

# BLOCK

# 'N'

# LOCK



3

1

2

4

**A Cooperative 2D Puzzle  
Jump'n'Run**

# Inhalt

- *Installation*.....2
  
- *Spiel ausführen*.....3
  - *Server erstellen*.....3
  - *Server beitreten*.....4
  
- *Lobby*.....5
  - *Buttons*.....5
  - *Chat*.....6
  - *Spiel erstellen/beitreten*.....7
  
- *im Spiel*.....9
  - *Beschreibung*.....9
  - *Steuerung*.....10
  - *Mechaniken*.....11
  
- *für fortgeschrittene Nutzer*.....14
  - *Level erstellen*.....14
  - *Level integrieren*.....18

# Installation

Eine Installation des Spiels ist nicht erforderlich, da es sich um eine JAR handelt. Sie sollten lediglich sicherstellen, dass die Java Version 1.8 oder höher installiert ist. Das ist bei den meisten Rechnern bereits der Fall, da viele Anwendungen Java benötigen.

Ist Java jedoch nicht installiert oder nicht auf der beschriebenen Version, finden Sie diese unter:

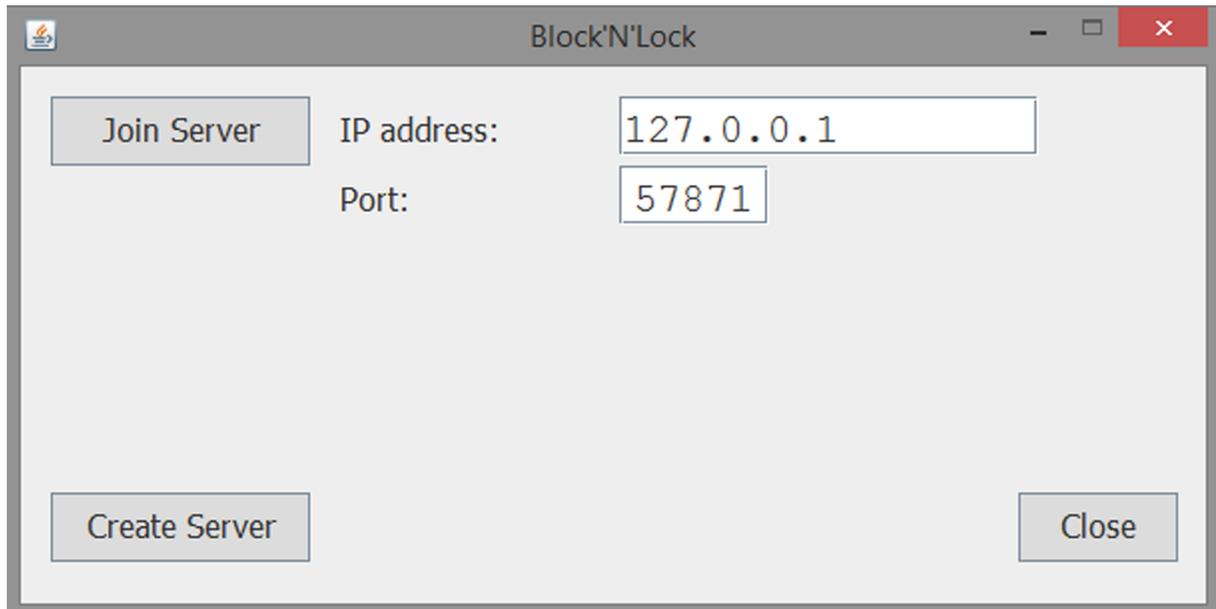
<https://www.java.com/de/download/>

Folgen Sie den Anweisungen der Installation. Ist diese abgeschlossen, können Sie das Spiel starten.

Für die Ausführung des Spiels reicht ein Doppelklick auf die JAR-Datei.

# Spiel ausführen

Beim Ausführen der JAR wird ein Fenster geöffnet.



## Server erstellen (Create Server)

Beim Erstellen eines Servers muss nur die Portnummer beachtet werden. Sie ist standartmässig auf "57871" gesetzt und braucht auch nicht verändert werden. Durch das Anklicken des **Create Server** Buttons wird der Server erstellt und man wird automatisch als Client verbunden. Man gelangt nun in die Lobby.

