Mise en évidence d'un critère sélectionné sur carte et liste

Besoin/Objectif

Améliorer la visibilité des données sur la carte du module Synthèse (= Observations) selon un critère (précision, période, enjeux…) sélectionné par l'utilisateur.

Ressources

- Tickets GeoNature Github :
  - #3258 - [SYNTHESE] Mise en évidence d'un critère sélectionné sur la carte et sur la liste de résultats
  - #1093 - Synthèse : affichage de mailles et observations géo-confidentielles
  - #968v - Synthèse : colorer les données invalides

Questions

- **Comment gère-t-on le cas où un "point" sur la carte contient plusieurs observations dont les valeurs du critère sélectionné sont différentes ?**
  - Utiliser une couleur "neutre" (gris ?) dans un premier temps car la liste des observations où nous colorerons l'icône "i" permettra de faire la distinction.
  - Autres solutions de coloration (⇒ non retenues), en fonction :
    - de l'observations de la plus récente
    - du pourcentage le plus élevés d'observations ayant la même valeur de critère
    - de l'ordre de la liste des valeurs

Principe général

Dans le module Synthèse (= Observations), un utilisateur doit pouvoir sélectionner, via une liste déroulante intégrée à la carte, un critère pour lequel il souhaite visualiser un affichage différencié sur la carte. L'indication de la valeur du critère sélectionné pourra être également reprise au niveau de la liste des observations et sur la fiche d'une observation.

Sur la carte, nous envisageons d'utiliser dans un premier temps des couleurs distinctes pour identifier les différentes valeurs des critères sur la carte. La couleur peut en effet facilement être utilisé à la fois pour les points et les polygones. Si cela s'avère envisageable des icônes pourront être utilisé pour différencier les valeurs mais cela pose des problèmes pour les polygones (⇒ *finalement, non, car cela pose des problèmes de performance et complexifie l'affichage sur la carte*).

Au niveau de la liste des observations, un surlignage et/ou une icône peuvent être utilisé (⇒ *finalement, c'est un carré coloré en fonction de la valeur de l'observation pour le critère sélectionné*

qui s'affiche sous l'bouton "i" permettant l'accès à la fiche d'une observation. C'est la solution qui prend le moins de place) .

Dans la fiche observation, un icône, une couleur et un texte sous forme d'info-bulle serviront à la mise en évidence du critère.

Nous rendrons paramétrable :

- l'activation de cette fonctionnalité
- le choix des critères
- l'attribution des couleurs aux valeurs des critères. Si possible nous proposerons également la possibilité de définir des icônes aux valeurs. Il est également envisagé pour chaque critère de pouvoir définir le niveau d'opacité et l'utilisation ou pas d'un contour (⇒ *finalement, non car cela rentre en conflit avec d'autres fonctionnalités et cela complexifie beaucoup la configuration des critères*).

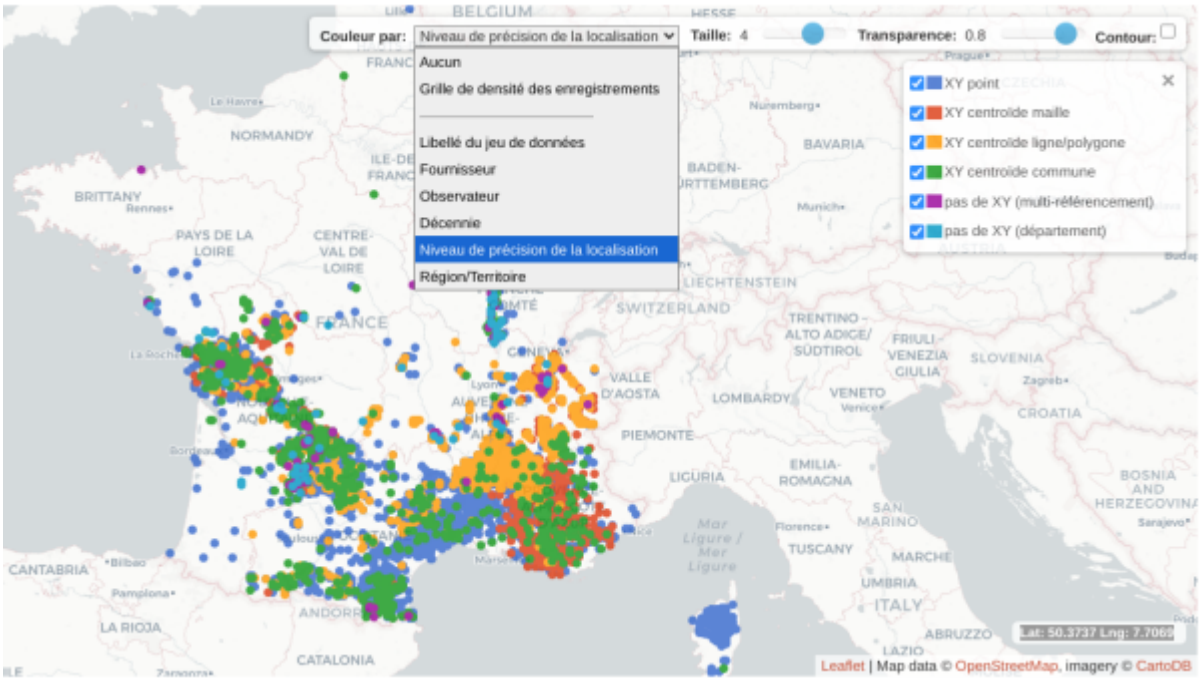
Les critères envisagés sont :

