

Entity NBT

About how entity NBTs and the different types there are.

- [NBT](#)
- [Entity NBT Format](#)
- [NBT definitions](#)

NBT













What is NBT

NBT or **N**amed **B**inary **T**ag, is a tree-like structure that Minecraft uses to store entity data. This data can include `attributes`, `definitions`, and `identifier` information. This information can control things what the entity looks like, how it interacts with the world, and its behavior. Knowing how NBT works can give you insight on certain things you may encounter or want to achieve. NBT however is not the entity itself but instead a snapshot of its runtime components that we can load to make a copy of the entity. In bedrock all NBT is encoded in [little-endian](#) which is important if you want to extract it into a stringified format like the one below.

Entity NBT Format

NBT FORMAT (WIP)

Legend

Icon	Meaning
	String
	Short
	Long Array
	Long
	JSON Object
	Integer Array
	Integer
	Float
	Double
	Byte Array
	Byte
	Boolean
	Array

Format Tree

root label



Where the NBT tree starts

identifier



Defines entity type ID.

Example: "**minecraft:pig**".

definitions



Adds or removes components from the from the baseline entity components. These are entity mostly specific. + adds a component not on the base entity and - removes a component on the base entity.

Example: "**+minecraft:can_ride_default**"

UniqueID



Unique numerical ID of the entity used to identify it from others of it type.

Example: "**-4294967289**".

Pos



Shows numerical values as to the position of the entity in 3D space.

X



Shows the X position of the entity.

Y



Shows the Y position of the entity.

Z



Shows the Z position of the entity.

Rotation



Defines the orientation angle of the entity.

Yaw



The orientation of the entity rotating clockwise around the X axis. (No more than 360°)

Pitch



The orientation of the entity rotating up and down along the Y axis. (No more than $\pm 90^\circ$)

Motion



Defines the momentum assigned to a given entity.

dX



Defines momentum along the X axis.

dY



Defines momentum along the Y axis.

dZ



Defines momentum along the Z axis.

CustomNameVisible



Shows a custom name given to an entity regardless if you are looking directly at them or not.

LastDimensionId



Last dimension visited by the entity. (Seemingly deprecated)

FallDistance



Distance the entity has fallen.

Fire



Number of ticks till the fire on the entity is extinguished.

OnGround



Defines if an entity is in the state of being on the ground. (Not falling)

Invulnerable



Makes the target entity unable to be harmed by any natural source. Entities with this tag cannot be moved by fishing rods. Creative players can deal damage regardless of the tag being applied to the target entity.

PortalCooldown



Number of ticks before a given entity can use a nether portal again.

IsGlobal



Defines if the entity can function in [unloaded chunks](#).

IsAutonomous



Exact use is unknown.

LinksTag



Defines a connection between a ride and a rider.

entityID



Unique numerical ID of the rider entity the ride entity has linked to.

Example: **"-3410204032983"**.

linkID



Numerical value used to determine which rider is which on the ride entity. (Starts at 0 and increments by 1 for each passenger)

LootDropped



Defines if an entity can or cannot drop loot. This is applied after the entity has received the Dead tag to ensure it doesn't drop loot more than once.

Color



Defines an entities color type. For example the difference between a black cat and a siamese cat is this value.

Color2



Defines an entities secondary color. Only tropical fish use this.

Strength



Defines how much a llama can carry. ($3 * \text{number} = \text{Items carriable}$)

StrengthMax



Does the same thing as "`Strength`" seemingly.

Sheared



Defines whether a sheep is sheared.

IsIllagerCaptain



Defines whether an entity is an Illager Captain.

OwnerNew



Shows new owner ID. This is often for projectiles.

Example: "-4294967295"

OwnerID



Shows current owner ID. This is often for projectiles.

Example: "-4294967295"

Sitting



Defines whether an entity is sitting.

IsBaby



Defines whether an entity is in its baby variant.

IsTamed



Defines whether an entity is tame.

IsTrusting



Defines whether an entity is trusting of the player.

IsOrphaned



Defines whether an entity is spawned without parents.

IsAngry



Defines whether an entity is hostile when it is normally neutral.

IsOutOfControl



Is set to true when a player is controlling it and it is a ship.

Variant



Defines entity variant type.

MarkVariant



Defines the mark variant of an entity.

Saddled



Is set to true when a player has mounted it and it is a living entity.

IsChested



Defines if a working animal such as a horse carries a chest on its back.

ShowBottom



Defines whether an end crystal shows the bedrock slab below it. (Used in the end dimension's crystals)

IsGliding



Defines whether an entity is in the gliding state with an elytra.

