This sheet contains references to all relevant workshop-related modding code as well as how to use them. Please read the table of contents and the START section below if you are beginning modding

Table of Contents	Section Type	Use Ctrl + F (PC) or Search function (mobile) in order to skip to a section you need, copy whats listed exactly	Function of section/code
Sections with brackets and no name or number (like [core]) you can only have 1 of	Single	[core]	Core unit functions such as HP and death animation
Sections with NAME at the end can have multiples and be named	Multi	canBuild_NAME	Build queues for building new units
	Single	[graphics]	Main image configuration and unit trails
	Single	[attack]	Attack permissions against other units
MAX: 30	Multi	turret_NAME	Turret that follows attack permissions and uses projectile
	Multi	projectile_NAME	Can be used by turret and can spawn more projectiles
	Single	[movement]	Speed and movement type characteristics
	Single	[ai]	How AI may use this unit, mostly for buildings
!! Arm and Leg can ONLY use numbers	Number	leg_ OR arm_	Moveable cosmetics for mechs, infantry etc.
	Multi	attachment_	Units stacked onto original to make compound units
	Multi	action_NAME OR hiddenAction_NAME	Actions that can do very wide range of things
	Multi	effect_NAME	Visual effects spawned by unit that eventually disappear
	Multi	animation_NAME	Animations for leg, arm or main body sprite
Not a section of its own	List	spawnUnits:LIST	Lists you can form for spawning units
Not a section of its own	List	spawnProjectiles:LIST	Lists you can form for spawning projectiles
Not a section of its own	Logic	logicBoolean_	Advanced logic to place across all logic fields
	Multi	placementRule_NAME	Allows creation of rules for buildings
Not a section of its own	List	Prices/Resources	Useable resources for prices using "price" function
	Multi	global_resource_NAME	Global resource collected by team shows up on top
	Multi	resource_NAME	Local resource used by units can be used for ammo
	Multi	decal_NAME	Versatile graphics can be stack to make 2.5D sprites
Use # before text to use in any section, example: #This is a note in [core]	Multi	comment_NAME	Used exclusively for notes, does not read anything
	Multi	template_NAME	Used to create sections that can be referenced in unit
Is a file that you must create	File	NAME.template	Create chains of unit files doesn't load unless copyFrom:
Is a file that you must create	File	all-units.template	Put in root of mod, will apply to ALL unit .ini files
Is a file that you must create	File	mod-info.txt	Put in root of mod, will change some settings of mod
	Ref	Projectile Draw Types	Default projectiles able to use in [projectile_NAME]

START
In order to create a unit, you need to create a file with extension of .ini, it can be named anything, even with unicode characters, and it must be in your mod folder inside the rustedWarfare/mods/units/MODNAME folder— Then, you add the sections in withe the code

The unit file extension needed: `unit.ini`

There are some lines of code that are required before a unit can be useable, these lines of code can be set to anything appropriate, just need to have their fields filled out for initializing or it will give an error

Error for not having needed code:
```
Error loading mod 'unit': Error loading unit: unit.ini (in mod unit)
could not find constructableImmediateInfo in configuration file in section=attack
```

!! THESE ARE REQUIRED CODE !!
Example

[core]	Code		Value Type
	name: ANY		unit name
	maxHp: 200		number
	price: 500		number
	mass: 50		number
	radius: 16		number

```
[core]
maxHp: 200
name: ANY
price: 100
mass: 50
radius: 16
```

| [graphics] | image: ANY.png | | image file |

```
[graphics]
image: ANY.png
```

[attack]	canAttack: true		true, false
	canAttackFlyingUnits: true		true, false
	canAttackLandUnits: true		true, false
	canAttackUnderwaterUnits: false		true, false

```
[attack]
canAttack: true
canAttackFlyingUnits: true
canAttackLandUnits: true

[movement]
movementType: LAND
```

movementType: NONE, LAND, AIR, WATER, HOVER, BUILDING, OVER_CLIFF, OVER_CLIFF_WATER

For starting out, it is recommended to check out the turret and projectile sections in order to get used to basic vanilla style units, but the modding reference offers a very wide range of things to change in your units, it's encouraged to take your time and read everything

Version Added	Section	Code	Value Type	Description	Example
		[core]			
		#====		**Necessary Code: will cause error if these are not included**	
	name:	string	Defines the unit one name, game uses it to identify as a unique name. (This is not displayed in-game)	name: customTank1
	mass:	int	The 'weight' of the unit, defines how it collides with other units, a greater value means it's tougher to push.	mass: 3000
	radius:	int	Circular area around the unit that makes it selectable. (mouse click/screen touch)	radius: 20
	price:	int / price	The unit cost from builders/buildings. Defaults to credits if no resource type is used	price: 500, gold=5, stone=10
	maxHp:	int	The max health for the unit. (will spawn with this value).	maxHp: 200
		#====		**Common Keys**	
	buildSpeed:	float / s	Time it takes to build the unit. (may multiply with builder speed)	buildSpeed: 3s
	class:	string	Reserved for future use, must be CustomUnitMetadata by default.	class: CustomUnitMetadata
	techLevel	int	Defines the Tech Level of the unit, there're 3 levels and each will appear in a different color in the GUI.	techLevel: 1
	altNames:	strings(s)	Comma separated list of names. Like **name** but lower priority, useful for multiple optional mods.	altNames: cusTank1, customTank1, cTank1
1.13.3	strictLevel:	float	Defaults to 0. 1 = Errors if keys are duplicated. Add to "all-units template" in not to apply to all units.	strictLevel: true
	isBio:	bool	Choose whether the unit is biological or not, affects sound and splat (unless **hideScorchMark=true**)	isBio: true
	isBug:	bool	Changes some death defaults, and sort order in Sandbox.	isBug: false
1.13	isBuilder:	bool	Normally required if this unit places buildings. Defaults to **[ai] useAsBuilder**.	isBuilder: true
1.13	streamingCost:	price	Like price but paid for overtime while this unit is being queued or built. Construction or queue is paused if resources run out while building.	streamingCost: gem=420
1.13	switchPriceWithStreamingCost:	bool	Shortcut to set streamingCost to price value and clear price, add to all-units.template to quickly switch a mod over to streaming resources.	switchPriceWithStreamingCost: true
		#====		**Unit State Keys**	
	selfRegenRate:	float	Passive self repair rate.	selfRegenRate: 0.01
	maxShield:	int	The max shield hitpoints of the unit. Can start with 0 hitpoints if **startShieldAtZero:true**.	maxShield: 500
	startShieldAtZero:	bool	Unit starts with a 0 hitpoints shield on created if true.	startShieldAtZero: true
	shieldRegen:	float	Passive shield regen rate.	shieldRegen: 0.15
	energyMax:	float	Defaults to 0. Energy that can be used as ammo for turrets, laser defense and actions.	energyMax: 1
	energyRegen:	float	Passive energy regen rate.	energyRegen: 0.001
	energyStartingPercentage:	float	Sets the percentage of charged energy when the unit is first built.	energyStartingPercentage: 0.5
	energyNeedsToRechargeToFull:	bool	Disables weapons using energy after reaching zero till fully recharged if true.	energyNeedsToRechargeToFull: true
	energyRegenWhenRecharging:	float	Regen rate while recharging.	
1.13	armour:	int	Damage taken away from each hit. (not currently used in any vanilla units)	armour: 6
1.13	armourMinDamageToKeep:	int	Min damage to keep from received damage. Defaults to 1.	armourMinDamageToKeep: 2
1.13.3	borrowResourcesWhileAlive:	price	Takes these resources when created and returns them when removed or destroyed.	borrowResourcesWhileAlive: gold=10
1.13.3	borrowResourcesWhileBuilt:	price	Like borrowResourcesWhileAlive but doesn't take affect till built. Mostly useful for buildings like houses that have negative resources to add to the unit cap, etc.	borrowResourcesWhileBuilt: supplyCap = -10
1.13.3	generation_resources:	price	Income and creates. (custom resource version)	generation_resources: credits=5, gold=20
1.13.3	generation_active:	logicBoolean	Disables **generation_resources/credits** when false. (logic_boolean)	generation_active: if not self.hpLessThan=100}
	generation_credits:	int	Income and creates. (credits only)	generation_credits: 2
	generation_delay:	int	How often **generation_resources/credits** is added. Defaults to 40. (charging not recommended)	generation_delay: 40
		#====		**UI and Graphics Keys**	
	showInEditor:	bool	Set to false to hide unit in Sandbox editor. (Defaults to true)	showInEditor: false
	displayText:	LocaleString	The unit name that the game shows to the player.	displayText: Custom Tank
1.13	displayText_[LANG]:	string	LANG = ISO 639-1 Code to show this text instead when game is in this language.	displayText_es: Tanque Personalizado
	displayDescription	LocaleString	Unit description that the game shows to the player	displayDescription: -Fast movement/+ Light damage
1.13	displayDescription_[LANG]:	string	LANG = ISO 639-1 Code to show this text instead when game is in this language	displayDescription_es: -Movimiento rápido/+Daño ligero
	displayLocaleKey:	string	Translation file key for unit name and description.	displayLocaleKey: units.mechArtillery
	displayRadius:	int	Defaults to **radius** value. Set to show a larger or smaller selection circle UI on units.	displayRadius: 20
	uiTargetRadius	int	Defaults to **displayRadius** value. Radius used when attacking/reclaiming/etc this unit	uiTargetRadius: 10
	shieldRenderRadius:	int	Defaults to a little bigger than **radius**. Set to show a larger or smaller shield circle on units.	shieldRenderRadius: 12
	shieldDisplayOnlyDeflection:	bool	Hide shield unless deflecting shot if true.	shieldDisplayOnlyDeflection: true
	shieldDeflectionDisplayRate:	float	Defaults to 4. High value causes shield deflection to fade disappear faster.	shieldDeflectionDisplayRate: 3
1.13.3	showOnMinimap:	bool	Set to true. Hide units on minimap if false.	showOnMinimap: false
1.13.3	showActionWithMaxSelectionIfOtherUnitsHaveTag	bool	Shows a merged action list if all units selected includes one of these tags. Useful for converted units.	showActionWithMaxSelectionIfOtherUnitsHaveTag: true
1.14	showOnMinimapToEnemies	bool	Useful for stealth units	showOnMinimapToEnemies: false
		#====		**Building Only Keys**	
	isBuilding:	bool	Defines if the unit is a building.	isBuilding: true
	footprint:	bool	Left, up, right, down. Tiles taken up which block unit movement. Defaults to 0,0,0,0 = 1 center tile.	footprint: 0,0,1,1
	constructionFootprint:	bool	Tiles taken up for placement of other buildings. Defaults to 0,0,0,0 = 1 center tile.	constructionFootprint: -1,-1,1,3
	displayFootprint:	bool	Left, up, right, down. Only applies to buildings, just used for GUI. Defaults to footprint.	displayFootprint: 0,0,1,1
	buildingSelectionOffset:	int	Defaults to 0. Adds or removes padding on the drawn selection rect in UI.	buildingSelectionOffset: 4
	buildingToFootprintOffsetX:	float	Defaults to 10. Change the building position in the footprint on the X-axis.	buildingToFootprintOffsetX: 4
	buildingToFootprintOffsetY:	float	Defaults to 10. Change the building position in the footprint on the Y-axis.	buildingToFootprintOffsetY: 6
	placeOnlyOnRealRoot:	bool	Normally used by extractors, forces building construction in a resource pool.	placeOnlyOnRealPool: true
	selfBuildRate:	float	Rate unit builds itself when placed without a builder.	selfBuildRate: 0.0008
1.14	ignoreUnitCapCalculation	bool	defaults to true for buildings otherwise false. Set to true to not count this unit in unit cap.	ignoreUnitCapCalculation: true
		#====		**Misc Keys**	
	copyFrom:	file(s) (int)	Uses unit data from another ini file as default for this unit, supports multiple files.	copyFrom: ROOT:defaultTanks.template, tankT1.ini
	dont_load:	bool	Do not load unit, and don't error on missing data. Can be useful when used with **copyFrom**.	dont_load: true
	overrideAndReplace:	string(s)	Overrides another unit with this unit. Build links and map positions to target unit will be replaced.	overrideAndReplace: builder, combatEngineer
1.13.3	onNewMapSpawn:	string	Values: emptyResourcePool=_asNeutral, emptyOrOccupiedResourcePool_asNeutral, mapCenter_asNeutral, mapCenter_eachActiveTeam, spawnPoint_eachActiveTeam (1.15)	onNewMapSpawn: spawnPoint_eachActiveTeam
		float		
	isLocked:	bool	Disallow building of this unit. Can be used with **overrideAndReplace** to restrict units player can build.	isLocked: true
1.13	isLockedIfGameModeNoNuke:	bool	Disallows building of this unit if nukes are disabled during match setup.	isLockedIfGameModeNoNuke: true
	experimental:	bool	Tag unit as experimental. Affects zoomed out icon and end game stats.	experimental: true
1.13	stayNeutral:	bool	Set to false to disable capture when unit is on the neutral team.	stayNeutral: false
1.13	createNeutral:	bool	Set to true to always spawn the unit on the neutral team.	createNeutral: true
	createOnAggressiveTeam	bool	Set to true to always spawn the unit on aggressive teams on single player matches.	createOnAggressiveTeam: true
1.13	tags:	string(s)	List of comma separated strings. Used to classify units, create special actions and balances.	tags: tank, smallTank, piercingDamage
1.15			Can define several variables for custom storage, unique for each unit. Allowed types: boolean, float/number, unit, string.	defineUnitMemory: boolean: nukeActive, boolean laserReady, float experience, unit nextTarget, unit homeBase, string customText
1.15p11	defineUnitMemory		Add a pair of square brackets to turn them into arrays for those specific data types	defineUnitMemory: unit[] squad, float[] mango
	fogOfWarSightRange:	int	Sets number of tiles this unit can see through the fog of war. Defaults to 15.	fogOfWarSightRange: 35
1.13	fogOfWarSightRangeWhileNotBuilt:	int	Fog of War range when unit/building is incomplete. Defaults to **fogOfWarSightRange**	fogOfWarSightRangeWhileNotBuilt: 10
1.13	softCollisionOffset:	float	Creates a soft collision effect when touching other units.	softCollisionOnAll: 3
	disableAllUnitCollisions:	bool	Unit cannot collide with others if true.	disableAllUnitCollisions: true
1.13	isUnrepairableUnit:	bool	No unit can repair this unit if true.	isUnrepairableUnit: true
	isUnselectable:	bool	If true unit cannot be selected. (includes AI players)	isUnselectable: true
1.13	isUnselectableAsTarget	bool	Defaults to **isUnselectable**. Can be used to create units that cannot be selected but can be targeted for attack, reclaim, etc.	isUnselectableAsTarget: false
1.13	isPickableStartingUnit	bool	If true, unit is added to dropdowns for starting unit in game setup menus.	isPickableStartingUnit: true
1.12	startFallingWhenStartingUnit	bool	Unit will appear falling from skies when starting unit if true.	startFallingWhenStartingUnit: true
	soundOnAttackOrder:	sound(s)	List of sound names. Only one will be played on each attack order. Only .ogg and .wav formats.	soundOnAttackOrder: tankAttackOrder1.ogg, tankAttackOrder2.ogg
	soundOnMoveOrder:	sound(s)	List of sound names. Only one will be played on each move order. Only .ogg and .wav formats.	soundOnMoveOrder: tankMoveOrder1.ogg, tankMoveOrder2.ogg
1.13.3	soundOnNewSelection:	sound(s)	List of sound names. Only one will be played on each selection. Only .ogg and .wav formats.	soundOnNewSelection: tankSelection1.ogg, tankSelection2.ogg
	canNotBeDirectlyAttacked:	bool	No unit can directly target this unit. If true this will also skip this unit in victory/defeat checks. If true, **canNotBeDamaged** is true by default	canNotBeDirectlyAttacked: true
	canNotBeDamaged:	bool	Defaults to value of **canNotBeDirectlyAttacked**. (be careful setting this without **canNotBeDirectlyAttacked**, as AI will attack forever)	canNotBeDamaged: true
1.13.3	canNotBeGivenOrdersByPlayer:	bool	If true unit will not take player or AI orders.	canNotBeGivenOrdersByPlayer: true
1.13.3	canOnlyBeAttackedByUnitsWithTags:	string(s)	List of tag strings, only units with these tags can directly target this unit.	canOnlyBeAttackedByUnitsWithTags: piercingTank, powerfulTank
1.14	disableDeathOnZeroHp	bool	Setting to true allows unit to continue living even at 0 HP, useful for custom "death" action. Warning: If not used with an autoTrigger, etc units will attack this unit forever	disableDeathOnZeroHp: true
1.15p9	allowCaptureWhenNeutralByAi	bool	When true, it can be captured on contact by AI as well. Defaults as false	allowCaptureWhenNeutralByAi: true
		#====		**Transport Keys**	

