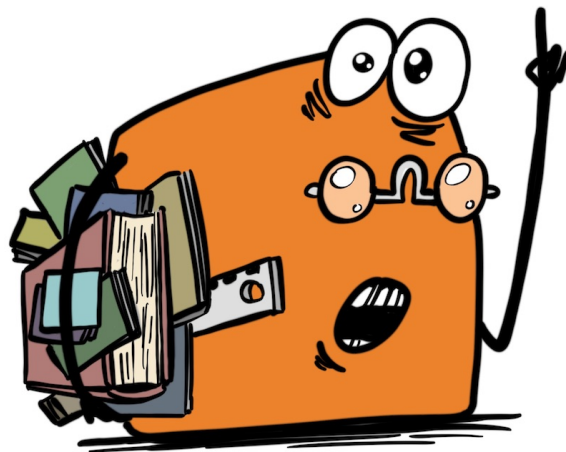


AgentCubes online

# AgentTalk Referenz



Michael Mittag / PH FHNW

# Inhalt

## Bedingungen

### **Grundlegende Bedingungen**

Sehe / See	4
Sehe eine/einen / See-a	5
Neben / next-to	5
Auf/unter / Stacked	6
Auf/unter einer/einem / Stacked-a	6
Leer / empty	7
Einmal-alle / Once-every	7
Ist ausgewählt / Is-selected	7
Prozentuale Wahrscheinlichkeit / Percent-chance	8
Aktuelle Welt / Current-world	8

### **Tastatur**

Taste	9
-------	---

### **Attribute**

Test	9
Hat Attribut / has-attribute	10

### **Kamerasteuerung**

Ich-Perspektive / First-person-view	11
-------------------------------------	----

## Methoden

while-running	12
on	12
when-creating-new-agent	12
before-running	12
on-mouse-hover	12
on-hand-tool	13

## Aktionen/Befehle / Actions

### **Grundlegende Aktionen / Basic Actions**

Bewege	14
Transportiere	15
Bewege dich zufällig	16
Bewege dich willkürlich auf	16
Wechsle	17
Neu	17
Lösche	17
Drehe dich um	18
Drehe dich bis	18
Teleportiere	19

### **Zeit / Time**

Warte	19
-------	----

### **Nachrichten / Messages**

Nachricht	20
Verzögerte Nachricht	20
Übertrage an alle	21
Nachricht im Stapel	21

### **Ton und Rede / Sound and speech**

Spiele Geräusch	21
Midi abspielen	21

Stoppe alle Geräusche	21
Sage	22
Drucke	22
<b>Attribute und Eigenschaften / Attributes and properties</b>	
Lege fest	22
Hinaufsteigen	22
<b>Visualisierung / Visualization</b>	
Lege Farbe fest auf	23
Lege die RGBA-Farbe fest	24
Bilde ab	24
Zeichne Plot in Fenster	25
Zeichne Plot des Agenten-Attributs	25
<b>Dialog / Dialog</b>	
Zeige Nachricht	26
<b>Steuerung der Simulation / Simulation Control</b>	
Führe Simulation durch	26
Mache eine schrittweise Simulation	26
Stoppe Simulation	26
Beende Animationen	26
<b>Steuerung der Welt / World control</b>	
Wechsle zur Welt	27
Starte die Welt neu	27
<b>Steuerung der Kamera / Camera Control</b>	
Wechsle zu Ich-Perspektive-Kamera	27
Wechsle zu Vogelperspektive-Kamera	28

# Bedingungen

Mit den Bedingungen kann ein Agent prüfen, was in der Welt um ihn herum vorgeht. AgentCubes geht bei der Ausführung von Programmen wie folgt vor:

- Die Regeln werden von oben nach unten geprüft.
- Eine Regel wird ausgeführt, wenn alle Bedingungen zutreffen.
- Es wird immer nur die erste Regel ausgeführt, deren Bedingungen zutreffen.

---

## Grundlegende Bedingungen

### Sehe / See



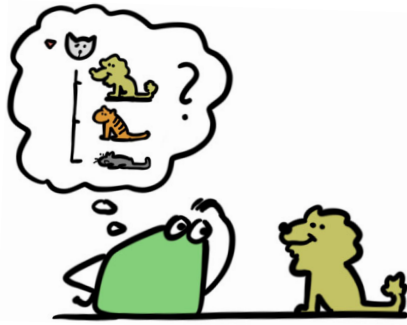
Prüft, ob der Agent einen Agenten mit einer bestimmten Form sieht.

- Richtung: Die Richtung, in welche der Agent schaut.
- Form: Die Form, deren Vorhandensein geprüft wird.
- Ebene: Die Ebene, relativ zur aktuellen Ebene. 0 (Standard) ist die aktuelle Ebene, -1 die Ebene darunter, 1 die Ebene darüber, und so weiter.

Bemerkung:

- Wenn der Agent das eigene Feld anschaut, dann bezieht sich der Befehl immer auf den Agenten selbst.
- Wenn der Agent ein anderes Feld als das eigene anschaut, dann bezieht sich der Befehl immer auf den obersten Agenten auf dem angesprochenen Feld.

## Sehe eine/einen / See-a



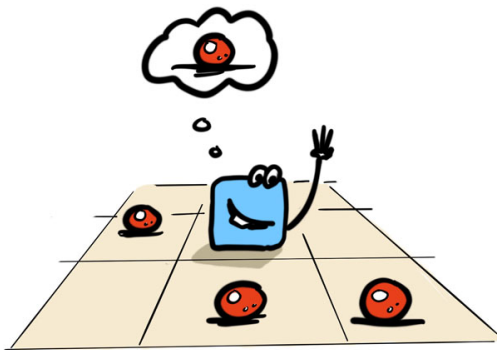
Prüft, ob der Agent einen Agenten mit einer beliebigen Form sieht.

