



Realm of Aethel

Minecraft Command Block Technical

This document serves as a resource for previous command block technical aspects of Aethel's Minecraft mechanics and items. This document has been discontinued past S3.

Technical

Com. Blocks	Gameplay		Items	
Limbo	Souls	Death Penalty	Relics	Crystals
Tick Clock	Location Titles	Eternity	Artifacts	Attuned Rings
	Dark Mist	Dungeons	Spellbooks	Idols

Limbo

In terms of Aethel SMP, Limbo is the name for the command block center located underground within the world's spawn chunks. Since the world's spawn chunks are always loaded, Limbo's various command block components can provide the world with its special properties no matter where the players are.

Tick Clock

Tick Rate

For events to have a passage of time in Minecraft, the game runs at a rate of 20 ticks per second. At every tick, hundreds, if not thousands, of calculations are done to progress things such as entity movements, redstone updates, and random number generation for random block updates (such as tree growth and copper oxidation stages).

Command Blocks

In terms of command blocks, there exist three different types of command blocks, each with condition types and power requirement variants. Impulse types operate once, chain types operate based on the previous result of the command block behind it, and repeat types operate once every game tick, which effectively means 20 times a second.

Below is a table of how each command block setting variant interacts with the world.

	Left: Impulse type Middle: Chain type Right: Repeating type Top: Unconditional mode Bottom: Conditional mode
---	---

Type	Condition	State	Effect
------	-----------	-------	--------

Impulse	Unconditional	Needs Redstone	Activates once when powered
Impulse	Unconditional	Always Active	Activates immediately
Impulse	Conditional	Needs Redstone	
Impulse	Conditional	Always Active	
Chain	Unconditional	Needs Redstone	Activates regardless of previous success when powered
Chain	Unconditional	Always Active	Activates regardless of previous successful
Chain	Conditional	Needs Redstone	Activates only if previous was successful when powered
Chain	Conditional	Always Active	Activates only if previous was successful
Repeat	Unconditional	Needs Redstone	Activates 20 times a second when powered
Repeat	Unconditional	Always Active	Activates 20 times a second
Repeat	Conditional	Needs Redstone	
Repeat	Conditional	Always Active	

Note: Effects left blank are unused by myself, have either very specific use cases, or can be replaced by better combinations of alternatives.

Conditional Checks

To check for events such as players holding, wearing, or equipping certain items, we utilize command blocks to compare player information with conditional instances. In this case, we're identifying unique items when they're given specific ids through commands.

Accessible player data within the game is stored in an NBT tag format, and we can access the type of information we need by parsing that NBT tag tree. In layman's terms, there are many headers and categories of how player data is stored, and we go from the broadest categories to increasingly specific ones.

Balancing Checks with Performance

How do we know then when to check for these items? Ideally, it should be as soon as possible. So, going off from the information provided in the above chart, we should use repeating unconditional command blocks to check as fast as possible, right? Now that would be the best plan on paper, but the truth is that trying to check for multiple items at a time from every player or entity's data adds to the workload that the server must do every tick. We must find a good balance between response times for checking conditions as well as respecting server workload for the maximal performance output.

Therefore, we spread out the tasks across a period rather than all at once to get the effect that we want without sacrificing too much performance. To explain this in a practical example, imagine that you need to check 500 different conditions. You could check all the conditions for every tick, but this would be significantly more demanding than evenly spreading out the same number of conditions over 20 ticks in a second. By spreading out the checks, we still get the result we want – the 500 different total conditions being checked, but now the server has only had to check 25 conditions per tick throughout one second.

Timed Intervals

Command blocks in Java don't inherently have a built-in feature to delay themselves, so we must utilize a different feature called scoreboards to time these intervals. Using a singular repeating unconditional command block always active, we can add a score of 1 to an objective arbitrarily called the "Tick Clock" every tick. With a chain conditional command block always active that checks when the score of the Tick Clock is 20 to reset the score back to 0, we now effectively have a clock that runs for a full second before restarting itself. All we need to do now is attach now to this clock tasks to do every tick.

Setting Up a Tick Clock

Using a repeating unconditional command block always active that checks whether the score of the tick of the Tick Clock matches its designated partition, we can fill a section of redstone blocks at a time to mass activate condition checks with impulse unconditional command blocks that need redstone. A chain conditional always active attached to the redstone block filling repeat command block immediately triggers after it fills redstone blocks and replaces it with air blocks, triggering the checks only once every tick and not any longer.

