
TRENDS.EARTH
tracking land change
from Conservation International

Trends.Earth - Guía para desarrolladores

Versión 2.1.8

Conservation International

06 de febrero de 2023

1	Desarrollo	2
1.1	Modificación del código del complemento QGIS	2
1.2	Modificar el código de procesamiento de Earth Engine	6
1.3	Edición de plantillas de capas vectoriales	8
1.4	Gestión de los metadatos de los conjuntos de datos	9
1.5	Actualización del marco de información	10
1.6	Contribuyendo a la documentación	13
2	Registro de cambios	17
2.1	2.1.8 (January 25, 2023)	17
2.2	2.1.6 (December 8, 2022)	17
2.3	2.1.6 (November 21, 2022)	17
2.4	2.1.2 (November 18, 2022)	18
2.5	2.1.0 (November 11, 2022)	18
2.6	2.0.7 (October 31, 2022)	18
2.7	2.0.5 (October 19, 2022)	18
2.8	2.0.3 (October 19, 2022)	18
2.9	2.0.1 (October 13, 2022)	18
2.10	2.0 (July 20, 2022)	19
2.11	1.0.10 (7 de julio de 2022)	19
2.12	1.0.8 (15 de octubre de 2021)	19
2.13	1.0.6 (15 de julio de 2021) < https://github.com/ConservationInternational/trends.earth/releases/tag/1.0.6 >	19
2.14	1.0.4 (30 de junio de 2021) < https://github.com/ConservationInternational/trends.earth/releases/tag/1.0.4 >	19
2.15	1.0.2 (14 de agosto de 2020) < https://github.com/ConservationInternational/trends.earth/releases/tag/1.0.2 >	20
2.16	1.0.0 (27 de abril de 2020) < https://github.com/ConservationInternational/trends.earth/releases/tag/1.0.0 >	20
2.17	0.98 (2 de abril de 2020)	21
2.18	0.66 (20 de julio de 2019)	21
2.19	0.64 (9 de julio de 2019)	21
2.20	0.62 (27 de enero de 2019)	22
2.21	0.60 (3 de diciembre de 2018)	22
2.22	0.58 (11 de agosto de 2018)	22
2.23	0.56.5 (21 de mayo de 2018)	22

2.24	0.56.4 (21 de mayo de 2018)	22
2.25	0.56.3 (21 de abril de 2018)	23
2.26	0.56.2 (10 de abril de 2018)	23
2.27	0.56.1 (10 de abril de 2018)	23
2.28	0.56 (9 de abril de 2018)	23
2.29	0.54 (8 de abril de 2018)	23
2.30	0.52.1 (21 de marzo de 2018)	23
2.31	0.52.1 (21 de marzo de 2018)	24
2.32	0.52 (19 de marzo de 2018)	24
2.33	0.50 (15 de marzo de 2018)	24
2.34	0.48 (13 de marzo de 2018)	24
2.35	0.46 (13 de marzo de 2018)	24
2.36	0.44 (12 de marzo de 2018)	24
2.37	0.42 (4 de febrero de 2018)	25
2.38	0.40 (4 de febrero de 2018)	25
2.39	0.38 (16 de enero de 2018)	25
2.40	0.36 (14 de diciembre de 2017)	26
2.41	0.34 (14 de diciembre de 2017)	26
2.42	0.32 (14 de diciembre de 2017)	26
2.43	0.30 (12 de diciembre de 2017)	26
2.44	0.24 (6 de diciembre de 2017)	26
2.45	0.22 (4 de diciembre de 2017)	26
2.46	0.18 (2 de diciembre de 2017)	26
2.47	0.16 (6 de noviembre de 2017)	26
2.48	0.14 (25 de octubre de 2017)	26
2.49	0.12 (6 de octubre de 2017)	26

▲TRENDS.EARTH es un software gratuito y de código abierto, licenciado bajo la [Licencia Pública General GNU](#), versión 2.0 o posterior.

Hay una serie de componentes para las ▲TRENDS.EARTH herramienta. El primero es un complemento QGIS que admite el cálculo de indicadores, el acceso a datos sin procesar, informes y producción de mapas impresos. El código para el complemento, y más instrucciones sobre cómo instalarlo si desea modificar el código, están en [trends.earth](#) Depósito de GitHub.

El complemento de QGIS ▲TRENDS.EARTH es compatible con varios scripts Python diferentes que permiten el cálculo de varios indicadores en Google Earth Engine (GEE). Estas secuencias de comandos se encuentran en la subcarpeta «gee» de ese repositorio de GitHub. Los scripts GEE son compatibles con el módulo Python *landdegradation*, que incluye código para procesar entradas y salidas para el complemento, así como otras funciones comunes que admiten el cálculo de integrales NDVI, significado estadístico y otro código compartido. El código para este módulo está disponible en [landdegradation](#) repositorio en GitHub.

A continuación encontrará más detalles sobre cómo contribuir a Trends.Earth trabajando en el código del complemento, modificando el código de procesamiento o contribuyendo a traducir el sitio web y el complemento.

1.1 Modificación del código del complemento QGIS

1.1.1 Descargando el código de Trends.Earth.

El código Trends.Earth, tanto para el complemento como para los scripts de Google Earth Engine, se encuentra en GitHub en [trends.earth](#) repositorio. Clone este repositorio en un lugar conveniente de su máquina para asegurarse de tener la última versión del código.

Hay varias ramas diferentes del repositorio trends.earth que están en desarrollo activo. El «plugin» es oficialmente compatible con QGIS3 y la mayor parte del desarrollo tiene lugar en la rama «develop». La rama «qgis2» es la versión más antigua del «plugin» y soporta la versión 2.18+ de QGIS2.

1.1.2 Instalar dependencias

Python

El complemento está codificado en Python. Además de usarse para ejecutar el complemento a través de QGIS, Python también se usa para admitir la administración del complemento (cambiar la versión, instalar versiones de desarrollo, etc.). Aunque Python está incluido con QGIS, también necesitará una versión local de Python que pueda configurar con el software necesario para administrar el complemento. La forma más fácil de administrar múltiples versiones de Python es a través de la distribución [Anaconda](#). Para trabajar desarrollando el complemento, se requiere Python 3. Para descargar Python 3.7 (recomendado) a través de Anaconda, [vea esta página](#).

Dependencias de Python

Para trabajar con el código de tendencias.earth, debe tener Invoke instalado en su máquina, así como una serie de otros paquetes que se utilizan para administrar la documentación, las traducciones, etc. Estos paquetes se enumeran en el «dev» archivo de requisitos para Trends.Earth, por lo que se pueden instalar navegando en un símbolo del sistema a la raíz de la carpeta de códigos trend.earth y escribiendo:

```
pip install -r requirements-dev.txt
```

Nota: Si está utilizando Anaconda, primero querrá activar un entorno virtual Python 3.7 antes de ejecutar el comando anterior (y cualquiera de los otros comandos de invocación enumerados en la página). Una forma de hacerlo es iniciando un «Anaconda prompt», siguiendo las instrucciones en esta página de Anaconda <<https://docs.anaconda.com/anaconda/user-guide/getting-started/#write-a-python-program-using-anaconda-prompt-or-terminal>>`_.

PyQt

PyQt5 es el kit de herramientas gráficas utilizado por QGIS3. Para compilar la interfaz de usuario de Trends.Earth para QGIS3 necesita instalar PyQt5. Este paquete se puede instalar desde pip usando:

```
pip install PyQt5
```

Nota: PyQt4 es el kit de herramientas gráficas utilizado por QGIS2. La mejor fuente para este paquete en Windows es el conjunto de paquetes mantenido por Christoph Gohlke en UC Irvine. Para descargar PyQt4, seleccione [el paquete apropiado de esta página](#). Elija el archivo apropiado para la versión de Python que está utilizando. Por ejemplo, si está utilizando Python 2.7, elija la versión con «cp27»; en el nombre del archivo. Si está utilizando Python 3.7, elija la versión con «cp37»; en el nombre del archivo. Elija «amd64»; para python de 64 bits y «win32»; para python de 32 bits.