- Richtung: Die Richtung, in welche der Agent schaut.
- Ziel: Der Agent, dessen Vorhandensein geprüft wird.
- Ebene: Die Ebene, relativ zur aktuellen Ebene. 0 (Standard) ist die aktuelle Ebene, -1 die Ebene darunter, 1 die Ebene darüber, und so weiter.

Bemerkung:

- Wenn der Agent das eigene Feld anschaut, dann bezieht sich der Befehl immer auf den Agenten selbst.
- Wenn der Agent ein anderes Feld als das eigene anschaut, dann bezieht sich der Befehl immer auf den obersten Agenten auf dem angesprochenen Feld.

## Neben / next-to



Prüft, wie viele Agenten mit einer bestimmten Form sich in der Umgebung des Agenten befinden. Geprüft werden alle 8 angrenzenden Felder.

Achtung: Gezählt werden jeweils nur die obersten Agenten auf den umliegenden Stapeln. Weiter unten liegende Agenten werden nicht gezählt.

- Vergleichsoperation: Die verwendete Vergleichsoperation, zum Beispiel > (grösser als), = (gleich), != (nicht gleich).
- Anzahl: Die Zahl, mit der verglichen wird.
- Form: Die Form, deren Vorhandensein geprüft wird.
- Ebene: Die Ebene, relativ zur aktuellen Ebene. 0 (Standard) ist die aktuelle Ebene, -1 die Ebene darunter, 1 die Ebene darüber, und so weiter.

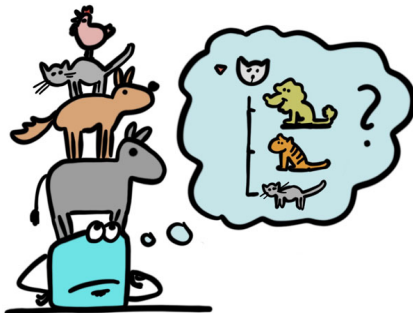
### Auf/unter / Stacked



Prüft, ob Agenten einer bestimmten Form sich im gleichen Stapel befinden wie der Agent.

- Platzierung: Wo im Stapel gesucht wird, zum Beispiel zu oberst, irgendwo oberhalb oder irgendwo im Stapel.
- Form: Die Form, deren Vorhandensein geprüft wird.

### Auf/unter einer/einem / Stacked-a



Prüft, ob Agenten einer beliebigen Form sich im gleichen Stapel befinden wie der Agent.

- Platzierung: Wo im Stapel gesucht wird, zum Beispiel zu oberst, irgendwo oberhalb oder irgendwo im Stapel.
- Form: Die Form, deren Vorhandensein geprüft wird.

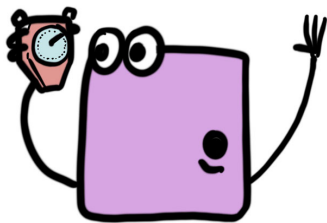
## Leer / empty



Prüft, ob der Stapel in der gewählten Richtung leer ist.

- Richtung: Die Richtung, in welche geprüft wird.
- Ebene: Die Ebene, relativ zur aktuellen Ebene. 0 (Standard) ist die aktuelle Ebene, -1 die Ebene darunter, 1 die Ebene darüber, und so weiter.

## Einmal-alle / Once-every

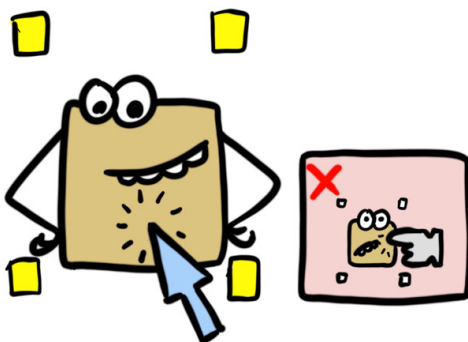


Wird einmal pro bestimmte Zeit ausgeführt.

- Zeit: Zeit in Sekunden. Oder: Slider() um die Geschwindigkeit des Systems zu übernehmen (Standardmässig ca. 1/4 Sekunde). Die Zeit kann auch als Formel angegeben werden.

Bemerkung: In AgentCubes wird die Zeit jeweils pausiert, während Bewegungen ausgeführt werden. Die Zeit bis zur tatsächlichen Auslösung kann sich dadurch verzögern. Das Kommando dient vor allem dazu, dass Befehle seltener ausgeführt werden als bei jedem Durchgang.

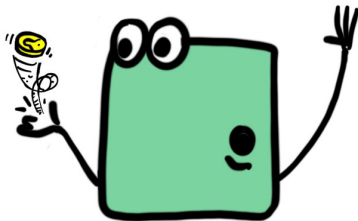
## Ist ausgewählt / Is-selected



Ist wahr, wenn der Agent mit dem Pfeilwerkzeug ausgewählt wurde. Der Agent ist in diesem Fall mit gelben Punkten markiert.

Bemerkung: Wenn der Agent mit dem Finger-Werkzeug angewählt wird, dann wird dies nicht gewertet (der Agent ist dann mit weissen Punkten markiert).

### Prozentuale Wahrscheinlichkeit / Percent-chance



Wird mit einer bestimmten Wahrscheinlichkeit als wahr gegeben. Diese Bedingung kann verwendet werden, damit ein bestimmtes Verhalten nur jedes zweite oder jedes zehnte Mal ausgeführt wird.