- précision (champ `precision`) : avant (précise) / après (approximative) un niveau de précision en mètre particulier.
- période (champ `date_max`) : avant (ancienne) / après (récente) une année (= date) particulière. Une alternative serait un affichage différencié par décennie.
- enjeux réglementaire (protection) : enjeu réglementaire (Oui) / pas d'enjeu réglementaire (Non) (⇒ *non pris en compte pour l'instant car cela demande l'amélioration des performances de la Synthese pour être pleinement fonctionnel*)
- menace (listes rouges) : enjeu de menace (Oui) / pas d'enjeu de menace (Non) (⇒ *non pris en compte pour l'instant car cela demande l'amélioration des performances de la Synthese pour être pleinement fonctionnel*)

Mais les critères suivant ont également étaient mentionné dans les tickets du dépôt GeoNature :

- type de l'observations (`id_nomenclature_observation_status`) : présence, absence.
- statut de validation (`id_nomenclature_valid_status`) : douteuse, probable, certaine...

Exemple d'OpenObs

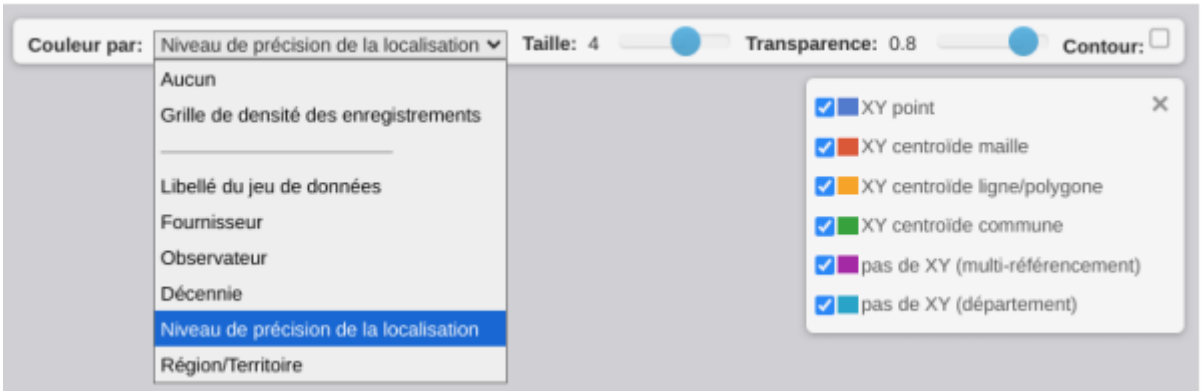


OpenObs permet de sélectionner des critères pour différencier les observations affichées sur la carte. Une liste intitulé "Couleur par" permet de sélectionner le critère d'affichage des observations sur la carte. Toutes les observations sont représentées par des points dont la couleur correspond à une valeur du critère.

L'interface permet à l'utilisateur de définir la taille du point, sa transparence et l'affichage d'un contour ou pas.