Command Block Abbreviations

From henceforth in the document, command block types and their variants will be referred to by their Type, Condition, and State by their first letter initials to save space and ease of ability to read. For example, an impulse unconditional needs redstone command block will be abbreviated as IUNR. In future technical tables, Sequence / Parallels represent the order in which command blocks are being triggered, and contain multiple rows that show how many are concurrently being triggered on the same tick.

Souls

For the most part, souls are player-activated events that are tied to a "trigger" item players hold in their hand. Some souls activate under different conditions, and they are documented in a section further below.

Souls have the highest number of unique conditional checks, ranging in things such as checking where a player is facing, whether they are holding a unique ID item, where they are located, or whether they're in the proximity of another entity.

Abilities with Cooldowns

For an explanation of how most cooldown-based abilities work, an IUNR checks a specifically named player for their Selected Item tag, and then sends a redstone block to a CCAA[C] to handle the effects. A CUCAA->CCAA handles the cooldown countdown every interval as well as its cooldown notification. For this chain, typically the CUCAA checks whether the score for that respective soul is above 0, and it decreases it by 1 every interval. The CCAA connected to the CUCAA checks every time this CUCAA is successful if the score for the soul is 0, and privately /tells the owner that the ability is ready. This order prevents the chain from repeatedly spamming the owner of the ability that the ability is ready while the cooldown is at 0.

Soul Documentation

Due to the number of miracles & souls tiers that exist, only brief explanations for each one that interacts with command blocks will be provided. For more information about the specifics, you can ask me personally. For brevity, assume ability trigger mechanisms (holding a specific item with a special id) are not listed.

Miracle/Soul	Ability A	Ability B	Ability C
Aether	Store user rotation into score, if match rotation for facing downwards, grant Slow Falling to all entities within distance of user	Check firework rockets used score, if firework rockets used is 1 or more, reset score to 0 & give firework rocket	
Aging	Apply cooldown, aligned with user's eyes, positioned distance away, apply Nausea, Blindness, Slowness II, Weakness within radius to entities that are not the user & use all entities' position within distance of previous position to teleport self-entity to other entities randomly within a distance		
Almighty	Apply Health Boost II	Apply cooldown, grant Levitation, store fall distance into a score, if falling, check for when contact with ground, grant Resistance 255 & Strength II, summon an instant fuse	

Creeper at landing location		
Architecture	Apply cooldown, aligned with user's eyes, positioned distance away, replace air with stone	
Bee	Execute as user's feet, if flower, add points to scoreboard, and if score is positive & user is holding activation item, grant user Levitation II and Speed III, and remove score from scoreboard	
Berserk	Use existing health scoreboard to detect when below a certain threshold to grant Strength I & II	Use existing health scoreboard to detect when below a certain threshold to grant Resistance 255, apply cooldown
Bunny	Check jumps score, if jumps score is 3 or more, reset score to 0 & give Jump Boost III & Speed II	
Celerity	Apply cooldown, use cooldown score to distribute when Speed II area of effect clouds should spawn	
Constitution	If selected item is Health II potion, give Health Boost II potion	Use existing health scoreboard to detect when below a certain threshold to grant Absorption V
Darkness	Apply cooldown, summon bats positioned from user with soul tags, use cooldown score to	

	apply Blindness aura around soul tagged bats	
Death	Apply cooldown, aligned with user's eyes, positioned distance away, apply Wither within radius to limit of entities that are not the user	
Demon	Apply cooldown, grant levitation & instant damage, timed effects using cooldown, summon lightning & tnt on user, grant ramping Strength I to III, Speed I to III	
Destruction	Apply cooldown, aligned with user's eyes, positioned distance away, summon Creeper with edited NBT tags	Apply cooldown, aligned with user's eyes, positioned distance away, summon Creeper with edited NBT tags
Disease	Wait for user to be afflicted with Poison, remove Poison, grant Regeneration & Instant Health	
Druid	Aligned with user's eyes, positioned distance away, check if block at location is composter, give bone meal to user	
Duplication	Apply cooldown, aligned with user's eyes, positioned distance away, summon Slime positioned above any player within	

