Object Storage Visuals

Einbauanleitung für Modder

Diese Anleitung beschreibt die XML-Einträge für Lager, Paletten und befüllbare Kisten. Zu jedem Eintrag steht, in welche XML er gehört und was die einzelnen Werte bewirken.

Was möchtest du einbauen?	Kapitel / Seite
Ballenlager erweitern	2-5 / 3-6
Palettenlager erweitern	6-8 / 7-9
Paletten stapelbar machen	9 / 10
Befüllbare Kisten einlagern	10 / 11
Einbaufehler nachschlagen	11 / 12

Wo kommen die Einträge hin?

Am Lager: baleStorageVisuals und palletStacking kommen in den vorhandenen objectStorage-Block der Lager-XML.
An der Palette oder Kiste: objectStorageStacking und objectStorageReusablePallet kommen in die jeweilige Objekt-XML, direkt unter vehicle.

> Die Spalte **Vorgabe** zeigt den Wert, den OSV verwendet, wenn du den Parameter weglässt. Bei Schaltern bedeutet true an und false aus. Maße werden in Metern mit Dezimalpunkt angegeben.

Die XML-Beispiele zeigen nur die Einträge für OSV. Vorhandene Einträge deiner Mod bleiben bestehen. OSV muss im Spiel aktiviert sein.

1 Das Lager vorbereiten

Ausgangspunkt ist ein funktionierendes GIANTS-objectStorage. OSV verwendet dessen storageAreas, um die Ballen oder Paletten im Lager zu platzieren. Die Start- und Endnodes dieser Flächen müssen zu den tatsächlichen Abmessungen des Lagers passen.

Lager	supportsBales	supportsPallets	OSV-Eintrag
Nur Ballen	true	false	baleStorageVisuals
Nur Paletten	false	true	palletStacking
Ballen und Paletten	true	true	keiner dieser Blöcke

Die folgenden Lagererweiterungen sind für reine Ballen- oder reine Palettenlager gedacht. Ein gemischtes Lager bleibt bei der normalen GIANTS-Darstellung.

Beispiel: Ballenlager

```
<objectStorage supportsBales="true"
    supportsPallets="false" capacity="110">
    <!-- Vorhandene Eintraege des Lagers bleiben. -->
    <baleStorageVisuals enabled="true" />
</objectStorage>
```

Beispiel: Palettenlager

```
<objectStorage supportsBales="false"
    supportsPallets="true" capacity="250">
    <!-- Vorhandene Eintraege des Lagers bleiben. -->
    <palletStacking dynamicCapacity="true"
        hardLimit="500" />
</objectStorage>
```

Ergänze den passenden OSV-Block in deinem vorhandenen objectStorage. Lege dafür keinen zweiten objectStorage-Block an.

2 Ballenlager: zusätzliche Paletten

Einbauort: Lager-XML > objectStorage

```
<baleStorageVisuals enabled="true"
    dynamicCapacity="true" hardLimit="500">
    <!-- Unterlegpaletten oder Palettenstapel: Kapitel 5 -->
</baleStorageVisuals>
```

Parameter	Vorgabe	Was bewirkt er?
enabled	false	true gibt die Darstellung der konfigurierten Unterlegpaletten und Paletten-Vorratsstapel frei. false zeigt diese zusätzlichen Paletten nicht.
dynamicCapacity	false	true begrenzt die Annahme nach dem tatsächlich verfügbaren Platz in den storageAreas. false verwendet die normale Stückzahlgrenze des Lagers.
hardLimit	500	Höchstens so viele Objekte annehmen, wenn dynamicCapacity aktiv ist. Mehr sichtbarer Platz erhöht dieses Limit nicht.

enabled schaltet nicht die Ballen selbst sichtbar. Die Ballen werden bereits vom GIANTS-objectStorage dargestellt. Der Schalter gehört zu den zusätzlichen Paletten unter den Ballen und zu den daneben aufgestapelten Vorratspaletten. Die Einträge dafür stehen in Kapitel 5.

