

Modpack-Entwicklung

Ein Blick auf die Modpack-Entwicklung

📅 8. Mai 2017 (<http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/>) 👤 seb (<http://sebinside.de/author/seb/>) ➦ Coding (<http://sebinside.de/category/coding/>), Minecraft (<http://sebinside.de/category/minecraft/>)

Die Modding-Community von Minecraft ist riesig. Allein die Download-Zahlen der Modpacks vom „Feed The Beast“-Team befinden sich im Millionenbereich. Auch das von mir entwickelte Modpack „Minecraft SKY“ (<https://www.technicpack.net/modpack/minecraft-sky-official.959027>), was ausschließlich für die deutschsprachige Minecraft-Community entworfen wurde, weist schon mehr als 250.000 Downloads auf. Es ist an der Zeit, mal einen Blick darauf zu werfen, was eigentlich ein Modpack ausmacht – und was alles zur Herstellung eines solchen notwendig ist.

Von vorne herein möchte ich zwei Dinge gesagt haben:

1. Das hier ist **kein** detailliertes Tutorial. Ich möchte nur grob zeigen, welche Schritte notwendig sind, um ein Modpack zu entwerfen. Die technische Umsetzung im Detail wird nicht erklärt. Hierfür gibt es Ressourcen im Internet, oder wenn ihr wollt, bald eine **Videoserie auf meinem Kanal**.
2. Eigentlich habe ich **auch keine Ahnung**, was ich hier mache. Minecraft SKY war mein erster Versuch, ein „erwachsenes“ Modpack auf die Beine zu stellen. Alles

Wissen hierfür habe ich mir selbst beigebracht. Und meiner Meinung nach sollte Minecraft SKY dringend von Grund auf erneuert werden...

Eigentlich ist alles ein Hack

Eigentlich sollte Modding in Minecraft gar nicht funktionieren. Oder zumindest war es ursprünglich weder vorgesehen, noch existiert bis heute für die Java-Version eine **offizielle API**. Dennoch haben schon vor vielen Jahren Modder damit begonnen, Teile von Minecraft um bzw. neuzubauen.

Um damals Mods zu installieren, musste man noch von Hand die Hauptdatei von Minecraft öffnen und Teile ergänzen oder austauschen. Natürlich ist dieser Weg weder skalierbar noch stabil, weswegen das **Minecraft-Forge-Projekt** ins Leben gerufen wurde. Heute ist Forge die Grundlage der modernen Mod-Entwicklung.

(Um ganz korrekt zu sein: Forge war nicht der erste Versuch einer Modding-Umgebung. Zu Zeiten der Alpha-Version wurde das Mod Coder Pack von Searge entwickelt, der heute bei Mojang arbeitet. Danach kam ModLoader, der allerdings Konflikte nicht ausreichend auflösen konnte. Hieraus ist dann aus Zusammenarbeit diverser Mod-Entwickler Minecraft Forge entstanden.)

Minecraft Forge ist ein **Modpack-Framework**. Die Software ermöglicht es, eigenen Code einzufügen, Abhängigkeiten zwischen Mods aufzulösen und Konflikten vorzubeugen. Hinzu kommen zahlreiche Hilfsfunktionen, wie z.B. eine Schnittstelle zur Speicherung von Mod-Einstellungen in Config-Dateien. Die Einrichtung von Forge ist einfach: Entweder per Installer von der Forge-Website (<http://files.minecraftforge.net/>) oder auch durch Hilfssoftware wie dem MultiMC Launcher (<https://multimc.org/>).

Die Idee

Am Anfang der Modpack-Entwicklung steht natürlich die **Idee**. Für welche **Zielgruppe** soll das Modpack entstehen? Welche **Story** liegt dem Modpack zugrunde? **Welche Mods** sollten unbedingt dabei sein? Wie passen diese zueinander? Und überhaupt: Welche **Minecraft-Version** kann verwendet werden? Nichts ist langweiliger, als ein Modpack voll mit wahllos zusammen gewürfelten Mods, ohne Anpassungen oder Hintergrundgedanken.

