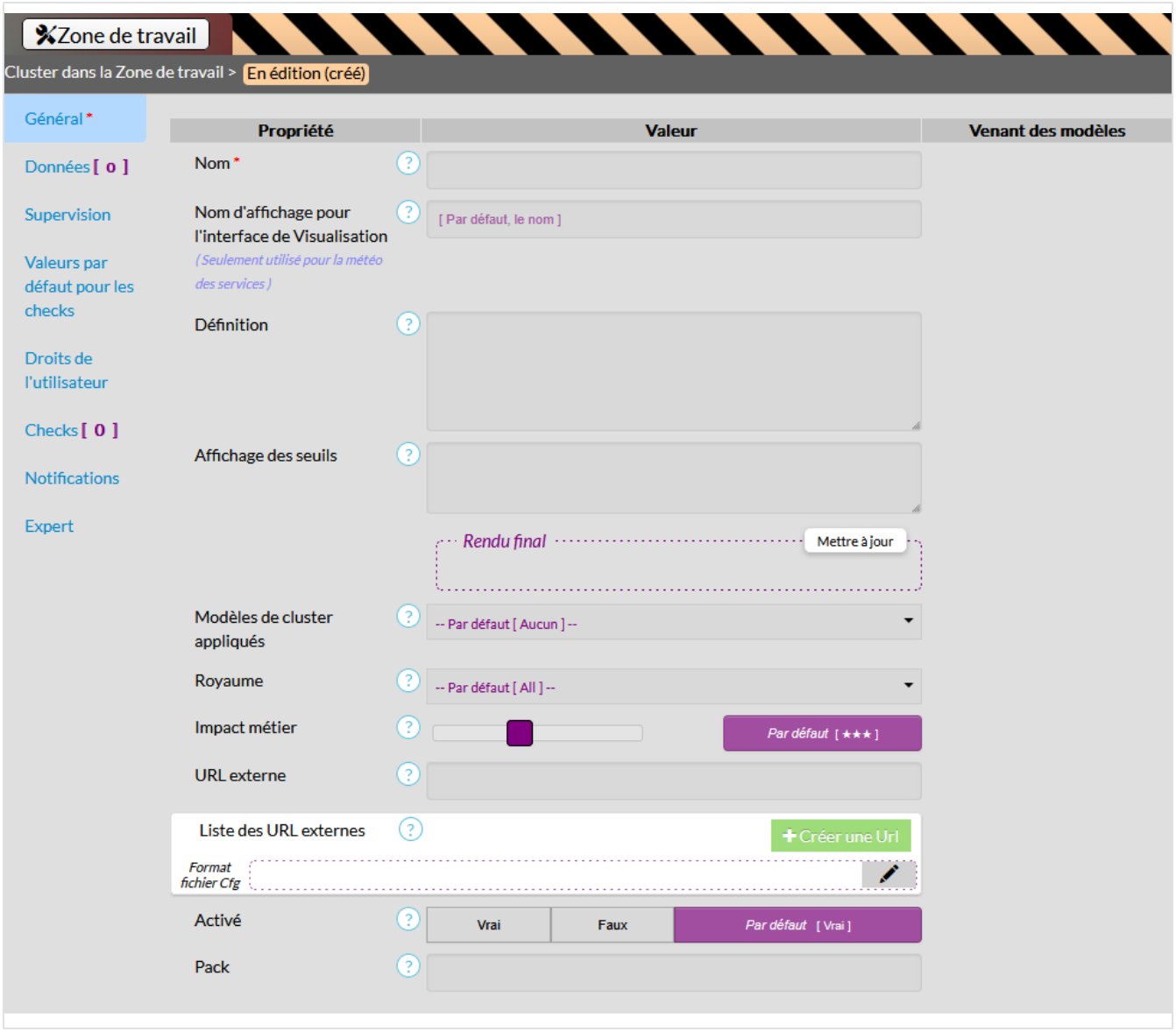
Editer un Cluster

Sommaire

- Onglet : Général
  - Propriétés spécifiques aux modèles de cluster
- Onglet : Données
- Onglet : Supervision
  - Données stockées
    - SLA
- Onglet : Valeurs par défaut pour les checks
- Onglet : Droits de l'utilisateur
- Onglet : Checks
  - Filtre rapide pour trouver un check
    - Filtre actif
    - Agir sur le filtre
    - Replacer une ligne dans son contexte
  - Essayer les checks
  - Les modulations
  - Affichage particulier
    - Checks ayant le même nom sur des modèles différent - Caché
    - Checks ayant le même nom sur le même modèle - Erreur et caché
    - Checks dupliqués avec des clés en double dans la donnée Duplicate Foreach - Erreur et caché
    - Checks dupliqués avec un caractère interdit dans le nom - Erreur
    - Checks dupliqués avec une donnée Duplicate Foreach vide - Avertissement et caché
- Onglet : Notifications
- Onglet : Expert
  - Paramétrage du Flapping
  - Modulations
  - Gestionnaire d'événements

Onglet : Général



Nom

- *Clé d'import* : host_name
- Cette propriété permet de définir le nom utilisé pour identifier le cluster.
- Son contenu est accessible par l'intermédiaire de la variable **$HOSTNAME$** .

Les caractères `~!$%^&*"'|<>?,()=/+ sont interdits dans ce champ.

Un caractère non autorisé provoquera un compteur erreur

Le nom doit être unique (*Cette règle s'applique au nom et au nom pour l'affichage de visualisation des clusters et des hôtes*).

Ce champ ne peut pas rester vide

Nom d'affichage pour l'interface de Visualisation

- *Clé d'import :* visualisation_name
- Cette propriété permet de définir le nom qui sera affiché dans l'interface de Visualisation.
- Son contenu est accessible par l'intermédiaire de la variable **$HOSTVISUALISATIONNAME$.**

Remarque 1 : Le nom d'affichage pour l'Interface de Visualisation vaut par défaut le nom du cluster.

Remarque 2 : Dans un premier temps, cette valeur est uniquement utilisée pour la "Vue météo".

Les caractères `~!$%^&*"'|<>?,()=/+ sont interdits dans ce champ.

⛔ Un caractère non autorisé provoquera un compteur erreur

⛔ Le nom doit être unique (*Cette règle s'applique au nom et au nom pour l'affichage de visualisation des clusters et des hôtes*).

Définition

- *Clé d'import* : bp_rule

  Cette propriété est utilisée pour configurer en une règle logique une agrégation de plusieurs autres états (*hôtes, clusters et checks*).

  - La syntaxe complète de définition d'un cluster se trouve dans la page Les Clusters .

- Voici quelque exemples :

Eléments avec des caractères spéciaux

- Règle par nom d'un cluster :

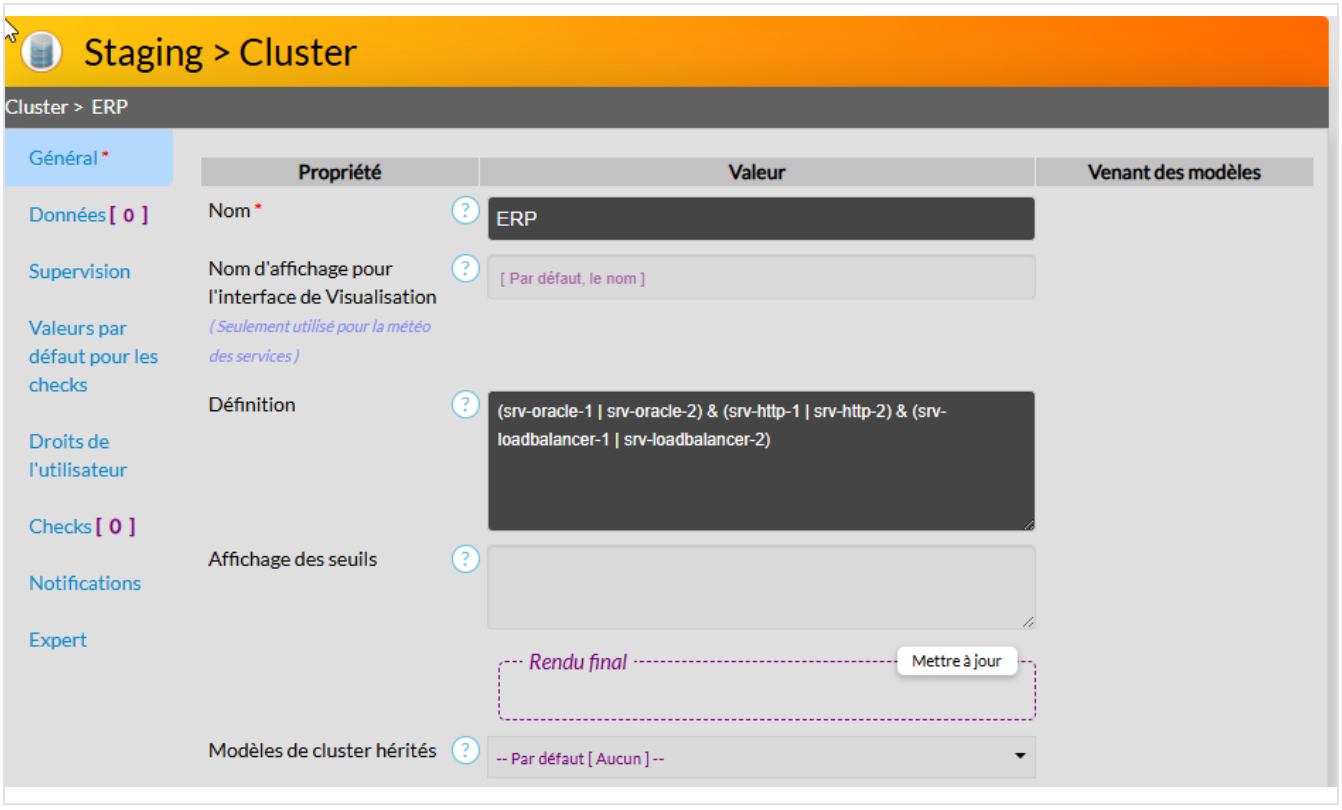
```
"cluster_1" & (!"cluster_2" | "cluster_3")
```

- Règle par nom de check :

```
"cluster_1","check_1" & "cluster_2","check_2"
```

- Règle par expression régulière sur le nom d'un cluster :

```
r:/^cluster_/
```



Affichage des seuils

- *Clé d'import* : thresholds_display

  Cette propriété a pour but de décrire et d'afficher les règles de calcul utilisées pour déterminer le Statut du Cluster (*OK, CRITIQUE, ...*).
  Le contenu est libre, donc il est possible de définir n'importe quel texte.