Alternativ kann die JAR auch mit dem folgenden Befehl in der Konsole (*Eingabeaufforderung/Terminal*) aufgerufen werden, um einen Server zu starten (*Der Pfad in der Konsole muss mit dem Pfad der JAR-Datei übereinstimmen*):

```
java -jar BLOCKnLOCK.jar server <portnummer>
```

# Server beitreten (Join Server)

Beim Beitritt eines Servers müssen zwei Dinge beachtet werden. Einerseits muss die IP-Adresse im Eingabefeld mit der des Servers übereinstimmen.

Diese kann der Host des Servers herausfinden, indem er in der Konsole den Befehl `ipconfig` (Windows), `ifconfig` (Mac) oder `ip addr show` (Linux Distributionen) eingibt.

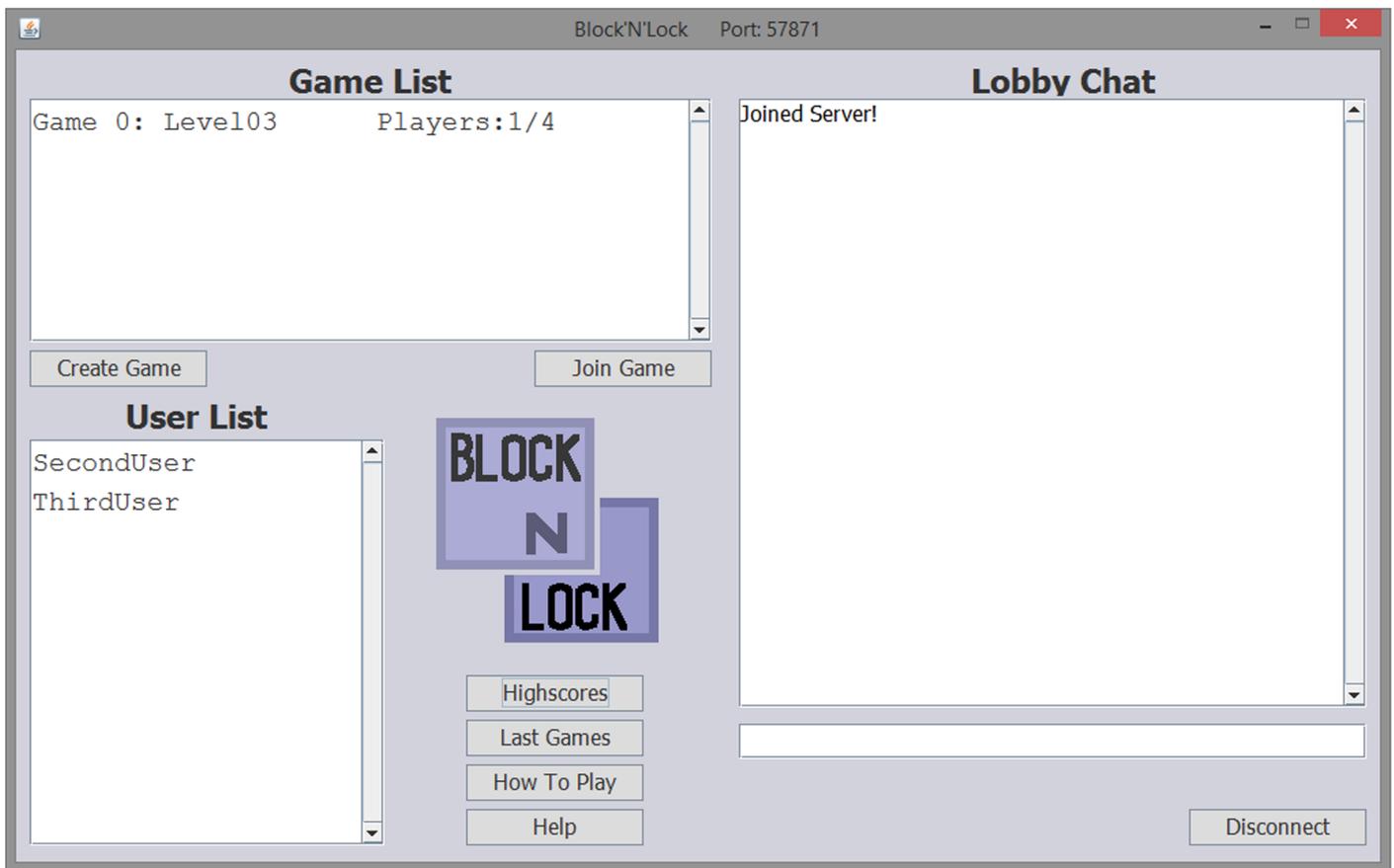
Alternativ kann er auch in der Netzwerkkumgebung die Verbindung oder den Typ des Netzwerks öffnen und die Details und Informationen prüfen.

