

Trends.Earth - Guia do desenvolvedor

version 2.3.6

Conservation International

October 09, 2026

Contents

Guia dos Programadores	1
Desenvolvimento	1
Modificando o código QGIS Plugin	1
Modificando o código de processamento do Earth Engine	5
A editar modelos de camada vetorial	6
Gestão dos metadados do conjunto de dados	8
A atualizar estrutura de relatório	8
Contribuindo para a documentação	11
Instalando dependências	12
Dependências do Python	12
Lançamento de uma nova versão do plugin	15
Sumário	15
Changelog	17
2.3.6 (October 9, 2026)	17
2.3.4 (September 29, 2026)	17
2.3.2 (July 2, 2026)	18
2.3.0 (May 12, 2026)	18
2.2.10 (April 20, 2026)	18
2.2.8 (April 17, 2026)	19
2.2.6 (April 13, 2026)	19
2.2.4 (March 19, 2026)	19
2.2.2 (28 de Janeiro de 2026)	19
2.2.0 (24 de Novembro de 2025)	20
2.1.20 (10 de Outubro de 2025)	20
2.1.18 (5 de Agosto de 2025)	21
2.1.16 (31 de Julho de 2024)	21
2.1.14 (24 de Fevereiro de 2023)	21
2.1.12 (17 de Fevereiro de 2023)	22
2.1.10 (7 de Fevereiro de 2023)	22
2.1.8 (25 de Janeiro de 2023)	22
2.1.6 (8 de Dezembro de 2022)	22
2.1.6 (21 de Novembro de 2022)	22
2.1.2 (18 de Novembro de 2022)	22
2.1.0 (11 de Novembro de 2022)	23
2.0.7 (31 de Outubro de 2022)	23
2.0.5 (19 de Outubro de 2022)	23
2.0.3 (19 de outubro de 2022)	23
2.0.1 (13 de outubro de 2022)	23
2.0 (20 de julho de 2022)	24
1.0.10 (7 de julho de 2022)	24

1.0.8 (15 de outubro de 2021)	24
1.0.6 (15 de julho de 2021)	24
1.0.4 (30 de junho de 2021)	24
1.0.2 (14 de agosto de 2020)	24
1.0.0 (27 de abril de 2020)	25
0.98 (2 de abril de 2020)	25
0.66 (20 de julho de 2019)	26
0.64 (9 de julho de 2019)	26
0.62 (27 de janeiro de 2019)	26
0.60 (3 de dezembro de 2018)	27
0.58 (11 de agosto de 2018)	27
0.56.5 (21 de maio de 2018)	27
0.56.4 (21 de maio de 2018)	27
0.56.3 (21 de abril de 2018)	27
0.56.2 (10 de abril de 2018)	28
0.56.1 (10 de abril de 2018)	28
0.56 (9 de abril de 2018)	28
0.54 (8 de abril de 2018)	28
0.52.1 (21 de março de 2018)	28
0.52.1 (21 de março de 2018)	28
0.52 (19 de março de 2018)	28
0.50 (15 de março de 2018)	28
0.48 (13 de março de 2018)	28
0.46 (13 de março de 2018)	29
0.44 (12 de março de 2018)	29
0.42 (4 de fevereiro de 2018)	29
0.40 (4 de fevereiro de 2018)	29
0.38 (16 de janeiro de 2018)	30
0.36 (14 de dezembro de 2017)	30
0.34 (14 de dezembro de 2017)	31
0.32 (14 de dezembro de 2017)	31
0.30 (12 de dezembro de 2017)	31
0.24 (6 de dezembro de 2017)	31
0.22 (4 de dezembro de 2017)	31
0.18 (2 de dezembro de 2017)	31
0.16 (6 de novembro de 2017)	31
0.14 (25 de outubro de 2017)	31
0.12 (6 de outubro de 2017)	31

Guia dos Programadores

Desenvolvimento

▲TRENDS.EARTH é um software livre e de código aberto, licenciado sob a [GNU General Public License, version 2.0 or later](#).

Há vários componentes na ferramenta ▲TRENDS.EARTH. O primeiro é um plugin QGIS que suporta cálculos de indicadores, acesso a dados brutos, relatórios e produção de mapas impressos. O código para o plugin e mais instruções sobre como instalá-lo, se você deseja modificar o código, estão em [trends.earth](#) Repositório no GitHub.

O plugin do QGIS ▲TRENDS.EARTH é suportado por diversos scripts de Python que permitem o cálculo dos vários indicadores no Google Earth Engine (GEE). Estes scripts encontram-se na subpasta «gee» desse repositório do GitHub.

O plugin é também suportado por uma série de outros módulos:

- O módulo *trends.earth-algorithms* inclui o código para o processamento de entradas e saídas do plugin, bem como outras funções comuns que suportam o cálculo de integrais de NDVI, significância estatística e outro código partilhado. O código deste módulo está disponível no repositório [landdegradation](#) no GitHub.
- O módulo *trends.earth-schemas* inclui o código para gerir os esquemas utilizados na entrada e saída de dados do trends.earth, incluindo a gestão de classes de ocupação do solo, parâmetros de tarefas, estruturação de relatórios para a UNCCD e outras funções relacionadas

Apresentam-se abaixo mais detalhes sobre como contribuir para o Trends.Earth, trabalhando no código da interface gráfica do plugin, modificando o código de processamento ou contribuindo para a tradução do website e do plugin.

Modificando o código QGIS Plugin

Fazendo o download do código trends.earth

O código Trends.Earth para os scripts do plugin e do Google Earth Engine que o suporta está localizado no GitHub na seção [trends.earth](#) repositório. Clone este repositório em um local conveniente em sua máquina para garantir que você tenha a versão mais recente do código.

Existem vários ramos diferentes do repositório trends.earth em desenvolvimento ativo. O plug-in suporta oficialmente o QGIS 3 e a maior parte do desenvolvimento está a ocorrer no ramo «develop». O ramo «qgis 2» é a versão mais antiga do plug-in e suporta versões do QGIS 2 a partir da 2.18.

Instalando dependências

Python

The plugin is coded in Python. In addition to being used to run the plugin through QGIS, Python is also used to support managing the plugin (changing the version, installing development versions, etc.). Though Python is included with QGIS, you will also need a local version of Python that you can setup with the software needed to manage the plugin. The easiest way to manage multiple versions of Python is through the [Anaconda distribution](#). For work developing the plugin, Python 3.12 or newer is required, both locally and in the QGIS runtime. To download Python through Anaconda, [see this page](#).

Dependências do Python

Para trabalhar com o código trends.earth, é necessário ter o Invoke instalado em sua máquina, além de vários outros pacotes usados para gerenciar a documentação, traduções, etc. Esses pacotes estão todos listados no arquivo «dev» de requisitos para Trends.Earth, para que eles possam ser instalados navegando em um prompt de comando até a raiz da pasta de código trends.earth e digitando:

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

Note

If you are using Anaconda, you will first want to activate a Python 3.12 or newer virtual environment before running the above command (and any of the other invoke commands listed on the page). One way to do this is by starting an «Anaconda prompt», by [following the instructions on this Anaconda page](#).

PyQt

PyQt5 é o kit de ferramentas gráficas usado pelo QGIS3. Para compilar a interface do usuário do Trends.Earth for QGIS3, você precisa instalar o PyQt5. Este pacote pode ser instalado a partir do pip usando:

```
pip install PyQt5
```

Note

PyQt4 é o kit de ferramentas gráficas usado pelo QGIS2. A melhor fonte para este pacote para Windows é o conjunto de pacotes mantidos por Christoph Gohlke na UC Irvine. Para baixar o PyQt4, selecione [o pacote apropriado nesta página](#). Escolha o arquivo apropriado para a versão do Python que você está usando. Por exemplo, se você estiver usando o Python 2.7, escolha a versão com «cp27» no nome do arquivo. Se você estiver usando o Python 3.7, escolha a versão com «cp37» no nome do arquivo. Escolha «amd64» para python de 64 bits e «win32» para python de 32 bits.

Após fazer o download no link acima, use pip para instalá-lo. Por exemplo, para a roda de 64 bits para Python 3.7, você executaria:

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

Alterando a versão do plugin

A convenção para Trends.Earth é que os números de versão que terminam em um número ímpar (por exemplo, 0,65) são versões de desenvolvimento, enquanto as versões que terminam em um número par (por exemplo 0,66) são versões de lançamento. As versões de desenvolvimento do plug-in nunca são liberadas via o repositório QGIS, para que eles nunca sejam vistos pelos usuários normais do plug-in. Versões de desenvolvimento com números ímpares são usadas pela equipe de desenvolvimento do Trends.Earth ao testar novos recursos antes de seu lançamento público.

Se você deseja fazer alterações no código e baixou uma versão pública do plugin (que termina em um número par), a primeira etapa é atualizar a versão do plugin para o próximo número ímpar seqüencial. Portanto, por exemplo, se você baixou a versão 0.66 do plugin, seria necessário atualizar a versão para 0.67 antes de começar a fazer as alterações. Existem vários locais no código em que a versão é mencionada (assim como em todos os scripts GEE), portanto, há uma tarefa de chamada para ajudar na alteração da versão. Para alterar a versão para 0,67, você deve executar:

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

A execução do comando acima atualizará o número da versão em todos os locais mencionados no código. Para evitar confusão, nunca mude a versão para uma que já foi lançada - sempre AUMENTE o valor da tag da versão para o próximo número ímpar.