Después de descargar desde el enlace anterior, use `pip` para instalarlo. Por ejemplo, para la rueda de 64 bits para Python 3.7, ejecutaría:

```
pip install PyQt4-4.11.4-cp37-cp37m-win_amd64.whl
```

1.1.3 Cambiar la versión del complemento

La convención para Trends.Earth es que los números de versión que terminan en un número impar (por ejemplo, 0.65) son versiones de desarrollo, mientras que las versiones que terminan en un número par (por ejemplo (0.66) son versiones de lanzamiento. Las versiones de desarrollo del complemento nunca se lanzan a través de el repositorio QGIS, por lo que nunca son vistos por los usuarios normales del complemento. El equipo de desarrollo de Trends.Earth utiliza versiones de desarrollo con números impares mientras prueba nuevas características antes de su lanzamiento público.

Si desea realizar cambios en el código y ha descargado una versión pública del complemento (uno que termina en un número par), el primer paso es actualizar la versión del complemento al siguiente número impar secuencial. Entonces, por ejemplo, si descargó la versión 0.66 del complemento, necesitaría actualizar la versión para que sea 0.67 antes de comenzar a hacer sus cambios. Hay varios lugares en el código donde se menciona la versión (así como dentro de cada script GEE), por lo que hay una tarea de invocación para ayudar a cambiar la versión. Para cambiar la versión a 0.67, debe ejecutar:

```
invoke set-version -v 0.67
```

La ejecución del comando anterior actualizará el número de versión en cada lugar al que se hace referencia en el código. Para evitar confusiones, nunca cambie la versión a una que ya haya sido lanzada, siempre AUMENTE el valor de la etiqueta de versión al siguiente número impar.

1.1.4 Probar cambios en el complemento

Después de realizar cambios en el código del complemento, deberá probarlos para asegurarse de que el complemento se comporta como se esperaba y para asegurarse de que no surjan errores o errores. El complemento debe someterse a pruebas exhaustivas antes de ser lanzado al repositorio QGIS (donde otros usuarios pueden acceder a él) para garantizar que cualquier cambio en el código no rompa el complemento.

Para probar cualquier cambio que haya realizado en el complemento dentro de QGIS, deberá instalarlo localmente. Hay tareas de invocación que ayudan con este proceso. El primer paso antes de instalar el complemento es asegurarse de haber configurado el complemento con todas las dependencias que necesita para ejecutarse desde QGIS. Para hacer esto, ejecute:

```
invoke plugin-setup
```

La tarea anterior solo debe ejecutarse inmediatamente después de descargar el código de tendencias de la tierra, o si se realizan cambios en las dependencias del complemento. Por defecto, `plugin-setup` reutilizará cualquier archivo en caché en su máquina. Para comenzar desde cero, agregue el indicador `-c` (limpiar) al comando anterior.

Después de ejecutar `plugin-setup`, está listo para instalar el complemento en la carpeta de complementos QGIS en su máquina. Para hacer esto, ejecute:

```
invoke plugin-install
```

Después de ejecutar el comando anterior, tendrá que 1) reiniciar QGIS, o 2) utilizar el [Plugin Reloader](#) para recargar el «plugin» Trends.Earth con el fin de ver los efectos de los cambios que ha realizado.

Por defecto, `plugin-install` sobrescribirá cualquier archivo de complemento existente en su máquina, pero dejará en su lugar cualquier dato (límites administrativos, etc.) que el complemento podría haber descargado. Para comenzar desde cero, agregue el indicador `-c` (limpio) al comando anterior. Es posible que deba cerrar QGIS para realizar una instalación limpia del complemento utilizando el indicador `-c`.

Nota: De forma predeterminada, `plugin-install` supone que desea instalar el complemento que se utilizará en QGIS3. Para instalar el complemento para su uso en QGIS3, agregue el indicador `-v 2` al comando `plugin-install`. Recuerde

que el complemento puede o no ser completamente funcional en QGIS3: el complemento se diseñó originalmente para QGIS2 y todavía se está probando en QGIS3.

1.1.5 Sincronizar e implementar cambios en los binarios

Para acelerar los cálculos en Trends.Earth, algunas de las herramientas permiten utilizar binarios precompilados que se han compilado usando `numba`. Numba es un compilador de código abierto que puede compilar código Python y NumPy, haciéndolo más rápido que cuando se ejecuta como Python ordinario. Para evitar que los usuarios de Trends.Earth necesiten descargar Numba y todas sus dependencias, el equipo de Trends.Earth pone a disposición binarios precompilados para descargar si los usuarios eligen instalarlos.

Para generar binarios precompilados para el sistema operativo, el bitness (32/64 bit) y la versión de Python que está ejecutando en su máquina, use:

```
invoke binaries-compile
```

Nota: Necesitará un compilador de C ++ para que funcione el comando anterior. En Windows, vea [esta página de github](#) para obtener detalles sobre cómo instalar el compilador de Microsoft Visual C ++ necesario para su versión de Python. En MacOS, lo más probable es que necesites instalar Xcode. En Linux, instale la versión adecuada de GCC.

Para que los binarios estén disponibles públicamente, se distribuyen a través de un bucket de servicios web de Amazon S3. Para cargar los binarios generados con el comando anterior en el depósito, ejecute:

```
invoke binaries-sync
```

Nota: El comando anterior fallará si no tiene teclas que permitan el acceso de escritura al bucket `trends.earth` en S3.

El comando anterior sincronizará cada archivo binario individual con S3. Sin embargo, los usuarios de la caja de herramientas descargan los archivos binarios como un único archivo zip vinculado a la versión del complemento que están utilizando. Para generar ese archivo zip para que los usuarios de Trends.Earth puedan acceder a él, ejecute:

```
invoke binaries-deploy
```

Nota: El comando anterior fallará si no tiene teclas que permitan el acceso de escritura al bucket `trends.earth` en S3.

1.1.6 Crear un archivo ZIP de complemento

Hay varias tareas de invocación para ayudar a construir un archivo ZIP para implementar el complemento en el repositorio QGIS, o para compartir la versión de desarrollo del complemento con otros. Para empaquetar el complemento y todas sus dependencias en un archivo ZIP que se puede instalar siguiendo [el proceso descrito en el archivo Léame de Trends.Earth](#), ejecutar:

```
invoke zipfile-build
```

Este comando creará una carpeta llamada `build` en la raíz de la carpeta de tendencias `trend.earth`, y en esa carpeta creará un archivo llamado `LDMP.zip`. Este archivo se puede compartir con otros, que pueden usarlo para [instalar manualmente](#)

Trends.Earth. Esto puede ser útil si es necesario compartir las últimas funciones con alguien antes de que estén disponibles en la versión pública del complemento.

1.1.7 Implementación del archivo ZIP de la versión de desarrollo

La página Trends.Earth GitHub proporciona un enlace a un archivo ZIP que permite a los usuarios que no sean desarrolladores acceder a la versión de desarrollo de Trends.Earth. Para crear un archivo ZIP y ponerlo a disposición en esa página (el archivo ZIP se almacena en S3), ejecute:

```
invoke zipfile-deploy
```

Este comando empaquetará el complemento y lo copiará a <https://s3.amazonaws.com/trends.earth/sharing/LDMP.zip>.

Nota: El comando anterior fallará si no tiene teclas que permitan el acceso de escritura al bucket `trends.earth` en S3.

1.2 Modificar el código de procesamiento de Earth Engine

Las secuencias de comandos de procesamiento de Google Earth Engine (GEE) utilizadas por Trends.Earth se almacenan en la carpeta «gee»; debajo de la carpeta Trends.earth principal. Para que este script sea accesible para los usuarios del plugin Trends.earth QGIS, deben implementarse en el servicio `api.trends.earth` que Conservation International mantiene para permitir a los usuarios del plugin usar Earth Engine sin la necesidad de saber cómo programar o tener cuentas de usuario individuales en GEE. A continuación se describe cómo probar e implementar scripts GEE para usar con Trends.Earth.

1.2.1 Configurar dependencias

trends.earth-CLI

Se requiere el paquete Python «trends.earth-CLI» para trabajar con el servidor `api.trends.earth`. Este paquete se encuentra en GitHub en [trends.earth-CLI](#) repositorio.