Stimmen die eingegebene IP-Adresse und die Portnummer mit jenen des Servers überein, kann der **Join Server** Button angeklickt werden und der Spieler wird mit dem Host verbunden. Danach gelangt er in die Lobby.

Auch ein Client kann per Konsole gestartet werden. Dazu muss in der Konsole folgender Befehl eingegeben werden (*Information zur Konsole auf Seite 3, Server erstellen*):

```
java -jar BLOCKnLOCK.jar client <serverip>:<serverport>
```

# Lobby



## Buttons

Highscore

Dieser Button zeigt die besten Zeiten für abgeschlossene Level des aktuellen Nutzers (an den Namen gebunden).

Last Games

Dieser Button zeigt die letzten 10 abgeschlossenen Level.

How To Play

Dieser Button ruft ein Fenster auf, welches das Spiel grundlegend erklärt und die Steuerung zeigt.

Disconnect

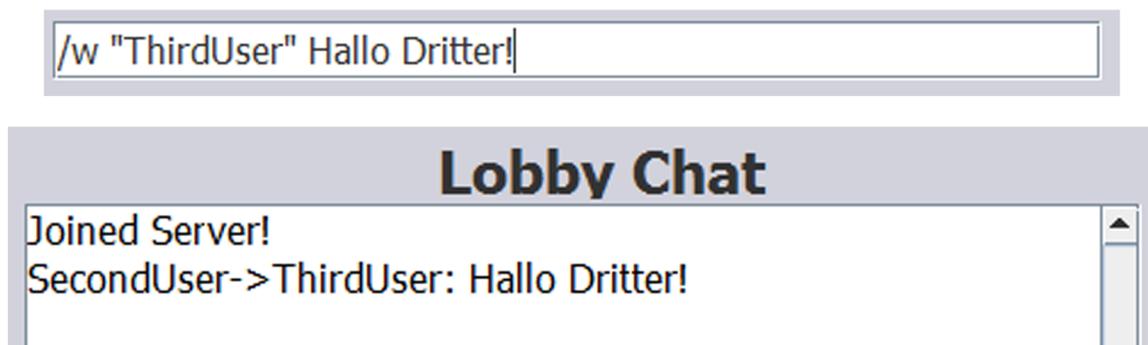
Dieser Button trennt die Verbindung zwischen Client und Host.

# Chat

Die rechte Seite des Lobby Menüs wird vom Chat eingenommen. Dieser steht allen Benutzern zur Verfügung, die auf der linken Seite in der **User List** stehen. Speziell für den Chat gibt es den Help Button, der die Chatbefehle erklärt. Alternativ kann im Eingabefeld */help* eingegeben werden.

## Chatbefehle

*/w "name"* Sendet einem Spieler eine Privatnachricht, wobei nur der Flüsternde und der Angeflüsterte jeweils die Nachricht sehen. Der Befehl kann durch ein Anklicken eines Spielers aus der Spielerliste aufgerufen werden.