Testando alterações no plugin

Depois de fazer alterações no código do plugin, você precisará testá-los para garantir que o plugin se comporte conforme o esperado e para garantir que não haja bugs ou erros. O plugin deve passar por testes extensivos antes de ser lançado no repositório QGIS (onde pode ser acessado por outros usuários) para garantir que quaisquer alterações no código não quebrem o plugin.

Para testar as alterações feitas no plugin no QGIS, será necessário instalá-lo localmente. Existem tarefas de chamada que auxiliam nesse processo. A primeira etapa antes da instalação do plug-in é garantir que você o tenha configurado com todas as dependências necessárias para executar no QGIS. Para fazer isso, execute:

```
invoke plugin-setup
```

A tarefa acima só precisa ser executada imediatamente após o download do código trends.earth ou se forem feitas alterações nas dependências do plugin. Por padrão, o `plugin-setup` reutiliza todos os arquivos em cache na sua máquina. Para começar do zero, adicione o sinalizador `-c` (clean) ao comando acima.

Após executar o `plugin-setup`, você está pronto para instalar o plugin na pasta de plug-ins do QGIS em sua máquina. Para fazer isso, execute:

```
invoke plugin-install
```

Depois de correr o comando em cima, precisará de ou 1) reiniciar o QGIS, ou 2) usar o [Plugin Reloader](#) para recarregar o plug-in Trends.Earth, de forma a ver os efeitos das alterações que fez.

Por padrão, `plugin-install` substituirá os arquivos de plugin existentes em sua máquina, mas deixará em vigor quaisquer dados (limites administrativos etc.) que o plugin possa ter baixado. Para começar do zero, adicione o sinalizador `-c` (clean) ao comando acima. Pode ser necessário fechar o QGIS para executar com êxito uma instalação limpa do plugin usando o sinalizador `-c`.

Note

Por padrão, a instalação assume que você deseja instalar o plugin a ser usado no QGIS3. Para instalar o plug-in para uso no QGIS3, adicione o sinalizador `-v 2` ao comando `plugin-install`.

Lembre-se de que o plugin pode ou não ser totalmente funcional no QGIS3 - o plugin foi originalmente projetado para o QGIS2 e ainda está sendo testado no QGIS3.

A atualizar a lista de limites em cache

Quando o conjunto de dados geoBoundaries for atualizado, atualize a lista em cache incluída no plugin executando:

```
invoke download-boundaries-cache
```

A tarefa efetua a autenticação na API do Trends.Earth. Configure as credenciais adicionando um ficheiro `invoke.yaml` na raiz do repositório com entradas como:

```
trends_earth_api:
  user: "you@example.com"
  password: "your-password"
```

Também pode fornecer as credenciais através das variáveis de ambiente `TRENDS_EARTH_API_USER` e `TRENDS_EARTH_API_PASSWORD` antes de invocar a tarefa.

Construindo um arquivo ZIP contendo o plugin

Existem várias tarefas de chamada para ajudar na criação de um arquivo ZIP para implantar o plugin no repositório QGIS ou para compartilhar a versão de desenvolvimento do plugin com outras pessoas. Para empacotar o plugin e todas as suas dependências em um arquivo ZIP que pode ser instalado seguindo o [procedimento descrito no Leiamos do Trends.Earth](#), execute:

```
invoke zipfile-build
```

Este comando criará uma pasta chamada `build` na raiz da pasta de código `trends.earth`, e nessa pasta criará um arquivo chamado `LDMP.zip`. Este arquivo pode ser compartilhado com outras pessoas, que podem usá-lo para [instalar manualmente o Trends.Earth](#). Isso pode ser útil se houver a necessidade de compartilhar os recursos mais recentes com alguém antes que eles estejam disponíveis na versão divulgada publicamente do plugin.

Implementando o Arquivo ZIP da versão em desenvolvimento

A página do Trends.Earth no GitHub disponibiliza uma ligação para um ficheiro ZIP que permite aos utilizadores que não sejam programadores aceder à versão de desenvolvimento do Trends.Earth. Para criar um ficheiro ZIP e disponibilizá-lo nessa página (o ficheiro ZIP é armazenado no S3), execute:

```
invoke zipfile-deploy
```

Este comando irá empacotar o plugin e copiá-lo para <https://s3.amazonaws.com/trends.earth/sharing/LDMP.zip>.

Note

O comando acima falhará se você não tiver chaves que permitam acesso de gravação ao bucket `trends.earth` no S3.

Modificando o código de processamento do Earth Engine

Os scripts de processamento do Google Earth Engine (GEE) usados pelo Trends.Earth são todos armazenados na pasta «gee», na pasta principal trends.earth. Para que esses scripts sejam acessíveis aos usuários do plugin trends.earth QGIS, eles precisam ser implantados no serviço api.trends.earth que a Conservação Internacional mantém para permitir que os usuários do plugin usem o Earth Engine sem a necessidade de saber como programar ou ter contas de usuário individuais no GEE. A seguir, descrevemos como testar e implantar scripts GEE para serem usados com o Trends.Earth.

Configurando dependências

trends.earth-CLI

O pacote Python «trends.earth-CLI» é necessário para poder trabalhar com o servidor api.trends.earth. Este pacote encontra-se no GitHub, no repositório [trends.earth-CLI](#).

O primeiro passo é clonar este repositório para a sua máquina. Recomendamos que clone o repositório para a mesma pasta onde tem o código do trends.earth. Por exemplo, se tiver uma pasta «Code» na sua máquina, clone tanto o repositório [trends.earth](#) (o código para o plugin do QGIS e os scripts do GEE associados) como também o repositório [trends.earth-CLI](#) para essa mesma pasta.

Quando você configura seu sistema como recomendado acima, o trends.earth-CLI funcionará com as tarefas de chamada usadas para gerenciar o trends.earth sem nenhuma modificação. No entanto, se você baixar o trends.earth-CLI em uma pasta diferente, será necessário adicionar um arquivo chamado «invoke.yaml» na raiz do repositório trends.earth e, nesse arquivo, informe ao Trends.Earth onde localizar o código trends.earth-CLI. Esse arquivo YAML deve ter a seguinte aparência (se você baixou o código no Windows em uma pasta chamada «C:/Users/azvol/Code/trends.earth-CLI/tecli»):

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

Novamente, você **não** precisa adicionar esse arquivo .yaml se configurar o sistema conforme recomendado acima.

docker

O pacote trends.earth-CLI requer [docker](#) para funcionar. [Siga estas instruções para instalar a janela de encaixe no Windows](#) e` estas instruções para instalar a janela de encaixe no Mac OS [<https://docs.docker.com/docker-for-mac/install/>](https://docs.docker.com/docker-for-mac/install/)`. Se você estiver executando o Linux, [siga as instruções nesta página](#) que são apropriados para a distribuição Linux que você está usando.

Testando um script do Google Earth Engine localmente

Depois de instalar o pacote trends.earth-CLI, você precisará configurar um arquivo .tecli.yml com um token de acesso a uma conta de serviço GEE para testar scripts no GEE. Para configurar a conta de serviço GEE para o tecli, primeiro obtenha a chave da sua conta de serviço no formato JSON (no console do Google Cloud) e, em seguida, codifique-a em base64. Forneça a chave codificada base64 ao tecli com o seguinte comando:

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

onde «chave» é a chave da conta de serviço no formato JSON codificado em base64.

Ao converter um script que especifica o código a ser executado no GEE de JavaScript para Python, ou ao fazer modificações nesse código, pode ser útil testar o script localmente, sem implementá-lo no servidor api.trends.earth. Para fazer isso, use a tarefa de chamada run. Por exemplo, para testar o script «land_cover», vá para o diretório raiz do código Trends.Earth e, em um prompt de comando, execute:

```
invoke tecli-run land_cover
```

Isso usará o pacote trends.earth-CLI para criar e executar um contêiner de docker que tentará executar o script «land_cover». Se houver algum erro de sintaxe no script, eles aparecerão quando o contêiner for executado. Antes de enviar um novo script para api.trends.earth, sempre verifique se `invoke tecli-run` é capaz de executar o script sem erros.