El primer paso es clonar este repositorio en su máquina. Recomendamos que clone el repositorio en la misma carpeta donde se encuentra el código de tendencias. Por ejemplo, si tenía una carpeta «Código» en su máquina, clone ambas tendencias. <<https://github.com/ConservationInternational/trends.earth>>_ repositorio (el código para el complemento QGIS y los scripts GEE asociados) y también las `trends.earth-CLI` <<https://github.com/Vizzuality/trends.earth-CLI>>_ repositorio en esa misma carpeta.

Cuando configure su sistema como se recomienda anteriormente, trends.earth-CLI trabajará con las tareas de invocación utilizadas para administrar Trends.earth sin ninguna modificación. Sin embargo, si descarga Trends.earth-CLI en una carpeta diferente, deberá agregar un archivo llamado «invoke.yaml» en la raíz del repositorio de tendencias.earth, y en ese archivo diga a Trends.Earth dónde para localizar las tendencias. Código Earth-CLI. Este archivo YAML debería parecerse al siguiente (si descargó el código en Windows en una carpeta llamada «C:/Users/azvol/Code/trends.earth-CLI/tecli»):

```
gee:
  tecli: "C:/Users/azvol/Code/trends.earth-CLI/tecli"
```

Nuevamente, **no** necesita agregar este archivo .yaml si configura su sistema como se recomienda anteriormente.

docker

El paquete trends.earth-CLI requiere docker para funcionar. Siga estas instrucciones para instalar Docker en Windows y estas instrucciones para instalar Docker en Mac OS <<https://docs.docker.com/docker-for-mac/install/>>. Si está ejecutando Linux, siga las instrucciones en esta página que son apropiados para la distribución de Linux que está utilizando.

1.2.2 Probar un script de Earth Engine localmente

Después de instalar el paquete trends.earth-CLI, deberá configurar un archivo .tecli.yml con un token de acceso a una cuenta de servicio GEE para probar los scripts en GEE. Para configurar la cuenta de servicio GEE para tecli, primero obtenga la clave para su cuenta de servicio en formato JSON (de la consola de Google Cloud), luego codifíquela en base64. Proporcione esa clave codificada en base64 a tecli con el siguiente comando:

```
invoke tecli-config set EE_SERVICE_ACCOUNT_JSON key
```

donde «key» es la llave de la cuenta de servicio de formato JSON codificada en base64.

Al convertir un script que especifica el código que se ejecutará en GEE de JavaScript a Python, o al realizar modificaciones en ese código, puede ser útil probar el script localmente, sin implementarlo en el servidor api.trends.earth. Para hacer esto, use la tarea de invocación ejecutar. Por ejemplo, para probar el script «land_cover», vaya al directorio raíz del código Trends.Earth y, en un símbolo del sistema, ejecute:

```
invoke tecli-run land_cover
```

Esto usará el paquete trends.earth-CLI para construir y ejecutar un contenedor acoplable que intentará ejecutar el script «land_cover». Si hay errores de sintaxis en el script, estos aparecerán cuando se ejecute el contenedor. Antes de enviar un nuevo script a api.trends.earth, asegúrese siempre de que invoke tecli-run pueda ejecutar el script sin ningún error.

Al usar invoke tecli-run, puede recibir un error que dice:

```
Invalid JWT: Token must be a short-lived token (60 minutes) and in a reasonable timeframe. Check your iat and exp values and use a clock with skew to account for clock differences between systems.
```

Este error puede ser causado si el reloj en el contenedor docker no está sincronizado con el reloj del sistema. Reiniciar Docker debería corregir este error.

1.2.3 Implementar un script GEE en api.trends.earth

Cuando haya terminado de probar un script GEE y le gustaría que fuera accesible usando el complemento QGIS (y por otros usuarios de Trends.Earth), puede implementarlo en el servidor api.trends.earth. El primer paso en el proceso es iniciar sesión en el servidor api.trends.earth. Para iniciar sesión, ejecute:

```
invoke tecli-login
```

Se le pedirá un nombre de usuario y contraseña. Estos son los mismos que el nombre de usuario y la contraseña que utiliza para iniciar sesión en el servidor de Trends.Earth desde el complemento QGIS. **Si no es administrador, podrá iniciar sesión, pero el siguiente comando fallará.** Para cargar un script (por ejemplo, el script «land_cover») en el servidor, ejecute:

```
invoke tecli-publish -s land_cover
```

Si este script ya existe en el servidor, se le preguntará si desea sobrescribir el script existente. Tenga mucho cuidado al cargar scripts con versiones pares, ya que estos son scripts disponibles públicamente y cualquier error que cometa afectará a cualquiera que use el complemento. Siempre que esté probando asegúrese de usar números de versión de desarrollo (números de versión impares).

Después de publicar una secuencia de comandos en el servidor, puede utilizar la tarea *tecli-info* para verificar el estado de la secuencia de comandos (para saber si se implementó correctamente, aunque la creación de la secuencia de comandos puede demorar unos minutos). Para verificar el estado de un script implementado, ejecute:

```
invoke tecli-publish -s land_cover
```

Si está realizando una nueva versión del complemento y desea cargar TODOS los scripts de GEE a la vez (esto es necesario siempre que cambie el número de versión del complemento), ejecute:

```
invoke tecli-publish
```

Nuevamente, nunca ejecute lo anterior en una versión pública del complemento a menos que tenga la intención de sobrescribir todos los scripts disponibles públicamente utilizados por el complemento.

1.3 Edición de plantillas de capas vectoriales

Trends.Earth permite a los usuarios digitalizar nuevas características vectoriales para delinear áreas de interés especial.

Por ahora sólo se soportan las capas «falso positivo/negativo», pero se pueden añadir más si es necesario. Cualquier capa vectorial se crea a partir de los archivos de plantilla GeoPackage, que se encuentran dentro de la carpeta «data/error_recode» del directorio de instalación del complemento o extensión. Para cada tipo de vector hay seis archivos de plantilla, uno por cada idioma oficial de la ONU. El código de idioma ISO se añade como sufijo al nombre del archivo. Esto es necesario para proporcionar etiquetas localizadas en los formularios de atributos. Cuando se solicite la creación de la capa vectorial, QGIS buscará el archivo de plantilla teniendo en cuenta la configuración regional del propio QGIS; como opción alternativa se utilizará la versión inglesa del archivo de plantilla.

Para cambiar el esquema de la capa es necesario cambiar los archivos de plantilla correspondientes en la carpeta «data/error_recode» del directorio de instalación del complemento o extensión. También el archivo de plantilla contiene una configuración de estilo y atributos por defecto que se aplicará automáticamente a la capa cuando se cargue en QGIS.

Para mostrar las listas en forma de atributos se utiliza un «widget» QML incorporado. Los datos para las listas se almacenan en la tabla de atributos de la capa vectorial. Los valores de los campos correspondientes se extraen con la ayuda de expresiones.

El código para generar las listas tiene el siguiente aspecto:

```
import QtQuick 2.0
import QtCharts 2.0

ChartView {
    width: 380
    height: 200
    margins {top: 0; bottom: 0; left: 0; right: 0}
    backgroundColor: "#eeeeec"
    legend.alignment: Qt.AlignBottom
    antialiasing: true
    ValueAxis {
        id: valueAxisY
        min: 0
        max: 100
    }
}
```

(continué en la próxima página)

(proviene de la página anterior)

```

BarSeries {
    id: mySeries
    axisY: valueAxisY
    axisX: BarCategoryAxis { categories: ["Productivity", "Land cover", "Soil_
↪organic carbon"] }
    BarSet { label: "Degraded"; color: "#9b2779"; values: [expression.evaluate("\
↪"prod_deg\""), expression.evaluate("\land_deg\""), expression.evaluate("\soil_deg\
↪")] }
    BarSet { label: "Improved"; color: "#006500"; values: [expression.evaluate("\
↪"prod_imp\""), expression.evaluate("\land_imp\""), expression.evaluate("\soil_imp\
↪")] }
    BarSet { label: "Stable"; color: "#ffffe0"; values: [expression.evaluate("\
↪"prod_stab\""), expression.evaluate("\land_stab\""), expression.evaluate("\soil_
↪stab\"")] }
}
}

```

To extract field value function `expression.evaluate("\prod_deg\"")` is used, the only argument it accepts is the name of the field. For false positive/negative layers chart contains three indicators: productivity, land cover and soil organic carbon. For each indicator plugin keeps three values stable, degraded and improved percentage of the polygon area. For example, in case of productivity indicator fields will be:

- `prod_deg`: productividad degradada
- `prod_stab`: productividad estable
- `prod_imp`: productividad mejorada

El mismo enfoque de la nomenclatura se aplica a la cobertura del suelo (`land_*` fields) y al carbono orgánico del suelo (`soil_*` fields).

