Walter Maps
User’s Guide
# Table of contents

Introduction .................................................................................................................. 5

More information and user forum............................................................................. 5

Design goals .................................................................................................................. 6

Basic functionality...................................................................................................... 7

Professional functionality ......................................................................................... 8

System requirements .................................................................................................. 10

Installing Walter Maps .............................................................................................. 11

Enabling Walter Maps professional functionality ............................................... 11

Terms of use for the Walter Maps Pro in app purchase ......................................... 11

Privacy Policy ............................................................................................................. 12

Easy-to-read summary .............................................................................................. 12

App permissions ......................................................................................................... 12

Supported geodata formats ....................................................................................... 14

Supported coordinate systems ............................................................................... 15

Background Maps ..................................................................................................... 16

Transferring and working with geodata .................................................................. 16

Language support ..................................................................................................... 17

The user interface ...................................................................................................... 18

Zooming, panning and moving the cursor ............................................................... 18

Selecting geometries ............................................................................................... 19

Drawing ...................................................................................................................... 19
Buttons...........................................................................................................19
Offline help ......................................................................................................20
Mode display....................................................................................................20
Slider dialogue boxes ....................................................................................21
Scroll lists .......................................................................................................21
Colour selection dialogue box.........................................................................22
Using Walter Maps ..........................................................................................23
First time use ..................................................................................................23
Map Display ....................................................................................................24
Measuring distances, areas and directions....................................................26
General Settings ..............................................................................................27
Managing your map layers............................................................................35
   Layer ordering .............................................................................................36
   Zoom to active layer ..................................................................................36
   Layer removal ............................................................................................36
   Layer visibility ...........................................................................................36
   Enable layer visibility toggling .................................................................36
   Enable vector layer geometry editing .......................................................37
   Enable vector layer attribute viewing and editing ....................................37
   Enable viewing of raster layer pixel values .............................................37
   Set a line layer as cutting layer for polygon splitting ................................38
   Locally backup and restore a vector layer ...............................................38
   Cut/save a raster layer from present map extent ......................................38
   Save a web map layer to an offline raster layer .......................................38
Adding map layers.................................................................39
Export layers........................................................................48
Change symbology for a map layer.......................................51
Collecting and editing data....................................................56
Collecting waypoints..............................................................56
Collecting tracks....................................................................59
Creating geometries..............................................................60
Deleting geometries..............................................................61
Splitting polygons..................................................................61
Merging lines and polygons ..................................................62
Exploding multi-lines and multi-polygons ...............................63
Cleaning lines and polygons ..................................................64
Viewing and editing attributes ..............................................65
Viewing raster data values....................................................71
Professional Toolbox functions ............................................72
Distribute sample plots .........................................................72
File formats ..........................................................................73
Import/export vector format .................................................73
Importing points in wmtxt file .............................................74
Exporting vector data in wmtxt file format.............................75
Exporting vector data in xls file format ................................75
Point symbols in SVG format ................................................76
Walter maps symbol package ..............................................76
Walter Maps layer package ................................................77
Walter Maps – User’s Guide

Walter Maps map package ......................................................... 78
Walter Maps language file .......................................................... 78
Advanced usage ........................................................................ 79
Creating validating data collection forms ..................................... 79
Rule file example using one set of rules ...................................... 79
Rule file example using a multiple set of rules ............................. 83
Rule file validation ..................................................................... 84
Building your own Walter Maps map packages .............................. 85
  Future safe ............................................................................ 85
  Naming convention ................................................................. 85
  Contents of the map package .................................................... 86
  The projection definition file ..................................................... 87
  The map document file ............................................................ 87
  Source code for saving and loading the map document ............... 100
Creating your own web map definition files .................................. 101
Translating Walter Maps to your own language ............................. 104
Glossary .................................................................................... 105
Introduction
Walter Maps is a professional tool for online and fully offline geodata collection and editing. You can use it stand alone or as a ready-to-use field data collection tool in your existing geodata environment. Walter Maps is available for all major mobile platforms.

Walter Maps has an intuitive user interface for cost effective field data collection and with its fully open data formats you can exchange geodata with anyone and with almost any geodata solution or system.

The app comes in two flavours, Walter Maps and Walter Maps Pro. Walter Maps contains some editing features, and could be regarded as a reader app for geodata and maps produced by Walter Maps Pro. The target audience for Walter Maps Pro are professionals who collect geodata and produce maps. Within Walter Maps, you access the Pro functionality by purchasing monthly unlocks of the Walter Maps Pro, i.e. you only pay for Pro features when you need them.

In this guide, professional features are shown using the ⚡ icon and important notes are shown using the ! icon.

More information and user forum
More information about Walter Maps can be found at:

http://waltermaps.com

There you can also access the Walter Maps User Forum.

© 2018, Dianthus AB, Sweden
Design goals

Walter Maps is designed and developed for professionals by professionals. Our goals for designing Walter Maps are:

- Walter Maps has an intuitive user interface for cost effective field data collection and will fully utilize the sensors of your device – You get things done
- Walter Maps only uses open data formats and data distribution channels – Your data is your data period
- Walter Maps will work with offline data – You are able to collect data everywhere
- Collected data is easily shared – In areas with internet connection, you can share data with other people anywhere and anytime through Dropbox
- Walter Maps is able to exchange data with other systems – Other systems are easily able to produce and consume data to or from Walter Maps through our open, easily maintained and well documented map package format
- Walter Maps is available on all major app stores -You can share your collected data with 97.5% of all mobile device users (Android, iOS and Windows 10). Additionally, your data can be shared with all users of Windows 10 desktop operating system
Basic functionality

Walter Maps and Walter Maps Pro contain the following basic features:

- Multi-platform support
- No ads
- Shape, TIFF, GeoTIFF & JPEG support
- Support for WMS, WMTS, and TMS online map layers
- Record waypoints & tracks
- Easy measurements of coordinates, lengths and areas
- Supports many length and area units, e.g. m, km, feet, miles, yards, sq.feet, m², ha, acres etc.
- Supports any map projection (projected coordinate system)
- Add and delete points, polygons & lines
- Collect 2 photos/videos per point, line or polygon feature
- View and edit attributes
- Create own symbologies (one per map layer)
- More than 670 built in symbols for points
- Use own point symbols in svg-format
- Download single or multiple symbols in svg-format
- Create and open map packages
- Copy GPS position to clipboard
- Use Dropbox for sharing and backup of data
- Download layers from http URLs
- Full access to data collected by Walter Maps Pro
- Multi lingual-support, i.e. create your own language files. Out of the box Walter Maps supports English and Swedish.
- Fully described JSON based map document and map package format, to develop individual read and write routines in other systems.
Professional functionality

Walter Maps Pro contains the following additional professional features:

- Merge polygon features
- Multi part features to single part features
- Split polygons by temporary drawn lines
- Split polygons by other line layer
- Clean polygons
- Single step undo of geometry edits
- Create new shape files
- Add new attribute fields to shape files. Deletion of attribute fields is not possible in Walter Maps in order to preserve data consistency with other systems.
- Use custom built rules (single or multiple) for creating data collection forms to ensure data quality and completeness
- Tools for assigning date, time, polygon area, line length or polygon perimeter, centroid, UUID, GPS position, cursor position and altitude, compass direction, device pitch and roll to field attributes
- Collect any number of photos/videos per feature
- Select photos from device album
- Import points from text files
- Export vector layers to text files or MS Excel (*.xls)
- Download layers from password protected http URLs
- Email complete map or selected layer
- Launch websites based on feature attribute values
- Save WMS, WMTS, and TMS layers to offline raster layers
- Possibility to use cached WMS, WMTS, and TMS in online mode
- Cut and resample raster layers
- Export present map view for fast and easy sharing of map snapshots with other people
- Any number of conditional symbologies, i.e. symbology by field based SQL-queries
- Pan to projected or geographic coordinate (x,y or lat,long)
- Display distance circles around current GPS position
- Display distance circles around cursor position
- Display proximity circles around current GPS position and around cursor position
- Measure raster values
- Copy coordinate for cursor position to clip board
- Quickly alternate layers in the map
- Local backup of vector data
- Energy save mode
- Toolbox functions:
  - Distribute sample plots within polygons and along lines
System requirements

Walter Maps works on Apple iOS (from version 7), Android (from version 4.4), Windows 10 mobile (both ARM and X86 processors), and Windows 10 desktop/laptop. These versions are available from the iOS App Store, Android Google Play, and the Microsoft Store respectively.

These are the system requirements for Walter Maps:

- Touch screen for optimal functionality. For Win10 devices lacking touch screen, zoom, pan, and editing functions, a standard 2-button scroll wheel mouse can be used.
- 2 Gbyte of RAM. The app itself requires about 350 Mbyte of RAM, but 2 Gbyte is needed due to other coexisting apps and system resources.
- GPS device for full functionality, built-in or external
- Network connection when wirelessly transferring maps to and from the device

Walter Maps is also available for Windows 7 desktop/laptop computers. More information is given upon request from Dianthus AB, Sweden, at e-mail info@dianthus.se.
Installing Walter Maps

Installation of Walter Maps is made directly from the iOS App Store, Android Google Play, and the Microsoft Store respectively.

Enabling Walter Maps professional functionality

Walter Maps Pro functionality is enabled by the purchase of a monthly subscription within the free version of Walter Maps (in app purchase). Purchasing the monthly subscription is described in the General Settings Section.

Terms of use for the Walter Maps Pro in app purchase

- The in app purchase Unlock Walter Maps Pro lets you unlock all Walter Maps Pro features on a monthly basis. Unlock Walter Maps Pro is only available in the free version Walter Maps.
- If you do not purchase the Unlock Walter Maps Pro, the app will still work, but without the Walter Maps Pro professional features.
- Data collected using the Walter Maps Pro functionality and features will be visible and usable in the free version of Walter Maps without the Pro unlock.
Privacy Policy

Easy-to-read summary
All geodata collected including collected photos, video and sound recordings using Walter Maps are stored on your device (phone/tablet/computer) and are never transferred to anyone else without you initiating it. On your command and only on your command, can your collected geodata be transferred to your Dropbox account or emailed to recipients of your choice.

App permissions
For Walter Maps to function it requires access to certain features on your device. Here is an explanation of each used feature:

Your location
Walter Maps requires your location for general orientation, and for collection of your geodata. No geodata is transferred from your device without you initiating it.

Network communication
Walter Maps uses an Internet connection to upload/download your geodata from your Dropbox account and to send geodata by email. Your data can also be downloaded from ordinary URLs using network communication.

Storage
Walter Maps saves your geodata to your internal device storage or removable sdcard.
System Tools

Walter Maps prevents your phone from sleeping so that the GPS device constantly receives data.

Use of Camera and Sound recording

The camera is used for collection of photos and video recordings. While you are video recording, sound recording is also performed. The use of camera and sound recording can only be initiated by you. No captured photos, videos, or sound recordings are transferred from your device without your initiation.

