_Player_GiveItem
idSnapMapAction_Player_GivePowerup
idSnapMapAction_Player_GiveTakeAbility
idSnapMapAction_Player_Respawn
idSnapMapAction_Player_SaveApplyRestorePoint
idSnapMapAction_Player_TakeDroppable
idSnapMapAction_Player_TakeItem
idSnapMapAction_PlayerCamera_FadeCamera
idSnapMapAction_PlayerCamera_FadeFromToCamera
idSnapMapAction_PlayerCamera_SetEnvironment
idSnapMapAction_PlayerCamera_Shake
idSnapMapAction_PlayerInput_SetEnable
idSnapMapAction_PlayerLoadout_GiveLoadout

idSnapMapAction_PlayerLoadout_SetDemonSelection
idSnapMapAction_PlayerLoadout_SetMaxWeaponSlots
idSnapMapAction_PlayerResource_Operation
idSnapMapAction_PlayerResource_Spend
idSnapMapAction_PlayMusic
idSnapMapAction_PlaySpeaker
idSnapMapAction_PostEvent
idSnapMapAction_PowerCoreReceptacle_RespawnCell
idSnapMapAction_PowerCoreReceptacle_SetCoreColor
idSnapMapAction_Prop_EnableFX
idSnapMapAction_RemoveAI
idSnapMapAction_RemoveEntity
idSnapMapAction_RemovePOI
idSnapMapAction_Repeater_Stop
idSnapMapAction_Repeater_Toggle
idSnapMapAction_ResetAicModuleCooldown
idSnapMapAction_ResetCountDown
idSnapMapAction_ResourceTest
idSnapMapAction_Respawn
idSnapMapAction_Revive
idSnapMapAction_SaveSubObjectiveProgress
idSnapMapAction_Secret
idSnapMapAction_Sequencer_Reset
idSnapMapAction_Sequencer_SetSequenceNumber
idSnapMapAction_SetColor
idSnapMapAction_SetConductorSpawningPauseState
idSnapMapAction_SetCountDown
idSnapMapAction_SetLaunchDestination
idSnapMapAction_SetLaunchSpeed
idSnapMapAction_SetPlayerModifier
idSnapMapAction_SetPlayerScore
idSnapMapAction_SetPOI
idSnapMapAction_SetSubObjectiveProgress
idSnapMapAction_SetTeam
idSnapMapAction_SetTeamScore
idSnapMapAction_Settings_SetActive
idSnapMapAction_SetTriggerActivatable
idSnapMapAction_SetTriggerVisible
idSnapMapAction_ShowMainObjective
idSnapMapAction_ShowMessage

idSnapMapAction_ShowObjective
idSnapMapAction_ShowObjectiveCounter
idSnapMapAction_ShowObjectiveHealth
idSnapMapAction_ShowObjectiveResource
idSnapMapAction_ShowObjectiveTimer
idSnapMapAction_ShowSubObjective
idSnapMapAction_SpawnEncounter
idSnapMapAction_SpawnInhibit_SetEnable
idSnapMapAction_SpawnItem
idSnapMapAction_StartCountDown
idSnapMapAction_StartIterator
idSnapMapAction_StartSnapEvent
idSnapMapAction_StopCountDown
idSnapMapAction_StopMusic
idSnapMapAction_StopSpeaker
idSnapMapAction_StringTest
idSnapMapAction_StringVar_Build
idSnapMapAction_StringVar_Set
idSnapMapAction_TeamResource_Operation
idSnapMapAction_TeamResource_Spend
idSnapMapAction_Teleport
idSnapMapAction_Teleporter_SetEnable
idSnapMapAction_ToggleShowHide
idSnapMapAction_UpdateSpeaker
idSnapMapAction_Volume_EnableDisableTouch
idSnapMapCapEntity
idSnapMapCustomEvent
idSnapMapEntity_IntCompare
idSnapMapEntity_NumCompare
idSnapMapEntity_ResourceCompare
idSnapMapEntity_StringCompare
idSnapMapGameEntity_AI
idSnapMapGameEntity_AIDirector
idSnapMapGameEntity_BouncePad
idSnapMapGameEntity_BouncePadDestination
idSnapMapGameEntity_Callout
idSnapMapGameEntity_Camera
idSnapMapGameEntity_CodexMessage
idSnapMapGameEntity_ComboStart
idSnapMapGameEntity_ControlPoint