El cálculo del porcentaje de área se realiza con una función de expresión personalizada, su código se puede encontrar en el archivo `charts.py` en el directorio raíz del «plugin». La función está optimizada para trabajar con polígonos grandes y utiliza el siguiente flujo de trabajo. Para una geometría dada, encontrar un `bbox` y extraer subset ráster usando este `bbox`. Efectuar la rasterización de la geometría en memoria y aplicar como una máscara al ráster. A continuación, se cuenta el número de píxeles que tienen un valor específico y se calcula el porcentaje. Puesto que el recuento de píxeles se basa en funciones de matriz de `numpy`, es muy rápido incluso para grandes polígonos.

En el primer intento de editar una capa vectorial, el usuario se encontrará con un diálogo en el que deberá seleccionar los conjuntos de datos que se utilizarán para los indicadores. A continuación, el «plugin» configurará valores de expresión por defecto para todos los campos de los indicadores, de modo que el valor se actualizará en cada cambio de geometría.

1.4 Gestión de los metadatos de los conjuntos de datos

Los metadatos del conjunto de datos se almacenan en el formato QMD de QGIS. Estos archivos QMD pueden crearse para cada trama individualmente y también para todo el conjunto de datos. El diálogo del editor de metadatos se abre desde el menú **Editar metadatos** en el «dock» de Trends.Earth.

Cuando el conjunto de datos se exporta a ZIP, la conversión a ISO XML se realiza mediante una transformación XLST. Las transformaciones correspondientes se encuentran en el subdirectorio `data\xsl` de la carpeta de instalación del «plugin».

1.5 Actualización del marco de información

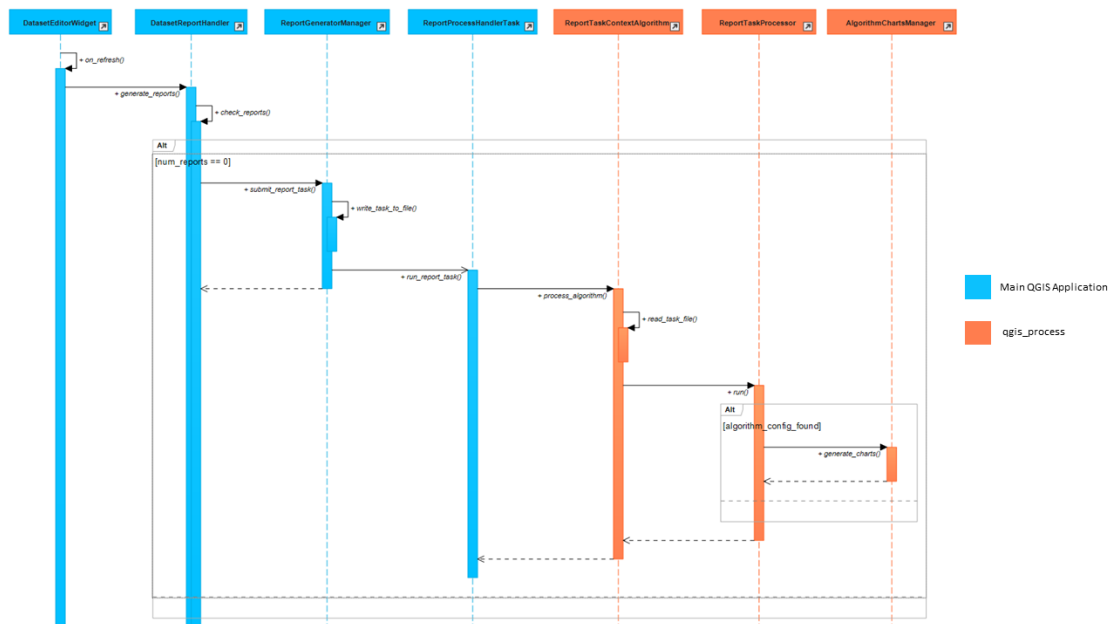
1.5.1 Resumen del marco de información

El marco de trabajo de los informes está diseñado para ser extensible mientras que también proporciona interactividad al usuario a través de operaciones no bloqueantes. El marco se apoya en las clases `QgsProject` y `QgsPrintLayout` que no son seguras para los hilos, de ahí el uso de `qgis_process` para hacer el trabajo pesado de generar informes (y listas). Puede encontrar más información sobre `qgis_process` [aquí](#).

Hay dos pasos principales que la caja de herramientas realiza al generar informes (y listas) para las capas por defecto de un trabajo:

1. Crea un objeto `ReportTaskContext` que compone un objeto `ReportConfiguration` (ver `config_report_params`) y un objeto `Job` que se representa en el panel **Datasets**. Este objeto `ReportTaskContext` se convierte en un archivo JSON y luego se pasa como uno de los argumentos de un objeto `ReportProcessHandlerTask` (que hereda de `QgsTask`).
2. El objeto `ReportProcessHandlerTask` inicia una instancia separada de `qgis_process` y pasa la ruta del archivo JSON como una entrada al algoritmo de procesamiento `trendsearch:reporttask`. Se trata de una fina envoltura que deserializa el archivo al objeto `ReportTaskContext` y lo pasa a un objeto `ReportTaskProcessor` que es responsable de generar los informes y el trabajo del proyecto QGIS. Para algoritmos que requieren listas, el objeto `ReportTaskProcessor` pasa el objeto de un trabajo a uno `AlgorithmChartsManager` que comprueba si hay una configuración de lista definida para el algoritmo del trabajo. Si está definida, genera las listas correspondientes como archivos PNG. (Véase [Añadir configuraciones de listas](#) para más información sobre las configuraciones de listas)

El siguiente diagrama ilustra a grandes rasgos este proceso:



* *Clique en la imagen para hacerla más grande.*

Nota: Algunos de los nombres de las funciones en el diagrama anterior se han simplificado con fines ilustrativos. Las clases mencionadas se encuentran en los módulos `LDMP.reports` y `LDMP.processing_provider.report` <https://github.com/ConservationInternational/trends.earth/tree/master/LDMP/processing_provider/report.py>`_.

1.5.2 Añadir variables de diseño del informe

Las variables del informe proporcionan información de contexto relacionada con un trabajo, capa (o banda) o `report_settings` durante el proceso de ejecución del informe. Actualmente, la caja de herramientas admite las variables enumeradas en la sección `layout_expr_vars`.

Cada variable se define como un `namedtuple` en el módulo `LDMP.reports.expressions` y, seguidamente, es actualizada y evaluada por el objeto `ReportTaskProcessor`.

Siga las siguientes pautas sobre cómo añadir nuevas variables de trabajo o de capa actual.