Für jeden erdenklichen Bereich in Minecraft gibt es heute zahlreiche Mods, aber nicht alle eignen sich für Modpacks. Wegen der großen Gemeinde an Entwicklern gibt es außerdem verschiedene Mods mit ähnlicher Funktionalität aber Unterschieden in Qualität und Umfang. Hinzu kommt, dass hinter den Modpacks natürlich separate Entwickler stehen, die sich manchmal etwas länger Zeit lassen, ihre Mods weiter zu entwickeln oder zu aktualisieren (so wie ich, lol).

Mögliche Ideen für Modpacks, die so auch schon umgesetzt wurden: „Ein Hardcore-Skyblock-Modpack“, „Ein Einsteiger-Tutorial-Pack“, „Ein Challenge-Pack mit speziellen Zielen“, „Ein Total-Conversion-Modpack mit zahlreichen Modifikationen an der Spielmechanik“. **Die Möglichkeiten sind tatsächlich nahezu unbegrenzt.**

Die Idee von „Minecraft SKY“ war: „Ein mit der Zeit wachsendes Skyblock-Modpack mit veränderten Rezepten und erhöhtem Schwierigkeitsgrad, aber nicht so schwer wie FTB Expert Mode“. Die größte Herausforderung war hierbei das „Mit der Zeit wachsen“, aber dazu später.

Die Auswahl der Mods

Sobald die Idee feststeht und zumindest grob skizziert wurde, geht es los mit der **Wahl der Mods**. An dieser Stelle muss zunächst die Minecraft-Version festgelegt werden. Wir wollten „Minecraft SKY“ ursprünglich in der 1.10 spielen lassen, aber leider gab es zu dieser Zeit noch kein „**Ex Nihilo**“ für neuere Versionen als 1.7.10.

Die Auswahl der Mods hat eine **große Auswirkung** auf die spätere Spielerfahrung. Die Entscheidung sollte sehr gründlich getroffen werden und nicht nur die eigenen Lieblings-Mods berücksichtigt. Natürlich kann man sich an dieser Stelle ohne Probleme von anderen Packs **inspirieren** lassen, die kochen alle nur mit Wasser.

Außerdem können sich durch falsche Kombination von Mods **Performance-Probleme**, schwere Bugs und Abstürzte mit Welt-Verlust oder potenzielle Exploits ergeben. Ersteres bekommt man nur durch Erfahrung, Testen oder Austausch von Mods geregelt. **Im Fall von Bugs** hilft vielleicht auch schon die Anpassung der Konfiguration. Typische Probleme sind Konflikte unter Verzauberungen oder Dimensionen, die wegen gleicher ID sich gegenseitig überschreiben. **Exploits** lassen sich meist ebenfalls durch Konfiguration oder Anpassung von Rezepten vermeiden. Hier liegt die Problematik eher beim Finden, als beim Beheben.

Konfiguration

Sobald eine Grundausswahl von Mods getroffen wurde und sichergestellt, dass diese immerhin ansatzweise miteinander klarkommen, folgt die **Konfiguration**. Die meisten Mods lassen sich mehr oder weniger genau konfigurieren, indem man die Einstellungsdateien im config-Ordner bearbeitet.

Typische Aufgaben sind das eben besprochene Beheben von Fehlern und Erhöhen der Kompatibilität untereinander. Ebenfalls kann man oft Werte wie den Energie- oder Mana-Verbrauch anpassen, bis diese zur Idee des Modpacks passen. Natürlich sollte man es hier nicht übertreiben. Ein Modpack wird nicht anfordernder, indem man unermesslich viel Energie braucht – es wird nur **nerviger** (SKY ist hier eine Ausnahme, weil genau dieser Aufwand gefordert wurde. Rückblickend hätte man manches aber auch besser lösen können). Der Fokus sollte vor allem die **Anpassung der Mods untereinander** sein.