idSnapMapGameEntity_Encounter
idSnapMapGameEntity_EncounterList
idSnapMapGameEntity_EndGame
idSnapMapGameEntity_GameplaySettings
idSnapMapGameEntity_HUDSettings
idSnapMapGameEntity_ItemSpawnSettings
idSnapMapGameEntity_Level
idSnapMapGameEntity_Light
idSnapMapGameEntity_Message
idSnapMapGameEntity_Module
idSnapMapGameEntity_Music
idSnapMapGameEntity_NextMapSettings
idSnapMapGameEntity_Objective
idSnapMapGameEntity_ObjectiveSP
idSnapMapGameEntity_ObjectiveSub
idSnapMapGameEntity_Player
idSnapMapGameEntity_Player_Loadout
idSnapMapGameEntity_PlayerCamera
idSnapMapGameEntity_PlayerInput
idSnapMapGameEntity_ScoreSettings
idSnapMapGameEntity_SettingsBase
idSnapMapGameEntity_SnapPoiSettings
idSnapMapGameEntity_Spawner
idSnapMapGameEntity_Spawner_Ammo
idSnapMapGameEntity_Spawner_Backpack
idSnapMapGameEntity_Spawner_Droppable
idSnapMapGameEntity_Spawner_Fx
idSnapMapGameEntity_Speaker
idSnapMapGameEntity_Team
idSnapMapGameEntity_TeleporterDestination
idSnapMapGameEntity_TeleporterPad
idSnapMapGameEntity_TransmissionMessage
idSnapMapGameEntity_WorldText
idSnapMapGameEntityIterator
idSnapMapGameEntityIterator_AI
idSnapMapGameEntityIterator_Player
idSnapMapGameEntityIterator_Team
idSnapMapLensFlare
idSnapMapListener
idSnapMapListener_Activator

idSnapMapListener_AIKilled
idSnapMapListener_CapturePointCaptured
idSnapMapListener_DroppableActivator
idSnapMapListener_IntCompare
idSnapMapListener_IntensityChanged
idSnapMapListener_NumCompare
idSnapMapListener_OnEncounterPercentageReached
idSnapMapListener_OnHealthPercentageReached
idSnapMapListener_OnSnapScoreReached
idSnapMapListener_PlayerEnteredModule
idSnapMapListener_PlayerIncapacitated
idSnapMapListener_PlayerInput
idSnapMapListener-SequencerFired
idSnapMapListener_Simple
idSnapMapListener_SnapAnimEvent
idSnapMapListener_StringCompare
idSnapMapListener_TeamScoreReached
idSnapMapLogic
idSnapMapLogic_Count
idSnapMapLogic_CountDown
idSnapMapLogic_Delay
idSnapMapLogic_Gate
idSnapMapLogic_RandomRelay
idSnapMapLogic_Relay
idSnapMapLogic_Repeater
idSnapMapLogic-Sequencer
idSnapMapLogic_Switch
idSnapMapParticleEmitter
idSnapMapPortal
idSnapMapStaticWaterEntity
idSnapMapUserFilter
idSnapMapVariable
idSoundDoor
idSoundEnvironment
idSoundTrigger
idSpawnArea
idSpawnNode
idSpawnNodeSplines
idSpawnPoint
idSpeaker

idSpeaker_Beam
idSpectatorCamera
idSpitfireCannon
idSplinePath
idSpringCamera
idStaticEntity
idStaticMultiGuiEntity
idStaticVisibilityBlocker
idStaticWaterEntity
idSummoningTemplate
idSyncEntity
idTarget_ActionScript
idTarget_ActivateOutline
idTarget_AdaptiveTickToggle
idTarget_AIEvent
idTarget_AmplitudeTrigger
idTarget_AnimWebChangeState
idTarget_AnimWebChangeStateVia
idTarget_AnimWebPause
idTarget_AnimWebUnpause
idTarget_ApplyExplosionImpulse
idTarget_ApplyImpulse
idTarget_Award_Adrenaline
idTarget_Award_RushAttack
idTarget_BinaryModelController
idTarget_Break
idTarget_ChangeColor
idTarget_ChangeMaterial
idTarget_ChangeVolume_PlayerEnvOverride
idTarget_Cinematic
idTarget_CinematicHud
idTarget_ClearFakeEnemy
idTarget_Codex
idTarget_CollisionDamage
idTarget_Command
idTarget_CompassPOI
idTarget_Conditional
idTarget_ConditionalAccessor
idTarget_Count
idTarget_Count_Random