Variable de trabajo

Permite añadir información sobre el trabajo actual (que se está ejecutando) a un diseño de informe. La información sobre cada variable de trabajo se encapsula en un objeto `:code:`JobAttrVarInfo`` que se compone de cuatro atributos:

Nombre del atributo	Descripción	Tipo de datos	Valor por defecto
<code>job_attr</code>	El nombre del atributo de un objeto <code>Job</code> utilizado en una notación de puntos. Por ejemplo, <code>id</code> corresponde a <code>job.id</code> . Incluso se puede utilizar la notación de puntos para referirse a los atributos en las clases anidadas internas, p. ej., <code>results.uri.uri</code> .	Cadena	N/A
<code>var_name</code>	Nombre de la variable de diseño del informe. Debe llevar el prefijo <code>te_job_</code> .	Cadena	N/A
<code>default_val</code>	Valor por defecto para <code>nombre_de_var</code> , que se aplica sobre todo al crear diseños.	objeto	objeto
<code>fmt_func</code>	Un objeto de función que se utilizará para convertir el valor del atributo del trabajo a un formato que sea compatible con las expresiones QGIS. Por ejemplo, <code>str</code> puede utilizarse para convertir el valor del <code>id</code> del trabajo de UUID a una cadena. Aquí también se pueden utilizar funciones lambda.	objeto de la función	Ninguno

El fragmento de código siguiente muestra cómo añadir una variable `te_job_result_name` que corresponde a `job.results.name`.

```
# LDMP/reports/expressions.py
def _job_attr_var_mapping() -> typing.List[JobAttrVarInfo]:
    return [
        ...
        JobAttrVarInfo('results.name', 'te_job_result_name', '', str),
        ...
    ]
```

Capa variable

Ofrece información sobre la actual capa ráster que está siendo ejecutada. Esta información variable se encapsula en un objeto `LayerVarInfo` que está compuesto por tres atributos:

Nombre del atributo	Descripción	Tipo de datos	Valor por defecto
<code>var_name</code>	El nombre de la variable de diseño del informe. Debe llevar el prefijo <code>te_current_layer_.</code>	Cadena	N/A
<code>default_value</code>	El valor por defecto para <code>nombre_de_var</code> , que se aplica sobre todo al crear diseños.	objeto	objeto
<code>fmt_func</code>	Un objeto de función que se utilizará para extraer o convertir un valor de un objeto <code>QgsRasterLayer</code> a un formato que sea compatible con expresiones QGIS. También puede utilizar funciones lambda aquí. Por ejemplo, <code>lambda layer: layer.name()</code> devuelve el nombre de la capa.	objeto de la función	Ninguno

El fragmento de código siguiente muestra cómo añadir una variable `te_current_layer_height` que corresponde a la altura de la capa del ráster.

```
# LDMP/reports/expressions.py
def _current_job_layer_var_mapping() -> typing.List[LayerVarInfo]:
    return [
        ...
        LayerVarInfo(
            'te_current_layer_height',
            '',
            lambda layer: layer.height()
        )
        ...
    ]
```

Nota: Estas variables están únicamente disponibles en el marco del diseño.

1.5.3 Añadir configuraciones de listas

Las listas se pueden agrupar usando un objeto de configuración de lista que corresponde a un algoritmo específico. La definición de una nueva configuración de lista es un proceso de tres pasos:

1. Crear una nueva clase de lista que herede de `BaseChart` en el módulo `LDMP.reports.charts`. Implementar la función `export` para especificar el tipo de lista, propiedades, etc. usando la biblioteca de Python `Plotly` que viene con QGIS. Finalmente, dentro de la función `export`, hay que llamar a la función `save_image` para escribir el objeto `Figure` de `Plotly` como un archivo de imagen utilizando cualquiera de los formatos soportados por la clase `QImageWriter` de Qt. También se puede especificar la ruta como relativa al directorio raíz de salida, el cual está disponible como un atributo en la clase base. Véase el siguiente fragmento de código:

```
# LDMP/reports/charts.py
class MyCustomChart(BaseChart):
    def export(self) -> typing.Tuple[bool, list]:
        status = True
        messages = []

        # Create chart Figure using Plotly and set properties
        fig = go.Figure(...)

        # Add warning or error messages
        messages.append('Colour list not supported.')

        # Set image path in dataset's reports folder
```

(continué en la próxima página)

(proviene de la página anterior)

```

img_path = f'{self.root_output_dir}/chart-NDVI.png'

# Save image and append its path
self.save_image(fig, img_path)
self._paths.append(img_path)

return status, messages

```

Puede consultar la clase `UniqueValuesPieChart` para ver un ejemplo más completo.

2. Crear una clase de configuración de lista que herede de `BaseAlgorithmChartsConfiguration` e implemente la función `BaseAlgorithmChartsConfiguration`. La clase de configuración de lista básicamente define qué listas se utilizarán para un algoritmo determinado. El atributo `layer_band_infos` es una lista de objetos `LayerBandInfo` que contiene los datos de capa y `band_info` necesarios para producir las listas. Puede consultar la clase `LandCoverChartsConfiguration` para ver un ejemplo más completo.
3. Por último, asigne un algoritmo (nombre) a la clase de configuración de la lista correspondiente en la clase `AlgorithmChartsManager` tal y como se muestra a continuación:

```

# LDMP/reports/charts.py
Class AlgorithmChartsManager:
    def _set_default_chart_config_types(self):
        ...
        self.add_alg_chart_config('land-cover', LandCoverChartsConfiguration)
        self.add_alg_chart_config('productivity', ←
↪MyCustomLandProductivityChartsConfiguration)
        ...

```

La clase `AlgorithmChartsManager`, que se instancia en el objeto `ReportTaskProcessor`, creará un nuevo objeto de configuración de listas para el algoritmo de un trabajo correspondiente cuando se creen informes.

1.6 Contribuyendo a la documentación

1.6.1 Visión general

La documentación para Trends.Earth se produce usando `Sphinx`, y está escrito en `reStructuredText` <http://docutils.sourceforge.net/rst.html> formato. Si no está familiarizado con ninguna de estas herramientas, consulte su documentación para obtener más información sobre cómo se utilizan.

La documentación de Trends.Earth se almacena en la carpeta `"docs"` en el directorio principal de `trends.earth`. Dentro de esa carpeta hay una serie de archivos y carpetas clave a tener en cuenta:

- `build`: contiene la documentación de construcción de `trends.earth` (en formato PDF y HTML). Tenga en cuenta que solo aparecerá en su máquina después de ejecutar la tarea de invocación `docs-build`.
- `i18n`: contiene traducciones de la documentación a otros idiomas. Los archivos aquí normalmente se procesan automáticamente mediante tareas de invocación, por lo que nunca debería tener motivos para modificar nada en esta carpeta.
- `recursos`: contiene todos los recursos (principalmente imágenes o PDF) a los que se hace referencia en la documentación. Actualmente solo hay una carpeta («EN», para inglés) ya que todas las imágenes en la documentación son de la versión en inglés del complemento, si se pueden agregar carpetas adicionales apropiadas en «recursos»; con códigos de idioma de dos letras para incluir imágenes específicas de un idioma en particular.
- `fuentes`: contiene los archivos fuente `reStructuredText` que definen la documentación (estos son el texto real en inglés de la documentación y son los archivos que es más probable que necesite modificar).

1.6.2 Instalar dependencias

Dependencias de Python

Para trabajar con la documentación, debe tener `invoke`, `Sphinx`, `sphinx-intl` y `sphinx-rtd-theme` (el tema para el sitio web Trends.Earth) instalado en su máquina. Todos estos paquetes se enumeran en el archivo de requisitos `requirements-dev.txt` para Trends.Earth, por lo que se pueden instalar navegando en un símbolo del sistema a la raíz de la carpeta de códigos de `tendencias.earth` y escribiendo:

```
pip install -r requirements-dev.txt
```

LaTeX

LaTeX se utiliza para producir salidas en PDF de la documentación para Trends.Earth.

Para instalar en Windows, [siga el proceso descrito aquí](http://ftp.math.purdue.edu/mirrors/ctan.org/systems/windows/protext/) para instalar la distribución `proTeXt` de LaTeX desde el archivo zip disponible aquí [<http://ftp.math.purdue.edu/mirrors/ctan.org/systems/windows/protext/>](http://ftp.math.purdue.edu/mirrors/ctan.org/systems/windows/protext/). El instalador de LaTeX es bastante grande (varios GB), por lo que puede llevar un tiempo descargarlo e instalarlo.

En MacOS, `MacTeX` es una buena opción y se puede instalar [siguiendo las instrucciones aquí](#).

En Linux, la instalación de LaTeX debería ser mucho más fácil: use el administrador de paquetes de su distribución para encontrar e instalar cualquier distribución de LaTeX que se incluya por defecto.

Qt Linguist

Qt Linguist también es necesario para extraer cadenas del código y la GUI para la traducción. El comando `lrelease` debe estar disponible y en su camino. Intente probar:

```
lrelease
```

dentro de una ventana de terminal. Si no se encuentra el archivo, deberá instalar Qt Linguist. [Esta página](#) es una fuente de instaladores para Qt Linguist. Una vez que instale Qt Linguist, asegúrese de agregar la carpeta que contiene `lrelease` a su ruta para que el script de invocación Trends.Earth pueda encontrarla.

