**Underhill Geomatics Ltd.**

# Copan *Lite* for Windows

# User Manual



updated: 25-Mar-2011

***Copan for Windows***

# Part I: Introduction



updated: 21-Sep-2010

**Copan *Lite* for Windows**

# Contents



updated: 30-May-2011

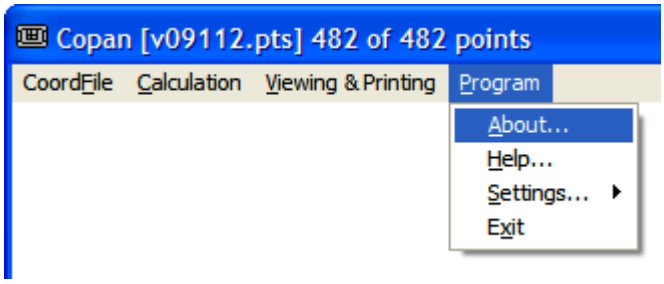
updated: 22-Dec-2011

**Copan *Lite***

# Software Version

This user manual is for **Copan *Lite***, version 11.11, which is freely available to the public.

### To determine your software version and release date




- Choose the **About...** item from the **Program** menu.


Please note the software version and release date in all communications.

### Other editions of Copan

There are other versions of Copan—freely available to the public:

- **Copan for Palm OS**.
- **Copan for Mobile and Pocket PC**.

And there are now also professional versions of Copan for Windows available to the public, for trial or purchase:

- **Copan *Pro***—Copan *Lite* plus many drawing functions and Leica digital level tools.
- **Copan *Pro+DWG***—Copan *Pro* plus the ability to export to AutoCAD DWG files.

updated: 22-Dec-2011

updated: 21-Sep-2010

***Copan for Windows***

# Getting Started

Contents



Copan is a powerful tool for land surveyors and civil engineers who do field and office calculations with plane coordinates and survey measurements. To use Copan, you must be familiar with basic Windows operations and with basic geomatics (or surveying engineering) concepts.

## 1. Some Conventions

### This Manual

- **Boldface** words are usually menu items or dialog buttons or field names.
- `Monospace` words usually mean folder or file names or data values.
- Where it is not spelled out, please read **A | B** as "Select item B from the pull-down menu A". For example, to do basic COGO, after you've opened a coordinates file, the instruction **Calculation | COGO...** means "Choose the **Calculation** menu then the **COGO...** item".
- The § symbol before capitalized text (e.g., § Map Traverses) means the text is the name of another web page or printed chapter.
- If you're reading this on-line, you can go to the table of contents, the previous page, or the next page via [contents] [previous] [next], etc, "buttons" at the top or bottom of each page. (If you have the Opera web browser, you can choose the same items on the its navigation bar.)
- To print any of this manual's web pages or chapters, you have two options:
  - Go to the desired web page and print it! It is already printer-friendly. (Headings can, however, get orphaned — except in Opera.)
  - Open the PDF version of the manual, User Manual.pdf (you will need an Adobe reader), and print the whole thing or just your desired pages. You can jump to specific chapters via the Bookmarks tab.

### The Software

- The page of this user guide that is relevant to a particular module is usually accessible via that module's **Help** button or menu item.

- Many menu items or dialog buttons are often grayed out — disabled — whenever their associated function or action is not appropriate at the time. For example, until a coordinate file with points is open, most CoordFile menu items and most Calculation menu items are disabled.
- Frequently, *shortcuts* can be made by pressing the Alt key in combination with a single letter abbreviation indicated by an underline. For example, in the COGO dialog, press Alt-T to jump to the **To point** field or Alt-E to **Erase lines**.
- Many dialogs (or windows) have a **Factory** button, which resets most values in the dialog to the state they were in when the program was first installed. Also, many dialogs have a **Reset** button, which resets most values to the state they were in when the dialog was just opened.
- Any menu item or dialog button that ends in ellipses (...), e.g., **Print...**, leads to another dialog or window rather than having an immediate effect.
- When viewing a graphic display of a file, you can *pan and zoom* as follows:
    - If you have a *mouse wheel*, spin it to zoom in or out; double-press it to zoom to the full extent of the file; and hold it down and move it to pan around.
    - Alternatively, use the **Zoom In**, **Zoom Out** and **Zoom to Extents** items under the **Viewing** menu for their associated tasks (Zoom In requires you to drag a rectangle over your intended target), and use the scroll bars to pan around.
- During calculations, all *distances* are assumed to be in the same units as those of the coordinates. Input distances, however, can usually be converted via a given units factor.
- When entering *distances* or *coordinates*, enter values explicitly, with a dot (.) as the decimal separator and without any thousand separators or any units. For example, enter 12,345.678m (i.e., 12km + 345m + 678mm) as `12345.678` or enter 123 456.78ft as `123456.78`. In many cases enter an *expression* to imply, for example, the distance betweeen two known points. (See § Distance and Offset Expressions.)
- If you measure angles in *degrees-minutes-seconds* (i.e., 1 circle = 360°), when entering angles, always use the format *`dd.mmss`*. For example, you must enter 24°52'10" as `24.5210` and you must enter 30' as `0.30` (or just `.3`).
- If you measure angles in *gons* or *grads* (i.e., 1 circle = 400$^{g}$), just enter angles *as is* (e.g., enter 24.52105$^{g}$ as `24.52105`).
- When entering *bearings* (or *azimuths*), enter angles explicitly, as above, or (in many cases) enter an *expression* to imply, for example, the direction betweeen two known points. Also, enter bearings in *whole-circle* or *quadrant* form at any time. (See § Bearing Expressions.)

## 2. Installing or Uninstalling Copan

If the software is not there already, see § Installing or Updating on how to install or update Copan on your computer.

Had enough already? See § Uninstalling on how to remove Copan.

## 3. Program Settings

There are various general settings you can make which control program behavior, such as the name and address of your compay, the the precision of listed or displayed distances and coordinates, or the tolerances for acceptable traverse closures. See § Program Settings on the various ways to customize Copan for your preference.

The most important program setting to make after installation may be to the *Angle Units* and *Bearings* (or azimuth) type. The default **Angle Units** setting is **Degrees Minutes Seconds**, but you may choose **Gons (Grads)** instead. By default, **Whole-circle bearings** are **North-based**, but you may choose **South-based** instead. Also you choose to **List/display bearings** in **Whole -circle form**, by default, or in **Quadrant form** instead.

## 4. CoordFile Functions

The Copan package includes a number of different files — some are essential and others are supplemental to using Copan. See § File Locations on the various files that make up the Copan package and how to change their locations.

Virtually everything in Copan involves a coordinate file, or *coordfile* for short. Even if you'd like only to process coordinates from another program, you must first create a new coordfile and then import them. Until a coordfile with points is open, most CoordFile menu items are grayed out.

A coordfile contains a single *head* record and a set of survey *point* records. See § Coordinate Files on how to do basic coordinate file management and § Point Records on the nature and management of point records.

See § Point Lists on listing points, § Bulk Point Editing on mass editing point attributes, § Import and Export on importing and exporting points from/to text files, and § Export to Total Station File on exporting points to survey instrument files.

See § Viewing and Printing Graphics on how to graphically navigate a displayed coordinate file, how to specify paper units, and how to print graphics to scale.

## 5. Survey Calculations

Copan provides many geomatic data processing functions:

See § COGO Calculations on how to do basic coordinate geometry — direct and inverse calculations, various kinds of intersection calculations, and corner angle calculations — using identified points, bearings, and horizontal distances and offsets.

See § Multiple Inverse Calculations on how to calculate bearings and horizontal distances, or distances and offsets, to many points from a single point.

For processing *raw* field survey data,

- See § Field Data Processing on calculating and adjusting traverses and side-shots, based on sequences of horizontal and vertical angles and slope distances.

- See § Field Bearings Processing on calculating and adjusting traverses and side-shots, based on sequences of bearings, vertical angles and slope distances.
- See § Freestation Processing on calculating resections and side-shots, based on rounds of horizontal and vertical angles and slope distances.

For processing *reduced* survey data, even those describing curved boundaries or alignments,

- See § Map Traverses on calculating and adjusting traverses, areas and perimeters, based on sequences of bearings and horizontal distances.
- See § Map Checks on calculating traverses, areas and perimeters, based on sequences of bearings and horizontal distances — without coordinate points.
- See § Area and Perimeter Calculations on calculating traverses, areas and perimeters, based only on sequences of existing points.

See § Coordinate Transformations on how to mathematically transform many points based on known rotation, scale and shift parameters, or on how to calculate the best-fit transformation between one set of points and another.

Remember that for most Calculation menu items to be enabled, a coordfile with points must be open. An exception is Map Check, which is only enabled when *no* coordfile is open.

## 6. Bugs, Crashes and Improvements

Copan is a work in progress and you may encounter some problems. Please be patient. See § Bugs and Other Problems regarding bugs, crashes, and error messages.

We are continuously trying to improve Copan: increasing its functionality, making it easier to use, or making it more stable.

Let us know what improvements you suggest. support@underhill.ca. Please note the software version and release date (see § Software Version) in all communications.

updated: 21-Sep-2010

updated: 2-Feb-2011

***Copan for Windows***

# Frequently Asked Questions



## 1. Why are the menu items disabled?

When you install Copan *for the first time*, most of the functionality is disabled unless

- you have a binary key file, CopanKey.bin, which (since version 09.06) is automatically stored in your Copan program folder by the installer,
- you correctly entered the password, which was sent by e-mail, and
- the key file and password are matching, i.e., they are for the same release/version.

See § Installing or Updating for more details.

The *usual* reason for any item or button being disabled is that, under the current conditions, that item or button is not appropriate. Thus, if you do not have a coordfile open, many menu items (e.g., List Points) are not applicable and not enabled. Or, if you have a coordfile open but with less than an appropriate number of points in it, most Calculation menu items (e.g., COGO) are not applicable and not enabled.

## 2. When will Copan have drawing capabilities?

On our in-house version of Copan, we currently have the ability to export symbolized drawings to AutoCAD DWG files. Also, we are working on a Copan drawing module that will have many useful CAD functions, independent of AutoCAD. When a CAD version of Copan is released, in 2011, we will be charging the public for it. Meantime, maybe you'd like to be a CopanCAD beta tester? However, the free version of Copan will remain free and will continue to evolve.

## 3. How can I do curve calculations?

The *hand-held* versions of Copan — for Palm OS and for Mobile or Pocket PC — have curve calculators where you can enter any suitable combinations of circular curve parameters and calculate dependent parameters, but *Copan for Windows* does not.

You *can*, however, use the Map Checks or Map Traverses modules to calculate curve parameters *if* you have the azimuth (or bearing) and distance from the beginning of curve in to

the centre of curve, and the direction out from the centre to the end of curve. The arc distance, chord distance, and arc angle will be calculated. See § Traversing Curves on how to define curves along a map traverse.

### 4. How can I delete, renumber, or edit a group of points at once?

You *can* do simple mass editing of points via **CoordFile | Bulk Edit...** (See § Bulk Point Editing.)

For sophisticated editing, the best way is to export the coordfile from Copan to an Ascii file, edit the Ascii file in an editor or spreadsheet, then, back in Copan, create a new coordfile and import the edited Ascii file. (See § Import and Export.)

Geometric transformations of points *en masse* can be done via **CoordFile | Transform Coords...** (See § Coordinate Transformations.)

### 5. Why won't my space-delimited text file import?

If you try to import a space-delimited file and Copan crashes or complains, it probably isn't really a space-delimited file! If there are a number of spaces separating each non-empty field, then the file is not space-delimited. Whatever the delimiter character is, there can be only one delimiter separating each non-empty field.

If the columns of data are aligned such that it is possible to draw a straight vertical line between each column, then it is a fixed-format file. To import such a file, choose the Fixed Format option and enter the Start and Width values for each field. (The imaginary straight vertical lines indicate the start positions.)

An alternative solution may be to use a spreadsheet, which is much more sophisticated at interpreting fixed-format files. If the file successfully imports into a spreadsheet, save it in comma- or tab-delimited text format, then import it back into Copan.

### 6. Why am I not being asked to accept the “Conditions of Use” or to “Enter Password” when I install Copan?

The usual reason is that you do not need to! If this is *not* the first time you have installed Copan on the same computer (i.e., you have successfully installed and used Copan before) then you are only *updating* Copan, and you should not be asked again to accept conditions or to enter a password.

If the menu items are disabled see item 1 above.

### 7. How can I assist UGL in developing or promoting Copan?

Thank you for considering it! Here are some things you could do:

- Give us a short testimonial on our Copan web page and, if you like, get a link back to your site.
- Submit a short review to a software downloads site where Copan is listed.
- Post a favorable comment to a geomatics or engineering resources web site where Copan or UGL is listed.

- Write a review of Copan for your local or national engineering or geomatics magazine or newsletter. You know which one is right for you. True, it'll take some time, but your name and company will also get some recognition. Check with the magazine or newsletter first to see if it's appropriate.
- Help us develop Copan tutorial materials. Do you have teaching tendencies? Send us some sample problems, tutorial notes, or data files relating to the use of Copan.
- Help support Copan usage in another language. Maybe you can translate the User Manual or offer to moderate a Copan forum?

updated: 2-Feb-2011

 ***Copan for Windows***

# Part II: CoordFile Functions

- Coordinate Files
- Point Records
- Point Lists
- Bulk Point Editing
- Import and Export
- Export to Total Station File

updated: 21-Sep-2010

updated: 21-Sep-2010

*Copan for Windows*

# Coordinate Files

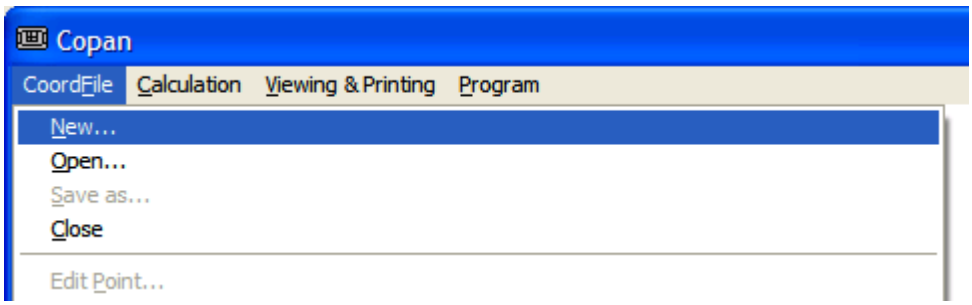
Contents



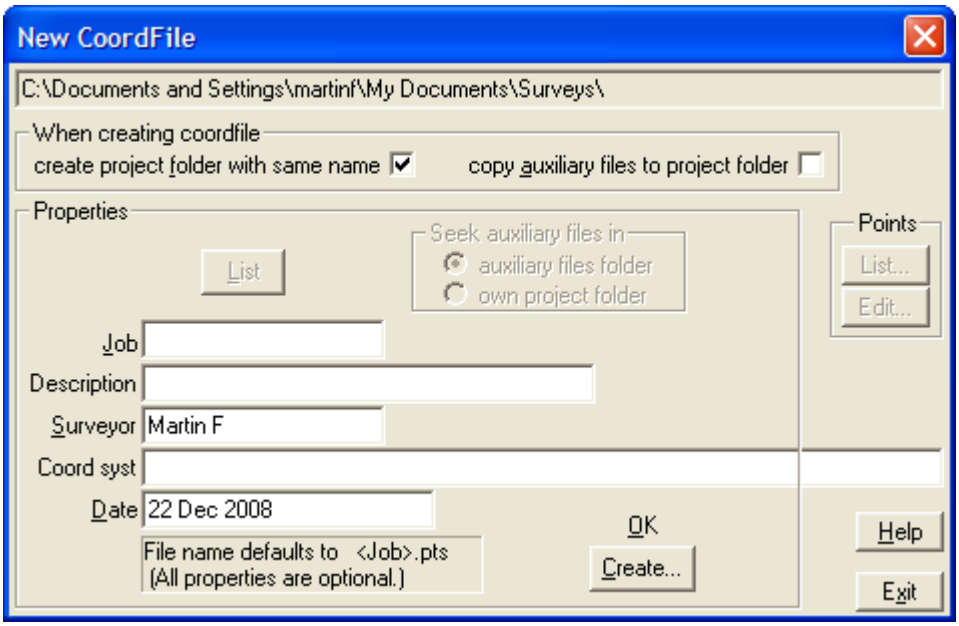
Virtually everything in Copan involves a coordinate file, or *coordfile* for short. Even if you'd like to process only coordinates from another program, you must first create a new coordfile and then import them.

A coordfile is a standalone file containing a *head* record and a set of survey *point* records (which are described in § Point Records). It is a binary file and has a **.pts** name extension.

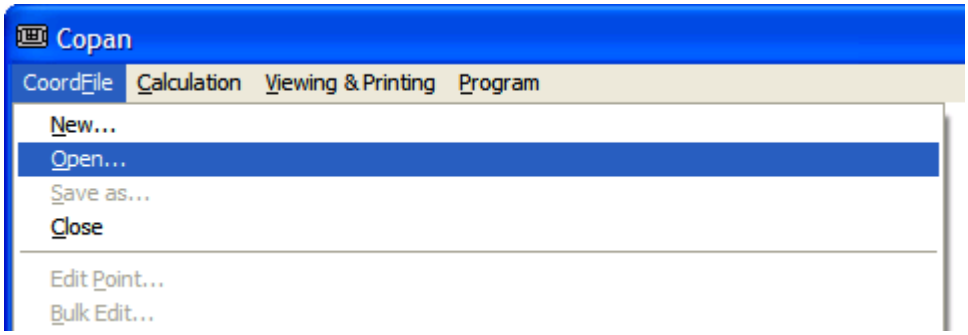
## 1. To Create a New Coordinate File




1. **CoordFile | New...**
2. Enter a **Job** name or ID and other details you'd like kept in the head record. You can always come back and edit the details later.
3. Optionally, check **Create project folder** to have Copan automatically create a project folder in the Initial Project Folder. Then check **Copy auxiliary files to project folder** to have Copan do that if you wish to have auxiliary files specific to the new project. (The default location for the new coordfile is the *current* Project Folder.)
4. **Create...** or **OK**.
5. Optionally, navigate to the another folder or enter an alternative file name. By default, the filename is the same as the Job #, plus the .pts extension.
6. **Save** the coordfile.

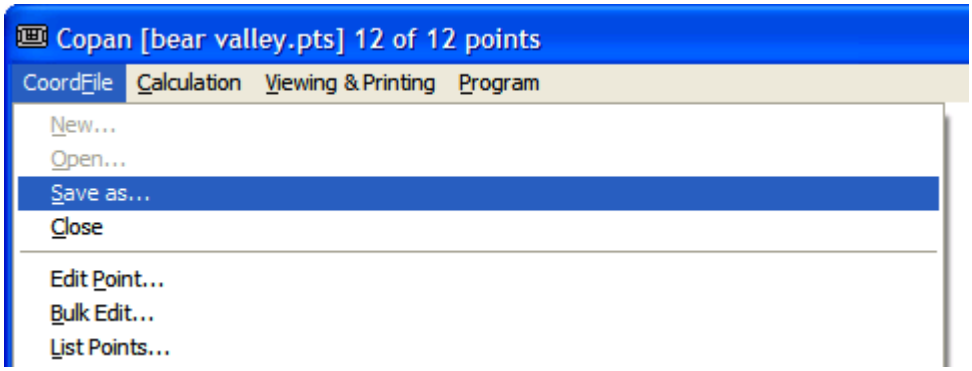



## 2. To Open an Existing Coordinate File



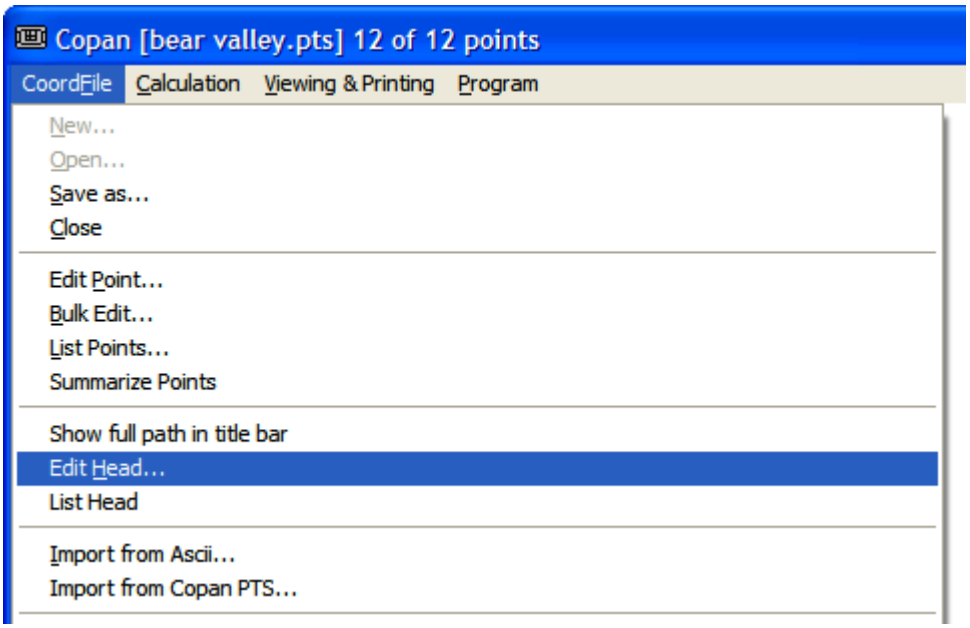

1. **CoordFile | Open...**
2. Navigate to the coordfile.
3. **Open** the coordfile.
4. If someone is already using that file, or if Copan has prematurely ended while that file was in use, you will be told that editing is disabled.