Ao usar `invoke tecli-run`, você pode receber um erro dizendo:

```
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 erro pode ser causado se o relógio no contêiner do docker ficar fora de sincronia com o relógio do sistema. Reiniciar docker deve corrigir esse erro.

Implantando um script GEE em api.trends.earth

Quando você terminar de testar um script GEE e desejar acessá-lo usando o plugin QGIS (e por outros usuários do Trends.Earth), poderá implantá-lo no servidor api.trends.earth. A primeira etapa do processo é efetuar login no servidor api.trends.earth. Para fazer login, execute:

```
invoke tecli-login
```

Você será solicitado a fornecer um nome de usuário e senha. Estes são os mesmos usuário e a senha que você usa para fazer login no Trends.Earth a partir do plugin QGIS. **Se você não for um administrador, poderá fazer login, mas o comando abaixo falhará.** Para fazer upload de um script (por exemplo, o script «land_cover») para o servidor, execute:

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

Se esse script já existir no servidor, você será perguntado se deseja substituir o script existente. Tenha muito cuidado ao fazer o upload de scripts com versões pares, pois esses são scripts publicamente disponíveis, e quaisquer erros que você cometer afetarão qualquer pessoa que esteja usando o plugin. Sempre que você estiver testando, use números de versão de desenvolvimento (números de versão ímpares).

Depois de publicar um script no servidor, você pode usar a tarefa *tecli-info* para verificar o status do script (para saber se ele foi implementado com sucesso - embora a criação do script possa demorar alguns minutos). Para verificar o status de um script implantado, execute:

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

Se você estiver fazendo uma nova versão do plugin e quiser fazer o upload de TODOS os scripts GEE de uma só vez (isso é necessário sempre que o número da versão do plugin for alterado), execute:

```
invoke tecli-publish
```

Novamente - nunca execute o procedimento acima em uma versão divulgada publicamente do plugin, a menos que você pretenda substituir todos os scripts publicamente disponíveis usados pelo plugin.

A editar modelos de camada vetorial

O Trends.Earth permite aos utilizadores digitalizar novos elementos vetoriais para delinear áreas de especial interesse.

Para já, apenas as camadas de «falso positivo/negativo» são suportadas, mas poderão ser adicionadas mais, se necessário. Qualquer camada vetorial é criada a partir dos ficheiros de modelo GeoPackage, que

podem ser encontrados na pasta `data/error_recode` do diretório de instalação do plugin. Para cada tipo de vetor, existem 6 ficheiros de modelo, um para cada língua oficial da ONU. O código de língua ISO é adicionado como um sufixo ao nome do ficheiro. Isto é necessário para fornecer etiquetas localizadas nos formulários de atributos. Quando a criação da camada vetorial é solicitada, o QGIS procurará o ficheiro de modelo tendo em conta o idioma do QGIS; como opção de recurso, será utilizada a versão em inglês do ficheiro de modelo.

Para alterar o esquema da camada, é necessário modificar os ficheiros de modelo correspondentes na pasta `data/error_recode` do diretório de instalação do plugin. Além disso, o ficheiro de modelo contém uma estilização predefinida incorporada e uma configuração de formulário de atributos que serão aplicadas automaticamente à camada quando esta for carregada no QGIS.

Para apresentar gráficos no formulário atributo, é usado um widget QML integrado. Os dados para os gráficos estão guardados na tabela de atributos da camada vetorial. Os valores dos campos correspondentes são extraídos com a ajuda de expressões.

O código para gerar gráficos tem este aspeto:

```
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
    }

    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: "#ffff00"; values: [expression.evaluate("\prod_stab"), expression.evaluate("\land_stab"), expression.evaluate("\soil_stab")] }
    }
}
```

Para extrair o valor do campo, é usada a função `expression.evaluate("\prod_deg")`, o único argumento que aceita é o nome do campo. Para camadas de falsos positivos/negativos, o gráfico contém três indicadores, produtividade, cobertura de solo e carbono orgânico do solo. Para cada indicador, o plug-in mantém três valores da área do polígono: estável, degradado e melhorado. Por exemplo, no caso do indicador produtividade, os campos serão:

- `prod_deg` - produtividade degradada
- `prod_stab` - produtividade estável
- `prod_imp` - produtividade melhorada

A mesma abordagem de nomenclatura é aplicada à cobertura do solo (campos `land_*`) e ao carbono orgânico do solo (campos `soil_*`).

O cálculo da percentagem de área é efetuado com uma função de expressão personalizada, cujo código pode ser encontrado no ficheiro `charts.py`, no diretório raiz do plugin. A função está otimizada para trabalhar com polígonos de grandes dimensões e utiliza o seguinte fluxo de trabalho: para uma determinada geometria, define uma caixa envolvente e extrai um subconjunto do raster utilizando essa `bbox`. Realiza a rasterização da geometria em memória e aplica-a como uma máscara ao raster. Em seguida, conta o número de píxeis que possuem um valor específico e calcula a percentagem. Como a contagem de píxeis se baseia em funções de matrizes `numpy`, o processo é muito rápido, mesmo para polígonos grandes.

Na primeira tentativa de editar uma camada vetorial, será apresentado um diálogo ao utilizador, onde este deverá escolher os conjuntos de dados a usar para os indicadores. Depois o plug-in irá configurar os valores de expressão por defeito para todos os campos indicadores, para que o valor seja atualizado em cada alteração de geometria.

Gestão dos metadados do conjunto de dados

Os metadados dos conjuntos de dados são armazenados no formato QGIS QMD. Estes ficheiros QMD podem ser criados para cada matriz individualmente e também para todo o conjunto de dados. O diálogo do editor de metadados é aberto a partir do menu **Editar metadados** na plataforma Trends.Earth.

Quando o conjunto de dados é exportado para ZIP, a conversão para ISO XML é realizada usando transformação XSLT. As transformações correspondentes estão localizadas na sub-pasta `data\xsl` da pasta de instalação do plug-in.

A atualizar estrutura de relatório

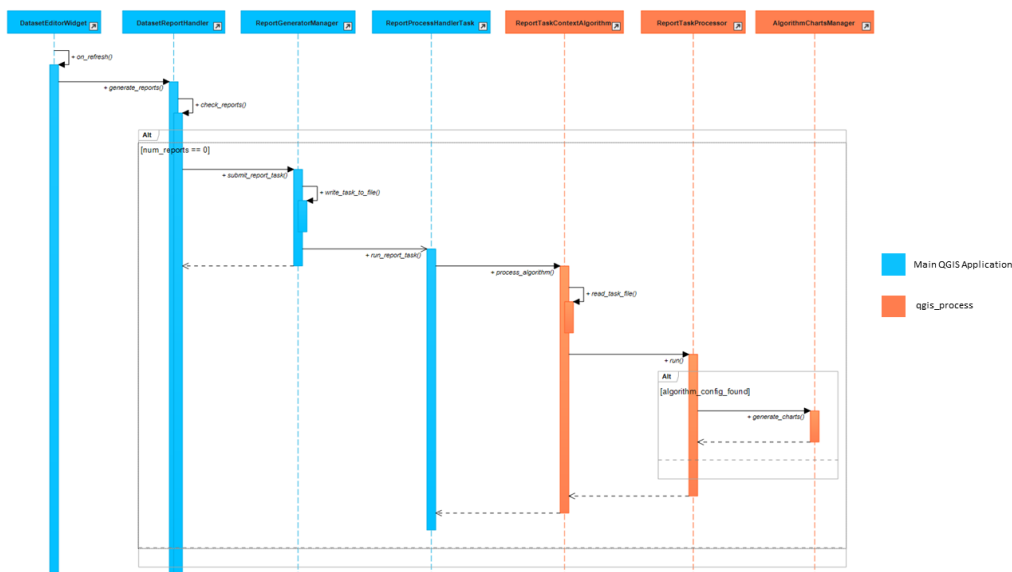
Resumo da estrutura de relatório

A Estrutura de Relatórios foi concebida para ser extensível, proporcionando ao mesmo tempo interatividade ao utilizador, através de operações sem bloqueio. A Estrutura baseia-se fortemente nas classes `QgsProject` e `QgsPrintLayout`, que não são seguras, daí a utilização de `qgis_process` para fazer o trabalho pesado de gerar os relatórios (e gráficos). Pode encontrar mais informação sobre `qgis_process` [aqui](#).

A caixa de ferramentas executa dois passos principais quando gera relatórios (e gráficos) para as camadas por defeito num trabalho:

1. Cria um objeto `ReportTaskContext` que constitui um objeto `ReportConfiguration` (ver: [ref:config_report_params](#)) e um objeto `Job` que é representado no painel de **Conjuntos de dados**. Este objeto `ReportTaskContext` é serializado para um ficheiro JSON e depois passado como um dos argumentos num objeto `ReportProcessHandlerTask` (que herda de `QgsTask`).
2. O objeto `ReportProcessHandlerTask` inicia uma instância separada de `qgis_process` e passa o caminho para o ficheiro JSON como uma entrada para o algoritmo de processamento `trendsearth:reporttask`. Este é um wrapper fino que desserializa o ficheiro para o objeto `ReportTaskContext` e o passa para um objeto `ReportTaskProcessor`, que é responsável por gerar os relatórios e o projeto QGIS do trabalho. Para algoritmos que requerem gráficos, o objeto `ReportTaskProcessor` passa o objeto do trabalho para um objeto `AlgorithmChartsManager` que verifica se existe uma configuração de gráficos definida para o algoritmo do trabalho. Se existir, gera os gráficos correspondentes como ficheiros PNG. (Ver Adicionar Configurações de Gráfico para mais informações sobre configurações de gráficos)

O diagrama em baixo apresenta uma ilustração de alto nível deste processo:



* Clique na imagem para uma vista ampliada.

Note

Alguns dos nomes de funções no diagrama em cima foram simplificados para fins ilustrativos. As classes cima mencionadas podem ser encontradas nos módulos [LDMP.reports](#) e [LDMP.processing_provider.report](#).

A adicionar variáveis de layout do relatório

As variáveis do relatório fornecem informações de contexto relacionadas com um trabalho, camada (ou banda) ou **report_settings** durante o processo de execução do relatório. Atualmente, a caixa de ferramentas suporta as variáveis listadas na secção Variáveis de Expressão de Esquema.

Cada variável é definida como uma `namedtuple` no módulo [LDMP.reports.expressions](#) e é subsequentemente atualizada e avaliada pelo objeto `ReportTaskProcessor`.

Siga as orientações em baixo sobre como adicionar novas variáveis de trabalho ou de camada atual.

Variável de trabalho

Permite que a informação sobre o trabalho atual - a ser executado - seja adicionada ao layout de um relatório. A informação sobre cada variável de trabalho é encapsulada num objeto `JobAttrVarInfo` que tem quatro atributos:

Nome de atributo	Descrição	Tipo de dados	Valor por defeito
<i>job_attr</i>	Nome de atributo de um objeto <code>Job</code> , como usado numa notação dot. Por exemplo, <i>id</i> corresponde a <code>job.id</code> . Pode até usar a notação dot para referir-se a atributos em classes internas de cadeias, por exemplo, <i>results.uri.uri</i> .	String	N/D
<i>var_name</i>	Nome da variável de layout do relatório. Deve ter o prefixo <i>te_job_</i> .	String	N/D
<i>default_value</i>	Valor por defeito a usar para <i>var_name</i> , aplicado principalmente ao desenhar layouts.	objeto	objeto
<i>fmt_func</i>	Um objeto de função que será usado para converter o valor do atributo do trabalho para um formato compatível com expressões QGIS. Por exemplo, <i>str</i> pode ser usado para converter o valor do <i>id</i> de um trabalho, de UUID para string. Também pode usar funções lambda aqui.	objeto de função	Nenhum

O fragmento de código em baixo mostra como adicionar uma variável *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),
```

```
    ...
]
```

