

# Steam Workshop für X Rebirth und X4

## Inhalt

- Allgemeine Regeln zur Veröffentlichung von Erweiterungen
- Vorbereitung
- Das Workshop-Tool benutzen
- Eine Erweiterung veröffentlichen
- Eine Erweiterung aktualisieren
- Mehrere Spielversionen unterstützen
- Eine Erweiterung sowohl für X Rebirth als auch für die VR Edition veröffentlichen

Mit dem Steam Workshop ist es jetzt einfacher als je zuvor, Erweiterungen für X-Spiele zu finden und zu installieren. Es gibt separate Workshops für X Rebirth, X Rebirth VR Edition und X4: Foundations. Stöbere im Workshop für Dein Spiel und abonniere Erweiterungen ("Objekte"), an denen Du interessiert bist. Sie werden automatisch installiert, wenn Du das Spiel das nächste Mal startest. Lies Dir aber zunächst die Beschreibungen durch und achte auf Inkompatibilitäten mit anderen abonnierten Erweiterungen.

Erweiterungen aus dem Workshop werden auch automatisch aktualisiert, wenn der Autor aktualisierte Inhalte zur Verfügung stellt. Um eine Erweiterung beim nächsten Spielstart wieder zu deaktivieren, bestell sie einfach ab, entweder direkt im Spiel oder auf der Steam-Website.



Mit einem eingeschränkten Steam-Account ist das Einsenden von Workshop-Inhalten nicht möglich.

X-Spiele auf Steam mit Workshop-Unterstützung:

- [X Rebirth \(Workshop\)](#)
- [X Rebirth VR Edition \(Workshop\)](#)
- [X4: Foundations \(Workshop\)](#) (⚠ UI-Modding zur Zeit instabil, siehe [Breaking Changes](#))

Eine Erweiterung mit dem X Workshop Tool zu veröffentlichen funktioniert für alle aufgeführten Spiele sehr ähnlich.

## Allgemeine Regeln zur Veröffentlichung von Erweiterungen

Allgemeine Regeln und Richtlinien:

- Veröffentliche keine Erweiterung, die du nicht selbst erstellt hast, es sei denn, der Autor der Erweiterung hat der Veröffentlichung zugestimmt.
- Veröffentliche ebenso keine Modifikationen von Erweiterungen anderer (z.B. korrigierte Versionen) ohne die Zustimmung des Autors.
- Veröffentliche die gleiche Erweiterung nicht mehrfach. Wenn du dir nicht sicher bist, ob das Hochladen funktioniert hat, überprüfe deine Workshop-Inhalte online.
- Wenn deine Erweiterung andere Erweiterungen voraussetzt oder möglicherweise mit anderen Erweiterungen inkompatibel ist, dann nutze den Beschreibungstext, um Spieler zu warnen.
- Erweiterungen unterstützen die Lokalisierung von Namen und Beschreibungen. Leider trifft dies für den Steam Workshop nicht zu. Der Titel und die Beschreibungstexte für Workshop-Inhalte sollten auf Englisch sein, es sei denn, der Inhalt richtet sich speziell an nicht-englischsprachige Spieler. Aber auch in diesem Fall stelle bitte sicher, dass englischsprachige Spieler dem Titel und der Beschreibung ansehen können, was die Erweiterung macht.

## Vorbereitung

Um eine Erweiterung im Steam Workshop zu veröffentlichen, benötigst du:

- Die [X Tools](#) (früherer Name: X Rebirth Tools)
- Den Ordner, der die Dateien deiner Erweiterung enthält, einschließlich der content.xml
- Eine Vorschaubilddatei, z.B. einen Screenshot (der entweder als JPG oder PNG vorliegen muss). Wir empfehlen ein Breitbildformat, 640x360 Pixel oder größer.