1.6.3 Actualización y construcción de la documentación.

Una vez que haya instalado los requisitos de `sphinx`, estará listo para comenzar a modificar la documentación. Los archivos a modificar se encuentran en la carpeta `docs\source`. Después de realizar cualquier cambio en estos archivos, deberá compilar la documentación para ver los resultados. Existen dos versiones de la documentación de Trends.Earth: una versión HTML (utilizada para el sitio web) y una versión PDF (para descargar sin conexión). Para compilar la documentación de Trends.Earth, use la tarea de invocación `docs-build`. De forma predeterminada, esta tarea generará la documentación completa para Trends.Earth, en HTML y PDF, para todos los idiomas admitidos. Esto puede tardar un tiempo en ejecutarse (hasta unas pocas horas). Si solo está probando los resultados de algunos cambios menores en la documentación, generalmente es mejor usar la opción `-f` (para «rápido»). Esta opción construirá solo la documentación HTML en inglés, que debería tomar solo unos segundos. Para construir usando la opción rápida, ejecute:

```
invoke docs-build -f
```

El comando anterior tardará unos segundos en ejecutarse, y luego si busca en `docs\build\html\en`, verá la versión HTML de la documentación. Cargue el archivo `index.html` en un navegador web para ver cómo se ve.

Para compilar la documentación completa, para todos los idiomas, en PDF y en HTML (recuerde que esto puede tardar algunas horas en completarse), ejecute:

```
invoke docs-build
```

Después de ejecutar el comando anterior, verá (para inglés) la documentación HTML en «docs\build\html\en», y los PDF de la documentación en «docs\build\html\en\pdfs».

Si desea probar un idioma específico (al probar traducciones, por ejemplo), puede especificar un código de idioma de dos letras para crear solo los documentos para ese idioma. Por ejemplo, para construir solo la documentación en español, ejecute:

```
invoke docs-build -l es
```

Tenga en cuenta que las opciones se pueden combinar, por lo que puede usar la opción rápida para crear solo la versión HTML de la documentación en español ejecutando:

```
invoke docs-build -f -l es
```

Al compilar la documentación completa para el sitio web, es una buena idea eliminar primero las compilaciones antiguas de la documentación, ya que pueden contener archivos que ya no se usan en la documentación actualizada. Para hacer esto, use la opción `-c` (limpiar):

```
invoke docs-build -c
```

En general, docs-build DEBE completarse sin ningún error si planea compartir la documentación o publicarla en el sitio web. Sin embargo, al probar cosas localmente, es posible que desee ignorar los errores de documentación que aparecen solo para algunos de los idiomas (debido a errores de sintaxis que surgen de errores de traducción, etc.) y continuar construyendo la documentación restante independientemente de si hay algún error. Para hacer esto, use la opción `-i` (ignorar errores):

```
invoke docs-build -i
```

Siempre que realice cambios en el texto de la documentación, es una buena idea enviar las últimas cadenas a Transifex para que puedan traducirse. Para actualizar las cadenas en Transifex con cualquier cambio nuevo, ejecute:

```
invoke translate-push
```

Nota: Para ejecutar con éxito el comando anterior, necesitará tener la clave para la cuenta Transifex de Trends.Earth.

1.6.4 Documentación de construcción para lanzamiento

Antes de lanzar nueva documentación, siempre obtenga las últimas traducciones de Transifex para que todas las traducciones estén actualizadas. Para hacer esto, ejecute:

```
invoke translate-pull
```

Para compilar una versión de la documentación para publicación pública (ya sea en el sitio web o en PDF), debe compilar toda la documentación utilizando docs-build sin parámetros adicionales:

```
invoke docs-build
```

Este proceso debe completarse con éxito sin errores. Si se produce algún error durante el proceso, revise el mensaje de error y realice las modificaciones necesarias para permitir que la compilación se complete con éxito. Una vez que la compilación se completa sin errores, los archivos están listos para implementarse en el sitio web.

Nota: Los dos comandos anteriores también tienen opciones `-f` (forzar) que fuerzan a tirar o empujar las últimas traducciones desde o hacia Transifex (respectivamente). Solo use estas opciones si está MUY seguro de lo que está haciendo, ya que pueden sobrescribir completamente las traducciones en Transifex, lo que lleva a la pérdida del trabajo realizado por los traductores si las últimas traducciones aún no se han comprometido con github.

1.6.5 Agregar nuevo texto de documentación

Todos los archivos `.rst` nuevos que se agreguen a la documentación deben agregarse a varios archivos de configuración para asegurarse de que aparezcan en el menú de navegación, que estén traducidos correctamente y (para tutoriales) que se generen en PDF para que puedan ser descargados para su uso sin conexión.

- `docs\source\index.rst`: agregue nuevos archivos `.rst` en el lugar apropiado aquí para asegurarse de que estén vinculados desde el menú de navegación.
- `.tx\config`: enumere los nuevos archivos `.rst` aquí (en el mismo formato que los otros archivos ya incluidos) para que el software de traducción los conozca y pueda traducirlos
- `docs\source\conf.py`: si desea generar un archivo PDF de la página del sitio web, debe incluir esa página aquí en la lista `latex_documents`. Por lo general, hacemos esto solo para las páginas de tutoriales que queremos poner a disposición de los participantes del taller en archivos PDF individuales. Cada página del sitio se incluirá en la versión PDF del sitio web en su conjunto, independientemente de si está en la lista de documentos de `latex`.

1.6.6 Agregar nuevas imágenes u otros recursos

Cualquier nueva imagen u otro recurso (PDF, etc.) que necesite la documentación debe agregarse en «`docs\resources\en`». Si lo desea, es posible cargar diferentes versiones de una imagen para que la imagen aparezca con las traducciones adecuadas. Esto podría ser útil si desea mostrar la interfaz GUI en el idioma apropiado, por ejemplo. para hacer esto, primero suba una copia de la imagen a «`docs\resources\en`» (con texto en inglés). Luego, cree una copia de la imagen con el texto traducido y colóquela en la carpeta correspondiente a ese idioma (por ejemplo, una imagen que muestre traducciones al español iría en «`docs\resources\es`»). La versión en inglés de la imagen se utilizará como predeterminada para todos los idiomas para los que no se proporciona una versión nativa de la imagen, mientras que se utilizará una versión localizada cuando esté disponible.

Nota: Hay otra carpeta, `docs\source\static`, que se usa para contener recursos temporalmente mientras se ejecutan los scripts que compilan la documentación de Trends.Earth. Es posible que tenga imágenes en esa carpeta si alguna vez ha creado la documentación en esa máquina. **Esta carpeta nunca debe usarse para agregar nuevos recursos** - los nuevos recursos siempre deben ir bajo `docs\resources\en` o, para las imágenes traducidas, la carpeta específica del idioma correspondiente bajo `docs\recursos`.

1.6.7 Contribuyendo como traductor

Transifex gestiona las traducciones para el complemento QGIS y también para este sitio. Si desea contribuir a traducir el complemento y la documentación (¡y nos encantaría contar con su ayuda!), Puede solicitar unirse a nuestro equipo a través de [transifex](https://www.transifex.com/), o enviándonos un correo electrónico a trends.earth@conservation.org.

Esta página muestra el historial de versiones de [▲TRENDS.EARTH](#).

2.1 2.1.8 (January 25, 2023)

- Output JSON field for Prais containing updated false positive/negative calculations.
- Fixes #752 - false positive/negative datasets not showing up in non-English versions

2.2 2.1.6 (December 8, 2022)

- Bumps maximum number of land cover classes to 45 (useful particularly for users seeking to use Corine legend)
- Fix issue that was arising on Mac/Linux due to missing QgsPanelWidget ui file
- Various more minor bug fixes

2.3 2.1.6 (November 21, 2022)

- Fixes #691 (issue that arose if the base data directory was cleared). When base directory is cleared plugin will now reset the value to the default.
- Fix python error coming up when settings window was closed while datasets screen was not open.