Die dynamische Kapazität und der bewegliche Einlagertrigger werden jeweils über ihre eigenen Schalter eingeschaltet. Sie benötigen nicht zusätzlich enabled="true" für die Palettendarstellung.

capacity und hardLimit

capacity ist die normale Stückzahlgrenze. Bei dynamicCapacity="true" entscheidet stattdessen, ob das nächste Objekt noch in die sichtbare Lagerfläche passt. hardLimit ist dabei die zusätzliche Obergrenze.

3 Ballenflächen nach Gruppen aufteilen

Einbauort: Lager-XML > objectStorage > storageAreas

Mit baleGroups kannst du einzelne Lagerflächen für Heu, Stroh oder gewickelte Ballen reservieren. Der Eintrag steht an der jeweiligen storageArea.

```
<storageAreas>
    <storageArea startNode="hayStart" endNode="hayEnd"
        maxHeight="4.6" baleGroups="hay" />
    <storageArea startNode="strawStart" endNode="strawEnd"
        maxHeight="4.6" baleGroups="straw" />
    <storageArea startNode="wrapStart" endNode="wrapEnd"
        maxHeight="4.6" baleGroups="wrapped" />
    <storageArea startNode="restStart" endNode="restEnd"
        maxHeight="4.6" />
</storageAreas>
```

Eintrag	Welche Ballen kommen dort hin?
hay	Heu und die von OSV als Trockenfutter zugeordneten Ballen.
straw	Strohballen.
wrapped	Gewickelte Ballen, unabhängig von ihrem Inhalt.
ohne baleGroups	Nicht zugeordnete Ballen und Ausweichfläche für volle Gruppenbereiche.

Mehrere Gruppen werden mit Leerzeichen getrennt, zum Beispiel baleGroups="hay straw". Ungewickeltes Gras gehört keiner dieser drei Gruppen an und braucht eine Fläche ohne baleGroups.

Bei dynamischer Kapazität wird nur auf passenden Flächen eingelagert. Ist die Heufläche voll, darf OSV auf eine Fläche ohne Gruppe ausweichen, aber nicht auf eine reine Strohfläche. Ohne baleGroups ist keine solche Aufteilung aktiv.

4 Einlagertrigger für Ballenlager

Einbauort: innerhalb von baleStorageVisuals

Der SlotTrigger ist der Annahmebereich, in dem ein Ballen eingelagert wird. Er wandert zum nächsten freien Lagerplatz. Der PlayerTrigger ist der Bereich, in dem der Spieler das Lagermenü öffnen kann.

```
<slotTrigger enabled="true" movePlayerTrigger="true"
    playerTriggerOffset="0 0 0" />
```

Parameter	Vorgabe	Was bewirkt er?
enabled	false	true ersetzt den festen objectTrigger durch den beweglichen Annahmebereich.
movePlayerTrigger	false	true bewegt auch den Bedienbereich zum aktuellen Annahmepunkt. false lässt ihn an seinem bisherigen Platz.
playerTriggerOffset	0 0 0	Versatz des mitbewegten Bedienbereichs: X, Y, Z in Metern, relativ zum SlotTrigger. Bei 0 0 2 liegt er 2 m in dessen lokaler Z-Richtung.

WEEKEND

Getrennte Annahmepunkte für Ballengruppen

```
<slotTriggers>
    <slotTrigger baleGroups="hay" />
    <slotTrigger baleGroups="straw" />
    <slotTrigger baleGroups="wrapped" />
    <slotTrigger />
</slotTriggers>
```

Diese Einträge verwenden die passenden Flächen aus Kapitel 3. Jeder Eintrag ist ohne weitere Angabe aktiv; enabled="false" schaltet ihn aus. Der Eintrag ohne Gruppe ist der Ausweichtrigger. Bei dieser Variante bleibt ein gemeinsamer PlayerTrigger für das Lagermenü bestehen.

Der alte objectTrigger wird ausgeblendet. Marker und Warnstreifen, die sichtbar bleiben sollen, deshalb nicht unter diesem Node anlegen.