In-app-purchase

Walter Maps contains the in-app-purchase that enables you to purchase a monthly unlock of all the professional features found in Walter Maps Pro. For Android and iOS, the purchase is an auto-renewal monthly subscription that has to be explicitly cancelled by the user in order for the monthly billing to end. For Windows 10, the purchase has to be renewed manually by the user every month for the app to maintain Walter Maps Pro functionality.
Supported geodata formats

Walter Maps supports ESRI Shape-files, Tiff (with tfw-world-files), GeoTiff, JPEG (with jgw-world-files), and WMS-, TMS-, and WMTS-services. For ESRI Shape-files, attributes for integers, floating point numbers, and strings can be viewed and edited. In Walter Maps Pro you are able to create your own point, line, or polygon shape-files. The following colour formats and data types are supported by the raster formats:

**Tiff/GeoTiff**

- 8 bit grayscale
- 8 bit colour indexed
- 24 or 32 bit colour
- Multispectral (n-bands) 8 bit, where the three first bands will be visualized as RGB colour
- Single band or Multispectral (n-bands) 16 bit (unsigned short or signed short), where the first band will be visualized in grayscale
- Single band or Multispectral (n-bands) 32 bit (unsigned integer or signed integer), where the first band will be visualized in grayscale
- Single band or Multispectral (n-bands) 32 bit (float), where the first band will be visualized in grayscale
- Single band or Multispectral (n-bands) 64 bit (double), where the first band will be visualized in grayscale

**JPEG**

- 8 bit grayscale
- 24 bit colour
Supported coordinate systems

Walter Maps supports approximately 4000 projected coordinate systems (X and Y coordinates). The first time you use Walter Maps you have to select your projected coordinate system for your region, state or country. If you do not know your projected coordinate system, Walter Maps will automatically choose one for you. ! Note that it is always better to manually choose your projected coordinate system if you know what to use. The automatically chosen one will work, but will often differ from the most commonly used coordinate system in your area, leading to misplacement of maps when adding them to Walter Maps.

You should be aware of the following:

- Offline map layers like shape files or raster files (JPEG or TIFF) will not be reprojected by Walter Maps, and will therefore get misplaced on the map display if their coordinate system differs from the presently chosen one. If you add a map layer with a coordinate system in conflict with the one in use, Walter Maps will warn you.
- Online sources like WMS, TMS, and WMTS maps will be reprojected to your chosen coordinate system. This is also valid for WMS and WMTS maps using geographic coordinate systems (Latitude and Longitude coordinates).
Background Maps

Walter Maps provides the world-wide Open Street Map, Open Topo Map and the Swedish Topographic Map off-the-shelf as web map services. You can use it online in all versions of Walter Maps and offline in Walter Maps Pro.

Transferring and working with geodata

All geodata is locally stored on your device. Your original data, copied from your Drobox account, from a URL (the Pro version also supports password protected URLs), or your devices shared storage (only applicable to Android devices and Windows 10 desktop/laptop computers), will be left untouched, serving as an original backup of your data. When editing your data (i.e. adding new data to your map, or editing geometries or attributes) your data is always instantly saved locally to your device. When you have finished working with your data, you can send the data to Dropbox, by e-mail (Pro version only) or to your devices shared storage (only applicable to Android devices and Windows 10 desktop/laptop computers). Your sent geodata is compressed into a zip-file named with the present time and date, making the zip-file itself a backup of your data.
You can easily set up a free or paid Dropbox account at www.dropbox.com.

**Language support**

Walter Maps supports English and Swedish off-the-shelf. If you want support for your own language, you can create your own language files as described in the Translating Walter Maps to your own language section.
The user interface

The user interface is simple, yet effective, and allows you to work effortlessly with gloves on a chilly day.

The orientation of the app is always in landscape mode and it will auto-rotate when rotating your device 180 degrees.

The preferred position of the device is horizontal and held using both hands, while the functionality is mainly accessed using your thumbs.

Zooming, panning and moving the cursor

You always zoom and pan using two fingers and you move the cursor (cross-hair) using one finger only. When moving your cursor, you should consider your touch screen as a mouse-pad, i.e. you don’t have to place your finger on top of the cursor to move it (effectively making touch screen pointing pens superfluous).

On Windows 10 desktop/laptop computers you can zoom and pan using a standard scroll-wheel mouse. Zooming as well as panning around the standard Windows cursor is performed by moving the scroll-wheel forward or backward. Panning without zooming is performed when moving the scroll-wheel one click only. Dragging
the mouse while left-clicking will move the Walter Maps cursor (cross-hair).

**Selecting geometries**

In the various geometry editing modes and in the attribute editing and viewing mode, you select or deselect geometries by directly touching them with one finger. If using a mouse, you left-click directly on the geometry you want to select. The selected geometries will be coloured in cyan, and if more than one geometry is selected the last one will be coloured a little bit darker.

**Drawing**

In the various geometry editing modes and in measuring mode you draw points, lines, or polygons. To draw a single point or a point within a line or within a polygon, move the Walter Maps cursor (cross-hair) to the point’s position using one finger and then touch the surface with another finger while still holding down your first finger. On a Windows 10 desktop/laptop computer you can add points using a standard 2-button mouse. You drag the Walter Maps cursor (cross-hair) using the left mouse button, and place a point using the right mouse button (or double-click left mouse button).

**Buttons**

The app functionality is reached through buttons along the sides of the screen. To exit a specific section or mode you press the ✗ button.
Offline help
Instant offline help is reached in each section of the app by pressing the 🎨 button. When pressed, the functionality for each button present is explained.

Mode display
For each app section or mode (e.g. General Settings mode and Manage Map Layers mode), the mode is displayed on the screen.
Slider dialogue boxes
Slider dialogue boxes appear when user input in a specific range is called for. Slide your finger horizontally wherever you want to change the value.

Scroll lists
A major part of the user interface consists of scroll lists. The scroll list consists of a heading and a scrollable list. The active list item (row) is indicated by a surrounding dark blue/purple rectangle.

For some scroll lists, the active item acts as a button. These buttons are highlighted with a dark filling colour when moving your finger above them. The highlighted button is clicked when releasing your finger from the surface.
**Colour selection dialogue box**

In the colour selection dialogue box you change colour for lines, polygons, labels etc. You can select from pre-set colours to the right, or by sliding your finger across the hue-saturation-lightness components of the dialogue box.
Using Walter Maps

In this guide, both the basic and professional features of Walter Maps are described. Professional features are shown using the icon.

First time use

The first time you use Walter Maps you have to select your projected coordinate system for your region, state or country. If you do not know your projected coordinate system, Walter Maps will automatically choose one for you. ! Note that it is always better to manually choose your projected coordinate system if you know what to use. The automatically chosen one will work, but will often differ from the most commonly used coordinate system in your area, leading to misplacement of maps when adding them to Walter Maps. The most commonly used coordinate system in your area is available from your data provider or from your local geological survey agency.

When manually selecting your projected coordinate system, you can filter the approximately 4000 available coordinate systems by pressing the button.
Map Display

The map display shows the present map layers, your Cursor coordinate, Map scale, and Current time and date. The displayed time and date strings can be setup to your preferred format, described in the General Settings section.

If your GPS position is within your current visible map, the GPS position is shown as ●. The cursor position is shown as a cross-hair ●.
The following buttons are shown when pressing the button.

- Hide all buttons
- Show context sensitive help
- Centre map around GPS position
- Zoom to all map layers
- Measure distances, areas and directions
- Professional tool-box functions
- General app settings
- Map layer settings
- View and edit vector layer attributes, in professional mode raster values can also be viewed
- Start/stop recording of tracks
- Add way-point at current GPS position
- Add way-point manually
- One-step geometry edit undo
- Draw new geometry (point, line, or polygon)
- Delete geometry or geometries
Measuring distances, areas and directions

You enter Measuring Mode by pressing the pencil button. You take measurements by moving the pointer with one finger and then you add measurement points with another finger as described in the Drawing section.

- Show context sensitive help
- Delete last measurement point
- Exit Measuring Mode
- Enter zoom mode to zoom and pan
General Settings

You open the General Settings scroll list by pressing the button. Here you change the Walter Maps overall settings. You return to Map Display mode by pressing the button.

The General Settings scroll list consists of the following items:

- **Online help**: Shows the online Walter Maps User’s Guide in pdf-format
- **Buy or restore Unlock Walter Maps Pro**: Buys or restores a monthly subscription of Walter Maps Professional features
- **About Unlock Walter Maps Pro subscription**: Shows information about the subscription
- **User name**: Stores your user name, and is used for automated tasks, e.g. automatically assigning your name to attributes when collecting tracks and way-points
<table>
<thead>
<tr>
<th>Task</th>
<th>Stores your task (task, project, study, investigation etc.) and is used for automated tasks, e.g. automatically assigning your task to attributes when collecting tracks and way-points.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Energy save mode</td>
<td>Turns on or off energy save mode. Slows down screen update when not touching the screen.</td>
</tr>
<tr>
<td>Use web map cache</td>
<td>Use cached web maps. If cached image tiles are missing, the app tries to download missing tiles from the online source.</td>
</tr>
<tr>
<td>Coordinate system</td>
<td>For choosing your present projected coordinate system</td>
</tr>
<tr>
<td>Use the GPS receiver</td>
<td>Turns on or off your GPS receiver</td>
</tr>
<tr>
<td>Centre map around GPS position</td>
<td>When the GPS position falls outside of the visible map, the map is automatically re-centred around the GPS position.</td>
</tr>
<tr>
<td>Show compass</td>
<td>Shows the compass. ! <strong>Note that</strong> most device compasses have poor precision.</td>
</tr>
<tr>
<td>Show background map</td>
<td>Turns on or off the Open TopoMap background map. It might be beneficial to turn off this map in areas of bad network reception.</td>
</tr>
</tbody>
</table>
Show distance circles around GPS position

Turns on or off distance circles around your current GPS position.

Show distance circles around pointer

Turns on or off distance circles around the pointer.
Proximity circle distance around GPS position
Draws a transparent orange circle around the GPS point with specified radius in present length unit

Proximity circle distance around pointer
Draws a transparent orange circle around the pointer (cross-hair) with specified radius in present length unit

Show GPS precision
The GPS precision is shown as a semi-transparent blue disk/circle around the GPS position

Show pointer coordinate
Displays the pointer (cross-hair) coordinate value

Show map scale
Displays the map scale
Show time and date  Displays time and date

Show zoom tools  Shows or hides the zoom tools in Map display mode. The buttons are found at the lower edge of the map:

The zoom tools are:

- Copy cursor (cross-hair) coordinate to clipboard

- Pan to any X/Y or WGS84 Latitude/Longitude coordinate 🗺️. If the coordinate varies between -90 and +90 and between -180 and 180 respectively, a Latitude/Longitude coordinate is assumed, otherwise an X/Y projected coordinate is assumed.

- Toggle between previous zoom/pan states 🌍
Pan in the four different directions. These tools are useful when using a device without touch screen.

Zoom out or in Pan in the four different directions. These tools are useful when using a device without touch screen.

Language

For choosing your preferred language. At present English and Swedish are supported. You can translate Walter Maps to your own language by create your own language file, as described in the Translating Walter Maps to your own language section.

Length unit

For choosing your preferred length unit:

- feet
- km
- m
- miles
- nautical miles
- yards

Area unit

For choosing your preferred area unit:

- acres
- are
- decare
- fen
- square feet
- ha
- km²
- lí
- m²
- square miles
For choosing your preferred date format:
- dd MM yyyy
- dd/MM/yyyy
- ddMMyyyy
- dd-MM-yyyy
- yyyy MM dd
- yyyy/MM/dd
- yyyyMMdd
- yyyy-MM-dd

For choosing your preferred time format:
- hhmmss (24)
- hh:mm:ss (24)
- hh.mm.ss (24)
- hhmmss AM/PM (12)
- hh:mm:ss AM/PM (12)
- hh.mm.ss AM/PM (12)

Sets the minimum time in seconds between points in a GPS track, to prevent creation of overlapping points. When set to 0 seconds, the time between track points is limited by the GPS.
### Minimum distance between track points (0-10 m)
Sets the minimum distance between track points, to decrease the amount of zigzagging GPS tracks.

### Maximum speed between track points (1-1000 m/s)
Sets the maximum speed between track points, to prevent erroneous spikes in GPS tracks.

### Copy position text to clipboard
Copies your current GPS position to the clipboard. Can be used for forwarding your position to other people through e-mail, SMS or by other ways of sending messages:

My position in WGS84 latitude and longitude is: 65.8235 21.6056, with a horizontal precision of: 65 m.