Ver	Key	Value Type	Description	Example
1.13	transportSlotsNeeded:	int	Defaults to 1. Number of slots this unit uses up in a transport, experimentals are often set to 5.	transportSlotsNeeded: 2
	maxTransportingUnits	int	Number of slots this units has for transporting other units.	maxTransportingUnits: 5
	transportUnitsRequireTag:	string(s)	Only allows trasport of units that have one of these tags.	transportUnitsRequireTag: smallTank, soldier
1.12	transportUnitsRequireMovementType:	movementTypes	Only allows trasport of units that have one of these movement types.	transportUnitsRequireMovementType: AIR, WATER
	transportUnitsBlockAirAndWaterUnits:	bool	Defaults to true. This unit can only transport LAND units if true.	transportUnitsBlockAirAndWaterUnits: false
1.12	transportUnitsEachUnitAlwaysUsesSingleSlot:	bool	Defaults to false. Units in the transport occupy 1 slot always if true, ignoring transportSlotsNeeded.	transportUnitsEachUnitAlwaysUsesSingleSlot: true
	transportUnitsKeepBuiltUnits	bool	Makes build units stay inside transport instead of exiting it once ready if true.	transportUnitsKeepBuiltUnits: true
	transportUnitsCanUnloadUnits	LogicBoolean	Defaults to (if not self.isOverLiquid() and not self.isMoving()). This unit cannot unload units if false.	transportUnitsCanUnloadUnits: false
	transportUnitsAddUnloadOption:	bool	Defines if unload button should be added to the unit menu	transportUnitsAddUnloadOption: false
1.13.2	transportUnitsUnloadDelayBetweenEachUnit:	float	Changes the delay it takes between each unit getting unloaded.	transportUnitsUnloadDelayBetweenEachUnit: 12
1.13	transportUnitsKillOnDeath:	LogicBoolean	Defaults to true. If false transported units don't die when transport dies.	transportUnitsKillOnDeath: if self.isOverLiquid()
1.13	transportUnitsHealby:	float	Rate to heal units that are being transported.	transportUnitsHealby: 0.1
	transportUnitsBlockOtherTransports:	bool	Defaults to true, if false this transports can hold other transports.	transportUnitsBlockOtherTransports: false
	whileNeutralTransportAnyTeam:	bool	This unit can transport units of any team while neutral if true.	whileNeutralTransportAnyTeam: true
1.13.3	whileNeutralConvertToTransportedTeam:	bool	Converts this unit to transported team while neutral. Useful with whileNeutralTransportAnyTeam.	whileNeutralConvertToTransportedTeam: true
	convertToNeutralIfNotTransporting:	bool	Reverts back this unit to neutral when unloaded. Useful with whileNeutralTransportAnyTeam.	convertToNeutralIfNotTransporting: true
1.13.2	transportUnitsOnTeamChangeKeepCurrentTeam:	bool	Keeps transported units on their original team when this unit is converted if true.	transportUnitsOnTeamChangeKeepCurrentTeam: true
1.13.3	#====	#====	**Resource Node Keys**	
	resourceRate:	float	Used with canReclaimResources. Allows other teams to reclaim this unit. Normally used with neutral team. Use price to set what resources are gained.	resourceRate: 100
	similarResourcesHaveTag:	string(s)	When this has been reclaimed harvester unit moves on to another resource with these tags.	similarResourcesHaveTag: goldResource
	resourceMaxConcurrentReclaimingThis:	int	Defaults to unlimited. Set to restrict how many units can reclaim this resource at the same time.	resourceMaxConcurrentReclaimingThis: 3
1.13.3	reclaimPrice:	price	Like price but for resources. Useful for buildable resources.	reclaimPrice: gold=1000
	#====	#====	**Resource Harvester Keys**	
	canReclaimResources:	bool	If true this unit can gather resources, useful with resourceRate.	canReclaimResources: true
	canReclaimResourcesNextSearchRange:	int	Defines the resource search range of this unit when its main gathered resource runs out.	canReclaimResourcesNextSearchRange: 100
	canReclaimResourcesOnlyWithTags:	string(s)	This unit is only allowed to gather resources with these tags.	canReclaimResourcesOnlyWithTags: foodResource, goldResource
	canReclaimUnitsOnlyWithTags	string(s)	This is for reclaiming units, not for resources. See canReclaimResourcesOnlyWithTags	canReclaimUnitsOnlyWithTags: reclaimable
1.13	resourceReclaimMultiplier	float	Multiplies the builder's reclaim speed. Different from the related key "nanoUnbuildSpeed"	resourceReclaimMultiplier: 1.5
	#====	#====	**Construction and Factory Keys**	
	canRepairUnitsOnlyWithTags	string(s)	Repairs units with the specified tags	canRepairUnitsOnlyWithTags: vulnerable
	canRepairBuildings:	bool	Can this can heal ally buildings. (isBuilder:true is required)	canRepairBuildings: true
	canRepairUnits:	bool	Can this can heal ally units. (isBuilder:true is required). canRepairBuildings required for buildings.	canRepairUnits: true
	autoRepair:	bool	Automatically try and repair damaged units in nano range. (isBuilder:true is required)	autoRepair: true
	nanoRange:	float	Defaults to 85. Defines the unit building/repair/reclaim range.	nanoRange: 110
	nanoRepairSpeed:	float	Defaults to 0.2. Defines the unit nano repair/reclaim speed.	nanoRepairSpeed: 0.01
	nanoBuildSpeed:	float	Defaults to 1. Defines the unit nano building speed. (May multiply with target's buildSpeed)	nanoBuildSpeed: 0.9
1.13	nanoUnbuildSpeed:	float	How fast a builder reclaims an incomplete building (defaults to 1)	nanoUnbuildSpeed: 1.4
	nanoReclaimSpeed:	float	How fast a builder reclaims a normal unit (not a resource unit)	nanoReclaimSpeed: 0.23
	nanoRangeForRepairIsMelee:	bool	Defines if this unit must touch its target to repair it.	nanoRangeForRepairIsMelee: true
1.13.3	nanoRangeForReclaimIsMelee:	bool	Defines if this unit must touch its target to reclaim it.	nanoRangeForReclaimIsMelee: true
	nanoRangeForRepair:	int	Defines a specific range for the repair action of this unit.	nanoRangeForRepair: 60
	nanoRangeForReclaim:	int	Defines a specific range for the reclaim action of this unit.	nanoRangeForReclaim: 60
	nanoFactorySpeed:	float	Defaults to 1. Multiplies the buildSpeed value of the created unit if this unit is a factory.	nanoFactorySpeed: 1.2
	extraBuildRangeWhenBuildingThis:	string(s)	Temporarily adds extra build range to builders to build this unit. Useful for water based buildings.	extraBuildRangeWhenBuildingThis: 90
	buildFrom_{NUM}_name:	string(s)	Useful if adding this unit to build to existing buildings. Like canBuild but in opposite direction.	buildFrom_1_name: landFactory, airFactory
	buildFrom_{NUM}_pos:	float	Order this build link appears in UI. Using canBuild instead is more recommended.	buildFrom_1_pos: 0.1
	buildFrom_{NUM}_forceNano:	bool	Build as if this is a building if true. (even if it's a unit)	buildFrom_1_forceNano: true
1.13.3	buildFrom_{NUM}_isLocked:	LogicBoolean	If true this unit cannot be built in this build link. (can be conditioned if logicBooleans are used)	buildFrom_1_isLocked: if self.hp(lessThan=100)
	buildFrom_{NUM}_isLockedMessage:	LocaleString	Message shown when this build link is locked.	buildFrom_1_isLockedMessage: -Needs more population
	exit_x:	float	Where created or unloaded units appears from the transport or building. Defaults to 0.	exit_x: 0
	exit_y:	float	Where created or unloaded units appears from the transport or building. Defaults to 5.	exit_y: 5
	exit_dirOffset:	float	Defaults to 180 for units and 0 for buildings. Defines the exit direction of created or unloaded units.	exit_dirOffset: 140
1.13	exit_heightOffset:	float	Defaults to 0.0. Defines the height where created or unloaded units appears.	exit_heightOffset: 18
	exit_moveAwayAmount:	float	Defaults to 70. Defines the distance that created or unloaded units move from this unit.	exit_moveAwayAmount: 10
1.14	exitHeightIgnoreParent:	bool	Ignores parent height for exit height; useful for separating attachments with their parents for building	exitHeightIgnoreParent: true
	#====	#====	**Death Keys**	
	dieOnConstruct:	bool	Deletes this unit when it starts to build if true. (target building/unit likely will need setIfBuildRate set)	dieOnConstruct: true
1.13	dieOnZeroEnergy:	bool	Kills this unit if energy level reaches zero when true.	dieOnZeroEnergy: true
	numBitsOnDeath:	int	Defines the number of scattered bit fragments when this unit dies.	numBitsOnDeath: 20
	nukeOnDeath:	bool	Unit will spawn a nuke detonation built-in effect when dies if true.	nukeOnDeath: true
	nukeOnDeathRange:	float	Defines the nuke effect range when using nukeOnDeath.	nukeOnDeathRange: 140
	nukeOnDeathDamage:	float	Defines the nuke effect area damage when using nukeOnDeath.	nukeOnDeathDamage: 2000
	nukeOnDeathDisableWhenNoNuke:	bool	Defaults to false. If true this unit will not explode with nuke when nukes are disabled in skirmish maps.	nukeOnDeathDisableWhenNoNuke: true
	fireTurretXAtSelfOnDeath:	turret ref	Auto-shoot a specific turret when this unit dies.	fireTurretXAtSelfOnDeath: turret_1
1.12	explodeOnDeath:	bool	Defaults to true. Disables the unit death explode built-in effect if false.	explodeOnDeath: false
	explodeOnDeathGroundCollision:	boolean	Defaults to true. Disables the explode built-in effect on death when unit touches ground if false.	explodeOnDeathGroundCollision: false
	explodeTypeOnDeath:	preset effects	example: wreckceil, small, normal, large, largeUnit, building, buildingMultiJuiceworldSmoke, veryLargebuilding	
1.12	effectOnDeath:	effect(s) ref	Spawns built-in or custom effects when unit dies.	effectOnDeath: shockwave, CUSTOM:pieces*3, CUSTOM:deathSound
1.13	effectOnDeathGroundCollision:	effect(s) ref	Like effectOnDeath but when unit touches ground. Useful for flying units	effectOnDeathGroundCollision: CUSTOM:bigExplosion
	unitsSpawnedOnDeath:	string(s)	Spawns these units when dies. Comma separated unit identifiers.	unitsSpawnedOnDeath: tank*5, hoverTank
	unitsSpawnedOnDeath_setToTeamOfLastAttacker:	bool	Units spawned on death will appear on the last attacker team if true.	unitsSpawnedOnDeath_setToTeamOfLastAttacker: true
	hideScorchMark:	bool	Disables the death scorch mark leaved when unit dies if true.	hideScorchMark: true
	soundOnDeath:	string(s)	Sets a custom sound for this unit death.	soundOnDeath: tankExplosion1.ogg, tankExplosion2.ogg
1.14	effectOnDeathIfUnbuilt:	effect(s) ref	If the unit was not completed, and is destroyed, play this effect. Defaults to effectOnDeath	effectOnDeathIfUnbuilt: CUSTOM:implode
	#====	#====	**Action Keys**	
1.13.2	autoTriggerCooldownTime:	time (seconds)	Post automatic action cooldown (Not detection cooldown). Defaults to 1s. Warning: Setting this too low for many units might effect performance depending on the action effects.	autoTriggerCooldownTime: 0.05s
	autoTriggerCooldownTime_allowDangerousHighCPU:	bool	Allows for auto action cooldown lower than 0.2s. Default to false. Not recommended.	autoTriggerCooldownTime_allowDangerousHighCPU: true
1.13	autoTriggerCheckRate:	enum	options: everyFrame (default), everyXFrames, everyXFrames. Note: all triggers regardless of check rate are checked when first created and after an auto trigger cooldown. Note: Adding [core] autoTriggerCheckRate:everyXFrames to all-units.template could have a large performance boost for mods with complex autoTriggers.	autoTriggerCheckRate:every8Frames
1.13	autoTriggerCheckWhileNotBuilt	bool	Defaults to false. autoTrigger of unit actions check even when not completely built if true	autoTriggerCheckWhileNotBuilt: true
1.15, 18	updateUnitMemory:	memory	Faster memory update than [action_#] setUnitMemory. Useful on many applications.	updateUnitMemory: timeCount = (1 + memory.timeCount)
	updateUnitMemoryRate:	int	Sets how often the memory is updated. Defaults at 1s. settings it to 0 will update memory every frame.	updateUnitMemoryRate: 0
1.14	@memory		A template-friendly method of defineUnitMemory. Declare name followed by type, separated by a color	@memory:fish:unit
			Add a pair of square brackets for array memory type	@memory:bullets:number
1.15p11				@memory:myTargets:unit[]

[canBuild_NAME]

Code	Value Type	Description	Example
name:	string(s)	List of unit identifiers this unit can create. Can be buildings or units. Add "setRally" to create a rally button	name: setRally, tank, hoverTank, heavyTank
pos:	float	Order build link appears in this unit UI.	pos: 0.1
tech:	int	Tech level. Mostly just affects build link colour in this unit UI. Defaults to 1.	tech: 2
forceNano:	bool	Builds target as if it was a building if true. (even if it's a unit)	forceNano: true
isVisible:	LogicBoolean	Hide this build link if true in this unit UI.	isVisible: if not self.energy(greaterThan=100)
isLocked:	LogicBoolean	Dynamically locks this build option and shows isLockedMessage if true.	isLocked: if self.energy(lessThan=100)
isLockedMessage:	string	Set to tell to players why a unit is locked.	isLockedMessage: -Needs 2 Barracks
1.13.3 isLockedMessage_{LANG}:	string	LANG = ISO 639-1 Code to show this text instead when game is in this language.	isLockedMessage_es: -Necesita 2 Cuarteles
isLockedAlt:	LogicBoolean	Another reason for this to be locked. Just allows a different message to be shown.	isLockedAlt: if self.energy(greaterThan=90)
isLockedAltMessage:	string	Message for isLockedAlt.	isLockedAltMessage: -Needs less energy
isLockedAlt2:	LogicBoolean	Like isLockedAlt but to show one more message.	isLockedAlt2: if self.isMoving()
isLockedAlt2Message:	string	Message for isLockedAlt2.	isLockedAlt2Message: -Needs to be quiet
addResources:	price(s)	Adds these resources to self when placing the building or producing the unit.	addResources: ammo=5, setFlag=1
price:	price(s)	Overrides builded units/buildings price. Defaults to target unit prices.	price: credits=1000, ammo=5
isGuiBlinking:	LogicBoolean	Generates a blinking effect in UI if true.	isGuiBlinking: true

[graphics]

Code	Value Type	Description	Example
#====	#====	**Necessary code**	
image:	file (image)	Image:	
#====	#====	**Common Keys**	
image:	file (image)	File path to png image.	image: body.png
1.13	teamColoringMode:	file (image)	How pixels are used for team coloring, options: pureGreen, hueAdd, hueShift, disabled	(Defaults to pureGreen if no code) teamColoringMode: disabled
	image_back:	file (image)	An optional image drawn behind other units. Useful for factories that units exit	image_back: floor.png
1.14	image_shield:	file (image)	Image to show as a custom shield	image_shield: forcefield.png
	image_weak:	file (image)	Image to use when unit dies. Can be NONE to leave no weak image.	image_weak: ded.png
	image_offsetX:	int	Use this to adjust the graphics of a unit if it is too far off one side	image_offsetX: 21
	image_offsetY:	int	Use this to adjust the graphics of a unit if it is too far off one side	image_offsetY: 41
	image_offsetR:	int	Use this to adjust the height of the graphic, especially when using decals	image_offsetR: 25
1.15p8	image_floatingPointSize	int	Fixes of by 1 pixel sizing for images with widths and/or height that has odd value	image_floatingPointSize: 10
1.13.3	isVisible:	logic	If false will hide the unit.	isVisible: false
1.14	isVisibleToEnemies	bool	Only visible to player and allies when false. Recommend with showOnMinimapToEnemies. Useful for stealth units.	isVisibleToEnemies: false
	scaleImagesTo:	float	Resize image to fit this value in pixels. Effects leg, and shadow images as well.	scaleImagesTo: 1
	imageScale:	float	Resize image. Defaults to 1. Effects leg, and shadow image as well.	imageScale: 1
	drawLayer:	string	Land units normally default to ground or ground2 if transport. Options: wreaks, underwater, bottom, ground, ground2, ... experimentals, air, top	drawLayer: bottom
1.12	whenBeingBuiltMakeTransparentTill	float	How long the transparent effect is applied to incomplete units, set to 0 to disable completely. Default: 1	whenBeingBuiltMakeTransparentTill: 0.4
	icon_zoomed_out	file (image)	Sets the custom image of the unit icon on the zoomed out battle map	icon_zoomed_out: unitIcon.png
	icon_zoomed_out_neverShow	bool	Sets wheather show the unit icon or not on the zoomed out battle map	icon_zoomed_out_neverShow: false
	icon_build	file (image)	Displays icon of the unit on Build Panel	icon_build: iconFish.png
	#====	#====	**Turrets (images can also be set on each turret)**	
	image_turret:	file (image)	Default image for all turrets, can also be set per turret	image_turret: gun.png
	teamColorsOnTurret	bool	Defaults false. Apply team colours on turret as well. Also affects pre-turret images	teamColorsOnTurret: true
	scaleTurretImagesTo:	float	Will cause resort if image_turret is not specified, even if image is set per turret	scaleTurretImagesTo: 1.2
	lock_body_rotation_with_main_turret	bool	Locks body locked to the direction of the turret named "1" (turret_1) or the first turret defined (if a turret_1 is not defined)	lock_body_rotation_with_main_turret: true
1.13	lock_leg_rotation_with_main_turret	bool	Locks legs and arms to the direction of the turret named "1" (turret_1) or the first turret defined (if a turret_1 is not defined)	lock_leg_rotation_with_main_turret: true
	#====	#====	**Shadow**	
	image_shadow:	file (image)	Image file, NONE, AUTO or AUTO_ANIMATED. (AUTO will use image and make it transparent black only.)	image_shadow: shadow.png
	shadowOffsetX:	float	Adjusts shadow horizontally	shadowOffsetX: 10
	shadowOffsetY:	float	Adjusts shadow vertically	shadowOffsetY: 10
1.13.3	image_shadow_frames	bool	If shadow image should use frame animation of main image. Alternative to AUTO_ANIMATED shadow value.	image_shadow_frames: true
	lock_shadow_rotation_with_main_turret	bool	Locks body image shadow locked to first turret's direction	
	#====	#====	**Effects and animation**	
	total_frames	int	Defaults to 1. Animations require this.	total_frames: 3
	frame_width:	int	Calculated for you if total frames is set, but can be overridden	frame_width: 40
	frame_height:	int	Defaults to image height	frame_height: 60
1.147	default_frame	int	Default frame when not playing an animation. First frame is 0	default_frame: 0
	splashEffect:	bool	True to create a water wave effect when over water. Default false.	splashEffect: true
	dustEffect:	bool	True to create a dust effect when over land. Default false	dustEffect: true
	splashEffectReverse:	bool	True to also create effect when unit is reversing	splashEffectReverse: true
	dustEffectReverse:	bool	True to also create effect when unit is reversing	dustEffectReverse: true
1.13	movementEffect:	effect	Custom movement effect, can be anything	movementEffect: smoke, CUSTOM:fastDust*2, CUSTOM:pop*5
1.13	movementEffectReverse:	effect	Reversed version of movementEffect: key	movementEffectReverse: smoke, CUSTOM:fastDust*2, CUSTOM:pop*5
1.13	movementEffectRate:	float	Sets the frequency of effects being spawned while moving	movementEffectRate: 50
1.13	movementEffectReverseFlipEffects:	bool	Create effect as if unit has rotated 180 when reversing	movementEffectReverseFlipEffects: true
	repairEffect:	effect ref	Custom repair effect, can be anything. Replaces default effect from builders	repairEffect: CUSTOM:nut*5
	repairEffectAtTarget:	effect	Custom repair effect towards the repaired target	repairEffectAtTarget: CUSTOM:greensparks*2
	repairEffectRate:	float	Sets the frequency of effects being spawned while repairing. Defaults to 5, effects both sides.	repairEffectRate: 20
1.13.2	reclaimEffect:	effect	Custom reclaim effect, can be anything. Replaces default effect from builders	reclaimEffect: CUSTOM:sap*5
1.13.2	reclaimEffectAtTarget	effect	Custom reclaim effect towards the reclaimed target	reclaimEffectAtTarget: CUSTOM:redsparks*2

Ver	Code	Value Type	Description	Example
1.13.2	reclaimEffectRate	int	Sets the frequency of effects being spawned while reclaiming. Defaults to 5, effects both sides.	reclaimEffectRate: 20
	rotate_with_direction:	bool	Defaults to true. Makes unit body image locked to 0 degrees when false. Often used with animation_direction_*.	rotate_with_direction: false
	animation_direction_units:	float	45 for 8 directions, 90 for 4 direction animation. Used with rotate_with_direction:false	animation_direction_units: 45
	animation_direction_strideX:		Animation frames to offset on direction change.	animation_direction_strideX: 40
	animation_direction_strideY:	int	Animation frames to offset on direction change. Used with frame_height.	animation_direction_strideY: 60
	animation_direction_starting:	float	Direction for first frame	animation_direction_starting: 90
	#=====	#=====	**Show and Hide**	
1.13.3	disableLowHpFire	bool	Defaults to false. When true, removes the flames on severely damaged unit/structure.	disableLowHpFire: false
1.13.3	disableLowHpSmoke	bool	Defaults to false. When true, removes the smokes on severely damaged unit/structure.	disableLowHpSmoke: false
1.13.2	showTransportBar	bool	Defaults to true. Shows the transport bar for transport units.	showTransportBar: false
1.13.3	showHealthBar	bool	Defaults to true. Shows the health bar of a unit. (Still hidden even if true when "Always show unit health" setting is disabled.)	showHealthBar: false
1.13.2	showEnergyBar	bool	Defaults to true. Shows the energy bar of a unit.	showEnergyBar: false
1.14	showShieldBar	bool	Defaults to true. Shows the shield bar of a unit. Also affects the presence of set custom shield sprite.	showShieldBar: false
1.14	showQueueBar	bool	Defaults to true. Shows the bar for build or action queue of a unit.	showQueueBar: false
0.99	showShotDelayBar	bool	Defaults to true. Shows the shot delay/warmup of a unit before firing.	showShotDelayBar: false
1.15p0	showSelectionIndicator	bool	Defaults to true. Shows select circle/box if true	showSelectionIndicator: false
	#=====	#=====	**Deprecated Keys (can be used but there are better, more adaptable ways)**	

(deprecated/strikethrough rows — faded, related to animation keys)

[attack]
The attack section is for global attack characteristics, per-turret overrides these

Ver	Code	Value Type	Description	Example
	canAttack:		If set to false, can not attack any unit. Overrides other canAttack options altogether.	canAttack: true
	canAttackFlyingUnits:	LogicBoolean	can also be narrowed per turret. Note: not required if canAttack is false.	canAttackFlyingUnits: false
	canAttackLandUnits:	LogicBoolean	can also be narrowed per turret in [turret_NAME]	canAttackLandUnits: true
	canAttackUnderwaterUnits:	LogicBoolean	can also be narrowed per turret in [turret_NAME]	canAttackUnderwaterUnits: false
	maxAttackRange:	float	(multiplied by globalScale)	maxAttackRange: 300
	shootDelay	float	Global delay; can also use delay on each turret. Set in seconds or ticks	shootDelay: 21, shootDelay: 5s
	canAttackNotTouchingWaterUnits:	LogicBoolean	Default true. If false unit can only attack units in contact with the water. Used for units with torpedos. (can also be set per turret)	canAttackNotTouchingWaterUnits: false
1.13.3	canOnlyAttackUnitsWithTags	tags	Will only attack units that has the specified tags.	canOnlyAttackUnitsWithTags: metallic, ceramic
1.13.3	canOnlyAttackUnitsWithoutTags	tags	Can only attack units without the specified tags.	canOnlyAttackUnitsWithoutTags: corrosive, magnetic
1.13	turretMultiTargeting	bool	Allow each turrets to fire at a different target at the same time. Very useful if [turret]limitingAngle is used	turretMultiTargeting: true
	isMelee:	bool	Used with a low attack range (like maxAttackRange: 0) makes src and target radius get added to range, and effects AI.	isMelee: true
1.13.3	meleeEngagementDistance	int	Makes unit move to attack nearby units. Defaults to 250 for melee, and 0 for non melee (Works even if non-melee, but might be unexpected to players)	meleeEngagementDistance: 400
	turretRotateWithBody	bool	Are all turrets rotated with body rotates. Defaults to true	turretRotateWithBody: true
	attackMovement	string	normal/bomber. bomber attack movement will retreat when energy runs out	attackMovement: bomber
	dieOnAttack:	bool	Will die when it attacks.	dieOnAttack: true
	isFixedFiring:	bool	Must aim body at target to shoot. Will often make the unit need to stop before it can aim and shoot.	isFixedFiring: true
	aimOffsetSpread:	float	Offset each shot multiplied by target radius. Defaults to 0.6. aimOffsetSpread:0 will make unit always attack center	aimOffsetSpread:0
1.12	stopTargetingAfterFiring	bool	Unit stops targeting after firing a shot. Rarely used or needed.	stopTargetingAfterFiring: true
	disablePassiveTargeting:	bool	Unit only attacks manually ordered target. Rarely used or needed.	disablePassiveTargeting: true
1.13.3	showRangeUIGuide	bool	Will it show the range indicator. Useful for showing ranges in radar and related structures.	showRangeUIGuide: true
0.99	shootDelayMultiplier	float	Defaults to 1. Can be dynamically changed with setUnitStats	shootDelayMultiplier: 0.75
1.15	shootDamageMultiplier	float	Defaults to 1. Can be dynamically changed with setUnitStats	shootDamageMultiplier: 2.43
	#=====		**Deprecated Keys - can be used better to set these per turret**	