5 Unterlegpaletten und Palettenstapel

Einbauort: innerhalb von baleStorageVisuals

baleStorageVisuals enabled="true" ist der gemeinsame Hauptschalter für Unterlegpaletten und Vorratspalettenstapel. palletUnderlays enabled="true" schaltet darunter die Paletten unter den Ballen ein. Für Unterlegpaletten müssen deshalb beide Schalter auf true stehen. Soll nur ein Vorratspalettenstapel neben dem Lager angezeigt werden, bleibt der Hauptschalter an und palletUnderlays auf false.

```
<palletUnderlays enabled="true"
    positionVariation="0.025" rotationVariation="1.5"
    rotationY="0" />
```

Parameter	Vorgabe	Was bewirkt er?
enabled	false	true stellt Paletten unter den belegten Ballenplätzen dar.
positionVariation	0.025	Leichter zufälliger Versatz der Paletten in Metern.
rotationVariation	1.5	Leichte zufällige Verdrehung in Grad.
rotationY	0	Feste Drehung der Paletten um die Y-Achse in Grad.

Bei aktiver dynamicCapacity prüft OSV auch, ob für den nächsten Ballen noch eine passende Unterlegpalette vorhanden ist.

Vorratspaletten neben dem Lager

```
<palletStacks>
    <palletStack linkNode="stackRef01"
        translation="0 0 0" rotation="0 0 0"
        maxPallets="20" collision="true" />
</palletStacks>
```

linkNode verweist auf den Node, an dem der Stapel stehen soll. translation verschiebt ihn relativ dazu in Metern, rotation dreht ihn in Grad; Vorgabe jeweils 0 0 0. maxPallets begrenzt den Stapel auf die angegebene Stückzahl, ohne Angabe wird die Zahl automatisch ermittelt. collision="true" schaltet seine Kollision ein; Vorgabe ist false.

6 Palettenlager und Einlagertrigger

Einbauort: Lager-XML > objectStorage

```
<palletStacking dynamicCapacity="true" hardLimit="500">
    <slotTrigger enabled="true"
        movePlayerTrigger="true"
        playerTriggerOffset="0 0 0" />
</palletStacking>
```

Am palletStacking	Vorgabe	Was bewirkt es?
enabled	true	true zeigt geeignete Paletten übereinander. false schaltet nur diese Stapeldarstellung aus.
dynamicCapacity	false	true prüft die belegbare Lagerfläche statt nur die Stückzahl.
hardLimit	500	Maximale Stückzahl bei dynamischer Kapazität.
stackInMultiLevelStorage	false	true erlaubt Stapel innerhalb von Regalfächern; siehe Kapitel 8.

Die drei SlotTrigger-Parameter

Am slotTrigger	Vorgabe	Was bewirkt es?
enabled	false	true bewegt den Annahmebereich zum nächsten freien Palettenplatz. Ohne ihn bleibt der normale objectTrigger aktiv.
movePlayerTrigger	false	true nimmt den Bedienbereich für das Lagermenü mit. false lässt den Bedienbereich am ursprünglichen Platz.
playerTriggerOffset	0 0 0	X/Y/Z-Versatz des mitbewegten Bedienbereichs relativ zum Annahmebereich, in Metern. Beispiel 0 0 2: 2 m entlang dessen lokaler Z-Achse.

Der Annahmepunkt steht beim leeren Lager am ersten Platz, danach am nächsten freien Platz und bei vollem Lager am letzten belegten Platz. SlotTrigger und dynamische Kapazität können auch mit palletStacking enabled="false" verwendet werden.

7 Paletten drehen

Einbauort: am palletStacking-Eintrag des Lagers

Mit diesen Schaltern kann OSV Paletten bei der Lagerdarstellung um 90 Grad drehen. Das ist zum Beispiel sinnvoll, wenn eine Palette quer besser in ein Regalfach passt.

```
<palletStacking rotatePallets="true"
    rotateBigBagPallets="false" />
```

Parameter	Vorgabe	Welche Paletten?
rotatePallets	false	Alle von OSV erkannten Palettengruppen.
rotateEuroPallets	false	Euro-Paletten.
rotateBigBagPallets	false	BigBags auf Paletten.
rotateSquarePallets	false	Quadratische Paletten, z. B. IBCs.
rotateLargePallets	false	Große oder lange Paletten.
rotateBigBags	false	Einzelne BigBags ohne Palette.