- Wahrscheinlichkeit: Prozentangabe oder Formel.

### Aktuelle Welt / Current-world



Prüft, ob die aktuelle Spielwelt eine bestimmte Welt ist. So kann Verhalten auf bestimmte Welten beschränkt werden.

- Welt: Eine der erstellten Welten.



---

## Tastatur

### Taste



Prüft, ob eine bestimmte Taste gedrückt ist.

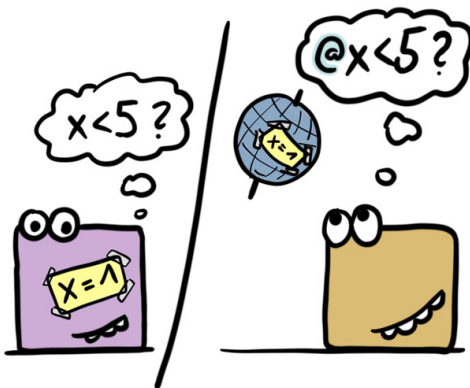
- Taste: Die Taste, die geprüft wird.

Bemerkung: AgentCubes verwendet jeweils die Tastenwiederholung des Systems. Wird die Taste gedrückt gehalten, dann ergibt sich dadurch eine leichte Verzögerung zwischen dem ersten Tastendruck und allen weiteren.

---

## Attribute

### Test

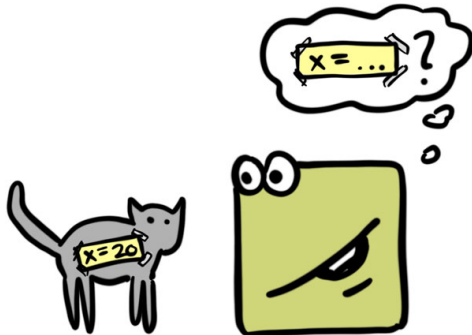


Prüft ein Attribut auf einen bestimmten Wert.

- Attribut: Attribut oder Formel. Gewicht[left] prüft beispielsweise das Attribut "Gewicht" des Agenten auf dem Feld links.
- Vergleichsoperation: Die verwendete Vergleichsoperation, zum Beispiel > (grösser als), = (gleich), != (nicht gleich).
- Vergleichswert: Zahl oder Formel.

Bemerkung: Durch Verwendung des Zeichens @ kann ich globale Variablen prüfen. @Timer bezieht sich beispielsweise auf die globale Variable "Timer" (falls diese existiert).

### Hat Attribut / has-attribute



Prüft, ob ein Agent ein Attribut besitzt.

- Richtung: Die Richtung, in welche geprüft wird.
- Attribut: Name des Attributs, das geprüft wird.
- Ebene: Die Ebene, relativ zur aktuellen Ebene. 0 (Standard) ist die aktuelle Ebene, -1 die Ebene darunter, 1 die Ebene darüber, und so weiter.

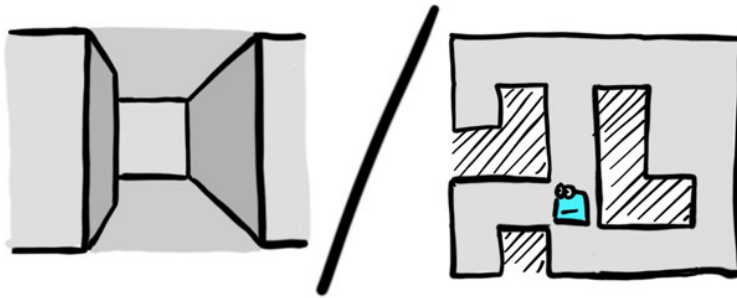
Bemerkung:

- Wenn der Agent das eigene Feld anschaut, dann bezieht sich der Befehl immer auf den Agenten selbst.
- Wenn der Agent ein anderes Feld als das eigene anschaut, dann bezieht sich der Befehl immer auf den obersten Agenten auf dem angesprochenen Feld.
- Die Attribute können in AgentCubes angesehen werden, indem im Menü Einstellungen (Zahnrad) auf "Zeige Attribute des Agenten" geklickt wird.

---

## Kamerasteuerung

### Ich-Perspektive / First-person-view



Prüft, ob AgentCubes derzeit in der Ich-Perspektive ist oder das Geschehen von oben zeigt.

# Methoden

Bedingungen werden in Methoden zusammengefasst. Die häufigste Methode ist "while-running". Während das Spiel läuft, wird sie fortlaufend aufgerufen. Eine weitere häufige Methode ist "on". Damit erstellte Methoden werden über Nachrichten gezielt aufgerufen. Damit kann das Programm gegliedert werden und Agenten können miteinander kommunizieren.

Wenn eine Methode ausgelöst wird, dann werden die darin enthaltenen Regeln von oben nach unten getestet, bis die erste Regel gefunden wird, bei der alle Bedingungen zutreffen. Danach werden keine weiteren Regeln ausgeführt.

## **while-running**

Wird regelmässig ausgelöst während das Spiel ausgeführt wird.

## **on**

Wird ausgelöst, wenn die Methode durch einen Befehl aufgerufen wird.

## **when-creating-new-agent**

Wird ausgelöst, wenn ein Agent neu erstellt wird auf dem Spielfeld.

## **before-running**

Wird ausgelöst, wenn das Spiel gestartet wird.