IsSwimming



Defines whether an entity is in the swimming state.

IsEating



Defines whether an entity is eating something.

IsScared



Defines whether an entity is in the scared state.

IsStunned



Defines whether ravager is stunned.

IsRoaring



Defines whether ravager is roaring.

SkinID



Defines a villager/zombie villagers skin texture.

Persistent



Defines if an entity can or cannot despawn when out of the player's mob spawning radius.

Tags



Holds player applied tags that an entity has. (these are applied via /tag)

tag (not a real namespace)



Tag identifier

Example: "CanopyAdmin"

CustomDisplayTile



Set to true when a minecart is displaying a custom block.

DisplayBlock



Set to true when a minecart is displaying a custom block.

block (not a real namespace)



Block ID of a block to be displayed within the minecart.

Example: "minecraft:dirt"

DisplayOffset



Number of pixels the display block of a minecart is offset along the Y axis.

Willing



Defines if a villager is willing to breed or not. This value is set to false for a period of time after breeding.

SpawnedByNight



Defines if a mob was spawned naturally by the night cycle or not.

ItemInHand



Defines what item an entity holding.

Count



Count of items in the hand.

Name



Defines the item type in the hand.

Example: "minecraft:crossbow"

WasPickedUp



Defines if the entity got the item by picking it up.

Damage



This may say damage but it does not store the damage for the item.

CanPlaceOn



Defines what the item be placed on when the user is in Adventure mode.

block (not real namespace)



Block ID it can be placed on.

Example: "minecraft:stone"

CanDestroy



Defines what the item can break when the user is in Adventure mode.

block (Not real namespace)



Block ID it can destroy.

Example: "minecraft:stone"

tag



Defines additional properties.

Mainhand



Defines what item is in the entities mainhand.

item (not a real namespace)



Item object.

Count



Count of items.

Name



Defines the item type.

Example: "minecraft:crossbow"

WasPickedUp



Defines if the entity got the item by picking it up.

Damage



This may say damage but it does not store the damage for the item.

CanPlaceOn



Defines what the item be placed on when the user is in Adventure mode.

block (not real namespace)



Block ID it can be placed on.

Example: "minecraft:stone"

CanDestroy



Defines what the item can break when the user is in Adventure mode.

block (Not real namespace)



Block ID it can destroy.

Example: "minecraft:stone"

tag



Defines additional properties.

Offhand



Defines what item is in the entities offhand.

item (not a real namespace)



Item object.

Count



Count of items.

Name



Defines the item type.

Example: "minecraft:crossbow"

WasPickedUp



Defines if the entity got the item by picking it up.

Damage



This may say damage but it does not store the damage for the item.

CanPlaceOn



Defines what the item be placed on when the user is in Adventure mode.

block (not real namespace)



Block ID it can be placed on.

Example: "minecraft:stone"

CanDestroy



Defines what the item can break when the user is in Adventure mode.

block (Not real namespace)



Block ID it can destroy.

Example: "minecraft:stone"

tag



Defines additional properties.

limitedLife



Sets a number of ticks before an entity is deleted.

TradeTier



Defines a villagers trading level.

TradeExperience



Defines a villagers experience progress to next level.

canPickupItems



Defines if an entity can pick up items.

hasSetCanPickupItems



Set to true if the canPickupItems component has been set by the game.

HurtTime



Duration an entity has the invulnerable tag set to true after taking damage. This also controls the duration in which the entity is red after taking damage.

DeathTime



Duration an entity executes its death animation to the player before deletion.

Dead



Defines if the entity has died.

AttackTime



Ticks before attack cooldown is complete.

Surface



True if the entity has spawned naturally on the surface. (IE, surface cap)

portal_transition



Internally used entity NBT for transitioning entities between dimensions. This is used because when an entity enters the portal it's reference is destroyed, so it needs some way to know the conditions in which it entered the portal.

axis



Saves the the axis in which the portal the entity entered is facing. (1 = z, 2 = x)

dimension



Defines which dimension the entity is coming from (0 = overworld, 1 = nether, 2 = end)

X



Shows the X position the entity entered a portal at.

Y



Shows the Y position the entity entered a portal at.

Z



Shows the Z position the entity entered a portal at.

NaturalSpawn



True if the entity has spawned naturally via cave or surface cap.

TargetID



Unique numerical ID of an entity that this entity is targeting for some purpose.

ActiveEffects



Lists the entities potion effects it has on it.