Weitere Voraussetzungen:

- Du musst einen Ordernamen gewählt haben. Wenn die Erweiterung heruntergeladen oder aktualisiert wird, wird sie in diesem Ordner unter "extensions" im Installationsordner des Spiels zu finden sein.
  - Sehr wahrscheinlich wirst du direkt aus dem Ordner "extensions" Dateien hochladen. In diesem Fall hast du bereits einen Ordernamen.
  - Es gibt mehrere Einschränkungen bei der Benennung:
    - Nur die folgenden Zeichen sind im Ordernamen erlaubt: a-z 0-9 . \_ - und Leerzeichen.
    - Der Name wird komplett kleingeschrieben. Jeder Großbuchstabe wird automatisch in einen Kleinbuchstaben

- umgewandelt.
    - Der Name kann maximal 32 Zeichen lang sein.
  - X Rebirth kann mehrere Workshop-Erweiterungen, die den gleichen Ordernamen nutzen, handhaben. Abonniert der Spieler mehrere von diesen gleichzeitig, wird es allerdings zu Fehlern kommen.
  - Wenn eine Erweiterung, die nicht aus dem Workshop stammt, den gleichen Ordner nutzt, wird das Spiel diesen nicht anrühren. Der Spieler wird den Konflikt manuell beheben müssen.
  - Wenn dein Ordnername sehr allgemein ist und wahrscheinlich auch von anderen Erweiterungen benutzt wird, solltest du in Betracht ziehen, ihn zu ändern.
  - Wenn du dich dazu entscheidest, deinen Ordner in einem Update umzubenennen, dann bedenke alle Konsequenzen. Vergiss z.B. das Aktualisieren deiner Index-Dateien nicht!
- Alle spielrelevanten Dateien müssen in Katalogdateien gespeichert werden. Allerdings kann dir das Workshop-Tool dabei helfen.
    - Wenn in deinem Erweiterungs-Ordner keine Katalogdateien vorhanden sind, kannst du den Schalter `-buildcat` nutzen, um den Katalog automatisch erstellen zu lassen.
    - Um Kataloge manuell zu erstellen, kannst du das Katalog-Tool aus den X Tools nutzen.
    - Nur die folgenden Dateitypen werden hochgeladen: `*.cat`, `*.dat`, `*.cur`, `*.txt`, `*.pdf`
    - Wenn deine Erweiterung Mauszeigerdateien (`*.cur`) enthält, müssen diese in deinem Hauptverzeichnis liegen, und nicht in einem Unterordner. Unterordner werden nicht hochgeladen.
  - Anmerkungen zur `content.xml`:
    - Der Wert des `id`-Attributs hat vor der Veröffentlichung keine Bedeutung. Er wird mit einer Workshop-spezifischen ID ersetzt.
      - Die Attribute `name` (Name) und `description` (Beschreibung) müssen im `<content>`-Element vorhanden sein. Diese werden für Titel und Beschreibung des veröffentlichten Workshop-Inhalts genutzt.
      - Um Absätze im `description`-Attribut einzufügen, nutze `&#10;` (`&#13;&#10;` funktioniert auch). `\n` funktioniert hingegen nicht.
      - Titel und Beschreibung können im Steam Workshop nach der Veröffentlichung geändert werden.
      - Das Spiel zeigt den aktuellen Titel und die Beschreibung aus dem Steam Workshop an, es sei denn, die `content.xml` liefert lokalisierte Texte für die aktuelle Spielsprache.
      - Wir empfehlen, die Versionsnummer nicht im Namen mit anzugeben.
    - Du benötigst ein Attribut `version`. Der Wert ist die Versionsnummer multipliziert mit 100. Um z.B. `v2.50` anzuzeigen, musst du `version="250"` angeben.
    - Standardmäßig wird als "Autor" im Spiel dein öffentlicher Steam-Community-Name angezeigt. Wenn du einen anderen Autor-Namen anzeigen möchtest, musst du das `author`-Attribut im `<content>`-Element angeben.
    - Deine Erweiterung kann andere Erweiterungen als "dependency" voraussetzen, aber diese müssen bereits im Workshop existieren. Nutze dafür die IDs aus deren `content.xml`-Dateien.
    - Du kannst außerdem eine X Rebirth-Version zur Voraussetzung erklären (d.h. die benötigte Mindestversion). Dies ist eine "dependency" ohne die ID einer Erweiterung.