## **on-mouse-hover**

Wird ausgelöst, wenn der Mauszeiger über den Agenten fährt. Die Methode wird nur ausgelöst, wenn der Mauszeiger neu auf dem Agenten zu liegen kommt. Wenn der Mauszeiger sich danach längere Zeit über dem Agenten befindet, wird die Methode nicht erneut ausgelöst.

## **on-hand-tool**

Wird ausgelöst, wenn der Agent mit dem Hand-Werkzeug angeklickt wird. Im Spielmodus ist immer das Hand-Werkzeug aktiviert, dort löst also jeder Klick auf den Agenten diese Methode aus. Mit dem Parameter "Modifier" kann ausserdem festgelegt werden, dass nur ein Klick bei gedrückter Umschalt- oder Wahltaste die Methode auslöst.

# Aktionen/Befehle / Actions

Mit den Aktionen kann ein Agent in der Welt agieren. Einfache Aktionen sind beispielsweise Bewegung oder das Erstellen und löschen von Agenten. Etwas komplexer sind Nachrichten und Variablen. Mit ihnen lassen sich auch komplexere Spiele umsetzen. AgentCubes umfasst auch Aktionen zur Simulationssteuerung und Datenauswertung, diese sind dann sehr fortgeschritten.

---

## Grundlegende Aktionen / Basic Actions

### Bewege

Bewegt den Agenten um ein Feld in die angegebene Richtung.

- Richtung: Die Richtung, in die der Agent bewegt werden soll.
- Ebene: Die Ebene, relativ zur aktuellen Ebene. 0 (Standard) ist die aktuelle Ebene, -1 die Ebene darunter, 1 die Ebene darüber, und so weiter.
- Animation: "Konstant" für eine konstante Bewegung, "beschleunigt" für eine beschleunigte Bewegung. Die beschleunigte Bewegung ist am Anfang und am Ende abgebremst. Wenn die Höhe wechselt, dann sieht die beschleunigte Bewegung aus wie eine Sprungbewegung, die konstante Bewegung verläuft in gerader Linie.
- Zeit: Zeit für die Bewegung. Slider() (StandardEinstellung) verwendet die Spielgeschwindigkeit des Schiebereglers. Kleine Werte sorgen für schnelle Bewegungen. Achtung: Ein Wert von 0 wird als slider() interpretiert. Für eine schnelle Bewegung sollte zum Beispiel ein Wert von 0.01 verwendet werden.

Bemerkung: AgentCubes wartet immer, bis alle Bewegungen abgeschlossen sind, bevor weitere Befehle ausgeführt werden. Die Zeit kann deshalb nicht verwendet werden, um Agenten verschieden schnell zu bewegen. Wir raten deshalb, für alle

Agenten die gleichen Werte zu verwenden. Dazu gibt es folgende Möglichkeiten:

- Man verwendet überall den Wert "Slider()" (die Standardeinstellung). Die Spielerinnen und Spieler können dann die Spielgeschwindigkeit mit dem Schieberegler selbst einstellen.
- Man verwendet überall den gleichen numerischen Wert.
- Man verwendet eine globale Variable. Wenn man die Simulationseigenschaft "Geschwindigkeit" setzt, dann kann man als Zeit "@Geschwindigkeit" angeben, um überall die gleiche Geschwindigkeit zu verwenden. So kann die Spielgeschwindigkeit präzise eingestellt werden und auch im Lauf des Spiels angepasst werden.

## Transportiere

Bewegt den Agenten um ein Feld in die angegebene Richtung und nimmt alle darüberliegenden Agenten auf dem Stapel mit. Mit diesem Befehl kann zum Beispiel ein Förderband umgesetzt werden.

- Richtung: Die Richtung, in die der Agent bewegt werden soll.
- Ebene: Die Ebene, relativ zur aktuellen Ebene. 0 (Standard) ist die aktuelle Ebene, -1 die Ebene darunter, 1 die Ebene darüber, und so weiter.
- Animation: "Konstant" für eine konstante Bewegung, "beschleunigt" für eine beschleunigte Bewegung. Die beschleunigte Bewegung ist am Anfang und am Ende abgebremst. Wenn die Höhe wechselt, dann sieht die beschleunigte Bewegung aus wie eine Sprungbewegung, die konstante Bewegung verläuft in gerader Linie.
- Zeit: Zeit für die Bewegung. Slider() (Standardeinstellung) verwendet die Spielgeschwindigkeit des Schiebereglers. Kleine Werte sorgen für schnelle Bewegungen. Achtung: Ein Wert von 0 wird als slider() interpretiert. Für eine schnelle Bewegung sollte zum Beispiel ein Wert von 0.01 verwendet werden.

Bemerkung: AgentCubes wartet immer, bis alle Bewegungen abgeschlossen sind, bevor weitere Befehle ausgeführt werden. Die Zeit kann deshalb nicht verwendet werden, um Agenten verschieden schnell zu bewegen. Wir raten deshalb, für alle

Agenten die gleichen Werte zu verwenden. Dazu gibt es folgende Möglichkeiten:

- Man verwendet überall den Wert "Slider()" (die Standardeinstellung). Die Spielerinnen und Spieler können dann die Spielgeschwindigkeit mit dem Schieberegler selbst einstellen.
- Man verwendet überall den gleichen numerischen Wert.
- Man verwendet eine globale Variable. Wenn man die Simulationseigenschaft "Geschwindigkeit" setzt, dann kann man als Zeit "@Geschwindigkeit" angeben, um überall die gleiche Geschwindigkeit zu verwenden. So kann die Spielgeschwindigkeit präzise eingestellt werden und auch im Lauf des Spiels angepasst werden.