### 3. To Save an Existing Coordinate File under a New Name

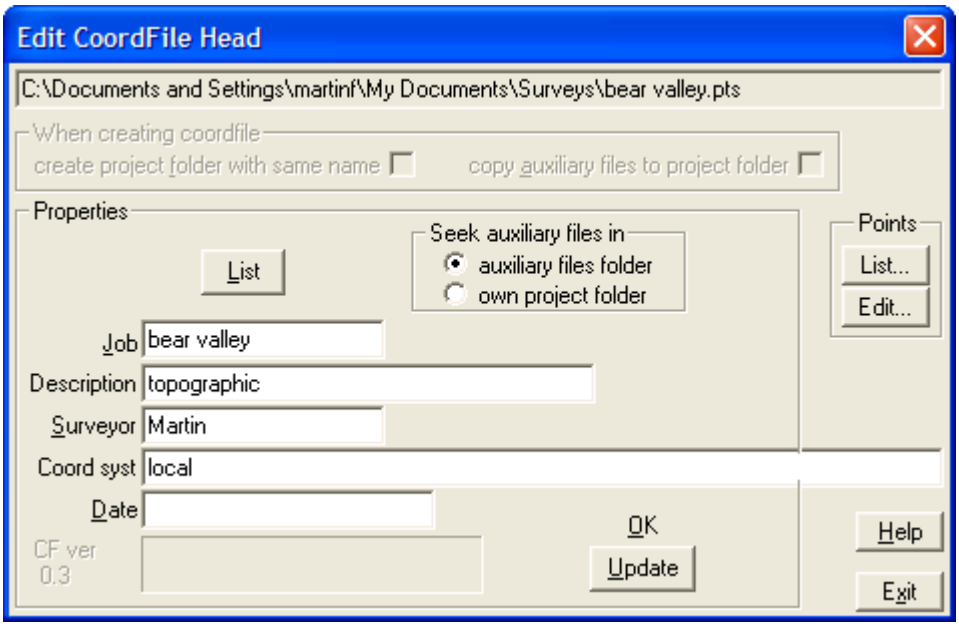



1. **CoordFile | Save as...**
2. Enter the new name, or navigate to one you wish to overwrite.
3. **Save** the coordfile. (This won't change its Job #.)

### 4. To Edit the Head Record




1. **CoordFile | Edit Head...**
2. Enter or edit the details. You may change the Job # but this won't change the name of the coordfile. Most coordfile head fields are not processed by Copan and can be omitted. The exceptions being Job # and Surveyor, which are used during manual entry of field data, and the **Seek auxiliary files in** option which tells Copan where to find the auxiliary files for this coordfile:
   - in the program's main **auxiliary file folder**, or
   - in the coordfile's **own project folder**.
3. **Update** the head.




## 5. Notes

- You may change the name of a coordfile using any normal Windows method, but this won't change its Job #.
- When working on the points in a coordinates file, you do *not* need to remember to save the file periodically — in fact there is no CoordFile | Save menu item — because whenever you add or update points, Copan updates the coordfile instantaneously, wherever it is stored.

updated: 21-Sep-2010

updated: 21-Sep-2010

***Copan for Windows***

# Point Records

Contents



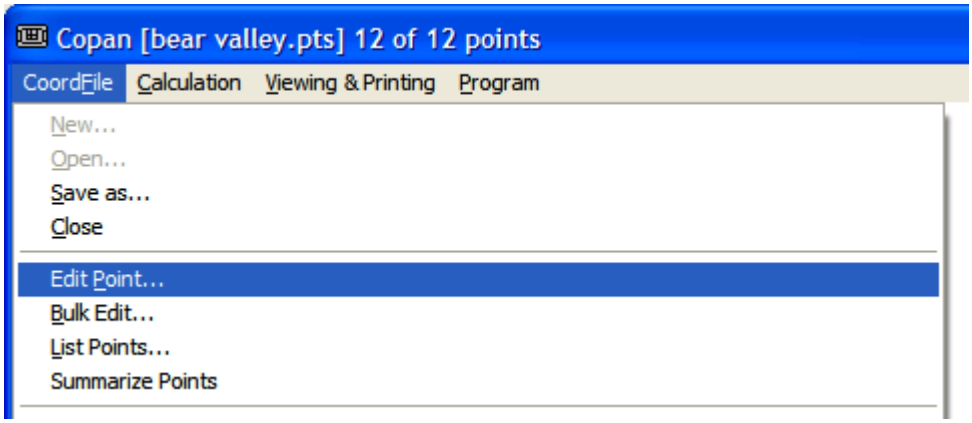
A coordfile contains a *head* record and a set of survey *point* records. A point record contains a group of fields: plane survey coordinates and a number of attributes.

## 1. Point Record Fields

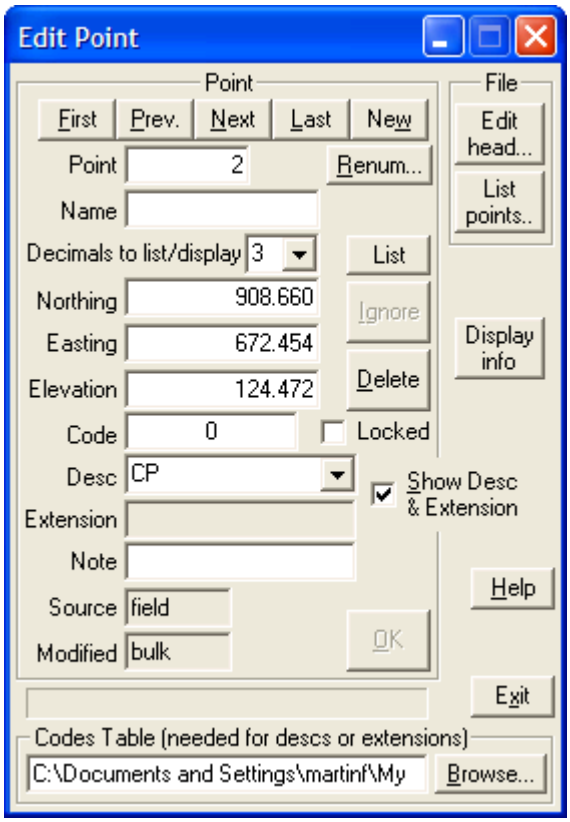
Some coordfile point record fields are numeric and others are alphanumeric (or textual). Some are editable and others read-only. Some are required and others optional.

- A point **Number**, required, must be any positive integer, up to nine digits, and be unique within a coordfile.
- A point may be **Locked** in which case it is protected from being deleted or edited by another module.
- An optional point **Name** can be any string of up to 15 printable characters, to help identify that specific point, such as how it is known by an outside organization.
- Point **Northings** and **Eastings**, required, can be any real or integer numbers.
- An optional point **Elevation** can be any real or integer number.
- An optional point **Code** (or *feature* code) can be any string of up to 13 printable (often numeric) characters, but please see § Point Codes concerning its intended use.
- A read-only code **Desc**, intended as a brief (up to 13 characters) explanation of the code.
- A read-only code **Description**, which is just a longer (up to 31 characters) version of the code Desc.
- A read-only code **Extension**, intended as an explanation (up to 17 characters) of the extended portion of the code, if it exists.
- An optional point **Note** can be any string of up to 17 printable characters to help describe that specific point, such as a statement about a feature's condition or construction or a reminder to check the field book.
- The point **Source** and **Modified** are read-only fields and indicate which module or program first created the point and which one last modified it. (See below for details on these 6-character code names.)

## 2. To Add, Delete, or Edit a Point Record

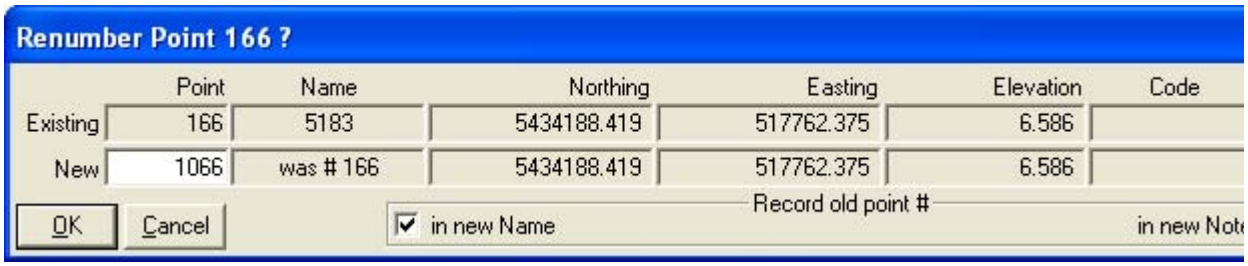



1. **CoordFile | Edit Point...**
2. Enter the **Point** number — which must be a positive integer — or navigate to it via **Prev.**, **Next**, etc.
3. Enter or edit the point's details.
    - The presence or absence of an Elevation (even a value of zero) implies whether the point is 3D or 2D.
    - Point Names and Notes are not processed by Copan, other than being input, filtered, or output.
    - To use the Code-dependent Desc or Extension fields, check the **Show Desc & Extension** box and ensure that a valid Codes Table path (see § Point Codes) is provided in the so-named box.
    - There are two alternatives to entering a numeric code, when the **Show Desc & Extension** box is checked:
        - pick its Desc from the drop-down Desc list
        - enter its Desc in the Code box
4. **Add**, **Delete**, or **Update** the point as appropriate.




### 3. To Renumber a Point Record

1. **CoordFile | Edit Point...**
2. Enter the **Point** or navigate to it via **Prev.**, **Next**, etc.
3. **Renum...** the point as appropriate, optionally recording the old number in the Name or Note.




### 4. Src/Mod Code Names

Various Copan modules and other programs are recorded, via a code name, as being the source or last modifier of a point:

| Src/Mod | Module or Program |
|---|---|
| `bulk` | Bulk Edit |
| `cogo` | COGO |

| | |
|---|---|
| `edit` | Edit Point |
| `field` | Field Data |
| `form` | Transformation |
| `import` | Import Ascii |
| `loop` | Map Traverse |
| `palm` | Palm Copan |
| `pocket` | PocketPC Copan |
| `hp42` | HP42 emulator |

## 5. Note

- When working on the points in a coordinates file, you do *not* need to remember to save the file periodically — in fact there is no CoordFile | Save menu item — because whenever you add or update points, Copan updates the coordfile instantaneously, wherever it is stored.

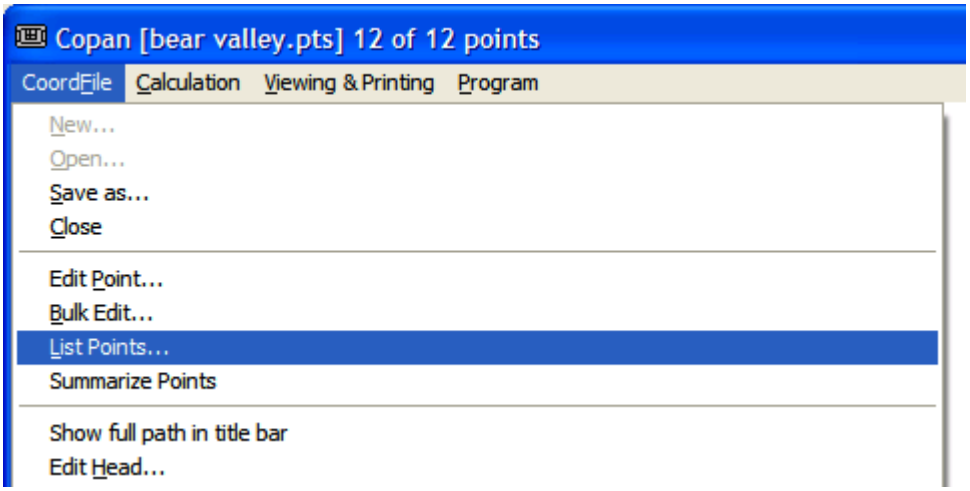
updated: 21-Sep-2010

updated: 21-Sep-2010

*Copan for Windows*

# Point Lists

You can list details of a set of points directly from the menu (via **CoordFile | List Points**) or from various modules. The Points List dialog allows some control over the nature the list that will appear in the § Info Display file.




## Selection

Which points get output? The options are

- All — every point.
- Filtered — a subset of points. (See § Point Filters.)

## Order

What's the output sequence? The options are

- Numerical — by point number.
- Chronological — by when the point was created.

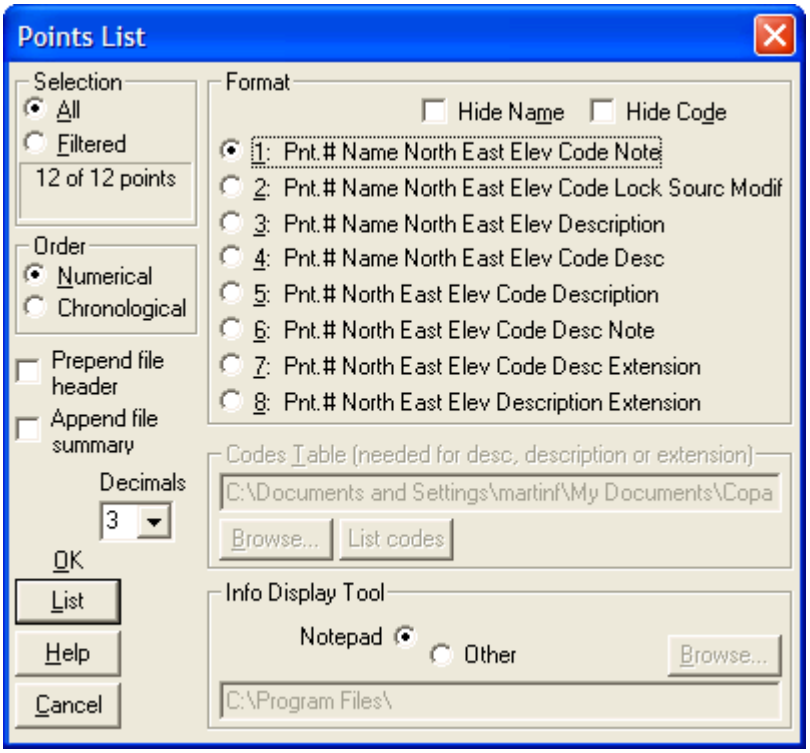
## Format

Which fields get output? (See Point Record Fields in § Point Records.) The eight options are

1. Number, Name, Northing, Easting, Elevation, Code, and Note.
2. Number, Name, Northing, Easting, Elevation, Code, Lock, Source, and Modified.
3. Number, Name, Northing, Easting, Elevation, and Description.
4. Number, Name, Northing, Easting, Elevation, Code, and Desc.

5. Number, Northing, Easting, Elevation, Code, and Description.
6. Number, Northing, Easting, Elevation, Code, Desc, and Note.
7. Number, Northing, Easting, Elevation, Code, Desc, and Extension.
8. Number, Northing, Easting, Elevation, Description, and Extension.

All but the last format option can be further modified by checking the **Hide Name** or **Hide Code** boxes to omit the name or code fields.




## Notes

- If you need a listing in a format not found here, try exporting to a text file instead. (See § Import and Export.)
- To use Format options 3 – 8, be sure to select a valid **Codes Table** which should contain a Description, Type and Desc for each integer code. (See § Point Codes.)
- If the Codes Table has no Descs, truncated Descriptions will be used instead.
- In option 2,
  - Lock indicates if the point is locked (via an "L" on output),
  - Source identifies the module or program which created the point, and
  - Modified identifies the module which last changed the point

  (see Src/Mod Codes in § Point Records).
- **Decimals** controls the precision of listed coordinates. Note that this setting is *global* (i.e., it affects various other modules that display or list coordinates and distances.)

- **Prepend file header** begins the listing with the coordfile head data (which can be edited via **CoordFile | Edit Head...**).
- **Append file summary** ends the listing with the coordfile summary:
  - Point count
  - Point numbers used
  - Minimum and maximum northings, eastings and elevations
  - Point numbers not yet used

updated: 21-Sep-2010

updated: 21-Sep-2010

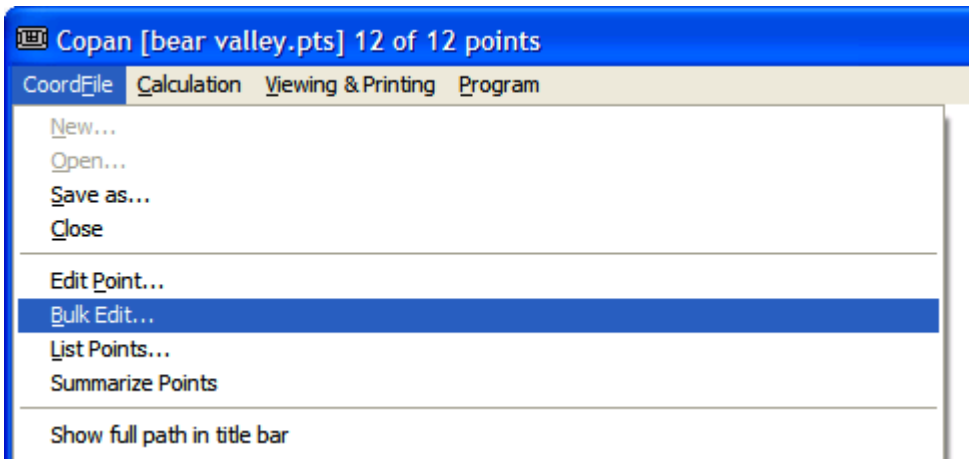
*Copan for Windows*

# Bulk Point Editing

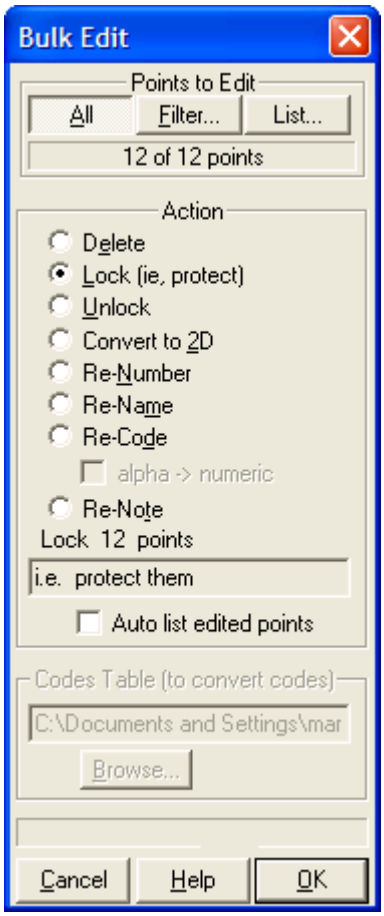
With this module you can delete, renumber, or edit the non-coordinate fields of a large group of coordfile points in one go.

Note that if you need to edit the *coordinates* of a group of points in a systematic manner, use the Coordinate Transformations module instead.

### To Edit Many Points at Once




1. **CoordFile | Bulk Edit...**
2. Choose the subset of **Points** to edit — All or a Filtered set (See § Point Filters).
3. Choose the editing **Action** to take:
    - **Delete** to remove points.
    - **Lock** or **Unlock** to protect or unprotect points. (When a point is locked, nothing about it can be changed until it is unlocked.)
    - **Convert to 2D** to remove elevation values.
    - **Re-Number** to change point numbers. (You'll be allowed to set a renumbering rule.)
    - **Re-Name**, **Re-Code** or **Re-Note** to change points' alphanumeric fields. Re-Code has a sub-option to replace alphabetic codes with numeric equivalents, as defined in the Codes Table (see § Point Codes).
4. If you are editing an alphanumeric field, enter an *update expression* (see below) in the box, to indicate what the new value for the field should be.
5. **OK**.




## Update Expression

Currently, the opportunities for specifying how a Name, Code, or Note field should be updated are very limited. Whatever you enter in the box is what the new value for the field will be.

For example, choosing Re-Note and leaving the box blank is a request to erase the Notes for all currently filtered points. And choosing Re-Code and typing “24” in the box is a request to set the Codes for all currently filtered points to “24”.

updated: 21-Sep-2010

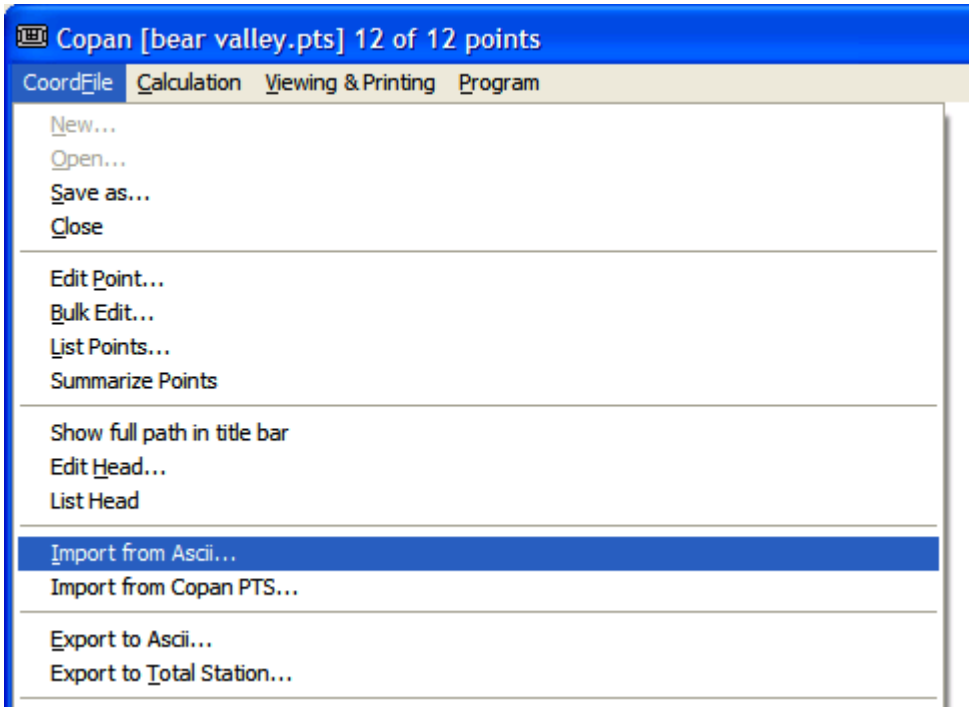
updated: 24-Mar-2011

***Copan for Windows***

# Import and Export

Contents






## 1. Importing and Exporting Text Files

Tabulated plain text files can be used both as a source of data for, or as a destination from, an open coordfile. The files can be in delimited (CSV) or fixed format.