Hier als Beispiel ein Auszug der Einstellungsmöglichkeiten von **Ex Nihilo 1.7**:

- Aktivieren und deaktivieren einzelner Barrel-Rezepte
- Einstellen, ob sich Barrels bei Regen mit Wasser füllen

- Kompatibilität mit Agriculture, Industrial Craft und Thermal Expansion
- Aktivieren und Deaktivieren einzelner Mechaniken und Items
- Detaillierte Einstellung der Drop-Chancen aller Items

Manche Mods gehen noch wesentlich weiter: Dort lässt sich jeder einzelne Wert im Detail festlegen. Ender IO beispielweise bietet sogar eigene Funktionalität zur Anpassung aller Crafting Rezepte und Energie-Werte, ohne externe Mods. Andere Mods wiederum lassen sich quasi gar nicht konfigurieren – was bei der Modpack-Erstellung echt hart nerven kann.

Anpassung der Rezepte

Der Schritt der Anpassung der Rezepte ist absolut **optional**. Es gibt auch viele erfolgreiche Modpacks, welche die Rezepte gar nicht verändert haben. Im Normalfall lassen sich Rezepte auch nicht ohne weiteres verändern; hierzu benötigt es zusätzliche Mods wie z.B. MineTweaker und ModTweaker.

Bei der Anpassung von Rezepten muss man zwischen zwei Bereichen unterscheiden: Der **Vermeidung von Exploits** bzw. Balancierung verschiedener Mods und der **vollständigen Veränderung** von Rezepten zur Erhöhung des Spielspaßes. Ein Beispiel für einen typischen Exploit: In der „Minecraft SKY“-Mod-Kombination konnte man Uhren sowohl aus Aluminium als auch Gold herstellen. Durch Einschmelzen konnte man aus einer Uhr Gold gewinnen. Sprich: Aus Aluminium lässt sich Gold herstellen. Eine ungewollte, zu starke Funktionalität. Die Lösung: Entweder das Einschmelzen vermeiden, oder das Herstellen vermeiden.

Alles was über solches Balancing hinaus geht, ist ein **gezielter Eingriff in die Spielmechanik**. Ein Beispiel hierfür sind Total-Conversion-Modpacks wie „Regrowth“ oder Hardcore-Modpacks wie „FTB Infinity Evolved Expert Mode“. Hier wurden hunderte Rezepte ausgetauscht um komplett neue Funktionalität einzufügen. Das Ziel hierbei war, dass man alle Mods kombinieren muss, um bessere Items und Tools zu bekommen. Ebenfalls wurden Rezepte für Items hinzugefügt, welche üblicherweise nur im Creative Mode zur Verfügung stehen.

Wie sieht hierbei die technische Seite aus? **MineTweaker** bringt zur Anpassung von Rezepten eine „eigene“ JavaScript-ähnliche Programmiersprache mit. Hiermit lassen sich Crafting-Rezepte im Detail deaktivieren und umschreiben. Außerdem lassen sich auch Rezepte von vielen Maschinen diverser Mods anpassen. Das ist aber wiederum eine komplett eigene Welt. Glücklicherweise gibt es in der offiziellen Wiki (<http://minetweaker3.powerofbytes.com>) sehr viele gute Ressourcen für den Einstieg!

Eigene Entwicklung

Was ist, wenn sich eine Mod nicht genau genug konfigurieren lässt? Was ist, wenn gewisse Funktionalität oder Items nicht existieren? Hierfür gibt es zwei Möglichkeiten: Entweder eine Alternative finden oder sich selbst an die Entwicklung setzen.

Viele bekannte Mods sind unter einer **Open-Source-Lizenz** veröffentlicht. Grob gesagt bedeutet dies, dass man den Quellcode anpassen darf und ihn neu veröffentlichen. Natürlich ist hierfür entsprechendes Know-How und Zeitaufwand notwendig. Wir haben für „Minecraft SKY“ beispielweise den Code von Ex Compressum überarbeitet, damit unsere eigenen Skins in den Maschinen angezeigt werden.

Testen

Die Idee liegt schon eine gefühlte Ewigkeit zurück, die Mod-Auswahl steht ebenfalls fest. Alle Konfigurationen wurden angepasst und eigentlich sollten auch alle veränderten Rezepte funktionieren. Außerdem sind garantiert keine Exploits mehr im Modpack enthalten. **Denkst du.**

Das **Testen** eines Modpacks ist mindestens genauso wichtig, wie die eigentliche Entwicklung. Egal wie gut du dich mit den einzelnen Mods auskennst, es wird immer jemanden geben, der mehr weiß und **dein Modpack auseinandernimmt**. Das FTB-

Team hat hierfür beispielweise mehrere Test-Teams, geschlossene wie auch offene, die die Vorversionen der Modpacks vor dem Release testen.