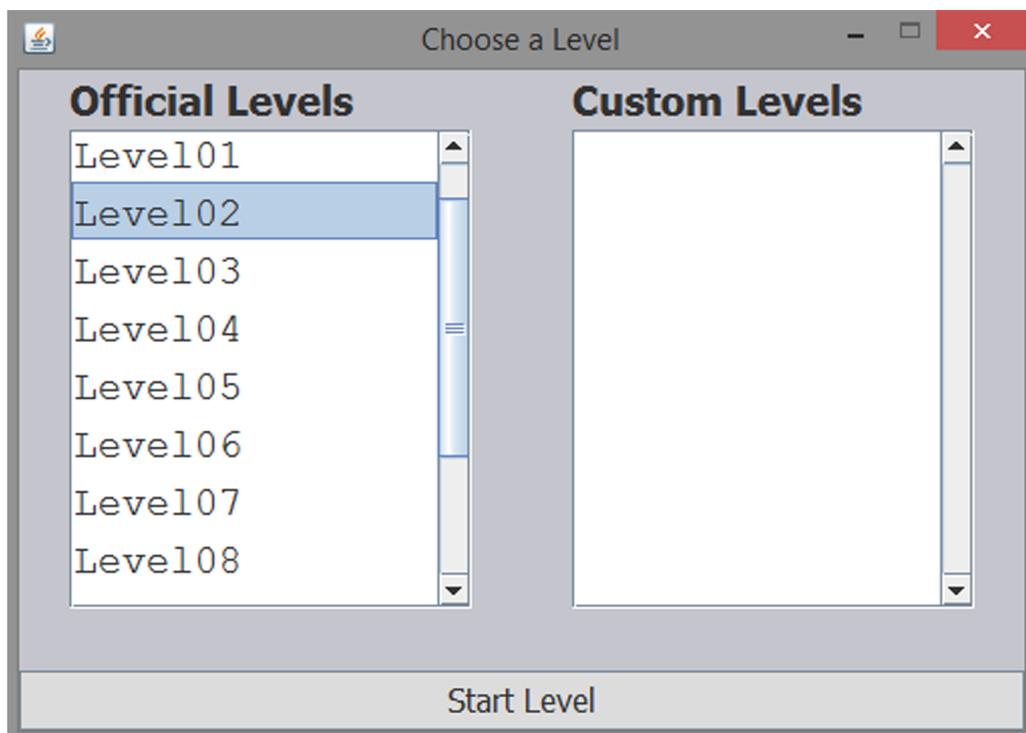
*/rename "name"* Benennt den Spieler zu dem gewählten Namen um.

# Spiel erstellen

Um ein Spiel zu erstellen genügt es, den **Create Game** Button anzuklicken.



Danach öffnet sich ein Fenster, worin offizielle Level und eigene Level aufgelistet werden. Die offiziellen Level sind die vorgegebenen Level des Entwicklerteams, die eigenen Level hingegen können von jedem Nutzer selbst erstellt und hinzugefügt werden. Doch dazu mehr im Abschnitt *für fortgeschrittene Nutzer*.



Wenn ein Level ausgewählt wurde, muss nur noch der **Start Level** Button angeklickt werden, dann wird der entsprechende Level gestartet.

# Spiel beitreten

In der Spielereihe können die aktuell laufenden Spiele auf dem Server eingesehen werden.



Die Spiele werden nummeriert in der Liste dargestellt. Danach steht der Name des gespielten Levels und die Anzahl Spieler, die sich gerade im Spiel befinden. Hat man eines ausgewählt, kann man den **Join Game** Button anklicken um in das Spiel zu gelangen.

Es können maximal vier Spieler einem Spiel beitreten.

***Bemerkung:*** Die Anzahl an Spielen, die ein Server hosten kann, ist nicht begrenzt. Allerdings sollten nicht zu viele Instanzen gleichzeitig gestartet werden, da das Netzwerk sonst zu stark ausgelastet wird. Das kann zum Stottern des Spiels oder zu Verbindungsproblemen bis hin zum Verbindungsverlust führen.

# Das Spiel

## Beschreibung

Bei diesem Spiel handelt sich um ein kooperatives 2D Jump`n`Run. Vier Spieler arbeiten zusammen, um ihre vier Spielfiguren durch den Level zu steuern und zum Ziel zu bringen. Die Spieler müssen Rätsel dazu lösen und Passagen durchqueren, um an die Zielorte zu gelangen. Die Kernmechaniken des Spiels sind das Konzept der Farbe und das Einfrieren der Spielerblöcke. Die Farben interagieren untereinander (siehe Mechaniken), während ein gefrorener Block solide wird und Kollision zulässt.

Das Spiel ist für vier Spieler ausgelegt, jedoch kann das Spiel auch allein, mit ein oder zwei Mitstreitern bewältigt werden. Empfohlen wird jedoch zu viert zu spielen.