Variável de Camada

Fornece informação sobre a camada raster atual a ser executada. Esta informação de variável está encapsulada num objeto `LayerVarInfo` que consiste em três atributos:

Nome de atributo	Descrição	Tipo de dados	Valor por defeito
<code>var_name</code>	Nome do variável de esquema de relatório of the report layout variable. Deve ter como prefixo <code>te_current_layer_</code> .	String	N/D
<code>default_value</code>	Valor por defeito a usar para <code>var_name</code> , aplicado principalmente ao desenhar layouts.	objeto	objeto
<code>fmt_func</code>	Um objeto de função que será utilizado para extrair e/ou converter um valor de um objeto <code>QgsRasterLayer</code> para um formato compatível com expressões QGIS. Pode utilizar funções lambda aqui. Por exemplo, <code>lambda layer: layer.name()</code> devolve o nome da camada.	objeto de função	Nenhum

O fragmento de código em baixo mostra como adicionar uma variável `te_current_layer_height` que corresponde à altura do camada raster.

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

Note

Estas variáveis estão apenas disponíveis no âmbito do esquema.

Adicionar Configurações de Gráfico

Os gráficos podem ser agrupados utilizando o objeto de configuração de gráficos que corresponde a um algoritmo específico. A definição de uma nova configuração de gráficos é um processo com três passos:

1. Crie uma nova classe de gráficos que herda de `BaseChart` no módulo `LDMP.reports.charts`. Implemente a função `export` para especificar o tipo de gráfico, propriedades, etc utilizando a biblioteca Python `Plotly` incluída com o QGIS. Por fim, dentro da função `export`, chame a função `save_image` para escrever o código de objeto `Figure Plotly` como um ficheiro de imagem, utilizando qualquer um dos formatos suportados pela classe `QImageWriter` de Qt. Também pode especificar o

caminho em relação com diretório de resultado de raiz, que também está disponível como um atributo na classe base. Veja o fragmento de código em baixo:

```
# 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
        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
```

Pode consultar a classe `UniqueValuesPieChart` para um exemplo mais completo.

2. Crie uma classe configuração de gráfico `chart configuration class` que herda de `BaseAlgorithmChartsConfiguration` e implemente a função `_add_charts`. A classe de configuração de gráfico define basicamente quais os gráficos que serão utilizados para um dado algoritmo. O atributo `layer_band_infos` é uma lista de objetos `LayerBandInfo` que contém a camada e dados `band_info` necessários para produzir os gráficos. Pode consultar a classe `LandCoverChartsConfiguration` para um exemplo mais completo.
3. Por fim, mapeie uma algoritmo (nome) à classe de configuração de gráficos correspondente na classe `AlgorithmChartsManager` tal como apresentado em baixo:

```
# 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)
        ...
```

A classe `AlgorithmChartsManager`, que é instanciada no objeto `ReportTaskProcessor`, irá criar um novo objeto de configuração de gráficos para o algoritmo de um trabalho correspondente quando os relatórios estiverem a ser gerados.

Contribuindo para a documentação

Sumário

A documentação para Trends.Earth é produzida usando o [Sphinx](#), e está escrito em `reStructuredText` <<http://docutils.sourceforge.net/rst.html>> formato. Se você não for familiarizado com nenhuma dessas ferramentas, consulte a documentação para obter mais informações sobre como elas são usadas.

A documentação do Trends.Earth é armazenada na pasta «docs», no diretório principal `trends.earth`. Dentro dessa pasta, há vários arquivos e pastas importantes que você deve conhecer:

- **build:** contém a documentação da build para trends.earth (no formato PDF e HTML). Observe que ele só aparecerá na sua máquina depois de executar a tarefa de chamada docs-build.
- **i18n:** contém traduções da documentação para outros idiomas. Os arquivos aqui são normalmente processados automaticamente usando tarefas com o comando «invoke», portanto, você nunca deve ter motivos para modificar nada nesta pasta.
- **recursos:** contém todos os recursos (principalmente imagens ou PDFs) mencionados na documentação. Atualmente, existe apenas uma pasta («EN», em inglês), pois todas as imagens na documentação são da versão em inglês do plugin - pastas adicionais apropriadas podem ser adicionadas em «resources» com códigos de duas letras identificando o idioma para incluir imagens específicas para um idioma específico.
- **source:** contém os arquivos de origem reStructuredText que definem a documentação (esse é o texto em inglês real da documentação e os arquivos que você provavelmente precisará modificar).

Instalando dependências

Dependências do Python

Para trabalhar com a documentação, você deve ter os pacotes invoke, Sphinx, sphinx-intl e sphinx-rtd-theme (o tema do site Trends.Earth) instalados em sua máquina. Esses pacotes estão todos listados no arquivo de requisitos «dev» do Trends.Earth, para que possam ser instalados navegando em um prompt de comando até a raiz da pasta de código trends.earth e digitando:

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

LaTeX

O LaTeX é usado para produzir saídas em PDF da documentação para Trends.Earth.

Para instalar LaTeX no Windows, [siga o processo descrito aqui](#) para instalar a distribuição proTeXt do LaTeX a partir do [arquivo zip disponível aqui](#). O instalador do LaTeX é bastante grande (vários GB), portanto, pode levar algum tempo para baixar e instalar.

No MacOS, o MacTeX é uma boa opção e pode ser instalado [seguindo as instruções aqui](#).

No Linux, a instalação do LaTeX deve ser muito mais fácil - use o gerenciador de pacotes da sua distribuição para encontrar e instalar qualquer distribuição do LaTeX incluída por padrão.

Qt Linguist

O Qt Linguist também é necessário para extrair strings do código e da GUI para tradução. O comando «lrelease» deve estar disponível e no seu caminho. Tente tentar

```
lrelease
```

dentro de uma janela do terminal. Se o arquivo não for encontrado, você precisará instalar o Qt Linguist. [Esta página](#) é uma fonte de instaladores para o Qt Linguist. Depois de instalar o Qt Linguist, adicione a pasta que contém lrelease ao seu caminho para que o script de chamada Trends.Earth possa encontrá-lo.

Atualizando e preparando a documentação

Depois de instalar os requisitos do pacote sphinx, você estará pronto para começar a modificar a documentação. Os arquivos a serem modificados estão localizados na pasta «docs\source». Depois de fazer alterações nesses arquivos, você precisará criar a documentação para visualizar os resultados. Existem duas versões da documentação do Trends.Earth: uma versão HTML (usada para o site) e uma versão PDF (para download offline). Para criar a documentação do Trends.Earth, use a tarefa «docs-build» do invoke. Por padrão, esta tarefa cria a documentação completa do Trends.Earth, em HTML e PDF, para todos os idiomas suportados. Isso pode levar algum tempo para ser executado (até algumas horas). Se você está apenas testando os resultados de algumas pequenas alterações na documentação, geralmente é melhor usar a opção -f (de «fast» = rápido). Esta opção criará apenas a documentação HTML em inglês, que deve levar apenas alguns segundos. Para criar usando a opção rápida, execute:

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

O comando acima levará alguns segundos para ser executado e, se você procurar em «docs\build\html\en», verá a versão HTML da documentação. Carregue o arquivo «index.html» em um navegador da Web para ver como fica.

Para criar a documentação completa, para todos os idiomas, em PDF e HTML (lembre-se de que isso pode levar algumas horas para ser concluído), execute:

```
invoke docs-build
```

Após executar o comando acima, você verá (em inglês) a documentação HTML em «docs\build\html\en», e os PDFs da documentação em «docs\build\html\en\pdfs».