Release und Updates

Nachdem das Modpack ausreichend getestet wurde, kann es **veröffentlicht** werden. Üblicherweise ist hier für noch etwas Artwork notwendig, etwa ein Logo oder ein eigenes Hauptmenü.

Zur Wahl der besten **Plattform** kann ich wenig sagen. Natürlich kann man auch einfach eine ZIP-Datei hochladen, aber normalerweise nimmt man hierfür einen Custom Launcher. Eine Möglichkeit ist die Curse-Plattform; den Technic Launcher kann ich aus eigener Erfahrung definitiv nicht empfehlen.

Auch nach dem Release können natürlich noch Unstimmigkeiten und Fehler enthalten sein. Diese müssen dann in Form von Updates behoben werden.

Ein Kommentar zu Minecraft SKY

Damit sind wir am Ende des Vorgehens angekommen. Das sind die üblichen Schritte zur Erstellung eines Modpacks. Wer das auch mal von Profis hören möchte, sollte sich diesen (<https://forum.feed-the-beast.com/threads/principles-in-design-of-ftb-modpacks.192873/>)Blogpost vom „Feed The Beast“-Team durchlesen.

Zum Schluss möchte ich noch einen Kommentar zu „*Minecraft SKY*“ geben. Rückblickend kann ich sagen, dass ich heute vieles anders gemacht hätte. Das Modpack war mein **erster, eigener Versuch**. Zu Beginn der Entwicklung hatte ich noch absolut keine Ahnung, wie man Rezepte richtig anpasst; weder von technischer Seite, noch aus Sicht eines Gamedesigners.


Hinzu kommt der **Zeitdruck** bei der Entwicklung. Wir hatten vor Release keine Möglichkeit, alles ausreichend zu testen, weswegen viele Fehler erst zur Laufzeit des Projekts sichtbar wurden. Ebenfalls ist das Konzept eines mit der Zeit **wachsenden Modpacks** ziemlich... bescheiden. Denn durch jeden Update-Schritt verändern sich Rezepte. Das bedeutet neue Bugs und neue Exploits, also nahezu den gleichen Testaufwand wie beim initialen Release. Ich muss nicht dazu sagen, dass natürlich auch hierfür die Zeit gefehlt hat.

Auch wenn das Modpack für einen ersten Versuch, meiner Meinung nach, gar nicht so übel war und ich viele Konzepte relativ cool finde, sollte man das Modpack eigentlich nochmal **komplett neu** von Grund auf entwickeln. Hierbei könnte man ebenfalls viel mehr Kommunikation der einzelnen Mods fördern, da die Liste mittlerweile final ist.

Aber das ist alles Zukunftsmusik. Durchaus möglich wäre eine Kombination aus **Tutorial-Serie** und Neuentwicklung von „Minecraft SKY“. Ihr könnt mir gerne eure Ideen und Vorstellungen hier lassen, mal sehen was ich daraus umsetze. Ansonsten hoffe ich, dass dieser Einblick in die Entwicklung von Modpacks spannend war, bis bald!

Teilen mit:

 E-Mail (<http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/?share=email&nb=1>)

 Facebook 1 (<http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/?share=facebook&nb=1>)

 Twitter (<http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/?share=twitter&nb=1>)

Gefällt mir:

Lade ...