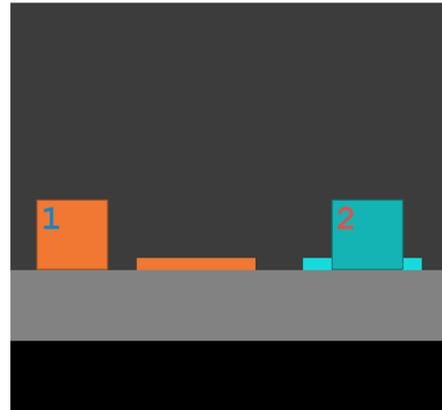
Wichtig ist, sich mit den Teammitgliedern zu koordinieren. Man muss die Elemente des Levels richtig deuten und durch Einfallsreichtum zu einer möglichen Lösung kommen. Natürlich ist es das Ziel, alle Blöcke so schnell wie möglich zu den Zielorten zu bringen, um die bestmögliche Zeit im Team zu erzielen.

# Steuerung

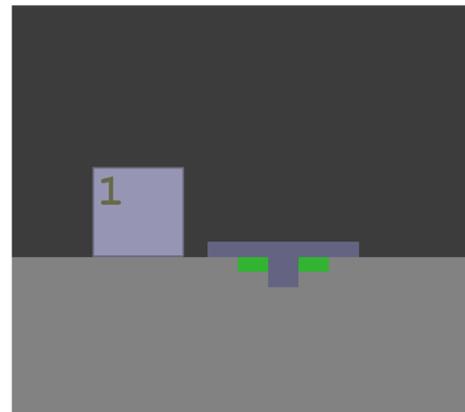
Taste	Beschreibung
A	nach links bewegen/wachsen (grün)
D	nach rechts bewegen/wachsen (grün)
W	springen/nach oben wachsen (rot/blau)
S	gedrückt halten, um den Block einzufrieren oder zu entfrieren
F	einmal drücken, um den Zustand des Blocks zu ändern (gefroren/ungefroren)
Q	zum nächsten, ungesteuerten Block wechseln
R	Neustart des Levels (alle Spieler müssen R drücken)
Maustaste	zeichnet einen Hinweis an der Position des Mauszeigers
1-4	zeichnet einen Hinweis mit der gedrückten Zahl, um Spieler direkt anzusprechen
K	hereinzoomen
L	herauszoomen

# Mechaniken

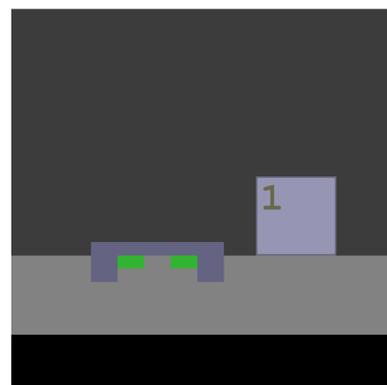
Der *Einfärber* ist eine Druckplatte, die den Spieler einfärbt, wenn er die Platte aktiviert. Aktivieren kann er diese, indem er sich darauf einfriert.



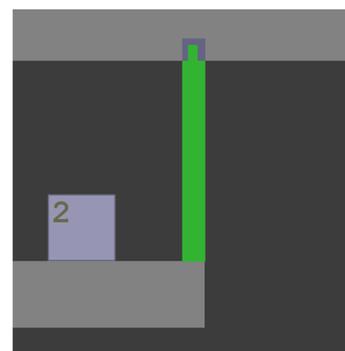
Eine *Druckplatte* ist ein interaktives Objekt, welches dazu dient, einen oder mehrere Laser ein- oder auszuschalten. Der Spieler muss dauerhaft darauf gefroren sein, damit sie aktiviert bleibt.



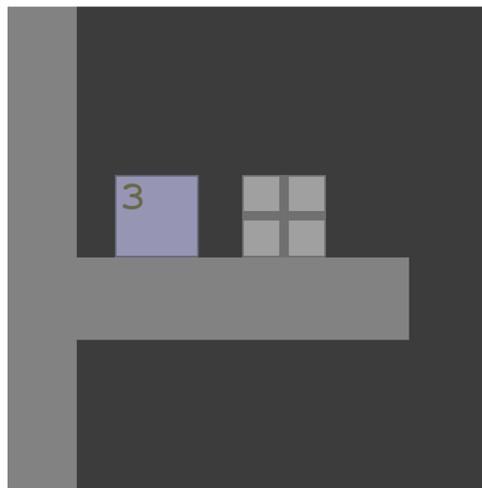
Ein *Schalter* funktioniert ähnlich wie eine Druckplatte, jedoch muss er nur einmal betätigt werden, damit der Effekt in Kraft tritt.