idTarget_Credits
idTarget_Cvar
idTarget_Damage
idTarget_DeactivateStatusEffects
idTarget_DisableEscMenu
idTarget_Disconnect
idTarget_DummyFire
idTarget_DynamicChallenge_End
idTarget_DynamicChallenge_FailChallenge
idTarget_DynamicChallenge_GiveBonus
idTarget_DynamicChallenge_PointTrigger
idTarget_DynamicChallenge_Start
idTarget_EnableAIEvent
idTarget_EnableGroup
idTarget_EnableTarget
idTarget_EndOfCampaign
idTarget_EndOfClassicDoomLevel
idTarget_EndOfLevel
idTarget_Enemy
idTarget_EquipItem
idTarget_FakeEnemy
idTarget_FireWeapon
idTarget_FirstThinkActivate
idTarget_ForceDormancy
idTarget_ForceGroupRole
idTarget_GameChallengeGameOver
idTarget_GameChallengeSetState
idTarget_GameStateIntInc
idTarget_GameStateIntSet
idTarget_GiveCollectible
idTarget_GiveItems
idTarget_GiveWeaponUpgradePoints
idTarget_GroupExpression
idTarget_GroupMessage
idTarget_GUICommand
idTarget_Hide
idTarget_InteractionAction
idTarget_Intro
idTarget_InventoryCheck
idTarget_LayerStateChange

idTarget_LevelTransition
idTarget_LightController
idTarget_MakeActivatable
idTarget_ModifyGroup
idTarget_ModifyTraversalClass
idTarget_Objective_Complete
idTarget_Objective_Give
idTarget_Objective_HideEntities
idTarget_Objective_Replace
idTarget_Objective_Triggered
idTarget_Path
idTarget_Perk
idTarget_Ping
idTarget_PlayerCheckpoint
idTarget_PlayerInvulnerability
idTarget_PlayerModifier
idTarget_PlayerStatModifier
idTarget_PlayerStatus
idTarget_PlayerViewEffect
idTarget_PlayerWhiplash
idTarget_PlayMovieClip
idTarget_PlayVoiceOver
idTarget_Print
idTarget_RandomImpulse
idTarget_Remove
idTarget_RemoveItems
idTarget_Secret
idTarget_SecurityUnlockChangeState
idTarget_SetGroupCombatStage
idTarget_SetInEncounterGroup
idTarget_SetSpawnSpot
idTarget_ShakeTrigger
idTarget_Show
idTarget_ShowGui
idTarget_Snap_Objective
idTarget_SoundDuck
idTarget_Spawn
idTarget_Spawn_Coop
idTarget_Spawn_Parent
idTarget_Spawn_Target

idTarget_StartSoundShader
idTarget_Subtitle
idTarget_SwapFaction
idTarget_Teleport
idTarget_Test
idTarget_TestActivation
idTarget_TestCommand
idTarget_TestEntity
idTarget_TestEntityBool
idTarget_TestExit
idTarget_TestInstruction
idTarget_TestOverrideStatus
idTarget_TestPlayerState
idTarget_TestScreenCompare
idTarget_TestScreenShot
idTarget_TestShotsFired
idTarget_Timeline
idTarget_Timer
idTarget_TutorialGui
idTarget_TutorialPopupLoadout
idTarget_UnlockArcadeMode
idTarget_UnlockUltraNightmare
idTarget_ViewPos
idTarget_VoiceCommunication
idTargetSpawnGroup
idTeamCapturePoint
idTeleporterPad
idTest_AimPoint
idTest_ConeConstraint
idTest_MetaData_Bounds
idTest_MetaData_Circle
idTest_MetaData_Cylinder
idTest_MetaData_Cylinder2
idTest_MetaData_Fov
idTest_MetaData_FovPitch
idTest_MetaData_FovYaw
idTest_MetaData_Line
idTest_MetaData_Model
idTest_MetaData_MoveableSpheres
idTest_MetaData_Plane