Se você quiser testar um idioma específico (testar traduções, por exemplo), poderá especificar um código de duas letras identificando o idioma para criar apenas os documentos para esse idioma. Por exemplo, para criar apenas a documentação em espanhol, execute:

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

Observe que as opções podem ser combinadas; portanto, você pode usar a opção rápida para criar apenas a versão HTML da documentação em espanhol executando:

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

Ao criar a documentação completa para o site, é uma boa idéia remover primeiro todas as versões antigas da documentação, pois elas podem conter arquivos que não são mais usados na documentação atualizada. Para fazer isso, use a opção -c (clean):

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

Em geral, a preparação dos documentos DEVE ser concluído sem erros se você planeja compartilhar a documentação ou publicá-la no site. No entanto, ao testar localmente, convém ignorar os erros de documentação que aparecem somente em alguns idiomas (devido a erros de sintaxe decorrentes de erros de tradução etc.) e continuar a compilar a documentação restante, independentemente de haver erros. Para fazer isso, use a opção -i (ignorar erros):

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

Sempre que você fizer alterações no texto da documentação, é uma boa ideia enviar as últimas atualizações para o Transifex, para que possam ser traduzidas. Para atualizar as seqüências de caracteres no Transifex com novas alterações, execute:

```
invoke translate-push
```

Note

Para executar com êxito o comando acima, você precisará da chave da conta transifex Trends.Earth.

Preparando a documentação para publicar a nova versão

Antes de liberar a nova documentação, sempre baixe as últimas traduções do Transifex para que todas as traduções estejam atualizadas. Para fazer isso, execute:

```
invoke translate-pull
```

Para criar uma versão pública da documentação (no site ou em PDF), você deve compilar toda a documentação usando `docs-build` sem parâmetros adicionais:

```
invoke docs-build
```

Esse processo deve ser concluído com êxito, sem erros. Se ocorrer algum erro durante o processo, revise a mensagem de erro e faça as modificações necessárias para permitir que a compilação seja concluída com êxito. Depois que a compilação é concluída sem erros, os arquivos estão prontos para serem implementados no site.

Note

Ambos os comandos acima também têm opções `-f` (force) que forçam a baixar ou subir as últimas traduções do ou para o Transifex (respectivamente). Use essas opções apenas se tiver MUITO certo do que está fazendo, pois elas podem substituir completamente as traduções no Transifex, resultando em perda de trabalho dos tradutores se as traduções mais recentes ainda não foram submetidas para o repositório github.

Adicionando novo texto de documentação

Quaisquer novos arquivos `.rst` adicionados à documentação precisam ser adicionados a vários arquivos de configuração para garantir que apareçam no menu de navegação, que sejam traduzidos corretamente, e (para tutoriais) para garantir que sejam gerados em PDF para que possam ser baixados para uso offline.

- `docs\source\index.rst`: adicione novos arquivos `.rst` no local apropriado aqui para garantir que eles estejam vinculados no menu de navegação.
- `.tx\config`: lista os novos arquivos `.rst` aqui (no mesmo formato dos outros arquivos já incluídos) para tornar o software de tradução ciente deles, para que possam ser traduzidos
- `docs\source\conf.py`: se você deseja gerar um arquivo PDF da página do site, deve listar essa página na lista `latex_documents`. Normalmente, fazemos isso apenas para as páginas de tutoriais que queremos disponibilizar aos participantes do workshop em PDFs individuais. Todas as páginas do site serão incluídas na versão PDF do site como um todo, independentemente de estar na lista `latex_documents`.

Adicionando novas imagens ou outros recursos

Quaisquer novas imagens ou outros recursos (PDFs etc.) necessários à documentação devem ser adicionados em «docs\resources\en». Se desejar, é possível fazer upload de versões diferentes de uma imagem para que a imagem apareça com as traduções adequadas. Isso pode ser útil se você quiser mostrar a interface da GUI no idioma apropriado, por exemplo. Para fazer isso, primeiro faça upload de uma cópia da imagem para «docs\resources\en» (com o texto em inglês). Em seguida, crie uma cópia da imagem com texto traduzido e coloque-a na pasta apropriada para esse idioma (por exemplo, uma imagem mostrando traduções em espanhol seria exibida em «docs\resources\es»). A versão em inglês da imagem será usada como padrão para todos os idiomas para os quais uma versão nativa da imagem não é fornecida, enquanto uma versão localizada será usada quando disponível.

Note

Existe outra pasta, docs\source\static, que é utilizada para guardar recursos temporariamente durante a execução dos scripts que compilam a documentação do Trends.Earth. Poderá ter imagens listadas nessa pasta se alguma vez tiver compilado a documentação nessa máquina.** Esta pasta nunca deve ser utilizada para adicionar novos recursos** — os novos recursos devem ser sempre colocados em docs\resources\en ou, no caso de imagens traduzidas, na pasta específica do idioma correspondente em docs\resources.

Contribuindo como tradutor

As traduções para o plugin QGIS e também para este site são gerenciadas pelo [transifex](#). Se você quiser contribuir com a tradução do plugin e da documentação (e gostaríamos muito de ter sua ajuda!), pode solicitar para se juntar a [nossa equipe por meio de transifex](#), ou enviando um email para trends.earth@conservation.org.

Lançamento de uma nova versão do plugin

Sumário

O lançamento de uma nova versão do Trends.Earth envolve a atualização dos números de versão, a criação de etiquetas de git, a compilação do pacote do plugin e a sua publicação tanto no GitHub como no repositório de plugins do QGIS. O processo é simplificado através de tarefas do invoke que automatizam a maioria destes passos.

Fluxo de trabalho de lançamento

Siga estes passos para criar um novo lançamento público:

1. Atualizar o registo de alterações

Primeiro, atualize o registo de alterações (changelog) em LDMP\metadata.txt com detalhes sobre o que mudou nesta versão. Inclua o número da versão e a data de lançamento, seguidos de pontos que descrevam as novas funcionalidades, correções de erros e outras alterações.

2. Definir o número da versão

Execute a tarefa set-version para atualizar os números de versão em toda a base de código:

```
invoke set-version -v X.Y.Z -m
```

Onde X.Y.Z é o novo número da versão (ex: 2.1.20). A bandeira -m garante que os números de versão sejam também atualizados nos módulos dependentes (trends.earth-schemas e trends.earth-algorithms).

Este comando irá:

- Atualizar o ficheiro LDMP/metadata.txt com a nova versão
- Gerar o ficheiro LDMP/_version.py com informações do Git
- Atualizar a versão na documentação (docs/source/conf.py)
- Definir o sinalizador experimental com base em números de versão pares/ímpares (par = estável, ímpar = desenvolvimento)
- Atualizar as referências de dependências nos ficheiros de requisitos

Note

Para lançamentos estáveis (números de versão pares, como 2.1.18), a tarefa set-version atualizará as referências de dependência para utilizarem versões etiquetadas. Para lançamentos de desenvolvimento (números de versão ímpares, como 2.1.19), utilizará o ramo principal.

3. Atualizar os scripts do GEE (se aplicável)

Se tiver feito alterações nos scripts do Google Earth Engine (na pasta gee) ou nos repositórios trends.earth-schemas ou trends.earth-algorithms, adicione a flag -g ao executar o comando set-version``:

```
invoke set-version -v X.Y.Z -m -g
```

Isto irá atualizar os números das versões em todos os ficheiros de configuração dos scripts do GEE. Antes de publicar, crie etiquetas de lançamento para os repositórios dependentes, de modo a que os scripts do GEE resolvam as versões corretas:

```
# In trends.earth-schemas (where x.y.z is the new version number
invoke set-tag -v x.y.z

# In trends.earth-algorithms (where x.y.z is the new version number)
invoke set-tag -v x.y.z
```

Assim que ambas as dependências estiverem etiquetadas, publique os scripts atualizados no servidor da API do Trends.Earth:

```
invoke tecli-publish
```

Isto carrega todos os scripts do GEE para o api.trends.earth, de modo a que fiquem disponíveis para os utilizadores do plugin. Deve ser administrador para executar este comando com sucesso.

4. Confirmar todas as alterações

Certifique-se de que todos os ficheiros modificados foram submetidos no git:

```
git add -A
git commit -m "Release version X.Y.Z"
```

5. Criar e enviar etiquetas de git

Após submeter as alterações de lançamento neste repositório, execute o comando invoke set-tag:

```
invoke set-tag -v x.y.z # (where x.y.z is the new version number
```

Changelog

Isto cria a etiqueta anotada de git do plugin (ex: v2.1.20) e envia-a para o GitHub. Se tiver alterações por submeter, a tarefa irá solicitar que as submeta primeiro.

6. Publish stable GitHub release

Pushing the version tag automatically builds and publishes the GitHub release when the final numeric component is even (for example, v2.1.20). The workflow builds the ZIP from that exact tag, embeds its commit SHA, and attaches the archive to the release. Tags whose final numeric component is odd are treated as development versions and do not create a GitHub release.