  À savoir :

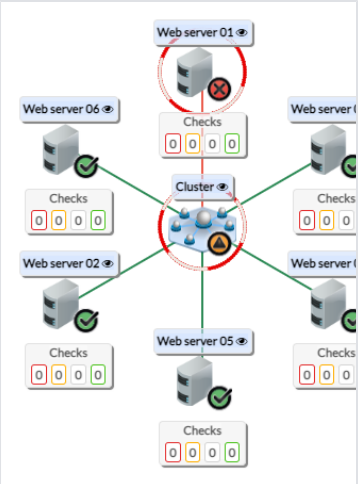
  - Les données seront interprétées dans le "Rendu final" pour avoir la dynamique de configuration des seuils (*voir la page* Les Variables (Remplacement dynamique de contenu - Anciennement les Macros)).
  - Le HTML sera interprété.
  - Le rendu sera calculé au chargement de la page, mais il nécessitera l'appui sur le bouton "Mettre à jour" lors d'une édition de l'élément.
  - Dès qu'une propriété ou une donnée influant sur le rendu final sera modifiée, ce dernier sera effacé et devra être mis à jour manuellement.

Exemple d'utilisation :

Si la règle de définition est :

2Critical->Critical|50%Warning->Critical|1Critical->Warning|30%Warning->Warning|default->OK of: r:/^Web server/

On peut expliquer plus facilement les règles en utilisant l'affichage des seuils de la manière suivante.



⚠ Un Avertissement sera remonté lors de la saisie de la donnée si celle-ci n'existe pas sur l'élément ou dans son héritage.

EXEMPLE : La donnée MEMORY_WARN est manquante

Modèles de cluster appliqués

- *Clé d'import* : use

  Les modèles de cluster qui sont attachés au cluster pour hériter de leurs propriétés et de leurs données.

  L'ordre des modèles est important.

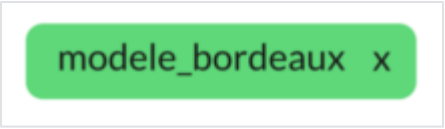
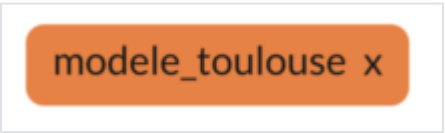
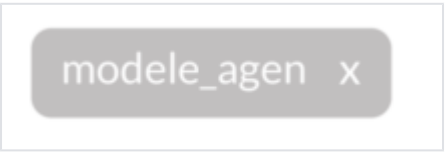
ⓘ **Les caractères `~!$%^&*"'|<>?,()=/+ sont interdits dans ce champ.**

⛔ Un caractère non autorisé provoquera un compteur d'erreur et il sera impossible d'enregistrer le cluster.

ⓘ Pour un administrateur de SI, certains modèles ne sont pas **visible** et ne seront pas disponibles dans la liste des choix.

Il existe un code couleur, correspondant aux différents états des modèles de clusters.

Code couleur	explication

	Un tag de couleur **vert** signifie que le modèle est **valide et activé** .
	Un tag de couleur **orange** signifie que le modèle est **désactivé**.
	Un tag de couleur **jaune** signifie que le modèle est en statut **Nouveau**
	Un tag de couleur **grise** signifie que le modèle **n'est pas pris en compte**, car déjà présent dans l'héritage des autres modèles.
	Un tag de couleur **rouge** signifie que le modèle est en **erreur**. Cela se produit lorsque le modèle est membre d'une boucle infinie de modèle. Par exemple : le **modele_01** utilise le **modele_02** et que le **modele_02** utilise le **modele_01**
	Un tag de couleur **blanche avec des bordures orange** signifie que le modèle ajouté **n'existe pas ou n'existe plus**
	Un tag de couleur **jaune avec des bordures orange** signifie que le modèle est en statut "**Nouveau**", mais que ce dernier est **désactivé.**
	Un tag de couleur **rouge avec une "cartouche" orange** signifie que le modèle est en **erreur** (*par exemple : une boucle infinie de modèle*) et que le modèle est **désactivé.**

Royaume

- *Clé d'import* : realm

  Cette propriété permet de définir le royaume où le cluster se trouvera.
  En plaçant un cluster dans un royaume particulier, un des Schedulers de ce royaume gérera ce cluster.

Pour rappel, un royaume doit obligatoirement avoir au moins un Scheduler défini dans sa configuration. Dans le cas contraire, le royaume apparaitra désactivé dans la liste avec une infobulle indiquant le problème.

Impact métier

- *Clé d'import :* business_impact

  Cette propriété permet de définir l'importance que l'on donne à un cluster pour le business (*valeur numérique entre 0 et 5 dans les fichiers d'imports, mais affichée dans l'interface avec des étoiles ou des dollars suivant le niveau*) :
  - du moins important (0 => *)
  - (1 => **)
  - (2 => ***)
  - (3 => $$$$)
  - (4 => $$$$$)
  - au plus important (5 => $$$$$$)

  La valeur par défaut est 2 (***).

  Il est possible de changer l'affichage de la valeur de l'Impact métier (*voir la page* Nom d'affichage des propriétés des éléments (Répertoire elements_default_properties)).

URL externe

- *Clé d'import* : notes_url

  Cette propriété est utilisée pour définir une URL qui peut être utilisée pour fournir plus d'information sur ce cluster.
  N'importe quelle URL valide peut être mise, commençant par *http:// ou https://*

  Cette URL sera disponible dans l'interface de visualisation dans le volet détail présentant le cluster.
  Cela peut être utile pour faire une description écrite détaillée du cluster, une procédure de reprise sur panne... qui sera visible pour les autres membres de l'équipe.

Une URL qui ne commence pas par *http:// ou https://* provoquera un compteur d'erreur et il sera impossible d'enregistrer le cluster

Les caractères "' et les espaces sont interdits dans ce champ .

Un caractère non autorisé provoquera un compteur d'erreur et il sera impossible d'enregistrer le cluster

Il est possible de changer le nom de cette propriété (*voir la page Nom d'affichage des propriétés des éléments (Répertoire elements_default_properties)*).

Liste des URL externes

- *Clé d'import* : notes_multi_url

  Cette propriété est utilisée pour définir plusieurs URL qui peuvent être utilisées pour fournir plus d'informations sur ce cluster.

(1) Nom de l'URL (*label pour l'interface de visualisation*), le nom à afficher est optionnel.

Les caractères <>& "'/ sont interdits dans ce champ .

Un caractère non autorisé provoquera un compteur d'erreur et il sera impossible d'enregistrer le cluster

(2) Adresse de l'URL, N'importe quelle URL valide peut être utilisée, commençant par **http://** or **https://**

Les caractères "' et les espaces sont interdits dans ce champ .

Un caractère non autorisé provoquera un compteur d'erreur et il sera impossible d'enregistrer le cluster

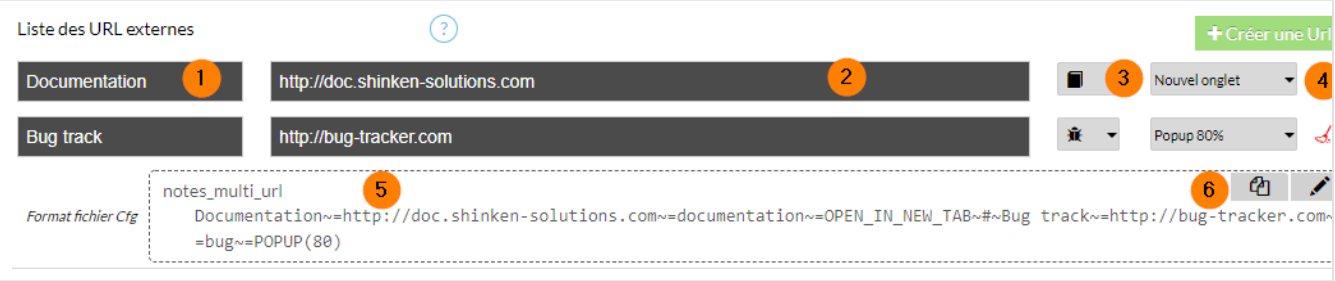
(3) Icône pour l'UI de visualisation
(4) Type d'affichage dans l'interface de Visualisation

- Nouvel onglet,
- Pop-up de différente taille (*30%, 40%, 50%, 60%, 70%, 80% de la taille de l'écran*)
- Caché

(5) Valeur à copier-coller dans un fichier .cfg pour retrouver cette propriété dans l'état

(6) Passer la souris en survol du champ format fichier .cfg pour faire apparaître les boutons de copie du format (*gauche*) et d'édition du format (*droite*).

- Le bouton d'édition permet d'éditer la propriété URL externe soit en collant un format précédemment copié ou bien en écrivant le format.



Une URL externe peut contenir le mot clé ##USER## qui sera remplacé par le nom de l'utilisateur courant dans l'interface de Visualisation

Important

Cette propriété, dans l'URL, accepte le remplacement dynamique de contenu des données (*macro*) de ce cluster.

- Pour intégrer une donnée ou une propriété dans l'URL, il suffit de l'ajouter comme pour les commandes.
  - Exemple : pour la donnée DATA1 : **$_HOSTDATA1$**
  - (*voir la page* Les Variables (Remplacement dynamique de contenu - Anciennement les Macros)Les Variables (Remplacement dynamique de contenu - Anciennement les Macros))

Les données (*macros*) qui peuvent être utilisées sont les suivantes :

- Données globales
- Données accrochées au cluster présentes dans l'onglet Données
- et les propriétés (*données statiques*) suivantes :
  - HOSTUUID
  - HOSTNAME
  - HOSTDISPLAYNAME
  - HOSTADDRESS
  - HOSTREALM

Ces URL seront disponibles dans l'Interface de Visualisation.

Cela peut être utilisé pour faire une description écrite détaillée du cluster, une procédure de reprise sur panne ou toute autre information qui sera visible pour les autres membres de l'équipe.

Activé

- *Clé d'import :* enabled

  Permettre que cet élément soit pris en compte lors du passage de **Staging** en **Production**.

Pack

- *Clé d'import* : pack

  Permet de donner un nom commun à un ensemble d'éléments de Shinken . Il sera ainsi possible de filtrer par ce nom commun dans les listes des éléments pour retrouver l'ensemble des éléments .

  Cette propriété n'est pas héritable.

Les caractères < > & " ' / sont interdits dans ce champ.

Un caractère non autorisé provoquera un compteur d'erreur et il sera impossible d'enregistrer.

