

# Nabaztag API (V1)

## De quoi s'agit-il ?

Il s'agit de la possibilité à partir d'une application extérieure de déclencher des événements (modification de la vitesse d'affichage, commande des oreilles, commande des leds, envoi d'un message en tts ou sur un nabcast) sur un Nabaztag.

Il s'agit de la version 1 de l'API, elle suit la version 0.

N'hésitez pas à nous contacter si vous souhaitez que nous implémentions d'autres fonctionnalités ([idee@nabaztag.com](mailto:idee@nabaztag.com), sujet : API).

## Comment envoyer un événement ?

---

Il suffit d'appeler l'url <http://www.nabaztag.com/v1/FR/api.jsp> avec les paramètres suivants :

sn	Numéro de série du nabaztag qui va recevoir les événements
idmessage	Numéro du message à envoyer. Ce numéro peut être un numéro de message de la bibliothèque ou d'un fichier MP3 perso que vous avez chargé. On trouve cet identifiant lorsque l'on écoute un morceau sous le titre du morceau.
posright	Position de l'oreille droite entre 0 et 16 (0=oreille horizontale)
posleft	Position de l'oreille gauche
token	Le token est une suite de chiffres donnée lorsque vous activez le Nabaztag récepteur. Cet identifiant limite les risques de spam puisque pour envoyer un message il faut connaître à la fois le numéro de série et ce chiffre.
idapp	C'est l'identifiant de votre application. Il permettra d'authentifier l'émetteur de l'événement. Pour l'instant ce paramètre n'est pas utilisé.
tts	Permet d'envoyer le texte donné en paramètre en synthèse vocale.
chor	Envoi d'une chorégraphie sur le Nabaztag
chortitle	Associe un nom à la chorégraphie
ears=ok	Envoi la position des oreilles
nabcast	Permet d'envoyer un message dans un Nabcast

### Par exemple :

<http://www.nabaztag.com/v1/FR/api.jsp?sn=00039D4022DE&token=112231049046144&posleft=0&posright=0&idmessage=10333&idapp=10>

### Comment envoyer un message en text to speech dans le nabcast qui a l'id 52 avec comme titre *monttsdansmonnabcast* ?

`api.jsp? ... nabcast=52&nabcasttitle=monttsdansmonnabcast&tts=le+nabcast`

### Comment envoyer le message 10333 dans le nabcast qui a l'id 52 avec, comme titre du son dans le nabcast, *monmp3dansmonnabcast* ?

`api.jsp? ... nabcast=52&nabcasttitle=monmp3dansmonnabcast&idmessage=10333`

## Activer la réception des événements sur un Nabaztag

---

1. Rendez vous sur [http://www.nabaztag.com/vl/FR/api\\_prefs.jsp](http://www.nabaztag.com/vl/FR/api_prefs.jsp)
2. Cochez la case « Autoriser les événements extérieurs » puis Valider
3. Le « token » s'affiche. Vous devez l'utiliser pour envoyer des messages depuis votre application. Si vous désactivez puis réactivez la réception d'événements, un nouveau token vous sera communiqué.

## Description du langage de chorégraphie

---

Une chorégraphie est une suite d'actions permettant de piloter l'affichage des leds et des oreilles.

La chorégraphie repose avant tout sur un tempo, qui est réglable. Une fois le tempo réglé, les actions s'enchaînent au rythme du tempo. Chaque action s'exécute à un moment précis, qu'on appellera « l'heure » de l'action : cette heure est exprimée en nombre de temps du tempo depuis le début de la chorégraphie.

Par exemple, avec un tempo d'un dixième de seconde, l'action associée à l'heure '10' se produira une seconde après le début de la chorégraphie. L'action associée à l'heure '25' se produira 2.5 secondes après le début de la chorégraphie...

La syntaxe de la chorégraphie est une suite de mots ou de nombres, séparés par des virgules. Les nombres sont toujours entiers, positifs, exprimés en base 10. Les mots sont exclusivement l'un des deux mots clefs : motor, led.

Cette suite commence par la valeur du tempo, puis se poursuit par la succession des actions de moteur et de led.

Chaque action (moteur ou led) commence par l'heure (exprimée en nombre de temps du tempo depuis le début de la chorégraphie), suivi du nom de la commande (motor ou led), et d'un nombre variable d'arguments.

Les actions doivent être listées dans l'ordre chronologiques (l'heure est donc croissante). Si deux actions ont la même heure, elles seront effectuées simultanément.

### 1. Commande du tempo

---

Le tempo est exprimé en Hz. On prendra donc « 10 » pour un tempo d'un dixième de seconde.

### 2. Commande des oreilles

---

Il s'agit d'une suite de valeurs séparées par des virgules.

La commande est la suivante :

- 2.1 Premier argument, l'heure de l'action (vaut donc « 0 » si c'est la première commande).
- 2.2 Deuxième argument, 'motor', pour bouger une oreille
- 2.3 Troisième argument :  
1 pour commander l'oreille gauche  
0 pour commander l'oreille droite
- 2.4 Quatrième argument, c'est l'angle de rotation de l'oreille. Valeur possible de 0 à 180.
- 2.5 Cinquième argument, inutilisé, mettre à 0
- 2.6 Sixième argument, c'est le sens de rotation des oreilles.  
1 : haut->derrière->bas->devant->haut...  
0 : haut->devant->bas->derrière->haut...

**Exemple** de commande qui fait tourner l'oreille gauche dans le sens contraire des aiguilles d'une montre avec un angle de 20 degrés, à l'heure « 0 ».

→ 0,motor,1,20,0,0

### 3. Commande des leds

---

Il s'agit d'une suite de valeurs séparées par des virgules.

- 3.1 Premier argument, l'heure de l'action (vaut donc « 0 » si c'est la première commande).
- 3.2 Deuxième argument, 'led', pour donner une couleur à la led.
- 3.3 Troisième argument, c'est pour définir la led sur laquelle on veut donner une couleur  
0 : led du bas  
1 : led de gauche du lapin  
2 : led du milieu  
3 : led de droite du lapin  
4 : led du haut
- 3.4 Quatrième, cinquième, sixième argument, c'est la couleur en RGB. Valeur de 0 à 255

**Exemple** de commande qui fait passer la led du milieu en vert, puis après celle de gauche en rouge puis celle du milieu s'éteint.

→ 0,led,2,0,238,0,2,led,1,250,0,0,3,led,2,0,0,0

#### 4. Combinaison de commande des oreilles et des leds.

Pour combiner les trois actions, avec le paramètre initial du tempo

→ 10,0,motor,1,20,0,0, 0,led,2,0,238,0,2,led,1,250,0,0,3,led,2,0,0,0