### Text File Format

To be compatible with Copan, such a text (or ASCII) file must have the following:

- Any name, though names with **.txt**, **.asc** or **.csv** extensions are advisable.
- Single-byte character encoding, such as **ANSI**, **DOS** or **UTF-8**. *Note*: multi-byte character encodings (such as Unicode or UTF-16) will *not* work.
- An optional heading line, at the top.
- One record per line, each record representing a point.

- Various fields per record, some of which must correspond in type, but not necessarily in order, to Copan point record fields. (See Point Record Fields in § Point Records.)
- Fields formatted in one of two ways:
  - *Delimited*, with fields separated by a *single* character (e.g., comma, space, or tab); or
  - *Fixed*, with fields aligned in specific column positions.
- The same field count, order, and format for each line.

The rules for Delimited Format files are similar to those for Excel's CSV files:

- Fields *must* be separated by the delimeter character (e.g., a comma-delimited *P-N-E-Z* line might be `3,1234.56,3456.78,90.12`).
- Unless the space character is the delimeter character, spaces may accompany the separators for human readability (e.g., `5, 1234.5, 3456.7, 90.1` would be interpreted as `5,1234.5,3456.7,90.1`).
- An imported field *may* be surrounded by double-quotes (e.g., `"7",123.4,"345.6",9.0` would be interpreted as `7,123.4,345.6,9.0`).
- Exported alphanumeric fields may be surrounded by double-quotes (see next two items) but numeric fields — point number and coordinates — are not.
- A field may *contain* delimiter characters as part of it's value, if so, the field must be quoted (e.g., the Note part of a line might be `"OIP, bent"` and would be interpreted as the single string `OIP, bent`).
- A field may contain double-quote characters as part of it's value, if so, the field must be quoted and each internal double-quote must be preceded by a double-quote (e.g., a Note might be `"""gold"" tag"` and would be interpreted simply as `"gold" tag`).

**Example 1: Comma-delimited text file, three columns, no headings.**

```
1,5699.4,695.2
2,5700.6,695.7
3,5700.0,696.3
4,5699.3,696.1
```

**Example 2: Fixed-field text file, four columns, with headings.**

```
 Number  Northing  Easting  Code
      1    5699.4    695.2     0
      2    5700.6    695.7     8
      3    5700.0    696.3     7
      4    5699.3    696.1
```

**Example 3: Comma-delimited text file, six columns, with headings.**

This is a hypothetical GPS data file, whose second and third fields do *not* correspond to Copan fields.

```
Point,UTC,Fix,Northing,Easting,Elev
1,155757,3,6306918.388,380793.430,3.07
2,155758,3,6306918.360,380793.445,3.07
```

```
3,155759,4,6306918.358,380793.449,3.09
4,155800,4,6306918.355,380793.444,3.08
```

**Example 4: Bar-delimited text file, with strange notes.**

This file

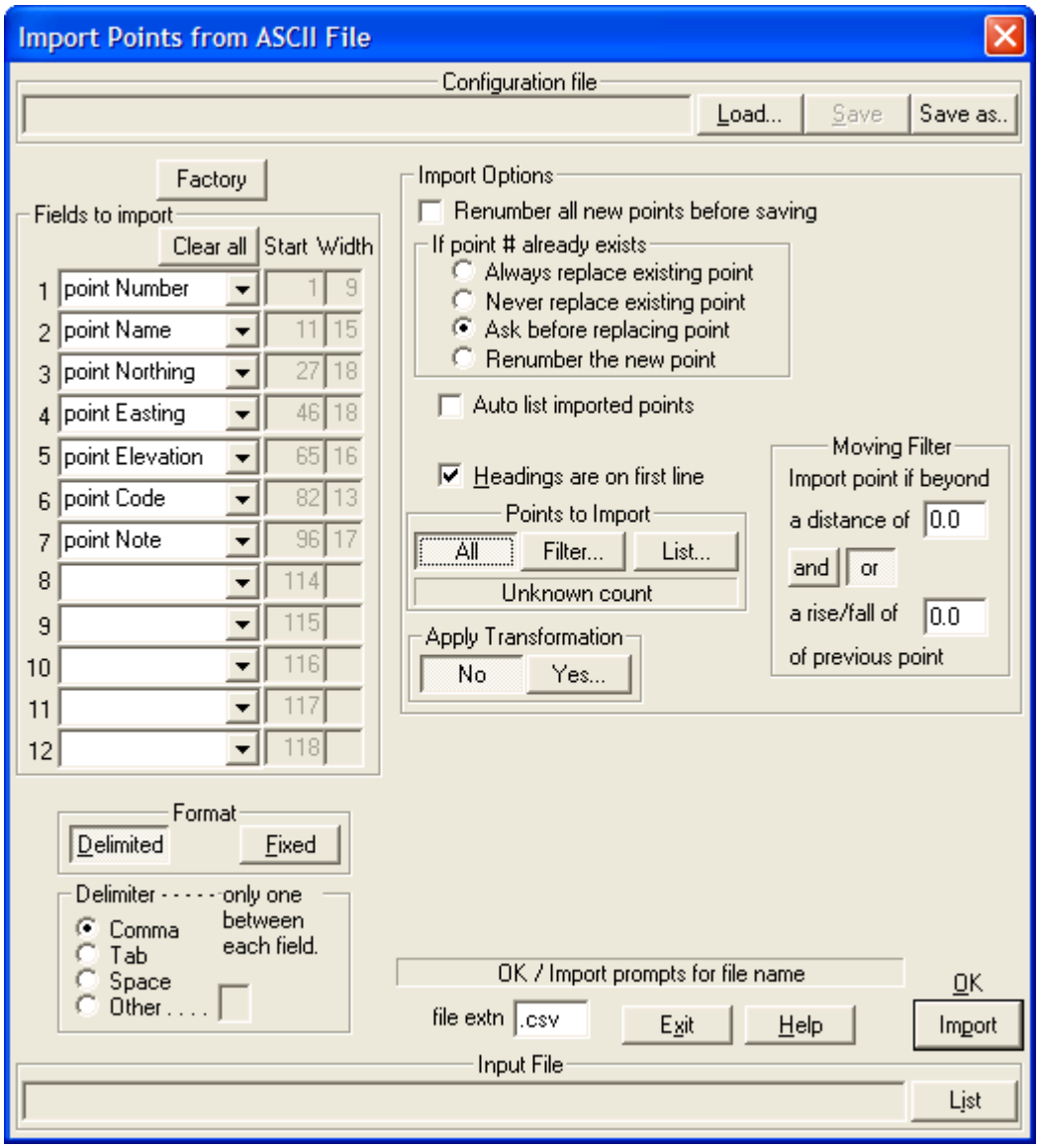
```
Point|Northing|Easting|Note
1|918.3|179.0|wall
2|618.6|180.5|"non |_ corn"
3|594.3|179.9|wall
4|901.8|193.4|"6"" pipe"
```

would be interpreted as

```
Number  Northing  Easting   Note
     1     918.3    179.0   wall
     2     618.6    180.5   non |_ corn
     3     594.3    179.9   wall
     4     901.8    193.4   6" pipe
```

## To Import a Text File

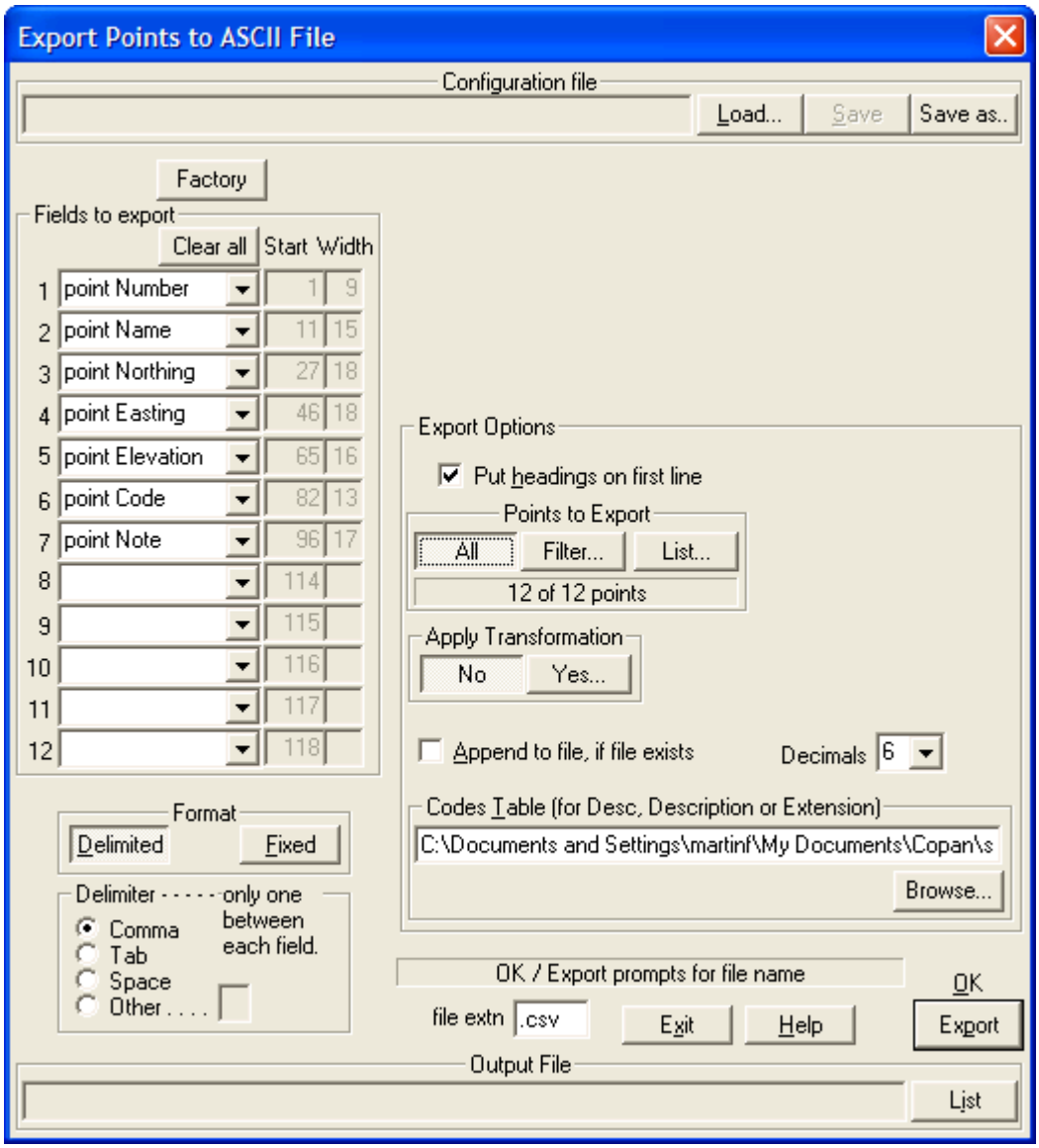
1. **CoordFile | Import from Ascii...**
2. Optionally **Load...** a **Configuration file**.
3. Indicate which Ascii file **Fields to import** to what point record fields. (See Point Record Fields in § Point Records.) The chosen point record fields must include at least **Number** and **Elevation**, or **Northing** and **Easting**. If an input file has fields that are not desired, choose the blank field for them. (So to import example file 3, above, choose Number, *blank*, *blank*, Northing, Easting, and Elevation.)
4. Choose either **Delimited** or **Fixed** format. (See Text File Format above.)
5. Choose which **Points** to import: **All** or **Filter..** a subset. (See § Point Filters.) After a potential import record has been converted to Copan point format according to your formatting criteria, it is tested against the import filter before actually being saved to the coordfile.
6. Optionally **Apply Transformation** to the points being imported. (The source text file will not be changed.) Clicking **Yes...** allows you to enter the transformation parameters (see § Coordinate Transformations.)
7. Choose other options as appropriate, such as whether **Headings are on the first line**, whether to **Renumber new points** or **replace existing points** (see § Point Renumbering or Replacement), or whether to use a **Moving Filter** (see below).
8. Optionally **Save** the **Configuration** for later reuse.
9. Optionally **List** this file for preview.
10. **Import** or **OK**, and enter the appropriate file name and location.




## To Export to a Text File

1. **CoordFile | Export to Ascii...**
2. Optionally **Load...** a **Configuration file**.
3. Choose which **Fields to export** to the Ascii file. (See Point Record Fields in § Point Records.) You may output Desc, Description, or Extension fields, as well as or instead of Code, but be sure to select a valid **Codes Table** which should contain a Desc, Description, and Type for each Code. (See § Point Codes.)
4. Choose either **Delimited** or **Fixed** format.
5. Choose which **Points** to export: **All** or **Filter..** a subset. (See § Point Filters.)
6. Optionally **Apply Transformation** to the points during export. (The coordfile will not be changed.) Clicking **Yes...** allows you to enter the transformation parameters (see § Coordinate Transformations.)

7. Choose other options as appropriate, such as whether to **Put headings on the first line**, **Append to file, if it exists** (the default is to overwrite), or the number of **Decimals** to output.
8. Optionally **Save** the **Configuration** for later reuse.
9. **Export** or **OK**, and enter an appropriate file name and location.
10. Optionally **List** the file for review.




## Notes

- A **Configuration file** has any name, preferably with a **.cfg** extension.
- The number of **Decimals** to output is *local* to this module (i.e., the setting doesn't affect other modules).

- The **Moving Filter** boxes allow you to specify the parameters of a spatial filter — to reduce the number of points imported from a densely-sampled stream of points. A point is accepted only if it lies at or outside a **distance** of, **and/or** at or outside a **rise/fall** of, the previously accepted point. (For any given distance and rise/fall, choosing **and** instead of **or** can result in slightly fewer points passing the filter.)
- The **Start** and **Width** boxes for Fixed Format fields should correspond to the first character position and the character count for the field, along a line of data.

## 2. Importing Copan Coordfiles

- Coord**File | Import from Copan PTS...**

## 3. Exporting to Total Station Files

If you wish to export points directly to a total-station-compatible file, see § Export to Total Station File.

updated: 24-Mar-2011

updated: 21-Sep-2010

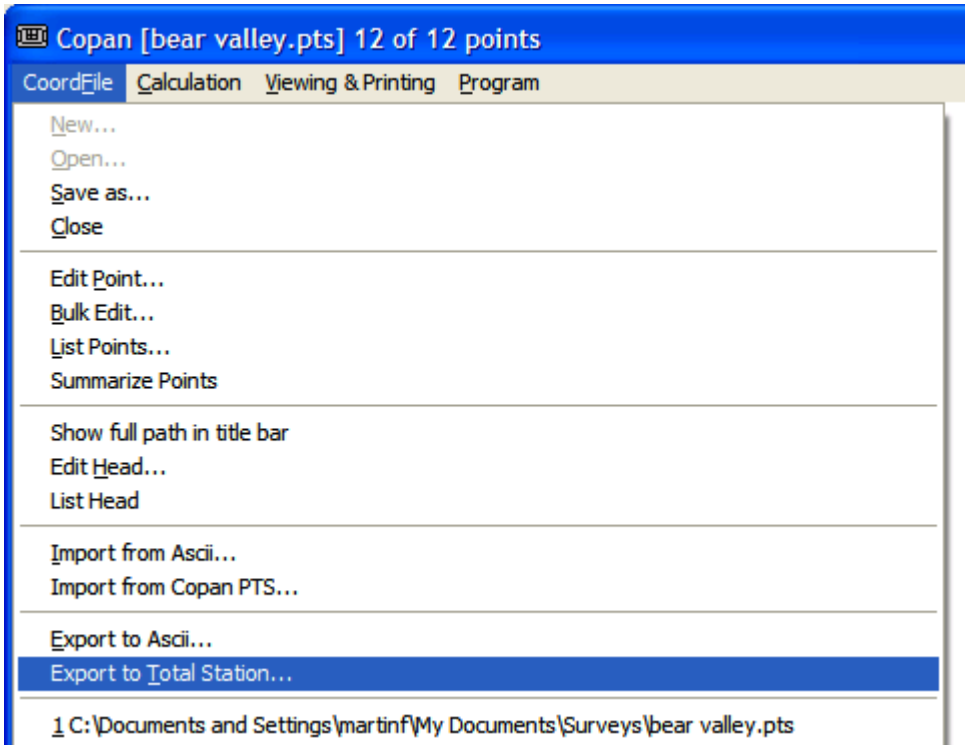
*Copan for Windows*

# Export to Total Station File

Suppose you have some coordfile points that you'd like to set out (or stake out) via a total station. With this module you can save a group of coordfile points to an ASCII file, ready for transferring to a total station.

While you *can* simply export the points to an ASCII file using the generic export facility (see § Import and Export) and then convert the export file using some appropriate external software to the total station format, here you can copy the points *directly* to a total-station-compatible file.

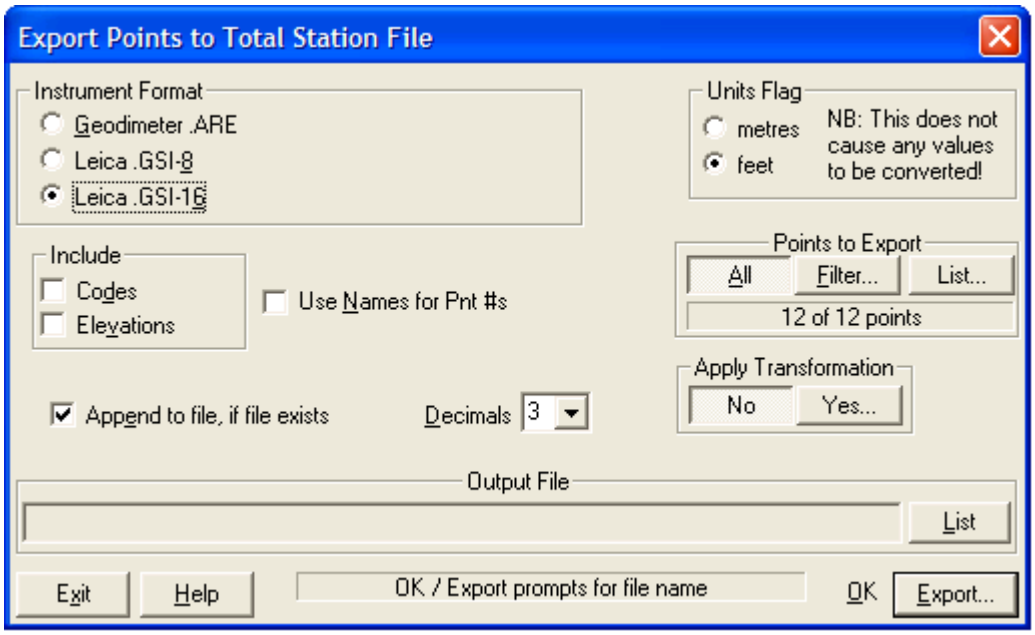
## To Export Points to a Total Station File




1. **CoordFile | Export to Total Station...**
2. Choose the required **Instrument Format**:
   - Geodimeter **.ARE**
   - Leica **.GSI-8**
   - Leica **.GSI-16**
3. Choose whether or not to **Include** point **Codes** or **Elevations**.
4. Choose whether or not to **Use** (alphanumeric) point **Names** instead of (integer) **Pnt #s**.

5. Choose the subset of **Points** to edit — All or a Filtered set (See § Point Filters).
6. Optionally **Apply Transformation** to the points during export. (The coordfile will not be changed.) Clicking **Yes...** allows you to enter the transformation parameters (see § Coordinate Transformations.)
7. Choose other options as appropriate.
8. **OK**.

Let us know ( at support@underhill.ca ) if you'd like Copan to save point data in other instrument formats. Please note the software version and release date (see § Software Version) in all communications.




## Notes

- Choosing a **Units Flag** only sets a flag in the total station file; it has *no* scaling effect on the coordinate values. To actually convert from coordfile units to field units, choose Yes... under **Apply Transformation** and enter an appropriate Units Factor in the other dialog.
- While you can choose the coordinate precision of an output file to be between 0 and 9 **Decimals**, a Leica GSI file is limited to just 3, 4 or 5 decimals.

updated: 21-Sep-2010

*Copan for Windows*

# Part III: Survey Calculations

- COGO Calculations
- Multiple Inverse Calculations
- Field Data Processing
- Field Bearings Processing
- Field FreeStations Processing
- Map Traverses
- Map Checks
- Area and Perimeter Calculations
- Coordinate Transformations

updated: 21-Sep-2010

updated: 21-Sep-2010

***Copan for Windows***

# COGO Calculations

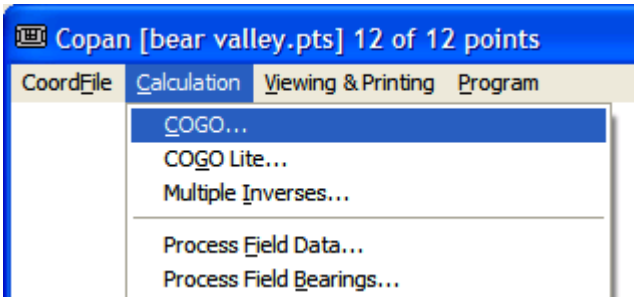
Contents



Using Copan's COGO — *coordinate geometry* — calculator you can do many different computations involving *bearings* (or *azimuths*), *distances* (or *ranges*), *offsets*, *angles*, and coordfile *points*.

Note that to do freestation calculations, use the § Field FreeStations Processing module.

### To COGO




- **Calculation | COGO...** or **CoGo Lite**.

