

1.Règles

Les règles permettent de contrôler finement la programmation musicale. Elles servent à spécifier quels morceaux peuvent passer après tels autres, combien de temps doit s'écouler avant qu'un même morceau ne soit programmé à nouveau, etc. Ces règles sont exprimées par une simple ligne de texte, dans un langage simple.

Les règles sont définies indépendamment de tout programme et rassemblées au sein de groupes de règles. Un programme est associé à un groupe de règle. Il est donc possible d'appliquer simplement la même politique de règles à plusieurs programmes sans avoir à les récrire à chaque fois.

1.Règles de séparation

Ces règles permettent de spécifier un écart minimum entre deux passages de morceaux vérifiant certains critères. L'écart peut-être donné en temps ou en nombre de morceaux. Toutes ces règles débutent par le mot-clef **SEPARATION** et expriment la durée de la séparation sous la forme ***N unité_de_durée*** où *N* est un nombre entier, et *unité_de_durée* est à choisir parmi :

- **SONG** : séparation en nombre de morceaux entre deux passages
- **MINUTE** : séparation de *N* minutes entre deux passages
- **HOURL** : séparation de *N* heures entre deux passages
- **DAY** : séparation de *N* jours entre deux passages

1.Séparation de morceaux

Cette règle permet de spécifier un écart minimal entre deux passage d'un même morceau.

Syntaxe de la règle :

SEPARATION SONG *N unité_de_durée*

Exemples :

SEPARATION SONG 12 SONG
SEPARATION SONG 45 MINUTE
SEPARATION SONG 6 HOURL
SEPARATION SONG 1 DAY

2.Séparation d'artistes

Cette règle permet de spécifier un écart minimal entre deux passage de morceaux d'un même artiste.

Syntaxe de la règle :

SEPARATION ARTIST *N unité_de_durée*

Exemples :

SEPARATION ARTIST 10 SONG
SEPARATION ARTIST 90 MINUTE
SEPARATION ARTIST 3 HOURL
SEPARATION ARTIST 1 DAY

3.Séparation de groupes

Ces règles permettent de spécifier un écart minimal entre deux passages de morceaux rassemblés au sein d'une même liste. Il y a deux règles, une permettant d'exprimer la séparation pour une liste donnée, et une permettant d'exprimer la séparation pour plusieurs listes (un groupe de liste en somme).

Première forme :

SEPARATION PLAYLIST [*id*] *N unité_de_durée*

id est un identifiant (numérique) de liste. La séparation est appliquée entre deux morceaux appartenant à la liste précisée.

Deuxième forme :

SEPARATION PLAYLISTS MARKED [*mark*] *N unité_de_durée*

La séparation est appliquée entre deux morceaux appartenant tout deux à l'une des listes marquée **mark**. C'est équivalent à utiliser des règles de séparation identiques pour chacune des listes marquées. Une marque est simplement un identifiant textuel alphanumérique permettant de caractériser une liste, par exemple pour regrouper des listes d'usage similaires (voir la section consacrée à la base de données pour l'emploi d'une table de marques).

Note : ces règles ne sont pas appliquées à la playlist utilisée au moment de sélectionner le morceau.

Par exemple au sein d'un programme aléatoire défini par **PLAYLIST n**, une règle **SEPARATION PLAYLIST [n] ...** n'a aucun effet.

Exemples :

SEPARATION PLAYLIST [15] 4 SONG

SEPARATION PLAYLISTS MARKED [Pour_Separation] 15 MINUTE

4.Séparation sur critère

Ces règles permettent de spécifier un écart minimal entre deux passages de morceaux vérifiant un même critère. Il y a deux règles, une permettant d'exprimer la séparation pour une valeur donnée d'un critère, et une permettant d'exprimer la séparation pour toutes les valeurs d'un critère.

Première forme :

SEPARATION CRITERIA [nom_du_critère] VALUE [valeur] N unité_de_durée

La séparation est appliquée entre deux morceaux pour lesquels le critère **nom_du_critère** est égal à **valeur**.

Deuxième forme :

SEPARATION CRITERIA [nom_du_critère] N unité_de_durée

La séparation est appliquée entre deux morceaux pour lesquels les valeurs du critère **nom_du_critère** sont égales.