La légende peut être affiché ou réduite pour prendre moins de place sur la carte. Il est également possible de sélectionner via des cases à cocher dans la légende les valeurs à afficher. Les critères proposés par OpenObs peuvent potentiellement contenir des dizaines de valeurs, la légende possède donc une liste déroulante.

Le critère "Grille de densité des enregistrements" correspond à l'actuel mode d'affichage "Mailles" de la carte de GeoNature.



Proposition d'implémentation

Nous envisageons de faire évoluer le mode d'affichage "Mailles" des données actuellement présent sur la carte.
D'un système de coche simple, nous le transformerions en **liste de bouton radio** permettant de sélectionner une entrée unique dans la liste qui s'ouvrirait. Le **même fonctionnement que la sélection du fond cartographique** sera utilisé ici.

La liste des modes d'affichage serait représentée par **une icone représentant une liste indiquant "Mode d'affichage" au survol**. Pour l'icône d'accès aux modes d'affichage, celui de Fontawesome représentant une liste `fa-list` peut convenir : . Mais une icône ressemblant à celle du Geoportail me semble plus expressive : 

Via le **paramètre de configuration `SYNTHESE.MAP_CRITERIA_LIST`**, sur le même principe que l'ajout des fonds cartographiques, il sera possible d'ajouter de nouveaux éléments à la liste des modes d'affichage. Ce paramètre sera spécifique au module Synthese. Sa présence activera l'apparition de l'icone "modes d'affichage". Chaque critère d'affichage sera associé à un champ de la vue `gn_synthese.v_synthese_for_web_app` avec possibilité d'affecté une couleur en fonction d'une **correspondance exacte** (champs de nomenclatures ; exemples `id_nomenclature_observation_status`, `id_nomenclature_valid_status`) de sa valeur ou de son appartenance à une **classe "numérique"** (champs d'entier, exemple `precision` ou à une **classe de dates** (champ date, exemple `date_fin`). La valeur du critère sera renvoyé par le backend via le web service unique de la Synthese (`synthese/for_web`). La coloration sur la carte sera gérée par Leaflet à l'aide des informations présentes dans le GeoJson.

La liste des modes d'affichage apparaîtra également sur la carte si le paramètre `AREA_AGGREGATION_ENABLED` est à true. Dans tous les cas, la liste contiendra une entrée nommée `Affichage par défaut` correspondant à l'affichage par défaut de GeoNature sous forme de cluster puis de cercle/polygone bleu représentant un emplacement d'une ou plusieurs observations. Cette entrée sera cochée par défaut sauf si le paramètre `AREA_AGGREGATION_BY_DEFAULT` est à `true`. Dans ce dernier cas, c'est l'entrée `Mailles` correspondant à l'agrégation des observations par mailles qui sera sélectionnée. Nous proposons de la renommer `Densité d'observations` via le mécanisme de surcharge des traductions par défaut de GeoNature.

Concernant les critères "enjeux réglementaire" et "menaces" nous envisageons de rajouter des champs spécifiques à la vue `gn_synthese.v_synthese_for_web_app` dans le cadre de notre installation de GeoNature afin de les traiter sur le même principe qu'un champ de type `classes`. Mais cette modification entraine de profond changement prévu dans le cadre de l'amélioration des développement de la Synthese. Elle sera donc réalisé dans un second temps.

Pour la liste des observations, nous proposons de colorer le fond du bouton contenant l'icône "i" de chaque observations en fonction du critère sélectionné. De cette façon, il n'y aura aucune perte d'espace pour les textes des colonnes de cette liste.

Dans le cas des **géométries regroupant des observations aux valeurs contradictoires** vis à vis du critère proposé. Nous proposons dans un premier temps de les afficher à l'aide d'une couleur neutre (**grise** ; configurable pour chaque critère). Un clic sur la géométrie affichera la liste des observations surlignés en vert en tête de liste des observations où la couleur de l'icone "i" permettra de les distinguer les valeurs des différentes observations en fonction du critère sélectionné.

Dans le cas des **géométries possédant des valeurs inconnues** (ou non prise en compte par l'administrateur du GeoNature dans sa configuration), nous les afficherons à l'aide d'une couleur

distinctive (**rouge** ; configurable pour chaque critère).