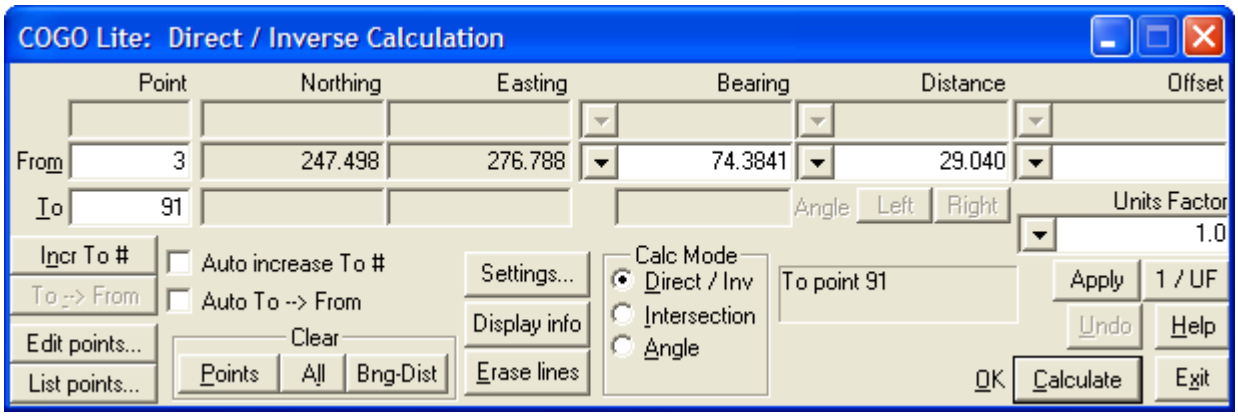
Note that in all of the COGO calculations below,

- **CoGo Lite** has all of the same functionality as has **COGO** except the ability to use an Azimuth Correction or Scale Factor or to view and enter point Names, Codes and Notes.
- Distances are all horizontal (not sloped).
- When entering bearings, distances and offsets, you may use bearing and distance *expressions* (see § Bearing Expressions and § Distance and Offset Expressions), or you can select a previously entered or computed value from the drop-down lists.
- An *offset* is a (relatively small) optional distance perpendicular to a given bearing — positive right and negative left.

- An intersection line may be specified as being offset from and parallel to a given point and bearing by entering an *offset expression* (see § Distance and Offset Expressions) along with the point and bearing.
- If you select **Auto increase To #**, you should not need to enter a new To# (unless you'd like an alternative to that provided). Copan will automatically provide an unused To# when a From# is entered.
- Use the **To --> From** button if you wish to transfer the just-calculated To point # to a From point #, ready for another calculation.
- If entering a **Note** (not in CoGo Lite) for a new point, you can select a previously entered value from the drop-down list.
- If desired, use an **Azimuth Correction** (not in CoGo Lite) to adjust a bearing: Enter an angle, optionally negate it via **–AC**, then **Apply** it to convert the current contents of the **Bearing** box or boxes.
- If desired, use a **Scale Factor** (not in CoGo Lite) or **Units Factor** to scale a distance and offset: Choose a factor, optionally invert it via **1/SF** or **1/UF**, then **Apply** it to convert the current contents of the **Distance** and **Offset** box or boxes.
- You do *not* need to save the coordfile — in fact there is no Coordfile | Save menu item — because whenever you add or update points, Copan updates the coordfile instantaneously, wherever it is stored.

### 1. To do a Direct (or *traverse*) Calculation (compute new point using bearing, distance and offset from existing point)

- Click the **Direct / Inv** radio button.
- Enter the **From** point #.
- Enter the **Bearing**.
- Enter the **Distance** and **Offset**, one of which may be blank or zero.
- Enter the new **To** point #.
- **Calculate** or **OK**. The new point will be added.

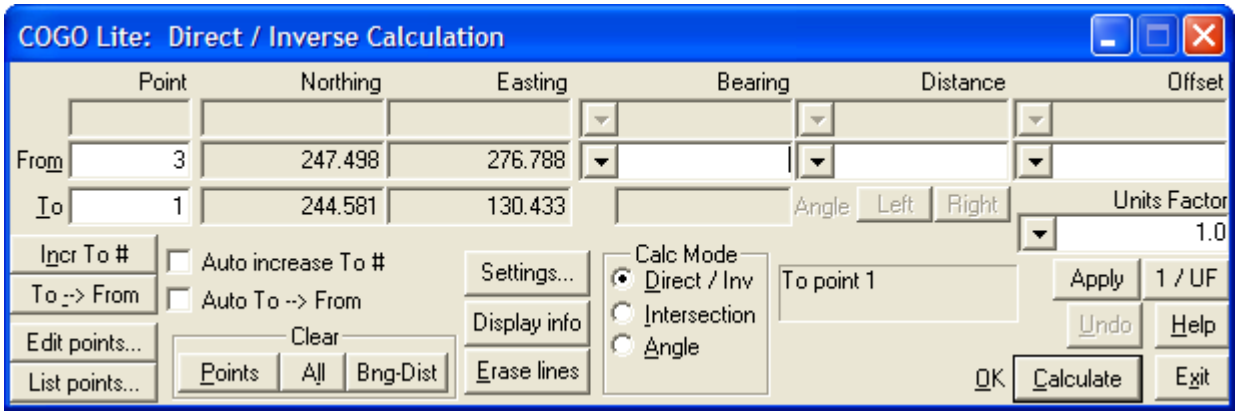



To do a long sequence of direct calculations — especially if you wish to adjust the traverse for closure — use the § Map Traverses module.

To do multiple direct calculations from one to many points, use the § Field Bearings Processing module.

### 2. To do an Inverse (or *join*) Calculation (compute bearing, distance and offset between two existing points)

- Click the **Direct / Inv** radio button.
- Enter the **From** point #.
- Enter the **To** point #.
- Optionally, enter a **Bearing** constraint or an **Offset** constraint.
- **Calculate** or **OK**. The bearing and distance will be computed. If you entered a bearing or offset constraint, the bearing, distance and offset will be computed.

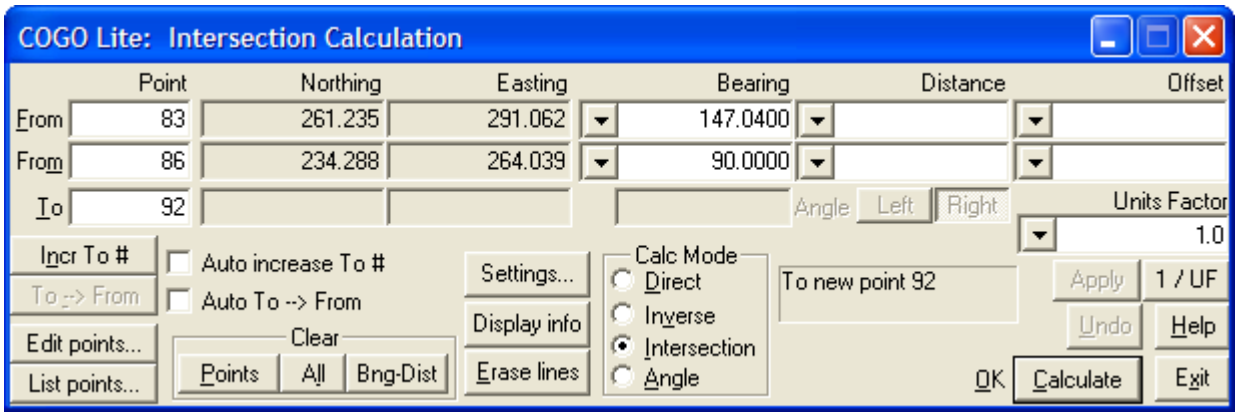



To do a long sequence of inverse calculations between successive points, use the § Area and Perimeter Calculations module.

To do multiple inverse calculations from one to many points, use the § Multiple Inverse Calculations module.

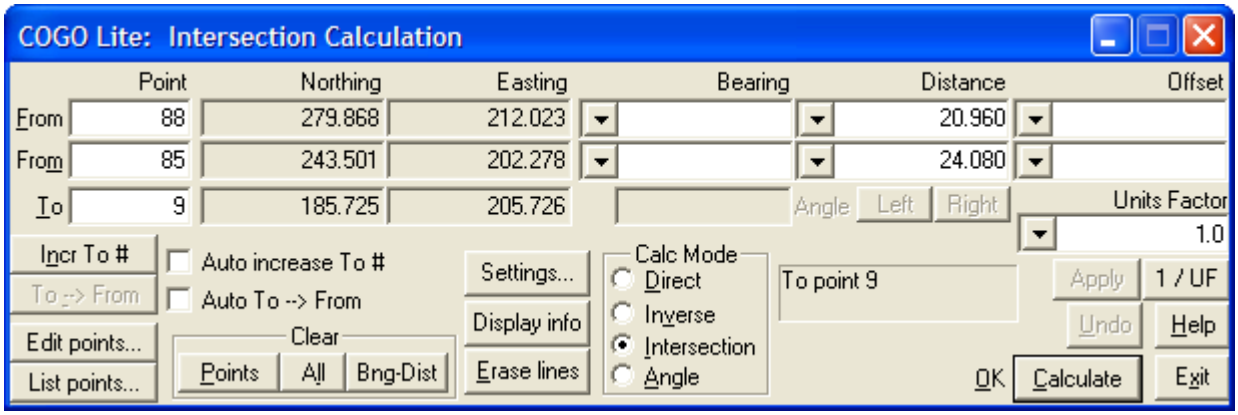
### 3. To do a Bearing-Bearing (or *line-line*) Intersection (intersect bearings from two existing points to new point)

- Click the **Intersection** radio button.
- Enter one **From** point # (in either of the two From boxes).
- Enter the relevant **Bearing** and optional **Offset** (in the same line).
- Enter the other **From** point # (in the other From box).
- Enter the relevant **Bearing** and optional **Offset** (in the same line).
- Enter the new **To** point #.
- **Calculate** or **OK**. The distances, or distances and offsets, to the new point will be computed and the new point will be added.




### 4. To do a Distance-Distance (or *arc-arc*) Intersection (intersect distances from two existing points to new point)

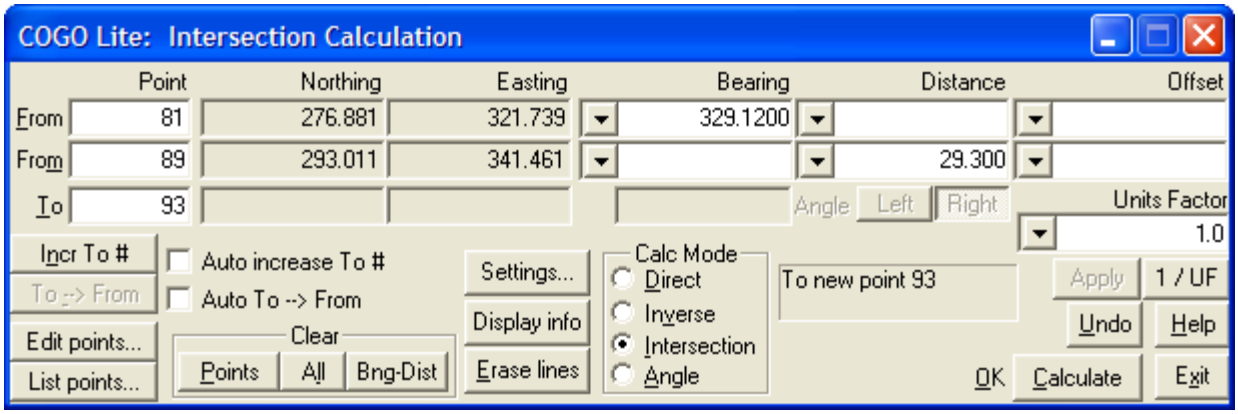
- Click the **Intersection** radio button.
- Enter one **From** point # (in either of the two From boxes).
- Enter the relevant **Distance** (in the same line).
- Enter the other **From** point # (in the other From box).
- Enter the relevant **Distance** (in the same line).
- Enter the new **To** point #.
- **Calculate** or **OK**.
- Choose which of the two solutions you require. The bearings to the new point will be computed and the new point will be added.




### 5. To do a Bearing-Distance (or *line-arc*) Intersection (intersect bearing from one point and distance from another, to new point)

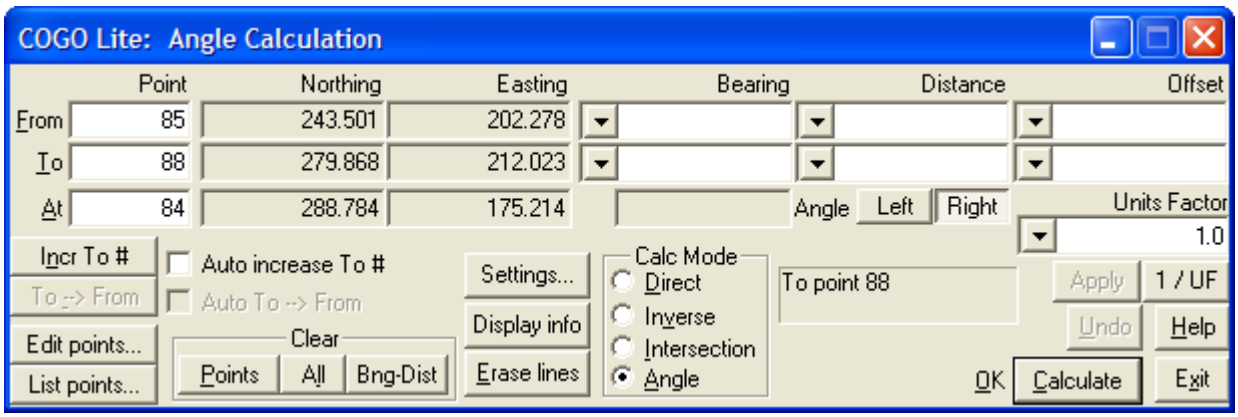
- Click the **Intersection** radio button.
- Enter one **From** point # (in either of the two From boxes).
- Enter the relevant **Bearing** and optional **Offset** (in the same line).
- Enter the other **From** point # (in the other From box).
- Enter the relevant **Distance** (in the same line).
- Enter the new **To** point #.
- **Calculate** or **OK**.

- If necessary, choose which solution you require. The missing dimensions to the new point will be computed and the new point will be added.




## 6. To do an Angle Calculation (compute corner angle at one point, between two others)

- Click the **Angle** radio button.
- Enter the *at* point # in the **At** box.
- Enter the *from* point # in the **From** box.
- Enter the *to* point # in the **To** box.
- Optionally, click Angle **Left** if you need the counter-clockwise angle.
- **Calculate** or **OK**. The bearings and distances from the *at* point to the *from* and *to* points will be computed.
- The resulting horizontal angle is shown under the two **Bearing**s (and by default is clockwise from *From*, to *To*, at *At*).




To do multiple angle calculations at one point, from one reference point, to many points, use the § Multiple Inverse Calculations module.

updated: 21-Sep-2010

updated: 21-Sep-2010

***Copan for Windows***

# Multiple Inverse Calculations

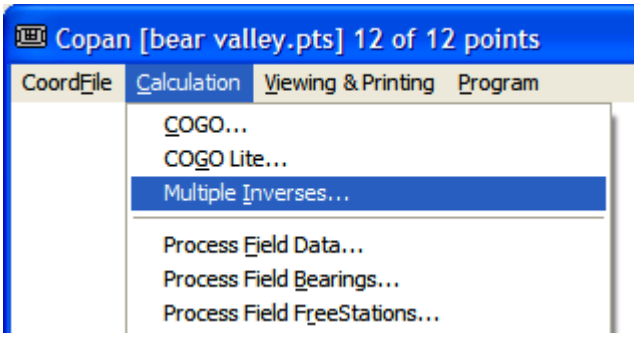
Sometimes you'd like to do several inverse calculations, between one *at point* (or *hub point*) and multiple *to points*. A list of angles and distances may be useful for *staking out* (or *setting out*) designed points. A table of bearings and distances (or *radial ties*) may be required on a legal plan. A display of distances ("chainages" or *stations*) and perpendicular offsets may be useful for checking the alignment and spacing of surveyed points.

Use this special-purpose coordinate geometry module to calculate sets of

1. radial *angles* (from a reference) and *distances*,
2. radial *bearings* and *distances*, or
3. *distances* along and perpendicular *offsets* from a base line,

*from* one start point *to* a set of other points, in a coordfile.

## To Multi Inverses



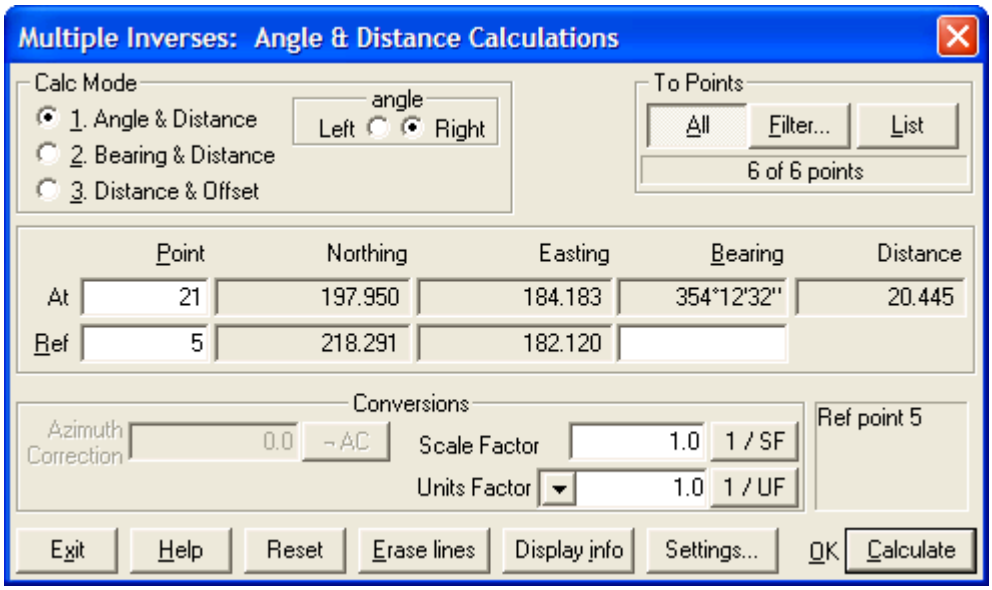

- **Calculation | Multiple Inverses...**

Note that in all of the Multi Inverse calculations below,

- You can choose which **To Points** to include in the results — **All** or a **Filter**ed set (see § Point Filters).
- If entering a bearing (or azimuth), you may use a bearing *expressions* (see § Bearing Expressions.
- Resulting distances are all horizontal (not sloped).
- Optionally, use a **Scale Factor** or **Units Factor**, *before* Calculate, to scale the computed distances.

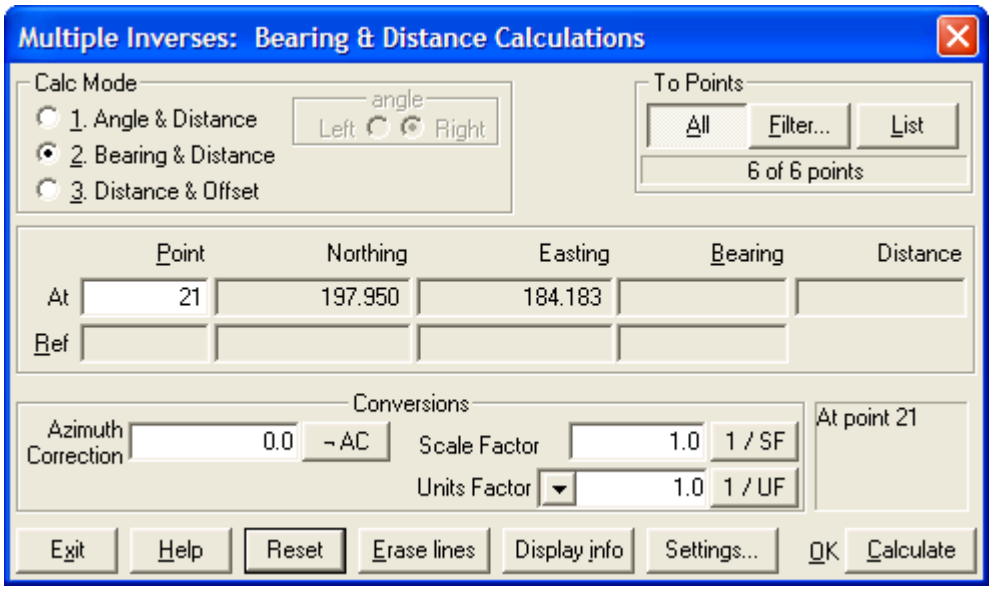
## 1. To Calculate Many Angle-Distance Pairs

1. Click the **1. Angle & Distance** Calc Mode, and optionally choose **angle Left** for negative, counterclockwise angles to be computed.
2. Enter the **At Point** (or hub point) number.
3. Enter the **Ref**erence **Point** number or **Ref**erence **Bearing**.
4. **Calculate** or **OK**.



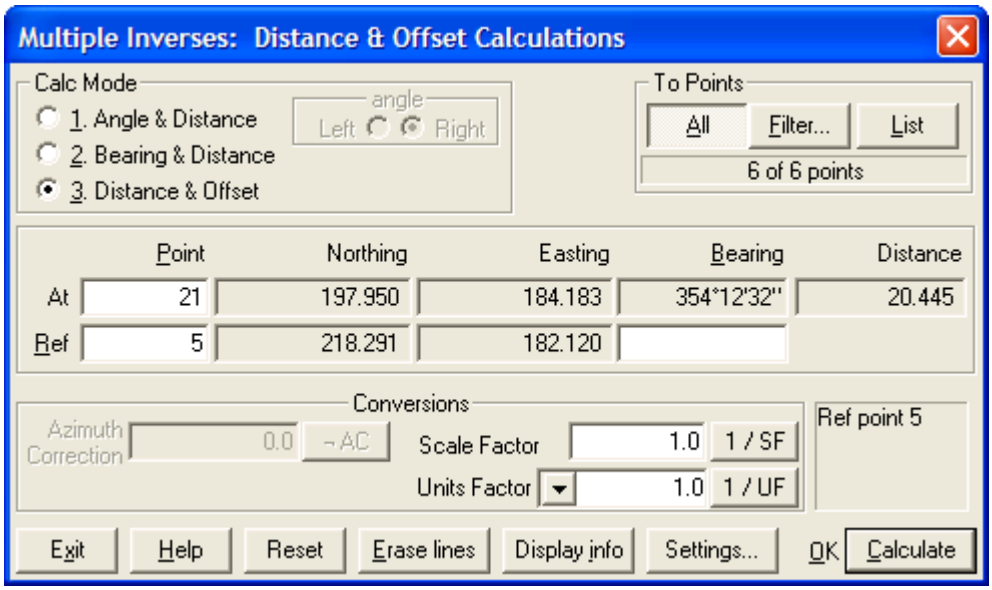

## 2. To Calculate Many Bearing-Distance Pairs

1. Click the **2. Bearing & Distance** Calc Mode.
2. Enter the **At Point** (or hub point) number.
3. Optionally, use an **Azimuth Correction**, *before* Calculate, to adjust the computed bearings.
4. **Calculate** or **OK**.