To rebuild or replace an existing release asset without moving its tag, run:

```
gh workflow run publish_plugin_repository.yaml -f tag=v2.3.4
```

7. Publicar no repositório do QGIS

Por fim, carregue manualmente o plugin para o repositório de complementos do QGIS:

1. Descarregue o ficheiro zip do plugin (ex: LDMP_2.1.20.zip) a partir do lançamento do GitHub que acabou de criar (este estará listado na secção «Assets»).
2. Inicie sessão no [repositório de complementos do QGIS](#)
3. Navegue até à sua página de gestão de complementos
4. Carregue a nova versão utilizando o ficheiro zip

O repositório do QGIS irá validar o ficheiro zip e disponibilizá-lo aos utilizadores através do gestor de complementos do QGIS.

Convenções de Numeração de Versões

O Trends.Earth segue as seguintes convenções de numeração de versões:

- O Trends.Earth segue as seguintes convenções de numeração de versões:
- **Números ímpares** (ex: 2.1.19, 2.1.21): Lançamentos de desenvolvimento para testar novas funcionalidades.

As versões de desenvolvimento nunca são publicadas no repositório do QGIS e são utilizadas apenas pela equipa de desenvolvimento para a realização de testes.

Changelog

Esta página lista o histórico das versões de **▲TRENDS.EARTH**.

2.3.6 (October 9, 2026)

- Ensure results VRTs do not reference tiffs written to temp folder
- Fixed drought vulnerability calculations to preserve population values as floating-point numbers
- Improved compatibility with QGIS 4 and Qt 6
- Requires QGIS 3.44 or newer, changed supported Python version to 3.12 or newer

2.3.4 (September 29, 2026)

Changelog

- Fix formatting of «Soil organic carbon change from initial year to final year» table
- Remove no longer used SOC by transition table from land degradation report template
- Add rows to drought excel report template to allow for longer timeseries
- Fix off-by-one bug in worst drought calculation (fourth year of every period was not considered)
- Added support for importing male and female population rasters
- Fixed WorldPop population exports and processing to preserve totals and precision during resampling
- Improved performance and progress reporting for zonal statistics, including FWv2 processing resolution
- Added options to configure notification emails during registration and profile updates
- Added experimental LDN planning analysis support
- Updated translations and dependencies
- Applied security and maintenance fixes
- Security fixes for QGIS plugin website

2.3.2 (July 2, 2026)

- Added support for 30-meter resolution land productivity datasets (FWv2)
- Fixed incorrect date range handling for JRC (Joint Research Centre) datasets
- Fixed Qt 6 compatibility issues causing charts not to display correctly
- Fixed a false «not logged in» error that could appear despite valid credentials
- Improved thread safety and credential caching for authentication to prevent intermittent login failures
- Translation updates and other minor bug fixes and performance improvements

2.3.0 (May 12, 2026)

- Added support for QGIS 4
- Added global SDG 15.3.1 and Trends.Earth results datasets to the download tool
- Added (experimental) option to connect a personal Google Earth Engine account via OAuth (browser-based sign-in)
- Fixed a bug in the Total Carbon analysis tool that caused failure to compute annual statistics
- Fixed loading of boundary files stored in compressed format
- Fixed issue where changing the study area didn't refresh the interface
- Fixed map layer style registration in the QGIS symbology panel
- Misc minor bug fixes and performance improvements

2.2.10 (April 20, 2026)

Changelog

- Fixed SDG 15.3.1 summary failing on multi-period datasets due to duplicate band name matching across periods

2.2.8 (April 17, 2026)

- Fixed custom data import (land cover, SOC, LPD) to correctly reproject rasters in non-WGS84 coordinate systems such as UTM

2.2.6 (April 13, 2026)

- Added experimental support for LDN counterbalancing analysis
- Moved file downloads to background tasks to keep the interface responsive
- Added local job logging for better troubleshooting of locally-run analyses
- Improved caching of area extents for faster dropdown loading
- Added support for parallel processing in `te_algorithms`
- Added documentation on SO2-3 indicator calculations
- Atualização de tradução e dependência

2.2.4 (March 19, 2026)

- Melhorias da segurança e estabilidade das «threads» na API, gestor de trabalhos, sistema de download e criação de relatórios
- Adicionada uma opção para alterar o URL do servidor da API
- Filtragem de conjuntos de dados e memória de widgets melhoradas para um melhor desempenho
- Resolvido o problema com diálogos de importação de dados em ecrãs pequenos
- Resolvida a resolução de trabalhos em séries temporais
- Resolvida o carregamento do módulo do plugin durante a melhoria para reduzir erros após a atualização
- Added in-plugin news feed
- Added handling for Google Earth Engine terms of service change
- Optimized local caching of job results to improve plugin responsiveness
- Add styles for SPEI layers and NDVI integrals
- Atualizações de documentação, incluindo uma nova secção do mapa de estado, guia de perguntas frequentes atualizado e métricas de produtividade
- Atualização de tradução e dependência

2.2.2 (28 de Janeiro de 2026)

- Adicionada a opção para ativar ou desativar as notificações de email para as atualizações do estado do trabalho

- Suporte para importar dados de produtividade do terreno (LPD) personalizados e usá-los nos relatórios UNCCD
- Suporte para carregar os ficheiros JSON de resultados diretamente no mapa
- Processo de redefinição da palavra-passe e inscrição de novos utilizadores atualizado para melhor segurança
- Resolvido o carregamento dos limites sem iniciar sessão
- Adicionado recorte de conjuntos de dados personalizados opcionais na importação
- Restringir o cálculo SOC para respeitar o ano dos dados SOC da linha de base (2000)
- Melhorias de segurança e estabilidade da thread
- Atualizações de documentação e tradução

2.2.0 (24 de Novembro de 2025)

- Mudar conjunto de dados de limite predefinido de Natural Earth para geoBoundaries (os mapas de base ainda usam Natural Earth)
- Assegurar que os nomes de execução personalizados se propagam pelos fluxos de trabalho do relatório UNCCD, vulnerabilidade de seca, alteração urbana, biomassa de restauro e resumo de carbono total
- Otimizar a preparação do mapa de base acelerando a extração de arquivos e reduzindo as operações duplicadas
- Comprimir os pedidos de API e reforçar a lógica de novas tentativas para reduzir as falhas transientes ao comunicar com a API trends.earth
- Atualizar as ligações de documentação e remover folhas informativas desatualizadas
- Conjuntos de dados atualizados disponíveis em <https://data.trends.earth>

2.1.20 (10 de Outubro de 2025)

- Adicionar funcionalidade de predefinição para cálculos do sub-indicador SDG 15.3.1, com predefinições que correspondem ao ciclo de relatórios UNCCD mais recente
- Melhorar a resolução de execuções falhadas ou pendentes - todas as execuções serão agora apresentadas na janela dos conjuntos de dados, exceto se forem eliminadas
- Suportar a visualização dos registos de trabalho para execuções remotas
- Sincronizar para as versões te_schemas e te_algorithms, incluindo um melhor suporte de processamento paralelo dos mais recentes te_algorithms
- Reparar a resolução da seleção de conjuntos de dados FAO-WOCAT/JRC/Trends.Earth no cálculo do sub-indicador SDG 15.3.1
- Otimizar as interações de API trends.earth (usando tokens de atualização e removendo pedidos desnecessários)
- Usar setuptools-scm para identificar versões de forma dinâmica (mais relevantes para correr as versões de desenvolvimento da trends.earth)
- Atualizar googleearthengine-api para 1.6.6

2.1.18 (5 de Agosto de 2025)

- resolvido: erro importante com a seleção predefinida do período de relatório por @merydian em #943
- resolvido: erro nos sub-indicadores para SDG 15.3.1 com períodos de progresso por @merydian em #945
- Atualizar traduções de @github-actions[bot] em #946
- [pre-commit.ci] pré-cometer a auto-atualização por @pre-commit-ci[bot] em #929
- Resolver o período de progresso incluído até quando não é verificado por @dimasciput em #954
- Adicionada nix shell para simplificação do fluxo de trabalho de programador NixOS por @timlinux em #950
- Processo Qgis por @timlinux em #955
- Final tweaks for experimental 2.1.18 by @timlinux in #956
- [pre-commit.ci] pré-cometer auto-atualização por @pre-commit-ci[bot] em #948
- Tentar resolver problema de configuração do plugin por @azvoleff em #951
- Atualizar traduções por @github-actions[bot] em #958
- Adicionar períodos de progresso múltiplos de suporte para SDG 15.3.1 por @merydian em #961
- Fazer com que os relatórios funcionem no modo de produtividade FAO-WOCAT para a tarefa local SDG 15.3.1 por @merydian em #963
- Atualizar traduções por @github-actions[bot] em #964
- Adicionar suporte para conjuntos de dados de precipitação CHIRPS e UK-CEH em ferramentas de seca

2.1.16 (31 de Julho de 2024)

- Remover versões de dependência para resolver #812
- Resolver depreciação de np.float
- Atualizar conjuntos de dados para as versões mais recentes (MODIS, CHIRPS, ESA CCI, PERSIAN, MODIS ET, Hansen Global Forest Change)
- Mudar para ruff para formatação/linting de código
- Para a tarefa de sub-indicadores SDG 15.3.1, quando são necessários os dados de população, voltar para a predefinição de um conjunto de dados que corresponde ao ano final dos dados de produtividade, ou dentro de três anos da data se não for possível encontrar uma correspondência exata
- Várias atualizações de versão de dependência