Au niveau de la fiche d'observation, nous testerons l'ajout au niveau de l'entête d'un badge coloré en fonction de la valeur du critère pour l'observation affichée. Si une icone a été associée au critère, elle sera affiché dans la couleur indiqué par la configuration au milieu du badge. Au survol, une infobulle affichera la description indiquée pour ce critère dans la configuration.

Détail du paramètre SYNTHESE.MAP_CRITERIA_LIST

Ce paramètre sera une liste de dictionnaire où chaque dictionnaire correspondra à un critère. Pour chaque critère (= dictionnaire) nous aurons les champs suivant où les champs **en gras** sont obligatoires :

- **code** (chaine) : code alphanumérique du critère utilisé pour identifier ce critère d'un point de vue technique. Ne devrait pas être utilisé pour l'affichage ou vis à vis de l'utilisateur. Devrait être unique dans la liste et devrait éviter de contenir des caractères spéciaux.
- **label** (chaîne) : intitulé par défaut du critère à utilisé. Servira à l'affichage dans la liste des modes d'affichage.
- **type** (chaîne) : le type de critère parmi : `nomenclatures`, `classes`, `dates`. Le critère de type `nomenclatures` est à utiliser avec tous les champs `field` correspondant à une nomenclature ou à un champ texte pour lequel on veut réaliser une coloration en fonction d'une valeur précise. Le critère de type `classes` est à utiliser avec les champs de type numérique et le type `dates` avec les champs date pour lesquels on veut établir des classes de valeurs.
- **field** (chaîne) : le nom du champ présent dans la vue `gn_synhtese.v_synthese_for_web_app` à utiliser pour le critère.
- `mnemonic` (chaine) : pour un critère de type `nomenclatures` permet d'indiquer la mnémonique de la nomenclature afin de pouvoir utiliser le `cd_nomenclature` des valeurs de la nomenclatures à la place des `id_nomenclature`, ci-dessous, dans `values.value`.
- `activate` (booléen, défaut : true) : indique si critère est à afficher (`true`) ou pas (`false`) dans la liste des modes d'affichage.
- `default` (booléen, défaut : false) : indique si ce critère doit être utilisé par défaut (`true`) ou pas (`false`) comme mode d'affichage. Le premier critère à `true` trouvé dans la liste sera pris en compte. Si, non présent, correspond à la valeur `false`.
- **values** (liste de dictionnaire) : liste de dictionnaire correspondant aux valeurs possibles du critère. Chaque dictionnaire d'une valeur pourra contenir les champs :
  - **value** (liste de chaines | chaîne | nombre) : dans le cas du type `nomenclature` contiendra un code ou une liste de codes de la nomenclature, champ `cd_nomenclature` si le champ `mnemonic` du critère a été renseigné ; sinon il est possible d'indiquer les valeurs du champ `id_nomenclature`. Dans le cas du type `classes`, il contiendra la valeur minimum de comparaison à utiliser vis à vis du contenu du champ `field`. Un objet sur la carte prend la couleur correspondante si sa valeur dans le champ `field` est supérieure ou égale à la valeur indiquée. L'ordre des entrées de la liste `values` n'est pas important car pour les *classes* et les *dates*, les valeurs du champs `value` sont triées par ordre décroissant avant d'être comparées.
  - `label` (chaîne) : intitulé court par défaut de la valeur. Servira à l'affichage dans la légende. Si cet attribut n'est pas renseigné, il prend la valeur de l'attribut "value".
  - `description` : intitulé long de la valeur. Affiché principalement via des info-bulles.
  - `color` (chaîne) : code hexadécimal de la couleur de la valeur. La valeur sera utilisée pour le style `fillcolor`. Si non renseigné, une couleur sera automatiquement attribuée. Il n'est pas garantie que la couleur soit systématiquement la même entre chaque