Beispiel für eine `content.xml`, um eine neue Erweiterung zu veröffentlichen:

```
<?xml version="1.0" encoding="utf-8" ?>
<!-- Meine Test-Erweiterung v0.10 -->
<content id="total_unwichtig" name="My Test Extension" description="This is just an example."
version="10">
  <!-- Benötigt Workshop-Objekt 12345, Mindestversion 1.00 -->
  <dependency id="ws_12345" version="100" />
  <!-- Benötigt mindestens Spielversion 1.50 -->
  <dependency version="150" />
  <!-- Optionale Lokalisierung (BEACHTET: Diese ist nicht sichtbar im Steam Workshop und kann
auf der Internetseite des Workshops nicht aktualisiert werden) -->
  <!-- Deutsch: -->
  <text language="49" name="Meine Test-Erweiterung" description="Dies ist nur ein Beispiel."
/>
</content>
```

(BEACHTET: Bei den Texten zwischen den Pfeilen `<!-- -->` handelt es sich um Kommentare, die hier der Veranschaulichung dienen. Sie müssen in der eigentlichen `content.xml` nicht mitangegeben werden.)

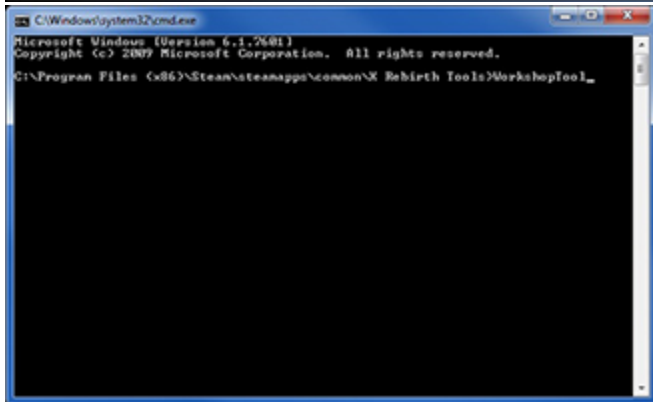
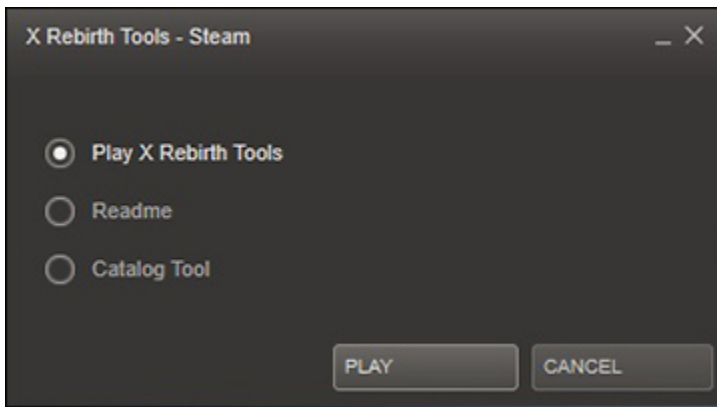
Minimalistisches Beispiel:

```
<?xml version="1.0" encoding="utf-8" ?>
<content id="" name="Foo" description="Test" version="100">
</content>
```

## Das Workshop-Tool benutzen

Das Workshop-Tool ist in den X Tools enthalten. Es ist ein Kommandozeilen-Programm, eine grafische Benutzeroberfläche haben wir (noch) nicht. Du benötigst daher eine Kommandozeile, um es laufen zu lassen. Der einfachste Weg dies zu tun, ist es, die "Spielen"-Option der X Tools auszuwählen. Dies wird die Windows-Eingabeaufforderung im X Tools Ordner öffnen, wo sich die `WorkshopTool.exe` befindet.