## 3. To Calculate Many Distance-Offset Pairs

1. Click the **3. Distance & Offset** Calc Mode.
2. Enter the **At Point** (or start point) number.
3. Enter the **Ref**erence **Point** number or **Ref**erence **Bearing**.
4. **Calculate** or **OK**.




updated: 21-Sep-2010

updated: 24-Mar-2011

*Copan for Windows*

# Field Data Processing

Contents

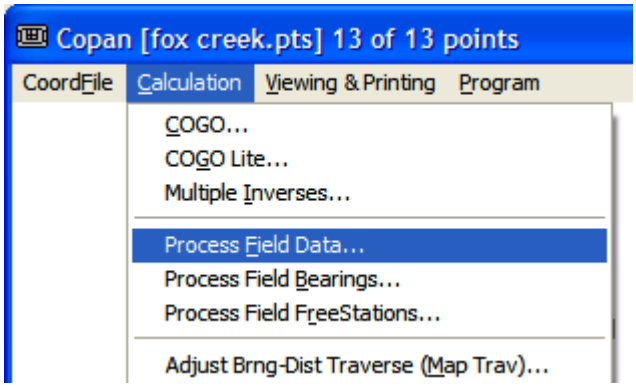


Use this module to process raw field data from survey instruments such as *total station*, or *theodolite* and *EDM*. Copan can process the data — horizontal and vertical circle readings, slope distances, and instrument and signal (or target) heights — whether concerning connected *traverses*, independent *radial surveys*, or combinations thereof, to calculate new points. Data may come from *data logger* files or be manually input from *field books*. Traverses with redundant angles will have *angles balanced*. Closed traverses may have their *coordinates adjusted* for misclosure and new points may be saved.

Note that intersections, resections, freestations, and redundant observations (other than at the end of a traverse) are *not* handled here. (For intersections, use the § COGO Calculations module. For freestations, use the § Field FreeStations Processing module.) And this module cannot easily be used for "map traverses". (For such needs, use the § Field Bearings Processing, § Map Traverses, or § Map Checks modules.)

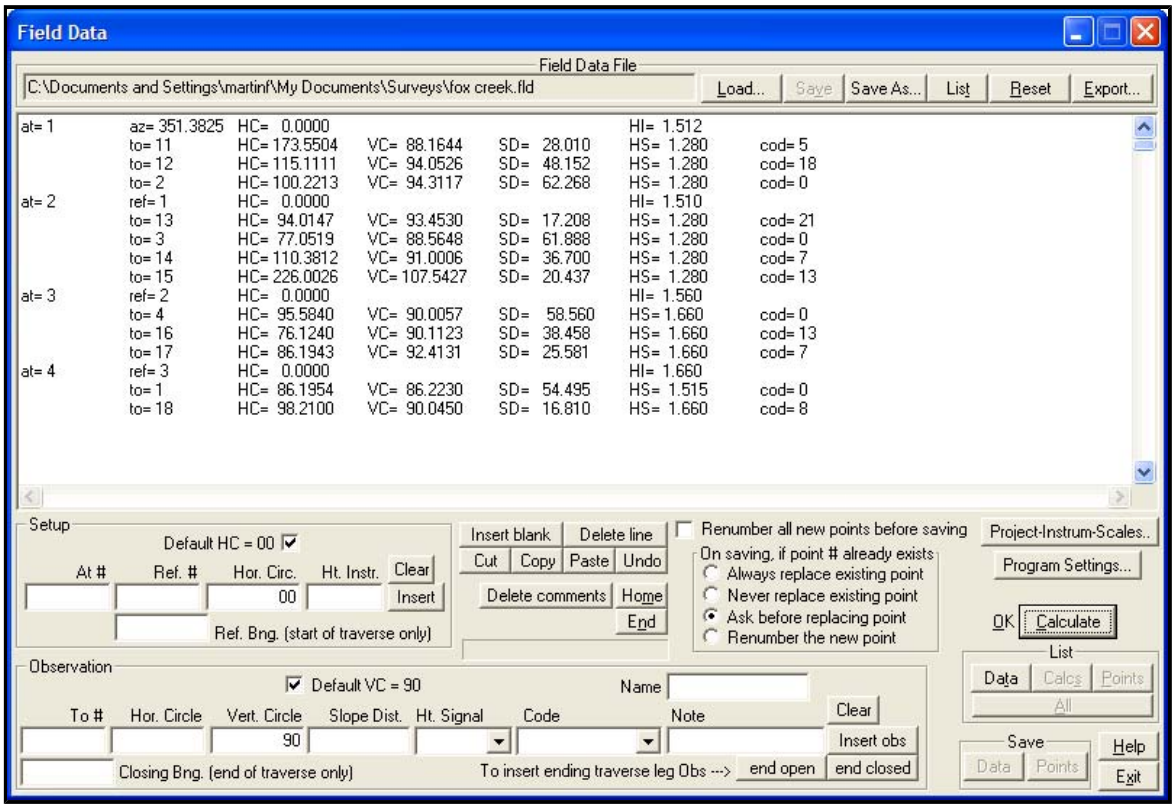
Also note that, while there are certain similarities between the Map Traverse and the Field Data modules, there are various operational differences (other than the type of map/survey data involved). If you are familiar with one and new to the other, please study the appropriate manual and dialog carefully.

## 1. To Process Field Data




1. **Calculation | Process Field Data...**
2. Optionally **Load...** a **Field Data File**. (See § Field Files.)

3. Edit data in the big edit box as required. NB: To move the text cursor: Use the arrow, Tab, or Enter keys, or the mouse pointer. To manually insert a tab, copy and paste an existing one. To go quickly to the beginning or end of your data in the big edit box — especially useful if you have a huge set of data to peruse — use the **Home** and **End** buttons, respectively.
4. To enter setups and observations data, use the mini edit boxes and the relevant **Insert** button. To enter project, instrument and scales data, use the supplementary dialog via the **Project-Instrum-Scales..** button.
5. Optionally **Export..** the **Field data file** in (the BitWise Ideas survey network least-squares adjustment program) GeoLab IOB format.
6. **Save** the **Field data file** for reuse.
7. Optionally **Delete comments**.
8. Optionally change the precision of displayed/listed distances and coordinates via **Settings...**.
9. Optionally **Scale extended-code sizes**. This will scale any observation P-codes that are sized according to the rules in § Point Codes by a Units Factor you have set.
10. **Calculate** (or **OK**) to process the field data, adjusting the traverses if appropriate (see § Traverse Processing).
11. To graphically view the shots: Close the Info Display window if it is open, and minimize or move aside — but do not close — the Field Data window.
12. Optionally **List** the field **Data**, the **Calcs**, or the **Points** that have been computed.
13. Optionally choose whether to **Renumber new points** or **replace existing points** (see § Point Renumbering or Replacement) and **Save** the **Points** that have been computed.




## 2. Field Data Entry

### Setups and Observations Data Entry

- For the *first* instrument **Setup** (and backsight) in a traverse — and for each non-traverse setup — insert the following:
  - **At #** the station point number,
  - *Either* **Ref.#** the reference (or backsight) point number,
    *or* **Ref. Bng.** an assumed reference azimuth (or bearing),
  - **Hor. Circ.** the reference horizontal circle value,
  - **Ht. Instr.** the instrument height — if elevations are required.

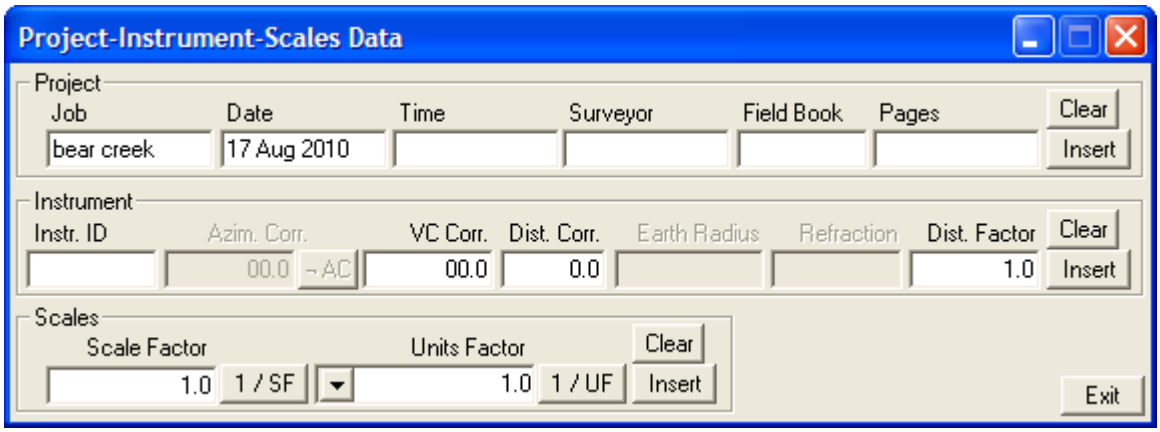
  They stay in effect until a new Setup is inserted. After a Setup line, multiple side-shots (or ties) and one traverse leg shot can appear in any order.
- For each traverse **Setup** (and backsight) — *except* the first — insert
  - **At #** the station point number,
  - **Ref.#** the reference (or backsight) point number,
  - **Hor. Circ.** the reference horizontal circle value,
  - **Ht. Instr.** the instrument height — if elevations are required.
- For each conventional *shot* **Observation** (whether a traverse leg or a radial point), insert the following:
  - **To #** the target point number,
  - **Hor. Circle** the horizontal circle value,
  - **Vert. Circle** the vertical circle value (or *zenith angle*),
  - **Slope Dist.** the slope distance,

- **Ht. Signal** the signal (*target*, *prism*, *reflector*, or *pogo*) height — if elevations are required,
- **Code** a point code, if needed,
- **Note** a point note, if needed,
- **Name** a point name, if needed.

- For a remote elevation measurement (REM), insert it as an **Observation** but without a Slope Distance (or possibly also without a Horizontal Circle). See Remote Elevation Measurement in § Field Files for details.
- For the *last* leg shot of each traverse, be sure to use one of these buttons:
  - **end open** for an *open* traverse (i.e., ending on an unknown point), or
  - **end closed** for a *closed* traverse (i.e., ending on a control point or on the traverse's initial setup point).

  If the last leg shot of a traverse has not been so identified, Copan may not report results properly.
- After the end of a traverse, a *closing angle* **Observation** may be inserted:
  - *Either* **To #** the closing target point number,
    *or* **Closing Bng.** the assumed closing azimuth (or bearing), and
  - **Hor. Circle** the horizontal circle value.
  - While you may include a Code, Note or Name here, they will be *ignored*.

  Copan will correct any angular misclosure, by distributing the error *equally* among the setups, prior to traverse adjustment.




**Project, Instrument and Scales Data Entry**

Keep these three types of data on separate lines:

- Insert any desired **Project** data (Job, Date, Time, Surveyor, Field book, or Pages) wherever convenient. Note, however, that such data are ignored during calculations.
- Insert any **Instrument** constants, whenever they differ from their previous values. These are (along with their default values)
  - **Instr. ID** () ignored during calculation,
  - **VC Corr.** (0.0) for adding to vertical circle values,
  - **Dist. Corr.** (0.0) for slope distance additive correction (e.g., prism constant),

    - **Dist. Factor** (1.0) for slope distance multiplicative correction (e.g., due to extreme temperature),

  and they stay in effect until new ones are inserted.

- Insert any **Scales**, whenever they differ from their previous values. These are
    - **Scale Factor** (1.0) for combined map projection and sea-level (or elevation) distortions — applies only to horizontal distances,
    - **Units Factor** (1.0) for height and distance unit conversion (e.g., feet to metres),

  and they stay in effect until new ones are inserted.

Be sure to use the proper correction. In most cases, the default values are appropriate.

## 3. Field Files

Copan can read field data in different industry formats as well as its own. A native Copan field file consists of numerous Setup and Observation data lines and may contain comment lines, as well as Project, Instrument, or Scale data lines, if required. Each data line consists of various tab-separated *attribute* = *value* pairs. See § Field Files for a more detailed description of different raw field file types and Copan's field file format.

## 4. Field Data File Examples

In all these examples, codes, notes and names have been omitted for brevity.

#### Example 1: 2D (closed) loop traverse.

A looping traverse can be oriented via an assumed bearing or via an assumed control point. Here's a traverse, oriented with an assumed azimuth of 180.

```
at= 509  az= 180.0000  HC=   0.0000
         to= 508       HC= 270.2051  VC=  90.0000  SD= 244.260
at= 508  ref= 509      HC=   0.0000
         to= 507       HC= 284.3940  VC=  90.0000  SD= 189.115
at= 507  ref= 50       HC=   0.0000
         to= 506       HC= 164.5417  VC=  90.0000  SD= 179.562
at= 506  ref= 507      HC=   0.0000
         to= 505       HC= 270.3249  VC=  90.0000  SD= 195.591
at= 505  ref= 506      HC=   0.0000
         to= 509       HC= 269.3231  VC=  90.0000  SD= 362.140
         end= closed
at= 509  ref= 505      HC=   0.0000
         az= 180.0000  HC= 360.0000
```

Here's the same traverse, oriented with reference to a dummy point 999, which has been placed due south of the start.

```
at= 509  ref= 999      HC=   0.0000
         to= 508       HC= 270.2051  VC=  90.0000  SD= 244.260
at= 508  ref= 509      HC=   0.0000
         to= 507       HC= 284.3940  VC=  90.0000  SD= 189.115
at= 507  ref= 50       HC=   0.0000
         to= 506       HC= 164.5417  VC=  90.0000  SD= 179.562
at= 506  ref= 507      HC=   0.0000
         to= 505       HC= 270.3249  VC=  90.0000  SD= 195.591
```

```
at= 505  ref= 506      HC=   0.0000
         to= 509       HC= 269.3231  VC=  90.0000  SD= 362.140
         end= closed
at= 509  ref= 505      HC=   0.0000
         to= 508       HC= 270.2051
```

**Example 2: 3D linear traverse.**

Here's a linear traverse, with the end point as an unknown (i.e., an open traverse).

```
at= 144  ref= 519      HC=   0.0000                              HI=  1.648
         to= 38        HC=  59.3848  VC=  89.1506  SD= 190.160  HS=  1.583
at= 38   ref= 144      HC=   0.0000                              HI=  1.583
         to= 39        HC= 152.1009  VC=  83.1043  SD=  90.118  HS=  1.553
at= 39   ref= 38       HC=   0.0000                              HI=  1.553
         to= 36        HC= 192.5206  VC=  86.0415  SD=  84.178  HS=  1.483
at= 36   ref= 39       HC=   0.0000                              HI=  1.483
         to= 198       HC= 232.5554  VC=  86.2758  SD= 111.153  HS=  1.510
         end= open
```

Here's the same traverse, with the end point as control (i.e., a connecting traverse).

```
at= 144  ref= 519      HC=   0.0000                              HI=  1.648
         to= 38        HC=  59.3848  VC=  89.1506  SD= 190.160  HS=  1.583
at= 38   ref= 144      HC=   0.0000                              HI=  1.583
         to= 39        HC= 152.1009  VC=  83.1043  SD=  90.118  HS=  1.553
at= 39   ref= 38       HC=   0.0000                              HI=  1.553
         to= 36        HC= 192.5206  VC=  86.0415  SD=  84.178  HS=  1.483
at= 36   ref= 39       HC=   0.0000                              HI=  1.483
         to= 198       HC= 232.5554  VC=  86.2758  SD= 111.153  HS=  1.510
         end= closed
```

Here's the same closed traverse, with a closing angle.

```
at= 144  ref= 519      HC=   0.0000                              HI=  1.648
         to= 38        HC=  59.3848  VC=  89.1506  SD= 190.160  HS=  1.583
at= 38   ref= 144      HC=   0.0000                              HI=  1.583
         to= 39        HC= 152.1009  VC=  83.1043  SD=  90.118  HS=  1.553
at= 39   ref= 38       HC=   0.0000                              HI=  1.553
         to= 36        HC= 192.5206  VC=  86.0415  SD=  84.178  HS=  1.483
at= 36   ref= 39       HC=   0.0000                              HI=  1.483
         to= 198       HC= 232.5554  VC=  86.2758  SD= 111.153  HS=  1.510
         end= closed
at= 198  ref= 36       HC=   0.0000
         to= 265       HC= 127.4300
```

**Example 3: 3D closed (loop) traverse with side-shots.**

Notice that side-shots (or ties) may precede leg-shots (e.g., 3 and 8) or they may follow leg-shots (e.g., 6 and 8).

```
at= 1    az= 351.3825  HC=   0.0000                              HI=  1.515
         to= 3         HC= 173.5504  VC=  88.1644  SD=  28.010  HS=  1.280
         to= 5         HC= 100.2213  VC=  94.3117  SD=  62.271  HS=  1.280
at= 5    ref= 1        HC=   0.0000                              HI=  1.510
         to= 7         HC=  77.0519  VC=  88.5648  SD=  61.888  HS=  1.280
```

```
         to= 6         HC=  94.0147  VC=  93.4530  SD=  17.208  HS=  1.280
at= 7    ref= 5        HC=   0.0000                             HI=  1.567
         to= 8         HC= 226.5026  VC= 107.5427  SD=  20.437  HS=  1.280
         to= 2         HC=  96.1240  VC=  90.1123  SD=  58.458  HS=  1.660
at= 2    ref= 7        HC=   0.0000                             HI=  1.660
         to= 1         HC=  86.1954  VC=  86.2230  SD=  54.495  HS=  1.515
         end= closed
         to= 9         HC=  73.5238  VC=  92.4451  SD=  19.872  HS=  1.510
at= 1    ref= 2        HC=   0.0000                             HI=  1.515
         az= 351.3825  HC= 360.0000
```

**Example 4: 3D topographic survey with REM points.**

Notice that a REM (remote elevation measured) point may be before (515) or after (507) its base point.

```
at= 1  ref= 5     HC= 00.0000                          HI= 1.614
       to= 506    HC= 60.2550  VC= 79.4345  SD= 7.048  HS= 1.418
       to= 507                 VC= 64.3500                       bas= 506
       to= 515                 VC= 70.4040                       bas= 516
       to= 516    HC= 25.1415  VC= 86.1720  SD= 11.581 HS= 1.418
```

## 5. Notes

- While repetition and redundancy are common in field surveying, Copan can only process simple traverses consisting of reduced observations. Multiple sets of observations of the same quantities and cross-ties cannot be processed without some manual editing. For example, prior to processing, repeated sets of angles and distances must be averaged and check shots to the same points from different setups must be renumbered.
- If you need to manually insert a tab, such as when you wish to add a code or note to an observation line, copy and paste an existing tab — there will always be one between two auto-inserted *attribute=value* pairs.
- When you Calculate, Copan always checks and computes the field data currently in the big edit box.
- Changes to field data in the big edit box are only saved (to disk) when you Save Data.
- Elevations are carried forward, from the known setup to a target, not from a known reference to the setup. To do the latter, you need to trick Copan by manually converting the trigonometric height observation from a backward to a forward form (i.e., switch the HI and HS, and suplement the VC).
- Both negative and zero values for HI or HS are correctly interpreted.
- When a setup point's elevation value is missing (or null), Copan does not use any *rises* to targets (i.e., target elevations will be null).
- Also, a missing HI or HS yields a null *rise* and a null target elevation.
- Newly calculated points are not saved to the coordfile until you click Save Points.
- When you Save Points, Copan first creates an automatic backup copy of the coordfile.

updated: 24-Mar-2011

updated: 24-Mar-2011

*Copan for Windows*

# Field Bearings Processing

Contents

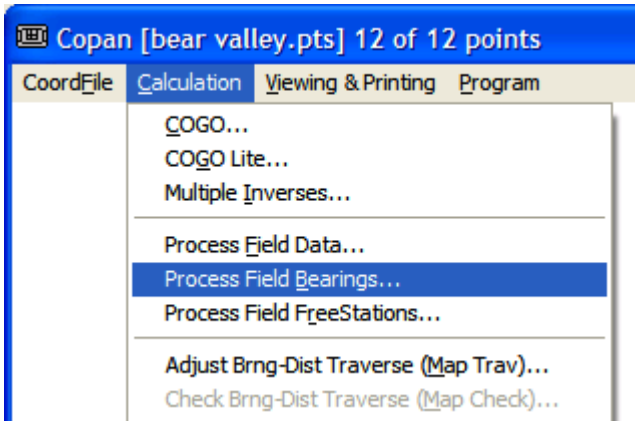


Use this module to process raw field data from hand-held survey instruments such as *compass*, *inclinometer* and *rangefinder*. Copan can process the data — azimuths (or bearings), vertical angles, slope distances, and instrument and signal (or target) heights — whether concerning one or more connected *traverses*, independent *radial surveys*, or combinations thereof, to calculate new points. Data may come from *data logger* files or be manually input from *field books*. Traverses may be adjusted for coordinate misclosure and new points may be saved.

Note that ordinary total station data, intersections, resections, and redundant observations (other than at the end of a traverse) are not handled here. (For ordinary field data, use the § Field Data Processing module. For intersections, use the § COGO Calculations module. For resections, use the § Field Resections Processing module.) And this module is best for raw, *field traverses* with slope distances, not for reduced, *map traverses* with horizontal distances. (For such needs, use the § Map Traverses or § Map Checks modules.)

