

Brique Copernicus 0.1 release notes

L'application permet la récupération et le traitement d'images spatiales simplifiées dans un langage de programmation graphique type Scratch. La configuration requise et les pré-requis pour utiliser l'application sont précisés dans le manuel d'installation et le manuel d'utilisation.

Limitations / Contraintes

- Il n'est pas possible de manipuler d'image directement depuis la brique de téléchargement.

Solutions non-abouties et bugs connus

- La prise en charge des périodes dans le traitement des données n'est pas fonctionnel. Celui-ci était en cours à la fin du projet. Ce traitement se fait grâce aux modules netCDF et xarray.
- Lors de l'exécution d'une requête copernicus, il est possible que la requête ne s'exécute pas totalement, et dans ce cas il faut relancer plusieurs fois la requête avec le bouton "Start" (sans forcément recharger la page) pour que la requête (ainsi que le traitement et l'organisation automatique) aboutisse. La cause de ce bug n'est pas connue et il n'est pas récurrent.
- L'enregistrement des images (ou des graphiques générés) sur le disque dur de l'utilisateur n'est pas prise en compte car il nécessite d'ouvrir une boîte de dialogue demandant le chemin de fichier (et donc interrompant forcément une boucle Scratch qui se voudrait enregistrer automatiquement des images en vue de concevoir et enregistrer une animation). Cette fonctionnalité a été délaissée pour l'enregistrement du code Blockly. Mais il y a une potentielle piste pour enregistrer un fichier texte via un JS qui appelle du Python sans demander un nom de fichier à l'utilisateur.

Solutions à mettre en place

- Si vous ajoutez de nouvelles requêtes il est possible qu'elles téléchargent des archives .zip ou .tar mais leur traitement ne sera pas pris en compte. Les archives ne seront pas décompressées ni ventilées. On pourra néanmoins les retrouver dans le dossier "Blockly" une fois elles sont téléchargées. Les requêtes peuvent aussi télécharger des fichiers .grib qui ne sont pas pris en compte par le traitement actuel. Comme pour les archives, les fichiers .grib ne seront pas ventilés et seront téléchargés dans le dossier "Blockly". La formation de la requête et du code de

traitement se font par le fichier copernicusRequest.js dans le dossier “./Blockly/asstes/js/”.

- Parmi les fonctionnalités souhaitées au début du projet, il y avait la possibilité de zoomer sur l'image de la zone d'affichage. Cette fonctionnalité n'a pas été mise en place et il n'y a pas de piste développée.
- Parmi les fonctionnalités souhaitées au début du projet, il y avait la possibilité d'insérer un texte ou une icône sur une image ce qui a été délaissé durant le projet. Les images et données s'affichent dans un canvas HTML, il est possible d'écrire et dessiner dans un canvas HTML, ainsi il est envisageable d'utiliser les outils du canvas HTML pour écrire et dessiner sur les images et données affichées.
- Dans la version actuel, le traitement des fichiers csv (la formation d'un graphique) n'en est qu'au stade de prototype. Ce traitement ne fonctionne que pour le fichier exemple.csv du dossier mesDatas. Le traitement se fait dans le fichier chart-maker.js. De plus un fichier ne peut pas être afficher en tableau brut, sans traitement et transformation, ce qui était une intention de départ. On peut envisager d'utiliser les fonctionnalités du canvas HTML pour le mettre en place.