idTest_MetaData_SectorPitch
idTest_MetaData_SectorYaw
idTest_MetaData_String
idTest_MetaData_Target
idTest_MetaData_WebPath
idTest_RandomAimPoint
idTest_RotationTracker
idTest_Sector
idTestEntity
idTestModel2
idTrailerCamera
idTrigger
idTrigger_AIInteraction
idTrigger_BouncePad
idTrigger_CoopSafeZone
idTrigger_Damage
idTrigger_DummyFire
idTrigger_DynamicHurt
idTrigger_EnergyField
idTrigger_Facing
idTrigger_ForcePlayerBodyReactionOnce
idTrigger_Hurt
idTrigger_Multiplayer
idTrigger_PlayerHud_PlayMovieClip
idTrigger_Push
idTrigger_RemoveEntities
idTrigger_RemoveInventoryItems
idTrigger_Repulsor
idTrigger_TakeDamage
idTrigger_Teleporter
idTrigger_TestPlayerState
idTrigger_VoiceCommunication
idTriggerToucher
idTumbleWeed
idTurret
idUmbraVolume
idUtilityCamera
idVectorFieldEntity
idVolume
idVolume_AREntity

idVolume_Blocking
idVolume_ControlPointSpawnInfluence
idVolume_CustomLedgeGrab
idVolume_Damage
idVolume_DamageOverTime
idVolume_DemonicBait
idVolume_DemonicPossession
idVolume_DeployedLaser
idVolume_Domination
idVolume_EmpField
idVolume_EnvironmentHazard
idVolume_Flight
idVolume_ForceDormant
idVolume_Gravity
idVolume_HandsEnvironmentalReaction
idVolume_Inhibit_AIDirector_Spawning
idVolume_InvalidDemonDropZone
idVolume_ItemGroup
idVolume_KillDownedInjured
idVolume_LocationCalling
idVolume_MancubusSteam
idVolume_MancubusSteamFX
idVolume_MatterBallCore
idVolume_MatterBallWarp
idVolume_OliviasGuardFight_DangerZone
idVolume_PauseArcadeModeFuse
idVolume_PlasmaPuddle
idVolume_PlasmaPuddleFX
idVolume_PlayerBodyReaction
idVolume_PlayerEnvOverride
idVolume_PlayerUseProxy
idVolume_PlayerViewEffect
idVolume_RevivePlayer
idVolume_RunCycleHint
idVolume_RuneHint
idVolume_Safehouse_Sensor
idVolume_Safehouse_Shield
idVolume_SecretHint
idVolume_Sector
idVolume_Siphon

idVolume_SnapLockdownDoor
idVolume_SoulDrop
idVolume_SpaceWarp
idVolume_Stairs
idVolume_StealthField
idVolume_ToggleableDamageOverTime
idVolume_Trigger
idVolume_Trigger_Editable
idVolume_Trigger_Editable_Damageable
idVolume_VirtualGUI
idWaterEntity
idWaterRippleEmitter
idWaveShield
idWeaponEntity
idWorldClipBounds
idWorldDestructibleEntity
idWorldspawn

idTarget_Command

An entity to trigger the specified console commands.

Usage

```
entity {
    entityDef example_target_command_2016 {
        class = "idTarget_Command";
        expandInheritance = false;
        poolCount = 0;
        poolGranularity = 2;
        edit = {
            'idEntity::entityFlags_t' flags = {
                'bool' noFlood = true;
            }
            commandText = "give ammo";
            spawnPosition = {
                x = 0;
                y = 0;
                z = 0;
            }
        }
    }
}
```

idTarget_Remove

An entity that will remove the listed entities when activated.

Usage

```
entity {
  entityDef example_target_remove_2016 {
    inherit = "target/remove";
    class = "idTarget_Remove";
    expandInheritance = false;
    poolCount = 0;
    poolGranularity = 2;
    edit = {
      idEntity::entityFlags_t flags = {
        bool noFlood = true;
      }
      spawnPosition = {
        x = 50;
        y = -2000;
        z = -100;
      }
      targets = { // list of entities to remove
        num = 1;
        item[0] = "example_func_emitter_1";
      }
    }
  }
}
```

idTarget_Hide

An entity that will hide the listed entities when activated.

Note that not all entities can be hidden.