(Beachte: Wenn die Tool-App bereits vor ihrer Umbenennung installiert war, bleibt der ursprüngliche Ordnername "X Rebirth Tools" unverändert.)



Tippe **WorkshopTool** und drücke Enter. Dies gibt dir eine Übersicht über die zur Verfügung stehenden Befehle und Schalter. Hab keine Angst etwas auszuprobieren, das Tool wird eine Bestätigung verlangen, bevor es irgendetwas hochlädt.

## Eine Erweiterung veröffentlichen

**Wichtig:** Du musst den [Workshop-Nutzungsbedingungen](#) von Steam zustimmen, um Inhalte zum Steam Workshop hochladen zu können.

Nehmen wir an, du hast eine Erweiterung für X Rebirth im "extensions"-Ordner vorbereitet:

```
...steamapps\common\X Rebirth\extensions\my first mod
```

Du hast außerdem ein Vorschaubild (siehe oben). Es heißt `preview.jpg` und es befindet sich in deinem Erweiterungsordner ("my first mod"). Beachte den kleingeschriebenen Ordernamen. Nun ist es Zeit, das `WorkshopTool` auszuführen! Die Kommandozeile, um deine Mod zu veröffentlichen, sollte wie folgt aussehen, abhängig vom Spiel:

Beispiel für X Rebirth:

```
WorkshopTool publish -path "..\X Rebirth\extensions\my first mod" -preview "..\X Rebirth\extensions\my first mod\preview.jpg" -buildcat
```

Beispiel für X Rebirth VR Edition:

```
WorkshopTool publishvr -path "..\X Rebirth VR\extensions\my first mod" -preview "..\X Rebirth VR\extensions\my first mod\preview.jpg" -buildcat
```

Beispiel für X4: Foundations:

```
WorkshopTool publishx4 -path "..\X4 Foundations\extensions\my first mod" -preview "..\X4 Foundations\extensions\my first mod\preview.jpg" -buildcat
```

Du kannst eine Erweiterung nur für ein Spiel veröffentlichen, das dir auf Steam gehört, ansonsten wird der entsprechende `publish`-Befehl fehlschlagen.

[In Zukunft möchtest du vielleicht auch den `-tags` Schalter nutzen. Tags sind aber noch nicht implementiert. Komme später wieder, um zu schauen, ob sich dies geändert hat.]

**Du musst bei Steam angemeldet sein, während das Workshop-Tool läuft.**

Also was passiert jetzt?

- Wir nutzen hier nicht den Schalter `-foldername` (zum Ändern des Ordernamens), sodass das `Workshop-Tool` den tatsächlichen

- Ordnernamen übernimmt: "my first mod" (Beachte, wie oben erläutert, dass dies ein schlechter Name ist!)
- Auch den `-contentdef` Schalter nutzen wir hier nicht, sodass das Workshop-Tool die `content.xml` im "my first mod"-Ordner übernimmt.
- Da wir nicht selbst Katalogdateien erstellt haben, nutzen wir den Schalter `-buildcat`, damit das Workshop-Tool alles für uns macht.
- Das Katalog-Tool baut `ext_01.cat / ext_01.dat`, die zusammen mit den anderen Mod-Dateien hochgeladen werden. Diese Katalogdateien werden automatisch gelöscht, sobald das Workshop-Tool seine Arbeit beendet hat. Du kannst die Löschung verhindern, indem du den `-keepcatfiles` Schalter nutzt. Vergiss in diesem Fall aber nicht, die Kataloge später manuell zu aktualisieren oder zu löschen!
- Das WorkshopTool überprüft die Metadaten in der `content.xml` und gibt einige Informationen aus. Lies dir bitte alles durch und stelle sicher, dass es keine Fehler gibt.
- Das Tool fragt nach einer Bestätigung: "Start upload to the Steam cloud (y/n)?" (dt.: "Beginne das Hochladen zur Steam Cloud (y/n)?"). Tippe Y (für Ja) oder N (für Nein) ein und drücke Enter.
- Warte jetzt einfach, bis das Hochladen beendet ist. Dies kann eine Weile dauern. Mit der Tastenkombination Strg+C kannst du den Vorgang abbrechen.
- Wenn das Hochladen beendet wurde, werden einige XML-Attribute in die `content.xml` zurückgeschrieben. Am wichtigsten hierbei ist das `id`-Attribut, welches die Erweiterung als Workshop-Objekt identifiziert.