### **Bewege dich zufällig**

Bewegt den Agenten in eine zufällige Richtung. Der Agent kann um beliebig viele Felder und in alle Richtungen bewegt werden, auch auf höhere oder tiefere Ebenen.

- Zellen: Anzahl Zellen, um die der Agent bewegt wird.
- Zeit: Zeit für die Bewegung. Slider() (Standardeinstellung) verwendet die Spielgeschwindigkeit des Schiebereglers. Kleine Werte sorgen für schnelle Bewegungen. Achtung: Ein Wert von 0 wird als slider() interpretiert. Für eine schnelle Bewegung sollte zum Beispiel ein Wert von 0.01 verwendet werden.
- Auf der gleichen Ebene bleiben: Der Agent wechselt nicht die Ebene.
- Nur auf leeren Zellen: Der Agent wechselt nur auf Zellen, die leer sind.

### **Bewege dich willkürlich auf**

Bewegt den Agenten in eine zufällige Richtung, bleibt dabei aber auf einer bestimmten Form. So können Agenten sehr einfach auf einem Boden bewegt werden. Die Agenten bewegen sich nur auf leere Bodfelder.

- Animation: "Konstant" für eine konstante Bewegung, "beschleunigt" für eine beschleunigte Bewegung. Die beschleunigte Bewegung ist am Anfang und am Ende abgebremst.



- Zeit: Zeit für die Bewegung. Slider() (Standardeinstellung) verwendet die Spielgeschwindigkeit des Schiebereglers. Kleine Werte sorgen für schnelle Bewegungen. Achtung: Ein Wert von 0 wird als slider() interpretiert. Für eine schnelle Bewegung sollte zum Beispiel ein Wert von 0.01 verwendet werden.

## Wechsle

Wechselt die Form eines Agenten. Es können nur Formen des gleichen Agenten verwendet werden. Wird die Form eines anderen Agenten ausgewählt, dann wird der Befehl ignoriert.

- Richtung: Die Richtung, in welcher der Agent verändert wird. Bezieht sich der Befehl auf das aktuelle Feld, dann wird immer der aufrufende Agent verändert. Bezieht sich der Befehl auf ein anderes Feld, dann wird immer der oberste Agent auf dem Zielfeld geändert.
- Form: Die neue Form des Agenten.

Bemerkung:

- Wenn sich der Befehl auf das Feld bezieht, auf dem der Agent steht, dann bezieht sich der Befehl immer auf den Agenten selbst.
- Wenn sich der Befehl auf ein anderes Feld bezieht, dann bezieht sich der Befehl immer auf den obersten Agenten auf dem angesprochenen Feld.

## Neu

Erstellt einen neuen Agenten.

- Richtung: Die Richtung, in welcher der neue Agent erstellt wird.
- Form: Die neue Form des Agenten.
- Ebene: Die Ebene, relativ zur aktuellen Ebene. 0 (Standard) ist die aktuelle Ebene, -1 die Ebene darunter, 1 die Ebene darüber, und so weiter

## Lösche

Löscht einen Agenten.

- Richtung: Die Richtung, in welcher der Agent gelöscht wird. Bezieht sich der Befehl auf das aktuelle Feld, dann wird der aufrufende Agent gelöscht.

- Ebene: Die Ebene, relativ zur aktuellen Ebene. 0 (Standard) ist die aktuelle Ebene, -1 die Ebene darunter, 1 die Ebene darüber, und so weiter.
- Zeit: Bei einem Wert grösser als 0 wird der Agent mit einer Animation entfernt.

Bemerkung: Die Befehle nach dem Löschen-Befehl werden noch ausgeführt, auch wenn der Agent dann bereits vom Spielfeld entfernt ist.

### **Drehe dich um**

Dreht einen Agenten um einen bestimmten Winkel (in Grad).

- Kreisel: Der Agent dreht sich um die senkrechte Achse (wie ein Kreisel). Positive Werte drehen nach links, negative nach rechts.
- Kippen: Der Agent kippt nach vorne oder hinten. Positive Werte kippen nach vorne, negative nach hinten.
- Rollen: Der Agent rollt nach rechts oder links. Positive Werte rollen nach rechts, negative nach links.
- Zeit: Die Zeit zum Drehen. 0 ist eine sofortige, stufenlose Drehung. Für eine stufenlose Animation sollte die Drehung im gleichen Rhythmus erfolgen wie die anderen Animationen, da AgentCubes jeweils mit dem Ausführen von weiteren Befehlen wartet, bis alle Animationen ausgeführt sind.

### **Drehe dich bis**

Dreht einen Agenten auf einen bestimmten Winkel (in Grad). Eine Drehung auf 0/0/0 bringt den Agenten wieder in die ursprüngliche Position.

- Kreisel: Der Agent dreht sich um die senkrechte Achse (wie ein Kreisel). Positive Werte drehen nach links, negative nach rechts.
- Kippen: Der Agent kippt nach vorne oder hinten. Positive Werte kippen nach vorne, negative nach hinten.
- Rollen: Der Agent rollt nach rechts oder links. Positive Werte rollen nach rechts, negative nach links.
- Zeit: Die Zeit zum Drehen. 0 ist eine sofortige, stufenlose Drehung. Für eine stufenlose Animation sollte die Drehung im gleichen Rhythmus erfolgen wie die anderen Animationen, da AgentCubes jeweils mit dem Ausführen

von weiteren Befehlen wartet, bis alle Animationen ausgeführt sind.