[turret_NAME]
Turrets fire projectiles with different traits

Ver	Code	Value Type	Description	Example
	#=====	#=====	**Necessary Code**	
	x:	float		
	y:	float		
	#=====	#=====	**Positioning/Stats**	
	x:	float	Sets the horizontal position of the turret. 0 is center, negatives toward left, positives toward right.	x: 0
	y:	float	Sets the vertical position of the turret. 0 is center, negatives toward bottom, positives toward top.	y: 40
	copyFrom:	turret ref	Copy all values from another turret as default values for this turret	copyFrom: 1
	projectile	projectile ref	Projectile fired from this turret.	projectile: torpedo
	altProjectile	projectile ref	Alternative projectile fired from this turret when altProjectileCondition is true	altProjectile: spear
	altProjectileCondition	LogicBoolean	Used with altProjectile	altProjectileCondition: if self.hp() < 100
1.13	barrelX	float	Defaults to 0. Controls horizontal position for projectile spawn.	barrelX: 0
1.13	barrelY	float	Defaults to size. Note: size and barrelY have the same meaning	barrelY: 5
1.13.3	barrelHeight	float	Height of barrel in 3d. Effect projectile and shoot flame starting height	barrelHeight: 10
1.15p0	height:	float	Height of the unit in 3D, to be used with decals	height: 30
	size:	float	Controls the distance between the center of the turret and the point from where projectiles spawn.	size: 5
	turnSpeed	float	Max turn speed of the turret	turnSpeed: 2
	turnSpeedAcceleration	float	Defaults to disabled, and full turn speed is used.	turnSpeedAcceleration: 0.4
	turnSpeedDeceleration	float	Defaults to turnSpeedAcceleration. Setting this higher than turn acceleration might allow faster targets to be hit	turnSpeedDeceleration: 0.2
	idleDir	float	Defaults to 0	idleDir: 180
1.13	idleDirReversing	float	Defaults to idleDir+180 unless attached to another turret (be attached turret will often be rotating when reversing).	idleDirReversing: 30
	shouldResetTurret:	bool	Defaults true. False to disable the reseting turret angle when idle	shouldResetTurret: false
	idleSpin:	float	Spin rate when idle, used on missile turrets	idleSpin: 2
1.14	idleSweepAngle	int	Disabled by default. Controls how far the turret will "look" left and right	idleSweepAngle: 35
1.14	idleSweepDelay	float	Controls the delay between idleSweep movements	idleSweepDelay: 20
1.14	idleSweepSpeed	float	Controls the speed with which the turret sweeps when idle	idleSweepSpeed: 0.3
1.14	idleSweepCondition	LogicBoolean	Disable idle sweep if false	idleSweepCondition: if self.hp() < 400
1.14	idleSweepAddRandomDelay	float	Default 1-20 depends on idleSweepDelay, used to stop sweep syncing up with other units	idleSweepAddRandomDelay: 30
1.14	idleSweepAddRandomAngle	float	Default 0	idleSweepAddRandomAngle: 15
	attachedTo:	turret ref	A specific turret to connect to, it will rotate with it. No correlation to [attachment_x]	attachedTo: base
	slave:	bool	Locks this turret's direction and shot cooldown to attached turret. Often used with warmup for multiple barrel guns	slave: true
	isMainNanoTurret:	bool	Defaults to false. Turret to use for creating buildings, etc. should only be true on one turret, and should have canShoot set to false	isMainNanoTurret: true
	energyUsage:	float	Required energy to fire weapon. Same as resourceUsage: energy=X	energyUsage: 1
	resourceUsage	price	can be in credits/energy/hp/shield/ammo. Stops firing if not met	resourceUsage: credits=5, energy=5, hp=100, shield=5, ammo=1
	#=====	#=====	**Timing**	
	delay:	float	Override global shootDelay for this turret	delay: 2s
	linkDelayWithTurret	turret ref	When this other turret fires the cooldown delay on this turret will be reset/removed	linkDelayWithTurret: 1
	warmup:	float	Delay before firing a shot.	warmup: 5s
	warmupCallDownRate	float	Rate to reduce warmup when turret is not ready to fire at any targets	warmupCallDownRate: 10
	warmupNoReset	bool	Defaults to false. When true warmup is not reset after firing a shot and turret doesn't wait for warmup. Used with warmupCallDownRate and warmupShootDelayTransfer.	warmupNoReset: true
	warmupShootDelayTransfer	float	Defaults to 0, amount to reduces the next shot delay depending on warmup. When used with warmupNoReset, can make a each shot faster.	warmupShootDelayTransfer: 20
1.13.3	onShoot_freezeBodyMovementFor:	float	Freezes body movement while shooting.	onShoot_freezeBodyMovementFor: 2s
	barrelOffsetX_onOddShots:	float	0 by default. Sets a barrelX offset only during odd numbered shots, useful for twin-barreled units. Use with barrelX. Use the opposite value of barrelX for most cases.	barrelOffsetX_onOddShots: -10
	#=====	#=====	**Targeting control**	
1.14	aimOffsetSpread:	float	Sets the shot inaccuracy. 0 will make all shots land on target's center	aimOffsetSpread: 14
	canShoot:	bool	Defaults to true	canShoot: true
	canAttackFlyingUnits:	LogicBoolean	Narrows targeting for this turret, note targeting for the whole unit in [attack] is applied first. (so you can only use this to target less not more)	canAttackFlyingUnits: false
	canAttackLandUnits:	LogicBoolean	Default true. If false unit cannot attack surface units.	canAttackLandUnits: true
	canAttackUnderwaterUnits:	LogicBoolean	Default true. If false unit cannot attack underwaterunits.	canAttackUnderwaterUnits: true
	canAttackNotTouchingWaterUnits:	LogicBoolean	Default true. If false unit can only attack units in contact with the water. Used for units with torpedos.	canAttackNotTouchingWaterUnits: false
	canOnlyAttackUnitsWithTags	tags	Only attacks potential targets with specific tags	canOnlyAttackUnitsWithTags: shelled, armoured
1.13.3	canOnlyAttackUnitsWithoutTags	tags	Only attacks potential targets without specific tags	canOnlyAttackUnitsWithoutTags: corrosive, flashy
	canAttackCondition:	LogicBoolean	Normally, used to optionally disable a turret based on a LogicBoolean. Eg this unit's height	canAttackCondition: if not self.flying
1.14	canAttackMaxAngle:	float	Max angle to target for turret to be allowed for this turret to fire. Defaults to 5, don't set lower. Can be set to 181 for turrets that don't need to turn to fire rockets.	canAttackMaxAngle: 181
1.13	clearTurretTargetAfterFiring:	bool	Clears the turrets sub-target when using multi-targeting	clearTurretTargetAfterFiring: true
	limitingRange:	float	Make this turret have less range than the maxAttackRange. Do not apply this to all turrets change maxAttackRange instead.	limitingRange: 100
1.13	limitingAngle	int	Linked with idleDir. Turret will only be able to fire at units +/- this angle.	limitingAngle: 60
1.13	limitingMinRange	float	Sets minimum range for turret.	limitingMinRange: 200
1.13	interceptProjectiles_withTags	tags	Currently used with anti-nuke units.	interceptProjectiles_withTags: nuke
	interceptProjectiles_andTargetingGroundUnderDistance:	int	Minimum distance for intercepting projectile	interceptProjectiles_andTargetingGroundUnderDistance: 100
	interceptProjectiles_andUnderDistance:	int	Defaults to 2000, distance inflight before firing.	interceptProjectiles_andUnderDistance: 300
	interceptProjectiles_andOverHeight:	int	Defaults to 0. Sets the projectile target height limit	interceptProjectiles_andOverHeight: 10
	laserDefenceEnergyUse:	float	Set to enable a projectile laser defence from this turret. Should also set the energyMax in core.	laserDefenceEnergyUse: 0.25
	#=====	#=====	**Graphics and effects**	
	invisible:	LogicBoolean	Don't render this turret, but still can shoot, etc.	invisible: true
	image:	file (image)	Use custom image. Overrides unit's main turret image	image: gun.png
	image_applyTeamColors	bool	When true, the specific turret applies team colors over the special greens in the sprite.	image_applyTeamColors: true
	image_drawOffsetX	float	Moves the image horizontally from the turret center.	image_drawOffsetX: 0
	image_drawOffsetY	float	Moves the image vertically from the turret center.	image_drawOffsetY: 30
	chargedEffectImage:	file (image)	Used with warmup. Shows a scaling effect image on turret barrel when charging.	chargedEffectImage: glow.png
	warmupStartEffect	effect ref	Spawns specified effects when the specific turret is charging/warming up	warmupStartEffect: CUSTOM:absorb
	shoot_sound:	string	Can be linked to an .ogg or .wav file, or one of the default sounds.	shoot_sound: tank_firing	shoot_sound: missile.wav	shoot_sound: ROOT/audio/shoot.ogg
	shoot_sound_vol:	float	Sets the volume of the shoot sound	shoot_sound_vol: 0.4
	shoot_flame:	effects	Current types are: small, large, smoke, shockwave, or CUSTOM: effectSectionName	eg: shoot_flame: smoke, CUSTOM:lightFade, CUSTOM:pop*5
	shoot_light	color	Produces a colored light upon shooting.	shoot_light: #fefefe
1.13	onShoot_playAnimation	animation ref	Play a custom animation from an [animation] section after firing this turret	onShoot_playAnimation: heatUp
1.14	onShoot_triggerActions	action refs	Trigger these actions each time this turret fires	onShoot_triggerActions: spawnShells
1.15p0	unloadUpToXUnitsAndGiveAttackOrder	int	Unloads X units at turret barrel locations and gives them the attack order of turret target	unloadUpToXUnitsAndGiveAttackOrder: 1
	recoilOffset:	float	Push turret forward or back after firing for a recoil effect. Value in pixels.	recoilOffset: 10
	recoilOutTime	float	Time to get to offset position after firing	recoilOutTime: 1s
	recoilReturnTime	float	Time to return to default position	recoilReturnTime: 0.5s
	showRangeUIGuide	bool	Defaults to true. Displays the attack range for this turret	showRangeUIGuide: true

[projectile_NAME]
Projectiles are often used to inflict damage on an enemy, but can also be used for other purposes

Ver	Code	Value Type	Description	Example
	#=====	#=====	**Necessary Code**	
	directDamage: or areaDamage:	int		
	life:	float		
	#=====	#=====	**Core**	
	life:	float	How long till this projectile gets removed if it hasn't hit a target. 300 might be a good starting point, change depending on speed and range	life: 300
	deflectionPower:	float	Defaults to 1. Energy needed for laser defence to deflect. -1 to disable deflection (only disable for special weapons like flames)	deflectionPower: -1
1.13	explodeOnEndOfLife	bool	Default to false. True to explode at end of life with all side effects and area damage instead of disappearing. Good for making area-denial weapons.	explodeOnEndOfLife: true
	autoTargetingOnDeadTarget:	bool	Retarget to nearby unit if target dies while in transit	autoTargetingOnDeadTarget: true
1.14	autoTargetingOnDeadTargetRange	float	The range in which it will select a new target if old target has died	autoTargetingOnDeadTargetRange: 100
1.14	autoTargetingOnDeadTargetLead	float	The lead it will try to have on the new target	autoTargetingOnDeadTargetLead: 300
1.13	unloadUpToXUnitsFromSource	int	Unload X units from source unit, to projectile explode location	unloadUpToXUnitsFromSource: 2
1.12	teleportSource	bool	Move unit that shot the projectile to projectile explode location	teleportSource: true
			NOTE: There is a section for all possible modifications to spawnUnit and spawnProjectile, type "spawnUnitsLIST" or "spawnProjectilesLIST" in the Search function	
1.13	spawnUnit	unit types	Spawn new units of this type at projectile explode location	eg: spawnUnit: heavyTank, tank*5, hoverTank(offsetX=+10)
1.13	spawnProjectilesOnEndOfLife	projectile ref	Spawns new [projectile] at end of life, useful for secondary weapons	spawnProjectilesOnEndOfLife: torpedo_split(offsetX=90), torpedo_split(offsetX=-90)
1.13	spawnProjectilesOnExplode	projectile ref	Projectiles to spawns when this projectile hits it's target	spawnProjectilesOnExplode: flechette*4
1.14	spawnProjectilesOnCreate	projectile ref	Spawns projectiles on creation of this projectile, useful for making true shotgun-like projectile spread	spawnProjectilesOnCreate: stage2
1.13	convertHitToSourceTeam	bool	Convert units hit to the team that fired this projectile. Useful to make capturing systems	convertHitToSourceTeam: true
	tags	tags	Used for projectile interceptions. (e.g. Nuke and Anti-Nuke Interaction) and in [action] autoTriggerOnEvent: lookDamage (with Tag="[tagName]")	tags: nuke, missile
	flameWeapon:	bool	Generates small flames on hit (only cosmetic)	flameWeapon: true
1.13	delayedStartTimer	float	Hide for a time before showing and updating effect.	delayedStartTimer: 2s

Damage

Version	Code	Value Type	Description	Example
#====	#====		Damage	#====
	directDamage:	int	Damage to target unit on hit. Does not work with targetGround: true as it won't have a clear unit to target	directDamage: 30
	areaDamage:	int	Damage on arrival of target with an area effect, use areaRadius to adjust size of damage. targetGround needs this to damage	areaDamage: 10
1.13	areaRadius:	float	How wide areaDamage effects. Note the drops off (unless areaDamageNoFalloff is used)	areaRadius: 100
	areaDamageNoFalloff	bool	Removes the falloff from areaDamage	areaDamageNoFalloff: true
	areaRadiusFromEdge	bool	Applies damage from edge of units instead of center. Mostly effects large units.	areaRadiusFromEdge: true
1.13	areaExpandTime	float	Applies area damage as an expanding blast wave rather than instantly. Useful for nuke projectiles	areaExpandTime: 1s
1.12	areaHitAreaLandAtSameTime	bool	Defaults to false	areaHitAreaLandAtSameTime: true
	areaHitUnderwaterAlways	bool	Defaults to false	areaHitUnderwaterAlways: true
	areaIgnoreUnitsCloserThan		Units closer than this range aren't effected. Rarely needed. Not recommended for normal projectiles	areaIgnoreUnitsCloserThan: 10
1.13	buildingDamageMultiplier	float	Defaults to 1	buildingDamageMultiplier: 3
1.13	shieldDamageMultiplier	float	Defaults to 1. 0 to do no damage to shields and 2 to do double	shieldDamageMultiplier: 1.4
1.13	shieldDeflectionMultiplier	float	Defaults to 1. The amount of shield to bypass. 0 to ignore shields and directly damage hull	shieldDeflectionMultiplier: 1.55
1.14	hullDamageMultiplier	float	Defaults to 1. Can be used to create EMP weapons that effect shields only, 0 to ignore hull and directly damage hull	hullDamageMultiplier: 2
0.00	ignoreParentShootDamageMultiplier	bool	Sets whether multipliers are applied on not regardless of multipliers set in attack or setUnitStats	ignoreParentShootDamageMultiplier: true
	armourIgnoreAmount	int	Amount of armour to ignore on target and do damage as if this armour was not there	armourIgnoreAmount
1.13	friendlyFire	bool/string	Lets area effect projectiles damage own team units (can't damage allies). Useful for nuke-like weapons	friendlyFire: false / friendlyFire: true / friendlyFire: only-ignoreEnemy
1.13.3	mutator1_ifUnitWithTags	tags	Applies mutators to this projectile if target has corresponding tags	mutator1_ifUnitWithTags: infantry
1.13.3	mutator1_ifUnitWithoutTags	tags	Same as ifUnitWithTags, but applies if target doesn't have the set tags	mutator1_ifUnitWithoutTags: strongArmour
1.13.3	mutatorX_directDamageMultiplier	float	Changes directDamage. Defaults to 1. Be careful not to confuse players using this as the effect may not be clear. Use amour instead when possible. Replace X with desired name	mutatorX_directDamageMultiplier: 5
1.13.3	mutatorX_areaDamageMultiplier	float	Same as directDamageMultiplier but for areaDamage. Defaults to 1	mutatorX_areaDamageMultiplier: 1.2
1.13.3	mutatorX_changedExplodeEffect	effect	Change explode effect if this mutator is active. Eg make a bounce off amour effect. Helps to make the damage change more clear to players (Doesn't work with targetGround.)	mutatorX_changedExplodeEffect: CUSTOM:specialBlast
1.14	mutatorX_addResourcesDirectHit	resource	Add resource to all direct hit units. Warning: Be careful not be break units from other mods by adding random resources or energy to them that they don't expect.	mutatorX_addResourcesDirectHit: oil=1
1.14	mutatorX_addResourcesAreaHit	resource	Add resource to all area hit units. Warning: Be careful not be break units from other mods by adding random resources or energy to them that they don't expect.	mutatorX_addResourcesAreaHit: rust=4

Movement

Version	Code	Value Type	Description	Example
#====	#====		Movement	#====
	targetGround:	bool	Target ground, and don't home in on target. Note: only areaDamage is applied if targeting ground.	targetGround: true
0.00	targetGround_includeTargetHeight	bool	Default false. for area effect AA weapons	targetGround_includeTargetHeight: 40
1.14	targetGroundHeightOffset	float	Default 0. for shooting over or under a target. Useful for projectiles that split and rain down.	targetGroundHeightOffset: 10
	speed:	float	Projectile default travel speed	speed: 3
	targetSpeed:	float	Accelerate to this speed	targetSpeed: 5
1.12	targetSpeedAcceleration	float	Controls the speed rampup for targetSpeed	targetSpeedAcceleration: 0.3
	ballistic:	bool	Makes projectiles fly up into the air and come down, instead of going in a straight line	ballistic: true
	ballistic_delaymove_height:	float	Sets up to how high the projectile needs to be before moving normally	ballistic_delaymove_height: 20
	ballistic_height:	float	Sets the target height of the projectile	ballistic_height: 10
	targetGroundSpread:	float	Randomly makes the shot inaccurate by this amount. Also used by weapons like the flamethrower	targetGroundSpread: 20
	speedSpread:	float	Randomly change the starting projectile speed by this amount	speedSpread: 3
	instant	bool	Hit target instantly	instant: true
	instantReuseLast	bool	Recycles last projectile fired, only one projectile ever exists. Can turn lasers into beam weapons by using lower rate of fire and setting this to true	instantReuseLast: true
1.14	instantReuseLast_alsoChangeTurretAim	bool	Make turret's aim include last projectile's spread and sweep offsets, useful for beam weapons	instantReuseLast_alsoChangeTurretAim: true
1.14	instantReuseLast_keepAreaDamageList	bool	Default false. Keeping the list was the normal behaviour in 1.13 making area damage not apply a second time but this is not useful. Use this only if you want the old behaviour.	instantReuseLast_keepAreaDamageList: true
1.14	interceptProjectile_removeTargetLifeOnly	bool	Default false. When false projectiles are just removed. Could be true to make hit projectiles explode or split when hit	interceptProjectile_removeTargetLifeOnly: true
1.13	disableLeadTargeting	bool	Disable the lead targeting calculations when aiming at a moving target. Defaults false.	disableLeadTargeting: true
1.13	leadTargetingSpeedCalculation	float	The expected speed of this projectile for targetGround lead target calculation. Defaults to 'targetSpeed' if set otherwise 'speed'	leadTargetingSpeedCalculation: 2
1.13	initialUnguidedSpeedHeight		Sets vertical speed for projectiles with targetGround. Use gravity to make smooth arching projectiles. Better have gravity value slightly slower then this key to produce the arches.	initialUnguidedSpeedHeight: 30
1.13.3	gravity:	float	Controls the pull for projectiles that target ground. Use together with initialUnguidedSpeedHeight	gravity: 29
1.14	turnSpeed:	float	Limits the turn speed of a projectile, making them inaccurate even with directDamage. Zero value will make it act like if targetGround is set true, but air target friendly option.	turnSpeed: 0
1.14	wobbleAmplitude	float	How wide the projectile will wobble	wobbleAmplitude: 10
1.14	wobbleFrequency	float	How often the projectile will wobble	wobbleFrequency: 4
1.14	pushForce	float	Push (or pull with a negative value) the units that get hit. Divided by target mass	pushForce: 2
1.14	pushVelocity	float	Push (or pull with a negative value) the units that get hit. Ignores target mass	pushVelocity: 5
1.14	moveWithParent	bool	Move projectile as parent moves. Useful for beam effects that need to stick to source turret.	moveWithParent: true
1.14	sweepOffset	float	Useful for beam effects.	sweepOffset: 0.5
1.14	sweepOffsetFromTargetRadius	float	Add to sweep offset by factor of target's radius. 0.4 would be 40%.	sweepOffsetFromTargetRadius: 0.4
1.14	sweepSpeed	float	Useful for beam effects.	sweepSpeed: 1.5
	retargetingInFlight	bool	Can retarget a new target mid-flight, perfect for flak-style weapons and projectiles that collide	retargetingInFlight: true
1.14	retargetingInFlightSearchDelay	float/time	How long between searching for new targets. Default 5	retargetingInFlightSearchDelay: 1s
1.14	retargetingInFlightSearchRange	float	Range which targets are reselected. Default 120	retargetingInFlightSearchRange: 300
1.14	retargetingInFlightSearchLead	float	The lead of the projectile to try to hit the target. Default 15	retargetingInFlightSearchLead: 30
1.14	retargetingInFlightSearchOnlyTags	tag ref	Only retarget units with these tags	retargetingInFlightSearchOnlyTags: lightArmor