### Maximum e-mail size (Mbyte)
Set the maximum size for sending e-mails from Walter Maps.

This value is automatically set and is used calculating the map scale. You should only set this value if the automatically setting fails.

### Screen height (cm)

### Remove connection to Dropbox
Removes your connection to Dropbox. After removal, you can always establish a new connection to Dropbox.

### Copy installation ID to clipboard
All installations are given a unique installation ID. This id can be used when automatically assigning field values when collecting data, see the **Default values** section for more information.
Managing your map layers

By pressing the button in Map Display mode, you open the Manage Map Layers mode where you organize your map layers. You return to Map Display mode by pressing the button.

In Manage Map Layers mode (see above image), the present map layers are shown in a scroll list. The present active map layer in the scroll list is surrounded by a dark blue/purple rectangle.
Layer ordering
You change the layer ordering by choosing/activating the layer you want to reorder (by scrolling) and then by pressing the up ↑ and down ↓ arrow buttons.

Zoom to active layer
You zoom to make the active layer fully visible by pressing the magnifying glass button.

Layer removal
You remove your layers one by one, or all layers, by pressing the delete button. Please note that removed layers cannot be recreated.

Layer visibility
You turn layer visibility on or off by pressing the visibility icon button.

Enable layer visibility toggling
In Walter Maps Pro you can quickly and easily toggle between any number of map layers. You enable layer visibility toggling by pressing the visibility icon button for each layer you want to toggle. Back in Map Display mode you toggle these layers by pressing the visibility icon button.
Enable vector layer geometry editing
In Map Display mode you edit vector geometries, as described in section Creating geometries and forward. Geometries can only be edited for one vector layer at a time. You enable geometry editing by pressing the button.

Enable vector layer attribute viewing and editing
In Map Display mode you can view and edit attributes for vector layers, as described in the Viewing and editing attributes section. Attributes can only be edited and viewed for one layer at a time. You enable layer attribute editing and viewing by pressing the button.

Enable viewing of raster layer pixel values
In Walter Maps Pro you can view values for single raster pixel values, as described in the Viewing raster data values section. Only one raster layer can be queried at a time. You enable pixel value viewing by pressing the button.
Set a line layer as cutting layer for polygon splitting
In Walter Maps Pro you can split vector polygons by selected lines in a line layer, as described in section Splitting polygons. To set one of the layers (and one only) as a cutting layer, you press the button (in the image above this button is not visible, because the active layer is a raster layer).

Locally backup and restore a vector layer
In Walter Maps Pro you can make local backups. When editing your vector layer it is convenient to have a local backup of the layer, if something goes wrong in the editing process. To make a local backup of your vector layer you press the button. If a backup is present you can restore the layer by pressing the button. Please note that the restore will overwrite your layer, and that the restore cannot be undone.

Cut/save a raster layer from present map extent
In Walter Maps Pro you can cut and save the active raster layer by pressing the button. If the raster intersects the edges of the visible map, the raster will be cut. If the raster layer is completely outside the visible map, no cutting/saving is performed. Before cutting/saving you have set the name and the resolution (pixel size) for the resulting raster layer. The default resolution is given automatically from the present map resolution.

Save a web map layer to an offline raster layer
In Walter Maps Pro you can cut and save the active web map layer (WMS, WMTS or TMS) by pressing the button (in the image above this button is not visible, because the active layer is not a web map layer). The web map layer will be cut by the edges of the visible map. Before saving, you have set the name and the resolution for the resulting raster layer. The default resolution is given automatically from the present map resolution. The maximum
size for the resulting raster layer is 500 Mbyte. Note that, offline raster layers will improve display performance considerably.

**Adding map layers**
You add map layers by pressing the button. You exit the Add Map Layer mode by pressing the button.

You can add map layers by

- Creating a new vector layer
- Adding an online web map layer
- Adding a map layer from your internal device storage (only applicable for Android devices and Windows 10 desktop/laptop computers)
- Adding a map layer from Dropbox
- Adding a map layer from a URL address
Creating a new vector layer

In Walter Maps Pro you can create a new vector layer, i.e. a point, line, or polygon layer, by pressing the button. You return to Add Map Layer mode by pressing the button.

You add a new vector layer from predefined templates. The three first templates are system defined templates and cannot be altered. To create or alter your own template, you use the following buttons:

- Add a new template
- Alter an existing user defined template
- Delete an existing user defined template
Add a new template

To add a new template, you press the button. The first step is to give the template a name. In this example we call it *my_new_template*. The second step is to add an attribute field to the template by pressing the add field button. In the following steps, you have to select Field Type (float, integer or string), Field Length, and Field Name.

In this example, a new field called AREA of type float and length 12 is created. You can continue adding as many fields as you want. To delete a field you press the button. To save and return to Create New Vector Layer mode you press the save button.

Alter a template

To alter a user defined template you press the button.

Delete a template

To delete a user defined template you press the button.
Choose a template and create your vector file

To actually create a new vector layer you press the desired (i.e. active) template. In the following final steps you give your new vector layer a name and you select geometry type, i.e. point, line, or polygon. After creation, the new empty vector layer is added to your map.
**Adding an online web map layer**

You add an online web map layer (WMS, WMTS or TMS) by pressing the button in Add Map Layer mode.

Off the shelf, *Open Street Map*, *Open Topo Map*, and the *Topographic map of Sweden* can be chosen.

You can add additional user defined web maps (*wmwx*) as described in the **Adding map layers** section. The wmwx file is an xml-file describing the capabilities of you online web map source. You can create your own wmwx files, as described in the **Creating your own web map definition files** section. Additional wmwx files can be found and downloaded at the Walter Maps community forum:

http://waltermaps.com/forums/forum/free-geodata
**Adding a map layer from Dropbox**

You add a map layer from Dropbox by pressing the button. The first time you connect to Dropbox, you log in using your Dropbox username and password. The next time you add data from Dropbox, a hidden connection key is used for automatically logging in. For security reasons, you might want to remove the connection to Dropbox (i.e. remove the hidden connection key). This is made in the General Settings section of Walter Maps as described earlier.

After establishing the connection to Dropbox, you navigate to your dropbox folders and files in the usual Walter Maps scroll list. Only files valid to Walter Maps are shown in the Dropbox scroll list. Valid files are:

- ***.shp**, i.e. vector files in ESRI shape format. For the shape file to be visible, it has to be accompanied by its required *.shx* and *.dbf* files. These two files are not visible for you as a user, but will automatically be transferred to Walter Maps when transferring the *.shp* file.
- ***.jpg**, i.e. raster files in JPEG format. For the JPEG file to be visible, it has to be accompanied by its required *.jgw* file, i.e. the file that defines one of the corner coordinates and the pixel resolution. Even though a *.wld* file contains the same information as the *.jgw* file, the *.wld* file is **not valid**. If resolution pyramids are present, i.e. an *.jpg.aux.xml* file and an *.rrd* file, they will automatically be transferred along with the JPEG-file. The accompanying files will not be visible for the user.
- ***.tif**, i.e. raster files in TIFF or GeoTIFF format. If resolution pyramids are present, i.e. an *.tif.aux.xml* file and an *.rrd* file, they will automatically be transferred along with the TIFF-file or GeoTIFF file. If the TIFF file is a plain TIFF file without spatial awareness built in, it has
to be accompanied by a *.tfw file, i.e. the file that defines one of the corner coordinates and the pixel resolution. If not, the raster will be misplaced in your map. The accompanying files will not be visible for the user.

- *wmwmx*, i.e. an xml-file describing the capabilities of an online web map source (WMS, WMTS or TMS). The format of the *wmwmx* file is described in the Creating your own web map definition files section.

- *wm_layer.zip*, i.e. any number of zipped Walter Maps map layers. The zip file can contain *.shp, *.jpg, *.tif, and *wmwmx* files (and their accompanying files). If the Walter Maps verification for one of the files fails, none of the files will be added to the map.

- *wm_map.zip*, i.e. a Walter Maps map package file, containing any number of map layers and a map document defining the symbology and behaviour of the map layers. The map package file is an open format and is described in the Walter Maps map package and The map document file sections.

- *wmtxt*, i.e. a text file containing point coordinates. The format is described in the Import/export vector format section.

- *.svg*, i.e. a user defined Scalable Vector Graphics file, to be used as a point symbol when visualising point vector layers. ! Note that gradients and strokes will not be rendered correctly in Walter Maps. Strokes can easily be transformed to paths, which are rendered nicely by Walter Maps.

- *wm_symbols.zip*, i.e. any number of zipped *.svg* files.

- *lang*, i.e. a user defined language definition file for translating Walter Maps to other languages than English.
and Swedish. To create your own language file, please see the Translating Walter Maps to your own language section. Additional language files can be found at http://waltermaps.com/forums.

**Adding a map layer from your internal device storage**

When running Walter Maps on an Android device or a Windows 10 desktop/laptop computer, you can add map layers and files from your internal device storage by pressing the 
button. You can add the same files as described in the Add from Dropbox section above.

On Android devices you have access to the folder /sdcard and its sub-folders.

On Windows 10 desktop/laptop computers you have access to the user_maps folder (residing in the Walter Maps installation folder) and its sub-folders. For easy access to the user_maps folder, please follow this instruction:

- Install Walter Maps
- Open Explorer (i.e. Windows file explorer)
- In Explorer, open the folder
  - c:users[your username]AppDataLocal or
  - simply [your username]AppDataLocal
- In the Explorer search field, enter the text user_maps
- When the folder user_maps is found, right-click on the found folder and choose Copy.
- Navigate to any folder where you want to paste the user_maps folder as a shortcut, e.g. navigate to your Documents folder or Desktop folder
- Right-click and choose Paste shortcut
- Use this folder shortcut for the files you would like to expose to Walter Maps
Adding a map from an URL address

You can download data to Walter Maps from an URL address by pressing the button. You can only download the following file types:

- *.wm_layer.zip
- *.wm_map.zip
- *.wm_symbols.zip

These file types are described in the Adding a map layer from section above. The address you enter in the input dialogue box should look like:

http://www.abcd.efg/example.wm_map.zip

In Walter Maps Pro you can also use password protected URLs, which should have the following syntax:

http://login:password@www.abcd.efg/example.wm_map.zip
Export layers
You can export map layers by pressing the button. You exit the Export Layer mode by pressing the button.

The following export possibilities are available:

- Export visible layers within your visible map to a single raster layer
- Save layer(s) to internal device storage (only applicable for Android devices and Windows 10 desktop/laptop computers)
- Save layer(s) to Dropbox
- Send layer(s) by e-mail

Export visible layers within your visible map to a single jpeg raster layer
In Walter Maps Pro you can export the present map rendering to a jpeg raster file (*.jpg) by pressing the button. This might be useful for quickly sharing geographic data with other people who might lack the skills to use GIS software. After export, the map
layer will be added to your map as the top layer. The exported layer can later be exported/saved by other exporting/saving functions.

**Save layer(s) to Dropbox**

You export a map layer to Dropbox by pressing the button. The first time you connect to Dropbox, you log in using your Dropbox username and password. The next time you save data to Dropbox, a hidden connection key is used to automatically log you in. For security reasons you might want to remove the connection to Dropbox (i.e. remove the hidden connection key). This is made in the General Settings section of Walter Maps as described earlier.