	distance of position that is not the user		
End	Check all entities if exists a target tagged with soul tag, if none, positioned at user, tag limit one entity, tagged enters soul team, applied Glowing, user can teleport to tagged entity, apply cooldown	Store user rotation into score, if match rotation for facing upwards, give special id snowball, check world for snowball, if detected, teleport tagged entity to snowball, apply cooldown	
Flesh	Use existing health scoreboard to detect when below a certain threshold to grant Regeneration III	Store result of rotten flesh existing in user's inventory into scoreboard, if score exceeds constant, clear constant of rotten flesh, give zoglin egg	Check if selected item is zoglin egg, aligned with user's eyes, positioned distance away, summon zoglin
Gravity	Grant Levitation V to entities within distance of user	Store user rotation into score, if match rotation for facing downwards, grant Slow Falling	Apply cooldown, aligned with user's eyes, positioned distance away, summon snowball with soul tag, use cooldown score as intervals of when to teleport entities within distance of snowball, remove snowball when cooldown score is at certain number
Holy	Apply cooldown, aligned with user's eyes, positioned distance away, apply Absorption & Instant Health to all players within distance		
Hydrodynamics	Check all entities for trident with specific id that has dealt damage or hit	Check all entities for trident with specific id that has dealt damage or hit	

	ground, run fill command with water replacing air positioned at trident, set mode to replace air with water, teleport trident to user	ground, run fill command with lava replacing air positioned at trident, set mode to replace air with lava, teleport trident to user
Magic	Apply cooldown, aligned with user's eyes, positioned distance away, summon fireballs	Apply cooldown, aligned with user's eyes, positioned distance away, tag limit one entity with soul tag, fill air around tagged entity with glass, use cooldown score to determine when to fill replace glass with air
Medusa	Aligned with user's eyes, positioned distance away, execute as entity in radius, aligned with entity's eyes, positioned distance away, if user in radius, grant self Slowness & Jump Boost 128	
Miner	Summon falling block with block data of pointed dripstone at locations around user, use cooldown to determine how many waves of dripstones to drop	
Motion	Check if distance sprinted within a period exceeds threshold, grant Speed VI & Strength II	
Nether	Fill replace fire & lava within radius with air if standing in fire or lava, add variable to	

	soul score based on fire or lava, use score to compare with constants, if score above threshold, apply Strength & Resistance, tick down soul score every interval	
Nuke	Apply cooldown, aligned with user's eyes, positioned distance away, summon fireballs with motion downwards	
Oracle		Apply cooldown, aligned with user's eyes, positioned distance away, apply Glowing to all entities within distance
Plague	Apply Poison to all entities within distance	
Pufferfish	Add to soul score if selected trigger mechanism, if score is 1, apply Poison II to all entities within distance, if score is 2, set score to 0	
Ram	Use existing health scoreboard to detect when below a certain threshold to grant Resistance I & II	Apply cooldown, grant Speed III, use cooldown score to determine when to grant Strength III
Sacrifice	Apply cooldown, aligned with user's eyes, positioned distance away, limit one player in distance, sorted by closest, apply Absorption X,	

	Strength II, Speed II, & /kill user		
Senator	Check if entity exists near user, grant Strength & Resistance		
Sorcery	Apply cooldown, aligned with user's eyes, positioned distance away, fill air with stone	Apply cooldown, aligned with user's eyes, positioned distance away, execute as entities in radius, positioned distance behind, if air, teleport self to position	
Space	Apply cooldown, aligned with user's eyes, positioned distance away, teleport user to location	Apply cooldown, aligned with user's eyes, positioned distance away, sorted by nearest entity limit one, summon snowball at entity location, teleport entity to user, teleport user to snowball, remove snowball	Apply cooldown, execute at user's location, tag all players including self within radius, teleport tagged players to location
Stone	Apply cooldown, aligned with user's eyes, positioned distance away, limit one entity, apply Slowness 255, Jump Boost 127, & Weakness 255		
Sustenance	Check hunger scoreboard, grant Saturation when below threshold	Apply cooldown, execute at user's position, in a radius, apply Hunger	Execute at user's position, in a radius, execute as player, if Hunger below threshold, apply Slowness II & Weakness II
Time	Apply cooldown, tag all existing soul tag armor stands to second soul tag,	Apply cooldown, execute at user's position, in a radius, teleport entities to	Apply cooldown, replace redstone block for Tick Clock with air, after period