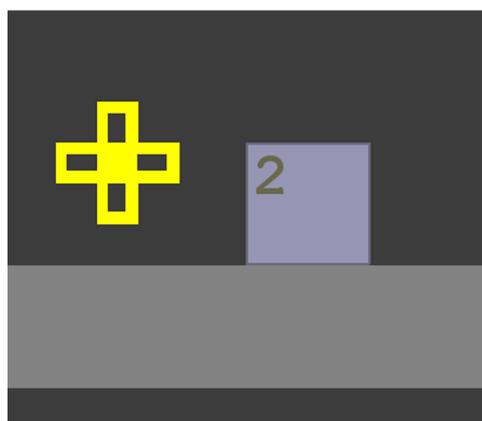
Ein *Laser* stellt eine Barriere dar. Er kann nicht passiert werden, es sei denn der Spieler hat die gleiche Farbe.



**Kisten** sind bewegliche Blöcke, die von Spielern geschoben werden können. Es ist möglich, sie einzufärben und mit ihnen Druckplatten und Schalter zu bedienen.

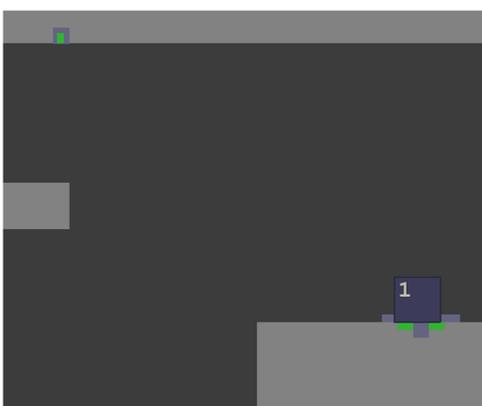
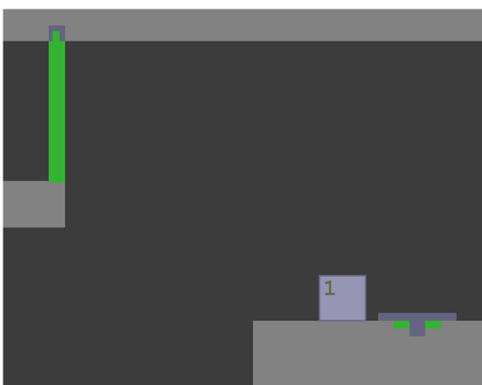


Die **Zielorte** markieren die Stellen, an denen sich jeweils ein Spieler befinden muss, um den Level abzuschliessen.

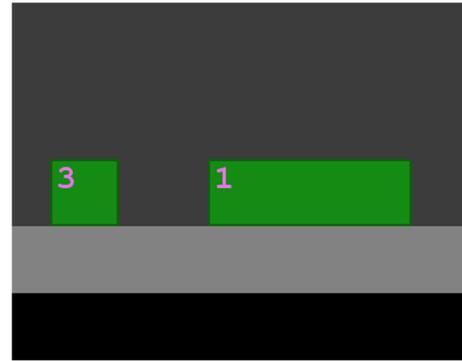


Hier ein Beispiel wie eine Druckplatte ausgelöst wird, indem der Spieler auf ihr einfriert.

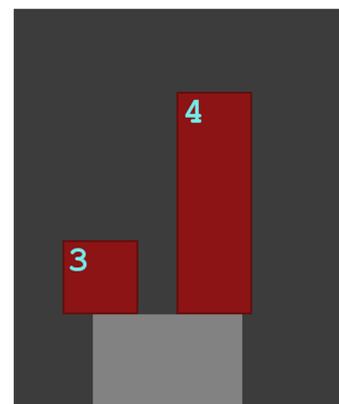
Die Platte hat grüne Farbelemente, was ein Hinweis darauf ist, dass etwas grünes damit in Verbindung steht. Im Bild ist klar erkenntlich, dass die Druckplatte den grünen Laser aktiviert und auch deaktiviert.