Exemples :

SEPARATION CRITERIA [Genre] 10 MINUTE

SEPARATION CRITERIA [Genre] VALUE [Musette] 5 SONG

2.Règles de contrôle des enchaînements

Ces règles permettent de contrôler le choix d'un morceau en fonction du précédent (ou des précédents directs). Toutes ces règles débutent par le mot-clef **SEQUENCE**.

5.Contrôle des enchaînements de groupes

Cette règle contrôle quels groupes de chansons peuvent suivre tels autres. Comme pour la règle de séparation, elle est déclinée en deux variantes : une pour interdire un groupe après un autre, et une pour forcer un groupe après un autre.

Interdiction :

SEQUENCE FORBID PLAYLIST [id1] AFTER PLAYLIST [id2] p%

Les chansons de la liste **id2** ne pourront jamais passer après les chansons de la liste **id1**. **p** est un paramètre optionnel spécifiant que la règle aura p% de chance d'être appliquée. Si **p** n'est pas donné, la règle est toujours appliquée (c'est équivalent à donner une valeur de **p** égale à 100).

Forcer un enchaînement :

SEQUENCE FORCE PLAYLIST [id1] AFTER PLAYLIST [id2] p%

Aura pour effet de faire en sorte qu'après une chanson de la liste **id1** passera une chanson de la liste **id2**. Là encore on peut préciser un pourcentage d'application de la règle.

Exemples :

SEQUENCE FORCE PLAYLIST [15] AFTER PLAYLIST [10]

SEQUENCE FORCE PLAYLIST [15] AFTER PLAYLIST [10] 50%

(50% de chance d'avoir un morceau de la liste 10 après un morceau de la liste 15)

SEQUENCE FORBID PLAYLIST [10] AFTER PLAYLIST [12]

SEQUENCE FORBID PLAYLIST [10] AFTER PLAYLIST [13] 80%

Notez que l'usage du pourcentage est surtout utile dans le cas d'une règle **FORCE**, puisque si une règle **FORBID** n'est pas appliquée, ça ne veut pas dire pour autant qu'un morceau de la liste interdite sera programmé.

6. Contrôle des enchaînements sur critère

Ces règles contrôlent les valeurs d'un critère pour un morceau en fonction de la valeur d'un critère du morceau précédent.

Contrôle de la différence entre deux critères :

SEQUENCE CONTROL DELTA CRITERIA [*critère*] *opérateur* *X*

X est une valeur numérique positive, *opérateur* est à choisir parmi < et >, et *critère* est un critère à valeurs numériques. La règle assure que la différence en valeur absolue entre les valeurs de *critère* pour deux morceaux consécutifs est inférieure (<) ou supérieure (>) à *X*.

Exemples :

SEQUENCE CONTROL DELTA CRITERIA [Beat] < 15

SEQUENCE CONTROL DELTA CRITERIA [Energy] < 2

SEQUENCE CONTROL DELTA CRITERIA [Yoyo] > 2

Contrôle de la moyenne mobile d'un critère :

SEQUENCE CONTROL SUM CRITERIA [*critère*] *opérateur* *X* **OVER** *N* **SONG**

Assure que la somme du critère *critère* sur les *N* derniers morceaux est supérieure ou inférieure à *X*.

Exemples :

SEQUENCE CONTROL SUM CRITERIA [Energy] < 15 OVER 3 SONG

SEQUENCE CONTROL SUM CRITERIA [Longueur] < 600 OVER 2 SONG

SEQUENCE CONTROL SUM CRITERIA [Vitesse] > 10 OVER 4 SONG

Interdiction :

SEQUENCE FORBID CRITERIA [*critère_1*] **VALUE** [*valeur_1*] **AFTER CRITERIA** [*critère_2*] **VALUE** [*valeur_2*] *p*%

Un morceau dont la valeur de *critère_1* est égale à *valeur_1* ne sera jamais suivi d'un morceau dont *critère_2* est égal à *valeur_2*. Le pourcentage a la même signification que pour la règle similaire avec des listes.

