

Learning Path | MusicMaster | Handout

Jingleplanung



SwissMediaPartners AG Waaghausgasse 18 3011 Bern



Dokumentenhistorie

Version	Datum	Autor	Bemerkung
1	19.08.2019	Kevin Hopkins	
1 1	10 09 2019	Mathieu Habegger	Annassungen Schweiz



Inhaltsverzeichnis

Ein	Einleitung					
	Feld erstellen					
2.	Attribute erstellen	5				
3.	Attribute für Songs, Jingles und Verpackungselemente erfassen	5				
4.	Regel erstellen	6				
5.	Regelmäßige Planung	7				
	!					



Einleitung

Gerade in Automationsstunden kann es zeitsparend sein, Jingles und andere Verpackungselemente zwischen den Songs automatisiert zu planen. Diese sollen sich in der Regel auf das Tempo der Songs beziehen. Auch Sounds oder ähnlichen Merkmalen können als Kriterium mit in die Planung integriert werden.

Alle notwendigen Arbeitsschritte sind in diesem Handout beschrieben.

Vorausgesetzt für die Arbeit mit diesem Dokument wird der Abschluss aller Basismodule des MusicMaster Learning Paths bzw. eine gute Kenntnis der Grundfunktionalitäten der Software.

Vorausgesetzt wird ebenso, dass in MusicMaster bereits Jingles und Verpackungselemente in NonMusic-Kategorien als Elemente vorhanden sind und Sendeuhren mit entsprechenden Positionen erstellt wurden.

1. Feld erstellen¹

Damit MusicMaster das Tempo eines jeden Songs/Verpackungselements versteht und passend verbinden kann, müssen wir zunächst ein entsprechendes Feld hinzufügen. Darin tragen wir das *In-Tempo* sowie das *Out-Tempo* sämtlicher benötigter Elemente ein.

Menü → Dataset → Library → Fields → Add Field

Wir benötigen an dieser Stelle ein Feld mit Data Type **Attribute-In/Out**, Vorschlag für die Bezeichnung: **Übergangstempo**.

Im Bereich *Length* tragen wir in unserem Beispiel "2" für zwei Positionen ein. Als erste Position vermerken wir später das Tempo des *Intros*, an zweiter Position das Tempo der *Outros*.

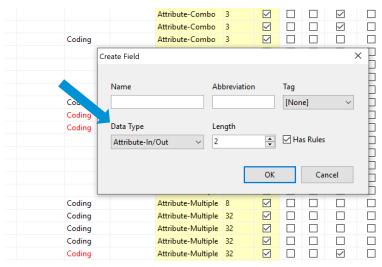


Abb. 1: Feld vom Type Attribute-In/Out mit zwei Positionen

¹ Bis MusicMaster Professional Edition Version 6 wende dich zur Felderweiterung in der Datenbank bitte an unseren Support unter support@onair.de.



2. Attribute erstellen

Für das erstellte Attribute-Feld stellen wir nun ein, welche Geschwindigkeiten für uns in der Planung wichtig sind. Im Beispiel entscheiden wir uns für 1, 2 und 3 (langsam, mittel, schnell).

Menü → Dataset → Library → Attributes → Übergangstempo

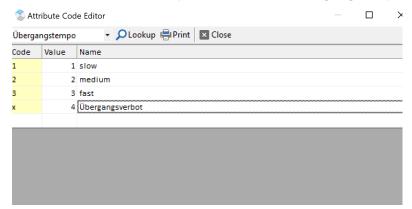


Abb. 2: Erfassung der Attribute-Werte im Attribute Code Editor

Um zu verhindern, dass ein Jingle oder Verpackungselement bspw. vor einem gebrandeten Song geplant wird, erstellen wir zusätzlich den Code "x" für ein Übergansgverbot.

3. Attribute für Songs, Jingles und Verpackungselemente erfassen

In der Datenbank müssen für alle Songs, Jingles und Verpackungselemente jetzt die beiden Übergangspositionen im erstellten Attribute-Feld erfasst werden. Ja. Ist Fleißarbeit. ;-)

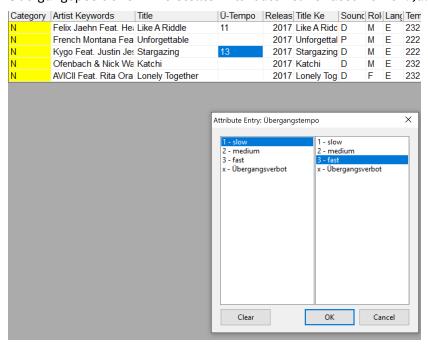


Abb. 3: Erfassung der Übergangspositionen im Attribute-Feld



4. Regel erstellen

Über eine Planungsregel erlauben wir MusicMaster jetzt bestimmte Attribute-Reihenfolgen bzw. wir verbieten sie und steuern damit die **Übergangsmöglichkeiten**.

Menü → Dataset → Rule Tree

Dabei nutzen wir die Regel **Seque Protection**. Ihr findet diese im Rule Tree unter

Attribute Fields → Übergangstempo2 → Segue Protection

Setzt nun ein "x" an den Stellen, an denen ein Übergang verboten werden soll und vermerkt, dass die Regel *Forward* und *Backwards* getestet werden soll.

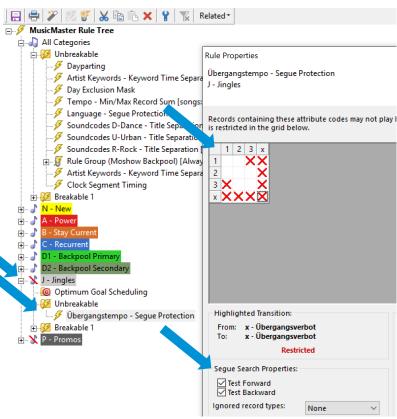


Abb. 4: Eigenschaften der Segue Protection

Die Regel wird dabei in die Kategorie der Jingles bzw. Verpackungselemente als *unbrechbare Regel* gezogen, in unserem Beispiel in die schon vorhandene Kategorie *J - Jingles*.

² Habt ihr eine andere Bezeichnung gewählt, dann steht diese an dieser Stelle.





Regelmäßige Planung 5.

Für die regelmäßige Planung starten wir nun den **Schedule Editor**.

Bei Erstellung der Playlist berücksichtigt MusicMaster nun die vorgegeben Übergangsmöglichkeiten zwischen Songs und Jingles bzw. Verpackungselementen.



Abb. 5: Playlist mit Songs und automatisch geplanten Jingles



Notizen		