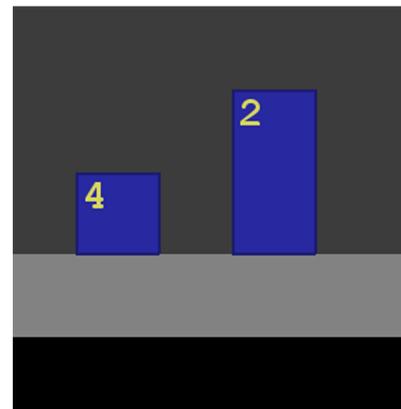
Ein grüner Block hat die Möglichkeit sich seitlich auszudehnen. Die Breite hat eine maximale Begrenzung.



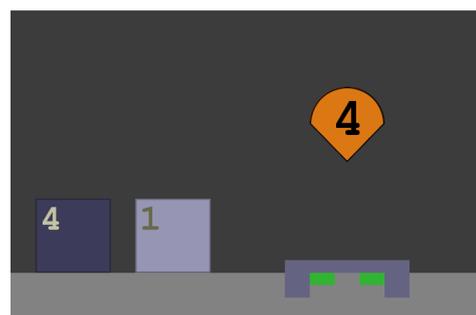
Ein roter Block hat die Möglichkeit sich nach oben auszudehnen. Die Höhe ist auf drei Blöcke begrenzt (maximale Sprunghöhe)



Ein blauer Block hat ebenfalls die Möglichkeit nach oben zu wachsen, jedoch liegt die maximale Begrenzung bei 2 Blöcken.



Ein Hinweis dient zur Kommunikation zwischen den Spielern. Sie haben eine orange Farbe.



# Für fortgeschrittene Nutzer

## Eigene Level erstellen

Jedes Block'n'Lock-Level ist eine abgespeicherte Textdatei, welche vom Spiel beim Start ausgelesen wird. Einerseits gibt es offizielle Level, die in der JAR-Datei gespeichert und von den Entwicklern vorgegeben sind. Andererseits werden die eigenen Level lokal im AppData-Ordner gespeichert. Das gibt dem Nutzer die Möglichkeit selbst Level zu entwerfen und sie zu spielen.

Die Struktur der Leveldateien hat eine spezifische Syntax.

Zuerst wird eine neue, leere Textdatei erstellt. Der Dateiname bestimmt den Levelnamen in der Levelauswahl. Dann wird die Datei beschrieben:

**Bemerkung:** Die eckigen Klammern markieren Abschnitte, während der Text in den runden Klammern optional ist. Der Text in Gänsefüßen dient als Hilfestellung, dieser muss geändert werden. Der Doppelslash am Anfang einer Zeile markiert diese als nicht relevanten Kommentar. Die vertikalen Striche sind Platzhalter für die Trennzeichen (delimiter). Wenn Farben angegeben werden, dann in Form einer Zahl. Eine Tabelle befindet sich am Ende des Kapitels.

Es werden im Einzelnen die Abschnitte erklärt, um dem Nutzer die Funktionalität der Syntax näher zu bringen.

Der Meta-Abschnitt kommt zu Beginn.

Der Autorenname zeigt, wer den Level erstellt hat. Das Trennzeichen kann frei gewählt werden, jedoch wird der Doppelpunkt empfohlen. Danach kann man dem Level einen Titel geben.

```
//That's a comment!  
  
<Meta  
(autor = "Name")(= "Unknown")  
(delimiter = "delimiter")(= :)  
(title = "Levelname")(= "")  
(gravity = "int")(= 0.01)  
width = "width"  
height = "height"  
>
```

Ausserdem hat man die Möglichkeit die Gravitation zu ändern, die Werte sollten jedoch nicht zu gross oder zu klein gewählt werden.

Wichtig ist es, die maximale Grösse des Levels anzugeben. Diese Werte definieren den Sandkasten.

Es folgt der Abschnitt für die Kisten.

```
<MovingCollisionCubes | "int"  
new|"xPosition"|"yPosition"(|"width"|"height"|"color"|"blocking")  
>
```

int entspricht der Anzahl an Kisten. Wenn keine Kisten benötigt werden, muss die Zahl 0 gewählt werden. Eine new-Zeile erzeugt eine Kiste an der Position (x,y). Es werden also so viele Zeilen wie Kisten benötigt. Es kann auch die Dimension der Kiste und ihre Farbe ändern sowie bestimmen, ob sie solide ist und Kollision

Dann muss der Hintergrund definiert werden.

```
<VisualBlocks
(defaultColor = "color")(= 1)
(mode = "fill/smartFill/void")(= smartFill)
(new|"xPosition"|"yPosition"|"width"|"height"(|"color"))
>
```

Die Farbe kann geändert werden. Standardmässig hat sie ein dunkles grau. Der Füllmodus wird im bestenfall auf smartFill belassen, man kann jedoch den Hintergrund weglassen (void) oder nur bestimmte Teile füllen (fill). Das wird mit einer new-Zeile gemacht.