	summon armor stand with soul tag at user's position, limit one entity, sorted by closest distance, teleport user to second soul tag variant armor stand & remove armor stand user teleported to	their self-location repeatedly for a period of time	of time, replace air with redstone block for Tick Clock
Vitality	Check damage dealt > 0, add to combo score, otherwise if no damage dealt, add to "out of combat" score, if out of combat score exceeds threshold, reset combo score to 0, apply effects as combo score increases		
Wither	Wait for user to be afflicted with Wither, remove Wither, grant Strength & Resistance, apply Wither III effect to all entities within distance		

Location Titles

To properly display location titles only once to a player when they first enter, we must separately label the players who are in the location from players who are in the location but have already seen the title.

To achieve this goal, we utilize an IUNR on the tick clock to label location tag any player that is within the radius of the center of a location and does not have a locationSeen tag. Then, following a CCAA chain, in this order, we display the title and subtitle to them, label tag them with a locationSeen tag, and then remove from all locationSeen tag players the location tag. Then, on a separate IUNR, we set it to remove any locationSeen tags from any players that are outside the radius of the center of that location.

Dark Mist

For any players that have strayed too far from spawn, we utilize the scoreboard to track how long they have stayed within the area by using distance ranges in the command block selector with an IUNR on the tick clock. We also tag these players with the respective zone they're in using an IUNR and based on their zone's score, then we apply negative effects to them and

reset that score back to zero using CCAA's. The zone's respective tags are removed when they leave the distance range that the zone was assigned with a final IUNR.

Death Penalty

The death penalty tracks recently spawned players through the TimeSinceDeath scoreboard. If a player's score is below a certain threshold, then utilizing an IUNR and CCAA chain, we apply negative effects to them.

Eternity

Eternity structures all utilize scoreboards to monitor contributions from players to their respective upgrades. If applicable, then the buildings' ability cooldowns are monitored in Limbo individually whenever a player is online or collectively. Since the Tick Clock's loop interval is every second, the cooldowns are registered in seconds.

Upgrade buttons conditionally check whether a player possesses any Mana Blossoms to contribute before they subtract it from their score and add it to the building's respective upgrade using an IUNR and CCAA chain.

Structure	Ability A	Ability B
Blacksmith	Positioned <xyz>, check for tempering item and equipment, merge data with equipment, kill tempering item	
Docks	Positioned <xyz>, unless DyeCrates score is 0, give player items and subtract from DyeCrates score - Unless DyeCrates score is <upperLimit-1>, reduce DyeCratesCooldown	Positioned <xyz>, unless BlocksCooldown, give player items
Jeweler	Execute as players with offhand specific item, give effect to self	Positioned as players with offhand specific item, give effect to entities within radius
Mall	Custom modified villager trades	
Mana Well	Positioned <xyz>, add xp to players within radius under level threshold	Positioned <xyz>, unless LiquidManaCooldown, give player item
Mine	Positioned <xyz>, unless YieldCooldown, give player items	
Post Office	As items with specific id, store age into self's scoreboard, if score exceeds <TeleportAge>, positioned on item, teleport all items in	

	radius to location, kill item with specific id on arrival	
Town Hall	Teleport player to location	Positioned <xyz>, within distance, add player to team (with EternityUpgrades scoreboard visible)
Warp Gate	<p>As players holding amethyst shard with specific item nbt, reduce channeling cooldown, give players within radius glowing, teleport when 0</p> <p>As players not holding amethyst shard with specific item nbt, set channeling cooldown to <maxChannelingTime></p>	<p>As items with specific sender id, summon snowball with specific tag, positioned on snowball, teleport items to receiver tag snowball (receiver tag snowball works similarly, but checks whether there is an existing receiver, and doesn't summon a snowball otherwise)</p> <p>As snowballs with specific id, store age into self's scoreboard if score exceeds <LifetimeAge>, kill snowball</p>

Dungeons

Enemy Summons

Due to dungeons being built outside of the playable area in the Dark Mist, players are set in adventure mode first before being given a set number of tag labels for dungeon-related use and then teleported to the location. Dungeon-related tags are how the set location's command block rooms detect whether a player has or has not completed a room to spawn enemies.