Propriétés spécifiques aux modèles de cluster

Modèle applicable par les administrateurs de SI ?

- *Clé d'import* : template_can_be_applied_by_si_admin

  Cette propriété permet d'autoriser les administrateurs SI à appliquer le modèle sur un cluster.

  Un modèle non applicable par les administrateurs SI reste visible sur le cluster s'il a déjà été appliqué par un administrateur Shinken.

Onglet : Données

Cet onglet définit des données qui pourront être utilisées en tant que Variable utilisable pour le remplacement dynamique de contenue (*voir la page* Les Variables (Remplacement dynamique de contenu - Anciennement les Macros)), notamment lors de l'utilisation des commandes des checks attachés à ce cluster (*voir la page* Les commandes).

Utiliser un nom de donnée protégée (*comme PASSWORD*) pour une donnée, empêche de modifier ce nom par la suite.

- Il faudra la supprimer puis la recréer pour la renommer.

Cette modification est interdite afin d'éviter qu'une donnée protégée devienne visible, en changeant son nom.

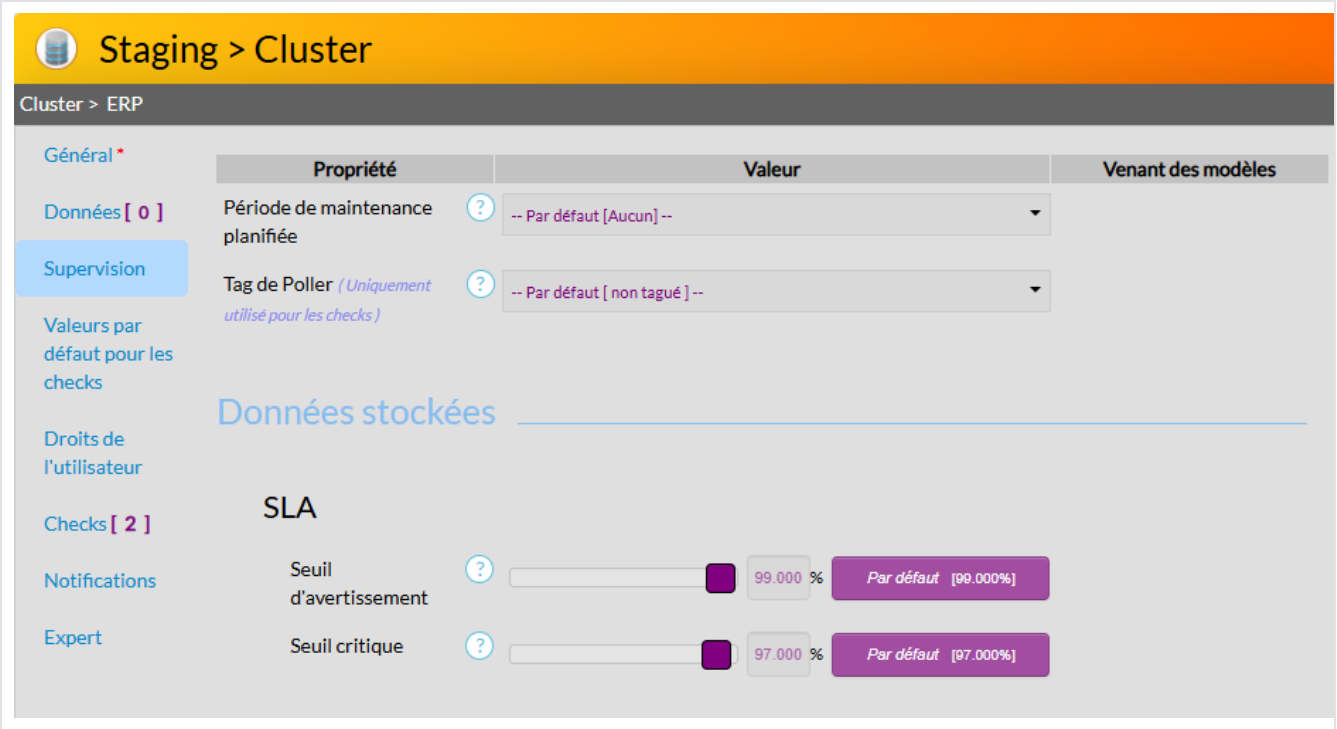
Seuls les caractères alphanumériques, le tiret (-) et le souligné (_) sont autorisés dans le nom d'une donnée

Un caractère non autorisé provoquera une erreur et il sera impossible d'enregistrer le cluster si une valeur est également associée à la donnée

La valeur de la donnée pouvant être longue, il est possible d'agrandir le champ de la valeur afin d'améliorer la lisibilité du champ.

Pour agrandir le champ de la valeur, il faut maintenir le clic sur l'icône située en bas à droite du champ et réajuster verticalement.

Onglet : Supervision



Période de maintenance planifiée

- *Clé d'import :* maintenance_period

  Cette propriété permet de définir une période de temps durant laquelle le cluster sera en maintenance, et en sortira dès que la période de temps sera fini.
  En maintenance, les checks du cluster seront toujours exécutés (*donc il n'y aura pas de manques dans les métriques ou l'historique du cluster*), mais les notifications et les commandes du gestionnaire d'évènements ne seront pas lancées.

Tag de Poller

- *Clé d'import :* poller_tag

  Cette propriété permet de définir le poller_tag du Cluster.
  Elle n'a pas d'effet sur le cluster en lui-même, mais si une valeur est définie, les checks du cluster pourront hériter de ce tag, suivant leur configuration.

Données stockées

SLA

Seuil d'avertissement

- *Clé d'import :* sla_warning_threshold

  Cette propriété permet de définir un seuil d'avertissement pour le module SLA.
  La valeur doit être un nombre décimal entre 0 (*inclus*) et 100 (*inclus*) avec jusqu'à 3 chiffres après la virgule.

Seuil critique

- *Clé d'import :* sla_critical_threshold

  Cette propriété permet de définir un seuil critique pour le module SLA.
  La valeur doit être un nombre décimal entre 0 (*inclus*) et 100 (*inclus*) avec jusqu'à 3 chiffres après la virgule.

Onglet : Valeurs par défaut pour les checks

Période de vérification

- *Clé d'import :* check_period

  Cette propriété permet de choisir la période de supervision pendant laquelle sont faites les vérifications de l'état du cluster par Shinken.

Temps maximum d'exécution d'un check (secondes)

- *Clé d'import :* check_running_timeout

  Cette propriété permet de définir la valeur en secondes du temps maximum permis pour le lancement d'une commande de supervision. Passé ce délai, la commande sera terminée. Si le check ou la commande définit également ce paramètre, leur valeur sera prioritaire par rapport à celle du cluster

  Aucune valeur ou une valeur a -1, indique qu'il n'y a pas de temps maximum défini.

  Si ni le cluster, ni le check, ni la commande ne définissent ce paramètre, alors la valeur définie globalement pour Shinken (*via le fichier shinken.cfg et le paramètre check_running_timeout*) sera utilisée à la place.

La limite est fixée à 157852800 **(** *soit 5 ans* **).**

Un caractère non autorisé provoquera un compteur d'erreur

Seuil d'alerte de l'utilisation CPU (secondes)

- *Clé d'import :* warning_threshold_cpu_usage

  Cette propriété permet de définir la durée d'exécution (*en seconde*) d'une commande avant d'apparaître en Warning dans le check 'Scheduler - Performance'
  La valeur sera récupérée dans l'ordre, par la commande, par l'élément Check, par l'élément Cluster, ou par la valeur par défaut (*via le fichier shinken.cfg et le paramètre warning_threshold_cpu_usage*)

La limite est fixée à 157852800 **(** *soit 5 ans* **).**

Un caractère non autorisé provoquera un compteur d'erreur

Onglet : Droits de l'utilisateur

Il est possible pour chaque cluster de définir qui peut voir, éditer, ou encore recevoir les notifications. Ces propriétés gèrent l' Héritage additif (le +).

Lors de la création d'un cluster, l'héritage additif est activé par défaut et l'utilisateur (*Administrateur Shinken ou Administrateur de SI*) créant le cluster est automatiquement ajouté aux droits utilisateurs suivants :

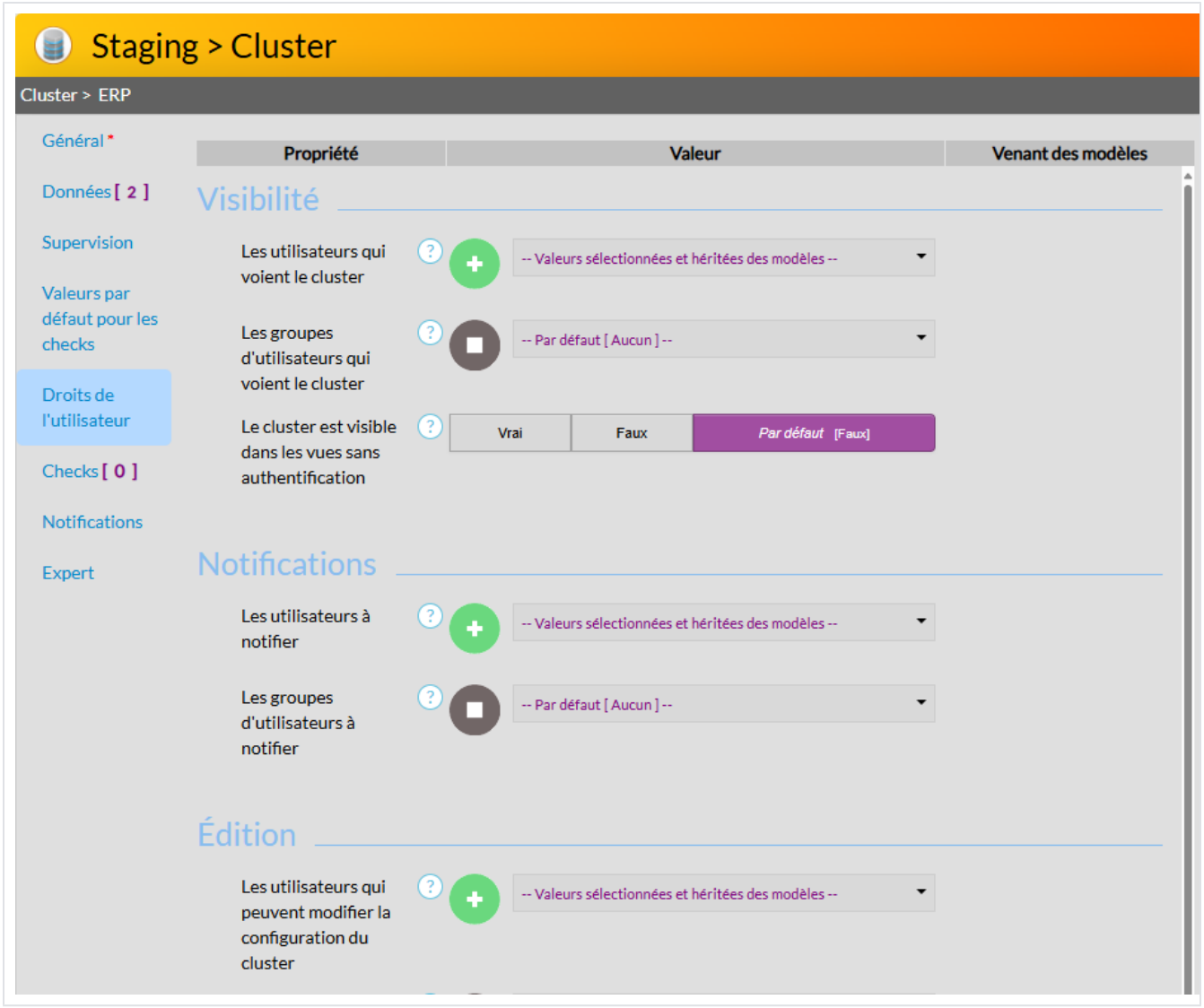
- Les utilisateurs qui voient le cluster,
- Les utilisateurs à notifier,
- Les utilisateurs qui peuvent modifier la configuration du cluster.