## Teleportiere

Teleportiert den Agenten zu einem bestimmten Feld. Wird ein Feld angegeben, das nicht existiert, dann wird der Befehl nicht ausgeführt. Standardmässig wird eine Animation angezeigt, während sich der Agent vom einen Ort zum anderen bewegt.

- Zeile: Die Zeile, zu der sich der Agent bewegt. Ganz vorne ist Zeile 0, grössere Werte bewegen den Agenten weiter nach hinten.
- Spalte: Die Spalte, zu der sich der Agent bewegt. Ganz links ist Zeile 0, grössere Werte bewegen den Agenten weiter nach recht.
- Ebene: Die Ebene, zu der sich der Agent bewegt. Die unterste Ebene ist 0.
- Welt: Die Welt, zu der sich der Agent bewegt.
- Animation: Definiert, ob sich der Agent beschleunigt oder konstant bewegt.
- Zeit: Die Zeit für den Übergang vom einen Ort zum anderen. Ein Wert von 0 wird als "Slider()" interpretiert. Für einen sofortigen Übergang sollte ein kleiner Wert verwendet werden, zum Beispiel 0.01.

---

## Zeit / Time

### Warte

Stoppt die Ausführung von Programmanweisungen für eine bestimmte Zeit. In dieser Zeit werden allerdings noch die Animationen zu Ende ausgeführt. Der Befehl hat also nur eine Wirkung, wenn eine längere Zeitdauer angegeben wird, als die Animationen dauern.

- Zeit: Die Zeitdauer, für welche die Ausführung der Befehle gestoppt werden soll.

### Nachricht

Ruft eine Funktion im aktuellen oder in einem anderen Agenten auf. Der Agent muss über eine Funktion mit entsprechendem Namen verfügen. Funktionen werden definiert, indem ein Anweisungsblock mit "on" und dem Namen der Funktion geschaffen wird.

- Richtung: Die Richtung, in welche die Nachricht gesendet, also die Funktion aufgerufen wird.
- Funktion: Der Name der Funktion, die aufgerufen wird.
- Ebene: Die Ebene, relativ zur aktuellen Ebene. 0 (Standard) ist die aktuelle Ebene, -1 die Ebene darunter, 1 die Ebene darüber, und so weiter.

Bemerkung:

- Wenn sich der Befehl auf das Feld bezieht, auf dem der Agent steht, dann bezieht sich der Befehl immer auf den Agenten selbst.
- Wenn sich der Befehl auf ein anderes Feld bezieht, dann bezieht sich der Befehl immer auf den obersten Agenten auf dem angesprochenen Feld.

### Verzögerte Nachricht

Ruft eine Funktion im aktuellen Agenten auf nach Ablauf einer bestimmten Zeitdauer. Der Agent muss über eine Funktion mit entsprechendem Namen verfügen. Funktionen werden definiert, indem ein Anweisungsblock mit "on" und dem Namen der Funktion geschaffen wird.

Diese Funktion kann verwendet werden, wenn Animationen oder Übergänge sehr abrupt scheinen. Dies ist meist der Fall, wenn eine Spielrunde gewonnen oder verloren wird. In der Regel ist es zu abrupt, wenn die Spielstufe unmittelbar neu geladen wird. Es erscheint natürlicher, wenn das Spiel noch 1-2 Sekunden weiterläuft, damit die Spielerin oder der Spieler verarbeiten kann, was passiert ist.

- Funktion: Der Name der Funktion, die aufgerufen wird.

## **Übertrage an alle**

Ruft eine Funktion in allen Agenten eines bestimmten Typs auf. Der Agent muss über eine Funktion mit entsprechendem Namen verfügen. Funktionen werden definiert, indem ein Anweisungsblock mit "on" und dem Namen der Funktion geschaffen wird.

Diese Funktion kann verwendet werden, um Agenten synchron zu bewegen oder um bestimmte Dinge in allen Agenten auf dem Spielfeld auszulösen.

- Agent: Der Typ von Agent, in dem die Funktion aufgerufen wird.
- Funktion: Der Name der Funktion, die aufgerufen wird.

## **Nachricht im Stapel**

Ruft eine Funktion in einem oder mehreren Agenten im gleichen Stapel auf. Beispielsweise kann so eine Funktion in der Bodenplatte unter dem aktuellen Agenten ausgelöst werden.

- Funktion: Der Name der Funktion, die aufgerufen wird.
- Empfänger: Die Empfänger im Stapel: Alle anderen Agenten, oberster Agent, unterster Agent und so weiter.

---

## **Ton und Rede / Sound and speech**

### **Spiele Geräusch**

Spielt ein Geräusch ab.

- Geräusch: Das Geräusch, das abgespielt werden soll.

### **Midi abspielen**

Spielt eine Note auf einem MIDI-Gerät ab. MIDI ist ein Musik-Austauschformat. Hier wird es verwendet, um eine Note auf einem von vier digitalen Instrumenten zu definieren. Die Auswahl erfolgt über eine virtuelle Klaviertastatur.

### **Stoppe alle Geräusche**

Stoppt die Ausgabe aller Geräusche. MIDI-Noten werden nicht unterbrochen.

## Sage

Gibt einen Text als Sprache aus.

## Drucke

Schreibt einen Text als Mitteilung in die Statuszeile (unter dem Spielbildschirm).

Text: Der Text, der in der Statuszeile ausgegeben wird.

Bemerkung: Variablen können in den Text mit der so genannten Tilde eingefügt werden:

- ~wert verweist auf ein Attribut des Agenten
- ~@wert verweist auf eine Simulationseigenschaft

---

## Attribute und Eigenschaften / Attributes and properties

### Lege fest

Weist einer Variable einen Wert zu.

