Item modification

> Item modification is a startup event.

`ItemEvents.modification` is a startup script event used to modify various properties of existing items.

```
ItemEvents.modification(event => {
  event.modify('minecraft:ender_pearl', item => {
    item.maxStackSize = 64
    item.fireResistant = true
    item.rarity = "UNCOMMON"
  })
  event.modify('minecraft:ancient_debris', item => {
    item.rarity = "RARE"
    item.burnTime = 16000
  })
  event.modify('minecraft:turtle_helmet', item => {
    item.rarity = "EPIC"
    item.maxDamage = 481
    item.craftingRemainder = Item.of('minecraft:scute').item
  })
})
```

Available properties:

Property	Value Type	Description
maxStackSize	int	Sets the maximum stack size for items. Default is 64 for most items.
maxDamage	int	Sets the maximum damage an item can take before it is broken.
craftingRemainder	Item	Sets the item left behind in the crafting grid when this item is used as a crafting ingredient (like milk buckets in the cake recipe). Most items do not have one.

Property	Value Type	Description
fireResistant	boolean	If this item burns in fire and lava. Most items are false by default, but Ancient Debris and Netherite things are not.
rarity	Rarity	Sets the items rarity. This is mainly used for the name colour. COMMON by default. Nether Stars and Elytra are UNCOMMON, Golden Apples are RARE and Enchanted Golden Apples are EPIC.
burnTime	int	Sets the burn time (in ticks) in a regular furnace for this item. Note that Smokers and Blast Furnaces burn fuel twice as fast. Coal is 1600.
foodProperties	FoodProperties	Sets the items food properties to the provided properties. Can be `null` to remove food properties.
foodProperties	Consumer<FoodBuilder>	Sets the properties according to the consumer. See below for more info.
digSpeed	float	Sets the items digging speed to the number provided. See table below for defaults.
tier	Consumer<MutableToolTier>	Currently BROKEN! Sets the tools tier according to the consumer. See below for more info.
attackDamage	double	Sets the attack damage of this item.
attackSpeed	double	Sets the attack speed of this item
armorProtection	double	Sets the armor protection for this item. 20 is a full armour bar.
armorToughness	double	Adds an armor toughness bonus.
armorKnockbackResistance	double	Add an armor knockback resistance bonus. Can be negative. 1 is full knockback resistance.

Tool defaults

Tier	level	maxDamage	digSpeed	attackDamage (this is a bonus modified by the tool type value, not the final value)	enchantmentValue
Wood	0	59	2	0	15
Stone	1	131	4	1	5
Iron	2	250	6	2	14
Diamond	3	1561	8	3	10
Gold	0	32	12	0	22
Netherite	4	2031	9	4	15

Armor defaults

All boxes with multiple values are formatted [head, chest, legs, feet]. Boxes with single values are the same for every piece.

Tier	maxDamage	armourProtection	armorToughness	armorKnockbackResistance
Leather	[65, 75, 80, 55]	[1, 2, 3, 1]	0	0
Chain	[195, 225, 240, 165]	[1, 4, 5, 2]	0	0
Iron	[195, 225, 240, 165]	[2, 5, 6, 2]	0	0
Gold	[91 ,105, 112, 77]	[1, 3, 5, 2]	0	0
Diamond	[429, 495, 528, 363]	[3, 6, 8, 3]	2	0
Turtle (only has helmet)	[325, nil, nil. nil]	[2, nil, nil, nil]	0	0
Netherite	[481, 555, 592, 407]	[3, 6, 8, 3]	3	0.1
Elytra (not actually armor)	[nil, 432, nil, nil]	0	0	0

Tier

> Broken at the moment! https://github.com/KubeJS-Mods/KubeJS/issues/662. Use the non tier methods instead.

Tools

```
ItemEvents.modification(event => {
  event.modify('golden_sword', item => {
    item.tier = tier => {
        tier.speed = 12
        tier.attackDamageBonus = 10
        tier.repairIngredient = '#forge:storage_blocks/gold'
        tier.level = 3
    }
  })
  event.modify('wooden_sword', item => {
    item.tier = tier => {
        tier.enchantmentValue = 30
    }
  })
})
```

Property	Value Type	Description
uses	int	The maximum damage before this tool breaks. Identical to maxDamage.
speed	float	The digging speed of this tool.
attackDamageBonus	float	The bonus attack damage of this tool.
level	int	The mining level of this tool.
enchantmentValue	int	The enchanting power of this tool. The higher this is, the better the enchantments at an Enchanting Table are.
repairIngredient	Ingredient	The material used to repair this tool in an anvil.

Armor

Doesnt actually exist/work at the moment. Sorry.

Food

```
ItemEvents.modification(event => {
  event.modify('minecraft:diamond', item => {
```

```
    item.foodProperties = food => {
        food.hunger(2)
        food.saturation(3)
        food.fastToEat(true)
        food.eaten(e => e.player.tell('you ate')) // this is broken, use ItemEvents.foodEaten
instead.
    }
  })

  event.modify('pumpkin_pie', item => {
    item.foodProperties = null // make pumpkin pies inedible
  })
})
```

Method	Parameters	Description
hunger	int h	Sets the hunger restored when this item is eaten
saturation	float s	Sets the saturation mulitplier when this food is eaten. This is not the final value, it goes through some complicated maths first
meat	boolean flag (optional, true by default)	Sets if this item is considered meat. Meat can be fed to wolves to heal them.
alwaysEdible	boolean flag (optional, true by default)	If this item can be eaten even if your food bar is full. Chorus Fruit has this true by default.
fastToEat	boolean flag (optional, true by default)	If this item is fast to eat, like Dried Kelp.
effect	ResourceLocation mobEffectId, int duration, int amplifier, float probability	Adds an effect to the entity who eats this, like a Golden Apple
removeEffect	MobEffect mobEffect	Removes the effect from the entity who eats this, like Honey Bottles (poison).
eaten	Consumer<FoodEatenEventJS> e	BROKEN! Use ItemEvents.foodEaten in server scripts instead.

Revision #8
Created 2023-07-21 10:32:11 UTC by ChiefArug
Updated 2023-09-03 20:23:36 UTC by ChiefArug