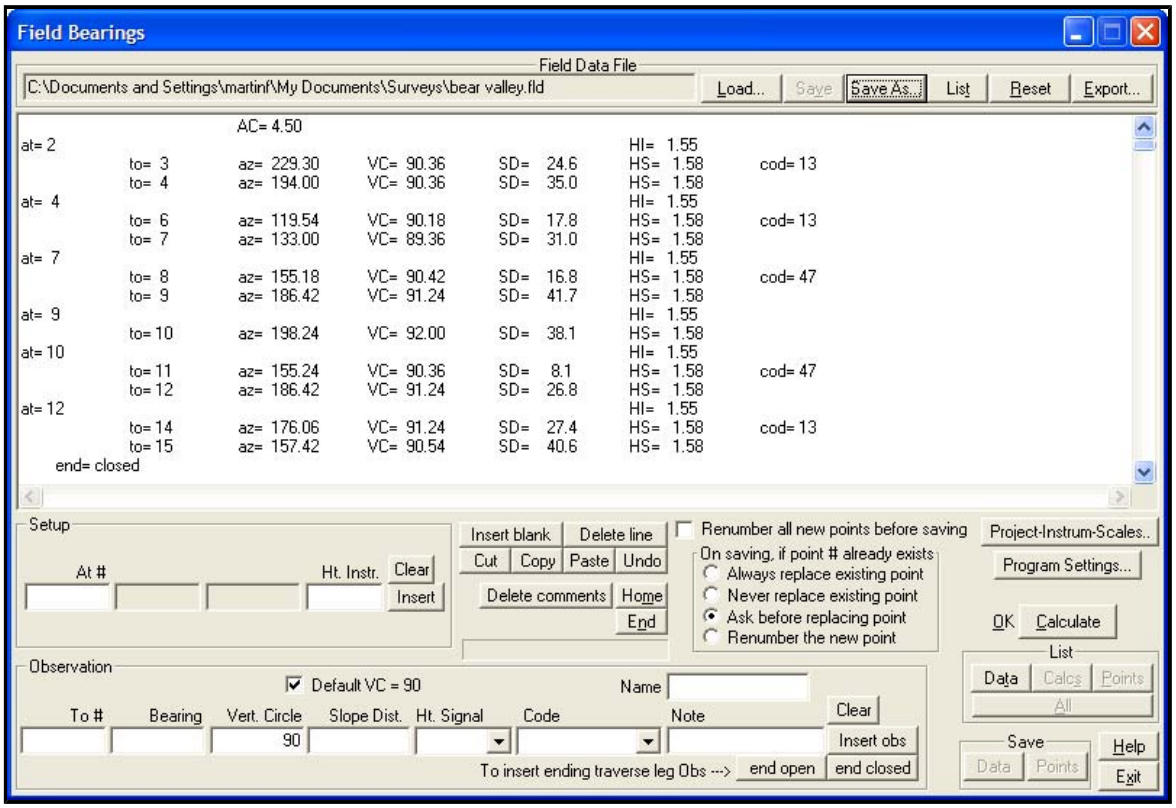
Also note that, while there are certain similarities between the Map Traverse and the Field Data modules, there are various operational differences (other than the type of map/survey data involved). If you are familiar with one and new to the other, please study the appropriate manual and dialog carefully.

## 1. To Process Field Bearings




1. **Calculation | Process Field Bearings...**

2. Optionally **Load...** a **Field Data File**. (See § Field Files.)
3. Edit data in the big edit box as required. NB: To move the text cursor: Use the arrow, Tab, or Enter keys, or the mouse pointer. To manually insert a tab, copy and paste an existing one.
4. To enter setups and observations data, use the mini edit boxes and the relevant **Insert** button. To enter project, instrument and scales data (such as magnetic-grid north correction), use the supplementary dialog via the **More data** button.
5. Optionally **Export..** the **Field data file** in (the BitWise Ideas survey network least-squares adjustment program) GeoLab IOB format.
6. **Save** the **Field file** for reuse.
7. Optionally change the precision of displayed/listed distances and coordinates via **Settings...**.
8. Optionally **Scale extended-code sizes**. This will scale any observation P-codes that are sized according to the rules in § Point Codes by a Units Factor you have set.
9. **Calculate** (or **OK**) to process the field data, adjusting the traverses if appropriate (see § Traverse Processing).
10. To graphically view the shots: Close the Info Display window if it is open, and minimize or move aside — but do not close — the Field Data window.
11. Optionally **List** the field **Data**, the **Calcs**, or the **Points** that have been computed.
12. Optionally choose whether to **Renumber new points** or **replace existing points** (see § Point Renumbering or Replacement) and **Save** the **Points** that have been computed.




## 2. Field Bearings Entry

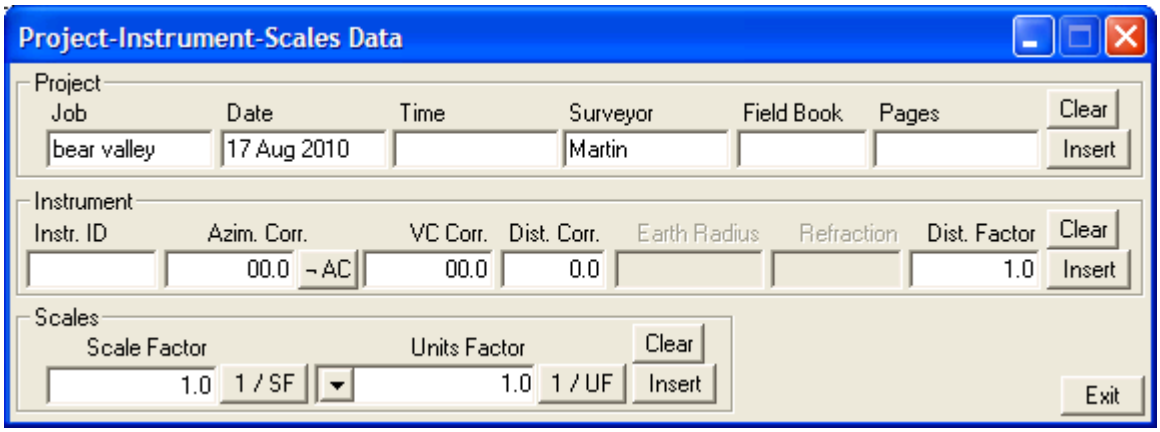
### Setups and Observations Data Entry

- For each instrument **Setup** insert only the following:
  - **At #** the station point number,
  - **Ht. Instr.** the instrument height — if elevations are required.

  They stay in effect until a new Setup is inserted. After a Setup line, multiple side-shots (or ties) and one traverse leg can appear in any order.
- For each *shot* **Observation** (whether a traverse leg or a radial point), insert the following:
  - **To #** the target point number,
  - **Bearing** the bearing value (or a *bearing expression*),
  - **Vert. Circle** the vertical circle value (or *zenith distance*),
  - **Slope Dist.** the slope distance,
  - **Ht. Signal** the signal (or *pogo*) height — if elevations are required,
  - **Code** a point code, if needed,
  - **Note** a point note, if needed,
  - **Name** a point name, if needed.
- For a remote elevation measurement (REM), insert it as an **Observation** but without a Slope Distance (or possibly also without a Horizontal Circle). See Remote Elevation Measurement in § Field Files for details.
- For the *last* leg shot of each traverse, be sure to use one of these buttons:
  - **end open** for an *open* traverse (i.e., ending on an unknown point), or

- **end closed** for a *closed* traverse (i.e., ending on a control point or on the traverse's initial setup point).

If the last leg shot of a traverse has not been so identified, Copan may not report results properly.




**Project, Instrument and Scales Data Entry**

Keep these three types of data on separate lines:

- Insert any desired **Project** data (Job, Date, Time, Surveyor, Field book, or Pages) wherever convenient. Note, however, that such data are ignored during calculations.
- Insert any **Instrument** constants, whenever they differ from their previous values. These are (along with their default values)
  - **Instr. ID** () ignored during calculation,
  - **Azim. Corr.** (0.0) for adding to azimuth (bearing) values (e.g., magnetic to grid),
  - **VC Corr.** (0.0) for adding to vertical circle values,
  - **Dist. Corr.** (0.0) for slope distance additive correction (e.g., prism constant),
  - **Dist. Factor** (1.0) for slope distance multiplicative correction (e.g., due to extreme temperature),

  and they stay in effect until new ones are inserted.
- Insert any **Scales**, whenever they differ from their previous values. These are
  - **Scale Factor** (1.0) for combined map projection and sea-level (or elevation) distortions — applies only to horizontal distances,
  - **Units Factor** (1.0) for height and distance unit conversion (e.g., feet to metres),

  and they stay in effect until new ones are inserted.

Be sure to use the proper correction. In most cases, the default values are appropriate.

## 3. Field Files

Copan can read field data in different industry formats as well as its own. A native Copan field file consists of numerous Setup and Observation data lines and may contain comment lines, as well as Project, Instrument or Scales data lines, if required. Each data line consists of various tab-separated *attribute* = *value* pairs. See § Field Files for a more detailed description of different raw field file types and Copan's field file format.

## 4. Field Bearings File Example

In this example, a two-leg, closed traverse with side-shots is preceded by an azimuth correction, and codes and notes have been omitted for brevity.

```
              AC= 5.24
at= 382                                                HI=  1.55
    to= 800   az= 235.42   VC= 90.18   SD=   4.6   HS=  1.58
    to= 801   az= 226.06   VC= 90.24   SD=  22.2   HS=  1.58
    to= 802   az= 219.54   VC= 90.48   SD=  25.6   HS=  1.58
    to= 804   az= 229.30   VC= 90.36   SD=  24.6   HS=  1.58
at= 804                                                HI=  1.55
    to= 805   az= 194.00   VC= 90.36   SD=  35.0   HS=  1.58
    to= 806   az= 134.18   VC= 90.06   SD=   7.8   HS=  1.58
    to= 807   az= 129.54   VC= 90.18   SD=  17.8   HS=  1.58
    to= 387   az= 133.00   VC= 89.36   SD=  31.0   HS=  1.58
    end= closed
```

## 4. Notes

- While repetition and redundancy are common in field surveying, Copan can only process simple traverses consisting of reduced observations. Multiple sets of observations of the same quantities and cross-ties cannot be processed without some manual editing. For example, prior to processing, repeated sets of angles and distances must be averaged and check shots to the same points from different setups must be renumbered.
- If you need to manually insert a tab, such as when you wish to add a code or remark to an observation line, copy and paste an existing tab — there will always be one between two auto-inserted *attribute=value* pairs.
- When you Calculate, Copan always checks and computes the field data currently in the big edit box.
- Changes to field data in the big edit box are only saved (to disk) when you Save Data.
- Elevations are carried forward, from the known setup to a target, not from a known reference to the setup. To do the latter, you need to trick Copan by manually converting the trigonometric height observation from a backward to a forward form (i.e., switch the HI and HS, and suplement the VC).
- Both negative and zero values for HI or HS are correctly interpreted.
- When a setup point's elevation value is missing (or null), Copan does not use any *rises* to targets (i.e., target elevations will be null).
- Also, a missing HI or HS yields a null *rise* and a null target elevation.
- Newly calculated points are not saved to the coordfile until you click Save Points.
- When you Save Points, Copan first creates an automatic backup copy of the coordfile.

updated: 24-Mar-2011

updated: 24-Mar-2011

*Copan for Windows*

# Field FreeStations Processing

Contents

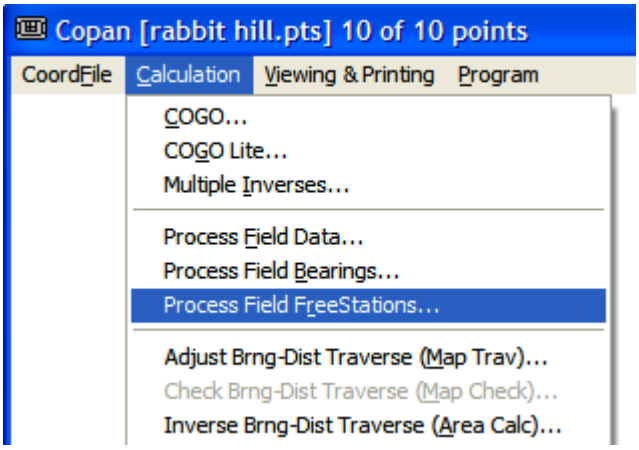


Use this module to process raw *free station* surveys — horizontal and vertical circle readings, slope distances, and instrument and signal heights — based on setups at unknown points and radial shots to known points. Data can come from total station files, data logger files or be manually input. Free stations (or "modern" *resections*) are calculated using the best-fit of angle and distance shots to (up to 10) control points. Resulting setup points and side shot points may be saved. Multiple free station surveys can be processed in a single session.

Note that "classical" resections (involving angles only), intersections, traverses, and two-faced (FL & FR) observations are *not* handled here. (For intersections, use the § COGO Calculations module. For ordinary traverses, use the § Field Data Processing module and for *bearing traverses* use the § Field Bearings Processing module.)

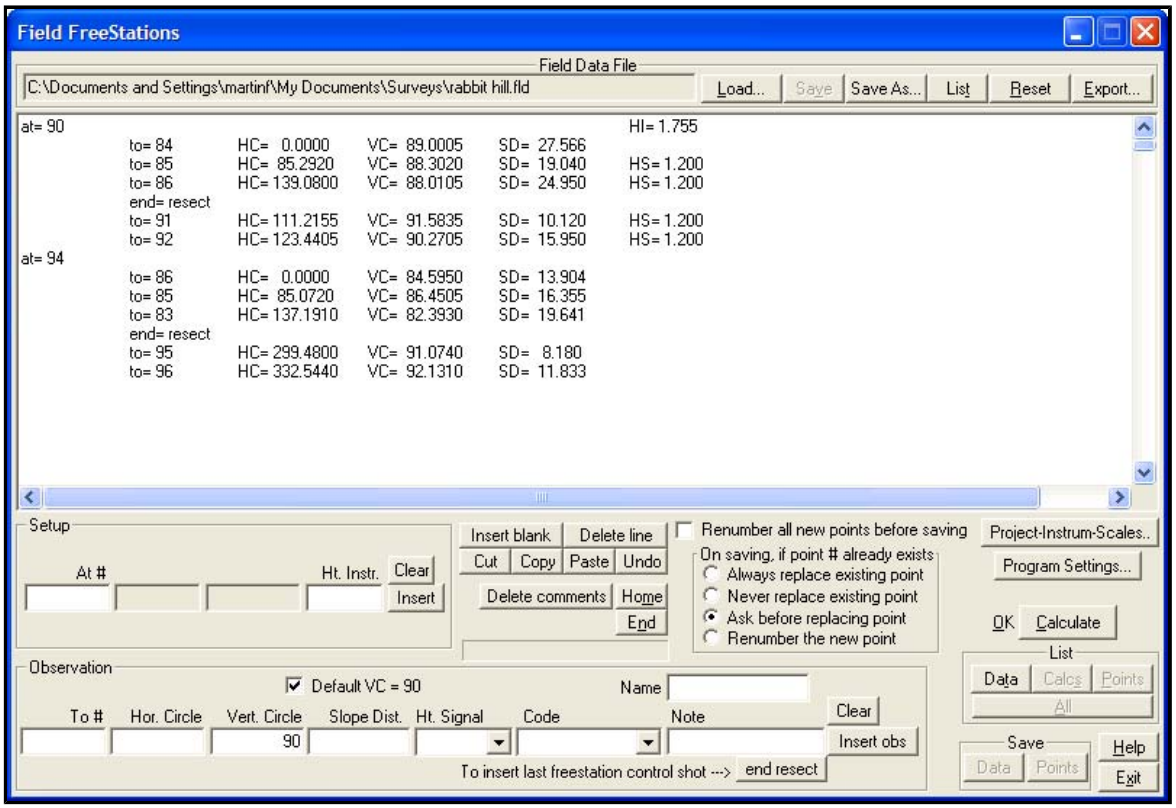
Also note that, while there are certain similarities between the Map Traverse and the Field Data modules, there are various operational differences (other than the type of map/survey data involved). If you are familiar with one and new to the other, please study the appropriate manual and dialog carefully.

## 1. To Process Field FreeStation Data




1. **Calculation | Process Field FreeStations...**
2. Optionally **Load...** a **Field Data File**. (See § Field Files.)

3. Edit data in the big edit box as required. NB: To move the text cursor: Use the arrow, Tab, or Enter keys, or the mouse pointer. To manually insert a tab, copy and paste an existing one.
4. To enter setups and observations data, use the mini edit boxes and the relevant **Insert** button. To enter project, instrument and scales data, use the supplementary dialog via the **More data** button.
5. Optionally **Export..** the **Field data file** in (the BitWise Ideas survey network least-squares adjustment program) GeoLab IOB format.
6. **Save** the **Field file** for reuse.
7. Optionally change the precision of displayed/listed distances and coordinates via **Settings...**.
8. Optionally **Scale extended-code sizes**. This will scale any observation P-codes that are sized according to the rules in § Point Codes by a Units Factor you have set.
9. **Calculate** (or **OK**) to process the field data, accepting the calculations as appropriate.
10. To graphically view the shots: Close the Info Display window if it is open, and minimize or move aside — but do not close — the Field FreeStations window.
11. Optionally **List** the field **Data**, the **Calcs**, or the **Points** that have been computed.
12. Optionally choose whether to **Renumber new points** or **replace existing points** (see § Point Renumbering or Replacement) and **Save** the **Points** that have been computed.

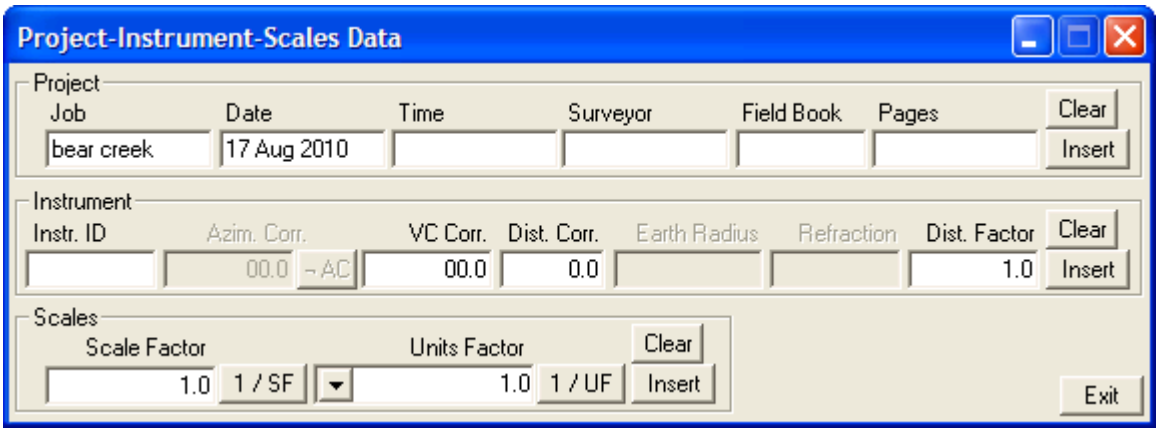


## 2. Field FreeStation Data Entry

### Setups and Observations Data Entry

- For each freestation instrument **Setup**, insert only the following:
  - **At #** the free station point number,
  - **Ht. Instr.** the instrument height — if elevations are required.

  They stay in effect until a new Setup is inserted. After a Setup line, first multiple control shots and then multiple side-shots can appear.
- For each *control* shot **Observation**, insert the following:
  - **To #** the control point number,
  - **Hor. Circle** the horizontal circle value,
  - **Vert. Circle** the vertical circle value (or *zenith angle*),
  - **Slope Dist.** the slope distance,
  - **Ht. Signal** the signal (*target*, *prism*, *reflector*) height — if elevations are required.
- For each *side* shot **Observation**, insert the following:
  - **To #** the target point number,
  - **Hor. Circle** the horizontal circle value,
  - **Vert. Circle** the vertical circle value (or *zenith angle*),
  - **Slope Dist.** the slope distance,
  - **Ht. Signal** the signal (*target*, *prism*, *reflector*, or *pogo*) height — if elevations are required,
  - **Code** a point code, if needed,
  - **Note** a point note, if needed,
  - **Name** a point name, if needed.
- For a remote elevation measurement (REM), insert it as an **Observation** but without a Slope Distance (or possibly also without a Horizontal Circle). See Remote Elevation Measurement in § Field Files for details.
- For the last control shot of each free station, be sure to use the **end resect** button so that an "end= resect" line precedes the side-shots.



#### Project, Instrument and Scales Data Entry

Keep these three types of data on separate lines:

- Insert any desired **Project** data (Job, Date, Time, Surveyor, Field book, or Pages) wherever convenient. Note, however, that such data are ignored during calculations.
- Insert any **Instrument** constants, whenever they differ from their previous values. These are (along with their default values)
  - **Instr. ID** () ignored during calculation,
  - **VC Corr.** (0.0) for adding to vertical circle values,
  - **Dist. Corr.** (0.0) for slope distance additive correction (e.g., prism constant),
  - **Dist. Factor** (1.0) for slope distance multiplicative correction (e.g., due to extreme temperature),

  and they stay in effect until new ones are inserted.
- Insert any **Scales**, whenever they differ from their previous values. These are
  - **Scale Factor** (1.0) for combined map projection and sea-level (or elevation) distortions — applies only to horizontal distances,
  - **Units Factor** (1.0) for height and distance unit conversion (e.g., feet to metres),

  and they stay in effect until new ones are inserted.

Be sure to use the proper correction. In most cases, the default values are appropriate.

## 3. Field Files

Copan can read field data in different industry formats as well as its own. A native Copan field file consists of numerous Setup and Observation data lines and may contain comment lines, as well as Project, Scale, or Instrument data lines, if required. Each data line consists of various tab-separated *attribute* = *value* pairs. See § Field Files for a description of different raw field file types and Copan's field file format.