- Variable: Attribut oder Simulationseigenschaft.
- Wert: Zahl oder Formel.

Bemerkung: Variablen können lokal oder global sein. Lokale Variablen sind die Attribute von Agenten, globale Variablen sind die Simulationseigenschaften. Beide können über das Eigenschaftsmenü aufgerufen werden (Zahnrad), über die Befehle "Zeige Attribute des Agenten" und "Zeige Eigenschaften der Simulation".

### Hinaufsteigen

Der Agent bewegt sich auf den höchsten Wert eines bestimmten Attributs in den umliegenden Agenten zu.

Die Funktion setzt folgendes voraus:

- Die Welt verfügt über einen Boden.
- Die Agenten der Bodenplatten verfügen über ein Attribut, welches die Entfernung zu einem Ziel angibt.

Das Aufrufen dieses Befehls bewegt den Agenten um ein Feld auf das Ziel zu.

- **Attribut:** Der Name des Attributs, welches die Entfernung zum Ziel speichert. Kleinere Werte bedeuten eine grössere Entfernung.
- **Vier/acht Richtungen:** Gibt an, ob sich der Agent in vier oder acht Richtungen bewegt.
- **Animation:** Definiert, ob sich der Agent beschleunigt oder konstant bewegt.
- **Zeit:** Die Zeit für den Übergang vom einen Ort zum anderen. Ein Wert von 0 wird als "Slider()" interpretiert. Für einen sofortigen Übergang sollte ein kleiner Wert verwendet werden, zum Beispiel 0.01.

Bemerkung: Die Funktion eignet sich als schnelle Variante, um eine Zielsuche umzusetzen. Es kann sich allerdings lohnen, eine eigene Funktion zu schreiben, da diese Funktion einige potenzielle Probleme hat:

- Sie prüft immer die obersten umliegenden Agenten. Wenn sich mehrere Gegner nahe beieinander befinden, oder falls andere Dinge auf den Bodenplatten liegen, dann wird diese Funktion nicht perfekt funktionieren.
- Der Agent rückt immer auf den höchsten Wert der umliegenden Felder vor - auch wenn dieser Wert gleich oder tiefer ist als der Wert des aktuellen Feldes.
- Der Boden muss auf der gleichen Ebene sein wie die Spielfiguren. Oft ist es eleganter und einfacher, den Boden auf eine separate Ebene zu legen.
- Ebenfalls muss darauf geachtet werden, dass die Spielfigur selbst das Attribut besitzt und einen hohen Wert darin hat. Dadurch wird sichergestellt, dass die Gegner auch auf die Spielfigur selbst vorrücken.

---

## Visualisierung / Visualization

### **Lege Farbe fest auf**

Färbt den Agenten in der gewählten Farbe ein.

- **Farbe:** Die Farbe, in welcher der Agent eingefärbt wird.

Das Einfärben kann unter anderem wie folgt verwendet werden:

- Weiss zeigt den Agenten in der ursprünglichen Farbe.

- Grautöne dunkeln den Agenten ab. Schwarz zeigt den Agenten als schwarzen Umriss.
- Weisse, transparente Farben machen den Agenten stufenweise durchsichtig.

### **Lege die RGBA-Farbe fest**

Färbt den Agenten in der angegebenen Farbe ein. Die Farbwerte werden als Zahlen zwischen 0 und 1 angegeben. Es können auch Werte grösser als 1 angegeben werden, um den Agenten künstlich zu beleuchten. Mit dieser Funktion kann die Farbe durch Programmierung festgelegt werden.

- Rot: Der Rotwert, mit dem der Agent eingefärbt wird.
- Grün: Der Grünwert, mit dem der Agent eingefärbt wird.
- Blau: Der Blauwert, mit dem der Agent eingefärbt wird.
- Alpha: Die Transparenz des Agenten.

Das Einfärben kann unter anderem wie folgt verwendet werden:

- Weiss (1 1 1 1) zeigt den Agenten in der ursprünglichen Farbe.
- Grautöne (zum Beispiel 0.5 0.5 0.5 1) dunkeln den Agenten ab. Schwarz zeigt den Agenten als schwarzen Umriss.
- Weisse, transparente Farben (zum Beispiel 1 1 1 0.5) machen den Agenten stufenweise durchsichtig.
- Farbwerte grösser als 1 (zum Beispiel 3 3 3 1) lassen den Agenten leuchten.

### **Bilde ab**

Färbt den Agenten auf einer Farbskala ein, je nach Wert eines Attributs.

Attribut: Der Name des Attributs, das als Farbe abgebildet wird  
 Farben: Die Farben, in denen der Agent eingefärbt wird. Standardmässig wird der Agent bei tiefen Werten rot eingefärbt, bei hohen Werten in grün.  
 Werte: Die Skala, auf welcher die Werte abgebildet werden. Standardmässig wird ein Wert von 0 als tiefster Wert abgebildet (rot), ein Wert von 100 als höchster Wert (grün).

Bemerkung: Der Befehl bietet einen schönen und effektiven Weg, den Zustand von Agenten abzubilden, beispielsweise in einer Simulation. Allerdings muss er fortlaufend aufgerufen



werden, um die Farbe jeweils zu aktualisieren. Dazu gibt es folgende Möglichkeiten:

- Der Befehl wird an alle Wenn-Dann-Regeln angehängt, so dass er immer ausgeführt wird. Dies ist meist sehr unübersichtlich und sollte vermieden werden.
- Die bisherige "while-running"-Methode wird umbenannt zu einer anderen Methode, zum Beispiel "Verhalten". Eine neue "while-running"-Methode führt das "Bilde ab" Kommando und ruft dann die Methode "Verhalten" auf.
- Der "Bilde ab"-Befehl wird in einer separaten Methode platziert. Diese Methode wird dann von einem anderen Agenten über die Funktion "Übertrage an alle" in jedem Durchgang des Spiels aufgerufen.