rechargement de la configuration.
    - `icon` (liste de dictionnaire) : liste de dictionnaire correspondant aux valeurs possibles du critère. Chaque dictionnaire d'une valeur pourra contenir les champs :
        - **`name`** (chaîne) : indiquer une icone à l'aide de son nom. Possibilité d'utiliser des icônes : Fontawsome, Angular Material ou personnalisé via le mécanisme de polices d'icones (à tester).
        - **`font`** (chaîne) : indiquer le nom de la police contenant l'icône. Par exemple, `fa` pour Fontawesome ou `material-icons-outlined` pour les polices Material. Pour l'instant, l'affichage a été testé avec les icônes "FontAwesome v4.7" et les icônes Angular Material.
        - `color` (chaîne) : code hexadécimal, valeur RGBA, nom web... de la couleur de l'icône. La valeur de couleur sera utilisée avec la propriété CSS `color`. Ce paramètre peut donc prendre n'importe quelle valeur autorisée par cette propriété. Si non précisé, la couleur par défaut `white` sera utilisée.
    - ~~`styles` (dictionnaire) : contient un dictionnaire des clés et valeurs de styles à utiliser en complément de la couleur de remplissage. Les styles disponibles sont ceux de l'option `style` de l'objet GeoJson de Leaflet.~~ ⇒ *Trop complexe à configurer et entre en conflit avec des fonctionnalités existantes de la Synthese (nombre d'observations affiché sur les marqueurs sous forme d'infor-bulle opaque centrée sur le marqueur).*

À faire

À faire sur branche `feat/sinp` :

- ☒ [✓ jpmilcent, 2025-02-12]~~Renommer "Font" en "IconFont"~~
- ☒ [✓ jpmilcent, 2025-02-12]~~Supprimer tous les console.log() et les print()~~
- ☒ [✓ jpmilcent, 2025-02-12]~~Créer un commit supprimer le code de la fiche InfoObs de la Synthese inutile à placer en premier~~
- ☒ [✓ jpmilcent, 2025-02-12]~~Dissocier dans son propre commit les modifs des fonctions translateToaster() et regularToaster() de CommonService~~
- ☒ [✓ jpmilcent, 2025-02-13]~~Garder seulement le champ "color" dans les paramètres et supprimer "styles"~~
- ☒ [✓ jpmilcent, 2025-02-13]~~Laisser le placement des contrôles dans SyntheseCarteComponent et garder uniquement onAdd() dans SyntheseCriteriaService~~
- ☒ [✓ jpmilcent, 2025-02-13]~~Ajouter une vérification de la structure de la config MAP_CRITERIA~~
- ☒ [✓ jpmilcent, 2025-02-13]~~Ajouter des fonctions haveSelectedCriteria(), getCurrentCriteria() à SyntheseCriteriaService~~
- ☒ [✓ jpmilcent, 2025-02-20]~~Régler le problème de la valeur OUI absente du fichier d'Export pour les données sensibles~~
- ☒ [✓ jpmilcent, 2025-02-20]~~Régler problème affichage du champ dans message d'avertissement à récupérer depuis la config~~
- ☒ [✓ jpmilcent, 2025-02-21]~~Ajouter le label en plus de la description dans l'infobulle de la mode d'info d'une obs~~ car la description n'est pas un champ obligatoire, ni l'icone...
- ☒ [✓ jpmilcent, 2025-05-22]~~Utiliser les cd_nomenclature et non pas les id_nomenclature dans la config MAP_CRITERIA~~

À faire uniquement sur intégration dans branche `develop` :

- ☒ [✓ jpmilcent, 2025-02-13]~~Remplacer "Valeurs inconnues" par "Valeur autre" en couleur grise~~
- ☒ [✓ jpmilcent, 2025-05-22]~~Récupérer les commits permettant l'override de l'I18N~~
- ☐ Ajouter une doc sur la config de MAP_CRITERIA
- ☒ [✓ jpmilcent, 2025-05-22]~~Voir si le mécanisme d'additional_fields ajouter à la Synthese peut être utilisé pour remplacer with_field du web service for_web~~

Amélioration pour plus tard :

- ☐ Réfléchir : à l'amélioration des performances en ne rechargeant pas les données à chaque changement de critère
- ☐ Voir le mécanisme de coloration utilisé dans le module Validation

From:
https://sinp-wiki.cbn-alpin.fr/ - **CBNA SINP**

Permanent link:
https://sinp-wiki.cbn-alpin.fr/fonctionnalites/geonature/synthese-mise-en-evidence-critere

Last update: **2025/05/22 15:55**