Nach dem erfolgreichen Hochladen wird du gefragt, ob du die Internetseite für dein Workshop-Objekt öffnen möchtest. Du solltest dies aus folgenden Gründen tun:

- Unter Umständen musst du online die Workshop-Nutzungsbedingungen bestätigen.
- Auch nach dem Hochladen ist das Workshop-Objekt noch nicht für andere Spieler sichtbar. Die Sichtbarkeit musst du auf der Workshop-Seite (entweder für jeden oder nur für Freunde) einstellen.
- Auf der Workshop-Seite kannst du Titel und Beschreibung direkt ändern, ohne den Inhalt aktualisieren zu müssen.

Wenn dir die Internetseite sagt, dass das Workshop-Objekt nicht existiert, bist du wahrscheinlich nicht eingeloggt. Logge dich ein, um deine Erweiterung zu sehen.

Wenn du später zu dieser Seite zurückkehren möchtest, lege entweder ein Lesezeichen in deinem Browser an, oder starte das Workshop-Tool mit dem Befehl `showpage`:

```
WorkshopTool showpage -path "..\X Rebirth\extensions\my first mod"
```

Wenn du eine Erweiterung im Workshop abonniert hast und das Spiel startest, werden deren Dateien heruntergeladen und überschreiben möglicherweise Dateien früherer Versionen. Dies passiert nicht für deine eigene Erweiterung, da das Workshop-Tool nach der Veröffentlichung `sync="false"` in deine `content.xml` schreibt. Somit kannst du deine Erweiterung sicher in deinem Spielordner halten, während du an Updates arbeitest. (Nichtsdestotrotz ist es immer eine gute Idee, deine Dateien regelmäßig zu sichern.) Wenn du die Synchronisierung dennoch aktivieren möchtest, ist dies auch ohne Weiteres möglich: Gehe ins Erweiterungen-Menü im Spiel, wähle deine Erweiterung und setze "Updates" auf "Ja".

Eine Nebenwirkung deaktivierter Updates ist, dass das Spiel den Namen und die Beschreibung deiner lokalen `content.xml` nicht ändern wird, wenn du sie im Steam Workshop änderst. Andere Spieler werden aber den aktualisierten Namen und die Beschreibung in ihren Spielen sehen.

Sobald Spieler deine Erweiterung im Workshop sehen können, können sie sie abonnieren und installieren, indem sie das Spiel starten. Erweiterungen werden deaktiviert, indem man sie wieder abbestellt. Wenn nur der Ordner gelöscht wird, ohne die Erweiterung abzubestellen, wird sie beim nächsten Spielstart erneut heruntergeladen.

Wenn deine Erweiterung von anderen Workshop-Erweiterungen abhängt, müssen Spieler all diese manuell abonnieren, andernfalls erhalten sie eine Fehlermeldung im Spiel. Die benötigten Erweiterungen werden im spielinternen Erweiterungen-Menü angezeigt und können dort ebenfalls abonniert werden.