You can either save the active layer or the complete map including symbology. If you choose to save your active layer, the layer will be zipped and saved into a single file named:

\[ \text{WALTERMAPS}_\text{YYYYMMDD}_\text{HHMMSS}.\text{wm}_\text{layer}.\text{zip} \]

If you choose to save the complete map, all your layers will be saved into a single file named:

\[ \text{WALTERMAPS}_\text{YYYYMMDD}_\text{HHMMSS}.\text{wm}_\text{map}.\text{zip} \]

If running Walter Maps Pro and saving a single active vector layer, you can also save the vector layer as a text or an MS Excel file. The text or spreadsheet file will contain all features and their attributes, and for point vector files the actual point coordinate will be saved. For line or polygon vector files the geometry centroid coordinate will be saved. The text and spreadsheet files will be named:

\[ \text{WALTERMAPS}_\text{YYYYMMDD}_\text{HHMMSS}.\text{wmtxt} \]

and
**WALTERMAPS_YYYYMMDD_HHMMSS.xls**

These file formats are described in the **Import/export vector format** section.

**Save layer(s) to internal device storage**

When running Walter Maps on an Android device or a Windows 10 desktop/laptop computer, you can save map layers to your internal device storage by pressing the button. You can save the same type of files as saving to Dropbox.

On Android devices you have access to the folder `/sdcard` and its sub-folders.

On Windows 10 desktop/laptop computers you have access to the `user_maps` folder, as described in the **Adding a map layer from your internal device storage** section.

**Send layers by e-mail**

You can e-mail the same type of files as when you save data to Dropbox by pressing the button. ! **Please note that**, if the file size exceeds the maximum e-mail size set in the General Settings section, the e-mail will fail (default max size is 20 Mbyte). ! **Please also note** that you have to install and set up an e-mail client outside of Walter Maps prior to e-mailing from Walter Maps.
Change symbology for a map layer
You can change properties, i.e. symbology, scale interval etc., for your map layers by pressing the button. You return to Manage Map Layers mode by pressing the button.

Each layer gets an initial setting called Layer Setting 1. The following buttons appear in the Layer Settings mode:

- Edit active layer setting
- Delete active layer setting
- Add layer setting
- Reorder active layer setting
Edit active layer setting

When pressing the button a scroll list with layer settings is displayed. Shared settings for all types of map layers are:

- Layer setting name, i.e. a user defined name for your layer setting.
- Transparency
- Min scale, i.e. the minimum map scale where rendering starts for the layer
- Max scale, the maximum map scale where rendering ends for the layer

For raster layers you can also edit the following settings:

- Transparent colours, i.e. what colours that should be rendered fully transparent. You either define the colours as a list of RGB triplets or as a list of single values, e.g. “0,0,0 33,14,234” or “1 4 34 255” respectively. The first definition of transparency tells Walter Maps to render the pixels with these two RGB-values fully transparent. The second definition tells Walter Maps to render pixels with the RGB values “1,1,1”, “4,4,4”, “34,34,34”, and “255,255,255” (or grey scale values 1, 4, 34, and 255) fully transparent.
- Threshold transparent colour, i.e. what colours above or below a certain pixel value that should be rendered fully transparent, e.g. “>180” or “<20” respectively.
Shared settings for all vector layers are:

- Layer selection, i.e. an SQL-query based on your attribute values, selecting what features to be rendered in the map, e.g.
  - AFFILIATION = “FRIEND” AND UNIT="BATTALION"
  - HEIGHT = 20 AND SPECIES = “Scots pine”
  The layer selection SQL-query is what differentiates the multiple layer settings from each other.
- Label colour, i.e. what colour should be used for labels.
- Label size, i.e. the height of the labels.
- Label size in meters or in ‰, i.e. whether the label size should be in meters or in ‰ of the screen height.
- Label field, i.e. what attribute field should be used for the labels.

Shared settings for line and polygon vector layers are:

- Line type, i.e. solid or none
- Line colour
- Line width
- Line width in meters or in ‰, i.e. whether the line width should be in meters or in ‰ of the screen height.

The remaining settings for polygon vector layers are:

- Fill type, i.e. solid or none
- Fill colour
The remaining settings for point vector layers are:

- Symbol. Opens a new scroll list for choosing your point symbol. You can filter your symbols by a search string by pressing the button. Your uploaded user defined symbols are placed at the end of the symbols list. Therefore, it is a good idea to give your symbols searchable names that make sense, e.g. `nature_horse_animal_recreation_sport.svg`. 
• Symbol colour. Opens the colour selection dialogue box, for choosing your symbol colour. If you press the button, the symbol will be coloured in its original colours. ! **Please note** when creating your own symbols, gradients and strokes will not be rendered correctly in Walter Maps, so please refrain from using these SVG features.

![](image)

• Symbol size
• Symbol size in meters or in %₀, i.e. whether the symbol size should be in meters or in %₀ of the screen height.
Collecting and editing data

Collecting waypoints
All waypoints are saved in a shape-file called *wm_waypoints.shp*. The file is recreated automatically if deleted by the user.

In Map Display mode you can add waypoints manually or by present GPS position. To add a waypoint manually you press the ✶ button. Then you add a single point to your map by moving the Walter Maps cursor (cross-hair) to the waypoint’s position using one finger and then you touch the surface with another finger, while still holding down your first finger. On a Windows 10 desktop/laptop computer you can add points using a standard 2-button mouse. You drag the Walter Maps cursor (cross-hair) using the left mouse button, and place a point using the right mouse button (or double-click left mouse button). After placing the waypoint, you create it by pressing the ✶ button again. After creating the waypoint, a scroll list with present waypoint attributes is shown. There you can view and edit the attributes. To close the scroll list and save the attributes values you press the ✖ button. Attribute editing is described in detail in the Viewing and editing attributes section.
To create a waypoint at your present GPS position, you simply press the button.

You can change the symbology for your waypoints as described in the Change symbology for a map layer section.

The waypoints attribute naming follows an extended version of the US Military S.A.L.U.T.E report, and is extended to D.O.S.A.L.U.T.E which stands for:

- Description
- Originator
- Size
- Activity
- Location
- Unit identification
- Time and date
- Equipment

The following attributes exist in the waypoint file:

- D_TASK, which in Walter Maps Pro automatically will be assigned the TASK registered in the General Settings section.
- D_NAME, any string, e.g. sub task
- D_DESC1, free text
- D_PR_DESC1, probability evaluation for D_DESC1
- D_DESC2, free text
- D_PR_DESC2, probability evaluation for D_DESC2
- D_DESC3, free text
- D_PR_DESC3, probability evaluation for D_DESC3
- D_DESC4, free text
- D_PR_DESC4, probability evaluation for D_DESC4
• O_ORIGINAT, which in Walter Maps Pro automatically will be assigned the NAME registered in the General Settings section
• O_RELIAB, reliability evaluation of the originator
• S_SIZE
• S_PR_SIZE, probability evaluation for S_SIZE
• A_ACTIVITY
• L_LOCATION
• L_X, will automatically be assigned the waypoint x coordinate
• L_Y, will automatically be assigned the waypoint y coordinate
• L_RADIUS_M, i.e. the radius of the waypoint
• L_W84_LAT, will automatically be assigned the waypoint latitude value
• L_W84_LONG, will automatically be assigned the waypoint longitude value
• L_ACC_HORI, if the waypoint was created by the GPS it will be assigned the present GPS horizontal accuracy
• L_ALTITUDE, if the waypoint was created by the GPS it will be assigned the present GPS altitude value
• L_ACC_VER, The vertical accuracy, will not be automatically assigned
• L_DIR_DEG, to be used for possible directional value
• L_INCL_DEG, to be used for possible inclination value
• L_DISTANCE, to be used for possible distance value
• U_UNIT
• U_SYMBOL
• T_246060, will automatically be assigned the time for waypoint registration
• T_PR_TIME, probability evaluation for T_246060
- T_YYYYMMDD, will automatically be assigned the date for waypoint registration
- T_PR_DATE, probability evaluation for T_YYYYMMDD
- E_EQUIPMENT

**Collecting tracks**
All tracks are saved in the shape-files *wm_tracks_points.shp* and *wm_tracks_lines.shp*. These files are recreated automatically if deleted by the user. You start acquiring a track by pressing the button. You stop collection of track data by pressing the button. **Please note** that track recording will be interrupted if the Walter Maps app is minimized or is out of focus. When the app is in focus again, the track recording continues. If the Walter Maps fully exits, the track recording continues after restart.

You can change the symbology for your track lines and track points, as described in the Change symbology for a map layer section.

The *wm_tracks_points.shp* and the *wm_tracks_lines.shp* contain the same D.O.S.A.L.U.T.E fields as the waypoints, described in previous section. The *wm_tracks_points.shp* contains the following two additional fields:

- LINE_UUID, which correspond to the UUID in the *wm_tracks_lines.shp* file
- ORDER, which is the automatically registered point order. i.e. 0, 1, 2 ... n
Creating geometries

Only one vector layer at a time can be edited. How to choose the present geometry editing layer is described in the **Enable vector layer geometry editing** section. In Map Display mode you can create new vector geometries by pressing the ✏️ button. How to draw new geometries is described in the **Drawing** section. Pressing the ✏️ button during the drawing process (only visible in Walter Maps Pro ✏️) removes the last added point. To finish drawing you press the ✏️ button again. To cancel the drawing you press the ✗ button.

![Drawing example](image.png)

When drawing a polygon and a line, it might look like this:
Deleting geometries
You can delete geometries in the present editing layer by pressing the 🗑️ button. This starts a session where you select (or deselect) the geometries for deletion by touch or mouse click. To delete the selected geometries you press the 🗑️ button again. To cancel the deletion you press the ❌ button.

Splitting polygons
In Walter Maps Pro you can split polygons by temporary drawn lines or by lines in another line layer. Choosing what line layer to be used as the splitting layer is made in the Manage Map Layers mode, described in section Set a line layer as cutting layer for polygon splitting. You start a splitting session by pressing the ✂️ button. You either draw a temporary splitting line by hand (as described in section Drawing) or you select splitting lines from your cutting layer. You can also use a combination of temporary drawn lines and selected lines from your cutting layer, as shown in the image below.
By pressing the button you select all the splitting lines in the present cutting layer. To perform the actual splitting you press the button again, which in this example gives the following result:

**Merging lines and polygons**

In Walter Maps Pro you can merge lines and polygons. To start a merging session you press the button. Then you select the lines or polygons you want to merge. To perform the actual merging you press the button again. The first image below shows the merging selection and the second image shows the merging result.
The new geometry created after merging will inherit the attribute values from last geometry selected, i.e. the geometry selected using a darker cyan colour. If you select all geometries by pressing the button, deselect and then reselect one of the geometries to actively decide what geometry should be selected last.

**Exploding multi-lines and multi-polygons**

In Walter Maps Pro you can explode multi-lines and multi-polygons to single geometry lines and polygons. You start an exploding session by pressing the button. Then you select the
geometries you want to explode. Alternatively you can select all geometries by using the button. To perform the actual exploding you press the button again.

**Cleaning lines and polygons**

In Walter Maps Pro you can clean lines and polygons, i.e. remove identical line segments within single lines, remove spikes in polygons, and remove zero area polygons or zero area rings. You start a cleaning session by pressing the button. Then you select the geometries you want to clean. Alternatively you can select all geometries by using the button. To perform the actual cleaning you press the button again.
Viewing and editing attributes

Only one vector layer at a time can be edited. How to choose the present attribute editing layer is described in the Enable vector layer attribute viewing and editing section. In the following example, we will update data for some roads. In the image below one single main road and three arterial roads are shown.

To open the Show/Edit Attributes mode, i.e. open the attribute scroll list for a single feature (one of the roads in this example), you press the button in the default Map Display mode. Then you select your geometry of interest in the map.

The attribute scroll list for the selected road is shown in the next image.
To change a value for an attribute you make it active by scrolling, and then pressing it. An edit dialogue box will open, for changing the attribute value. For convenience, the button will automatically assign a UUID text string (e.g. ed734385-b8cd-4a75-b31b-035ee6e93afe) to the active attribute. Assigning a UUID to a none-text attribute will fail.