Usage

```
entity {
  entityDef example_target_hide_2016 {
    inherit = "target/hide";
    class = "idTarget_Hide";
    expandInheritance = false;
    poolCount = 0;
    poolGranularity = 2;
    edit = {
      idEntity::entityFlags_t flags = {
        bool noFlood = true;
      }
    }
    spawnPosition = {
      x = 666;
      y = 1000;
      z = 24;
    }
    targets = { // list of entities to hide
      num = 1;
      item[0] = "example_emitter";
    }
    reusable = true; // if false, this target will be single use
  }
}
```

See Also

- [idTarget_Show](#)

idTarget_Show

An entity that will show the listed entities when activated.

Usage

```
entity {
  entityDef example_target_show_2016 {
    inherit = "target/show";
    class = "idTarget_Show";
    expandInheritance = false;
    poolCount = 0;
    poolGranularity = 2;
    edit = {
      idEntity::entityFlags_t flags = {
        bool noFlood = true;
      }
      spawnPosition = {
        x = 1848;
        y = -200;
        z = 36;
      }
      targets = {
        num = 1;
        item[0] = "entity_name";
      }
      reusable = true; // if false, this target will be single use
    }
  }
}
```

See Also

- [idTarget_Hide](#)

idTarget_Spawn

An entity to spawn other entities. Primarily used for idAI2 entities.

Usage

```
entity {
  entityDef example_target_spawn_2016 {
    inherit = "target/spawn";
    class = "idTarget_Spawn";
    expandInheritance = false;
    poolCount = 0;
    poolGranularity = 2;
    edit = {
      idEntity::entityFlags_t flags = {
        bool noFlood = true;
      }
      spawnEditable = {
        overwriteTraversalFlags = true; // if set to true, use traversalClassFlags to overwrite the
        ai's flags
        traversalClassFlags = "CLASS_A";
        spawnAt = ""; // idEntity; the entity (demon) spawned by this spawn target will spawn at the
        listed entity.
        copyTargets = true; // // if true the targeted entity's target list will be copied to the
        spawned entity
        additionalTargets = {
          hum = 0;
          item[0] = ""; // idEntity
        }
        spawnAnim = ""; // aiStateOverride must be set to AIOVERRIDE_PLAY_ENTRANCE_ANIMATION in order
        to use spawnAnim
        aiStateOverride = "AIOVERRIDE_TELEPORT"; // aiStateOverride_t
      }
      spawnConditions = { // Inherited from idTarget_Spawn_Parent's idSpawnConditions
      }
      maxCount = 0; // maximum number of entities (demons) that this spawn target will spawn
      reuseDelaySec = 0; // delay in seconds, between uses
      doBoundsTest = false;
```

```

    boundsTestType = "BOUNDSTEST_NONE"; // boundsTest_t
    fovCheck = 0; // Whether the spawn target is (not) in the player's FOV, in degrees. Use
    positive values for within, and negative values for outside of
    minDistance = 0; // The player cannot be closer than the specified distance to the spawn
    target.
    maxDistance = 0; // The player cannot be further than the specified distance to the spawn
    target.
    neighborSpawnerDistance = -1;
    LOS_Test = "LOS_NONE"; // targetSpawn_LOSTest; to be combined with an FOV test. LOS_NONE no
    test, LOS_TRUE must be in LOS, LOS_FALSE must not be in LOS
    playerToTest = "PLAYER_SP"; // which player to test for this spawn target; leave as is for
    the primary player
    conditionProxy = ""; // idInfo_SpawnConditionProxy; entity at location to do LOS, FOV, and
    distance checks
}
spawnPosition = {
    x = 0;
    y = 0;
    z = 0;
}
spawnOrientation = { // 0 0 0
    mat = {
        mat[0] = {
            x = 1;
            y = 0;
            z = -0;
        }
        mat[1] = {
            x = 0;
            y = 1;
            z = 0;
        }
        mat[2] = {
            x = 0;
            y = 0;
            z = 1;
        }
    }
}
entityDefs = { // entity list to be spawned by the spawn target

```



```
num = 1;
item[0] = {
    name = "mod_ai_imp_1";
}
targetSpawnParent = "mod_master_spawn_parent";
}
```

idTarget_Teleport

An entity to teleport the player to the specified location when used.