## 4. Field FreeStation File Example

In this example, codes and notes have been omitted for brevity. The survey from setup 19 is a 3D freestation to three control points followed by two side-shots. The survey from setup 21 is a 2D freestation to three control points followed by two side-shots.

```
at= 19                                                          HI= 1.7
        to= 84   HC=  0.0000   VC= 89.0005   SD= 27.566
        to= 85   HC= 85.2920   VC= 88.3020   SD= 19.040   HS= 1.2
        to= 86   HC=139.0800   VC= 88.0105   SD= 24.950   HS= 1.2
        end= resect
        to= 91   HC=111.2155   VC= 91.5835   SD= 10.120   HS= 1.2
        to= 92   HC=123.4405   VC= 90.2705   SD= 15.950   HS= 1.2
at= 21
        to= 86   HC=  0.0000   VC= 84.5950   SD= 13.904
        to= 85   HC= 85.0720   VC= 86.4505   SD= 16.355
        to= 83   HC=137.1830   VC= 82.3930   SD= 19.641
        end= resect
        to= 95   HC=299.4800   VC= 91.0740   SD=  8.180
        to= 95   HC=302.5440   VC= 92.1310   SD= 11.833
```

## 5. FreeStation Calculation Method

The 2D portion of a freestation is calculated is as follows:

1. Treat each control point's horizontal circle reading as a temporary bearing and, with reduced distances, compute temporary local coordinates for each control point relative to the setup point.
2. Determine the best-fit rigid-body transformation, by least-squares, of the control points from their temporary to their actual coordinates and show the residuals.
3. Transform the setup point using the estimated parameters.

Assuming the appropriate elevations and heights are present, the 1D portion of a freestation is calculated is as follows:

1. Compute temporary local rises for each control point relative to the setup point.
2. Determine the average rise and show the residuals.
3. Transform the setup point using the estimated rise.

## 6. Notes

- *Classic* resection observations, involving angles only, are not supported.
- While repetition and redundancy are common in field surveying, Copan can only process reduced observations. Multiple sets of observations of the same quantities and cross-ties cannot be processed without some manual editing. For example, prior to processing, repeated sets of angles and distances must be averaged and check shots to the same points from different setups must be renumbered.
- If you need to manually insert a tab, such as when you wish to add a code or remark to an observation line, copy and paste an existing tab — there will always be one between two auto-inserted *attribute=value* pairs.
- Both negative and zero values for HI or HS are correctly interpreted.
- If a setup's HI value is missing (or null), no elevations are computed.
- If either a freestation control point's elevation or its HS value are missing, Copan ommits it from the elevation portion of the freestation estimation.
- If a side-shot's HS value is missing, its elevation will be null.

updated: 24-Mar-2011

updated: 24-Mar-2011

***Copan for Windows***

# Map Traverses

Contents

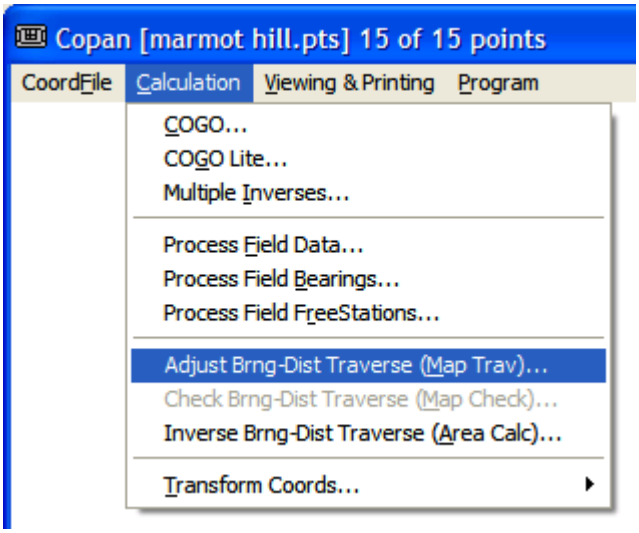


Azimuths (or bearings) combined with horizontal distances — pairs of values that are often known as *calls*, *courses*, or *metes* — are common on many cadastral maps and plans/plats. Copan refers to *sequences* of such "reduced" land survey data as *map traverse* data.

Use this module to compute and adjust for misclosure any number of map traverses and save the new points, whether or not the traverses are open, closed or adjusted. Use it to traverse around lot (or parcel) boundaries, including ones with curves, and to calculate the lot/parcel areas.

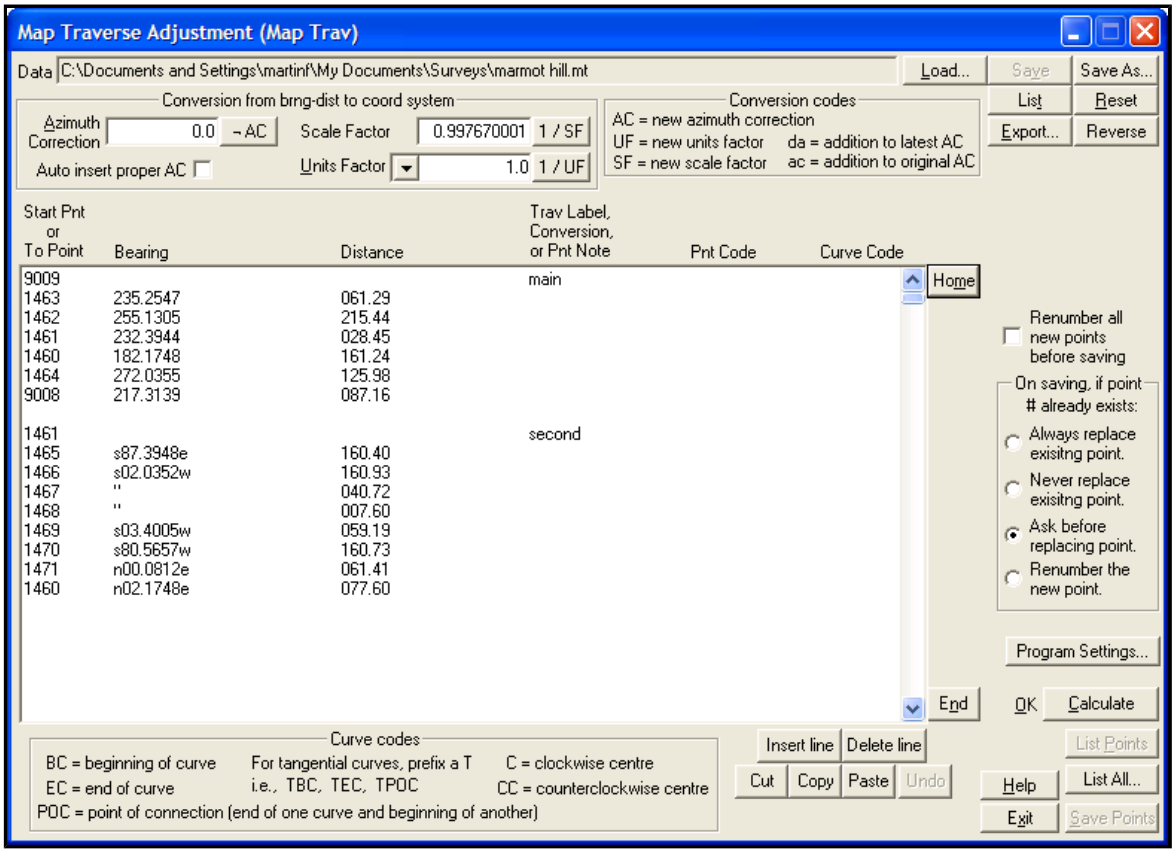
Note that if you are just *checking* the bearing and distance data from a map or deed, and not using a coordfile, you should use the § Map Check module. Also, this module cannot be used for "raw" *field* data, involving horizontal or vertical circle readings and slope distances. For such needs, use the § Field Data Processing module or its Field Bearings variation. Finally, while there are certain similarities between the Map Traverse and the Field Data modules, there are various operational differences (other than the type of map/survey data involved). If you are familiar with one and new to the other, please study the appropriate manual and dialog carefully.

## 1. To Adjust Bearing-Distance Traverses




1. **Calculate | Adjust Brng-Dist Traverse (Map Trav)...**
2. Optionally **Load...** a traverse **Data** file (see Map Traverse Files below).
3. Optionally enter conversions: an **Azimuth Correction**, to be added to every bearing; a **Units Factor**, for converting distance units to coordinate units; a **Scale Factor**, for combined map projection and sea-level (or elevation) factor conversion.
4. Enter or edit the bearing-distance traverse data in the big edit box. See below for a description of map traverse data for adjustment.
   To move the text cursor within the box, use the Arrow, Tab, or Enter keys, or the mouse pointer. Do not use the Space key to separate fields. To delete a chunk of text, select it with the mouse then click Cut (or type Ctrl-X) but be careful not to delete the embedded tabs within a line. To manually insert a tab, Copy and Paste an existing one. The Ctrl-Insert and Ctrl-Delete key combinations act like the Ins and Del buttons, that is, they insert and delete a line of data. To add a Point number automatically to the next blank line, press Enter when in the Distance column of the previous line.
5. Optionally **List** the map traverse **Data**.
6. Optionally **Export..** in ESRI format or **Reverse** the direction of the map traverse **Data**.
7. Optionally check the **Auto insert proper AC** box to have an azimuth correction automatically inserted at the start of the traverse such that a recalculation of the traverse would have no overall azimuth error.
8. **Save** the **Data** for reuse.
9. **Calculate** (or **OK**) the traverses, adjusting each one as desired (see § Traverse Processing).
10. To graphically view the traverses: Close or hide the Info Display window if it is open, and minimize or move aside — but do not close — the Map Traverse window.
11. Optionally choose whether to **Renumber new points** or **replace existing points** (see § Point Renumbering or Replacement) and **Save** the **Points** that have been computed.

12. Optionally **List** the **Points** that have been computed.




## 2. Map Traverse Data for Adjustment

- Begin each traverse by a *starter* line, which *must* contain the first point number, under **Start Pnt**, and *may* contain a traverse label, under **Trav Label**, identifying the traverse.
- The first point must already exist in the current coordfile or be in a prior traverse.
- A traverse must have a sequence of *leg* lines. Each leg
  - *must* contain these three items: **To Point** number, **Bearing** (or azimuth) and **Distance**,
  - and *may* contain a **Pnt Note**, **Pnt Code**, and a **Curve Code** (see § Traversing Curves on how to define curves along a map traverse).
- For a Bearing or Distance, an appropriate *expression* may be used (see § Bearing Expressions and § Distance and Offset Expressions), or, if it is the same as the previous one, a double-quote character (") may be used for *ditto*.
- If points are saved, Pnt Notes and Pnt Codes are added to the point (and Curve Codes are not added).
  NB: If a point appears in more than one leg and is saved, its later Note and Code may replace its earlier ones.
- A *closed* traverse, where the last point is known, may be adjusted or not. An *open* traverse (where the last point is unknown) is allowed.

- Treat radial *side shots* (or ties) as separate, single-leg, open traverses.
- At any point along a traverse you may need to change the scale factor, units factor or the azimuth correction. To do this, enter a separate line with
  - `SF=value`,
  - `UF=value`, or
  - `AC=value`

  under **Conversion**. From that point on the new value is used *instead* of the original *head* value. Optionally, a *differential* azimuth correction
  - `da=value`

  may be used, where the value is *added* to the *current* value for subsequent azimuth corrections. Or, a *relative* azimuth correction
  - `ac=value`

  may be used, where the value is *added* to the original *head* value for subsequent azimuth corrections. Note that these commands are case-sensitive.
- Blank lines may be used for readability but have no affect on traverse demarcation or calculation.

### 3. Map Traverse Files

A map traverse file, as it is plain text (or Ascii), can have any name, though names with the **.mt** extension is advisable. You can edit it outside of Copan, but be sure to maintain proper formatting:

- The first line *must* contain only this text:

  ```
  Map Traverses
  ```

- The next three lines can contain anything as they are not currently used.
- The next three lines *must* contain the "head" correction values — *scale-factor*, *units-factor*, and *azimuth-correction* — one each per line, respectively.
- Each subsequent line must be either a comment line or a tab-delimited data line with these six fields:

  *point-num*, *bearing*, *distance*, *trav-label*/*conversion*/*pnt-note*, *pnt-code*, and *curve-code*

  While some or all of those fields may be empty there *must* be *five* tabs per line. Which fields can be left blank depends on the purpose of the particular data line (see Map Traverse Data for Adjustment above).

- A valid comment line begins with zero or more spaces or tabs, then has two commas (,,), periods (..), slashes (//), semi-colons (;;), or backslashes (\\), which are followed by any character string. Comments are not processed or stored as point notes (but are saved within the traverse file). Use them to temporarily remove an observation or to remark on a traverse leg.
- Single-byte character encoding, such as **ANSI**, **DOS** or **UTF-8** must be used. *Note*: multi-byte character encodings (such as Unicode or UTF-16) will *not* work.

## 4. MT File Examples

Here are two map traverse files ready to be **Load**ed for calculation and adjustment.

**Example 1: Two simple traverses.**

This file has a primary traverse and a secondary traverse. Three of the second traverse's legs are in a straight line.

```
Map Traverses
1st dummy line
2nd dummy line
3rd dummy line
0.9997670000
1.0000000000
0.0000000000
9009                                primary
1463     235.2547     061.29
1462     255.1305     215.44
1461     232.3944     028.45
1460     182.1748     161.24
1464     272.0355     125.98
9008     217.3139     087.16
1461                                secondary
1465     092.2112     160.40
1466     182.0352     160.93
1467     "            040.72
1468     "            007.60
1469     183.4005     059.19
1470     260.5657     160.73
1471     000.0812     061.41
1460     002.1748     077.60
```

**Example 2: A traverse with curves.**

This traverse has a series of compound curve radials (see § Traversing Curves on how to define curves along a map traverse).

```
Map Traverses
Copan
V07152.pts
C:\Documents and Settings\martinf\My Documents\Whistler\13031.mt
1.0000000000
0.3048000000
0.0000000000
5746
                                    AC=-0.0436
5819     089.1140     166.64
5820     s11.5550w     34.06
5821     "            108.57                bc
5827     s78.0410e   3105.89                cc
5823     n79.1910w     "                    poc
5822     s79.1910e   2342.01                cc
5824     n89.4134w     "                    poc
5822     s89.4134e     "                    cc
5825     s89.1943w     "                    poc
```

```
5822     n89.1943e     "                    cc
5826     s86.3512w     "                    ec
5787     w           103.24
5788     s08.02w      46.4
```

## 5. Curve Adjustments

If a traverse that contains curves is adjusted, its curves are affected as follows:

- For non-tangential curves, the curve chords are treated as traverse legs for the adjustment, and then the curve centres are recomputed based on the original curve radii and new curve begin-end points.
- For tangential-curves, the external tangents are treated as traverse legs for the adjustment. The curve begin-end points are recomputed based on the new curve intersection points and external tangents, and then the curve centres are recomputed as above.

## 6. Export

Use the **Export...** button to export the current map traverses (starting from where you specify), in ESRI Traverse File format.

You specify the location and file name (the default being the current folder and current map traverse file, without the .mt) and Copan will save to *file-name**file-name*.txt (that's the funny way ArcGIS expects it).

Copan uses the latest *saved* coordinates for the start and end of each traverse, whether or not the traverse was adjusted. Also, it outputs traverse data in their combined *interpreted* and *converted* form (i.e., after expressions are evaluated *and* conversions are applied), but *not* in their *adjusted* form (i.e., before any traverse adjustments).

## 7. Reverse

Occasionally, you'd like to use an already entered traverse but in the reverse direction. Use the **Reverse** button to reverse the current map traverses (from last to first), each one in reverse point order.

Note that reversed traverse data are shown in their combined *interpreted* and *converted* form (i.e., after expressions are evaluated *and* conversions are applied). So if you want only the "raw" (or unconverted) data reversed, you must first temporarily remove the conversions.

## 8. Notes

- If you have multiple traverses, when you Calculate you may begin from a specific traverse rather than from the top by entering the start point number or traverse label.
- When you Calculate, Copan always checks and computes the traverse data currently in the edit boxes.
- Changes to the data in the edit boxes are only saved (to disk) when you Save Data.

- After you Calculate, Copan lists the *reduced* traverse data (i.e., legs are in their combined *interpreted* and *converted* form).
- If a traverse is a loop (i.e., it ends on its own start point) Copan provides the enclosed area in squared distance units. And if a units factor was involved, the area is also given in converted units. For example, if distances are entered in feet and a UF=0.3048 conversion is entered, then area is given in $m^2$ and in $ft^2$.
- Copan does not calculate areas that are only enclosed by separate different traverses, but it does still calculate all the other paramaters of any non-loop traverses.
- If Copan complains about the data when calculating, and you cannot see what is wrong, visually inspect the data by selecting it all with the mouse cursor. Often, you can see the inconsistency via the selection block outline.
- If you **List** traverse data to the Info Display file,
  - multiple blank lines are ignored, and
  - azimuth (or bearing) and distance expressions or dittos are shown in their *interpreted* form (i.e., expressions are evaluated), but *not* in their *converted* form (i.e., AC, SF, UF are not yet applied).
- **List All...** is equivalent to the sequence **List data**, **Calculate**, and **List Points**.
- Newly calculated points are not saved to the coordfile until you click Save Points.
- When you Save Points, Copan first creates an automatic backup copy of the coordfile.

updated: 24-Mar-2011

updated: 24-Mar-2011

***Copan for Windows***

# Map Checks

Contents

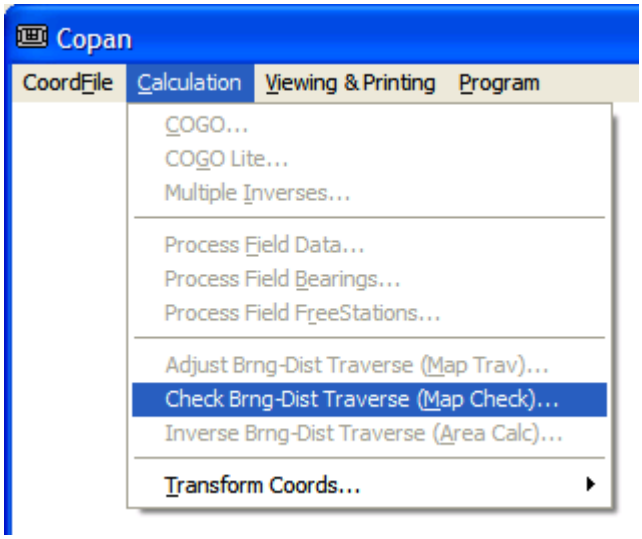


Property (lot or parcel) boundaries on many maps, plans/plats or deeds are commonly annotated with azimuths (or bearings), horizontal distances, curve parameters, and areas. Bearing and distance pairs are often known as *calls*, *courses*, or *metes*. Copan refers to sequences of such reduced data as *map traverse* data and the process of checking the consistency of such data as *map check*. (Elsewhere, it may be known as *closure check*, *lot check* or *deed check*.)

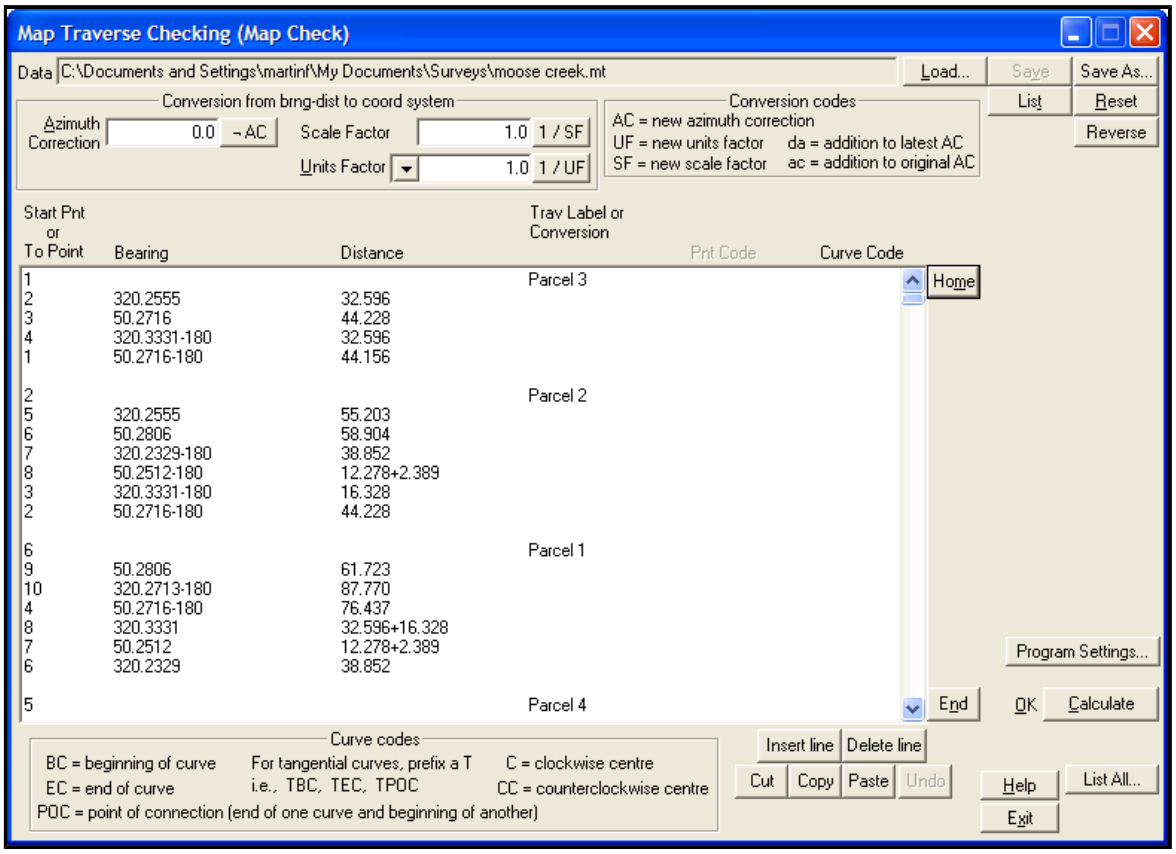
Use this module to check the closures and areas (where appropriate) of any number of map traverses, including ones with curves, independently of any coordinate file. In fact, ***Map Check is disabled if a coordfile is open***.