2.1.14 (24 de Fevereiro de 2023)

- Melhorar a ferramenta de importação de produtividade do terreno, incluindo suporte para dados JRC, FAO-WOCAT ou Trends.Earth e uma opção para especificar os ano inicial e final

2.1.12 (17 de Fevereiro de 2023)

- Melhorar a ferramenta de download de dados, incluindo o suporte para descarregar dados de população WorldPop usados no relatório UNCCD

2.1.10 (7 de Fevereiro de 2023)

- Resolver erro ao usar shapefiles/geojsons para definir aoi (#768)
- Assegura que as definições do utilizador são mantidas e não são redefinidas pelas verificações do modo offline
- Atualizar traduções

2.1.8 (25 de Janeiro de 2023)

- Campo JSON de saída para Praias que contém cálculos de falsos negativos/positivos atualizados.
- Resolve #752 - conjuntos de dados de falsos positivos/negativos que não são apresentados na versão não em Inglês

2.1.6 (8 de Dezembro de 2022)

- Incrementa número máximo de classes de cobertura do solo para 45 (particularmente útil para utilizadores que queiram usar a legenda Corine)
- Resolver problema que estava a surgir em Mac/Linux devido a ficheiro de interface QgsPanelWidget em falta
- Várias correções de bugs menores adicionais

2.1.6 (21 de Novembro de 2022)

- Correções #691 (erro que surgiu ao limpar pasta dos dados de base). Quando o diretório dos dados de base for limpo, o plugin irá agora repor o valor para o definido por defeito.
- Corrigir erro de python que surge quando a janela de configurações é fechada antes de o ecrã dos conjuntos de dados ser aberto.

2.1.2 (18 de Novembro de 2022)

- Correções #724 - assegurar que o número máximo de classes de cobertura do solo é definido para 38 por defeito (ignorando o valor por defeito previamente definido em qsettings, que era 32).
- Correções #722 - colocar nome de ficheiro em maiúscula quando o código se refere a terra_cover_transition_matrix_unccd.json.
- Correções #726 - tornar o ano do conjunto de dados visível na janela de conjuntos de dados, para conjuntos de dados importados

2.1.0 (11 de Novembro de 2022)

- Configurações de plugin movidas para dentro da janela principal de configurações de plugin QGIS
- Versão de marshmallow-dataclass incrementada para 8.5.10 e de marshmallow para 3.18.0. Isto exigiu mudanças em trends.earth-schemas, pelo que é possível que alguns trabalhos antigos não consigam ser carregados nesta nova versão (sugerir voltar a executá-los, se necessário).

2.0.7 (31 de Outubro de 2022)

- Ativar a geração de tabelas do Excel para os ODS 15.3.1 ao utilizar classes personalizadas - os vários pequenos erros que existiam já devem estar todos corrigidos.
- Limitar a geração de relatórios a 3 tentativas - depois disso, parar de tentar gerar relatórios até ao reinício seguinte (impede que o Trends.Earth esteja constantemente a criar processos de geração de relatórios em caso de falha dessa funcionalidade).

2.0.5 (19 de Outubro de 2022)

- Corrigir a geração do resumo de ODS ao utilizar o LPD JRC ou o LPD WOCAT.

2.0.3 (19 de outubro de 2022)

- Permitir contornar a reposição da legenda da cobertura do solo ao escolher predefinições na ferramenta de ODS.
- Desativar a geração de ficheiros do Excel quando a legenda subordinada não tiver 7 classes.
- Corrigir a geração de relatórios ao utilizar uma legenda personalizada.

2.0.1 (13 de outubro de 2022)

- Capacidade adicional de personalizar completamente a legenda da cobertura do solo utilizada no Trends.Earth, incluindo o número, o nome, a cor e a codificação de cada classe incluída na legenda. Isso aplica-se a todos os cálculos que utilizam dados sobre a cobertura do solo, incluindo os da ferramenta de ODS 15.3.1 e os das ferramentas de indicadores de produtividade, carbono orgânico e cobertura do solo.
- Ferramenta de importação da cobertura do solo reformulada, a fim de permitir a importação de dados e a atribuição de classes a legendas personalizadas.
- Várias correções de pequenos erros.
- Adicionar a capacidade de incluir um conjunto de dados de falsos positivos/negativos no relatório da CNUCD. A próxima versão incluirá uma funcionalidade adicional para aplicar e comunicar falsos positivos/negativos.
- Esta versão experimental desativa a criação de tabelas de resultados do Excel para a ferramenta dos ODS 15.3.1. Esta funcionalidade vai estar presente na próxima versão (estável).

2.0 (20 de julho de 2022)

- Um interface completamente novo com ecrãs de navegação e carregamento de conjuntos de dados simplificados
- Documentação e website atualizados
- A Trends.Earth está agora traduzida para todas os idiomas oficiais da UN, sendo que estão também planeadas traduções para Português, Suaíli e Persa (para o novo interface).
- Novas funções para suportar o mais recente ciclo de reportagem da UNCCD (incluindo suporte direto para exportar resultados para carregamento no sistema UNCCD PRAIS)
- Novas funções que suportam a avaliação do risco, vulnerabilidade e exposição de seca
- Novas funções de ferramentas de séries de tempo (suporta agora “plotting trend”, WUE, etc.)

1.0.10 (7 de julho de 2022)

- Resolver problema com TransformDirection relacionado com mudanças em QGIS 3.22+

1.0.8 (15 de outubro de 2021)

- Resolver problema com a gravação da última camada para o cálculo de ODS (relacionado com problemas #500 e #505)

1.0.6 (15 de julho de 2021)

- Remova trends.earth-schemas como submódulo e instale através de setup.py (muito mais limpo para gestão de versões, desenvolvimento, etc.)
- Corrigido bug ao carregar dados NDVI de 2020 em código GEE (nome do ativo não estava definido corretamente, por isso 2020 não era carregado)
- Várias tarefas de invocação adicionadas para ajudar no desenvolvimento/lançamento de plug-ins.

1.0.4 (30 de junho de 2021)

- Adicionar WorldPop e «Gridded Population of the World» versão 4 (GPWv4) aos conjuntos de dados disponíveis a partir do Trends.Earth
- Atualizar para permitir acesso à cobertura do solo da ESA-CCI até 2020
- Atualizar conjuntos de dados GPCC e GPCP
- Corrigir bug quando são usados tampões não inteiros
- Correções de pequenos erros de documentação

1.0.2 (14 de agosto de 2020)

Changelog

- Corrigir código de área urbana para permitir processamento de AOI com áreas entre 10 000 e 25 000 km².
- Adicionar os últimos dados do MERRA2 (até 2019).
- Remover a limitação de área máxima da ferramenta de transferência.
- Incrementar versão httpLib2 to 0.18.0.
- Atualizar de GPCC V6 para GPCC V7
- Adicionar dados de desflorestação de 2019 de Hansen et al.
- Atualizar para o código de cores mais recente para dados de degradação (tendo em conta questões de daltonismo vermelho/verde).
- Adicionar suporte para mais tipos de dados em ficheiros de forma de entrada (adicionar PointZ, MultiPoint, MultiPointZ, PolygonZ, MultiPolygonZ).
- Correções de vários bugs para resolver erros de Python que estavam a surgir com algumas mensagens do QMessageBox.

1.0.0 (27 de abril de 2020)

- Adicione capacidade de baixar e usar binários pré-compilados (compilados com o Numba) para acelerar alguns cálculos locais. No momento, isso está disponível apenas para o cálculo da tabela de resumo do ODS 15.3.1, mas eventualmente ele também será expandido para outras ferramentas.
- Relacionado ao item acima, agora existe um botão de configurações «avançadas» na janela de configurações, que permitirá aos usuários baixar binários pré-compilados e ativar ou desativar o log detalhado de mensagens enquanto a ferramenta está em execução. Essas mensagens de log podem ser úteis ao tentar solucionar problemas, se você encontrar alguma.
- Melhora as verificações de validade da geometria em arquivos de entrada e fornece uma mensagem de erro em vez de lançar uma exceção quando houver erros de geometria.
- Corrigir o processamento usando novos dados MODIS (os arquivos não foram atualizados corretamente na última versão)
- Corrigir o código de cores das coberturas da terra na janela de agregação de classes de cobertura da terra
- Adiciona informações de versão mais detalhadas ao diálogo sobre.
- Adiciona detalhes adicionais à ferramenta de download de dados.
- Adiciona o ID do trabalho à janela de downloads para facilitar o relatório de erros.
- Pequenas correções de bugs (classificação de trabalhos e tabelas de downloads).

0.98 (2 de abril de 2020)

- Primeira versão do QGIS3 - muitas correções para atualizar para a API Qt5 e QGIS3.
- Atualiza todas as dependências do plugin para as versões mais recentes a partir de janeiro de 2020.
- Corrige a ferramenta de download de dados para ter 1style padrão para todos os conjuntos de dados disponíveis.