2.4 2.1.2 (November 18, 2022)

- Fixes #724 - ensure max number of land cover classes is set to 38 by default (ignoring the previously set default value in qsettings, which used to be 32).
- Fixes #722 - file name casing when code refers to land_cover_transition_matrix_unccd.json.
- Fixes #726 - make dataset year visible in datasets window for imported datasets

2.5 2.1.0 (November 11, 2022)

- Moved plugin settings to live within main QGIS plugin settings window
- Bumped marshmallow-dataclass to 8.5.10 and marshmallow to 3.18.0. This necessitated changes in trends.earth-schemas, so it is possible some old jobs may not load in this new version (suggest re-running them if need be).

2.6 2.0.7 (October 31, 2022)

- Enable Excel table generation for SDG 15.3.1 when using custom classes - all the various minor bugs should now be worked out.
- Limit report generation to 3 tries - then quit trying to generate reports until next restart (prevents Trends.Earth continually spawning processes for report generation when they are failing).

2.7 2.0.5 (October 19, 2022)

- Fix SDG summary generation when using JRC LPD or WOCAT LPD.

2.8 2.0.3 (October 19, 2022)

- Allow bypassing reset of land cover legend when choosing defaults in the SDG tool.
- Disable excel generation when child legend does not have 7 classes.
- Fix report generation when using custom legend.

2.9 2.0.1 (October 13, 2022)

- Added ability to completely customize the land cover legend used in Trends.Earth, including the number, name, color, and coding of each class in the legend. This applies to all calculatons using land cover data, including the SDG 15.3.1 tool, and the productivity, soil organic carbon, and land cover indicator tools.
- Reworked land cover import tool to allow importing data and assigning classes to custom legend.
- Misc minor bug fixes.
- Add ability to include false positive/negative dataset in UNCCD report. Next release will include further functionality to apply and report on false positive/negatives.

- This experimental release disables the creation of Excel output tables for the SDG 15.3.1 tool. This functionality will be present in the next (stable) release.

2.10 2.0 (July 20, 2022)

- Interfaz totalmente nueva con pantallas de navegación y carga de conjuntos de datos simplificadas
- Documentación y sitio web actualizados
- Trends.Earth está ahora traducido a todos los idiomas oficiales de la ONU, así como al portugués. Está previsto traducir también el swahili y el farsi (para la nueva interfaz).
- Nuevas funciones para apoyar el último ciclo de presentación de informes de la UNCCD (incluido el apoyo directo a la exportación de los resultados para cargarlos en el sistema PRAIS de la UNCCD)
- Nuevas funciones para apoyar la evaluación del peligro de sequía, la vulnerabilidad y la exposición
- Nuevas funciones de la herramienta de series de tiempo (ahora admite el trazado de tendencias, WUE, etc.)

2.11 1.0.10 (7 de julio de 2022)

- Se soluciona un error con TransformDirection relacionado con los cambios en QGIS 3.22+

2.12 1.0.8 (15 de octubre de 2021)

- Solucionar un error al guardar la capa final para el cálculo del SDG (relacionado con los problemas #500, y #505)

2.13 1.0.6 (15 de julio de 2021) <<https://github.com/ConservationInternational/trends-earth>>

- Elimine tendencias.earth-schemas como un submódulo e instálelo a través de setup.py (mucho más limpio para el seguimiento de versiones, desarrollo, etc.)
- Corrección de error al cargar datos NDVI 2020 en código GEE (el nombre del activo no se configuró correctamente, por lo que 2020 no se estaba cargando)
- Se agregaron varias tareas de invocación para ayudar con el desarrollo/lanzamiento del complemento.

2.14 1.0.4 (30 de junio de 2021) <<https://github.com/ConservationInternational/trends-earth>>

- Agregue WorldPop y Gridded Population of the World versión 4 (GPWv4) a los conjuntos de datos disponibles a través de Trends.Earth
- Actualización para permitir el acceso a la cobertura terrestre de ESA-CCI hasta 2020
- Actualizar conjuntos de datos de GPCC y GPCP
- Corrección de errores cuando se utilizan búferes no enteros

- Corrección de errores tipográficos menores en la documentación

2.15 `1.0.2 (14 de agosto de 2020) <<https://github.com/ConservationInternational/trends.earth/releases/tag/1.0.2>>`

- Corregir el código de área urbana para permitir el procesamiento de AOI con áreas entre 10 000 km² y 25 000 km².
- Agregue los últimos datos de MERRA2 (hasta 2019).
- Eliminar la limitación de área máxima de la herramienta de descarga.
- Mejorar httpLib2 a 0.18.0.
- Actualización de GPCC V6 a GPCC V7
- Añadir 2019 Hansen et al. datos de deforestación
- Actualizar al mapa de colores más reciente para los datos de degradación (abordar problemas con el daltonismo regular/verde).
- Agregue soporte para más tipos de datos en los archivos de forma de entrada (agregue PointZ, MultiPoint, MultiPointZ, PolygonZ, MultiPolygonZ).
- Varias correcciones de errores para abordar los errores de Python que aparecían con algunos mensajes de QMessageBox.

2.16 `1.0.0 (27 de abril de 2020) <<https://github.com/ConservationInternational/trends.earth/releases/tag/1.0.0>>`

- Agregar capacidad de descargar y usar binarios precompilados (compilados con Numba) para acelerar algunos cálculos locales. En este momento, esto solo está disponible para el cálculo de la tabla de resumen SDG15.3.1, pero eventualmente también se ampliará a otras herramientas.
- En relación con lo anterior, ahora hay un botón de configuración «avanzada» en la ventana de configuración, que permitirá a los usuarios descargar binarios precompilados y activar o desactivar el registro detallado de mensajes mientras se ejecuta la herramienta. Estos mensajes de registro pueden ser útiles al intentar solucionar problemas si encuentra alguno.
- Mejore las verificaciones de la validez de la geometría para las geometrías en los archivos de entrada y envíe un mensaje de error en lugar de lanzar una excepción cuando haya errores de geometría.
- Se corrigió el procesamiento utilizando nuevos datos MODIS (los archivos no se actualizaron correctamente en la última versión)
- Se corrigió la codificación por colores de las cubiertas terrestres en la ventana de agregación de clases de cubierta vegetal
- Agregue información de versión más detallada sobre el cuadro de diálogo.
- Agregue detalles adicionales a la herramienta de descarga de datos.
- Agregue ID de trabajo a la ventana de descargas para facilitar la notificación de errores.
- Pequeñas correcciones de errores (clasificación de trabajos y tablas de descargas).

2.17 0.98 (2 de abril de 2020)

- Primera versión de QGIS3: muchas correcciones para actualizar a Qt5 y QGIS3 API.
- Actualización de todas las dependencias del complemento a las últimas versiones a enero de 2020.
- Se corrigió la herramienta de descarga de datos para tener estilos predeterminados para todos los conjuntos de datos disponibles.
- Se comenzaron a pasar a los marcos QgsProcessing y QgsTask: actualmente solo se ha migrado la herramienta de carbono, pero todas las herramientas se migrarán antes de la versión 1.0.
- Formateo de la hoja de cálculo de salida de la herramienta de carbono para aclarar el significado de cada columna.
- Actualización de todos los scripts de GEE para usar la última versión de la API de GEE (0.1.213).
- Se guardan más configuraciones elegidas en cuadros de diálogo de herramientas en las sesiones de QGIS.
- Limpieza del código de búfer, para usar proyecciones de área Lambert Equal Area centradas en centroides poligonales para el almacenamiento en búfer.
- Mueva la documentación a la carpeta docs en la raíz del repositorio de trends.earth.
- Agregue más detalles sobre cómo documentar cómo contribuir al desarrollo de Trends.Earth,
- Limpie el repositorio eliminando los archivos de traducción compilados y agregando estos tipos de archivos a .gitignore.
- Cambie el nombre del proyecto transifex para que sea «trendsearth».
- Varias mejoras en compatibilidad y correcciones de errores menores.

2.18 0.66 (20 de julio de 2019)

- Limite el área máxima para tareas a 10,000,000 km², excepto para tareas de área urbana, que está limitada a 10,000 km².
- Agregue la sección de fondo en ODS 11.3.1 y actualice el tutorial ODS 11.3.1.
- Actualice el código ODS 11.3.1 para incluir 1998 en la serie (internamente durante el cálculo) para filtrar el ruido desde el principio de la serie urbana.
- Fije las restricciones de fecha en la herramienta todo en uno ODS 15.3.1 para tener en cuenta la disponibilidad de ESA y MODIS.
- Agregue una sección a Léame sobre cómo instalar las versiones de Github.
- Actualice y revise las traducciones al español, actualice las traducciones de google para otros idiomas.