Effect (Not A Real Namespace)



Effect json object

Ambient



True if the effect was applied by a beacon. This is to make the particles less invasive on the screen.

Amplifier



The potion effect level. 0 is level 1.

DisplayOnScreenTextureAnimation



Displays a screen animation when an effect is applied. This is used in bad omen, totems, and the hero of the village to display the floating icon once you get the effect.

Duration



Number of ticks before the effect wears off.

DurationEasy



Number of ticks before the effect wears off for Easy mode.

DurationNormal



Number of ticks before the effect wears off for Normal mode.

DurationNormal



Number of ticks before the effect wears off for Hard mode.

Id



Numerical potion Id.

example: 28

ShowParticles



Defines if particles are shown or not.

FactorCalculationData



Unknown usage

change_timestamp

I

Unknown

factor_current**F**

Unknown

factor_previous**F**

Unknown

factor_start**F**

Unknown

factor_target**F**

Unknown

had_applied**B**

Unknown

had_last_tick



Unknown

padding_duration



Unknown

IsInRaid



Boolean only present in villagers that defines if the villager is currently in a raid.

ReactToBell



Defines if a villager will react to a bell ringing and go to its home or not.

TargetCaptainID



Unique numerical ID of the Illager captain of a patrol. Used by pillagers in patrols.

Example: **"-4294967289"**.

isCreative



Defines if a given projectile was fired from a player in creative mode. (Only projectiles have this)

enchantPower



Defines the power level given to an arrow by the power enchantment on the shooters bow.

enchantPunch



Defines the punch level given to an arrow by the power enchantment on the shooters bow.

enchantFlame



Defines the flame level given to an arrow by the power enchantment on the shooters bow.

enchantInfinity



Defines the infinity level given to an arrow by the power enchantment on the shooters bow.

MobEffects



List of effects that a given entity can pass onto others. This does not apply to itself. (Used in area effect clouds and tipped arrows)

Effect (Not A Real Namespace)



Effect json object

Ambient



True if the effect was applied by a beacon. This is to make the particles less invasive on the screen.

Amplifier



The potion effect level. 0 is level 1.

DisplayOnScreenTextureAnimation



Displays a screen animation when an effect is applied. This is used in bad omen, totems, and the hero of the village to display the floating icon once you get the effect.

Duration



Number of ticks before the effect wears off.

DurationEasy



Number of ticks before the effect wears off for Easy mode.

DurationNormal



Number of ticks before the effect wears off for Normal mode.

DurationNormal



Number of ticks before the effect wears off for Hard mode.

Id



Numerical potion Id.

example: 28

ShowParticles



Defines if particles are shown or not.

FactorCalculationData



Unknown usage

change_timestamp

I

Unknown

factor_current**F**

Unknown

factor_previous**F**

Unknown

factor_start**F**

Unknown

factor_target**F**

Unknown

had_applied**B**

Unknown

had_last_tick

B

Unknown

padding_duration

I

Unknown

auxValue

B

Defines a numerical ID for what arrow type it is, whether it is tipped and with what potion.

example: 23 (Healing II)

SelectedInventorySlot

I

Numerical ID of a slot selected in a players inventory.

SelectedContainerId

I

Numerical ID of a container selected by the player.

Sleeping



Defines if the player is sleeping or not.

SleepTimer



Defines for how long the player has been sleeping in a bed.

SpawnBlockPositionX



Defines the X location for the players respawn block like a bed or respawn anchor.

SpawnBlockPositionY



Defines the Y location for the players respawn block like a bed or respawn anchor.

SpawnBlockPositionZ



Defines the Z location for the players respawn block like a bed or respawn anchor.

SpawnDimension



Defines what dimension the player's spawn point is located in.

SpawnX



Defines the X location for the players respawn point.

SpawnY



Defines the Y location for the players respawn point.

SpawnZ



Defines the Z location for the players respawn point.

EnchantmentSeed



RNG seed for enchantment tables to deterministically give you the same enchantments.

RideID



Unique numerical ID of an entity that the player is riding.

LeftShoulderRiderID



Unique numerical ID of an entity that the player has on its left shoulder.

RightShoulderRiderID

L

Unique numerical ID of an entity that the player has on its right shoulder.

PlayerGameMode

I

Numerical ID of the gamemode the player is in currently.

PlayerLevel

I

How many experience levels the player has.

PlayerLevelProgress

I

How much experience progress a player has to the next level.

Not all NBT tags are used in a standard entity, this just includes all known NBT labels and their known purpose.

NBT definitions