Graphics and effects

Version	Code	Value Type	Description	Example
#====	#====		Graphics and effects	#====
	color:	color	Recolors this projectile using a hex value.	color: #bebe50
1.13.2	invisible	bool	When true, the projectile is not rendered but still functional. Defaults: true.	invisible: true
	image:	file (image)	Use custom image. Overrides drawType and frame	image: bullet
	drawType	int	Built-in image to use. 0:projectiles.png 1:projectiles_large.png 2:projectiles2.png. Refer to end of sheet for the projectile images.	drawType:1
	drawSize:	float	Scale image. Defaults to 1	drawSize: 1.5
	frame	int	Built-in image frame to use, starts at zero.	frame: 0
	hitSound:	bool	Default true	hitSound: true
1.13	explodeEffect	effect ref list	Produces the specified effects upon explosion.	explodeEffect: smallExplosion, CUSTOM:myExplodeEffect
1.13	explodeEffectOnShield	effect ref list	Use this effect if shield is active on target	explodeEffectOnShield: CUSTOM:EMPwave
1.13	teamColorRatio	float	Mix 0-1 of team colour into color field	teamColorRatio: 1
1.14	teamColorRatio_sourceRatio	float	default is (1-teamColorRatio). Keep more of **color** when mixing. Note this might saturate colors.	teamColorRatio_sourceRatio: 1
1.13	drawUnderUnits	bool	If true, renders the projectile under units. Great for torpedoes	drawUnderUnits: true
1.13	effectOnCreate	effect ref list	Produces specified effects upon creation of the projectile.	effectOnCreate: CUSTOM:puff
1.13	shouldRevealFog	bool	Reveal fog to player on explode	shouldRevealFog: true
1.13	alwaysVisibleInFog	bool	Renders the projectile even when fog is present	alwaysVisibleInFog: false
1.13	nukeWeapon	bool	Shows on mini-map when fired. Some other side effects as well, like nuke explosion effect	nukeWeapon: true
	trailEffect	bool/effect	true for built-in defaults, but can also point to any custom effects	trailEffect: true, trailEffect: CUSTOM:rocketThrust
1.13	trailEffectRate	float	Defaults to 3	trailEffectRate: 4
	lightCastOnGround	bool	Renders light under the projectile	lightCastOnGround: true
	lightSize:	float	Sets the size of the light emitted by the projectile. 1 value = 1 tile.	lightSize: 1.5
	lightColor:	color	Sets the color of the light emitted by the projectile.	lightColor: #ffe92b
	targetHitEffect:	bool	Creates a large explosion and accompanying sound on hit (only cosmetic)	targetHitEffect: true
	lightingEffect:	bool	Draw as lighting works best with instant true	lightingEffect: true
	laserEffect:	bool	Draw as laser works best with instant true	laserEffect: true
1.14	beamImage	file (image)	Image to use for beam and laser effect type projectiles. Image is repeated vertically depending on the distance to target. Beam image should be 20 pixels or longer. Width does not matter, only the height.	beamImage: beam.png
1.14	beamImageOffsetRate	float	Sets how fast the beam image moves towards or away from the target	beamImageOffsetRate: 1
1.14	beamImageStart	file (image)	Sprite for the origin point of the custom beam	beamImageStart: beamStart.png
1.14	beamImageStartRotated	bool	Defaults false. True to rotate with turret angle	beamImageStartRotated: true
1.14	beamImageEnd	file (image)	Sprite for the end point of the custom beam	beamImageEnd: beamEnd.png
1.14	beamImageEndRotated	bool	Defaults false	beamImageEndRotated: true

[movement]

These are traits the unit has as far as movement goes, such as rotation and acceleration speed

Code	Value Type	Description	Example
movementType:	string	Defines what kind of terrain the unit will be able to move, along with other properties. Can use only one: NONE, LAND, AIR, WATER, HOVER, BUILDING, OVER_CLIFF, OVER_CLIFF_WATER	movementType: LAND
slowDeathFall:	bool	Used with large aircraft. Makes the unit fall slowly while maintaining its speed at the time of death.	slowDeathFall: true
moveSpeed:	float	Maximum movement speed of the unit. A value of 1 means (1 pixel * 60) per second (or 3 tiles per second)	moveSpeed: 1.2
moveAccelerationSpeed:	float	Defines how fast units accelerate to max speed.	moveAccelerationSpeed: 0.07
moveDecelerationSpeed:	float	Don't make this too low or units will have trouble stopping at waypoints	moveDecelerationSpeed: 0.17
reverseSpeedPercentage:	float	0.6 default. Over 0.4 will reverse for short distances (at 40% speed). If set to 1 will drive in reverse same as forwards. Affects minor turning.	reverseSpeedPercentage: 0
landOnGround:	bool	Should flying unit land when idle.	landOnGround: false
targetHeight:	float	Defaults to 0 but if AIR movementType default is 35	targetHeight: 25
targetHeightDrift:	float	Smooth animated height change. Defaults to 0 but if AIR movementType default is 1.5	targetHeightDrift: 1
startingHeightOffset:	float	Sets the initial height on spawn. Defaults at 0.	startingHeightOffset: 40
heightChangeRate: (0.00)	float	Rate at which the unit changes height, either from converting or drifting	heightChangeRate: 3
fallingAcceleration:	float	The acceleration in which a unit drops	fallingAcceleration
fallingAccelerationDead: (1.14)	float	The acceleration but when destroyed	fallingAccelerationDead
maxTurnSpeed:	float	Sets the top turning speed of a unit	maxTurnSpeed: 4
turnAcceleration:	float	Defines how fast units accelerate to max turn speed.	turnAcceleration: 1
moveSlidingMode:	bool	Makes the unit slide when moveDecelerationSpeed is lower, making them drift and feel natural	moveSlidingMode: true
moveIgnoringBody:	bool	Allows the unit to move without fully turning in the direction its moving, useful for ships and air units	moveIgnoringBody: true
moveSlidingDir:	int	Sets direction when sliding	moveSlidingDir: 180
joinsGroupFormations:	bool	Defaults to true. Changing not recommended. When false, the unit will directly move to the assigned waypoint without taking space consideration from neighboring units.	joinsGroupFormations: false

[ai]

This determines what the AI will use the unit for, does not effect player

Code	Value Type	Description	Example
useAsBuilder:	bool	Set to true if can build or repair buildings. Defaults to [core] isBuilder.	useAsBuilder:
useAsTransport	bool	Defaults to true if unit can transport units	useAsTransport
useAsHarvester	bool	Defaults to true if unit can reclaim resources	useAsHarvester
disableUse	bool	Disallow AI building this unit or building	disableUse:
ai_upgradePriority	float	Defaults to 0.06. Set between 0-1, higher means AI is more likely to upgrade this unit before others	ai_upgradePriority

Buildings

Version	Code	Value Type	Description	Example
#====	#====		Buildings	#====
	maxGlobal:	int	Maximum amount allowed for a specific structure for an AI team per map	maxGlobal: 50
	maxEachBase:	int	Maximum allowed for a specific structure for an AI team per "base", usually around an extractor or spawn point. To check for an AI base in sandbox, enable Debug Mode and Click Shift + F3	maxEachBase: 10
	buildPriority:	float	0-1. AI uses 0.9 for first land factory, 0.48 for air factory, 0.47 for first turret.	buildPriority: 1
	noneInBaseExtraPriority:	float	Adds to buildPriority if this unit doesn't exist in the Als base	noneInBaseExtraPriority: 2
	noneGlobalExtraPriority:	float	Acms to buildPriority if this unit doesn't exist in the any where on the map	noneGlobalExtraPriority: 4
	recommendedEachBaseNum	float	Defaults to 0	recommendedEachBaseNum: 2
	recommendedEachBasePriorityIfUnmet	float	Defaults to 0.5. Overrides buildPriority if recommended in base is too low.	recommendedEachBasePriorityIfUnmet: 4
	upgradedFrom:	string	Create link to another unit to preserve max counts for upgraded and non-upgraded types in same base.	upgradedFrom: builderLevel1
1.14	notPassivelyTargetedByOtherUnits	bool	Useful for walls, etc	notPassivelyTargetedByOtherUnits: true
1.14	lowPriorityTargetForOtherUnits	bool	Useful for units that cannot attack back. Eg walls	lowPriorityTargetForOtherUnits: true
1.13.3	whenUsingAsHarvester_recommendedInEachBase	float	Assigns unit of the same name with certain amount on an AI base	whenUsingAsHarvester_recommendedInEachBase: 3
1.13.3	whenUsingAsHarvester_recommendedGlobal	int	Assigns unit of the same name with certain amount on map	whenUsingAsHarvester_recommendedGlobal: 20
1.13.3	whenUsingAsHarvester_includeOtherHarvesterCounts	bool	Includes all harvester types on a single counter	whenUsingAsHarvester_includeOtherHarvesterCounts: true
1.13.3	onlyUseAsHarvester_ifBaseHasUnitTagged	string	Only sets the harvester unit as a harvester if an AI base has a unit with particular tag(s)	onlyUseAsHarvester_ifBaseHasUnitTagged:

[leg_#] / [arm_#]

Legs can move around when unit moves, Arms need an animation or convert

Version	Code	Value Type	Description	Example
	x:	float	Sets position of the foot on the X axis.	x: 10
	y:	float	Sets position of the foot on the Y axis	y: 20
	copyFrom:	int	Copy from another leg. Useful to only need to set leg values once	copyFrom: 1
	attach_x:	float	Sets the leg's attach point on the X axis.	attach_x: 10
	attach_y:	float	Sets the leg's attach point on the Y axis.	attach_y: 0
	rotateSpeed:	float	Sets the leg's rotation speed on movement	rotateSpeed: 2
	endDirOffset:	int	Target foot/end rotation relative to body	endDirOffset: 45
	lockMovement:	bool	Lock to unit body. Useful if walking unit converted to a flying unit	lockMovement: true
	heightSpeed:	float	Sets how fast the leg rises while walking	heightSpeed: 2
	moveSpeed:	float	Sets how fast the leg moves while walking	moveSpeed: 3
	moveWarmUp:	float	Delay before the leg moved	moveWarmUp: 2
	holdDistMin:	float	Defaults to 7. Reposition leg at this distance if neighbor legs are not already repositioning.	holdDistMin: 10
	holdDistMin_maxMovingLegs	float	Defaults to 16. Force reposition of leg at this distance. Repositions leg at this distance even if a neighboring leg is moving	holdDistMin: 20
	hold_moveOnlyIfFurthest	bool	Sets maximum amount of legs to check before applying distance holding	holdDistMin_maxMovingLegs: 4
	holdDistMin_checkNeighbours	bool	Defaults to true. When true, starting moving leg only if it is currently the furthest leg from where it should be	hold_moveOnlyIfFurthest: true
	hardLimit:	float	Checks neighbors before applying distance holding	holdDistMin_checkNeighbour: true
	estimatingPositionMultiplier	float	Defaults to 50. Force leg to never go this far. Better to not be reached.	hardLimit: 60
			defaults to 1. Predicts were unit will be for leg placement based on unit speed.	estimatingPositionMultiplier: 2

Graphics and effects

Version	Code	Value Type	Description	Example
#====	#====		Graphics and effects	#====
	hidden:	logic boolean	When true, hides the arm/leg	hidden: true
1.13	image_end	file (image)	Sets the main arm and sprite	image_end: rotor.png
1.13	image_end_shadow	file (image)	Sets the shadow for the arm	image_end_shadow: AUTO
1.13	image_end_teamColors	bool	When true, the arm end will apply team colors	image_end_teamColors: true
1.12	image_foot	file (image)	same as image_end, but acts as the foot for the leg	image_foot: foot.png
1.12	image_foot_shadow	file (image)	Sets the shadow for the foot	image_foot_shadow: AUTO

Ver	Code	Value Type	Description	Example
1.13	image_middle	file (image)	Sets the arm image	image_middle: NONE
	image_leg	file (image)	Sets the leg image	image_leg: legR2.png
1.12	draw_foot_on_top	bool	Renders foot above leg	draw_foot_on_top: true
	drawOverBody	bool	Draw over body	drawOverBody: false
	drawUnderAllUnits	bool	Draw over all units	drawUnderAllUnits: true
	drawDirOffset	float	Rotates the footprint sprite to a specified direction	drawDirOffset: 73
	dust_effect:	float	Spawns dust particles on each step	dust_effect: true
	spinRate	float	Makes arm/leg spin, like idleSpin for turrets. Great for helicopters	spinRate: 2
	flavourOppositeSideNeighbours	bool	calculate neighbours with X 10 times closer than Y	flavourOppositeSideNeighbours: true
	drawLegWhenZoomedOut	bool	For performance, defaults changes based on unit size	drawLegWhenZoomedOut: false
	drawFootWhenZoomedOut	bool	For performance, defaults changes based on unit size	drawFootWhenZoomedOut: false
	liftingHeightOffset	int	Often used with decals	liftingHeightOffset: 20
1.15p9	targetHeight	int	Sets height of the leg	targetHeight: 10
	targetHeightRelative	bool	Apply height relative to unit's height	targetHeightRelative: true

[attachment_NAME] — Attachments are slots where other units can be positioned or carried

Ver	Code	Value Type	Description	Example
1.13.2	x	float	Sets the horizontal position of the attachment	x
1.13.2	y	float	Sets the vertical position of the attachment	y
1.13.3	height	float	Sets the elevation of the attachment	height
1.13.3	idleDir	float	Sets the direction of the attachment when idle	idleDir
1.14	idleDirReversing	int	Sets the direction of the attachment when the base unit is moving in reverse	idleDirReversing
1.13.3	isVisible	bool	Defaults to true. If false, the attachment is hidden	isVisible
1.13.3	onCreateSpawnUnitOf	unit ref	Upon spawning the unit, specified attached unit is also spawned as attachment	onCreateSpawnUnitOf
1.13.2	isUnselectable	bool	Defaults to false. When true, the player cannot click the attachment.	isUnselectable
	canAttack	bool	Defaults to true. Set to false to stop the attachment attacking.	canAttack
1.13.2	canBeAttackedAndDamaged	bool	When true, the attachment is vulnerable to attacks	canBeAttackedAndDamaged
1.13.2	deattachIfWantingToMove	bool	If the unit is ordered to move, it will detach. This includes waypoints from actions.	deattachIfWantingToMove
1.13.3	lockLegMovement	bool	Locks the leg movement while attached.	lockLegMovement
1.13.3	keepAliveWhenParentDies	bool	Defaults to false.	keepAliveWhenParentDies: true
1.13.3	setDrawLayerOnTop	bool	Renders the attachment above the base unit	setDrawLayerOnTop
1.13.3	setDrawLayerOnBottom	bool	Renders the attachment below the base unit	setDrawLayerOnBottom
1.13.3	addTransportedUnits	bool	Often used with transport units, when true, attaches one of the unit passenger to this attachment slot	addTransportedUnits
1.13.3	lockRotation	bool	When true, stops the attachment from rotating	lockRotation
1.13.3	rotateWithParent	bool	When true, the attachment rotates with the parent unit	rotateWithParent
1.13.3	resetRotationWhenNotAttacking	bool	Similar to shouldResetTurretsfor turrets.	resetRotationWhenNotAttacking
1.13.3	prioritizeParentsMainTarget	bool	It will priotize targeting the main target. Defaults to true.	prioritizeParentsMainTarget
1.13.3	alwaysAllowedToAttackParentsMainTarget	bool	Will always attack the parents main target	alwaysAllowedToAttackParentsMainTarget
	onParentTeamChangeKeepCurrentTeam	bool	Defaults false. If true attached units are not converted when parent changes team. Eg from [projectile] convertAllToSourceTeam	onParentTeamChangeKeepCurrentTeam
1.14	onConvertKeepExistingUnitInSameSlot	bool	When true, the attachment is retained on the same attachment slot when the parent is converted to another unit	onConvertKeepExistingUnitInSameSlot
1.14	unloadInCurrentPosition	bool	Defaults false. If true transported attached units are kept current attached location when unloading	unloadInCurrentPosition
1.14	keepWaypointsNeedingMovement	bool	Defaults false. If true attached units keep waypoints with movement even while they cannot move. Useful if they will be automatically deattached soon.	keepWaypointsNeedingMovement
1.14	smoothlyBlendPositionWhenExistingUnitAdded	bool		smoothlyBlendPositionWhenExistingUnitAdded
1.14	showAllActionsFrom	LogicBoolean	Show all actions of the units attached in the parent unit list when selected	showAllActionsFrom
1.14	createIncompleteIfParentIs	bool	If parent hasn't been built, create attachment with the same built value. Links built values till attachment is complete. Useful for buildings built with mass.	createIncompleteIfParentIs:
1.14	redirectDamageToParent	bool	Redirects damage done to the attachment to the parent instead of damaging itself directly	redirectDamageToParent:
1.14	redirectDamageToParent_shieldOnly	bool	When enemies attack the attachment, all damage are redirected to the parents shield	redirectDamageToParent_shieldOnly:

[action_NAME] / [hiddenAction_NAME] — Actions that can dynamically cause changes to units and resources

Ver	Code	Value Type	Description	Example
	text	string	Labels the action. Supports dynamic text	text: Catch Fish, text: Fire: %(self.resource.ammo)
	textPostFix:	string	Text shown as suffix, useful with textAddUnitName to create text UI	text: [textPostFix: [textAddUnitName: unitRef self.attachment(slot="${slotId}")
	text_(LANG)	string	Alternative text for different language. Use ISO 639-1 Language code on the (LANG) prefix. Supports dynamic text	text_es: Fuego %(self.resource.ammo)
	description	string	A display text when you select your unit's action, used to explain it's purpose. Supports dynamic text	description: Fires shot on target area
	description_(LANG)	string	Alternative description for different language. Use ISO 639-1 Language code on the (LANG) prefix. Supports dynamic text	description_es: Disparos en el área objetivo
	displayType	list	Affects how the action button and text is displayed. Options: none, rally, upgrade, queueUnit, building, action, infoOnly, infoOnlyNoBox, infoOnlyStockpile	displayType: upgrade
	displayRemainingStockpile	bool	Queue is shown as number of times action can be triggered based on price. Use dynamic text so text as alternative.	displayRemainingStockpile: true
	pos	float	Order action appears in UI. Merges with positions from canBuild buttons	pos: 1
	iconImage	file (image)	Sets a thumbnail image for the action button	iconImage: fireShot.png
1.14	iconExtraImage	file (image)	Drawn over top of icon image. Useful for upgrade icons, etc	iconExtraImage: fireShotNo.png
1.14	iconExtraColor	colour	Defaults to #64FFFFFF	iconExtraColor: #ff0000
1.14	iconExtraIsVisible	LogicBoolean	When the condition is met, the extras for icon is visible.	iconExtraIsVisible: if self.resource.ammo < 1
1.13.3	unitShownInUI	unitRefOnlyType	Display this unit. (as if this action built this unit)	eg: unitShownInUI: unitRef self.transporting(slot=0) or unitShownInUI: heavyTank
	guiBuildUnit	unitRefOnlyType	Uses the UI similar when building structures. An alternative to fireTurretX on some cases	guiBuildUnit: placeholderUnit
1.14	setBuilt	float	Designates how built the unit is from a percentage of 0-100% with a number between 0 and 1.	setBuilt: 0.5
1.15p9	tags	tags	To be used with the withTag parameter for self.queueSize (withTag=x) and queueItemAdded and queueItemCanceled events	tags: actionFire
1.15p11	id	string	Allow same/equivalent actions to be more easily connected when converting between units, to preserve queues, cailldowns, etc. (Normally action order is used but can be unstable.)	id: fireShot

#===== Unit Reference - Dynamically parts from already existing units, useful ie inAlsoViewableByEnemies

Ver	Code	Value Type	Description	Example
1.14	textAddUnitName	unitRefOnlyType	Add this unit's name to this action's text	eg textAddUnitName: unitRef self.attachment(slot="1")
1.14	descriptionAddFromUnit	unitRefOnlyType	Add this unit's description to this action's description	descriptionAddFromUnit: builder
1.14	descriptionAddUnitStats:	unitRefOnlyType	Add this unit's stats (eg HP, energy, resources) to this action's description	descriptionAddUnitStats: unitRef self.parent()
1.14	unitShownInUIWithHpBar	bool	default true. Only used when unitShownInUI is a unitRef	unitShownInUIWithHpBar: false
1.14	unitShownInUIWithProgressBar	bool	default true. Only used when unitShownInUI is a unitRef. Replaces HP bar if active	unitShownInUIWithProgressBar: false

#===== Requirements for player/AI to use in UI

Ver	Code	Value Type	Description	Example
1.14	alwaysSinglePress	bool	Defaults false. When true no confirmation needed on mobile, when used with canPlayerCancel false and allowMultipleInQueue false will also hide the queue interface.	alwaysSinglePress: true
	price	resources	The price of your action for the unit. Disables action if not available. Defaults to credits if unlabelled	price: credits=5, energy=5, hp=100, shield=5, ammo=1
	isActive	LogicBoolean	Defaults true. If false then action is disabled and shown in red in UI.	isActive: true
	isVisible	LogicBoolean	Defaults false. If false action is hidden from UI and disabled.	isVisible: true
	isLocked	LogicBoolean	Defaults false. If true action is disabled, and a lock icon is shown. Mostly used for no nuke game modes	isLocked: if self.resource.ammo < 1
	isLockedMessage	LocaleString	Shows the message when the isLocked's condition is met	isLockedMessage: Not enough ammunition
1.13.3	isLockedAlt	LogicBoolean	Another reason for the to be locked. Can just use OR on isLocked, but this allows a different message to be shown	isLockedAlt: if numberOfUnitsInGame(withTag="factory") < 2
1.13.3	isLockedAltMessage	LocaleString	Message for isLockedAlt	isLockedAltMessage: Not enough factories!
1.13.3	isLockedAlt2	LogicBoolean	Second isLocked alternative	isLockedAlt2: if nearestUnit(withTag="explosive", withinRange="500", relation="own").hp() < 10
1.13.3	isLockedAlt2Message	LocaleString	Message for isLockedAlt2	isLockedAlt2Message: Explosive stockpile is heavily damaged.
	allowMultipleInQueue	bool	When false makes it so only one action can be queued of this type (useful for keeping actions with conditions from being spammed)	allowMultipleInQueue: true
	onlyOneUnitAtATime	bool	When action is picked in UI, only one unit selected will get this action. Defaults to false.	onlyOneUnitAtATime: true
1.13.3	isGuiBlinking	LogicBoolean	Flashes in UI to draw attention to it. Might be annoying if used often, recommended only for temporarily states/messages	isGuiBlinking: true
1.14	isAlsoViewableByAllies	bool	Allows ally players to see actions from this unit, useful for showing stats to other players (eg missile count, items collected)	isAlsoViewableByAllies: true
1.14	isAlsoViewableByEnemies	bool	Allows enemy players to see actions from this unit, useful for showing stats to other players (eg missile count, items collected)	isAlsoViewableByEnemies: true