Ces propriétés se synchronisent les unes avec les autres, pour éviter d'avoir une incohérence.

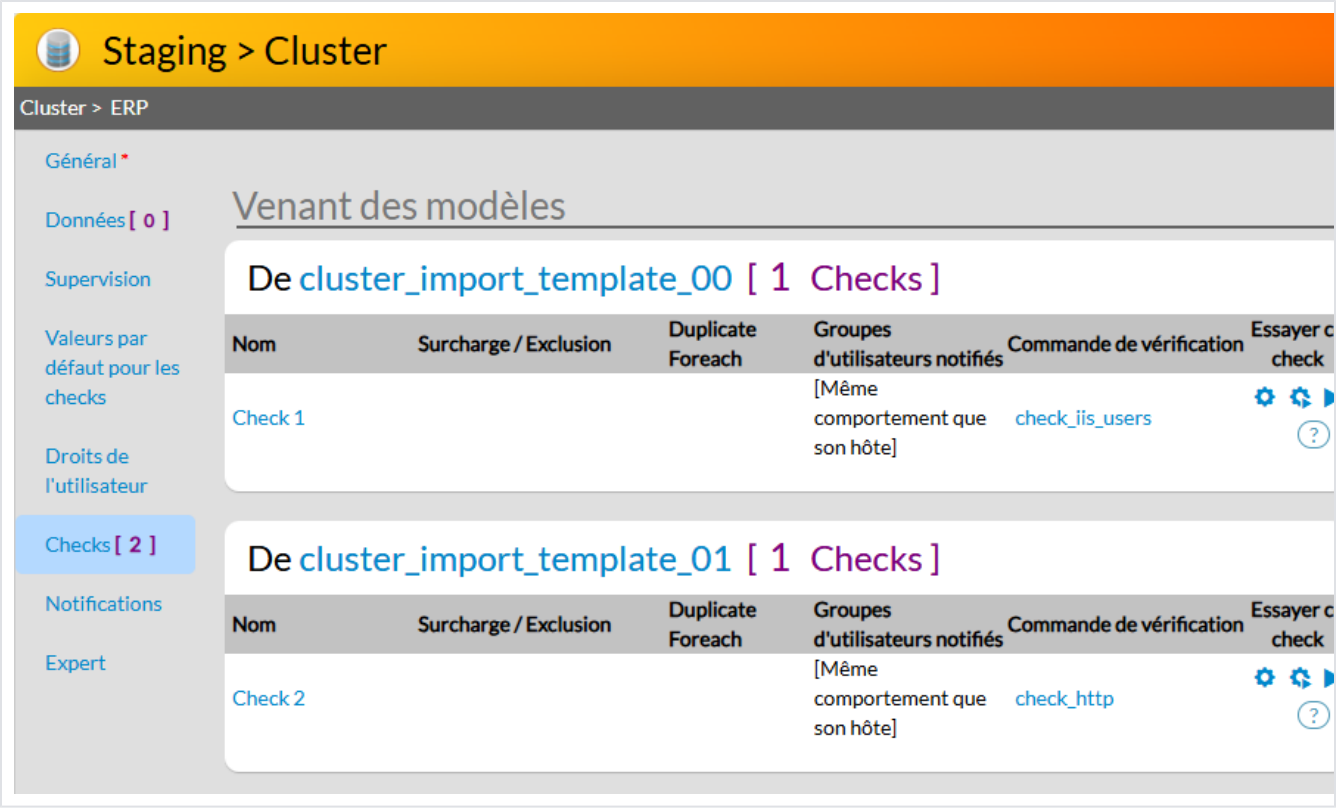
Par exemple : un utilisateur qui aurait le droit d'éditer le cluster sans le droit de le voir.

Les accès sont séparés en 3 sections :

- L'accès en visualisation du cluster
  - Ce paramètre permet d'autoriser certains utilisateurs ou groupes d'utilisateurs à voir le cluster concerné (*dans l'interface de Visualisation*).
  - Il est possible de lister des utilisateurs individuellement, ainsi que des groupes d'utilisateurs.
  - Il est aussi possible, dans cette section, de choisir si le cluster peut être affiché sur les vues ne nécessitant pas une authentification à Shinken.
- L'accès aux notifications d'un cluster
  - Cette section permet de lister les utilisateurs ou les groupes d'utilisateurs qui recevront les notifications du cluster.
- L'accès en édition sans authentification peut être modifiée
  - Ce paramètre permet de lister les utilisateurs et les groupes pouvant éditer le cluster.



Onglet : Checks



Cet onglet détaille la liste des checks qui seront appliqués au cluster, et leur provenance (*venant de quel modèle de cluster, ou directement appliqué au cluster*) :

- Tout d'abord, les checks accrochés directement sur le cluster sont listés.
- Ensuite sont listés les modèles dans l'ordre de leur apparition dans la liste des modèles hérités.
  - Et pour chacun leurs checks.
  - Seront aussi listés les checks "Duplicate Foreach" (*voir la page* Dupliquer des checks en fonction d'une liste de valeurs présentes dans la Donnée d'un hôte (duplicate_foreach)).
    Ici, dans l'image, un seul check utilise des Duplicate Foreach (*la colonne Duplicate Foreach a une valeur pour ce check*).

Pour chaque check, il est possible d'essayer le check depuis sa configuration actuelle afin de vérifier son résultat.

Filtre rapide pour trouver un check

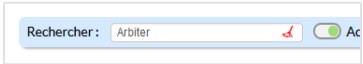
L'onglet check peut comporter de nombreux check.

Un champ texte permet de faire une recherche sur le nom d'un check (*Les autres champs seront pris en compte dans un prochain temps*).

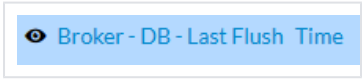
- Le filtre prend en compte uniquement le nom du check.

Filtre actif

Lorsque le filtre est actif (*le champ a été renseigné*) :



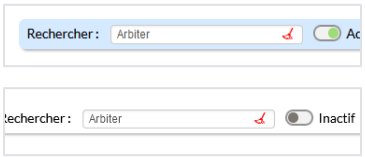
- Un bouton Actif/Inactif apparait permettant d'enlever le filtre sans perdre la valeur renseigné
- le balai permet de remettre à zéro la valeur du filtre.
- Les lignes qui correspondent aux champs seront préfixées d'une icône . Un clic sur cette icône permet d'afficher à nouveau la liste complète en focalisant sur le check en question (*perm et de vérifier le check au milieu de ses checks voisin*).



- Les lignes qui correspondent aux champs seront préfixées d'une icône . Un clic sur cette icône permet d'afficher à nouveau la liste complète en focalisant sur le check en question (*perm et de vérifier le check au milieu de ses checks voisin*).

Agir sur le filtre

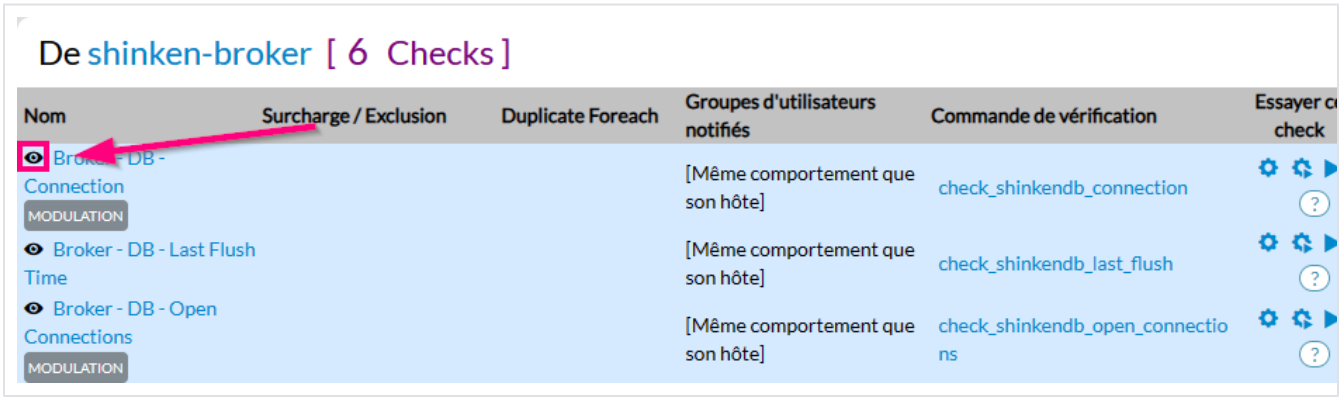
Cliquer sur le bouton permet d'activer / désactiver le filtre.



- Cela permet de garder le texte de filtre saisi, et de basculer (*rapidement et simplement*) des éléments correspondants a la liste complète (*et inversement*).