Im Beispiel dürfen alle erkannten Gruppen gedreht werden, außer BigBags auf Paletten. Ohne rotatePallets="true" werden nur Gruppen gedreht, deren eigener Schalter auf true steht.

Die gedrehte Palette muss mit ihren tatsächlichen Maßen ins Fach passen. Die Ausrichtung der Lagerflächen selbst wird weiterhin über ihre Start- und Endnodes in der I3D festgelegt.

WEEKEND FARMERS

8 Palettenregale

Bei einem Regal bekommt jedes Fach oder jede Ebene eine eigene storageArea. Die Fläche darf nicht durch Pfosten, Wände oder Regalböden verlaufen. maxHeight wird auf die nutzbare Höhe des jeweiligen Fachs gesetzt.

```
<storageAreas>
    <storageArea startNode="shelf01Start"
        endNode="shelf01End" maxHeight="1.65" />
    <storageArea startNode="shelf02Start"
        endNode="shelf02End" maxHeight="1.65" />
</storageAreas>
<palletStacking dynamicCapacity="true" hardLimit="60"
    stackInMultiLevelStorage="true" />
```

Im Fach übereinander stapeln

OSV erkennt mehrere Lagerebenen, wenn die Startnodes mehr als 0.35 m Höhenunterschied haben. Zusätzliches Stapeln ist dort zunächst ausgeschaltet. Mit stackInMultiLevelStorage="true" dürfen geeignete Paletten auch innerhalb eines Fachs übereinander stehen.

Beispiel: Ein 1.65 m hohes Fach kann zwei 0.80 m hohe Paletten aufnehmen, sofern deren Stapelabstand ebenfalls 0.80 m beträgt. Passt nur eine Lage unter den Regalboden, darf maxHeight nicht größer angegeben werden, nur um weitere Paletten einzulagern.

Bestimmte Paletten ausschließen

maxWidth, maxLength und maxHeight am objectStorage prüfen die normale Objektgröße. Damit kannst du zu große Paletten ausschließen. Diese Prüfung erfolgt vor der sichtbaren Rotation. Kleinere stackWidth- oder stackLength-Werte ändern daran nichts.

Freier Restplatz eines Regalfachs wird weiter genutzt, wenn eine andere Palette noch hineinpasst. Ein Fach wird nicht automatisch für eine Sorte reserviert.

9 Eigene Paletten stapelbar machen

Einbauort: Paletten-XML, direkt unter vehicle

Mit objectStorageStacking legst du fest, wie deine Palette im Lager übereinander gestellt wird. Füge den Eintrag vor der abschließenden Zeile </vehicle> ein.

```
<objectStorageStacking enabled="true"
    stackWidth="1.24" stackLength="0.84"
    stackHeight="0.82" maxStackLayers="2"
    stackFullOnly="true" fullFillLevel="1000" />
```

Parameter	Vorgabe	Was bedeutet der Wert?
enabled	false	true erlaubt OSV, diese Palette sichtbar übereinanderzustellen.
stackWidth / stackLength	normale Palettenmaße	Breite und Länge des Stellplatzes einer Palette in Metern.
stackHeight	normale Palettenhöhe	Um wie viele Meter jede weitere Palette höher gesetzt wird. 0.82: zweite Palette 0.82 m über der ersten, dritte 1.64 m.
maxStackLayers	kein eigenes Limit	Maximale Anzahl übereinanderstehender Paletten. Die Lagerhöhe begrenzt zusätzlich.
stackFullOnly	false	true stapelt nur volle Paletten. false erlaubt auch leere oder teilweise gefüllte.
fullFillLevel	automatisch	Füllmenge, die für diese Palette als voll gilt. Ohne Angabe aus Startfüllung bzw. Kapazität.
palletGroup	automatisch	Ordnet die Palette einer Gruppe für die Drehschalter aus Kapitel 7 zu.