#===== AI - How the AI uses this action

Ver	Code	Value Type	Description	Example
1.13.3	ai_isHighPriority	LogicBoolean	Use this for faction selection actions or other high priority actions such as building high priority units	ai_isHighPriority: true
	ai_isDisabled	bool	Defaults false. Stop AI using this action. (Note when ai_isHighPriority is true this might be ignored)	ai_isDisabled: true
1.13.3	ai_considerSameAsBuilding	bool	Be careful with	ai_considerSameAsBuilding: true

#===== Triggers - These skip the queue and do not use price. Ignores isLocked, buildTime, etc (Use 2 actions and alsoQueueAction to automatically add an action to the queue)

Ver	Code	Value Type	Description	Example
1.13.3 - 1.15 p9	autoTriggerOnEvent	event	Action will be triggered when an event is happening in a particular unit. Options: created, completeInArchive, destroyed, killedAnyUnit, queueUnitFinished, queueItemAdded(withActionTag="#"), queueItemCanceled(withActionTag="#"), teleported, touchTargetSuccess, newWaypointGivenByPlayer, teamChanged, transportingNewUnit, transportUnloaded(OrRemovedUnit, tookDamage(withTag="#"), newMessage(withTag="#"), enteredTransport, leftTransport, attachmentRemoved, withTag parameter for tookDamage uses tag from projectile and it is optional. withTag parameter for newMessage uses message tags.	autoTriggerOnEvent: queueItemCanceled(withActionTag="actionFire")
1.14	autoTriggerOnEventRecursionLimit	int	Defaults to 1. Prevents loops, useful with triggering itself so it doesn't infinitely triggers, good for repeating action effect on event	autoTriggerOnEventRecursionLimit: 4
	autoTrigger	LogicBoolean	When true triggers the effects of this action instantly (ignoring price, isActive, isVisible, buildSpeed, etc)	autoTrigger: if self.overWater(), autoTrigger: if self.customTimer(laterThanSeconds=5)
	autoTriggerCheckRate	enum	options: everyFrame (default), everyXFrames, everyXFrames. This overrides autoTriggerCheckRate set on [core] Note: all triggers regardless of check rate are checked when first created and after an auto trigger cooldown. Note: Adding (useQueueTriggerCheckRate: everyXFrames to all units template could have a large performance boost for mods with complex autoTriggers	autoTriggerCheckRate: every6Frames

#===== While action is queued

Ver	Code	Value Type	Description	Example
	buildSpeed	time	Sets how fast the action has to be queued before doing the action behaviors	buildSpeed: 5s
	highPriorityQueue	bool	Defaults to false. If true this action skips all other low priority actions in queue. Useful for fireTurret actions	highPriorityQueue: true
	canPlayerCancel	bool	Defaults to true. When false, players cannot cancel this particular action.	canPlayerCancel: false
	whenBuilding_cannotMove	bool	Stops unit moving while action is being applied. Useful for deploy like actions.	whenBuilding_cannotMove: true
	whenBuilding_playAnimation	animation ref	Plays a specified animation while the action is queued.	whenBuilding_playAnimation: firePrepare
	whenBuilding_rotateTo	float	Rotate unit body to this direction when action is in active queue.	whenBuilding_rotateTo: 45
	whenBuilding_rotateTo_orBackwards	bool	If true allow rotation in 180 degrees from whenBuilding_rotateTo when this is a smaller angle	whenBuilding_rotateTo_orBackwards: true
	whenBuilding_rotateTo_waitTillRotated	bool	Pause action queue till rotation is finished	whenBuilding_rotateTo_waitTillRotated: true
	whenBuilding_temporarilyConvertTo	unit ref	Convert to another unit while action is in active queue. Note: actions from the original unit will be kept	whenBuilding_temporarilyConvertTo: cannon_uberState
1.14	whenBuilding_temporarilyConvertTo_keepFields	fields	Don't change these fields when using whenBuilding_temporarilyConvertTo (both to and from)	whenBuilding_temporarilyConvertTo_keepFields: maxHp, maxEnergy, moveSpeed
	whenBuilding_triggerAction	action ref	While action is queued, another action is triggered	whenBuilding_triggerAction: spawnMinions
1.14	whenBuilding_rotateTo_aimAtActionTarget	bool	While action is queued, the unit is rotated to the target. Often used with fireTurretX actions	whenBuilding_rotateTo_aimAtActionTarget: true
1.14	whenBuilding_rotateTo_rotateTurretX	turret ref	While action is queued, a specified turret is aimed at the target. Often used with fireTurretX actions	whenBuilding_rotateTo_rotateTurretX: cannon
	spawnEffectsOnQueue	effect ref	Effects to spawn at unit when action is first added to queue	spawnEffectsOnQueue: CUSTOM:steam
	playSoundToPlayerOnQueue	sound ref	Global sound to play to unit's player only when action is first added to queue	playSoundToPlayerOnQueue: eva_building.ogg

#===== Misc outcomes / Results (What happens) (Note: Must be at least one outcome for an action to show)

Ver	Code	Value Type	Description	Example
	requireConditional	LogicBoolean	Skip all effects of this action if this evaluates to false	requireConditional: if self.resource.mass < 300
1.13.3	convertTo	unit ref	Convert your unit into another unit, properties are preserved	convertTo: fishLevel2
1.14	convertTo_keepCurrentTags	bool	Keep current and temporarily tags and ignore default tags on convertTo target.	convertTo_keepCurrentTags: true
1.14	convertTo_keepCurrentFields	fields	Don't change these fields when converting, useful with setUnitStats (Allowed fields: maxHp, maxShield, shieldRegen, maxEnergy, armour, mass, shootDelayMultiplier, moveSpeed, maxRotateRange.)	convertTo_keepCurrentFields: armour, maxEnergy, moveSpeed
	addEnergy	addEnergy	Adds energy to unit. Has no effect unless energyMax is set. (Same as addResources: energy=X)	addEnergy: 10
	addResources	resources	Add these resources when action finishes.	addResources: credits=5, energy=-5, hp=-100, shield=5, ammo=1
	addResourcesScaledByAiHandicaps	resources	Same as addResources, but increased or decreased depending on AI difficulty level	addResourcesScaledByAiHandicaps: credits=5, hp=-100
1.14	addResourcesWithLogic	dynamic resources	Like addResources but allows logic to be used for the resource amounts	addResourcesWithLogic: hp = select(self.parent.energy>5, 10, 20)
1.14	setResourcesWithLogic	dynamic resources	Sets target resources to this value instead of adding. Be careful with global resources.	setResourcesWithLogic: hp=self.parent.hp - 10, energy = self.energy / 2
	deleteSelf	bool	Remove self with no explosions or sounds	deleteSelf: true
	resetCustomTimer	LogicBoolean	Reset timer used with self.customTimer()	resetCustomTimer: true
1.13.2	setBodyRotation	int	Rotates the unit to a particular direction. Supports dynamic values	setBodyRotation: 270, setBodyRotation: directionBetween(self, customTarget1)
1.14	setUnitStats	fields values	Allows changing of a select number of fields dynamically without converting. Supports +/-/=/*=, with dynamic maths/logic. Changeable fields: maxHp, hp, maxShield, shield, shieldRegen, maxEnergy, energy, armour, mass, shootDelayMultiplier, shootDamageMultiplier, moveSpeed, maxTurnSpeed, maxAttackRange, fogOfWarSightRange, nanoRange, selfRegenRate	setUnitStats: maxShield += 20, moveSpeed += 0.1
1.14	resetUnitStats	boolean	Reset changes made by setUnitStats to base values	resetUnitStats: true
1.14	setUnitMemory	key value pairs	Change this unit's memory, values can be set with logic. Memory must first be defined with defineUnitMemory	setUnitMemory: ""; isAlone="+memory.customText+"hello"; nukeActive=true; nextTarget=self.attacking.nearestUnit(withinRange=300, withTag="x", relation=enemy")
1.15p11			NOTE: In setting array memories, it is important to put the array index (supports dynamic values)	
1.14	setHeight	logicNumber	Change unit height based on this height value	setHeight: parent.height+5
1.14	setCustomTarget1	unit ref	Used for unit linking without requiring unit memories or markers. It is the unit that built this by default.	setCustomTarget1: self.parent
1.14	setCustomTarget2	unit ref	Like setCustomTarget1 and with the same use. It is none by default.	setCustomTarget2: self.DamagedBy
1.13.3	sendMessageTo	unit ref	Sends a message to a targeted unit	sendMessageTo: unitref nearestUnit(withinRange=100, withTag="fish")
1.14	sendMessageWithTags	Message Tag	Useful for message detection in an autoTriggerOnEvent event, tag auto trigger onEvent newMessage(withTag="xyz")	sendMessageWithTags: hitZone
1.14	sendMessageWithData	key-value	The data that will be sent to the targeted unit. Allows multiple key-value pairs with any dynamic data type, use eventData() to read this data in the event	sendMessageWithData: fish="nice", cat=self.active(withpointTarget, amount=4, xyz=memory.something
1.14	refundAllQueuedItems	boolean	Refunds the exact price in the queue of a specific action; includes set flags in price	refundAllQueuedItems: true
1.14	removeAllQueuedItemsWithoutRefund	boolean	Similar to refundAllQueuedItems, but does not refund	removeAllQueuedItemsWithoutRefund: true

#===== Outcome - Chaining Actions

Ver	Code	Value Type	Description	Example
1.14	alsoTriggerAction	action refs	Trigger to results of another action as well. Ignores action's requirements.	alsoTriggerAction: addCredits, playSound