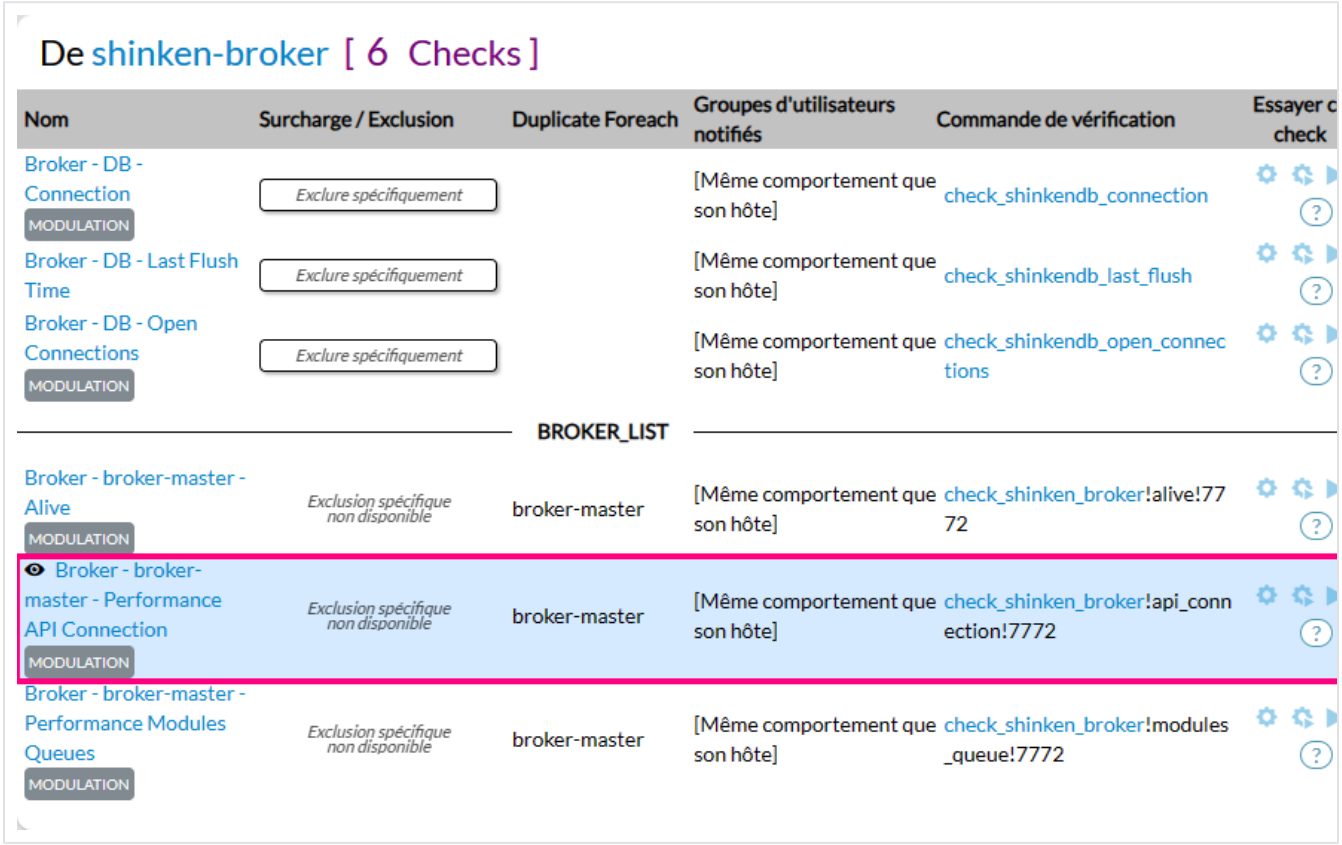
Replacer une ligne dans son contexte

Le clic sur l'icône , permet de replacer ce check au milieu des autres check. (*les checks qui ne correspondent pas aux filtres réapparaissent avec un fond blanc*)



La ligne qui a été cliqué, sera isolée et le fond bleu permettra de la repérer directement.

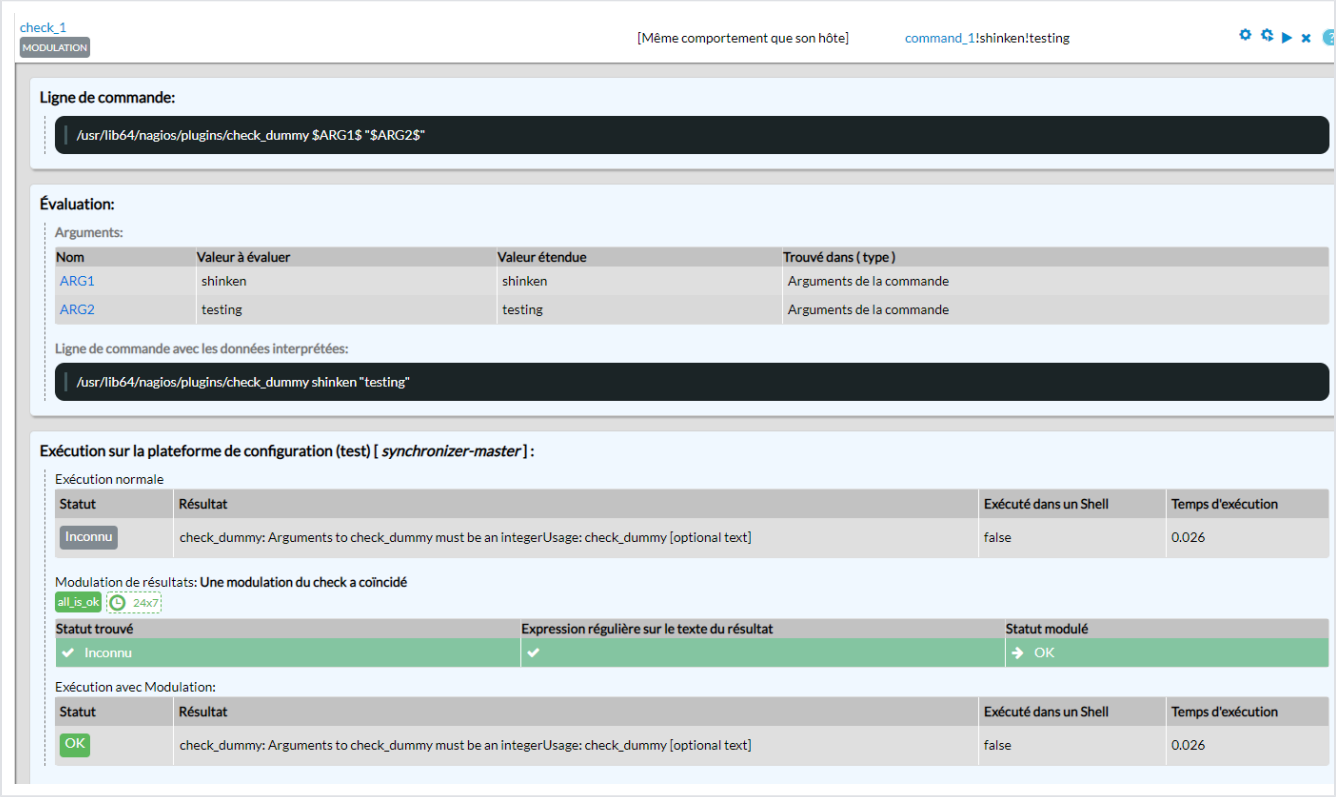
Si le filtre retournait plusieurs lignes, elles garderont toutes leur fond bleu, mais seule celle cliqué gardera l'icône œil .



Essayer les checks

Les boutons dans la colonne [Essayer ce check] permettent d'évaluer ou d'essayer directement l'exécution d'un check, avec la résolution de ses données.

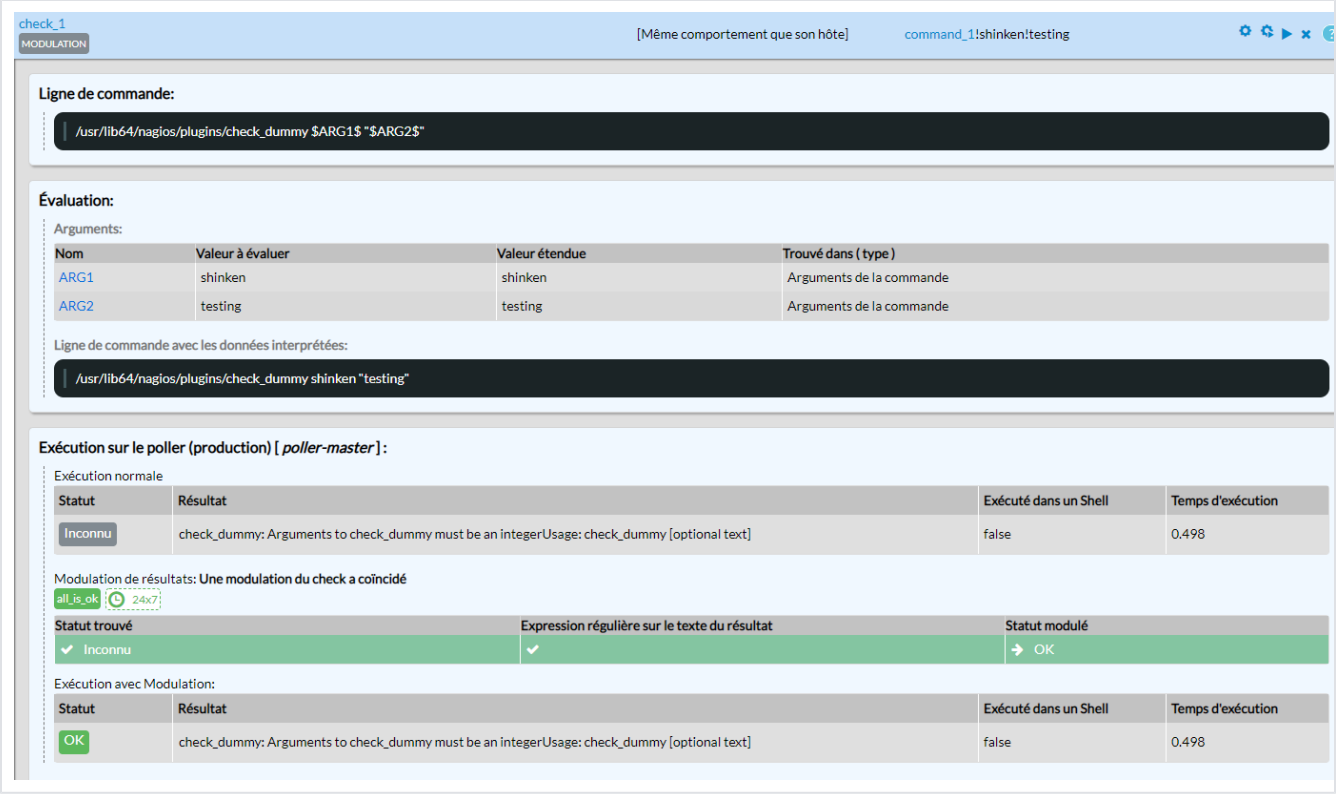
- Le premier bouton (*roue crantée*) permet de simuler la résolution des arguments, sans l'exécution du check. Cette évaluation permet d'afficher la commande avec les données interprétées.
- Le second bouton (*roue crantée + icône play*) permet d'évaluer et de simuler son exécution depuis la plateforme de configuration (*Synchronizer*). Cette exécution n'utilisera pas les Pollers. Il est possible d'utiliser ce bouton pour tester la commande sans affecter les serveurs Pollers en production.
  Le tableau récapitulatif présente les données récupérées et le résultat de la commande en prenant en compte les éventuelles modulations.