Für palletGroup sind euroPallet, bigBagPallet, squarePallet, largePallet und bigBag möglich. Die Gruppe allein schaltet das Stapeln nicht ein.

Greifen Füße oder Stapelhilfen ineinander, kann stackHeight kleiner als die volle Kistenhöhe sein. Verwende dann den Abstand, bei dem zwei aufeinandergesetzte Kisten tatsächlich aufliegen.

10 Befüllbare Kisten einlagern

Eine Kartoffelkiste oder Gitterbox wird nicht immer vom GIANTS-objectStorage angenommen, auch wenn sie sich im Spiel ganz normal befüllen lässt. Mit diesem Eintrag kann OSV solche Kisten in einem Palettenlager einlagern.

In die XML der Kiste eintragen

Suche die abschließende Zeile **</vehicle>** und füge den Eintrag unmittelbar davor ein:

```
<objectStorageReusablePallet enabled="true" />
```

Die übrigen Einträge deiner Kiste bleiben unverändert. In einem vorhandenen Palettenlager ist kein zusätzlicher Eintrag für diese Kiste nötig.

Parameter	Vorgabe	Was bewirkt er?
enabled	false	true erlaubt die Einlagerung dieser Kiste über OSV. false schließt die Kiste ausdrücklich aus.
fillUnitIndex	1	Nummer der fillUnit in der Kisten-XML. Bei einer Kiste mit nur einer fillUnit ist das 1. Diesen Parameter kannst du dann weglassen.

Welche Kisten sind dafür geeignet?

Die Kiste braucht **genau eine fillUnit für ihren Inhalt**. Der <fillUnit>-Eintrag legt zum Beispiel die Kapazität und die erlaubten Früchte fest. fillUnitIndex ist die Nummer dieses Eintrags, gezählt ab 1 - keine Literzahl und keine Frucht-ID.

Der Eintrag ist für normale, wiederbefüllbare Kisten gedacht. Er ist nicht für selbstfahrende Fahrzeuge, Anhänger, ankuppelbare Geräte, BigBags oder Verbrauchsmaterial-Paletten vorgesehen. Eine fehlende Füll- oder Entladefunktion ergänzt er nicht.

Die Kiste auch stapeln

Soll die Kiste im Lager übereinander stehen, ergänze zusätzlich objectStorageStacking aus Kapitel 9. Breite, Länge und Stapelabstand müssen zu deiner Kiste passen. Die Einlagerungsfreigabe allein legt diese Stapelwerte nicht fest.

11 Häufige Einbaufehler

Problem	Ursache / Eintrag prüfen
Unterlegpaletten fehlen	baleStorageVisuals enabled="true" und palletUnderlays enabled="true" müssen beide gesetzt sein.
Annahmebereich bewegt sich nicht	slotTrigger enabled="true" im passenden Lagerblock eintragen. Bei Paletten gehört er unter palletStacking, bei Ballen unter baleStorageVisuals.
Lagermenü am falschen Platz	movePlayerTrigger und playerTriggerOffset prüfen. Der Bedienbereich und der Einlagerbereich sind zwei getrennte Trigger.
Regal stapelt nur eine Lage	stackInMultiLevelStorage, Fachhöhe maxHeight und maxStackLayers der Palette prüfen.
Paletten stehen in Pfosten	storageAreas dürfen die Pfosten nicht einschließen. OSV erkennt die Gebäudegeometrie nicht automatisch.
Zwischen gestapelten Kisten ist Luft	stackHeight ist zu groß. Den Abstand zweier korrekt aufeinanderstehender Kisten verwenden.
Kisten stecken ineinander	stackHeight ist zu klein oder die Modellmaße stimmen nicht. Die Werte müssen zur wirklichen Kiste passen.
Kiste wird nicht angenommen	objectStorageReusablePallet muss in der Kisten-XML stehen. Außerdem müssen Inhalt und Größe zu den normalen Annahmeregeln des Lagers passen.
Lager nimmt trotz freier Stückzahl nichts an	Bei dynamicCapacity zählt der nutzbare Platz. Gruppenflächen, Abmessungen und hardLimit prüfen.