## Eine Erweiterung aktualisieren

Um eine vorhandene Workshop-Erweiterung zu aktualisieren, könnte eine Kommandozeile wie folgt aussehen:

```
WorkshopTool update -path "..\X Rebirth\extensions\my first mod" -buildcat -changenote "This is my first update"
```

Der Befehl "update" unterscheidet sich vom Befehl "publish" zum Veröffentlichenden einer Erweiterung in einigen Punkten:

- Der Schalter `-changenote` (Änderungshinweis) wird für alle Updates benötigt. Schreibe `-changenote ""` wenn du keine Beschreibung bereitstellen möchtest. Du kannst auch die Änderungshinweise auf der Workshop-Seite nach dem Aktualisieren editieren.
- Das Workshop-Tool erwartet, dass du die Versionsnummer erhöht hast. Wenn du die Versionsnummer in deiner `content.xml` absichtlich unverändert lassen möchtest, nutze den Schalter `-minor`
- Für gewöhnlich lässt ein Update den Titel und die Beschreibung im Workshop unverändert. Um die Attribute `name` (Name) und `description` (Beschreibung) von der `content.xml` dennoch zum Steam Workshop hochzuladen, nutze den Schalter `-namedesc up` (ohne Anführungsstriche).
- Die Änderung der Vorschaudatei ist optional. Wenn du die Vorschaudatei ändern möchtest, ohne neue Inhalte hochzuladen, nutze den Befehl `updatepreview` statt `update`.
- Es ist möglich, den Ordnernamen der Erweiterung zu ändern. **Wichtig:** Stelle sicher, dass du dies nicht versehentlich tust, z.B. indem du von einem anderen Ordner aus hochlädst! Das Workshop-Tool zeigt vor dem Hochladen den alten und neuen Ordnernamen an. Bitte überprüfe diese Ausgabe sorgfältig.
- Nach dem Hochladen werden einige XML-Attribute zurück in die `content.xml` geschrieben, wie beim Veröffentlichenden, aber das `sync`-Attribut

ut wird dabei nicht hinzugefügt oder geändert. Hast du also nach der Veröffentlichung Updates aktiviert, werden diese jetzt nicht deaktiviert.

- Der Befehl "update" funktioniert für alle unterstützten X-Spiele. Das entsprechende Spiel wird automatisch erkannt anhand der ID des Workshop-Beitrags in der content.xml.

Wenn das Spiel eine Erweiterung installiert, überprüft es die heruntergeladene content.xml nach einer vorausgesetzten Spielversion. Wenn die Erweiterung eine höhere als die zurzeit installierte Spielversion benötigt, wird der Download (mit einer Warnung) verworfen und stattdessen weiterhin die bereits installierten Dateien genutzt. Änderst du als Workshop-Autor also die benötigte Spielversion, wird dies nicht das Spiel von anderen kaputt machen. Solche Sicherheitsüberprüfungen gibt es jedoch nicht, wenn du die "dependencies" von anderen Workshop-Erweiterungen und ihrer Versionen änderst – das Update wird dann trotzdem heruntergeladen und installiert.

## Mehrere Spielversionen unterstützen

Unter Umständen möchtest du vielleicht verschiedene Dateien für verschiedene Spielversionen in der gleichen Workshop-Erweiterung bereitstellen. Das ist nützlich, wenn Egosoft eine neue Beta-Version des Spiels bereitstellt und du deine Mod für diese Beta anpassen möchtest, ohne dass alle Spieler der öffentlichen Version deiner Mod von diesem Update beeinflusst werden. Zum Beispiel könnte eine neue Beta-Version neue Funktionen einführen, die du in deiner Mod nutzen möchtest oder deine Mod könnte inkompatibel werden und muss angepasst werden. Du kannst die Beta-Phase nutzen, um die notwendigen Änderungen vorzunehmen und kannst damit beide Versionen, öffentliche und Beta, gleichzeitig unterstützen.

Und so funktioniert es: Wie in der Dokumentation zum Katalog-Tool beschrieben, wird das Spiel in deinem Erweiterungsordner nach einem Versionskatalog suchen, der deiner aktuellen Spielversion entspricht, z.B. würde die Spiel-Version 1.50 nach ext\_v150.cat suchen. Existiert dieser Katalog, wird er genutzt, um die anderen Kataloge deiner Erweiterung zu überschreiben (im Normalfall nur ext\_01.cat).