### **Zeichne Plot in Fenster**

Zeichnet ein Liniendiagramm aus Werten des Agenten. Dieser Befehl wird oft für Simulationen verwendet. So kann zum Beispiel das Verhältnis von Raubtieren und Beute veranschaulicht werden.

- Wert: Das Attribut, das dargestellt werden soll. Es kann auch eine Formel verwendet werden. Die Formel kann zum Beispiel die Anzahl Agenten eines bestimmten Typs erfassen.
- Fenstername: Der Name des Fensters, in dem die Werte ausgegeben werden. Wenn mehrere Werte ausgegeben werden, dann werden alle im gleichen Fenster ausgegeben.
- Wertename: Der Name der angezeigten Linie im Diagramm.
- Farbe: Die Farbe, in welcher die Linie dargestellt wird.

### **Zeichne Plot des Agenten-Attributs**

Veranschaulicht die Werte eines Attributs als Hügellandschaft über der Spielwelt. Dies ist eine schnelle, effiziente, manchmal aber auch etwas unübersichtliche Methode, um den Zustand einer Simulation darzustellen. Eine etwas aufwendigere, aber oft bessere Möglichkeit ist der Befehl "Bilde ab".

- Wert: Der Wert, der abgebildet werden soll.
- Zeilen: Für welche Zeilen die Werte abgebildet werden.
- Spalten: Für welche Spalten die Werte abgebildet werden.

- Farbe: In welcher Farbe die Werte abgebildet werden. Es kann auch eine transparente Farbe verwendet werden.
- Höhe: Wie hoch über dem Spielfeld die Werte veranschaulicht werden sollen.
- Typ: Logarithmisch oder linear. Logarithmisch eignet sich, wenn einzelne Werte sehr gross werden. Die Werte werden dann einheitlicher.

---

## Dialog / Dialog

### **Zeige Nachricht**

Zeigt eine Nachricht als Dialogbox. Die Nachricht muss mit "OK" bestätigt werden.

Titel: Die erste Zeile der Box. Text: Der weitere Text.

Standardmässig wird hier ein Text angezeigt. Falls dieser nicht gewünscht ist, muss er entfernt werden.

---

## Steuerung der Simulation / Simulation Control

### **Führe Simulation durch**

Dieser Befehl entspricht dem Drücken des "Play"-Knopfes.

### **Mache eine schrittweise Simulation**

Dieser Befehl entspricht dem Drücken des "Schrittweise Ausführen"-Knopfes.

### **Stoppe Simulation**

Dieser Befehl entspricht dem Drücken des "Stopp"-Knopfes.

### **Beende Animationen**

Wartet, bis alle Animationen beendet sind. AgentCubes wartet ohnehin periodisch, bis die Animationen beendet sind. Üblicherweise ist es nicht notwendig, mit diesem Befehl darauf zu warten.

---

## Steuerung der Welt / World control

### Wechsle zur Welt

Wechselt zu einem anderen Level. Der Status des aktuellen Levels bleibt dabei erhalten.

Bemerkung: Es ist nicht ganz einfach, einen Level beim Verlassen wieder in den Ursprungszustand zurückzusetzen. Das kann aber notwendig sein, wenn ich ein Spiel mit Levels programmiere, die mehrmals gespielt werden können. Das Problem ist, dass ich zum Zurücksetzen den Befehl "Starte die Welt neu" brauche, und dass Befehle danach nicht mehr ausgeführt werden. Um die Welt beim Verlassen zurückzusetzen, kann man wie folgt vorgehen:

- Vor dem Verlassen setze ich eine globale Variable, zum Beispiel setze ich "@zurueck" auf den Wert "1".
- Dann starte ich die Welt neu mit dem Befehl "Starte die Welt neu". Dabei muss das Kästchen "Simulationseigenschaften neu laden" leer sein, damit der Wert nicht gleich wieder überschrieben wird.
- Der Spielfigur (oder einem anderen Agenten, der in jedem Level vorkommt) füge ich eine "before-running" Methode hinzu. Diese wird nun als erstes aufgerufen. In dieser Methode teste ich, ob "@zurueck" auf "1" gesetzt ist und wechsle dann zum anderen Level, wenn das so ist. Auf diese Weise lasse ich den Level frisch geladen zurück.

### Starte die Welt neu

Startet den aktuellen Level neu. Code, der nach diesem Befehl folgt, wird nicht mehr ausgeführt.

- Simulationseigenschaften neu laden: Die Eigenschaften der Simulation (also die globalen Variablen) werden zurückgesetzt.

---

## Steuerung der Kamera / Camera Control

### Wechsle zu Ich-Perspektive-Kamera

Wechselt in die Ich-Perspektive des betreffenden Agenten.

Bemerkung: Es kann sein, dass in der Ich-Perspektive Teile des Agenten sichtbar sind und die Sicht verdecken. In diesem Fall kann man beispielsweise einen kleineren Agenten zeichnen für die Ich-Perspektive, oder mit dem Befehl "Lege Farbe fest" den Agenten durchsichtig einfärben.

### **Wechsle zu Vogelperspektive-Kamera**

Wechselt in die Standardansicht.