Usage

```
entity {
    entityDef example_target_teleport_2016 {
        inherit = "target/teleport";
        class = "idTarget_Teleport";
        expandInheritance = false;
        poolCount = 0;
        poolGranularity = 2;
        edit = {
            spawnPosition = { // destination for teleport
                x = -1911.95;
                y = 1800;
                z = -190;
            }
            spawnOrientation = {
                mat = {
                    mat[0] = {
                        x = 0;
                        y = -1;
                        z = -0;
                    }
                    mat[1] = {
                        x = 1;
                        y = 0;
                        z = 0;
                    }
                    mat[2] = {
                        x = 0;
                        y = -0;
                        z = 1;
                    }
                }
            }
        }
    }
}
```



```
targetActivator = true; // if true, will teleport the activator
```



```
}
```

```
}
```

idTrigger

An entity that will trigger its listed targets when used.

Usage

```
entity {
  entityDef example_trigger_2016 {
    inherit = "triggers/trigger";
    class = "idTrigger";
    expandInheritance = false;
    poolCount = 0;
    poolGranularity = 2;
    edit = {
      saveTriggerState = true; // Whether to save the trigger state upon checkpoint (?)
      spawnPosition = {
        x = 1110;
        y = 12448;
        z = 164;
      }
      wait = 0; // Seconds to wait before this idTrigger can be retriggered.
      waitDelta = 0; // Additional seconds added to the wait value with each trigger. This is
        cumulative.
      waitMin = 0; // Minimum seconds the wait period can become if using waitDelta.
      waitMax = 0; // Maximum seconds the wait period can become if using waitDelta.
      delay = 0; // Delay, in seconds, between activating this idTrigger and when it activates its
        targets.
      triggerOnce = true; // If true, will only trigger its targets once.
      triggerFirst = true; // If true, this entity must be activate before it can be used.
      triggerOnEnter = true; // If true, it will only activate its targets when entering the
        trigger.
      triggerOnExit = false; // If true, it will only activate its targets when exiting the
        trigger.
      aiCanActivate = false; // If true, ai can activate this idTrigger.
      targets = { // list of entities to be triggered
        num = 1;
        item[0] = "example_relay_entity";
```

```
    }  
    renderModelInfo = {  
        model = NULL;  
    }  
    clipModelInfo = {  
        clipModelName = "";  
        type = "CLIPMODEL_CYLINDER";  
        size = {  
            x = 100;  
            y = 100;  
            z = 100;  
        }  
    }  
}  
}
```

The spawnPosition and clipModelInfo determine the physical location that the player has to walk into to activate the idTrigger. Alternatively, the idTrigger can also be activated by another entity or eventCall.

idResurrectionProxy

An entity to serve as a summoning location for Summoners.

Usage

```
entity {
  entityDef example_ai_rez_proxy_1 {
    inherit = "ai/rez_proxy/default";
    class = "idResurrectionProxy";
    expandInheritance = false;
    poolCount = 0;
    poolGranularity = 2;
    edit = {
      renderModelInfo = {
        scale = {
          x = 3;
          y = 3;
          z = 3;
        }
      }
    }
    availableSummons = { // all entitydefs this thing can spawn; this list should remain
      unchanged
      num = 14;
      item[ 0] = {
        characterDef = "ai/zombie/hell_base";
        portalDef = "ai/misc/generic_portal";
        fxDecl = "fx/creatures/rez/archvile_spawn";
      }
      item[ 1] = {
        characterDef = "ai/demon/lostsoul";
        portalDef = "ai/misc/generic_portal";
        fxDecl = "fx/creatures/rez/archvile_spawn";
      }
      item[ 2] = {
        characterDef = "ai/demon/imp";
        portalDef = "ai/misc/generic_portal";
```

```
    fxDecl = "fx/creatures/rez/archvile_spawn";
}

item[3] = {
    characterDef = "ai/demon/imp/imp_albino";
    portalDef = "ai/misc/generic_portal";
    fxDecl = "fx/creatures/rez/archvile_spawn";
}

item[4] = {
    characterDef = "ai/hellified/marine_rifle";
    portalDef = "ai/misc/generic_portal";
    fxDecl = "fx/creatures/rez/archvile_spawn";
}

item[5] = {
    characterDef = "ai/hellified/hell_base";
    portalDef = "ai/misc/generic_portal";
    fxDecl = "fx/creatures/rez/archvile_spawn";
}

item[6] = {
    characterDef = "ai/demon/revenant";
    portalDef = "ai/misc/generic_portal";
    fxDecl = "fx/creatures/rez/archvile_spawn";
}

item[7] = {
    characterDef = "ai/demon/cacodemon";
    portalDef = "ai/misc/generic_portal";
    fxDecl = "fx/creatures/rez/archvile_spawn";
}

item[8] = {
    characterDef = "ai/demon/pinky";
    portalDef = "ai/misc/generic_portal";
    fxDecl = "fx/creatures/rez/archvile_spawn";
}

item[9] = {
    characterDef = "ai/demon/pinky_spectre";
    portalDef = "ai/misc/generic_portal";
    fxDecl = "fx/creatures/rez/archvile_spawn";
}

item[10] = {
    characterDef = "ai/demon/hellknight";
    portalDef = "ai/misc/generic_portal";
```

```

    fxDecl = "fx/creatures/rez/archvile_spawn";
}
item[11] = {
    characterDef = "ai/demon/mancubus";
    portalDef = "ai/misc/generic_portal";
    fxDecl = "fx/creatures/rez/archvile_spawn";
}
item[12] = {
    characterDef = "ai/demon/mancubus_cyber";
    portalDef = "ai/misc/generic_portal";
    fxDecl = "fx/creatures/rez/archvile_spawn";
}
item[13] = {
    characterDef = "ai/demon/baron";
    portalDef = "ai/misc/generic_portal";
    fxDecl = "fx/creatures/rez/archvile_spawn";
}
}

radius = 242; // spawns are valid within this radius of the rez proxy
fxBaseEntity = "func_fx"; // base fx decl to use for summons
spawnTimeDecl = "base"; // decl describing spawn times for each ai type
spawnPosition = { // location of the rez proxy/summoning rift
    x = -877;
    y = 1823;
    z = -383.9997253418;
}
}
}