2.19 0.64 (9 de julio de 2019)

- Arreglo de manejo de nodata en herramienta de carbono total.
- Agregue soporte para los datos de Hansen 2018 en la herramienta de carbono total.
- Agregue soporte para datos globales de biomasa de 30m de Wood'S Hole en carbono total
- Establezca el año final máximo para que la herramienta ODS 15.3.1 de un solo paso sea 2015 (coincida con los datos de la ESA).

- Haga de Trends.Earth productividad el conjunto de datos predeterminado en la herramienta de un paso de ODS para 15.3.1.

2.20 0.62 (27 de enero de 2019)

- Agregue una herramienta experimental para mapear los rendimientos potenciales de carbono de las intervenciones de restauración alternativas.
- Añadir 2018 datos MODIS.
- Correcciones diversas al tamaño de la ventana para ventanas GUI.
- Actualice a la última versión de openpyxl - corrige la carga del logotipo de Trends.Earth en las tablas de resumen.
- Añadir lista de publicaciones a documentos de ayuda.

2.21 0.60 (3 de diciembre de 2018)

- Agregue el cálculo del cambio en el área urbana y la tasa de crecimiento de la población (ODS 11.3.1)
- Fijar alturas predeterminadas de botones / campos de entrada Agregar selección de ciudad para AOI
- Añadir búfer opcional de AOI

2.22 0.58 (11 de agosto de 2018)

- Añadir una sección de prueba a la página de cálculos.
- Agregue la versión de prueba de carbono total (por encima y por debajo del suelo) y las emisiones debidas a la deforestación
- Correcciones de errores menores, incluso para polígonos no válidos en entradas AOI

2.23 0.56.5 (21 de mayo de 2018)

- Solucione el error con la importación de LPD solicitando un año de datos.

2.24 0.56.4 (21 de mayo de 2018)

- Siempre vuelva a muestrear los datos importados a la resolución más alta.
- Arregle las zonas climáticas de importación de SOC personalizadas para usar un conjunto de datos de zonas climáticas expandido para eliminar ningún dato.
- Actualiza MOD16A2 con los últimos datos.
- Forzar la entrada de la fecha en la importación de datos SOC y LC
- Agregue los resultados globales de Trends.Earth a la herramienta de descarga.
- Arreglar el manejo de valores NULL en las leyendas.

2.25 0.56.3 (21 de abril de 2018)

- Corrija el cálculo de las tablas de resumen para los AOI que se dividen en el meridiano 180 (Fiji, Rusia, etc.).
- Modifique el cálculo del estado para que las áreas con cambios de magnitud muy pequeños en la integral NDVI (< .01 unidades NDVI durante todo el período) se consideren estables.

2.26 0.56.2 (10 de abril de 2018)

- Correcciones menores de Unicode.

2.27 0.56.1 (10 de abril de 2018)

- Solucionar error de malvavisco en la carga del complemento

2.28 0.56 (9 de abril de 2018)

- Solucionar el problema con rasterizar datos (rásteres vacíos en la salida)
- Forzar al usuario a elegir la resolución de salida si rasteriza un vector
- Cálculo de soporte de la degradación de SOC a partir de datos de SOC y LC personalizados

2.29 0.54 (8 de abril de 2018)

- Admite la carga de datos personalizados de LPD, SOC y LC.
- Estilos de limpieza para que coincidan con maps.trends.earth
- Actualizar pyopenxl
- Añadir iconos de importación / carga a todos los cuadros de selección de capas

2.30 0.52.1 (21 de marzo de 2018)

- Corrección de errores menores durante el taller de Antalya.

2.31 0.52.1 (21 de marzo de 2018)

- Corrección de errores menores durante el taller de Antalya.

2.32 0.52 (19 de marzo de 2018)

- Código de procesamiento AOI limpio.

2.33 0.50 (15 de marzo de 2018)

- Pase la excepción si solo está relacionada con la adición del logotipo de Trends.Earth en un archivo Excel.
- Varias correcciones de errores menores.

2.34 0.48 (13 de marzo de 2018)

- Formato de tabla de arreglos

2.35 0.46 (13 de marzo de 2018)

- Soporte para el cálculo de tablas de informes con múltiples geometrías (Fiji, Rusia)
- Agregue las tablas LPD y LC a la pestaña de la hoja de trabajo UNCCD
- Limpie el mensaje de advertencia en la herramienta de importación LPD
- Fijar la carga final de la capa de productividad combinada de TE
- Reparar tareas de descarga (aún sin estilos)

2.36 0.44 (12 de marzo de 2018)

- Añadir JRC LPD
- Agregar herramienta para cargar datos de cobertura terrestre personalizados
- Agregar herramienta para cargar datos de productividad personalizados
- Añadir nota de que la subida de SOC personalizada vendrá pronto
- Agregar herramienta para agregar mapas base usando datos de Natural Earth
- Agregue la herramienta todo en uno para calcular los tres subindicadores a la vez
- Cambie el nombre de la clase «Tierras desnudas» como «Otras tierras» para mantener la coherencia con la CNUCLD
- Actualizar documentos
- Actualizar a Marshmallow 3.0.0b7
- Mueva el código GEE al repositorio de trends.earth principal

- Mejore el manejo de los AOI, particularmente cuando se usan shapefiles para entrada
- Maneje salidas de múltiples archivos de GEE agrupándolas en VRTs
- Soporte de datos de procesamiento para países que cruzan el meridiano 180
- Mejorar el formato de la tabla de resumen.
- A partir de ahora, las versiones del script GEE se combinarán con la versión del complemento.

2.37 0.42 (4 de febrero de 2018)

- Arreglar la falla en el cambio de la agregación de LC (se debe establecer el conjunto en la etiqueta eliminada)

2.38 0.40 (4 de febrero de 2018)

- Eliminar el uso del modo para el indicador de cobertura terrestre.
- Combine la tabla de resumen y las herramientas de creación de mapas de indicadores ODS.
- Agregue el talón donde estará disponible el producto JRC LPD.
- Guarde el subindicador de productividad como banda 2 en el archivo indicador ODS.
- Bump script de GEE a v0.3.
- Se corrige el error debido a la división por cero en la generación de la tabla de resumen cuando una clase tiene un área cero.
- Predeterminado a MODIS para cálculos de productividad.

2.39 0.38 (16 de enero de 2018)

- Agregue el cálculo anual de carbono orgánico del suelo.
- El código de procesamiento de AOI de limpieza permite varios polígonos de entrada en los AOI de shapefile
- Agregar sombreado al lado de los elementos de la tabla de agregación de cobertura de tierra
- Solucionar el problema de FirstShow en la tabla de agregación
- Revise el resultado de la tabla de resumen para proporcionar más información sobre cada uno de los tres indicadores
- Agregue conjuntos de datos complementarios al rendimiento, estado, cobertura del suelo y producción de carbono orgánico del suelo.
- No actualice los datos y los valores de enmascaramiento para que consistentemente sean -32768 (sin datos) y -32767 (datos enmascarados)
- Permitir el nombramiento de descargas de archivos
- Añadir icono al menú de la barra de herramientas, corregir el nombre del complemento.
- Código de diseño de la capa de refactor para extraer información de la banda de la salida de GEE.
- Agregue una herramienta para cargar los conjuntos de datos de trends.earth existentes en QGIS.
- Fije los límites de fecha de cobertura del suelo: no permita que se seleccionen fechas no válidas a partir de los datos de CCI.

2.40 0.36 (14 de diciembre de 2017)

- Solucionar el problema con showEvent en la herramienta de creación de informes de mapas.

2.41 0.34 (14 de diciembre de 2017)

2.42 0.32 (14 de diciembre de 2017)

2.43 0.30 (12 de diciembre de 2017)

2.44 0.24 (6 de diciembre de 2017)

2.45 0.22 (4 de diciembre de 2017)

2.46 0.18 (2 de diciembre de 2017)

2.47 0.16 (6 de noviembre de 2017)

2.48 0.14 (25 de octubre de 2017)

2.49 0.12 (6 de octubre de 2017)