In Walter Maps Pro you have access to additional auto-assignment buttons, shown in the image below:
In Walter Maps Pro the additional auto-assignment buttons are:

- Assign time in user defined time format
- Assign date in user defined date format
- Assign geometry centroid x coordinate
- Assign copied x coordinate
- Assign geometry centroid y coordinate
- Assign copied y coordinate
- Assign geometry length or perimeter in user defined length unit
- Assign geometry area in user defined area unit
- Assign GPS altitude in meters
- Assign present compass direction of your device in degrees (0-360°).
- Assign present device surface pitch in degrees (-90° to 90°)
- Assign present device surface roll in degrees (-90° to 90°)

**Collecting photos and video clips**
If your shape file contains an attribute field called UUID and if the UUID field for your present feature contains a non-empty string (preferably, but not necessarily, a proper UUID), you can collect photos and video clips for your features. In the free version of
Walter Maps you can collect 1 photo and 1 video clip per feature. In Walter Maps Pro you can collect an unlimited number of photos and video clips per feature. If your present UUID field isn’t empty, the following buttons will be visible in Show/Edit Attributes mode:

- Take a photo
- Record a video (and sound) clip
- View, delete and rename collected photos and video clips
- Select a photo or video clip from your device photo album/folder, which makes it possible to collect photos or video clips from Wi-Fi connected external DSLRs etc.

The collected photos and video clips are linked to the features using the UUIDs registered in the UUID field. ! Please note that changing the UUID value of a feature after collecting a photo or video clip, will remove the connection between the feature and the photo or video clip.

When exporting all map layers at once to Dropbox, e-mail or your device storage, the photos and video clips will be added to the resulting Walter Maps map package (*.wm_map.zip). The internal structure of the Walter Maps map package and where to look for your photos and video clips is described in the Pairing media files to single points, lines or polygons section.
If you want to view, delete, or rename your photos and video clips, you open the View Media Files mode by pressing the button.

In this mode you can view (by clicking on the active media file), delete or rename your files. ! Please note that file removal cannot be undone. As you see in the example above, all media files collected with Walter Maps will be named using present date and time.

Open a web page using an attribute value
In Walter Maps Pro you can open a web page for an active attribute if its value is recognized as a valid URL, e.g. http://waltermaps.com or www.dianthus.se. If it is recognized as valid, you can open the address by pressing the button.
Using validating data collection forms

In many field data collection scenarios, you may want to guarantee the quality of the collected data, e.g. you won’t want to forget to collect data for certain fields, and you may only want to collect values within specific ranges or sets. In Walter Maps Pro it is easy to add that functionality by adding a rule-file to your shape file. The rule-file is a JSON text file and is created outside of Walter Maps. It is saved alongside your shape file, using the same name and with the *.rul extension. See the Creating validating data collection forms section for details how to create and use rule-files.
Viewing raster data values

In Walter Maps Pro you can view raster data values. Raster values can only viewed for one raster layer at a time. How to enable raster value display is described in the Enable viewing of raster layer pixel values section. To start a raster value display session you press the button. Then you drag and release the Walter Maps cursor (cross-hair) above the raster element of interest. The value(s) of the raster bands will be displayed at the bottom of the map.
Professional Toolbox functions

In Walter Maps Pro you access the professional toolbox functions by pressing the button in default map display mode.

The following toolbox functions exist at present:

- Distribute sample plots within a polygon or along a line

More toolbox functions will be added continually based on the demands in the Walter Maps User Forum at waltermaps.com.

Distribute sample plots

This tool distributes sample plots within polygons or along lines. After pressing the button you will be guided in the process of creating the plots. They are distributed by numbers or by a certain distance between the plots. The number or distance are manually chosen or given by a specified field within the feature class where the plots are distributed. In the example below, sample plots are distributed within a polygon and along a line.
File formats

Beside the supported geodata formats described in the Supported geodata formats section, the following file formats are used:

- Import/export vector point formats
  - *.wmtxt
  - *.xls
- Point symbols in SVG-format
  - *.svg
  - *.wm_symbols.zip
- Layer packages
  - *.wm_layer.zip
- Map packages
  - *.wm_map.zip
- Language extension file
  - *.lang

Import/export vector format

In Walter Maps Pro you can import/export point features in the wmtxt text format. You also export to MS Excel xls format (importing MS Excel files are not possible).

The wmtxt file is a simple semi-colon separated text file for point features (point geometries and their attributes).

The structure of the text file is as:

```
X;FIELDB;y;FIELDD;FIELDE
345;;234.6;33.8;
366;22;22;44;23.5
223;;334;55;TEXT EXAMPLE 2
334;text example 1;123;i;
```
Importing points in wmtxt file

For importing a wmtxt file, the following requirements must be fulfilled:

- All field names and values must be separated by semi colons “;”
- Field names can only contain the characters “A-Z”, “a-z”, “_”, and “0-9”
- The number of semi colons per row must always be equal to (number of fields -1), even for rows with missing values
- All floating point numbers shall use “.” as decimal delimiters
- The first row must consist of unique field names
- There must exist one Y and one X field (both upper and lower case allowed).
- There must not exist any row with missing X and Y values
- Missing values are allowed for other fields than the X and Y fields
- No other existing map layer in Walter Maps should have the same base name as the file you are about to import
- If the length of a text fields exceeds 253 characters, the text field will be shortened to 253 characters.

The wmtxt files can be imported from Dropbox, from your device (applicable to Android devices and Windows 10 desktop/laptop computers only), and from URL downloads.

**Please note** that the wmtxt can neither be included in Walter Maps map packages *.wm_map.zip, nor in Walter Maps layer packages *.wm_layer.zip.
Exporting vector data in wmtxt file format
All vector layers in Walter Maps, even line and polygon layers, can be exported to wmtxt format. They can be exported to Dropbox, your device (applicable to Android devices and Windows 10 desktop/laptop computers only), or to e-mail.

As X and Y values, the centroid is calculated for each geometry. The centroid is guaranteed to be within or on the boundary of the geometry. For line and polygon layers, beside the centroid, no geometry is exported.

Exporting vector data in xls file format
All vector layers in Walter Maps, even line and polygon layers, can be exported to MS Excel format (*.xls). The structure of the xls-file is the same as for the wmtxt format (semicolons omitted). The vector layers can be exported to Dropbox, to your device (applicable to Android devices and Windows 10 desktop/laptop computers only), or to e-mail.
Point symbols in SVG format

All points in a vector point layer are visualised by symbols in SVG format (*.svg). You can use your own point symbols and you download them from Dropbox or from your device (applicable to Android devices and Windows 10 desktop/laptop computers only). SVG symbols can be produced in, for example, Adobe Illustrator or in the open source software Inkscape. Walter Maps does not support all features of the SVG format. Therefore, in order for the SVG files to work with Walter Maps, please follow these simple technical guidelines:

- Convert all text to paths
- Convert all strokes to paths
- Do not use colour gradients, use only single colours

Walter maps symbol package

Walter Maps symbol package *.wm_symbols.zip is a simple compressed zip-file containing one or more SVG files. You can easily produce your own symbol packages by compressing all SVG files to the root of a zip file. Symbol packages containing files not recognized as SVG files will not be accepted by Walter Maps.

The symbol packages can be opened from Dropbox, your device (applicable to Android devices and Windows 10 desktop/laptop computers only), and from an URL-link.
Walter Maps layer package

Walter Maps layer package *.wm_layer.zip is a simple compressed zip-file containing one or more map layers and their required files. You can easily produce your own layer packages by compressing all required files to the root of a zip file. Layer packages containing files not recognized as map layers will not be accepted by Walter Maps.

In Walter Maps, when saving one of your layers to Dropbox, device (applicable to Android devices and Windows 10 desktop/laptop computers only), or to e-mail, a layer package will always be created and saved.

The layer packages can be opened from Dropbox, your device (applicable to Android devices and Windows 10 desktop/laptop computers only), and from an URL-link. Any layer package produced by Walter Maps Pro can be opened in the free version of Walter Maps.
Walter Maps map package
Walter Maps map package *.wm_map.zip is a compressed zip-file containing any number of map layers, symbologies for all layers, all media files (photos and videos clips), projection information, and all user symbols. In Walter Maps, when saving all your map layers to Dropbox, to your device (applicable to Android devices and Windows 10 desktop/laptop computers only), or to e-mail, a map package will automatically be created and saved.

The saved map package can be opened from Dropbox, your device (applicable to Android devices and Windows 10 desktop/laptop computers only), and from an URL-link. Any map package produced by Walter Maps Pro can be opened in the free version of Walter Maps.

The Walter Maps map package is an open format, i.e. you can produce your own map packages by hand or by your in-house GIS platform or systems. Please see the Building your own Walter Maps map packages section if you need to build your map packages from scratch.

Walter Maps language file
Walter Maps language files *.lang are simple text files, defining all text strings in Walter Maps in different languages. If you want to create your own language files, please follow the guidelines provided in the Translating Walter Maps to your own language section.

The Walter Maps language files can be opened from Dropbox or from your device (applicable to Android devices and Windows 10 desktop/laptop computers only).
Advanced usage

This section is intended for advanced users, wanting to customize Walter Maps to their specific needs or wanting to integrate Walter Maps with their own GIS platform or other systems.

Creating validating data collection forms

When editing attributes in Walter Maps, your field names and values are visible in a scroll list view. The only built-in validations while changing or adding values are that you can’t add non-numerical strings to number fields and you can’t add floating point numbers into integer number fields.

In many field data collection scenarios you will want to guarantee the quality of the data that is collected, e.g. you won’t want to forget to collect data for certain fields and you will only want to collect values within specific intervals or sets. In Walter Maps Pro it is quite easy to add that functionality by adding a rule-file to your shape file. If your shape file is called parcels.shp, your rule file must be called parcels.rul. The rule file cannot be created within Walter Maps. This is something you have to create in a standard text editor.

Rule file example using one set of rules

The rule file is a text file in JSON format, and must be coded in ordinary ANSI 8-bit text format (for now, UTF-8 text files are not supported). An example (parcels.rul) is shown in the image below.

```json
{
    "attributes": [
        "ZIP", "REP", "INCIDENT", "FLX", "DESIGN", "MARK", "DATE", "GROUP", "ID", "DESC"
    ],
    "rules": {
        "ZIP": {
            "default": "00000", "valid": true
        },
        "REP": {
            "default": "0", "valid": true
        },
        "FLX": {
            "default": "false", "valid": true
        },
        "DESIGN": {
            "default": "false", "valid": true
        },
        "MARK": {
            "default": "false", "valid": true
        },
        "DATE": {
            "default": "2023-01-01", "valid": true
        },
        "GROUP": {
            "default": "true", "valid": true
        },
        "ID": {
            "default": "1", "valid": true
        },
        "DESC": {
            "default": "Description", "valid": true
        }
    }
}
```

The first key “attributes” is a value-array of the fields in your shape file that you want to be visible and affected by your rules. Fields
that you don’t want to be visible should be omitted from the “attributes” value-array. The field/attribute names are case sensitive, i.e. the field/attribute names in the rule file must exactly match the field names in the corresponding dbf file.

The next keys “UUID”, “REF” etc., are the actual rules for each field. The rules are implemented as JSON key:value pairs. All keys and values must be text strings and must be surrounded by double quotes, except for the boolean values true and false.