```

The location of the rez proxy is the physical location where the Summoner goes to place a summoning rift and summon.

idSummoningTemplate

An entity to control summoning behaviour of a Summoner.

Usage

```
entity {
  entityDef example_summoning_template {
    class = "idSummoningTemplate";
    expandInheritance = false;
    poolCount = 0;
    poolGranularity = 2;
    edit = {
      fixedSummoningWaves = { // List of tiered summoning waves, executed sequentially.
        num = 1;
        item[0] = {
          tier1_SpawnList = { // An Imp or Possessed Soldier can be summoned for this tier.
            num = 2;
            item[0] = "ENCOUNTER_SPAWN_IMP";
            item[1] = "ENCOUNTER_SPAWN_HELL_SOLDIER";
          }
          tier2_SpawnList = {
            num = 2;
            item[0] = "ENCOUNTER_SPAWN_IMP";
            item[1] = "ENCOUNTER_SPAWN_ZOMBIE";
          }
          tier3_SpawnList = {
            num = 2;
            item[0] = "ENCOUNTER_SPAWN_SHOTGUN_SOLDIER";
            item[1] = "ENCOUNTER_SPAWN_BEAM_SOLDIER";
          }
          completionTest = "SUMMON_CONDITIONS_ALL_DEAD"; // SUMMON_CONDITIONS_ALL_DEAD,
          SUMMON_CONDITIONS_FIXED_TIMER, or SUMMON_CONDITIONS_WEIGHTED_TIMER
        }
      }
      randomSummoningWaves = { // List of tiered summoning waves, executed in order but looped
        until the summoning encounter is complete.
      }
    }
  }
}
```

```

    num = 3;
    item[ 0] = {
        tier1_SpawnList = {
            num = 2;
            item[ 0] = "ENCOUNTER_SPAWN_IMP";
            item[ 1] = "ENCOUNTER_SPAWN_ZOMBIE";
        }
        tier2_SpawnList = {
            num = 1;
            item[ 0] = "ENCOUNTER_SPAWN_REVENANT";
        }
        completionTest = "SUMMON_CONDITIONS_ALL_DEAD";
    }
    item[ 1] = {
        tier1_SpawnList = {
            num = 2;
            item[ 0] = "ENCOUNTER_SPAWN_ZOMBIE";
            item[ 1] = "ENCOUNTER_SPAWN_ZOMBIE";
        }
        tier2_SpawnList = {
            num = 1;
            item[ 0] = "ENCOUNTER_SPAWN_PINKY";
        }
        completionTest = "SUMMON_CONDITIONS_ALL_DEAD";
    }
    item[ 2] = {
        tier1_SpawnList = {
            num = 3;
            item[ 0] = "ENCOUNTER_SPAWN_IMP";
            item[ 1] = "ENCOUNTER_SPAWN_ZOMBIE";
            item[ 2] = "ENCOUNTER_SPAWN_IMP";
        }
        tier2_SpawnList = {
            num = 2;
            item[ 0] = "ENCOUNTER_SPAWN_PINKY";
            item[ 1] = "ENCOUNTER_SPAWN_REVENANT";
        }
        completionTest = "SUMMON_CONDITIONS_ALL_DEAD";
    }
}