Boss Room

When a player enters within an arbitrary distance of a point positioned in the dungeon, one of their tags is removed. Players are not permitted to teleport into the boss arena until all their required respective dungeon tags have been removed.

Boss Mechanics

Bosses' mechanics are timed using their internal clock utilizing scoreboards. At a set score, the boss will perform a certain action or spawn several minions to antagonize the player, and when their interval score reaches its maximum, it is reset to loop the boss mechanics again.

Difficulty

Dungeon difficulties check whether there is a player outside of the entry room but inside the dungeon to determine whether to change the difficulty.

Relics

All relics are dependent on being present in the player's inventory, and a scheduled interval checks if any player's NBT inventory data contains the item and said item matches the custom ID. If the test passes by using an IUNR, then an effect is applied through that block, and CCAA blocks behind it apply additional effects.

Cooldowns

In terms of interval-based relics, if the score for that item on the scoreboard is 0, it is implied it is ready for use, and the test passes, a fill command for a redstone block replacing air is run at a different location for space and organization purposes. An IUNR immediately fills that redstone block with air again, and this triggers CCAA chain (CCAA[C]) to add a score for the appropriate relic in the scoreboards to serve as a cool down and apply the effects necessary. At every scheduled interval, a CUCAA removes a score of 1 from the interval-based relic score up until it is down to 0 again for its next usage.

Name	ID	Sequence / Parallels
Mercurial Pendant	RelicMercurialPendant	IUNR -> CCAA
Mole Goggles	RelicMoleGoggles	IUNR -> CCAA
Odysseus's Voyage	RelicOdysseussVoyage	IUNR -> CCAA
Orb Of Flames	RelicOrbOfFlames	IUNR -> CCAA
Organ Engine	RelicOrganEngine	IUNR -> CCAA[C] CUCAA
Pharaoh's Bandages	RelicPharaohsBandages	IUNR -> CCAA[C] CUCAA
Self-Refilling Goblet	RelicSelfRefillingGoblet	IUNR -> CCAA[C] CUCAA
Tether of the Twins	RelicTetherOfTheTwins+	IUNR -> CUCAA(x3)
	RelicTether OfTheTwins-	IUNR -> CUCAA(x3)
The Giant's Belt	RelicTheGiantsBelt	IUNR -> CCAA
Monument Blueprints	RelicMonumentBlueprints	IUNR -> CCAA
White Peaks Solstice Star	RelicWhitePeaksSolsticeStar	IUNR -> CCAA

Artifacts

Artifacts come in different forms of conditional checks. Current command block-based artifacts can be split into four different categories – held main-hand, toggle, damage dealt, and damage taken.

Main Hand

For most main-hand artifacts, a simple IUNR is enough to facilitate their usage. However, with those that have additional effects, an IUNR tags the wielder with the respective artifact tag and then uses the location of all those tagged to execute the effect based on their location. A CUCAA removes the artifact tag from those who do not have the item selected.

Toggles

For toggleable artifacts, an IUNR checks for those wearing the item and tags them with the respective artifact tag. Another IUNR enables those with the tag to /trigger an artifact objective for their score to turn it on or off. By default, all tagged are set to 0, and once they /trigger the objective, 1 is added to their score. When this is triggered again and their score is 2, a CUCAA is set back to 0. A CUCAA selecting the artifact tag checks if the wielder's score is 1, which activates the effect. Another CUCAA removes the artifact tag from all those who are not wearing the item.

Damage Dealt & Taken

Damage-dealt artifacts use an IUNR to check for those wielding the item and tags them with the respective artifact tag. Selecting all tagged using an IUNR, a scoreboard for their damage dealt is accessed and set to 0. Using an IUNR, their future damage is recorded from the damage dealt scoreboard into an objective for damage dealt using artifacts every interval. When the damage dealt score exceeds a threshold constant set in another objective, the effect occurs, and their damage dealt, and damage dealt using artifacts score is set back to 0 using a CCAA[C]. For damage taken artifacts, substitute damage dealt with taken instead in their respective unique scoreboards.

Empowered Main Hand

During Juggernaut events, the Juggernaut receives some empowered versions of the artifacts which require the item to be in their main hand for additional effects. The equivalent IUNR command block for normal artifacts checks for the same selected item, but with the additional condition for those on the team Juggernaut to apply more potent effects.