- Le troisième bouton (*icône play*) permet d'évaluer et de simuler son exécution directement sur les Pollers, comme lors de l'exécution normale sur l'architecture Shinken. Il est possible d'utiliser ce bouton pour tester la commande sur un environnement de production.
  Le tableau récapitulatif présente les données récupérées, et le résultat de la commande en prenant en compte les éventuelles modulations.

> Si le check utilise des tags de Poller, l'exécution ne peut avoir lieu que si l'un des Pollers définis dans l'architecture dispose d'un tag de Poller correspondant à celui du check essayé.
> Si le check est dans un sous royaume, il faut que l'un des Pollers du sous-royaume ait le tag recherché (*voir la page* Le Poller).

- Le dernier bouton (*croix*) permet de cacher le bloc d'essai du check.



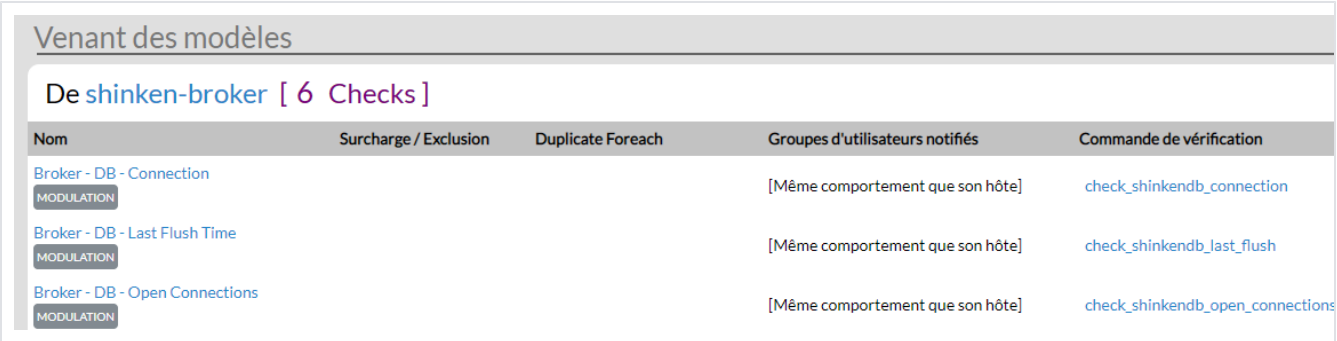
Si une erreur survient pendant l'essai du check, celle-ci sera affichée à la place des résultats.

Les checks ayant une commande de supervision **bp_rule** ne pourront pas effectuer d'évaluation ou d'essai.

Les modulations

Lorsqu'une ou plusieurs modulations de résultat ou de données sont associées au cluster ou au check, un tag **MODULATION** est affiché à côté du nom du check sur lequel vont être appliquées la ou les modulations.

- Si le tag est affiché avec la couleur orange, cela signifie qu'au moins une modulation est **désactivée** ou en statut **Nouveau**.
- Dans ce cas, cette ou ces modulations seront ignorées et n'affecteront pas l'exécution du check.

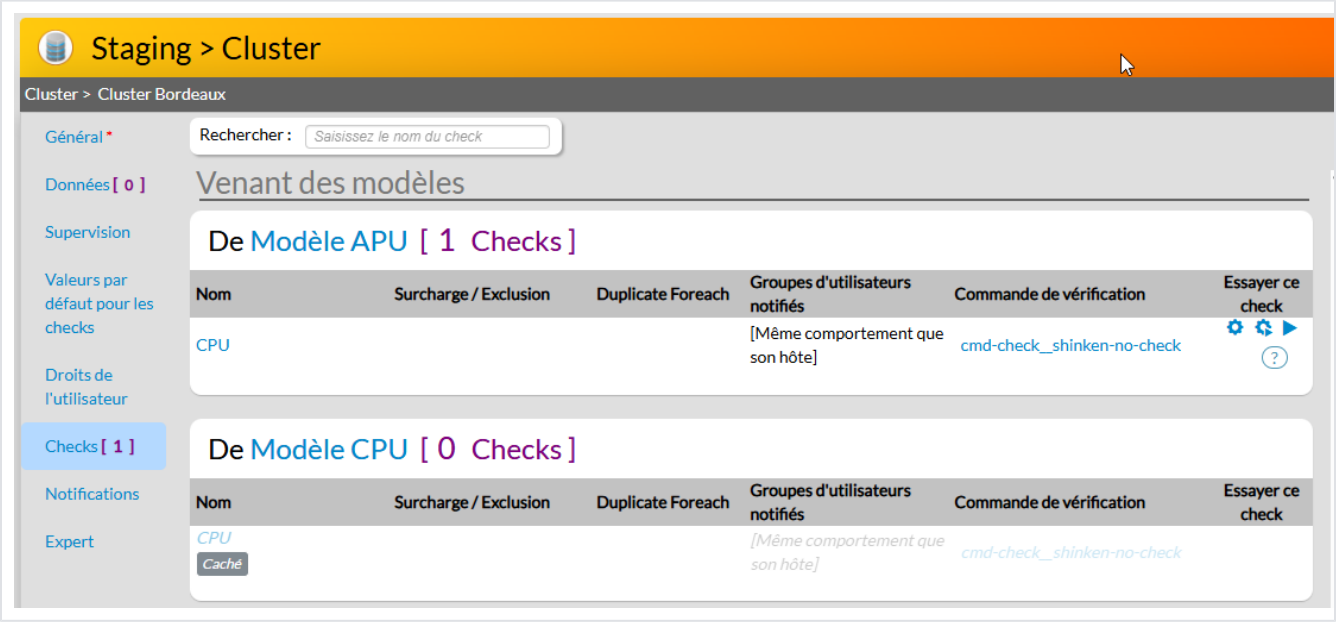


Affichage particulier

Checks ayant le même nom sur des modèles différent - Caché

Le nom des checks n'étant pas unique, il est possible, selon les modèles utilisés, d'avoir deux checks portant le même nom sur un même élément.

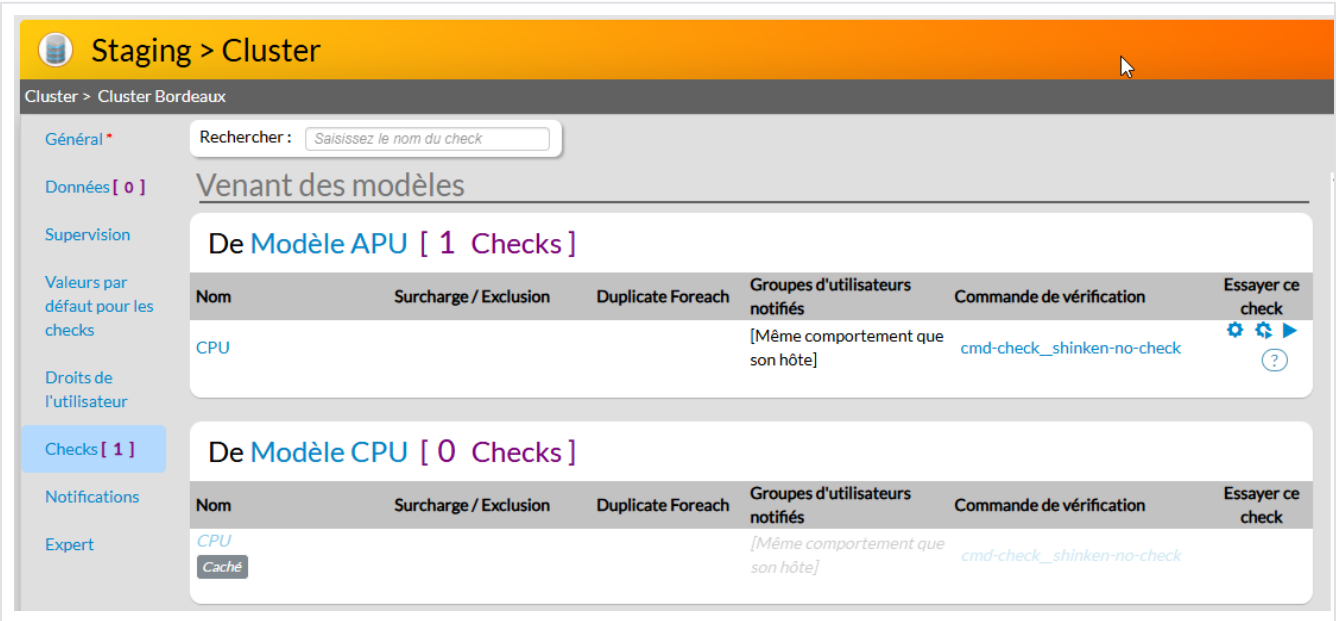
- Dans ce cas, l'ordre d'attachement des modèles détermine quel check sera actif, et lesquels autres seront cachés



Checks ayant le même nom sur le même modèle - Erreur et caché

Il est possible d'avoir deux checks avec le même nom sur un même modèle (*par exemple avec des règles complexes sur le check, voir la page* Editer un check appliqué à un modèle d'hôte) :