Exemple :

SEQUENCE FORBID CRITERIA [Genre] VALUE [Rock] AFTER CRITERIA [Genre] VALUE [Techno]

Forcer un enchaînement :

SEQUENCE FORCE CRITERIA [*critère_1*] **VALUE** [*valeur_1*] **AFTER CRITERIA** [*critère_2*] **VALUE** [*valeur_2*] *p*%

Un morceau dont la valeur de *critère_1* est égale à *valeur_1* sera suivi d'un morceau dont *critère_2* est égal à *valeur_2* (avec *p*% de chances).

Exemple :

SEQUENCE FORCE CRITERIA [Genre] VALUE [Rock] AFTER CRITERIA [Genre] VALUE [Jazz] 20%

7. Contrôle de fréquence des groupes

Cette règle contrôle le nombre de morceaux appartenant à une liste donnée sur les *N* derniers morceaux ou au cours d'un laps de temps écoulé. Il y a deux formes pour la règle : une opérant sur une liste en particulier, et une opérant sur un ensemble de listes marquées.

Pour une liste :

SEQUENCE CONTROL PLAYLIST [*id*] *Nb* **OVER** *N* *unité_de_durée*

Pour des listes marquées :

SEQUENCE CONTROL PLAYLISTS MARKED [mark] Nb OVER N unité_de_durée

Cette règle est équivalente à donner une règle pour chaque liste marquée *mark*.

Exemples :

SEQUENCE CONTROL PLAYLIST [15] 4 OVER 10 SONG

(Pas plus de 4 morceaux de la liste 15 dans 10 morceaux successifs)

SEQUENCE CONTROL PLAYLIST [15] 4 OVER 30 MINUTE

(Pas plus de 4 morceaux de la liste 15 en 30 minutes)

SEQUENCE CONTROL PLAYLISTS MARKED [Listes_limitees] 3 OVER 10 SONG

8. Contrôle de fréquence d'un critère

Cette règle contrôle le nombre de morceaux vérifiant un critère donné sur les N derniers morceaux ou au cours d'un laps de temps écoulé. Il y a deux formes pour la règle : une opérant sur une valeur particulière d'un critère, et une opérant sur toutes les valeurs d'un critère.

Pour une valeur particulière :

SEQUENCE CONTROL CRITERIA [critère] VALUE [valeur] Nb OVER N unité_de_durée

Pour des listes marquées :

SEQUENCE CONTROL CRITERIA [critère] Nb OVER N unité_de_durée

Cette règle est équivalente à donner une règle pour chaque valeur possible du critère précisé.

Exemples :

SEQUENCE CONTROL CRITERIA [Genre] VALUE [Rock] 4 OVER 10 SONG

(Pas plus de 4 morceaux de rock dans 10 morceaux successifs)

SEQUENCE CONTROL CRITERIA [Genre] 4 OVER 30 MINUTE

(Pas plus de 4 morceaux du même genre en 30 minutes)

9. Contrôle intelligent des longueurs

Cette règle automatise la gestion des enchaînements des morceaux en fonction de leur durée. Elle fait en sorte de ne pas enchaîner trop de morceaux trop longs ou trop courts, en tenant compte de la durée moyenne des morceaux du programme concerné.

Cette règle s'exprime simplement :

SMART LENGTH CONTROL

3. Autres règles

10. Contrôle de valeur d'un critère

Cette règle permet de contrôler les valeurs autorisées pour un critère donné, sans tenir compte des morceaux précédents.

CONTROL CRITERIA [critère] opérateur valeur

opérateur peut être < (inférieur), > (supérieur), = (égal), ou <> (différent). *valeur* peut-être numérique ou alphanumérique.

Exemples :

CONTROL CRITERIA [Genre] = Rock

CONTROL CRITERIA [Tempo] > 3

CONTROL CRITERIA [Length] < 300

CONTROL CRITERIA [Genre] <> Musette

11. Roulement des plages horaires

Ces deux règles permettent d'appliquer un roulement entre les plages horaires où est diffusé un morceau.

Roulement automatique :

SMART DAYPARTS

active le roulement « intelligent » des plages horaires : un morceau ne sera pas diffusé dans la même plage que sa diffusion précédente. La plage est définie par la tranche horaire de 3 heures avant et après la diffusion précédente.

Roulement spécifique :

SONG DAYPARTS

active l'utilisation de la table des roulements, qui permet de définir avec précision des plages d'exclusion pour la diffusion des morceaux, des artistes, et des listes.