Version	Code	Value Type	Description	Example
	alsoQueueAction	action refs	Adds another action into the normal unit's queue. Ignores action's requirements	alsoQueueAction: spawnMinions
	alsoTriggerOrQueueActionConditional	LogicBoolean	Defaults true. alsoTriggerAction and alsoQueueAction are ignored if this works out to be false.	alsoTriggerOrQueueActionConditional: false
1.15p9	alsoTriggerOrQueueActionWithTarget	unitref	Changes the target of the triggered action, normally defaults to the current action target. Effects things like fireTurretXAtGround, spawnUnits, thisActionTarget), etc	alsoTriggerOrQueueActionWithTarget: lastDamagedBy
1.15p11	alsoTriggerActionRepeat	logicNumber	Repeats the alsoTriggerAction call, thisActionIndex changed on each repeat - Useful to create loops or work with arrays	alsoTriggerActionRepeat: 5
	#====		**Outcome - Sounds**	
	playSoundAtUnit	sound ref	Local sound to play when action finishes	playSoundAtUnit: engineStart.ogg
	playSoundGlobally	sound ref	Global sound to play to all players in game	playSoundGlobally: hornWarn.ogg
	playSoundToPlayer	sound ref	Global sound to play to unit's player only	playSoundToPlayer: confirm.wav
	#====		**Outcome - Fire projectile from turret**	
	fireTurretXAtGround	turret ref	When action finishes fire target turret at point on ground, bypasses canShoot rules in turret	fireTurretXAtGround: nukeSilo
	fireTurretXAtGround_withOffset	point	If not set player targets the ground with GUI, if a point is set this step is skipped	fireTurretXAtGround_withOffset: 0,0
	fireTurretXAtGround_withProjectile	projectile ref	Used with fireTurretXAtGround. Defaults to target turret's normal projectile.	fireTurretXAtGround_withProjectile: nuke
1.14	fireTurretXAtGround_withTarget	unitref	Fires a turret aimed at the location of the indicated unit or marker	fireTurretXAtGround_withTarget: lastDamagedBy
1.13.3	fireTurretXAtGround_count		Number of projectiles to fire. Defaults to 1	fireTurretXAtGround_count: 3
1.12.3	fireTurretXAtGround_onlyOverPassableTileOf		Only allow tiles crossable by this movement type to be selected (e.g., LAND,BUILDING,WATER,HOVER)	fireTurretXAtGround_onlyOverPassableTileOf: HOVER
1.15p11	fireTurretXAtGround_showGuideDecals	decal refs	Draws decals at the target location. Recommend setting up decals with layer: inactive	fireTurretXAtGround_showGuideDecals: strikeZone300
	#====		**Outcome - Spawning**	
			NOTE: There is a section for all possible modifications to spawnUnit, type "spawnUnits:LIST" in the Search function of the spreadsheet	
1.14	spawnUnits	unitref	Spawn units at action's target. See 'Spawn units line' section in this doc for details.	eg: spawnUnits: heavyTank, tank*3, hoverTank(offsetX=10)
1.13.3	produceUnits	unitref	Like spawnUnits but unit exits as if it was produced normally, and gets a move away waypoint	produceUnits: builder*4, plasmaTank*8
	spawnEffects	effect ref	Effects to spawn at unit	spawnEffects: CUSTOM:puff
	#====		**Outcome - Position**	
1.14	offsetSelfAbsolute:	pointSd	Offsets unit position absolutely by this point. Format: [x,y, height]	offsetSelfAbsolute: 0, 0, 40
1.15p11	teleportTo	unit refmarker	Changes unit position to this position. Great alternative to fireTurretX with teleporting projectile	teleportTo: memory.lastLocation
	#====		**Outcome - Transport Changes**	
	addUnitsIntoTransport	unitTypes	Creates and add units into transport, use self transportingCount) to check for space before adding	addUnitsIntoTransport: tank*3, heavyTank(neutralTeam=true)
	transportTargetNow	unit ref	(Currently broken in 1.15 - don't use) Instantly tries to transports existing units on the map into this transport. Might fail if rules don't allow this unit to be transported.	
	deleteNumUnitsFromTransport	int	Removes a specified amount of cargo units	deleteNumUnitsFromTransport: 2
1.13.2	deleteNumUnitsFromTransport_onlyWithTags	string(s)	Removes a specified amount of cargo units, but only those with specified tags	deleteNumUnitsFromTransport_onlyWithTags: cheapStuff
1.13.3	startUnloadingTransport	bool	Unloads all cargo units normally	startUnloadingTransport: true
1.13.3	forceUnloadTransportNow	bool	For unload all units, or slot targeted by forceUnloadTransportNow_onlyOnSlot. Unloads even if no space or overwater, etc	forceUnloadTransportNow: true
1.14	forceUnloadTransportNow_onlyOnSlot	int	Focuses the force unload to a specific slot	forceUnloadTransportNow_onlyOnSlot: drop1
	#====		**Outcome - Waypoint Changes**	
1.13.2	clearAllWaypoints	bool	Clears all waypoints, be careful not to annoy players by removing their orders, prepending waypoints is often better	clearAllWaypoints: true
1.13.2	clearActiveWaypoint	bool	Clears only the current waypoint	clearActiveWaypoint: true
	addWaypoint_type	enum	Adds a waypoint with a specific purpose. Options: move, attackMove, guard, loadInto, loadUp, attack, reclaim, repair, touchTarget, build, follow, selfPassiveTarget	addWaypoint_type: move
1.13.2	addWaypoint_unitType	unitTypes	Only for use with addWaypoint_type:build	addWaypoint_unitType: turret
1.13.3	addWaypoint_prepend	bool	Add to the start of the waypoint queue or the end	addWaypoint_prepend: false
1.13.3	addWaypoint_triggerActionIfFailed	actions	If target_nearestUnit fails to find a match so waypoint cannot be added then trigger this action	addWaypoint_triggerActionIfFailed: retreat
1.13.4	addWaypoint_triggerActionIfMatched	actions	Triggers an action if a waypoint is legal	addWaypoint_triggerActionIfMatched: VIP_target
1.13.3	addWaypoint_maxTime	time	Automatically remove this waypoint if it has been active for longer than this time.	addWaypoint_maxTime: 20s
1.13.3	addWaypoint_target_nearestUnit_tagged	tags	Pulls the waypoint to a nearest unit with specific tags	addWaypoint_target_nearestUnit_tagged: assault, mechanized
1.13.3	addWaypoint_target_nearestUnit_team	relation	Pulls the waypoint to a nearest unit with a specific relation to the player's unit. Options: own, neutral, allyNotOwn, ally, enemy, any, notOwn	addWaypoint_target_nearestUnit_team: ally
1.13.3	addWaypoint_target_nearestUnit_maxRange	float	Pulls the waypoint to a nearest unit within the maximum range specified.	addWaypoint_target_nearestUnit_maxRange: 2000
1.13.3	addWaypoint_target_mapMustBeReachable	float	Pulls the waypoint only if the waypoint target is reachable by the unit (e.g., if a unit cannot cross two islands, the waypoint is not placed.)	addWaypoint_target_mapMustBeReachable: true
1.13.2	addWaypoint_position_fromReference	unit ref	Pulls the waypoint to a marker or a reference unit	addWaypoint_target_fromReference: self.memory.lastDock
1.13.2	addWaypoint_position_offsetFromSelf	point	Pulls the waypoint to an absolute coordinate	addWaypoint_position_offsetFromSelf
1.13.3	addWaypoint_position_fromAction	bool	Pulls the waypoint taken from an action that triggered this action	addWaypoint_position_fromAction: fireShot
1.13.3	addWaypoint_position_randomOffsetFromSelf	point	Pulls the waypoint to a random relative coordinate	addWaypoint_position_randomOffsetFromSelf
1.14	addWaypoint_position_relativeOffsetFromSelf	point	Pulls the waypoint to a relative coordinate	addWaypoint_position_relativeOffsetFromSelf: 10, 20
1.14p8	addWaypoint_target_randomUnit_tagged	tags	Pulls the waypoint to a random unit with specific tags	addWaypoint_target_randomUnit_tagged: chargers, strikers
1.14p8	addWaypoint_target_randomUnit_team	relation	Pulls the waypoint to a random unit with a specific relation to the player's unit. Options: own, neutral, allyNotOwn, ally, enemy, any, notOwn	addWaypoint_target_randomUnit_team: ally
1.14p8	addWaypoint_target_randomUnit_maxRange	float	Pulls the waypoint to a random unit within the maximum range specified.	addWaypoint_target_randomUnit_maxRange: 1000
	#====		**Outcome - Cooldown**	
1.14	addAllActionCooldownsTime	time	Adds cooldown on all actions, including canBuilds	addAllActionCooldownsTime: 4s
1.14	addActionCooldownTime	time	Player cannot use action again for this amount of time	addActionCooldownTime: 20s
1.14	addActionCooldownApplyToActions	action ids	Sets addActionCooldownTime's target. Defaults to this action.	addActionCooldownApplyToActions: retreat, transformFins
1.14	clearAllActionCooldowns	bool	Removes all existing cooldowns on all buttons	clearAllActionCooldowns: true
	#====		**Outcome - Animation**	
1.13.3	playAnimation	animation id	Plays animation when the action is triggered	playAnimation: engageThrusters
1.13.2	playAnimationIfNotPlaying	bool	Don't restart animation if this animation is already playing	playAnimationIfNotPlaying: true
1.13.3	finishPlayingLastAnimation	bool	Finish last animation, including blend out	finishPlayingLastAnimation: true
1.13.3	stopLastAnimation	bool	Stop last animation, skipping blend out	stopLastAnimation: true
1.13.3	switchToNeutralTeam	boolean	Change team to neutral. This team is allied to all other teams. Will be captured by nearby units unless [core]stayNeutral:true is used	switchToNeutral: false
1.13.3	switchToAggressiveTeam	boolean	Change to a built-in team that is aggressive to all other teams. Does not get captured.	switchToAggressive: true
1.14	switchToTeam	logicNumber	Team id to switch to. Starts at 0. (but -1 for a neutral team, -2 for aggressive Team). Also supports dynamic values	switchToTeam: 2, switchToTeam: lastDamagedBy.teamId
	#====		**Outcome - Take Resources from other units**	
1.13.3	takeResources	customPrice	Resources to take (required to use take resources). And at-least 1 include key is needed.	takeResources: hp=5, gold=10
1.13.3	takeResources_includeUnitsInTransport	bool	Includes cargo units in taking resources	takeResources_includeUnitsInTransport: true
1.13.3	takeResources_includeParent	bool	Include attachment parent or transport parent	takeResources_includeParent: true
1.14	takeResources_includeReference	unit ref	Includes referenced units in taking resources	takeResources_includeReference: lastDamagedBy
1.13.3	takeResources_includeUnitsWithinRange	float	Includes all units within the specified range	takeResources_includeUnitsWithinRange: 300
1.13.3	takeResources_includeUnitsWithinRange_team	TeamRelation	Used with includeUnitsWithinRange, defaults to own. Can be: own/ally/allyNotOwn/enemy/neutral/any	takeResources_includeUnitsWithinRange_team: own
1.13.3	takeResources_excludeUnitsWithoutTags	tags	Includes all units without the specified tags in taking resources	takeResources_excludeUnitsWithoutTags: truck, tanker
1.13.3	takeResources_excludeUnitsWithTheseResources	customPrice	Excludes all units with custom prices in them	takeResources_excludeUnitsWithTheseResources: oil_juice, sauce
1.13.3	takeResources_excludeUnitsWithoutAllResources	bool	Defaults to true.	takeResources_excludeUnitsWithoutAllResources: true
1.13.3	takeResources_triggerActionIfAnyCollected	action refs	Triggers an action if any amount of resources is collected in the current collection action	takeResources_triggerActionIfAnyCollected: purify
1.13.3	takeResources_triggerActionIfNoneCollected	action refs	Triggers an action if no amount of resources is collected in the current collection action	takeResources_triggerActionIfNoneCollected: callOtherTanker
1.15p11	takeResources_triggerActionForEach	action refs	Calls this action for each unit found by takeResource with the unit as thisActionTarget, and thisActionIndex counting up from zero	takeResources_triggerActionForEach: manufacture
1.13.3	takeResources_discardCollected	bool	Just take resources from targets, don't add(or remove) to self	takeResources_discardCollected: false
1.13.3	takeResources_keepResourcesOnTarget	bool	Don't add/remove resource from target. This clones resources. Use with takeResources_discardCollected and takeResources_triggerActionIfAnyCollected to make a resource detector.	takeResources_keepResourcesOnTarget: true
	takeResources_maxUnits	int	Defaults to 1. Takes resources from specified amount of units	takeResources_maxUnits: 10
	takeResources_searchOnly	bool	Shortcut for **maxUnits: 200, discardCollected: true, keepResourcesOnTarget: true** to detect resources only	takeResources_searchOnly: true
1.14	takeResources_directTransferStoppingAtZero	bool	If less resources on target than transfer amount, only remaining resources will be transfered. Doesn't support use with some other takeResources_* keys	takeResources_directTransferStoppingAtZero: true
	#====		**Outcome - Convert Resources**	
1.13.3	convertResource_from	customResource	Name of custom resource to take from	convertResource_from: juice
1.13.3	convertResource_to	customResource	Name of custom resource to give to	convertResource_to: sauce
1.13.3	convertResource_minAmount	float	Skip if less than this amount in 'from'. Defaults to 0. Likely not needed for most use cases	convertResource_minAmount: 10
1.13.3	convertResource_maxAmount	float	Max amount to transfer between 'from' and 'to'	convertResource_maxAmount: 100
1.13.3	convertResource_multiplyAmountBy	float	Defaults to 1. Amount to multiply when adding on 'to' (does not effect amount taken on 'from')	convertResource_multiplyAmountBy: 1.5
	#====		**Outcome - Set Resource**	
1.13.3	resourceAmount	customResource	Name of custom resource to set with the below 3 keys. All keys are optional, and can be used together.	resourceAmount: oil
1.13.3	resourceAmount_setValue	float	Absolute value to set this resource to, ignores current value of resource. Skipped by default	resourceAmount_setValue: 20
1.13.3	resourceAmount_addOtherResource	customResource	Name of another custom resource to add to this on. Can be used without resourceAmount_setValue, to just add resources. Or with resourceAmount_setValue:0 to copy a resource value.	resourceAmount_addOtherResource: juice
1.13.3	resourceAmount_multiplyBy	float	Defaults to 1. Multiple the current or new value by	resourceAmount_multiplyBy: 2
	#====		**Outcome - Attachment changes**	
1.13.2	attachments_addNewUnits	unit types	Adds specified units to attachments	attachments_addNewUnits: coreRenew*3
1.13.3	attachments_deleteNumUnits	int	Removes a specified amount of units attached	attachments_deleteNumUnits: 3
1.13.3	attachments_onlyOnSlots	attachment ids	Restrict attachments_* actions to these attachments	attachments_onlyOnSlots: drop1, drop2, drop3
1.14	attachments_unload	bool	Unload all attachments. Can be used with **attachments_onlyOnSlots**. Same as unloading transported units	attachments_unload: true
1.14	attachments_disconnect	bool	Disconnect all attachments in the place they are right now. Can be used with **attachments_onlyOnSlots**.	attachments_disconnect: true
1.13.2	disconnectFromParent	bool	Disconnects this unit from parent's attachment slot	disconnectFromParent: true
	#====		**Outcome - Tag changes**	
1.13.2	temporarilyAddTags	tags	Add tag to this unit until it is converted or reset (unless **convertTo_keepCurrentTags** is used)	temporarilyAddTags: emptyJuice, fullSauce
1.13.3	temporarilyRemoveTags	tags	Remove tag from this unit until it is converted or reset (unless **convertTo_keepCurrentTags** is used)	temporarilyRemoveTags: fullJuice, emptySauce
1.13.3	resetToDefaultTags	bool	Reset to standard tags	resetToDefaultTags: true
1.13.3	addGlobalTeamTags	tags	Add a tag to player's team. Use with self.globalTeamTags() to create unlocks and upgrades. Unique tags are best to not conflict with other mods.	addGlobalTeamTags: upgrade_energized, upgrade_research2
1.13.3	removeGlobalTeamTags	tags	Remove from player's team.	removeGlobalTeamTags: buff_immune
	#====		**Outcome - Show Message**	
1.13.3	showMessageToPlayer	string	Sends a message to the player controlling the unit	showMessageToPlayer: There is a hidden enemy near your defenses
1.13.3	showMessageToPlayer_[LANG]	string	Note: This format is support on nearly all strings that show to player even when reference doesn't show it. Use ISO 639-1 Language Code on the [LANG] placeholder	showMessageToPlayer_fil: May nakatagong kalaban na malapit sa mga depensa mo
1.13.3	showMessageToAllPlayers	string	Sends a message to all players	showMessageToAllPlayers: %{playerName} has captured a point
1.14	showMessageToAllEnemyPlayers	string	Sends a message to all enemy players only	showMessageToAllEnemyPlayers: Team %{perif.playerName} has %{perif.resource.gold}
1.13.2	showQuickWarLogToPlayer	string	Sends a Quick War Log message to the player controlling the unit (in the lower left)	showQuickWarLogToPlayer: Unit destroyed
1.13.2	showQuickWarLogToAllPlayers	string	Sends a Quick War Log message to all players (in the lower left)	showQuickWarLogToAllPlayers: 500 oil sold to allied market
1.13.2	debugMessage	string	Shows in Sandbox with Debug mode on.	debugMessage: [action log] unit launch on %{thisActionTarget.x}, %{thisActionTarget.y}
	Section		**[effect_NAME]**	Effects are purely visual, but can be important for a mod
	Code	**Value Type**	**Description**	Example
1.15p9	drawType	normal/displacement	Defaults to normal. Changes how effect is rendered. Options: normal(displacement ((displacement only) shows when shader details are turned on)	drawType: normal
1.13	life	float	Defaults 200. Time till effect is removed. Set low as possible to reduce effect overhead.	life: 70
1.13	lifeRandom	float	Random offset life by +/- this value	lifeRandom: 12
1.13	alsoEmitEffects	effect ref	Create more effects when created, useful for meta-effects. Note: other 'alsoEmitEffects' on created effects are ignored.	alsoEmitEffects: CUSTOM:extraSparks*2
1.13	alsoEmitEffectsOnDeath	effect ref	Create these effects when life runs out.	alsoEmitEffectsOnDeath: CUSTOM:finalPuff
	ifSpawnFailsEmitEffects	effect ref	If 'spawnChance' for this effects fails then emit these effects instead	ifSpawnFailsEmitEffects: CUSTOM:greenSparks*4
1.13	alsoPlaySound	sound ref	Plays sound upon spawning the effect. Use OGG or WAV audio files. Appending a colon with a float value sets the volume	alsoPlaySound: meow.wav:0.5
1.13	createWhenOffscreen	bool	Defaults false. When true, effect is created when the unit is offscreen	createWhenOffscreen: true
1.13	createWhenZoomedOut	bool	Defaults true. Effect is created even when the map is zoomed out. Set to false for improving performance	createWhenZoomedOut: false
1.13	createWhenOverLiquid	bool	Defaults true. Effect is created when the unit is over water	createWhenOverLiquid: false
1.13	createWhenOverLand	bool	Defaults true. Effect is created when the unit is over land	createWhenOverLand: false
1.13	spawnChance	float	Default 1. If less than 1 effect only has a random chance of being created	spawnChance: 1
1.12	showInFog	bool	Default false. When true, effect is still visible on fog of war	showInFog: true
1.13	delayedStartTimer	time	Hide for a time before showing and updating effect.	delayedStartTimer: 2s
1.13	liveAfterAttachedDies	bool	Defaults false when attachedToUnit is being used	liveAfterAttachedDies: false
1.12	priority	string	Defaults to high. very/low/low/high/very/high/critical. Takes effect when too many effects are being shown at once.	priority: critical
	#====		**Movement**	
1.13	attachedToUnit	bool	Attach to unit or projectile that created this effect. Will move with this object. Useful for thrust effects	attachedToUnit: true
1.13	alwayStartDirAtZero	bool	Ignore source/attached unit dir	alwayStartDirAtZero: true
1.13	atmospheric	bool	Apply drag to slow this effect down and add small wind effects	atmospheric: true
1.13	physics	bool	Fall to ground and bounce. Needs height to take effect.	physics: true
1.13	physicsGravity	float	Defaults to 1, height speed acceleration when physics: true	physicsGravity: 0.5
1.13	xOffsetRelative	float	Offset starting effect position. Relative to direction of attached turret, projectile, unit	xOffsetRelative: 2
1.13	yOffsetRelative	float	Offset starting effect position. Relative to direction of attached turret, projectile, unit	yOffsetRelative: 2
1.13	xOffsetRelativeRandom	float	Random offset by +/- this value	xOffsetRelativeRandom: 4
1.13	yOffsetRelativeRandom	float	Random offset by +/- this value	yOffsetRelativeRandom: 4
1.13	xOffsetAbsolute	float	Offset starting effect by position ignoring direction of attached turret, projectile, unit	xOffsetAbsolute: 2
1.13	yOffsetAbsolute	float	Offset starting effect by position ignoring direction of attached turret, projectile, unit	yOffsetAbsolute: 2
1.13	xOffsetAbsoluteRandom	float	Random offset by +/- this value	xOffsetAbsoluteRandom: 5
1.13	yOffsetAbsoluteRandom	float	Random offset by +/- this value	yOffsetAbsoluteRandom: 5
1.13	xSpeedRelative	float	Moves the effect sprite on relative horizontal position with specified speed. Negative will move left relative to source, positive will move right relative to source. Useful for thrust effects for moving units	xSpeedRelative: 2
1.12	ySpeedRelative	float	Moves the effect sprite on relative vertical position with specified speed. Negative will move down relative to source, positive will move up relative to source. Useful for thrust effects for moving units	ySpeedRelative: 3
1.12	xSpeedRelativeRandom	float	Randomly change by -value to value	xSpeedRelativeRandom: 2
1.12	ySpeedRelativeRandom	float	Randomly change by -value to value	ySpeedRelativeRandom: 2
1.12	xSpeedAbsolute	float	Moves the effect sprite on absolute horizontal position with specified speed. Negative will move left relative to map, positive will move right relative to map. Useful for smoke effects on structures	xSpeedAbsolute: 5
1.12	ySpeedAbsolute	float	Moves the effect sprite on absolute vertical position with specified speed. Negative will move down relative to map, positive will move up relative to map. Useful for smoke effects on structures	ySpeedAbsolute: 5
1.13	xSpeedAbsoluteRandom	float	Randomly change by -value to value	xSpeedAbsoluteRandom: 2
1.13	ySpeedAbsoluteRandom	float	Randomly change by -value to value	ySpeedAbsoluteRandom: 2
1.13	hOffset	float	Height offset from source. May be mistakenly confused with hOffsetAbsolute	hOffset: 4
1.13	hOffsetRandom	float	Randomly change by -value to value	hOffsetRandom: 5

Ver	Code	Value Type	Description	Example
1.13	hSpeed	float	Sets the speed to change the height of the effect	hSpeed: 1
1.13	hSpeedRandom	float	Randomly change by -value to value	hSpeedRandom: 1
1.13	dirOffset	float	Sets the static direction of the effect	dirOffset: 43
1.13	dirOffsetRandom	float	Randomly change by -value to value	dirOffset: 50
1.13	dirSpeed	float	Sets the rotation speed of the effect	dirSpeed: 2
1.13	dirSpeedRandom	float	Randomly change by -value to value	dirSpeedRandom: 3
	Graphics			
1.13	frameIndex	int	Use a specific frame from strip image	frameIndex: 0
1.13	frameIndexRandom	bool -?	Use random frame from strip index	frameIndexRandom: true
1.13	stripIndex	int/string	A built-in image set to use. Cannot be used with custom image. Options: effects/explode_big/light_50/flame/effects/effects2/projectiles/projectiles2/explode_bits	stripIndex: projectiles
	image	image	Custom image file to use. Cannot be used with stripIndex.	image: whiteFull.png
	imageShadow	image	Custom image file to use for shadows	imageShadow: AUTO
1.13	scaleTo	float	Defaults to 1. Resizes the unit into the specified scale	scaleTo: 2
1.13	scaleFrom	float	Defaults to 1. Resizes the unit from the specified scale	scaleFrom: 4
1.13	color	color	Default=#FFFFFFFF. Changes the color of the effect sprite. Use pure white sprite for most use cases.	color: #ff00ff
	teamColorRatio		Sets team coloration between 0-1. Team color is dependent from source unit	teamColorRatio: 1
1.13	drawUnderUnits	bool	Renders the effect under all units	drawUnderUnits: false
1.13	fadeInTime	bool	Fade alpha from 0% to 100% for this time at start	fadeInTime: 2s
1.13	fadeOut	bool	Fade alpha from 100% to 0% based on life. Set alpha is higher than 1 to delay fade	fadeOut: 4s
	alpha		Capped between 0-1. Can be set higher than 1 to delay fadeOut effects	alpha: 1
	shadow		True to draw a shadow. Forced true if imageShadow is used	shadow: false
1.14	pivotOffset	float	Similar to dirOffset, this will also rotate relative keys and child elements	pivotOffset: 20
1.14	pivotOffsetRandom	float	Offsets pivot between +- or specified value	pivotOffsetRandom: 34
	Animation			
1.13	total_frames		Total frames of 'image', used with animation or frameIndex. Only needed with custom images	total_frames: 10
1.13	animateFrameStart	int	Starting frame for the animation	animateFrameStart: 0
1.13	animateFrameEnd	int	Ending frame for the animation	animateFrameEnd: 3
1.13	animateFramePingPong	int	If true, animation will go back and forth within the frames	animateFramePingPong: true
1.13	animateFrameSpeed	time	Sets the animation speed. The lower the value, the slower it plays	animateFrameSpeed: 10
1.13	animateFrameSpeedRandom	time	Randomizes how fast the effect animation plays	animateFrameSpeedRandom: 20
1.14	animateFrameLooping		Defaults false. When false effect is removed when animation ends	animateFrameLooping: true

	Section		[animation_NAME]	Use this to make intricate animations based on different circumstances
	Code	Value Type	Description	Example
1.13	onActions	enum	Automatically plays the animation on specified events. Options: move, attack, idle, underConstruction, underConstructionWithLinkedBuiltTime, queuedUnits, repair	onActions: move
1.13	onActionsQueuedUntilPlayAt	time	For onAction: queuedUnits. Amount queue needs to reach before starting, set between 0-1	onActionsQueuedUntilPlayAt: 0
1.13	blendIn	time	Blend with last animation for this time	blendIn: 2s
1.13	blendOut	time	Blend with next animation for this time	blendOut: 3s
1.12	pingPong	float	Play animation in reverse after it ends	pingPong: true
1.13	KeyframeTimeScale:	float	Scales all keyframe times, useful to make an animation faster/slower without changing everything	KeyframeTimeScale: 1
	Keyframes - create as many as needed			
1.13	arm#_[time]		Adds a keyframe at time. Use multiple times to create animation.	arm1_3s: {x: 5, dir: 90 }
	leg#_[time]		Adds a keyframe at time. Use multiple times to create animation.	leg1_3s: {dir: 300}
	body_[time]		Adds a keyframe at time for body. Only frame and scale allowed on body	body_4s: {frame: 4, scale: 0.5}
1.14	effect_[time]		Spawn effects while playing an animation	effect_2s: {name:CUSTOM,myExplode, x: 0,y: 5}
	direction_units	float	Overrides [graphics]animation_direction_units while this animation is playing	direction_units: 45
	direction_strideX	int	Overrides [graphics]animation_direction_strideX	direction_strideX: 20
	direction_strideY		Overrides [graphics]animation_direction_strideY	direction_strideY: 50
	direction_starting	float	Overrides [graphics]animation_direction_starting	direction_starting: 0
	Deprecated Keys (can be used but there are better ways)			
1.12				
1.13				
1.13				
1.13				
1.13				

	Type		spawnUnits:LIST	Spawn lines specifically for units, used with "unit ref" value types
	Code		Description	Example
			Most units spawning keys support multiple units with parameters	
1.13.2	neutralTeam		Spawn the unit on the neutral team instead of the same team as source	spawnUnits: crates*10{neutralTeam=true}, tank{spawnChance=0.5}
1.13.3	setToTeamOfLastAttacker		Spawn the unit as the last attacker of source (useful on [core] unitsSpawnedOnDeath)	spawnUnits: egg{setToTeamOfLastAttacker=true}
1.13.3	spawnChance		Chance this unit will spawn. Defaults to 1.	spawnUnits: shards{spawnChance=0.3}
0.99	spawnSource	unit ref	Changes spawn location and team of spawned units to this unit ref	spawnUnits: tank{spawnSource=memory.lastLocation}
1.13.3	maxSpawnLimit		Useful with spawnChance, max number of units to spawn in total	spawnUnits: tree{spawnChance=0.5, maxSpawnLimit=1}, tree{maxSpawnLimit=1}
1.13	gridAlign		Align spawn location to grid, useful for buildings	spawnUnits: hovertank{gridAlign: true}
1.13.3	skipIfOverlapping	bool	Don't spawn this unit if spawn in an invalid location. Eg on units or over water when LAND based	spawnUnits: crates*10{skipIfOverlapping=true, offsetRandomX=40, offsetRandomY=40, gridAlign=true}
1.13.2	offsetX	float	Sets horizontal position relative to source	spawnUnits: jet{offsetX=20), jet{offsetX=20}
1.13.5	offsetY	float	Sets vertical position relative to source	spawnUnits: scout{offsetY=40), scout{offsetY=-40}
1.13.2	offsetDir	float	Set the facing direction of spawned units	spawnUnits: bike{offsetDir=45}
1.13.3	offsetHeight	float	Sets the height for spawned units	spawnUnits: drone{offsetHeight=20}
1.13.5	offsetRandomX	float	Sets random horizontal position relative to source	spawnUnits: kli*5{offsetRandomX=45}
1.13.3	offsetRandomY	float	Sets random vertical position relative to source	spawnUnits: scavs*10{offsetRandomY=80}
	offsetRandomDir	float	Set random facing direction of spawned units	spawnUnits: chickens*20{offsetRandomDir=380}
1.13.2	addResources	resource ref	Give spawn unit those resources, can be used to set flags that trigger actions	spawnUnits: crates{addResources=gold:30;stone:10, spawnChance=0.5}
1.14	transportedUnitsToTransfer		Puts the designated amount of transported units into the transport of the spawned unit.	spawnUnits: transporter{transportedUnitsToTransfer=5}
1.16	copyWaypointsFrom	unit ref	Copies all waypoints on target to created units.	spawnUnits: tank{copyWaypointsFrom=self}
1.14	alwaysStartDirAtZero		Sets the direction of spawned unit to 0 degrees.	spawnUnits: builder{alwaysStartDirAtZero=true}
	damagingBorder		Used with techLevel	
	zoneMarker		Used with techLevel	
	techLevel	int	Usually used in IXR maps, controls the size of safe zone markers and damaging borders. 1 techLevel = 10x10 tiles = 200 pixels in diameter	spawnUnits: damagingBorder{techLevel=100}, zoneMarker{techLevel=40}

	Type		spawnProjectiles:LIST	Spawn lines specifically for projectiles, used with "proj ref" value types
1.14	Code		Description	Example
			NOTE: If you need to use offsetDir to rotate your projectiles in a different direction (like for shrapnel), you must also set these projectile speeds low enough to force their trajectory	Example: [projectile_main] spawnProjectilesOnExplode: shrapnel{offsetDir=90}, shrapnel{offsetDir=90} [projectile_shrapnel] turnSpeed: 0
1.14	spawnChance		**Most projectile spawning keys used for projectile ref**	
1.14	spawnChance		Chance this projectile will spawn. Defaults to 1.	spawnProjectilesOnEndOfLife: secondary*3{spawnChance=0.5}
1.14	maxSpawnLimit		Maximum amount to spawn	spawnProjectilesOnExplode: bomblet*3{spawnChance=0.5, maxSpawnLimit=2}
1.14	recursionLimit	int	Prevents loops, useful with spawning itself so it doesn't infinitely spawn, good for chain exploding. (Recommended no more than 4 if spawning more than 3 projectiles)	spawnProjectilesOnEndOfLife: flamingBits*4{spawnChance=0.35, recursionLimit=2}
1.14	offsetX	float	Sets horizontal position	spawnProjectilesOnCreate: shot{offsetX=10, recursionLimit=0}, shot{offsetX=-10, recursionLimit=0}, shot{offsetX=10, recursionLimit=0}
1.14	offsetY	float	Sets vertical position	spawnProjectilesOnCreate: shot{offsetY=-10, recursionLimit=0}, shot{offsetY=-10, recursionLimit=0}, shot{offsetX=10, recursionLimit=0}
1.14	xOffsetRelative	float	Similar to offsetX, but the offset is relative to the position of the projectile	spawnProjectilesOnCreate: homingEnergy{xOffsetRelative=10}
1.14	yOffsetRelative	float	Similar to xOffsetRelative, but for Y axis	spawnProjectilesOnCreate: homingEnergy{yOffsetRelative=5, yOffsetRelative=10}
1.14	offsetRandomX	float	Random value to offset in the X axis only	spawnProjectilesOnExplode: strayBullet{offsetRandomX=10, offsetRandomY=30}
1.14	offsetRandomY	float	Random value to offset in the Y axis only	spawnProjectilesOnExplode: strayBullet{offsetRandomY=10, offsetRandomX=30}
1.14	offsetRandomXY	float	The offset in both directions to randomly spawn, makes truly random projectiles within an area	spawnProjectilesOnExplode: shrapnel*20{spawnChance=0.2, offsetRandomXY=100}
1.14	offsetHeight	float	Sets height of the projectile	spawnProjectilesOnCreate: scythe{offsetHeight=20}
1.14	offsetDir	degrees	Sets direction of the projectile	spawnProjectilesOnCreate: stars{offsetDir=45}
1.14	offsetRandomDir	degrees	Sets random direction of the projectile	spawnProjectilesOnEndOfLife: fireworks{offsetRandomDir=72}