**Basic rules**

- **unique** (true/false) if true the field value must be unique. The check is performed when editing/adding attribute values. **Please note**, this is not checked when editing your geometries, e.g. while splitting a polygon. Default false.
- **modifiable** (true/false) if false the field value cannot be modified once set. Default true.
- **editable** (true/false) if false the field is visible but not modifiable. Default true.
- **not_null** (true/false) if true the field cannot be empty and is only applicable to text fields. Default false.
- **not_zero** (true/false) if true a number field cannot be empty or cannot be 0. Default false.

**Default values**

If a field is not already set, you can specify what default value that field should be get when you view or edit the attributes for the first time. The key is **default** and the value can be any number, any string, or a default value auto-assigner, i.e. the values starting with **waltermaps** in the above example. As mentioned earlier, even the number values must be surrounded by double quotes.
The following auto-assigners can be used:

- `waltermaps_name` assigning the Name string registered in the General Settings section
- `waltermaps_task` assigning the Task string registered in the General Settings section
- `waltermaps_installation_id` assigning a string that contains the Walter Maps unique installation ID
- `waltermaps_uuid` assigning a UUID string
- `waltermaps_time` assigning a time string according to the current time format setting in the General Settings section
- `waltermaps_date` assigning a date string according to the current date format setting in the General Settings section
- `waltermaps_date_time` assigning a date + "_" + time string according to the current date and time format settings in the General Settings section
- `waltermaps_date_dd_space_MM_space_yyyy` assigning date string of format “31 01 2017”
- `waltermaps_date_dd_forslash_MM_forslash_yyyy` assigning date string of format “31/01/2017”
- `waltermaps_date_ddMMyyyy` assigning date string of format “31012017”
- `waltermaps_date_dd_hyphen_MM_hyphen_yyyy` assigning date string of format “31-01-2017”
- `waltermaps_date_yyyy_space_MM_space_dd` assigning date string of format “2017 01 31”
- `waltermaps_date_yyyy_forslash_MM_forslash_dd` assigning date string of format “2017/01/31”
- `waltermaps_date_yyyyMMdd` assigning date string of format “20170131”
- `waltermaps_date_yyyy_hyphen_MM_hyphen_dd` assigning date string of format “2017-01-31”
• **waltermaps_time_hhmmss_24** assigning time string of format “231855”
• **waltermaps_time_hh_colon_mm_colon_ss_24** assigning time string of format “23:18:55”
• **waltermaps_time_hh_dot_mm_dot_ss_24** assigning time string of format “23.18.55”
• **waltermaps_time_hhmmss_space_PMorAM** assigning time string of format “111855 PM”
• **waltermaps_time_hh_colon_mm_colon_ss_space_PMorAM** assigning time string of format “11:18:55 PM”
• **waltermaps_time_hh_dot_mm_dot_ss_space_PMorAM** assigning time string of format “11.18.55 PM”
• **waltermaps_centroid_x** assigning the x centroid of present geometry
• **waltermaps_centroid_y** assigning the y centroid of present geometry
• **waltermaps_length** assigning the length or the perimeter of the present geometry
• **waltermaps_area** assigning the area of the present geometry

**Value constraints**
The following keys are used for value constraints:

• **regexp** constraining a string value by a regular expression
• **set** constraining a string or a number value to a predefined set of values e.g. [“1”, “2”, “3”] or [“DODGE”, “FIAT”, “TOYOTA”, “VOLVO”]
• **range** constraining a number value to a specific interval, e.g. [“0.1”, “62”]
• **ge** constraining a number value to be greater or equal to a specific value
- **gt** constraining a number value to be greater than a specific value
- **le** constraining a number value to be lesser or equal to a specific value
- **lt** constraining a number value to be lesser than a specific value

**Aliases and message strings**
The following keys are used for aliases and message strings in order to make data collection more understandable:

- **alias** changing the field name to a more understandable name
- **description** adding a description to the input dialog box while editing the field
- **error_message** adding a more understandable error message, displayed in the error message dialog box

**Example file using one set of rules**
The above rule file example can be downloaded at:

http://waltermaps.com/wp-content/tutorials/waltermaps_rule/parcels.rul

and **please note**! that the JSON file must be coded in ANSI 8-bit text format (for now, UTF-8 text files are not supported).

**Rule file example using a multiple set of rules**
You can also create a rule file with a multiple set of rules. The first time the user views/edits the data in a session, the user will be asked what rule should be applied. In the example below the user will be asked whether to “Change basic data” or to “Change more things”. The multiple rule file is recognized by Walter Maps by the first key:value pair “multirule”: true. The rules are given in the “rules” array, containing one to any number of rules.
Example file using a multiple set of rules

The above rule file example can be downloaded at:

http://waltermaps.com/wp-content/tutorials/waltermaps_rule/parcels_multirule.rul

and please note! that the JSON file must be coded in ANSI 8-bit text format (for now, UTF-8 text files are not supported).

Rule file validation

Structural validation of your rule file can be made at:

https://jsonlint.com/.

Once structurally validated, you should validate the functionality in Walter Maps (no validation schema exists due to the fully dynamic content of the JSON file).
Building your own Walter Maps map packages

Walter Maps can easily be used as a field data collection tool for entrepreneurs, organizations and systems in need of the functionality it provides. Through our open and well documented map package format and map document format, you can easily create your own collection of maps for distribution to colleagues, partners, customers, and friends. Of course, you can use Walter Maps to produce your map packages, but sometimes you might want produce the map packages automatically from your own systems. The formats for the map package and the map document are based on open standards, i.e. they are based on the zip-file format, standard text files, and JSON text files. The map package consists of a single zip file where you put all your map layers and other needed configuration files.

Future safe

The Walter Maps map document is built to cope with future expansions, i.e. a lot of the fields in the map document are not used initially. Only a few and small future format changes will therefore be required. If future format changes occur, your old document versions will still work due to proper version handling in Walter Maps.

Naming convention

The map package should always end with *.wm_map.zip in order to be recognized by Walter Maps. When exporting a map package from Walter Maps it will always be named like:

WALTERMAPS_20170323_123838.wm_map.zip

i.e. it will be named by the current device date and time. When producing your own map packages, you can name them differently, as long as they end with the extension *.wm_map.zip.
Contents of the map package

The map package should at a minimum contain the following files:

- The map document, i.e. `waltermapsdoc.json`
- Walter Maps projected coordinate system definition file, i.e. `waltermapsdeviceprojection.txt`
- At least one map layer, i.e. GeoTIFF/TIFF-, JPEG-, Shape-, or WMS/WMTS-definition files
  - For GeoTIFF:
    - Required file:
      - `*.tif`
    - Optional files:
      - `*.rrd` and `*.aux`
      - `*.tif.aux.xml`
  - For ordinary TIFF:
    - Required files:
      - `*.tif`
      - `*.tfw`
    - Optional files:
      - `*.rrd` and `*.aux`
      - `*.tif.aux.xml`
  - For JPEG:
    - Required files:
      - `*.jpg`
      - `*.jgw`
    - Optional files:
      - `*.rrd` and `*.aux`
      - `*.jpg.aux.xml`
  - For WMS/WMTS:
    - Required file:
      - `*.wmwmx`
For Shape:
- Required files:
  - *.shp, *.shx, and *.dbf
- Optional files:
  - *.prj

The map package can also continue any number of SVG point symbols (*.svg) placed in the root folder user_symbols.

The required and optional files shall all be placed in the root of the compressed map package zip file.

The projection definition file
The file waltermapsdeviceprojection.txt contains a single line of text identifying the current projected coordinate system. The text string you should use is the human readable EPSG string identifier (do not use the EPSG numerical ID). For example, the file waltermapsdeviceprojection.txt can contain the string “sweref99 tm” or “nad83(harn) / new york central” (without quotation marks). For a full list of projected coordinate systems please visit:

http://spatialreference.org/

The map document file
The file waltermapsdoc.json is the glue organizing and visualising your present map layers. In the examples below, comments will be added after two slashes “//” even though they are not allowed in JSON files. Mandatory fields will be marked with M, non-used fields for future expansion will be marked with F, valid ranges will be given after letter R, and default mandatory values will be given after letter D. The only valid data types are integers (e.g. 3 or 48), doubles (e.g. 3.14159265), booleans (i.e. true or false), or text strings. Max and Min below denotes the maximum and minimum value for the present data type.
**JSON schema verification**

When developing your JSON file export, you can verify your JSON file against the schema file:

http://waltermaps.com/waltermapsdoc_schema.json

Verification can be made at:

http://www.jsonschemavalidator.net/

where you paste the schema to the left and your JSON file to the right. **Please note!** valid values, ranges, logic between keys, etc. are not validated in this schema validation.

**A basic none-sense example**

The `waltermapsdoc.json` is a human readable utf-8 encoded JSON text file containing the information of all layers in the map package and how to visualize them. In it its simplest form it looks like:

```json
{
  "WalterMapsDoc": {
    "m_version": 1, // map document version, integer, M, D:1
    "m_scale": 50000, // map scale, integer, M, R:100-Max
    "m_centre_x": 805070.14, // map centre x, double, M, R:Min-Max
    "m_centre_y": 7316487.44, // map centre z, double, M, R:Min-Max
    "m_centre_z": 0, // map centre z, double, M, F, D:0
    "m_layers": [], // Array of map layers, in this case an empty array
    "m_epsg": "" // Projected coord system string, M, F, D:""
  }
}
```

This simple example actually makes no sense as your map package should contain at least one map layer.
Example 1, one raster layer
The map package for example 1 can be downloaded from:

http://waltermaps.com/wp-content/tutorials/waltermaps_map_package_and_document_format/map_package_1_raster.wm_map.zip

The map package can also be directly opened in Walter Maps Pro through direct URL download as described in the Adding a map from an URL address section.

The projected coordinate system identifier file waltermaps-deviceprojection.txt contains the single text line:

SWEREF99 TM

The map in this example looks as below and contains only one GeoTiff raster layer called topomap.tif.
The symbology definitions (WalterMapsLayerSymbology section in the JSON file below) for raster layers and vector layers are exactly the same. Even though some of the fields are only valid for raster layers and vice versa, they should not be omitted in the JSON file.

The map document waltermapsdoc.json for this map package looks like this:

```json
{
  "WalterMapsDoc": {
    "m_version": 1, // Map document version, integer, M, D:1
    "m_scale": 25889, // Map scale, integer, M, R:100-Max
    "m_centre_x": 805626.3946720988, // Map centre x, double, M, R:Min-Max
    "m_centre_y": 7316273.550794094, // Map centre y, double, M, R:Min-Max
    "m_centre_z": 0, // Map centre z, double, M, F, D:0
    "m_layers": [
      "WalterMapsLayer": {
        "m_version": 1, // Map layer version, integer, M, D:1
        "m_name": "topomap.tif", // Layer file name, string, M
        "m_path": "maps\topomap.tif", // Full path to layer, string, always starts with
        // the folder "maps\". The folder delimiter "\\" is always escaped by "\\", M
        // Note that the DOS/Windows back slash path delimiter "\\" is not allowed, and not "\"
        "m_uuid": "9838f62e-51dd-48cd-9547-61694098a0ff", // Unique identifier for the layer, string, // M, D:uuid
        "m_type": 2, // Layer type, integer, M, R:1-2
        // 1 = vector
        // 2 = raster
        "m_subType": 4, // Layer sub type, integer, M, R:1-5 and 7
        // 1 = vector point
        // 2 = vector line
        // 3 = vector polygon
        // 4 = Raster Tiff or GeoTiff
        // 5 = Raster JPEG
        // 7 = Raster WMS or WMTS
        "m_order": 0, // Map layer order, integer,
        // M, R:0-(m_numLayers-1)
        // 0 = top layer in the map
        "m_selected": false, // Layer is selected or not, boolean, M, F,
        "m_visible": true, // Layer is visible or not, boolean, M
        "m_geometryEditable": false, // Layer choosen for edit, boolean, M
        // Only one of the layers can be choosen for edit
        "m_attributeEditable": false, // For future expansion, boolean, M, F, D:false
        "m_infoable": false, // Vector layer attributes can be shown and
        // edited, or Raster layer values can be shown, // boolean, M. This setting can only be
        // true for one of the layers, regardless of type
        "m_isClipLayer": false, // Vector line layer can be used for clipping of
        // polygon layers, boolean, M
        // This setting can only be true for one of the // line layers
        "m_valid": true, // Layer is valid or not, boolean, M, D:true
        "m_epsg": "-", // EPSG for layer, string, M, F, D:"-"
      }
    ]
  }
}
```
// Layer is a so called quick alternating layer, boolean, M
"m_browsable": false,