Wenn du das Katalog-Tool nicht manuell nutzt, kann das Workshop-Tool die ganze Arbeit für dich übernehmen. Angenommen du hast eine Erweiterung, die mit der aktuellen Version von X Rebirth (sagen wir mal 1.50) funktioniert und es gibt eine Beta für X Rebirth (sagen wir mal Version 2.00). Du möchtest nun Änderungen an deiner Erweiterung vornehmen, um sie für v2.00 vorzubereiten. Dann gehst du wie folgt vor:

- Du erstellst einen Unterordner mit dem Namen "v150" für Spielversion 1.50.
- Du kopierst \*alles\* aus deinem bestehenden Erweiterungsordner in diesen Ordner (mit Ausnahme der content.xml und früheren Versions-Ordner, wie "v140"). Dies ist der Inhalt, den Spieler in Spielversion 1.50 sehen werden.
- In deinem normalen Erweiterungs-Ordner nimmst du die Änderungen für Version 2.00 vor.
- Setze die benötigte Spielversion von X Rebirth in der content.xml auf 1.50, nicht 2.00, da diese Datei ebenfalls in Version 1.50 genutzt wird. (Wenn du Spielversionen älter als 1.50 erlaubst, dann solltest du auch Versions-Ordner für diese bereitstellen.)
- Wenn du bereit bist, das Update hochzuladen, nutze das Workshop-Tool mit *-buildvcat* statt *-buildcat*. Es ist prinzipiell der gleiche Schalter, aber *-buildvcat* sucht auch nach solchen Versions-Ordner. Es wird v150 finden und einen diff-Katalog ext\_v150.cat/dat erstellen. Dieser wird nur Dateien enthalten, die vom 2.00-Inhalt abweichen.
- Beide Kataloge ext\_01 (mit dem 2.00-Inhalt) und ext\_v150 werden hochgeladen und den Spielern bereitgestellt. Spieler der Version 1.50 werden den 1.50-Inhalt sehen und Spieler der Beta-Version 2.00 den neuen Inhalt (da ext\_v150 nicht geladen wird).
- Sobald Version 2.00 für alle Spieler veröffentlicht wurde, kannst du den v150-Ordner entfernen und die benötigte Spielversion in der content.xml auf "200" setzen, es sei denn, du möchtest die Unterstützung für Spieler beibehalten, die die Spielupdates deaktiviert haben, die Workshop-Updates aber aktiviert haben.

Wenn du vorhast, alle deine Kataloge und die Versions-Kataloge manuell zu erstellen, sei dir bewusst, dass du sie mit jedem Update neu erstellen musst – die Versions-Kataloge müssen stets ein "diff" zur jeweils aktuellen Version sein.

## Eine Erweiterung sowohl für X Rebirth als auch für die VR Edition veröffentlichen

Es ist wie folgt möglich, dieselbe Erweiterung für beide Spiele zu veröffentlichen:

- Stelle sicher, dass Deine Erweiterung mit beiden Spielen kompatibel ist. Es gibt subtile Unterschiede in XR VR, die Probleme mit bestehenden XR-Erweiterungen verursachen können, und umgekehrt.
  - Wenn Du vom jeweiligen Spiel abhängiges Verhalten in Scripts benötigst, kannst du in Scripts erkennen, ob die VR-Version läuft. Auf diese Weise kann das gleiche Script in beiden Spielen eingesetzt werden.
- Du wirst zwei Workshop-Objekte bekommen, nicht einen. Daher brauchst du auch zwei content.xml-Dateien, selbst wenn du sie mit einer einzigen Erweiterung benutzt.
  - Erstelle zwei Kopien dieser Datei, eine pro Spiel. Bei der Veröffentlichung und bei Updates deiner Erweiterung musst du mit *-contentdef* den jeweiligen Dateinamen angeben.
  - Veröffentliche die Erweiterung zweimal: Zuerst mit "publish", dann mit "publishvr", mit den jeweiligen content.xml-Versionen.

Um die Erweiterung für beide Spiele zu aktualisieren, verwende einfach den Befehl "update" für beide Workshop-Objekte wie oben beschrieben.