	Type	Returns	LogicBoolean_	Advanced code to create conditionals and triggers
	Code	Returns	Description	Example
	true	bool	This value will meet the condition	autoTrigger: true
	false	bool	This value will not meet the condition	autoTrigger: false
	if	bool	Start all logic booleans with if, unless just using booleans (true/false)	isActive: if self.hasFlag(id=1)
	and	bool	Connector: Adds another condition to the list. All conditions linked by this must be reached to return true.	autoTrigger: if self.isInWater and self.energy>=1
	or	bool	Connector: Adds another condition to the list. Just one of the conditions linked must be reached to return true	autoTrigger: if {self.energy>=2 or self.ammo>=1} and self.isFlying
	not	bool	Conditional: Reverts meaning of the next logic boolean. Useful to set negative conditions	if not self.isOverLiquid (the opposite of being over liquid)
1.16	<	bool	Return true if number on left is smaller. Means less than next value	isLocked: if self.ammo < 1
1.16	>	bool	Return true if number on right is smaller. Means greater than next value	isLocked: if self.isInWater and self.energy>=1
1.16	<=	bool	Return true if number on left is smaller or equal. Means less or equal than next value	requireConditional: if numberOfUnitsInTeam{withTag="fish"} >= 10
1.16	>=	bool	Return true if number on left is smaller or equal. Means greater or equal than next value	requireConditional: if numberOfUnitsInTeam{withTag="fish"} <= 11
1.16	==	bool	Return true if number, unit, string, bool on both sides is the same. Means equal to next value	hidden: if memory.tal == 0
1.16	!=	bool	Return true if number, unit, string, bool on both sides different. Means different to next value	isVisible: if memory.message != parent.readUnitMemory{name="stateMessage", type="string"}
1.16	+	same type	Add two numbers or join two strings. Means addition to this value	setUnitMemory: stockpile = self.ammo - customTarget1.resource.reserve
1.16	-	same type	Subtract two numbers. Means subtraction to this value	setUnitMemory: stockpile = self.ammo - customTarget2.resource.reserve
1.16	/	same type	Divide two numbers. Means division of two values	setUnitMemory: damageMultiplierBuffer = memory.population / memory.death
1.16	*	same type	Multiply two numbers. Means multiplication of two values	setUnitMemory: reserveCash = memory.population * self.resource.credits
1.16	%	same type	Divides two numbers and returns the remainder. Means a percentage of a value	setUnitMemory: isEvenX = select{self.x % 2, true, false}
			Unit location and movement	"self" prefix can be replaced with other unit-based prefixes (more information below)
	self.isUnderwater()	bool	Checks if unit is underwater	autoTrigger: if self.isUnderwater()
	self.isAtGroundHeight()	bool	Checks if unit is in surface level	autoTrigger: if thisActionTarget.isAtGroundHeight()
	self.isFlying()	bool	Checks if unit is flying	isActive: if self.isFlying()
	self.isMoving()	bool	Checks if unit is moving on by any means	isLocked: if self.isMoving()
0.99	self.isReversing()	bool	Checks if unit is moving backwards	isLockedAt2: if self.isReversing()
	self.isAtTopSpeed()	bool	Checks if the unit current speed matches the moveSpeed properly	autoTrigger: if self.isAtTopSpeed()
	self.isInWater()	bool	Checks if unit touches a water tile	isLocked: if not self.isInWater()
	self.isOverwater()	bool	Checks if unit is touching or over a water tile	isLocked: if not self.isOverwater()
	self.isOverLiquid()	bool	Checks if unit is touching or over a liquid tile (water, lava)	isLocked: if not self.isOverLiquid()
	self.isOverCliff()	bool	Checks if unit is touching a cliff tile	isLocked: if not self.isOverCliff()
	self.isOverPassableTile()	bool	Checks if unit is touching or over a specific tile (parameters: type, Type Value: NONE, LAND, BUILDING, HOVER, OVER_CLIFF, CLIFF, OVER_CLIFF, WATER, AIR, WATER)	isLockedAt: if self.isOverPassableTile{type=OVER_CLIFF}
	self.isOverOpenLand()	bool	shortcut for self.isOverPassableTile{type="LAND"}	hidden: if not self.isOverOpenLand()
			Unit state	
1.13.4	self.hasResources()	bool	Checks if unit has these resources. Can check multiple resources at the same time (all price parameters)	isActive:self.hasResources{credits=1, energy=2}
	self.resource()	bool / bool	Checks a single resource (parameters: type, greaterThan, lessThan) (1.15 returns float with no parameters)	isActive: self.resource{type=gold} >= 10
1.16	self.resource.RESOURCE_TYPE	float	Shortcut for: self.resource{type="RESOURCE_TYPE"}	addResourcesWithLogic: hp += self.resource.gold
1.167	self.isResourceLargerThan()	bool	Compare two resources based each other, note multiplyTargetBy doesn't make any changes. (parameters: source=x, compareTarget=x, byMoreThan=x, multiplyTargetBy=x)	autoTrigger: self.isResourceLargerThan{source=oil, compareTarget=sauce, byMoreThan=100, multiplyTargetBy=0.2}
	self.hp()	float / bool	(parameters: greaterThan, lessThan, full). (1.15 returns float with no parameters)	addResourcesWithLogic: credits += select{self.hp > selfmaxHp, 10, 3}
0.99	self.maxHp()	float / bool	(parameters: greaterThan, lessThan, full), (1.15 returns float with no parameters)	setUnitMemory: sauce = self.maxHp * self.energy
	self.height()	float / bool	(parameters: greaterThan, lessThan, full). (1.15 returns float with no parameters)	imageScale: 1 + (self.height * 0.1)
	self.ammo()	int / bool	Returns int if no parameters, boolean with any parameters (parameters: greaterThan, lessThan, empty, full)	isLocked: if self.ammo < 1
	self.ammoIsEmpty()	bool	shortcut for self.ammo{empty=true}	isLockedAt: self.isAmmoEmpty()
	self.ammoIncludingQueued()	int / bool	Also includes ammo from actions still in queue (parameters: greaterThan, lessThan, empty, full)	isLocked: if self.ammoIncludingQueued{lessThan=10}
	self.energy()	float / bool	(parameters: greaterThan, lessThan, empty, full). (1.15 returns float with no parameters)	isVisible: if self.energy() > 50
	self.energyIncludingQueued()	float / bool	Also includes energy from actions still in queue (parameters: greaterThan, lessThan, full)	isVisible: if self.energyIncludingQueued()
	self.isEnergyFull()	bool	shortcut for self.energy{full=true}	isActive: if self.isEnergyFull()
	self.isEnergyEmpty()	bool	shortcut for self.energy{empty=true}	isActive: if self.isEnergyEmpty()
1.15p10	self.maxEnergy()	float / bool	Returns maximum energy (defaults to energyMax value in core, dynamically adjusts to changed value made by setUnitState)	isLocked: if memory.kills < self.maxEnergy()
	self.isEnergyRecharging()	bool	Checks if unit energy is passively recharging	text: Reload %[select{self.isEnergyRecharging(), '', "ready"]}
	self.shield()	float / bool	(parameters: greaterThan, lessThan, empty, full). (1.15 returns float with no parameters)	isActive: if self.shield() == 0
1.15p10	self.maxShield()	float / bool	Returns the maximum shield value (defaults to maxShield value in core, but dynamically adjusts to changes made with setUnitState)	isLockedAt2: if self.energy += self.maxShield()/2
	self.kills()	int / bool	(parameters: greaterThan, lessThan). (1.15 returns int with no parameters)	setUnitMemory: score = self.kills()*3
1.13.3	self.queueSize()	int / bool	(parameters: greaterThan, lessThan, full, empty, equalTo) (1.15 returns int with no parameters). 1.15p9 adds a new	isLocked: if self.queueSize() >= 4
	self.teamId()	int	Return team id of unit or marker. Starts at 0. (but -1 for a neutral team)	switchToteam: lastDamagedBy.teamId()
	self.teamName()	string	Returns allied group team name	showMessageToAllPlayers: %{attacking.teamName()} dominated %{thisActionTarget.teamName()}
1.16	self.playerName()	string	Returns player's name	showQuickWarLogToAllPlayers: %[self.playerName()] destroyed %[thisActionTarget.playerName()]'s %[thisActionTarget]
	self.x()	float	Returns the horizontal position of the unit	description: [Position]n{X}: %[self.x()n{Y}: %[self.y()n{Z}: %[self.z()]D}: %[self.dir]
	self.y()	float	Returns the vertical position of the unit	description: [Position]n{X}: %[self.x()n{Y}: %[self.y()n{Z}: %[self.z()]D}: %[self.dir]
	self.z()	float	Returns the elevation position of the unit (identical to self. height)	description: [Position]n{X}: %[self.x()n{Y}: %[self.y()n{Z}: %[self.z()]D}: %[self.dir]
	self.dir()	float	Returns the unit's direction	description: [Position]n{X}: %[self.x()n{Y}: %[self.y()n{Z}: %[self.z()]D}: %[self.dir]
	self.priceCredits()	float	Returns the price of the unit. Only returns the credit price. Does not support custom resources.	addResourcesWithLogic: credits = attacking.priceCredits()
1.15p9	self.buildAmount()	float	Returns the current value of build progression of the unit	stack_indexCount: int(12 * self.buildAmount() * 0.1)
1.15p9	self.isCompleted()	bool	Shortcut for self.buildAmount() == 1	autoTrigger: if self.isCompleted()
1.16	self.speed()	float	Returns current speed of the unit	isLockedAt: if self.speed < 2
1.15p9	self.maxMoveSpeed()	float	Returns the maximum possible speed set for moveSpeed property, may be manipulated with setUnitState	autoTrigger: if memory.defaultSpeed = self.maxMoveSpeed
			Misc	
	self.hasFlag()	bool	Boolean flags send units for mods to use. (parameters: id=0-31).	autoTrigger: if self.hasFlag(id=1)
	self.isInTag()	bool	Checks in own tags of a unit (parameters: includes)	autoTrigger: if self.isInTag{includes="fish"}
1.13.3	self.globalTeamTags()	bool	Checks or returns global tags on a unit (parameters: includes)	autoTrigger: if self.globalTeamTags{includes="animal"}
1.13.3	self.transportingCount()	int	(parameters: greaterThan, lessThan, empty) (1.15 returns int with no parameters)	text: Load ~ %[self.transportingCount()]

Version	Code	Value Type	Description	Example
	self.numberOfAttachedUnits()	int / bool	(withTag, greaterThan, lessThan) (1.15 returns int with no parameters)	text: Cores - %{self.numberOfAttachedUnits()}
	self.isAttacking()	bool	Checks if the unit is attacking something	setUnitStats: moveSpeed=select(self.isAttacking, 4, 2)
	self.hasActiveWaypoint()	bool	(parameters: type=WAYPOINT_TYPE)	text: Waypoint - %{self.hasActiveWaypoint(type='move')}
1.15	self.numberOfQueuedWaypoints()	int / bool	(parameters: type=WAYPOINT_TYPE, amount=#)	text: Waypoint Queue - %{self.numberOfQueuedWaypoints(type='attackMove')} >= 2
	self.transportingUnitWithTag()	bool	(parameters: includes)	requireConditional: if self.transportingUnitWithTag(includes='human')
	self.hasParent()	bool	Checks if a unit is attached or a passenger of a unit. For both attachments and transports (parameters: [withTag=x])	isVisible: if self.hasParent
	self.hasTakenDamage()	float / bool	Checks when the unit last took damage (parameters: withinSeconds=X, laterThanSeconds=X). (v1.15 returns float in seconds with no parameters)	isActive: if self.hasTakenDamage(withinSeconds=1)
	self.timeAlive()	float / bool	Checks how long the unit is alive (parameters: withinSeconds=X, laterThanSeconds=X). (v1.15 returns float in seconds with no parameters)	isLocked: if self.timeAlive() > 100
	self.lastConverted()	float / bool	Check the time the unit last converted from (parameters: withinSeconds=X, laterThanSeconds=X). (v1.15 returns float in seconds with no parameters)	isLockedAlt2: if self.lastConverted()
	self.customTimer()	float / bool	Returns the current value in the unit's custom timer (parameters: withinSeconds=X, laterThanSeconds=X). (v1.15 returns float in seconds with no parameters)	autoTrigger: if self.customTimer() > 120
	self.isOnNeutralTeam()	bool	Check if the unit belongs to the neutral team (Team ID -1)	
	self.numberOfUnitsInTeam()	int / bool	Checks amount of player units that meet the parameters (parameters: withTag, greaterThan, lessThan, withinRange, incompleteBuildings, factoryQueue) (v1.15 returns int with no parameters)	isLocked: if numberOfUnitsInTeam(withTag='techUnlockBuilding', lessThan=5)
	self.numberOfUnitsInAllyTeam()	int / bool	Checks amount of allied and player units that meet the parameters (parameters: withTag, greaterThan, lessThan, withinRange, incompleteBuildings, factoryQueue) (v1.15 returns int with no parameters)	description: TEAM %{self.teamName()}\n\nTank Population: %{self.numberOfUnitsInAllyTeam()}
1.15	self.numberOfUnitsInAllyNotOwnTeam()	int / bool	Checks amount of allied only units that meet the parameters (parameters: withTag, greaterThan, lessThan, withinRange, incompleteBuildings, factoryQueue) (v1.15 returns int with no parameters)	isActive: if self.numberOfUnitsInAllyNotOwnTeam() > 100
1.15.3	self.numberOfUnitsInEnemyTeam()	int / bool	Checks amount of enemy units that meet the parameters (parameters: withTag, greaterThan, lessThan, withinRange, incompleteBuildings, factoryQueue) (v1.15 returns int with no parameters)	autoTrigger: if self.numberOfUnitsInEnemyTeam(withinRange=500) > 0
	numberOfUnitsInNeutralTeam()	int / bool	Checks amount of neutral units that meet the parameters (parameters: withTag, greaterThan, lessThan, withinRange, incompleteBuildings, factoryQueue) (v1.15 returns int with no parameters)	isVisible: if memory.deadzone.numberOfUnitsInNeutralTeam() >= 10
1.14	numberOfUnitsInAggressiveTeam()	int / bool	Checks amount of aggressive neutral units that meet the parameters (parameters: withTag, greaterThan, lessThan, withinRange, incompleteBuildings, factoryQueue) (v1.15 returns int with no parameters)	isVisible: if memory.deadzone.numberOfUnitsInAggressiveTeam() == 30
1.15p9	numberOfUnitsInAllTeams()	int / bool	Checks amount of any units that meet the parameters (parameters: withTag, greaterThan, lessThan, withinRange, incompleteBuildings, factoryQueue) (v1.15 returns int with no parameters)	text: Birds: %{numberOfUnitsInAllTeams(withTag='bird')}
	self.hasUnitInTeam()	bool	(parameters: withTag, withinRange, incompleteBuildings, factoryQueue)	isLocked: if self.hasUnitInTeam(withTag='fish')
1.13.3	self.notUnitInTeam()	bool	(parameters: withTag, withinRange, incompleteBuildings, factoryQueue) also for numberOfUnitsInTeam	isLocked: if self.notUnitInTeam(withTag='infected')
	self.isControlledByAi()	bool	Checks if a unit is controlled by AI	text: Robot? %{select(self.isControlledByAi(), "Yes", "No")}
1.14	self.readUnitMemory()	any type	(name,string, type:string[boolean,unit,float,string], [default])	isActive: if parent.readUnitMemory('boostTarget', type='unit') == self
	**===== **	**#===== **	**Unit references**	
	thisActionTarget	unit / marker	Current target or location targeted.	fireTurretAtAllGround: mainGun: (thisActionTarget==Marker with ground location) aliasTriggerAction: x (thisActionTarget=Same as original action) [turret]onShoot_triggerAction: x (thisActionTarget=Target that was shot at) takeResources_triggerActionIfAnyCollected: x (thisActionTarget=Target with resources) addWaypoint_triggerActionWhenMet: x (thisActionTarget == for Marker for moveTarget for attack, etc. Note: use addWaypoint_maxTime:0 if you want to search only)
	thisActionIndex	int	Used with alsoTriggerActionRepeat and takeResources_triggerActionForEach	alsoTriggerActionRepeat: thisActionIndex
	eventSource	unit / marker	Current trigger from an autoTriggerOnEvent, otherwise null	autoTriggerOnEvent: tookDamage (eventSource==Unit that caused damage) autoTriggerOnEvent: killedAnyUnit (eventSource==Unit that was killed) autoTriggerOnEvent: transportingNewUnit (eventSource==Unit that was transported) autoTriggerOnEvent: transportUnloadedOrRemovedUnit (eventSource==Unit unloaded) autoTriggerOnEvent: queuedUnitFinished (eventSource==New unit made) autoTriggerOnEvent: touchTargetSuccess (eventSource==Target touched)
	attachment	unit	Returns unit attachment as reference (parameters: [slot], [withTag])	setCustomTarget2: self.attachment(withTag='x').lastDamagedBy.getAsMarker()
	transporting	unit	Returns unit passenger as reference (parameters: [slot], [withTag])	autoTrigger: if self.transporting(slot=0).hasResources(gold=100)
	attacking	unit	Current target this is attacking, might not be the current waypoint target.	isLocked: if attacking.tags(includes='bug') and attacking.hp < 20
	lastDamagedBy	unit	Last unit that attacked this.	teleportTo: lastDamagedBy
	parent	unit	The transporter or attachment parent. (Note: units are suspended state when transported without attachment slot)	autoTrigger: if parent.energy < 100
	activeWaypointTarget	unit	Current active waypoint target. Includes attacking, transporting, repairing, etc.	isActive: if distanceBetween(self, activeWaypointTarget) < 100
	customTarget1	unit	Custom memory, defaults to the unit that created this unit.	isLocked: if parent.customTarget1 == self
	customTarget2	unit	Custom memory, defaults to null	fireTurretAtAllGround_withPosition: customTarget2
	nearestUnit	unit	(withinRange=500, withTag='x', relation='any') Search for a unit (not recommended in autoTrigger check for performance)	setCustomTarget2: nearestUnit(withTag='derrick')
	globalSearchForFirstUnit	unit	(withTag=, relation) - Returns first (and oldest) unit found matching the filter. Slow, avoid using in autoTrigger checks	self.globalSearchForFirstUnit(withTag='gameController', relation='neutral')
	nullUnit	unit	returns a null unit reference, useful for comparisons	isLockedAlt2: if self.parent == nullUnit
	**===== **	**#===== **	**Marker functions**	
	getAsMarker	marker	creates a temporary marker at the position a unit is right now. Markers are very fast to create and automatically removed when no longer needed. Is not linked to any unit and still exists when the unit dies, and stays the same when source moves.	lastDamagedBy.getAsMarker()
	getOffsetAbsolute	marker	([x],[y],[height]) Returns marker with absolute offset (-y is north, +x is east)	lastWaypoint_target_fromReference: unitref.getOffsetAbsolute(self.x, self.y+1000, self.z)
	getOffsetRelative	marker	([x],[y],[height],[dirOffset]) Returns marker with relative offset. (y= is forwards)	self.getOffsetRelatively=100).nearestUnit(withinRange=70, withTag='mouse') != null
	eventSource	event	Returns the event from autoTriggerOnEvent, null if used outside of a valid autoTriggerOnEvent	text: Event: %{eventSource}
	**===== **	**#===== **	**Global functions**	
	readUnitMemory	all arguments	Reads memory from the unit reference (e.g. self. readUnitMemory, parent.readUnitMemory, attacking. readUnitMemory)	self.readUnitMemory('ammoType', type='string'), parent.readUnitMemory('attachmentArray', type='unit', ['', index=5), attacking.readUnitMemory('wishlist', type='string[7]')[3]
	memory.NAME	self only	Shortcut to read the self with current defineUnitMemory types. Cannot be used on other units, use readUnitMemory for that. Don't call with **self**.	autoTrigger: if memory.experience > 100
	**===== **	**#===== **	**Array Memory methods. usable on both memory logic functions above.**	
	memory.NAME.size	int	Returns the size of the array (including empty indexes between indexes)	setUnitMemory: squadSize = memory.squad.size
	memory.NAME.contains(query)	bool	Returns true if the query is found within the array. Basically searches something if it exists.	autoTrigger: if memory.savedCoord.contains(lastDamagedBy.customTarget1.customTarget1)
	distance(x1, y1, x2, y2)	float	Returns the difference between two points	isLocked: if distance(self.x, self.y, thisActionTarget.x, thisActionTarget.y) > 350
	distanceSquared(x1, y1, x2, y2)	float	Returns the squared difference between two points. Bit faster than distance	isLocked: if distanceSquared(self.x, self.y, thisActionTarget.x, thisActionTarget.y) > 19
	distanceBetween(unit1, unit2)	float	Returns the distance between two units or markers	isLocked: if distanceBetween(self, memory.antenna) < 1000
	distanceBetweenSquared(unit1, unit2)	float	Returns the squared distance between two units. Bit faster than distanceBetween	isLocked: if distanceBetweenSquared(self, memory.antenna) > 100
	game.nukesEnabled()	bool	Returns true if nukes are enabled in this game's settings.	isActive: if not game.nukesEnabled()
	int(x)	int	Removes decimal places from a number	stack_indexCount: int(12 * (self.buildAmount * 0.1))
	select(bool, textA, textB)		returns textA if bool is true otherwise returns textB	setUnitMemory: isEvenX = select(self.x % 2, true, false)
	debug(logicBoolean)	string	Returns a text string helping to explain the reason for the current result. Can see into nested logic, comparisons, and operators.	description: [DEBUG] %{debug(self.x > self.y)}
	str(x)	string	Convert a number, unit or boolean into a string	str(self.energy)+'x' == '100x'
1.14	substring(text,start,end)	string	Takes a part of the string from the indicated start and end parts	substring('hello',2,2) == 'he'
	length(string)	int	Returns string length as number	length(memory.sms)
	squareRoot(num)	float	Requires square root of a number	squareRoot(self.x + self.y)
	min(num1, num2)	float	Returns the smallest number	damage = min(self.hp, self.energy)
	max(num1, num2)	float	Returns the biggest number	max(5, 10) == 10
	createMarker(x, y, [height], [team=id], [dir])	marker	Creates a marker that can be used to place coordinates for various actions	fireTurretAtAllGround_withTarget: createMarker(self.x + 500, self.y + 500)
	eventData(name, type, [default])	any type	Displays the data from a specified message from some unit. Only use in autoTriggerOnEvent.newMessage events	text: %{eventData(name='msg', type='string')}
	sin(angle)	float	Takes the sine value of a specified number	sin(10) = 0.173
	cos(angle)	float	Takes the cosine value of a specified number	cos(60) = 0.5
	rnd(min, max)	float	Produces a random number between minimum and maximum input.	rnd(66, 666)
	lowercase(string)	string	Sets all letters from a string into lowercase	lowercase("Spaghetti Macaroni Lasagna") -> "spaghetti macaroni lasagna"
	uppercase(string)	string	Sets all letters from a string into uppercase	uppercase("fish dish wish") -> "FISH DISH WISH"
	direction(x1, y1, x2, y2)	float	Returns the relative angle of x1/y1 and x2/y2	direction(self.x, self.y, attacking.x, attacking.y)
	directionBetween(unit1, unit2)	float	Returns the relative angle of marker A and unitmarker B	directionBetween(self, lastDamagedBy)
	self.isInMap()	bool	Returns true if unit (or marker) is with the game area; Returns false if outside bounds	autoTrigger: if not self.isInMap()
	game.mapWidth()	float	Returns the width of the map	fireTurretAtAllGround_withTarget: createMarker(rnd(0, game.mapWidth), rnd(0, game.mapHeight))
	game.mapHeight()	float	Returns the height of the map	fireTurretAtAllGround_withTarget: createMarker(rnd(0, game.mapWidth), rnd(0, game.mapHeight))
	self.teamDefeatedTech()	bool	Returns true if team is defeated	autoTrigger: if attacking.teamDefeatedTech()
	self.teamWipedOut()	bool	Returns true if team is defeated and wiped out	autoTrigger: if self.teamWipedOut()
1.15p9	self.teamVictory()	bool	Returns true if team is victorious	autoTrigger: if self.teamVictory()