```

```

fixedSummonGroups = { // List of groups that is sequentially run
    item[ 0] = {
        aiSpawnTypeList = {
            item[ 0] = "ENCOUNTER_SPAWN_BEAM_SOLDIER";
            item[ 1] = "ENCOUNTER_SPAWN_BEAM_SOLDIER";
            item[ 2] = "ENCOUNTER_SPAWN_IMP";
        }
    }
    item[ 1] = {
        aiSpawnTypeList = {
            item[ 0] = "ENCOUNTER_SPAWN_BEAM_SOLDIER";
            item[ 1] = "ENCOUNTER_SPAWN_IMP";
            item[ 2] = "ENCOUNTER_SPAWN_HELL_KNIGHT";
        }
    }
}

randomSummonPool = { // List of group to randomly run after all fixed encounters are complete
    item[ 0] = {
        aiSpawnTypeList = {
            item[ 0] = "ENCOUNTER_SPAWN_IMP";
            item[ 1] = "ENCOUNTER_SPAWN_BEAM_SOLDIER";
            item[ 2] = "ENCOUNTER_SPAWN_BEAM_SOLDIER";
        }
    }
    item[ 1] = {
        aiSpawnTypeList = {
            item[ 0] = "ENCOUNTER_SPAWN_IMP";
            item[ 1] = "ENCOUNTER_SPAWN_BEAM_SOLDIER";
            item[ 2] = "ENCOUNTER_SPAWN_HELL_KNIGHT";
        }
    }
}

spawnPosition = {
    x = 8170;
    y = 6877;
    z = 8000;
}
}

```

- `fixedSummoningWaves` : The summoner summons up to three demons, each one drawn from the given tier list. Once `completionTest` is fulfilled, the rift will summon the next wave. These waves are executed in the listed order, ie. `item[0]` -> `item[1]` -> `item[2]` etc. Ends once all waves are complete.
- `randomSummoningWaves` : Similar to `fixedSummoningWaves` , but will loop upon completion, instead of stopping.
- `fixedSummonGroups` : List of groups that is sequentially run (?); the function of this summoning behaviour is not fully understood at this time.
- `randomSummonPool` : Once the preceding `fixedSummoningWaves` or `fixedSummonGroups` is complete, the summoning rift will randomly cycle through the defined `aiSpawnTypeLists` and summon.
- `completionTest` : All `idSummoningTemplates` in the vanilla game use `SUMMON_CONDITIONS_ALL_DEAD` . While `SUMMON_CONDITIONS_FIXED_TIMER` , or `SUMMON_CONDITIONS_WEIGHTED_TIMER` are also options, it is currently not understood how to use those conditions.

Note that once the summoning rift is created, these behaviours will execute on their own, as long as the Summoner remains alive. If killed, the rift will close, and anything defined by the `idSummoningTemplate` will stop immediately.

idVolume_PlayerEnvOverride

An entity to overwrite the envSettings decl set by idWorldspawn when within the defined volume.

Usage

```
entity {
    entityDef example_volume_player_env_override {
        inherit = "volume/player_env_override";
        class = "idVolume_PlayerEnvOverride";
        expandInheritance = false;
        poolCount = 0;
        poolGranularity = 2;
        edit = {
            spawnPosition = {
                x = -1947;
                y = 787;
                z = -500;
            }
            renderModelInfo = {
                model = NULL;
            }
            clipModelInfo = {
                clipModelName = "";
                type = "CLIPMODEL_CYLINDER";
                size = {
                    x = 30000;
                    y = 30000;
                    z = 10000;
                }
            }
        }
        fadeInTime = 0; // time in milliseconds to fade in
        fadeOutTime = 0; // time in milliseconds to fade out
        startEnabled = true; // if set to false, entity must be activated first before use
        allowToggleOnActivate = true; // if true, activating the entity will toggle it on/off
        declEnv = "vista/bfg_division_ext"; // the env decl to use
    }
}
```