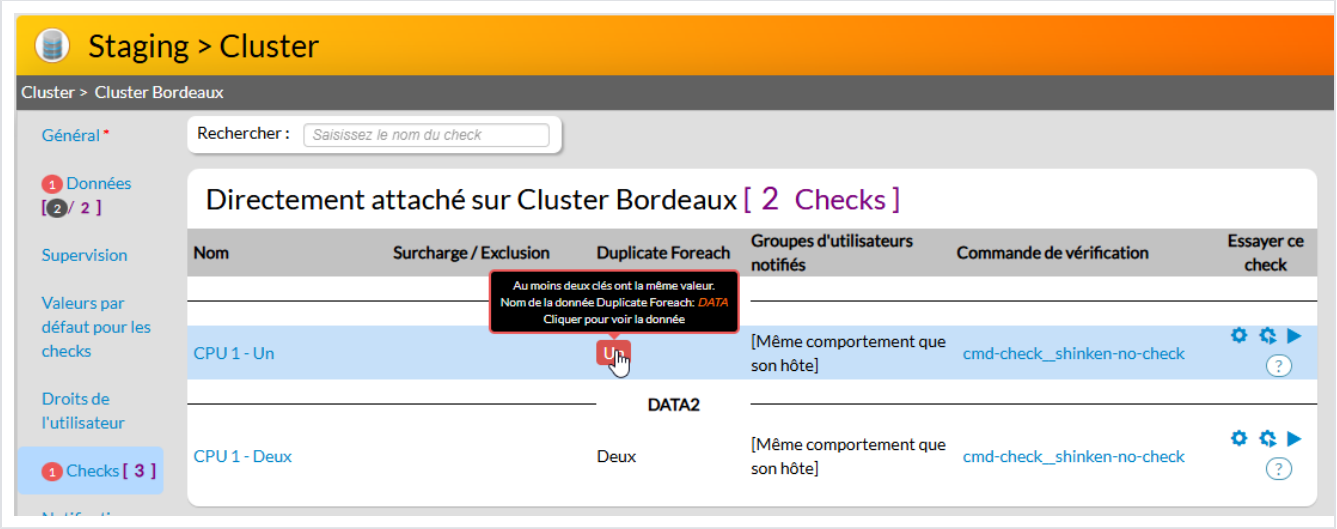
- Dans ce cas particulier, Shinken n'est pas capable de choisir et les deux checks passent en erreur.



Checks dupliqués avec des clés en double dans la donnée Duplicate Foreach - Erreur et caché

Les valeurs dans les données "Duplicate ForEach" doivent toutes être différentes entre elles (*Un, Deux, Trois*), en cas de doublons (*Un, Deux, Un*) une erreur sera présente.

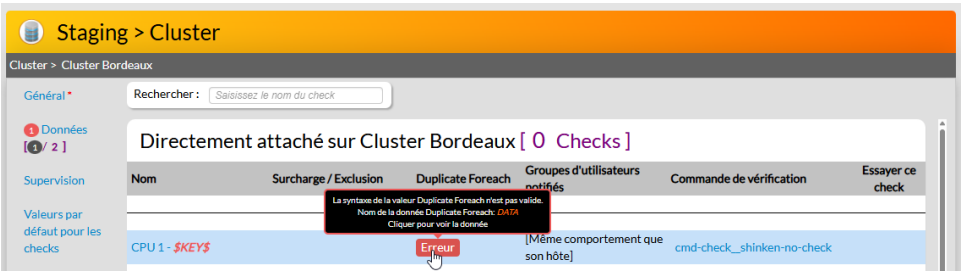
- Dans l'exemple suivant, la donnée DATA contient : Un, Deux, Un. la liste des checks affichera donc une erreur indiquant la valeur en erreur Un .
- Cliquer dessus, bascule directement vers la donnée pour la corriger.



Checks dupliqués avec un caractère interdit dans le nom - Erreur

Les valeurs dans les données "Duplicate ForEach" seront présents dans le nom du check, il faut donc que ces valeurs respecte les mêmes contraintes que celle que possède le nom d'un check.

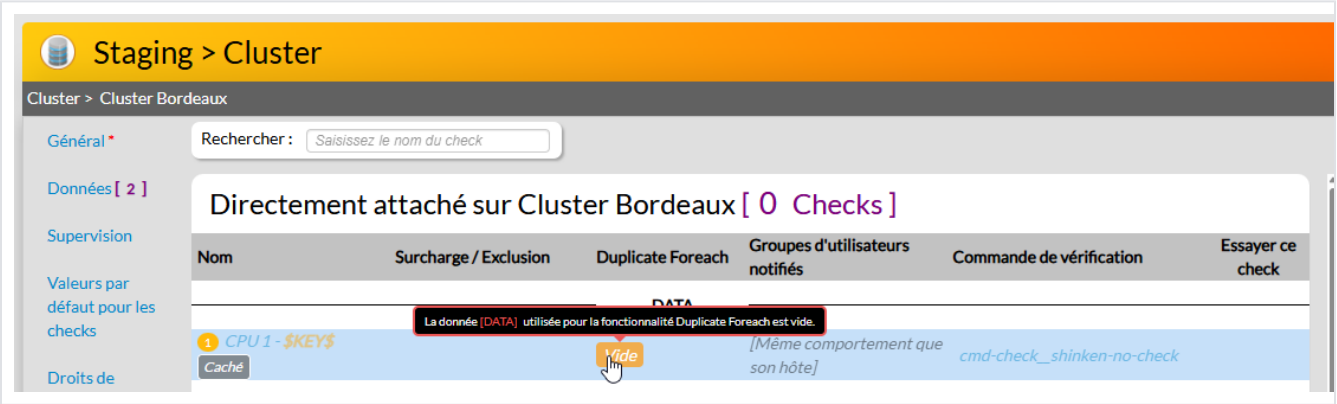
- Si une valeur contient un caractère interdit, alors le check apparaitra en erreur,
- et ne pourra pas être sauvegardé.
- le clic sur Erreur conduit à la valeur problématique.



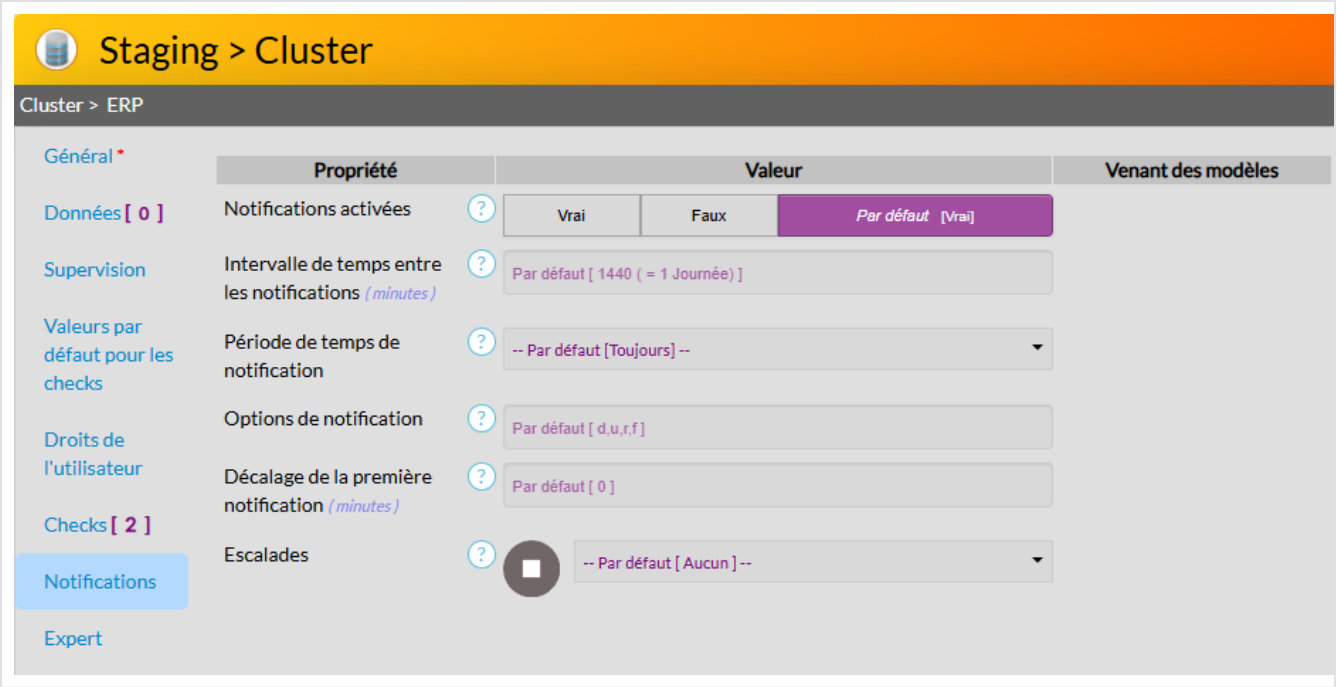
Checks dupliqués avec une donnée Duplicate Foreach vide - Avertissement et caché

Si la donnée "Duplicate ForEach" est vide alors, un check (*non fonctionnel*) utilisant cette donnée est néanmoins créé :

- Il apparaitra en avertissement.
- Le clic sur Vide conduit à la donnée problématique.



Onglet : Notifications



- **Notifications activées**
    - *Clé d'import* : notifications_enabled

      Cette propriété permet de définir si Shinken va envoyer des notifications aux utilisateurs listés comme destinataires (*Utilisateurs ou Groupes d'utilisateurs*) sur ce cluster.

      Les notifications sont envoyées (*voir la page* Logique de notification) :
        - Premièrement, lorsqu'un statut non-OK est confirmé ;
        - ensuite après chaque **intervalle de notification**, exprimé en minutes ;
        - quand le statut du cluster redevient OK ;

      Les valeurs possibles sont :
        - **Vrai** : Shinken va envoyer les notifications.
        - **Faux** : Shinken n'enverra pas les notifications.

Intervalle de temps entre les notifications (minutes)

- *Clé d'import :* notification_interval

  Cette propriété permet de définir la période de temps durant laquelle les notifications sont autorisées.
  En dehors de cette période, aucune notification ne sera envoyée.

  Par défaut, il n'y a pas de période de temps, et donc les notifications ne seront jamais bloquées.

  La limite est fixée à 2630880 **(** *soit 5 ans* **).**
  Une option non existante provoquera un compteur erreur

Période de temps de notification

- *Clé d'import :* notification_period

  Cette propriété permet de définir la période de temps durant laquelle les notifications sont autorisées.
  En dehors de cette période, aucune notification ne sera envoyée.

  S'il n'y a pas de période de temps définit (*comportement par défaut*) l'envoi de notification n'est pas impacté par ce paramètre.

Options de notification

- *Clé d'import :* notification_options

Cette propriété permet de définir quels états et contextes du cluster peuvent envoyer des notifications.