	Type		[placementRule_NAME]	Allows creation of rules for requiring buildings
	Code	Value Type	Description	Example
1.14	anyRuleInGroup:	string	(Only require 1 of the rules in this group pass, instead of all. Use the same group name on other placement rules to create a group.)	anyRuleInGroup: struct
1.14	searchTags:	tag list	Search for any unit with at least one of these tags	searchTags: factory
	searchTeam:	relation	Teams to include in search, can be: own/neutral/ally/notOwn/ally/enemy/any	searchTeam: own
1.14	searchOffsetX:	float	defaults to 0	searchOffsetX: 100
1.14	searchOffsetY:	float	defaults to 0	searchOffsetY: 200
1.14	**searchDistance:**	float	Required	**searchDistance: 500**
1.14	excludeIncompleteBuildings:	bool	Defaults to false. Might want to set to true depending on the requirement reason	excludeIncompleteBuildings: true
1.14	excludeNonBuildings:	bool	Defaults to false	excludeNonBuildings: true
1.14	minCount:	int	Set min amount of units that need to be found in search. (eg needs to be near something). Defaults to 0	minCount: 0
1.14	maxCount:	int	Set max amount of units before match fails (eg cannot be close to something). Defaults to unlimited	maxCount: 1
1.14	blocksPlacement:	bool	Defaults to true.	blocksPlacement: false
1.14	cannotPlaceMessage:	LocaleString	Highly Recommended: Message shown to player if the rule fails (will be first failing rule if using anyRuleInGroup).	cannotPlaceMessage: "No factory is nearby"
1.14	checkEachTile:	bool	defaults to true (set to false to only test unit center; true checks each tile under the unit which shows up on the placement grid. Can be easier to see requirements with true)	checkEachTile: false

	Type		Prices/Resources lines - used by addResources, price, etc	
	Code	Targets	Description	Example
	credits		Global resource	price: 100
	energy		Energy used for laser shield and ammunition	resourceUsage: energy=1
	hp		Unit hitpoints	addResources: hp == 100
	shield		Shielding for units	price: hp=100, shield=100
	ammo		Hidden value on each unit for use by mods	price: hp=100, shield=101
	setFlag		use with addResources, resourceUsage or price. 0-31. Flags are stored on each unit	addResources: setFlag=1
	unsetFlag		use with addResources, resourceUsage or price. 0-31	
	hasFlag		use with price or resourceUsage	
	hasMissingFlag		use with price or resourceUsage	
1.13.3	X		Any resource defined in [global_resource_x] or [resource_x] sections	gold=5, stone=21, credits=2001

	Section		[global_resource_NAME]	
	Code	Value Type	Description	Example
	**===== **	**#===== **	**Define a new resource shared with all units in a team, works just like the built-in credits resource. Add to 'all-units.template' (at mod root) for easy use in all of your mods**	
1.13.3	displayName	LocaleString	Name of this resource in UI	displayName: Crude Oil
1.13.3	displayNameShort	LocaleString	Resource name on smaller UI elements like action hovertext (Defaults to displayName)	displayNameShort: Oil
1.13.3	hidden	bool	Hide the resource from the player	hidden: false
1.13.3	priority	float	If 2 or mods/units define a resource with the same NAME, the displayName/displayColor with the highest priority is used	priority: 0.5
1.14	displayColor	color	Color, can be hex with optional alpha	displayColor: #FF0000
1.14	displayRoundedDown	bool	Don't show decimal places to the player	displayRoundedDown:
1.13	displayTextPrefix	string	Adds a string before the resource value	displayTextPrefix: -
1.13	displayTextPostfix	string	Adds a string after the resource value	displayTextPostfix: x
	displayTextAppendResource	resource	Appends another resource after this resource. It's recommended to give a "hidden" key on the appended resource to hide duplicate.	displayTextAppendResource: sauce
1.13	displayWhenZero	bool	Displays the resource even when empty. Set as false by default	displayWhenZero: true
	displayPos	int	Sets the position of the globe resource in the screen	displayPos: 1
1.14	iconImage	image	Shows a custom icon for the specific resource in the HUD and in text	iconImage: icon_oil.png
1.14	iconImageUseInText	bool	Default as true; Shows resource icon in action description	iconImageUseInText: false
1.14	displayNameHideWhenIconShownInText	bool	Default as false	displayNameHideWhenIconShownInText: true
1.14	displayNameHideWhenIconShownInHUD	bool	Hides the icon in menus referring to resource when true. Default as false	displayNameHideWhenIconShownInHUD: true
1.14	displayColorUseInText	bool	Shows color in action description	displayColorUseInText: false
1.14	displayInHud	bool	For resources used in **appendResourceHUD** that shouldn't be **hidden: true**. Defaults true.	displayInHud: false
	appendResourceInHUD	resource	stacks another resource specified after this resource on the HUD.	appendResourceInHUD: sauce
1.14	appendResourceInHUD_whenThisZero	bool	Defaults true. When set as false, allows appended resource to be hidden with the parent resource.	appendResourceInHUD_whenThisZero: true
1.13	displayPrefixInHUD	string	Displays text to show before resource value - replaces resource name and removes the colon separator	displayPrefixInHUD: -
1.14	displayPostfixInHUD	string	Similar to the prefix counterpart, but is displayed after the resource value	displayPostfixInHUD: x
1.13	valueInStats	float	Affects post game stats and replay leaderboard. Defaults to 1.	valueInStats: 0.5
1.14	displayTextAppendResourceWithGap	bool	Adds a space between this and the appended resource. Defaults as false.	displayTextAppendResourceWithGap: true
1.15p9	displayDigitGrouping	enum	Sets the symbol for separating place units in: none, comma, space	displayDigitGrouping: comma

	Section		[resource_NAME]	
	Code	Value Type	Description	Example
	**===== **	**#===== **	**Define a new resource local to unit. Works like the built-in ammo resource**	
1.13.3	displayName	LocaleString	Name of this resource to UI (eg. hovering over unit info)	displayName: Sauce
1.13.3	displayNameShort	LocaleString	Shorter version of resource name used on some text ui	displayNameShort: Sus
1.13.3	hidden	bool	Hide the resource from the player	hidden: true
	equivalentGlobalResourceForAi	resource	Used to hint to the AI that a resource node with a local resources could be used to get a different global resource. Eg. when a harvester unloads the resource	equivalentGlobalResourceForAi: universal_sauce

Version	Key	Value Type	Description	Example
1.14	displayRoundedDown	bool	Rounds off resource values with decimals	displayRoundedDown: true

Type	value type	[decal_name]	Example

Version	Key	Value Type	Description	Example
1.15p9	# Graphical #			
1.15p9	image	image	Takes a single image for the decal - NOT TO BE CONFUSED WITH IMAGE STACK	image: fish.png
1.15p9	layer	enum	Sets the layer of the decal, values from lowest to highest - shadow, beforeBody, afterBody, onTop, beforeUI, inactive	layer: beforeUI
1.15p9	order	float	Defaults as 0, sets a more discrete layering if 2 or more decals takes same layer type. Otherwise order in INI file will be used	order: 2
1.15p9	teamColors	bool	When true, automatically reassigns team color to respective teams	teamColors: true
1.15p9	alpha	logicNumber	0-1, sets transparency of decal (images or line)	alpha: 0.5
1.15p9	isVisible	LogicBoolean	Dynamic Value, draws the decal on a specific condition	isVisible: if self.hp > self.maxHp/2
1.15p9	# Player Conditions #			
1.15p9	onlyWhenSelectedByOwnPlayer	bool	Defaults as false, displays the decal only if the owner player clicks on the unit	onlyWhenSelectedByOwnPlayer : true
1.15p9	onlyWhenSelectedByEnemyPlayer	bool	Defaults as false, displays the decal only if the enemy player clicks on the unit	onlyWhenSelectedByEnemyPlayer: true
1.15p9	onlyWhenSelectedByAllyNotOwnPlayer	bool	Defaults as false, displays the decal only if the allied player clicks on the unit	onlyWhenSelectedByAllyNotOwnPlayer: true
1.15p9	onlyWhenSelectedByAnyPlayer	bool	Defaults as false, displays the decal only if any player clicks on the unit	onlyWhenSelectedByAnyPlayer: true
1.15p9	includeParentsSelection	bool	Useable with onlyWhenSelectedBy[insert player], will also check parent's selection when true. Useful with attachments.	includeParentsSelection: true
1.15p9	# "Only if" Conditions #			
1.15p9	onlyTeam	TeamRelation	Draws the decal if value matches the player from the following: own, notOwn, ally, allyNotOwn, enemy, any	onlyTeam: any
1.15p9	onlyWhileActive	bool	Draws the decal only if the unit is fully built	onlyWhileActive: true
1.15p9	onlyWhileAlive	bool	Draws the decal only if the unit is alive (If beforeUI layer default true, else default false)	onlyWhileAlive: false
1.15p9	onlyInPreview	bool	Only show in sidebar, and building placement preview	onlyInPreview: false
1.15p9	onlyWithZoomLevelOrMore	float	Draws the decal only if the zoom level matches or exceeds it, very useful for making 3D units more optimized.	onlyWithZoomLevelOrMore: 0.4
1.15p9	onlyOnNonPreview	bool	Draws the decal on the unit itself, not on the preview from building or in sidebar interface	onlyOnNonPreview: true
1.15p9	onlyPlayersWithUnitControl	bool	Draws the decal if a player that selects it has control (that includes shared units in multiplayer)	onlyPlayersWithUnitControl: true
1.15p9	onlyOnBodyFrameOf	int	Only draw decal when body frame is equal to this	onlyOnBodyFrameOf: 0
1.15p9	# Offsets (Positioning) #			
1.15p9	xOffsetRelative	int	Sets horizontal offset relative to the unit	xOffsetRelative: 50
1.15p9	yOffsetRelative	int	Sets vertical offset relative to the unit	yOffsetRelative: 50
1.15p9	xOffsetAbsolute	logicNumber	Dynamic value	xOffsetAbsolute: 40
1.15p9	yOffsetAbsolute	logicNumber	Dynamic value	yOffsetAbsolute: 40
1.15p9	hOffset	int	Sets the height of the whole decal	hOffset: 10
1.15p9	dirOffset	int	Sets the direction of the whole decal.	dirOffset: 45
1.15p9	pivotOffset	int	only affects relative offsets without rotating image	pivotOffset: 45
1.15p9	alwaysStartDirAtZero	bool	Keeps the decal on north direction regardless of the unit's direction value, useful for custom in-unit interfaces	alwaysStartDirAtZero: true
1.15p9	# Image specifics / Stacks #			
1.15p9	imageStack	images	Takes one or more image with occasional multipliers for stacking. Also useful for using image stack from a MagicaVoxel slice export	Multiple image file stack: imageStack: carFloor.png, carBody.png*3, carRoof.png*2, antenna.png*10 Stack from a sprite sheet - imageStack: carModel.png*14
1.15p9	imageScale	logicNumber	Dynamic value, scales the decal	imageScale: 1 + (self.height * 0.1)
1.15p9	imageScaleX	logicNumber	Dynamic value	imageScaleX: 1
1.15p9	imageScaleY	logicNumber	Dynamic value	imageScaleY: (self.hp/self.maxHp)*100
1.15p9	stack_hOffset	float	Sets the height of every layer in a stack. 1 layer = 1 pixel. Can take negative values.	stack_hOffset: 4
1.15p9	stack_frameOffset	int	Useful for making 3D units, frame to offset by in imageStack. Often a value of 1 is useful in a sprite sheet.	stack_frameOffset: 1
1.15p9	stack_drawInReverseOrder	bool	Renders the stack in the opposite way - last frame goes on bottom, first frame goes to top	stack_drawInReverseOrder: true
1.15p9	stack_indexStart	logicNumber	Offset to start drawing images in the image stack	stack_indexStart: 0
1.15p9	stack_indexCount	logicNumber	Number of images in the image stack to draw	stack_indexCount: 10
1.15p9	total_frames	int	Sets the number of frames in an image from an imageStack that uses a spritesheet.	total_frames: 10
1.15p9	frame_width	int	Sets the frame width in the decal stack	frame_width: 20
1.15p9	frame_height	int	Sets the frame height in the decal stack	frame_height: 40
1.15p9	frame	logicNumber	Dynamic value, useful for animations	frame: memory.frame
1.15p9	addBodyFrameMultipliedBy	int	Add body frame number to this decal when set to 1.	addBodyFrameMultipliedBy: 2
1.15p9	# Shadows #			
1.15p9	image_shadow	image	Sets shadow for the decal	image_shadow: AUTO
1.15p9	shadowOffsetX	int	Sets the horizontal position of the decal's shadow	shadowOffsetX: 0
1.15p9	shadowOffsetY	int	Sets the vertical position of the decal's shadow	shadowOffsetY: 0
1.15p9	# Marker Positioning #			
1.15p9	basePosition	marker	Attaches the decal on the specified marker, use "self" if attaching it to the unit itself	basePosition: self
1.15p9	basePositionFromLegEnd	legArm	Attaches the decal to the specified leg or arm ending.	basePosition: memory.launchPoint basePositionFromLeg: leg_3, basePositionFromLeg: arm_5
1.15p9	basePositionFromTurret	turret	Attaches the decal to the specified turret	basePositionFromTurret: rocketLauncherBase
1.15p9	# Waypoint Lines #			
1.15p9	drawLineTo	marker ref	Draws a line from the unit to the specified marker, useful for custom waypoints	drawLineTo: attacking.customTarget2
1.15p9	color	color hex	Sets the color of the drawn line	color: #ff00
1.15p9	lineWidth	float	Sets the width of the drawn line	lineWidth: 2

Section		[comment_NAME]	
Code	Value Type	Description	Example
#****	#****	Comment sections can have any keys and have no effect.	
		List of default game noises (for turret fire sound or other sound references):	bug_attack, bug_die, building_explode, cannon_firing, click, click_add, click_remove, firing3, firing4, gun_fire, interface_error, large_gun_fire1, large_gun_fire2, laser_deflect, laser_deflect2, lighting_burst, message,

Section		[template_NAME]	
Code	Value Type	Description	Example
1.13.3	#****	#****	Template sections can have any keys and have no effect by themselves.	
	#****	#****	Template can get included from other files with [core] copyFrom. Eg: [core]copyFrom: ROOT: effects/explodeEffects.template (Note that copyFrom can include multiple files.)	
	#****		--All these below features can be used with any section not just templates--	
1.13.2	@copyFromSection		Use in any section to include keys from a section or template (Comma separated for multiple)	@copyFromSection: template_name/action_name/projectile_name
	@copyFrom_skipThisSection		Use in any section to make [core]copyFrom not copy into it. Eg not copy an action when overriding.	@copyFrom_skipThisSection
	@define X		Define a local variable within a section (best outside of template)	@define targetEffect: boom
	@global X		Define a global variable used in all sections. Local variables have a higher priority	@global targetEffect: pop
1.13.3	@memory X		Define a memory for this unit, its type must be defined as well. $[X] can be used to reference variables (can also be done outside of a template). It is calculated when loading and remains static. Has no impact on runtime performance.	@memory transportCount: float spawnEffects: effect_$[targetEffect]
1.13.3			$[section.key] can be used to reference another key (can also be done outside of a template).	addResources: credits=$[core.price * 2 + 10]
0.7x			%[X] can be used to add dynamic logic into some strings. (CAN NOT be used everywhere). The value will update every frame.	[action]text: Missing hp %[self.maxHp - self.hp]
0.12			""" text """ can be used for multiline strings. Newlines with be removed from final result.	copyFrom:""" ROOT:a.ini, ROOT:b.ini, ROOT:c.ini """
	File	NAME.template	A template that can be used to share many parts across units, used with copyFrom However, cannot define variables	copyFrom:ROOT:templates/cruiser.template, copyFrom:ROOT: templates/infantry.template, copyFrom:ROOT:templates/hq. template,

Section	Key	Value type	Description	Example
Any				
	File	all-units.template	Put at the root of the mod folder, applies anything in it to ALL units across all .ini files, can have any sections but cannot define variables	[core] explodeTypeOnDeath: large energyMax: 26

Section	Key	Value type	Description	Example
Any				
	File	mod-info.txt	This file defines some of properties of a mod for display in mod browser and in Steam Workshop. It should be placed in the top of mod's file hierarchy, otherwise will result in error	
Section	Key	Value type	Description	Example
	title	String	Sets the title of the mod package	title: Sample mod
[mod]	description	String	Sets the description of the mod package. Doesn't support line breaks.	description: This is a sample of mod that samples a unit to the game
	tags	String	Sets varied tags depending on what is specified, can be multiple with comma as separator	tags: units, sample, demo
	minVersion	String	Declares the minimum version compatible for the mod. It is important to declare one when using decals and other modding	minVersion: 1.15p9
	thumbnail	image	Sets the mod's thumbnail in Steam Workshop	thumbnail: assets/images/otherthumb.png
[music]	sourceFolder	audio dir	Sets where the custom music will be played for the mod	sourceFolder: assets/audio/music
	whenUsingUnitsFromThisMod_playExclusively	bool	Only plays music from this mod	whenUsingUnitsFromThisMod_playExclusively: true

	Projectile Draw Types		In [projectile_#], a unit may use a built-in frame from the res/drawable directory. However, only three strips are designated for this use case. Use this value for the drawType key and a frame number for a desired appearance.	If there is no image or frame supplied, the game will display a white dot instead, wherein a modder can change its color using the color key. This is observable in Command Center, Heavy Tank, and Missile Tank
Id	name	usable frames	image (with frame numbers)	
0	projectiles.png	12	0 1 2 3 4 5 6 7 8 9 10 11	
1	projectiles_large.png	3	0 1 2	
2	projectiles2.png	6	0 1 2 3 4 5	