Changelog

- Começa a mudar para as estruturas QgsProcessing e QgsTask - atualmente apenas a ferramenta carbon é migrada, mas todas as ferramentas serão migradas antes da versão 1.0./
- Formatação de limpeza da planilha de saída da ferramenta de carbono para tornar mais claro o significado de cada coluna.
- Atualiza todos os scripts GEE para usar a versão mais recente da API GEE (0.1.213).
- Salva mais configurações escolhidas nas caixas de diálogo das ferramentas nas sessões do QGIS.
- Limpa o código de buffer, para usar as projeções de Área Igual de Lambert centradas nos centróides de polígono para buffer.
- Movemos a documentação para a pasta docs na raiz do repositório trends.earth.
- Adicionamos mais detalhes sobre como contribuir para o desenvolvimento de Trends.Earth,
- Limpamos o repositório removendo os arquivos de tradução compilados e adicionando esses tipos de arquivo ao arquivo «.gitignore».
- Altera o nome do projeto Transifex para «trendsearth».
- Várias compatibilidade e pequenas correções de bugs.

0.66 (20 de julho de 2019)

- Limitamos a área máxima para tarefas a 10.000.000 km², exceto para tarefas na área urbana, que é limitada a 10.000 km².
- Adicionamos a seção de plano de fundo ao ODS 11.3.1 e atualize o tutorial do ODS 11.3.1.
- Atualizamos o código ODS 11.3.1 para incluir 1998 na série (internamente durante o cálculo) para filtrar o ruído desde o início da série urbana.
- Corrijimos as restrições de data na ferramenta tudo-em-um do ODS 15.3.1 para considerar a disponibilidade do ESA e do MODIS.
- Adiciona uma seção ao leia-me sobre como instalar as versões do Github.
- Atualiza e revisa as traduções em espanhol, atualize as traduções do Google para outros idiomas.

0.64 (9 de julho de 2019)

- Corrije o manuseio de NoData na ferramenta de carbono total.
- Adiciona suporte para os dados Hansen 2018 na ferramenta de carbono total.
- Adiciona suporte para dados globais de biomassa de 30m de carbono total do Wood's Hole
- Define o ano final máximo para que uma ferramenta ODS 15.3.1 de uma etapa seja 2015 (correspondendo aos dados da ESA).
- Torna a produtividade do Trends.Earth o conjunto de dados padrão na ferramenta de uma etapa do ODS para 15.3.1.

0.62 (27 de janeiro de 2019)

Changelog

- Adiciona uma ferramenta experimental para mapear possíveis retornos de carbono de intervenções alternativas de restauração.
- Adiciona dados MODIS 2018.
- Correções diversas no dimensionamento de janelas para janelas da GUI.
- Atualiz para o openpyxl mais recente - corrige o carregamento do logotipo Trends.Earth nas tabelas de resumo.
- Adiciona a lista de publicações para ajudar os documentos.

0.60 (3 de dezembro de 2018)

- Acrescenta o cálculo da mudança na área urbana e na taxa de crescimento populacional (ODS 11.3.1)
- Corrige alturas padrão do botão/campo de entrada Adicionar seleção de cidade para AOI
- Adiciona buffer opcional da AOI

0.58 (11 de agosto de 2018)

- Adiciona uma seção de teste à página de cálculos
- Adiciona versão de teste do carbono total (acima e abaixo do solo) e das emissões devido ao desmatamento
- Pequenas correções de bugs, incluindo polígonos inválidos nas AOIs de entrada

0.56.5 (21 de maio de 2018)

- Corrige o erro com a importação de LPD solicitando um ano de dados.

0.56.4 (21 de maio de 2018)

- Sempre redefina os dados importados para a resolução mais alta.
- Corrige as zonas climáticas personalizadas de importação do SOC para usar um conjunto de dados expandido das zonas climáticas para eliminar nenhum dado.
- Atualiza MOD16A2 com os dados mais recentes.
- Forçar entrada da data na importação de dados SOC e LC
- Adicione saídas globais Trends.Earth à ferramenta de download.
- Corrija o manuseio de valores NULL nas legendas.

0.56.3 (21 de abril de 2018)

- Corrija o cálculo das tabelas de resumo das AOIs divididas no 180° meridiano (Fiji, Rússia etc.).
- Modifique o cálculo do estado para que áreas com mudanças de magnitude muito pequenas na integral NDVI (<0,01 unidades NDVI durante todo o período) sejam consideradas estáveis.

0.56.2 (10 de abril de 2018)

- Pequenas correções unicode.

0.56.1 (10 de abril de 2018)

- Corrija o erro do marshmallow na carga do plugin

0.56 (9 de abril de 2018)

- Corrigir problema com rasterização de dados (rasters vazios na saída)
- Forçar o usuário a escolher a resolução de saída ao rasterizar um vetor
- Suporte ao cálculo da degradação do SOC a partir de dados SOC e LC personalizados

0.54 (8 de abril de 2018)

- Suporte ao carregamento de dados LPD, SOC e LC personalizados.
- Estilos de limpeza para que correspondam a maps.trends.earth
- Atualiza pyopenxl
- Adicione ícones de importação/carregamento a todas as caixas seletoras de camadas

0.52.1 (21 de março de 2018)

- Pequenas correções de bugs durante a oficina de Antalya.

0.52.1 (21 de março de 2018)

- Pequenas correções de bugs durante a oficina de Antalya.

0.52 (19 de março de 2018)

- Limpe o código de processamento da AOI.

0.50 (15 de março de 2018)

- Passe a exceção se relacionado apenas à adição do logotipo Trends.Earth no arquivo do Excel.
- Várias pequenas correções de bugs.

0.48 (13 de março de 2018)

- Corrigir formatação da tabela

0.46 (13 de março de 2018)

- Suporte ao cálculo da tabela de relatórios com várias geometrias (Fiji, Rússia)
- Adicionar tabelas LPD e LC à guia da planilha UNCCD
- Limpe a mensagem de aviso na ferramenta de importação LPD
- Corrigir o carregamento final combinado da camada de produtividade da TE
- Corrigir tarefas de download (ainda sem estilos)

0.44 (12 de março de 2018)

- Adicionar JRC LPD
- Adicionar ferramenta para fazer upload de dados personalizados de cobertura do solo
- Adicionar ferramenta para fazer upload de dados de produtividade personalizados
- Adicione que o upload personalizado do SOC será lançado em breve
- Ferramenta Adicionar para adicionar mapas base usando dados do Natural Earth
- Adicione a ferramenta tudo-em-um-passo para calcular os três subindicadores de uma só vez
- Renomeie a classe «Solo Exposto» para «Outra classe de cobertura» para obter consistência com a UNCCD
- Atualizar documentos
- Atualize para o marshmallow 3.0.0b7
- Mova o código GEE para o repositório principal trends.earth
- Melhore o manuseio de AOIs, principalmente quando os shapefiles são usados para entrada
- Manipule saídas de vários arquivos do GEE, agrupando-as em VRTs
- Suporte ao processamento de dados para países que cruzam o 180º meridiano
- Melhorar a formatação da tabela de resumo
- A partir de agora, as versões do script GEE serão correspondidas à versão do plugin

0.42 (4 de fevereiro de 2018)

- Corrigir falha na alteração da agregação de LC (devido setEnabled no rótulo removido)

0.40 (4 de fevereiro de 2018)

- Remova o uso do modo para o indicador de cobertura do solo.
- Combine a tabela de resumo e as ferramentas de criação de mapa de indicadores SDG.
- Adicione stub para onde o produto JRC LPD estará disponível.
- Salve o subindicador de produtividade como banda 2 no arquivo do indicador SDG.
- Bump script GEE para v0.3.

Changelog

- Corrija o erro devido à divisão por zero na geração da tabela de resumo quando uma classe tiver área zero.
- O padrão é MODIS para cálculos de produtividade.

0.38 (16 de janeiro de 2018)

- Adicionar cálculo anual de carbono orgânico do solo
- Limpe o código de processamento da AOI, permita vários polígonos de entrada nas AOIs do shapefile
- Adicionar sombreamento ao lado dos itens da tabela de agregação da cobertura da terra
- Corrija o problema do firstShow na tabela de agregação
- Revise a saída da tabela de resumo para fornecer mais informações sobre cada um dos três indicadores
- Adicione conjuntos de dados suplementares ao desempenho, estado, cobertura do solo e produção de carbono orgânico do solo.
- Atualize nenhum dado e mascaramento para ser consistentemente -32768 (sem dados) e -32767 (dados mascarados)
- Permitir nomeação de downloads de arquivos
- Adicione ícone ao menu da barra de ferramentas, corrija o nome do plugin.
- Refatorar o código de estilo da camada para extrair informações da banda da saída GEE.
- Adicione uma ferramenta para carregar os conjuntos de dados trends.earth existentes no QGIS.
- Fixar limites de data de cobertura do solo - não permita que datas inválidas sejam selecionadas a partir dos dados da CCI.

0.36 (14 de dezembro de 2017)

- Corrija o problema com showEvent na ferramenta de criação de relatórios de mapas.

Changelog

0.34 (14 de dezembro de 2017)

0.32 (14 de dezembro de 2017)

0.30 (12 de dezembro de 2017)

0.24 (6 de dezembro de 2017)

0.22 (4 de dezembro de 2017)

0.18 (2 de dezembro de 2017)

0.16 (6 de novembro de 2017)

0.14 (25 de outubro de 2017)

0.12 (6 de outubro de 2017)