// Vector layer is a quick info layer, string, M, boolean, M, F, D:""
"m_quickInfoAttribute": "",

// Array of symbolologies for the present layer, in this case 1 symbolology
"m_symbology": [

  // Map layer symbolology version, integer, M, D:1
  "m_version": 1,

  // Name of symbolology, string, M, R:"" - any string
  "m_name": "",

  // Attribute query, string, M, R:"" - any valid query string
  "m_selectionQuery": "",

  // Placement, M, F, D:""
  "m_placement": "",

  // Array of fill colors for polygons. At present one and only one fill color is used regardless of layer type
  "m_fillColors": [
    // Red value for fill color, integer, M, R:0-255
    "m_red": 221,

    // Green value for fill color, integer, M, R:0-255
    "m_green": 221,

    // Blue value for fill color, integer, M, R:0-255
    "m_blue": 221,

    // Alpha value for fill color, integer, M, R:0-255
    "m_alpha": 255
  ],

  // Array of line colors for polygon borders and lines. At present one and only one line color is used regardless of layer type
  "m_lineColors": [
    // Red value for line color, integer, M, R:0-255
    "m_red": 51,

    // Green value for line color, integer, M, R:0-255
    "m_green": 51,

    // Blue value for line color, integer, M, R:0-255
    "m_blue": 51,

    // Alpha value for line color, integer, M, R:0-255
    "m_alpha": 255
  ],

  // Array of symbol colors for points
  // At present one and only one symbol color is used regardless of layer type
  // Note that: if color for symbol is {0,0,0,0} the symbol will be colored with the original colors of the SVG-symbol
  "m_symbolColors": [
    // Red value for symbol color, integer, M, R:0-255
    "m_red": 51,

    // Green value for symbol color, integer, M, R:0-255
    "m_green": 51,

    // Blue value for symbol color, integer, M, R:0-255
    "m_blue": 51,

    // Alpha value for symbol color, integer, M, R:0-255
    "m_alpha": 255
  ],

  // Array of raster statistics, mean, leave this empty
  "m_mean": [],

  // Array of raster statistics, stddev, leave this empty
  "m_stdDev": [],

  // Array of raster statistics, min, leave this empty
  "m_min": [],

  // Array of raster statistics, max, leave this empty
  "m_max": [],

  // Array of raster statistics, userMin, leave this empty
  "m_userMin": [],

  // Array of raster statistics, userMax, leave this empty
  "m_userMax": [],

  // Array of raster statistics, stdDevFactor, leave this empty
  "m_stddevFactor": [],

  // Raster band RGBA-mapping 1, integer, M, F, D:1
  "m_band1": 1,

  // Raster band RGBA-mapping 2, integer, M, F, D:2
  "m_band2": 2,

  // Raster band RGBA-mapping 3, integer, M, F, D:3
  "m_band3": 3,

  // Raster band RGBA-mapping 4, integer, M, F, D:4
  "m_band4": 0,

  // Raster band RGBA-mapping alpha, integer, M, F, D:4
  "m_bandAlpha": 4,

  // Vector of color ramp colors, leave this empty
  "m_colorRampColors": [],

  // Layer is visible from scale, integer, M, R:1-Max
  "m_visibleFromScale": 1,

  // Layer is visible to scale, integer, M, R:1-Max
  "m_visibleToScale": 1000000000,

  // Layer transparency, integer, M, R:0-100
  "m_transparency": 0
]
"m_transparentRasterColor": "", // Single or RGB values for full raster transparency, // string, M, D:"" see the Change symbology for a map layer section // e.g. "1 34 235" or "1,1,1 34,34,34 235,235,235"
"m_transparentThreshold": "", // Threshold gray scale value for full raster transparency, // string, M, D:"" // e.g. "<34" or "<210"
"m_fillType": 1, // Polygon fill type, integer, M, R:0-1 // 0 = none // 1 = solid
"m lineWidth": 1, // Line width for polygon borders or lines, // integer, M, R:0-100
"m_isLineWidthProc": false, // Line width is in meters (false) or % of screen width (true), // boolean, M
"m_lineType": 1, // Line type for polygon borders or lines, // integer, M, R:0-1 // 0 = none // 1 = solid
"m_symbolSize": 3, // Size for point symbols, integer, M, R:0-100
"m_isSymbolSizeProc": false, // Symbol size is in meters (false) or % of screen width (true), // boolean, M
"m_symbolType": "", // Full path for the present SVG point symbol, // string, M
"m_stretchType": 0, // Raster stretch type, integer, M, F, D:0
"m_shallLabel": false, // Shall label (note for future use), boolean, M, F, D:false
"m_labelField": "—", // Label attribute field, string, M, R:"—" – any valid // attribute name // If a valid attribute name is given, the layer will be labelled
"m_labelPlacement": "", // Label placement, string, M, F, D:""
"m_labelTypeface": "Arial", // Label typeface, string, M, F, D:"Arial"
"m_labelSize": 8, // Label text height, integer, M, R:0-100
"m_isLabelSizeProc": false, // Label text height is in meters (false) or % of screen width (true), M
"m_labelColors": [ // Array of label colors. // At present one and only one // label color is used regardless of layer type
  {
    "m_red": 51,
    "m_green": 51,
    "m_blue": 51,
    "m_alpha": 255
  }
],
"m_labelBold": false, // Is label text bold, boolean, M, F, D:false
"m_labelItalic": false // Is label text italic, boolean, M, F, D:false
} // End WalterMapsLayerSymbology

} // End m_symbology array
"m_readFromURL": "", // Layer data pull adress for future use, string, // M, F, D:""
"m_writeToURL": "" // Layer data push adress for future use, string, // M, F, D:""
} // End WalterMapsLayer

} // End m_layers array
"m_epsg": "" // EPSG for document, string, M, F, D:""
} // End WalterMapsDoc
} // End JSON file
Example 2, one raster layer and one vector layer

The map package for example 2 can be downloaded from:

http://waltermaps.com/wp-content/tutorials/waltermaps_map_package_and_document_format/map_package_1_raster_1_vector.wm_map.zip

The map package can also be directly opened in Walter Maps through direct URL download as described in the Adding a map from an URL address section.

In this example, the projected coordinate system identifier file waltermapsdevice-projection.txt contains the following single text line:

OWN_PROJ

That line tells Walter Maps that this map package provides its own projection definition. If that’s the case, the definition must always be provided in the file waltermaps.prj which is an ESRI projection definition file. In this particular example the file contains the following text information:

PROJCS["OWN_PROJ",GEOGCS["GCS_SWEREF99",DATUM["D_SWEREF99",SPHEROID["GRS_1980",6378137.0,298.25722101]],[PRIME_MERIDIAN["Greenwich",0.0]],UNIT["Degree",0.0174532925199433]],PROJECTION["Transverse_Mercator"],PARAMETER["False_Easting",500000.0],PARAMETER["False_Northing",0.0],PARAMETER["Central_Meridian",15.0],PARAMETER["Scale_Factor",0.9996],PARAMETER["Latitude_Of_Origin",0.0],UNIT["Meter",1.0]]

The map in example 2 looks as below and contains the GeoTiff raster topomap.tif and a vector polygon layer called estate.shp.
The map document `waltermapsdoc.json` for this map package looks like (already explained fields are omitted):

```json
{
  "WalterMapsDoc": {
    ...
    "m_layers": [                         // Array of 2 map layers
      "WalterMapsLayer": {              // Map layer 1 (vector polygon layer)
        "m_version": 1,
        "m_name": "estate.shp",
        "m_path": "maps/estate.shp",
        "m_order": 0,                     // Top most layer in map
        "m_symbology": [                // Array of symbologies for the present layer,
          "WalterMapsLayerSymbology": {  // in this case 1 symbology
            "m_version": 1,
            "m_name": "My default symbology",
            "m_selectionQuery": "",    // This time it is given a name
            "m_fillColors": [         // No selection query is set, i.e.
              {                     // all polygons are shown
                "m_red": 0,
                "m_green": 107,
                "m_blue": 179,
                "m_alpha": 255
              }
            ],
          }
        ]
      }
    ...
  }
}
```
"m_lineColors": { // Lines are in yellow color
  "m_red": 255,
  "m_green": 236,
  "m_blue": 1,
  "m_alpha": 255
},
...
"m_transparency": 50, // 50% transparency
...
"m_lineWidth": 5,
"m_isLineWidthProc": false, // Line width is 5 meters and not % of screen width
...
"m_labelField": "TEXT", // Polygons will be labelled with the content of the "TEXT" field
...
"m_labelSize": 36,
"m_isLabelSizeProc": false, // Label text height is 40 meters and not % of screen width
"m_labelColors": { // Labels are in red color
  "m_red": 225,
  "m_green": 3,
  "m_blue": 80,
  "m_alpha": 255
},
...
"WalterMapsLayer": { // Map layer 2 (raster layer)
  "m_version": 1,
  "m_name": "topomap.tif",
  "m_path": "maps\topomap.tif",
  ...
  "m_order": 1,
  ...
  "m_symbology": [ // As in example 1 above
    "WalterMapsLayerSymbology": { // As example 1 above
      ...
    },
    "m_readFromURL": "",
    "m_writeToURL": ""
  ]
},
"m_epsg": "" // End WalterMapsDoc
Example 3, a complex one

This map package contains one raster layer and two vector layers with complex symbology, own SVG point symbols and media files (photos and video clips). The map package can be downloaded from:

http://waltermaps.com/wp-content/tutorials/waltermaps_map_package_and_document_format/map_package_1_raster_2_vector_multiple_symbology.wm_map.zip

The map package can also be directly opened in Walter Maps through direct URL download as described in the Adding a map from an URL address section.

The map in example 3 looks as below and contains the GeoTiff raster *topomap.tif*, the polygon vector layer *estate.shp*, and one vector point layer called *POI.shp*. 
Multiple symbology and user symbols

This example shows multiple symbologies for one of the vector layers. Vector layers can be symbolised in more than one way, depending on one or more fields and their values. In this particular case, the POI layer is symbolised in three ways depending on the field “TYPE” in and its values (’FRIENDS’;’TRUCK’;’REINDEER’). Three SVG-symbols are used for the symbology, where two of the symbols, user_symbols/reindeer.svg and user_symbols/friends.svg are user defined symbols. Please note, all the user defined symbols must be stored in the map package zip file in the folder user_symbols.

Pairing media files to single points, lines or polygons

In this example, the two media files

album/POI/560e4475-fa62-45a7-bde3-006538afc21c/WM_20170424_203521.jpg
and
album/POI/51746716-3ec5-44cf-8fc3-6075c69e02b8/WM_20170424_203433.mp4

are paired with two of the points, the most southern truck and the reindeer respectively. In order to pair media files with a particular feature, the vector layer has to have a string field named “UUID”. For the particular point, line, or polygon feature, the field “UUID” has to be filled with a unique value (preferably, but not necessarily, filled with a properly generated UUID). The media files should be stored in a folder with the name built from “album”, the base name of the particular vector file (e.g. “POI” from the file name POI.shp), and the UUID field value (e.g. “560e4475-fa62-45a7-bde3-006538afc21c”). The resulting path for the media file will be album/POI/560e4475-fa62-45a7-bde3-006538afc21c/WM_20170424_203521.jpg.