La valeur est une liste d'options séparées par des virgules :
- c : envoie une notification sur un état CRITIQUE;
- w : envoie une notification sur un état AVERTISSEMENT;
- u : envoie une notification sur un état INCONNU;
- d : envoie une notification sur un état CRITIQUE et AVERTISSEMENT;
- r : envoie une notification lors d'un rétablissement (*signifie que l'état redevient OK*);
- f : envoie une notification lors de l'entrée ou la sortie en FLAPPING;
- s : envoie une notification lors de l'entrée ou la sortie en DOWNTIME;
- n : désactive toutes les notifications;

Par défaut, tous les états permettent d'envoyer une notification.

Une option non existante provoquera un compteur erreur

Décalage de la première notification (minutes)

- *Clé d'import :* first_notification_delay

Cette propriété permet de définir combien de minutes Shinken doit attendre avant d'envoyer la première notification.
Ce temps additionnel peut être utilisé pour limiter une avalanche de notifications ; en effet, les clusters n'ont pas de gestion HARD/SOFT et leur état est donc susceptible de changer plus fréquemment.
Ce temps additionnel peut être mis à profit par les utilisateurs pour prendre en compte le cluster depuis l'interface de visualisation avant que la notification ne soit envoyée.
Par défaut la valeur est 0, ceci signifie que la première notification sera envoyée sans attendre.

La limite est fixée à 2630880 **(** *soit 5 ans* **).**
Une valeur non numérique provoquera un compteur erreur

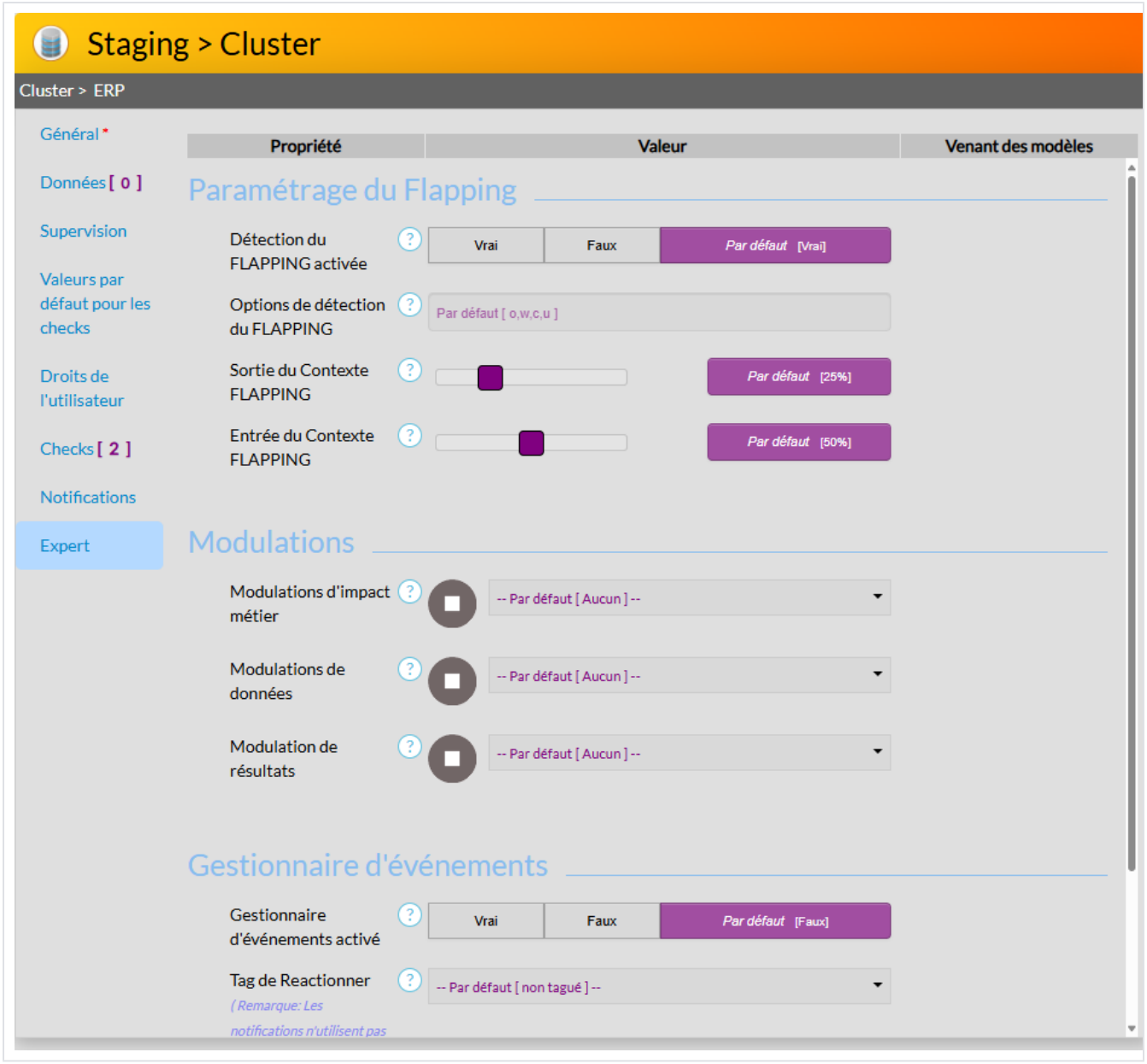
Escalades

- *Clé d'import :* escalations

Cette propriété permet de lier ce cluster à une ou plusieurs définitions d'escalade.
Si, au bout d'un certain temps, le cluster n'est toujours pas revenu OK ou pas pris en compte (*Contexte ACKNOWLEDGE ou DOWNTIME*), la règle d'escalade sera appliquée.

Cette propriété gère l'héritage additif (le +) (*voir la page Héritage additif (le +)*).

Onglet : Expert



Paramétrage du Flapping

Détection du FLAPPING activé

- *Clé d'import* : flap_detection_enabled

  Cette propriété permet de définir si la détection du Contexte **FLAPPING** est active sur ce cluster.
  Sa valeur peut être :
  - **Vrai** : active la détection du contexte **FLAPPING**.
  - **Faux** : désactive la détection du contexte **FLAPPING**.

Options de détection du FLAPPING

- *Clé d'import* : flap_detection_options

  Cette propriété permet de définir quels statuts d'un cluster sont pris en compte pour le calcul du pourcentage de **FLAPPING**.
  C'est une combinaison de l'une ou de plusieurs des valeurs suivantes :
  - **o** : statut **OK**
  - **d** : statuts **CRITICAL**
  - **u** : statuts **UNREACHABLE**

  Le séparateur est la virgule.
  La valeur o est obligatoire.

Exemple : o,d,u
Une option non existante provoquera un compteur d'erreur
L'absence de l'option **o** provoquera un compteur d'erreur

Sortie du contexte FLAPPING

- Clé d'import : low_flap_threshold

  Sur les 21 derniers statuts, chaque fois qu'un statut est différent du précédent (*de **OK** à **Warning** par exemple*), le pourcentage de **FLAPPING** augmente. Donc 10 changements représenteront un pourcentage de **FLAPPING** de 50% et 20 représenteront 100%.
  Si ce pourcentage calculé est **inférieur** au **pourcentage de sortie du Contexte FLAPPING** , alors le Contexte du cluster ne sera plus **FLAPPING**.

Entrée du contexte FLAPPING

- *Clé d'import* : high_flap_threshold

  Sur les 21 derniers statuts, chaque fois qu'un statut est différent du précédent (*de **OK** à **Warning** par exemple*), le pourcentage de **FLAPPING** augmente. Donc 10 changements représenteront un pourcentage de **FLAPPING** de 50% et 20 représenteront 100%
  Si ce pourcentage calculé est **supérieur** au **pourcentage d'entrée dans le Contexte FLAPPING** , alors le Contexte du cluster deviendra **FLAPPING**.
  Il sortira de ce Contexte quand le pourcentage calculé sera **inférieur** au **pourcentage de sortie du Contexte FLAPPING.**

Modulations

Modulations d'impact métier

- *Clé d'import :* business_impact_modulations

  Cette propriété permet de définir une ou plusieurs modulations d'impact métier. Les modulations ont une période temps durant laquelle elles sont actives.
  Pendant cette période, la valeur d'impact métier du cluster sera changée par celle de la modulation.

  Cette propriété gère l' Héritage additif (le +) .

Modulations de données

- *Clé d'import :* macromodulations

  Cette propriété permet de définir une ou plusieurs modulations de données. Les modulations ont une période temps durant laquelle elles sont actives.
  Pendant cette période, les données du cluster seront changées par celle de la modulation.

  Cette propriété gère l'héritage additif (le +) (*voir la page Héritage additif (le +)*).

Modulations de résultats

- *Clé d'import :* resultmodulations

  Cette propriété permet de définir une ou plusieurs modulations de résultats (*quatre au maximum*).
  Les modulations de résultats redéfinissent le statut de sortie du cluster, en fonction de son statut initial, d'une période de temps, ou de sa sortie.

  Cette propriété gère l'héritage additif (le +) (*voir la page Héritage additif (le +)*).

Gestionnaire d'événements

Gestionnaire d'événements activé

- *Clé d'import* : event_handler_enabled

  Cette propriété permet de définir si Shinken va lancer une commande (*définie par le paramètre **commande de gestionnaire d'événements***) à des étapes spécifiques lors des changements du statut du cluster :
  - quand le cluster passe d'un statut **OK** à un statut à problème (***CRITICAL** ou **UNKNOWN***)
  - à chaque vérification additionnelle, tant que le statut n'est pas confirmé, donc un nombre **maximum de confirmations du statut**
  - quand le statut du cluster passe d'un statut à problème vers le statut **OK**

  (*voir la page* Gestionnaire d'événements).

- Si la propriété est :
  - **Vrai** : va lancer la commande quand nécessaire.
  - **Faux** : ne va pas lancer cette commande.

Tag de Reactionner

- *Clé d'import* : reactionner_tag

  Cette propriété permet de définir le **reactionner_tag** du cluster.
  La commande lancée par le gestionnaire d'événements définie sur ce cluster sera exécutée uniquement par les Reactionners qui ont dans leur paramètre **reactionner_tags** la valeur définie par cette propriété.
  Remarque : ce Tag n'est utile que pour la commande de Gestionnaire d'événements. Les notifications utilisent le tag défini sur la commande de notification.

  Par défaut, la valeur de **Tag de Reactionner** est **non tagué**, donc les Reactionners n'ayant aucun **reactionner_tag** géreront les événements des checks d'un cluster non tagué, car la valeur par défaut pour les Reactionners est aussi **non tagué**.

Commande lancée par le gestionnaire d'événements

- *Clé d'import* : event_handler

  Cette propriété permet de définir la commande que lancera le gestionnaire d'événements pour ce cluster.