Danach folgt der Abschnitt für die solide Levelstruktur.

```
<StaticBlocks
(defaultColor = "number")(= 2)
(border = "number")
new|"xPosition"|"yPosition"|"width"|"height"(|"color")
>
```

Die Farbe kann wieder geändert werden. Die Standardfarbe ist ein helles Grau. Die Grenze kann mit einer bestimmten Dicke (beispielsweise 3) gewählt werden. Sie erstellt einen Rand, der in den zuvor definierten Bereich des Levels gezeichnet wird. Die restlichen einzelnen Rechtecke werden mit einer new-Zeile definiert. Wichtig sind die Position und die Dimension dieser Flächen. Diese können auch unterschiedliche Farben haben.

## Nun der Abschnitt mit den Mechanikelementen.

```
<Actors : "number of Actors without the goalTargets"
newLaser|"xPosition"|"yPosition"|"color"|"SocketColor"|"activated"|"xDirection"|"yDirection"
newPressurePlate|"xPosition"|"yPosition"|"color"|"backgroundColor"|"priority"|"laser"
newSwitch|"xPosition"|"yPosition"|"color"|"backgroundColor"|"priority"|"laser"
newColoriser"H/Y"|"xPosition"|"yPosition"|"color"|"backgroundColor"|"colorAction"(|"elongation")
newColoriser|"xPosition"|"yPosition"|"color"|"backgroundColor"|"colorAction"|"colorActionType"
setTarget|"xPosition"|"xPosition"|"xPosition"|"xPosition"|"yPosition"|"yPosition"|"yPosition"|"yPosition"|"color"|"backgroundColor"
newGravityInverter|"xPosition"|"yPosition"|"color"|"backgroundColor"(|"isAtCeiling")
(A line which is given to the getPlayerCube without interpreting.)
1..,arg1,arg2,arg3,...
>
```

Auch hier muss die Anzahl der Actors anfangs definiert werden. Die Targets müssen jedoch nicht mitgezählt werden.

Die new-Zeilen werden hier für einzelne Elemente gebraucht. Diese lassen sich unter Mechaniken einsehen. Die meisten Argumente, die ein Actor hat, sind selbsterklärend. Ein Laser kann zu Beginn aktiviert oder deaktiviert sein und hat entweder eine horizontale (1 und 0) oder vertikale (0 und 1) Ausrichtung (vgl. Vektor). Druckplatten und Schalter haben ausserdem eine Priorität (über andere Platten und Schalter). Sie kann auch eine Referenz auf einen erstellten Laser haben (in der Reihenfolge nach Zahlen).

Danach folgt der Abschnitt für die solide Levelstruktur.

```
<Players
newPlayer|"xPosition"|"yPosition"(|"colorAction"|"color"|"freezeColor")
...
...
...
>
```

Hier werden vier new-Zeilen benötigt, um die Spielfiguren zu erzeugen. Man kann die Farben ändern, wenn man das möchte.

## Die Farben

0	Ultragrün	15	durchscheinendes, helles Grau
1	dunkles Grau	16	Schwarz
2	helles Grau	17	helles Blau
3	Gelb	18	dunkles Blau
4	helles Blaugrau	19	helles Violett
5	eher dunkles Rot	20	dunkles Violett
6	Stahl	21	helles Orange
7	dunkles Blaugrau	22	dunkles Orange
8	helles Grün	23	helles Türkis
9	dunkles Grün	24	dunkles Türkis
10	helles Rot	25	schmutziges Orange
11	dunkles Rot	26	Orange
12	sehr helles Grau	27	Weiss
13	sehr dunkles Graurot		
14	sehr dunkles Grau		

## Level integrieren

Um einen Level im Spiel spielen zu können, muss die Textdatei in den entsprechenden Ordner im BlockNLock-Ordner abgelegt werden. Der Ordner befindet sich unter Windows im AppData/Roaming-Ordner, unter Mac im Library-Ordner und unter Linux-Distributionen im Home Ordner.