The actual media file does not necessarily have to be named as WM_20170424_203521.jpg. Any valid file name will do fine.
The map document

The map document `waltermapsdoc.json` for this map package looks like (already explained fields are omitted):

```
{
  "WalterMapsDoc": {
    "m_layers": [                 // Array of 3 map layers
      {                           // Layer 1 as in example 2 above
        "WalterMapsLayer": {
          "m_version": 1,
          "m_name": "POI.shp",
          "m_path": "maps\POI.shp",
          "m_uuid": "f91b6ec-b150-4ffc-ae7d-0f7c9b883103",
          "m_type": 1,
          "m_subType": 1,
          "m_order": 0,
          "m_symbology": [               // Array of 3 symbologies
            {                           // Start symbology no 1
              "WalterMapsLayerSymbology": {
                "m_version": 1,
                "m_name": "Trucks",
                "m_selectionQuery": "TYPE='TRUCK'",
                "m_symbolColors": {        // Symbol is filled with a single red color
                  "m_red": 227,
                  "m_green": 0,
                  "m_blue": 27,
                  "m_alpha": 255
                }
              }
            },
            {                           // Start symbology no 2
              "WalterMapsLayerSymbology": {
                "m_version": 1,
                "m_name": "Friends",
                "m_selectionQuery": "TYPE='FRIENDS'",
                "m_symbolColors": {        // {0,0,0,0} The symbol will
                  "m_red": 0,
                  "m_green": 0,
                  "m_blue": 0,
                  "m_alpha": 0
                }
              }
            }
          ]
        }
      },
      {                           // Map layer 3 {vector point layer}
        "WalterMapsLayer": {
          "m_version": 1,
          "m_name": "POI.shp",
          "m_path": "maps\POI.shp",
          "m_uuid": "f91b6ec-b150-4ffc-ae7d-0f7c9b883103",
          "m_type": 1,
          "m_subType": 1,
          "m_order": 0,
          "m_symbology": [               // Array of 3 symbologies
            {                           // Start symbology no 1
              "WalterMapsLayerSymbology": {
                "m_version": 1,
                "m_name": "Trucks",
                "m_selectionQuery": "TYPE='TRUCK'",
                "m_symbolColors": {        // Symbol is filled with a single red color
                  "m_red": 227,
                  "m_green": 0,
                  "m_blue": 27,
                  "m_alpha": 255
                }
              }
            },
            {                           // Start symbology no 2
              "WalterMapsLayerSymbology": {
                "m_version": 1,
                "m_name": "Friends",
                "m_selectionQuery": "TYPE='FRIENDS'",
                "m_symbolColors": {        // {0,0,0,0} The symbol will
                  "m_red": 0,
                  "m_green": 0,
                  "m_blue": 0,
                  "m_alpha": 0
                }
              }
            }
          ]
        }
      }
    ]
  }
}
```
Walter Maps – User’s Guide

"m_green": 0,
"m_blue": 0,
"m_alpha": 0
},
,
"m_symbolSize": 80,
"m_isSymbolSizeProc": false,  // Symbol size is 80 m and not % of screen
// width
"m_symbolType": "user_symbols\friends.svg",
// User provided symbol
// Present symbol is a user provided
// symbol. Note that the path for
// user symbols shall start
// with the folder "user_symbols\"

,...
// End symbology no 2
},
,
"WalterMapsLayerSymbology": {
// Start symbology no 3
"m_version": 1,
"m_name": "Reindeers",
"m_selectionQuery": "TYPE='REINDEER'",  // Layer selection

"m_symbolColors": {
// (0,0,0,0) The symbol will
// be colored with the original
// colors of the SVG-symbol

"m_red": 0,
"m_green": 0,
"m_blue": 0,
"m_alpha": 0
},
,
"m_symbolSize": 80,
"m_isSymbolSizeProc": false,  // Symbol size is 80 m and not % of screen
// width
"m_symbolType": "user_symbols\reindeer.svg",
// User provided symbol
// Present symbol is a user provided
// symbol. Note that the path for
// user symbols shall start
// with the folder "user_symbols\"

,...
"m_labelField": "TEXT",  // Points will be labeled with
// the content of the "TEXT" field

,...
"m_labelSize": 22,
"m_isLabelSizeProc": false,  // Label height 22 m and not % of screen
// width
"m_labelColors": {
// Label is in blue color

{
"m_red": 0,
"m_green": 107,
"m_blue": 179,
"m_alpha": 255
}
},
,
,...
// End symbology 3
},
,
"m_readFromURL": ",",
"m_writeToURL": ""
}
,
},
"m_epsg": ""  // End WalterMapsDoc
Source code for saving and loading the map document

From this link:

http://waltermaps.com/wp-content/tutorials/waltermaps_map_package_and_document_format/waltermaps_standalone_serialize.h

you can download a single C++ header file, implementing saving and loading of the Walter Maps map document. The code is only dependent on the small and efficient ESJ library (extremely simple JSON for C++):

https://github.com/g40/esj

The code can be used directly in a C++ project, or it can serve as pseudo code for your programming language of choice.
Creating your own web map definition files

Walter Maps supports viewing of online web map services in WMS, TMS, and WMTS format. In Walter Maps Pro you can also export these to offline raster files in GeoTIFF format. The connection to the specific web map server is made through a local service description XML file. The format of the WMS and WMTS xml files is described in the GDAL links

http://gdal.org/frmt_wms.html

and

http://gdal.org/frmt_wmts.html

In order for the service description xml file to work in Walter Maps, it must be saved using the file extension *.wmwmx (Walter Maps web map xml).

The WMS service description XML file can be generated manually, or created using the command line gdal_translate tool, which is a part of the open source GDAL software package.

To create a WMS service is a little tricky, but the following instruction will guide you in the right direction:

1. Install the GDAL software package
2. Find a WMS provider of your interest by browsing the web
3. Click on the WMS link given by your specific provider, e.g. http://geodpags.skogsstyrelsen.se/arcgis/services/Geodataportal/GeodataportalVisaSkoghistoria/MapServer/WmsServer which is a Swedish forestry history WMS.
4. After clicking on the link, you should see a WMS capabilities xml in the browser, alternatively it will be downloaded as a file to your Download folder.
5. If that isn’t working, add
   `?version=1.1.0&service=WMS&request=GetCapabilities`
to the link address

6. View the shown xml-file or open it. The capabilities xml
   often contains several layers, so search for the `<Layer>` tag
   and the `<Name>` tag until you find your layer of interest
   and copy the values for `<Name>`, and for `<BoundingBox>`
   (choose one of the bounding boxes). In this example the
   name found is “SkoghistoriaYta_Skogsstyrelsen”

7. Open the command line shell on your computer and
   navigate to your folder of choice.

8. Give the following command at the command line:

   ```
gdal_translate
"WMS:http://geodpgs.skogsstyrelsen.se/arcgis/services/Geodataportal/GeodataportalVisaSkoghistoria/MapServer/WmsServer?SERVICE=WMS&srs=EPSG:3006&LAYER=S=SkoghistoriaYta_Skogsstyrelsen&BBOX=200001.000000,610000.000000,1000000.000000,7700000.000000" forest_and_history.wmwmx -of WMS
```

9. As you can see, the **blue** coloured text is the same as the
   address at item 3 above and the **red** coloured text is based
   on the information in item 6 above.
10. Open the *forest_and_history.wmwmx* in an ordinary text editor. Its content will look similar to this:

```
<GDAL_WMS>
  <Service name="WMS">
    <Version>1.1.1</Version>
    <ServerUrl>http://geodags ... /WmsServer?SERVICE=WMS</ServerUrl>
    <Layers>SkoghistoriaYta_Skogsstyrelsen</Layers>
    <SRS>EPSG:3006</SRS>
    <ImageFormat>image/jpeg</ImageFormat>
    <Transparent>FALSE</Transparent>
    <BBoxOrder>xyXY</BBoxOrder>
  </Service>
  <DataWindow>
    <UpperLeftX>200000.000000</UpperLeftX>
    <UpperLeftY>7700000.000000</UpperLeftY>
    <LowerRightX>1000000.000000</LowerRightX>
    <LowerRightY>6100000.000000</LowerRightY>
    <SizeX>536870912</SizeX>
    <SizeY>1073741824</SizeY>
  </DataWindow>
  <BandsCount>3</BandsCount>
  <BlockSizeX>1024</BlockSizeX>
  <BlockSizeY>1024</BlockSizeY>
  <OverviewCount>20</OverviewCount>
</GDAL_WMS>
```

11. Finally, add the following lines prior to the `</GDAL_WMS>` end tag

```
<Cache>
  <Path>gdalwmscache</Path>
  <Depth>1</Depth>
  <Extension>.png</Extension>
</Cache>
<OfflineMode>false</OfflineMode>
<UserAgent>Mozilla/5.0</UserAgent>
<MaxConnections>20</MaxConnections>
<Timeout>10</Timeout>
<UnsafeSSL>true</UnsafeSSL>
<ZeroBlockHttpCodes>204,404</ZeroBlockHttpCodes>
<ZeroBlockOnServerException>true</ZeroBlockOnServerException>
```

Additional *wmwmx* files can be found and downloaded at the Walter Maps community forum:

http://waltermaps.com/forums/forum/free-geodata
Translating Walter Maps to your own language

You can “easily” translate Walter Maps to your own language (approximately 700 text strings have to be translated). To get started, you need the English language file `english.lang`. It can be downloaded from:

http://waltermaps.com/wp-content/tutorials/waltermaps_lang/english.lang

For editing multi-lingual text files you need a text editor that will support it (i.e. supporting saving files in UTF-8 format). You can use BabelPad and it can be downloaded from:

http://www.babelstone.co.uk/Software/BabelPad.html

Open the downloaded file `english.lang`. The first lines will look similar to:

```
English
ENERGY_SAVE_MODE Energy save mode
USE_WEB_MAP_CACHE Use web map cache
NOT_VALID_COLORS Not valid colours, e.g. 123 234 or 23,12,34 23,45,233
```

The first line in the file should be replaced by the name of your language, e.g. Español. Then you start translating the blue text in the above example (and all the other ~700 strings). ! **Please note**, do not change the first upper-case key string on each line. When you have finished translating, save the file using a relevant file name e.g. `espanol.lang`.

When new versions of Walter Maps are distributed on the app stores, the default language file might be extended. Walter Maps will still accept your language file, even though a small amount of the user interface will be in English (until you have added new string translations).
# Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature</td>
<td>A representation of a real-world object in a geographic information system, often represented by one geometry and a set of attribute field values</td>
</tr>
<tr>
<td>Feature class</td>
<td>Collection of features, where all geometries have the same spatial representation, i.e. the same geometry type and the same spatial reference</td>
</tr>
<tr>
<td>Geometry</td>
<td>A point, a line, or a polygon</td>
</tr>
<tr>
<td>GIS</td>
<td>Geographic information system, i.e. a system for collecting, storing, visualizing, and analysing geographic data</td>
</tr>
<tr>
<td>Raster data</td>
<td>Geographic data represented in a uniform and quadratic grid</td>
</tr>
<tr>
<td>Symbology</td>
<td>Definition of how to represent features on a map</td>
</tr>
<tr>
<td>UUID</td>
<td>Universally Unique Identifier is a unique text string, used to uniquely identify information in computer systems</td>
</tr>
<tr>
<td>Vector data</td>
<td>Geographic data represented by points, lines, or polygons</td>
</tr>
</tbody>
</table>