Name	ID	Type	Sequence / Parallels
Blood Gouger	ArtifactBloodGouger	Damage Dealt	IUNR -> CCAA IUNR -> CCAA[C] CUCAA
Bramblethorn	ArtifactBramblethorn	Damage Dealt	IUNR -> CCAA IUNR -> CCAA[C] CUCAA
Crown Of Cinders	ArtifactCrownOfCinders	Toggle	IUNR IUNR CUCAA CUCAA CUCAA
Dorsal Fin	ArtifactDorsalFin	Main Hand	IUNR
Flesh Hook	ArtifactFleshHook	Main Hand	IUNR -> CCAA -> RUNR IUNR -> CCAA IUNR -> CCAA IUNR CUCAA CUCAA

Infernal Flame	ArtifactInfernalFlame	Main Hand	IUNR -> CCAA CUCAA
Lilith's Wings	ArtifactLilithsWings	Main Hand	IUNR
Merciless Antumbra	ArtifactMercilessAntumbra	Damage Taken	IUNR -> CCAA IUNR -> x2 CCAA[C] CUCAA
Rhinomane	ArtifactRhinomane	Damage Taken	IUNR -> CCAA IUNR -> CCAA[C] CUCAA
Skysplitter	ArtifactSkySplitter	Damage Dealt	IUNR -> CCAA IUNR -> CCAA[C] CUCAA
The Worldbreaker	ArtifactTheWorldBreaker	Main Hand	IUNR
Void Stone	ArtifactVoidStone	Main Hand	IUNR

Spellbooks

Spellbooks perform a conditional check of the main hand and the offhand and monitor the cooldown of the usage of the item through a scoreboard. All spellbooks are conditionally the same, their only differences are their effects.

An IUNR checks whether a player is holding a specific book and adds a tag to them. As all players of that tag, if the cooldown of the spellbook is ready, a CUCAA+CCAA chain on top of the IUNR checks whether they are also off handing lapis lazuli, removes the lapis lazuli, adds a cooldown, and performs the effect. As all players not holding the book, the tag is removed from them.

Name	ID
Bull's Might	SpellbookStrength
Call of Thunder	SpellbookThunder
Fireball	SpellbookFireball
Lightning Strike	SpellbookLightning
Rejuvenate	SpellbookNature
Wind Spirit	SpellbookWind

Crystals

Crystals interact with the main hand and on the ground as an entity. They have different conditional checks depending on their type.

Type	ID	Sequence / Parallels
Tempering	Quality+<#>	IUNR -> CCAA
Homebound	Homebound	IUNR -> CCAA [C] CUCAA CUCCA CUCCA
Item Gateway	ItemGateway <Send/Receive><#>	

Attuned Rings

Attuned rings interact with the offhand. Most attuned rings are conditionally the same, their only differences are their effects. Some rings have a cooldown specific to the user.

Ring conditional checks are performed as all players whose offhand inventory slot is an item with a specific id to grant their effects. For most rings, an IUNR is the only conditional check. For rings with cooldowns, an IUNR -> CCAA with a parallel CUCAA is necessary. The IUNR adds or removes from the cooldown (depending on your perspective), and if the score is a specific value, it grants the effect. The CUCCA resets all values beyond a certain limit to begin the cooldown again.

Attunement	ID
Fortune	RingOfFortune
Water	RingOfWater
Altitude	RingOfAltitude
Air	RingOfAir
Agility	RingOfAgility
Flight	RingOfFlight
Nourishment	RingOfNourishment
Fire	RingOfFire
Endurance	RingOfEndurance
Force	RingOfForce
Earth	RingOfEarth
Perception	RingOfPerception
Plague	RingOfPlague
Detection	RingOfDetection
Time	RingOfTime
Depths	RingOfDepths
Paranoia	RingOfParanoia

Shadow	RingOfShadow
Fatigue	RingOfFatigue
Safety	RingOfSafety
Anguish	RingOfAnguish
Despair	RingOfDespair

Idols

Idols only function on the ground. They are conditionally checked by an IUNR | CUCCA | CUCCA. The IUNR stores the idol's age into a scoreboard, and a CUCCA checks for all idols' ages that exceed a certain value and perform its effect. The last CUCCA kills the item if it's past that same threshold.

Type	ID
Courier	CourierIdol