[feed the beast](http://sebinside.de/tag/feed-the-beast/) (<http://sebinside.de/tag/feed-the-beast/>) [minecraft](http://sebinside.de/tag/minecraft/) (<http://sebinside.de/tag/minecraft/>)

[minecraft novus](http://sebinside.de/tag/minecraft-novus/) (<http://sebinside.de/tag/minecraft-novus/>)

[minecraft sky](http://sebinside.de/tag/minecraft-sky/) (<http://sebinside.de/tag/minecraft-sky/>) [minetweaker](http://sebinside.de/tag/minetweaker/) (<http://sebinside.de/tag/minetweaker/>)

[modding](http://sebinside.de/tag/modding/) (<http://sebinside.de/tag/modding/>) [modpack](http://sebinside.de/tag/modpack/) (<http://sebinside.de/tag/modpack/>)

[skate702](http://sebinside.de/tag/skate702/) (<http://sebinside.de/tag/skate702/>)

5 Gedanken zu „Ein Blick auf die Modpack-Entwicklung“

WOLF066LP ([HTTP://WWW.WOLF066LP.DE](http://www.wolf066lp.de))

8. Mai 2017 um 16:02 Uhr (<http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/#comment-107>)

Sehr interessantes Thema! Würde mich über eine Einstiegerserie schon freuen... Dich code zu sehen ist einfach immer spannend ;D egal zu welchem Thema!

ANTWORTEN ([HTTP://SEBINSIDE.DE/2017/05/08/EIN-BLICK-AUF-DIE-MODPACK-ENTWICKLUNG/?REPLYTOCOM=107#RESPOND](http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/?replytocom=107#respond))

HANNES

22. Mai 2017 um 10:37 Uhr (<http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/#comment-110>)

Es wäre echt cool, wenn du zu solchen Themen Tutorials machen könntest. Mich interessiert besonders das Programmieren von Websites aber es wäre auch mal interessant zu sehen, wie man sein eigenes Programm schreiben kann. Wenn du so etwas mal machen könntest, wäre das echt cool.

ANTWORTEN ([HTTP://SEBINSIDE.DE/2017/05/08/EIN-BLICK-AUF-DIE-MODPACK-ENTWICKLUNG/?REPLYTOCOM=110#RESPOND](http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/?replytocom=110#respond))

KORSA92

29. Mai 2017 um 17:15 Uhr (<http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/#comment-111>)

Also Skate man muss sagen das du nicht viel Ahnung von den Modpack Sky hattest ist nicht weiter schlimm ich selber habe auch schon ein Modpack gemacht und ich weiß wie es alles vernünftig zusammen zu würfeln.

Trotz möchte ich dir sagen das Sky mir sehr viel Spaß gemacht und mich einige Herausforderung gekostet hat es zu spielen.

Allerdings ist der große Fehler so wenig Endgame Mods hinzuzufügen das wäre der einzige punkt wo ich spontan dir sagen würde was du ändern musst aber das weißt du sicherlich selber.

Ich freue mich schon riesig auf dein neues Modpack.

Sonst gibt es zu sagen: mach dir kein Stress prüfe alles lieber zwei mal und sei nicht als zu streng mit dir selber.

Bis dann vielleicht sieht man sich mal online .

Dein Kors92

ANTWORTEN ([HTTP://SEBINSIDE.DE/2017/05/08/EIN-BLICK-AUF-DIE-MODPACK-ENTWICKLUNG/?REPLYTOCOM=111#RESPOND](http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/?replytocom=111#respond))

KURT

31. Mai 2017 um 16:00 Uhr (<http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/#comment-112>)

Ziemlich informativ. Ich würde mich sehr freuen wenn du die Videoreihe umsetzen würdest (wo bleiben die Japan Vlogs ?).

Ich arbeite jetzt auch an einem eigenen Modpack, wenn ich fertig bin werde ich es dir mal zusenden vielleicht hast irgendwann mal zeit es zu testen.

ANTWORTEN ([HTTP://SEBINSIDE.DE/2017/05/08/EIN-BLICK-AUF-DIE-MODPACK-ENTWICKLUNG/?REPLYTOCOM=112#RESPOND](http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/?replytocom=112#respond))

KURTB

12. August 2017 um 16:40 Uhr (<http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/#comment-126>)

weiß jemand wie man bei einem modpack einstellen kann das bei „create new world“ eine z.B. Skyblock map entsteht ?

ANTWORTEN ([HTTP://SEBINSIDE.DE/2017/05/08/EIN-BLICK-AUF-DIE-MODPACK-ENTWICKLUNG/?REPLYTOCOM=126#RESPOND](http://sebinside.de/2017/05/08/ein-blick-auf-die-modpack-entwicklung/?replytocom=126#respond))

KOMMENTAR VERFASSEN

Gib hier deinen Kommentar ein ...

◀ 14 coole Scala-Features (Teil 3 von 3) (<http://sebinside.de/2016/10/05/14-coole-scala-features-teil-3-von-3/>)

Let's Play Automatisierung ▶ (<http://sebinside.de/2017/06/26/lets-play-automatisierung/>)

SOCIAL MEDIA

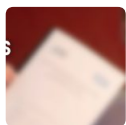


(htt (htt (htt (htt
p://t p://f p://y (htt
witt aceb outu p://g
er.c ook. be.c ithu
om/ com om/ b.co
skat /seb skat m/s
e70 insid e70 ebin
2) e) 2) side)

BLOG VIA E-MAIL ABONNIEREN

Gib Deine E-Mail-Adresse an, um diesen Blog zu abonnieren und Benachrichtigungen über neue Beiträge via E-Mail zu erhalten.

BELIEBTE BEITRÄGE



Rückblick auf den Bachelor (<http://sebinside.de/2017/04/13/rueckblick-auf-den-bachelor/>)

13 Apr , 2017



Programmierparadigmen
(<http://sebinside.de/2016/07/02/programmierparadigmen/>)

02 Jul , 2016



Webseiten kommen und gehen (<http://sebinside.de/2016/06/15/webseiten-kommen-und-gehen/>)

15 Jun , 2016

KATEGORIEN

📁 Allgemein (<http://sebinside.de/category/allgemein/>)

📁 Coding (<http://sebinside.de/category/coding/>)

📁 Minecraft (<http://sebinside.de/category/minecraft/>)

📁 Studium (<http://sebinside.de/category/studium/>)

Suche ...



NEUESTE KOMMENTARE

💬 Bowser bei Hallo. äh.. Welt! (<http://sebinside.de/2016/03/22/hallo-ae-h-welt/#comment-138>)

💬 daeaevaeaed bei Rückblick auf den Bachelor (<http://sebinside.de/2017/04/13/rueckblick-auf-den-bachelor/#comment-137>)

💬 coernerbrot bei Programmierparadigmen
(<http://sebinside.de/2016/07/02/programmierparadigmen/#comment-136>)

💬 seb (<http://sebinside.de>) bei Eine neue Plugin-Architektur für Code Overflow
(<http://sebinside.de/2018/01/16/eine-neue-plugin-architektur-fuer-code-overflow/#comment-135>)

💬 Sven (<https://usacookie.de>) bei Rückblick auf den Bachelor
(<http://sebinside.de/2017/04/13/rueckblick-auf-den-bachelor/#comment-134>)

ARCHIV

Januar 2018 (<http://sebinside.de/2018/01/>)

Juni 2017 (<http://sebinside.de/2017/06/>)

Mai 2017 (<http://sebinside.de/2017/05/>)

April 2017 (<http://sebinside.de/2017/04/>)

Oktober 2016 (<http://sebinside.de/2016/10/>)

September 2016 (<http://sebinside.de/2016/09/>)

August 2016 (<http://sebinside.de/2016/08/>)

Juli 2016 (<http://sebinside.de/2016/07/>)

Juni 2016 (<http://sebinside.de/2016/06/>)

April 2016 (<http://sebinside.de/2016/04/>)


März 2016 (<http://sebinside.de/2016/03/>)

SOCIAL MEDIA



(<http://twitter.com/sebinside>) (<http://facebook.com/sebinside>) (<http://youtube.com/sebinside>) (<http://skate702.de>)

FEED

 (<http://sebinside.de/feed/>) RSS - Beiträge (<http://sebinside.de/feed/>)

 (<http://sebinside.de/comments/feed/>) RSS - Kommentare (<http://sebinside.de/comments/feed/>)



Ich heiße Sebastian, studiere Informatik am KIT in Karlsruhe und betreibe den YouTube Kanal skate702 (<http://youtube.com/skate702>). Auf diesem Blog geht es um Technik, Software und Minecraft!