Note that if you need to adjust azimuth and distance traverse data or save new points, you should use the § Map Traverses module. Also, this module cannot be used for field data, involving horizontal or vertical circle readings or slope distances. For such needs, use the § Field Data Processing module. Finally, hile there are certain similarities between the Map Check and the Field Data modules, there are various operational differences (other than the type of map/survey data involved). If you are familiar with one and new to the other, please study the appropriate manual and dialog carefully.

## 1. To Check Bearing-Distance Traverses




1. If you have a coordfile open, close it.
2. **Calculate | Check Brng-Dist Traverse (Map Check)...**
3. Optionally **Load...** a traverse **Data** file. This may be an older-format loop file or a newer -format map traverse file (see Map Traverse Files below).
4. Optionally enter conversions: an **Azimuth Correction**, to be added to every bearing; a **Units Factor**, for converting distance units to coordinate units; a **Scale Factor**, for a combined map projection and sea-level (or elevation) factor.
5. Enter or edit the bearing-distance traverse data in the big edit box. See below for a description of map traverse data for checking.
   To move the text cursor within the big edit box, use the Arrow, Tab, or Enter keys, or the mouse pointer. Do not use the Space key to separate fields. To delete a chunk of text, select it with the mouse then click Cut (or type Ctrl-X) but be careful not to delete the embedded tabs within a line. To manually insert a tab, Copy and Paste an existing one. The Ctrl-Insert and Ctrl-Delete key combinations act like the Ins and Del buttons, that is, they insert and delete a line of data. To add a Point number automatically to the next blank line, press Enter when in the Distance column of the previous line.
6. Optionally **List** the map traverse **Data**.
7. **Save** the **Data** for reuse.
8. Optionally **Reverse** the direction of the map traverse **Data**.
9. **Calculate** (or **OK**) the traverses.
10. To graphically view the traverses: Close or hide the Info Display window if it is open, and minimize or move aside — but do not close — the Map Check window.




## 2. Map Traverse Data for Checking

- Begin each traverse by a *starter* line, which *must* contain the first point number, under **Start Pnt**, and *may* contain a label, under **Trav Label**, identifying the traverse. Since there is no use of coordfiles, any valid point number may do.
- A traverse must have a sequence of *leg* lines. Each leg
  - *must* contain these three items: **To Point** number, **Bearing** (or azimuth) and **Distance**,
  - and *may* contain a Pnt Note (under Trav Label), Pnt Code, and a **Curve Code**. However, Pnt Notes and Pnt Codes are ignored.
- For a Bearing or Distance, an appropriate *expression* may be used (see § Bearing Expressions and § Distance and Offset Expressions), except that, because no coordfile is open, no point numbers may be used. If a Bearing or Distance is the same as the previous one, a double-quote character (") may be used for *ditto*.
- At any point along a traverse you may need to change the scale or distance factors or the azimuth correction. To do this, enter a separate line with
  - `SF=value`,
  - `UF=value`, or
  - `AC=value`

  under **Conversion**. From that point on the new value is used *instead* of the original *head* value. Optionally, a *differential* azimuth correction
  - `da=value`

may be used, where the value is *added* to the *current* value for subsequent azimuth corrections. Or, a *relative* azimuth correction
    - `ac=value`

  may be used, where the value is *added* to the original *head* value for subsequent azimuth corrections. Note that these commands are case-sensitive.
- Blank lines may be used for readability but have no affect on traverse demarcation or calculation.

### 3. Map Traverse Files

A map traverse file, as it is plain text (or Ascii), can have any name, though names with the **.mt** extension is advisable. You can edit it outside of Copan, but be sure to maintain proper formatting:

- The first line must contain only this text:

  ```
  Map Traverses
  ```

- The next three lines can contain anything as they are not currently used.
- The next three lines *must* contain the "head" correction values — *scale-factor*, *units-factor*, and *azimuth-correction* — one each per line, respectively.
- Each subsequent line must be either a comment line or a tab-delimited data line with these six fields:

  *point-num*, *bearing*, *distance*, *trav-label*/*conversion*/*pnt-note*, *pnt-code*, and *curve-code*

  While some or all of those fields may be empty there *must* be *five* tabs per line. Which fields can be left blank depends on the purpose of the particular data line (see Map Traverse Data for Checking above).

- Single-byte character encoding, such as **ANSI**, **DOS** or **UTF-8** must be used. *Note*: multi-byte character encodings (such as Unicode or UTF-16) will *not* work.

#### Example Map traverse file.

Here's a map traverse file ready to be **Load**ed for checking.

```
Map Traverses
1st dummy line
2nd dummy line
3rd dummy line
1.0000000000
1.0000000000
0.0000000000
1                       Parcel B
2   092.2112    160.40
3   182.0352    160.93
4   "           048.32
5   183.4005    059.19
6   260.5657    160.73
```

```
7   000.0812    061.41
8   002.1748    077.60
1   002.1748    161.24
```

## 4. Reverse

Occasionally, you'd like to use an already entered traverse but in the reverse direction. Use the **Reverse** button to reverse the current map traverses (from last to first), each one in reverse point order.

Note that reversed traverse data are shown in their combined *interpreted* and *converted* form (i.e., after expressions are evaluated *and* conversions are applied). So if you want only the "raw" (or unconverted) data reversed, you must first temporarily remove the conversions.

## 5. Notes

- If you have multiple traverses, when you Calculate you may begin from a specific traverse rather than from the top by entering the start point number or traverse label.
- When you Calculate, Copan always checks and computes the traverse data currently in the edit boxes.
- Changes to the data in the edit boxes are only saved (to disk) when you Save Data.
- After you Calculate, Copan lists the *reduced* traverse data (i.e., legs are in their combined *interpreted* and *converted* form).
- If a traverse is a loop (i.e., it ends on its own start point) Copan provides the enclosed area in squared distance units. And if a units factor was involved, the area is also given in converted units. For example, if distances are entered in feet and a UF=0.3048 conversion is entered, then area is given in $m^2$ and in $ft^2$.
- Copan does not calculate areas that are only enclosed by separate different traverses, but it does still calculate all the other paramaters of any non-loop traverses.
- If Copan complains about the data when calculating, and you cannot see what is wrong, visually inspect the data by selecting it all with the mouse cursor. Often, you can see the inconsistency via the selection block outline.
- If you **List** traverse data to the Info Display file,
  - multiple blank lines are ignored, and
  - azimuth (or bearing) and distance expressions or dittos are shown in their *interpreted* form (i.e., expressions are evaluated), but *not* in their *converted* form (i.e., AC, SF, UF are not yet applied).
- **List All...** is equivalent to the sequence **List data** and **Calculate**.

updated: 24-Mar-2011

updated: 24-Mar-2011

***Copan for Windows***

# Area and Perimeter Calculations

Contents

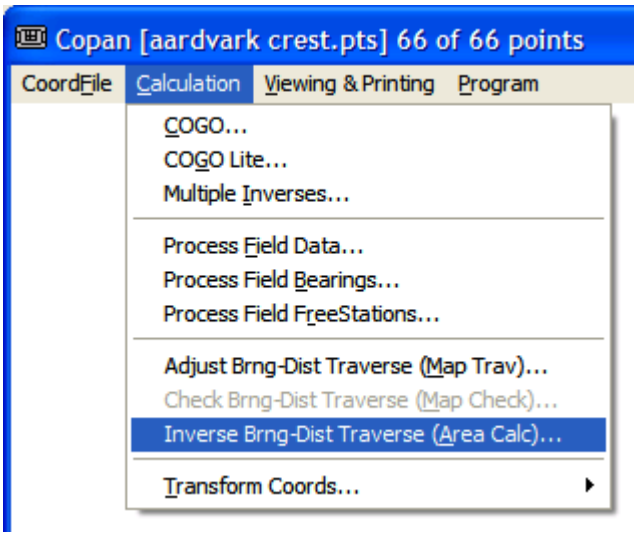


If you have a set of points in a coordinate file, any suitable sequence of such points can define a structural *alignment* or property *boundary*. An alignment or boundary is a sequence of straight line segments and circular curve segments, joined end to end.

Use this module to calculate the *geometry* of any number of such sequences, including ones with curves. The resulting geometry includes azimuths (or bearings) and horizontal distances of straight segments, parameters of curved segments, *perimeters* (or overall lengths) of lines, and *areas* of fully enclosed shapes (or loops).

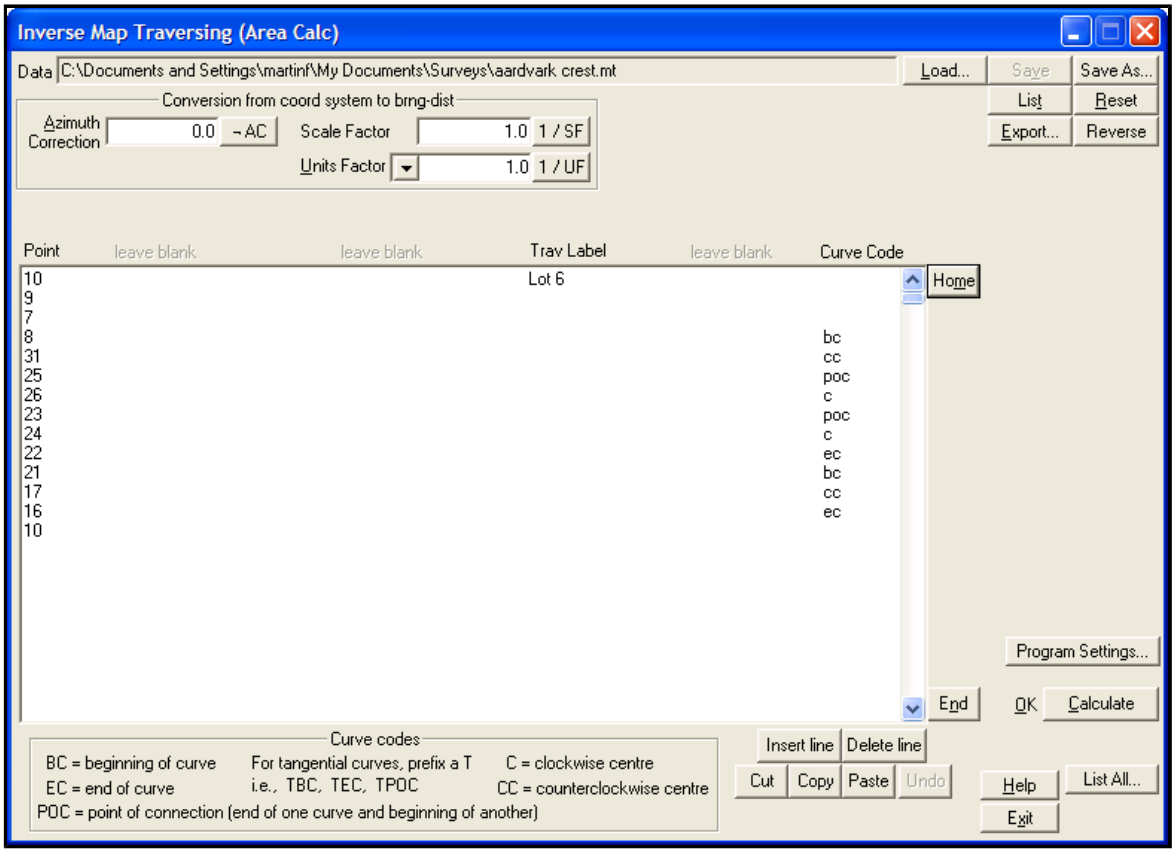
We call it “inverse map traversing” or “inverse bearing and distance traversing” because, like inverse calculations in the § COGO module, it determines bearings and distances between given points, and because the interface and reporting are virtually identical to those of the § Map Traverse and § Map Check modules (which you should use if you need to actually apply or check bearing and distance traverse data).

## 1. To Inverse Traverse Coordfile Points




1. **Calculate | Inverse Brng-Dist Traverse (Area Calc)...**
2. Optionally **Load...** a traverse **Data** file. (see Map Traverse Files below).

3. Optionally enter conversions: an **Azimuth Correction**, to be added to computed bearings; a **Units Factor**, for converting computed distances to other units; a **Scale Factor**, for combined map projection and sea-level conversion from computed distances to other coordinate system distances.
4. Enter or edit the point traverse data in the big edit box. See below for a description of map traverse data for calculating areas and perimeters.
   To move the text cursor within the big edit box, use the Arrow, Tab, or Enter keys, or the mouse pointer. Do not use the Space key to separate fields. To delete a chunk of text, select it with the mouse then click Cut (or type Ctrl-X) but be careful not to delete the embedded tabs within a line. To manually insert a tab, Copy and Paste an existing one. The Ctrl-Insert and Ctrl-Delete key combinations act like the Ins and Del buttons, that is, they insert and delete a line of data. To add a Point number automatically to the next blank line, press Enter when in the Distance column of the previous line.
5. Optionally **List** the map traverse **Data**.
6. **Save** the **Data** for reuse.
7. Optionally **Export..** in ESRI format or **Reverse** the direction of the map traverse **Data**.
8. **Calculate** (or **OK**) the traverses.
9. To graphically view the traverses: Close or hide the Info Display window if it is open, and minimize or move aside — but do not close — the Area Calc window.



## 2. Map Traverse Data for Calculating Areas and Perimeters

- Begin each traverse by a *starter* line, which *must* not only contain the first point number, under **Point**, but *also* contain a traverse label, under **Trav Label**, identifying the traverse.
- A traverse must have a sequence of *leg* lines. Each leg must contain a **Point** number and may contain a **Curve Code** (see § Traversing Curves on how to define curves along a map traverse). Nothing else is allowed.
- All identified points *must* already exist in the coordfile.
- Blank lines may be used for readability but have no affect on or calculation.

## 3. Map Traverse Files

A map traverse file, as it is plain text (or Ascii), can have any name, though names with the **.mt** extension is advisable. You can edit it outside of Copan, but be sure to maintain proper formatting:

- The first line must contain only this text:
  ```
  Map Traverses
  ```
- The next three lines can contain anything as they are not currently used.
- The next three lines *must* contain the "head" correction values — *scale-factor*, *units-factor*, and *azimuth-correction* — one each per line, respectively.
- Each subsequent line must be a tab-delimited data line with six fields:

  *point-num*, *blank*, *blank*, *trav-label*, *blank*, and *curve-code*

  Most of those fields are empty but there *must* be five tabs per line. Which fields are left blank depends on the purpose of the particular data line (see Map Traverse Data for Calculating Areas and Perimeters above).

- Single-byte character encoding, such as **ANSI**, **DOS** or **UTF-8** must be used. *Note*: multi-byte character encodings (such as Unicode or UTF-16) will *not* work.

#### Example Map traverse file.

Here's a map traverse file ready to be **Load**ed for checking.

```
Map Traverses
1st dummy line
2nd dummy line
3rd dummy line
1.0000000000
1.0000000000
0.0000000000
101     parcel D
102
103             tbc
104             c
105             tec
```

```
106
101
```

## 4. Export

Use the **Export...** button to export the current map traverses (starting from where you specify), in ESRI Traverse File format.

You specify the location and file name (the default being the current folder and current map traverse file, without the .mt) and Copan will save to *file-name**file-name*.txt (that's the funny way ArcGIS expects it).

Copan uses the latest coordinates for the start and end of each traverse, and outputs traverse data in their *converted* form (i.e., after conversions are applied).

## 5. Reverse

Occasionally, you'd like to use an already entered traverse but in the reverse direction. Use the **Reverse** button to reverse the current map traverses (from last to first), each one in reverse point order.

## 6. Notes

- If you have multiple traverses, when you Calculate you may begin from a specific traverse rather than from the top by entering the start point number or traverse label.
- If a traverse is a loop (i.e., it ends on its own start point) Copan provides the enclosed area in squared distance units. And if a units factor was involved, the area is also given in converted units. For example, if distances are entered in feet and a UF=0.3048 conversion is entered, then area is given in $m^2$ and in $ft^2$.
- Copan does not calculate areas that are only enclosed by separate different traverses, but it does still calculate all the other paramaters of any non-loop traverses.
- If Copan complains about the data when calculating, and you cannot see what is wrong, visually inspect the data by selecting it all with the mouse. Often, you can see the inconsistency via the selection block outline.
- **List All...** is equivalent to the sequence **List data** and **Calculate**.

updated: 24-Mar-2011

updated: 24-Mar-2011

***Copan for Windows***

# Coordinate Transformations

Contents



You will often need to geometrically transform the coordinates of a group of points, for example, between a local ground coordinate system and a regional grid system. The need may be for a permanent change to a coordfile or for just a temporary transformation during an import/export process.

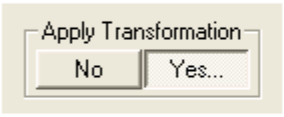
Use this module to geometrically transform a set of points, either

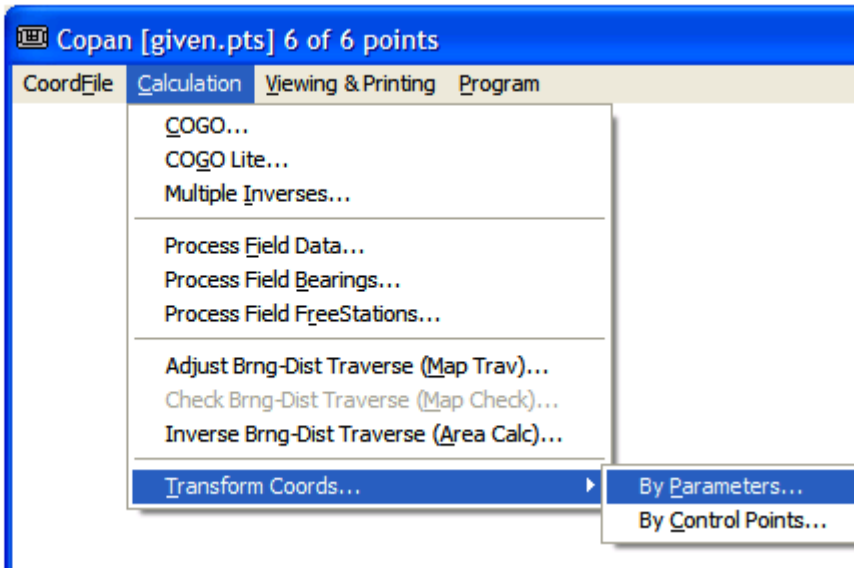
- **directly** by specifying the *known parameters* of an affine transformation, or
- **indirectly** by specifying pairs of *control points* for a similarity transformation from one coordinate system to another.

In each case, rotations are always about a vertical axis, i.e., the horizontal planes of each coordinate system are assumed to be parallel.

## 1. Transforming Coordinates by Parameters

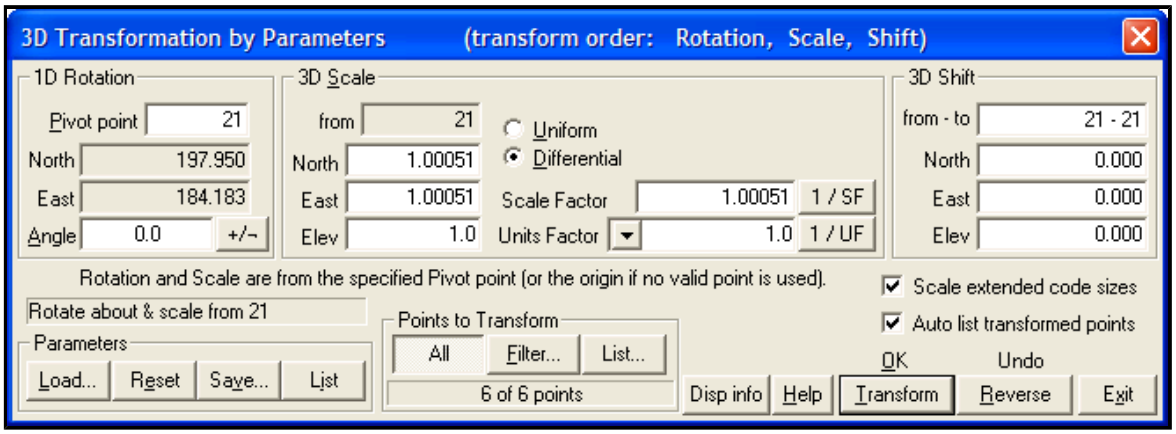
You can specify up to "seven plus two" parameters of an *affine* transformation. The seven affine transformation parameters are a 1D rotation, a 3D scaling, and a 3D shift (slide or translation). Of course, any of the parameters may be left in their trivial form (i.e., rotation = 0, scale = 1, shift = 0). The additional two parameters are the 2D coordinates of a "pivot point" that is temporarily fixed with respect to rotation and scale.




## To Transform by Parameters

1. Either **Calculation | Transform Coords... | By Parameters...** to change a coordfile's point coordinates,
   or choose **Yes...** under **Apply Transformation** from within an import/export module to change the coordinates during the import/export process and leave the source points unchanged.
2. Optionally **Load...** a transformation **Parameters** file.
3. Enter the **Pivot point** number or the **Northing** and **Easting** of a location about which the rotation should pivot or from which the scaling should occur.
4. For a **Rotation**, enter the **Angle** (+ve clockwise).
5. For a **Scale** change,
   - choose **Uniform** or **Differential** scaling, and
   - enter the **Scale Factor** and choose the **Units Factor** (optionally inverting them via **1/SF** and **1/UF**) as appropriate, or enter the scale values directly.
6. Optionally **Scale extended-code sizes**. This will scale any point codes that are sized according to the rules in § Point Codes by the Elev Scale present. (The Z-scale is used, rather than the N/E-scale, to prevent influence from the SF.)
7. For a **Shift**, enter either a shift expression (see § Shift Expressions) or the **Northing**, **Easting** and **Elevation** values to be added.
8. Optionally **Save** the **Transformation** parameters file for later reuse.
9. Choose which **Points to transform**: **All** of them or **Filter...** a subset (See § Point Filters).
10. **Transform** or **OK**. The sequence will be rotation, scale, then shift.
11. Optionally **Reverse** transform the points. This performs the specified transformation in reverse (so effectively acts as an Undo button).




### Notes

- A transformation **Parameters** file is a binary file, and while it can have any name, it ought to have a **.tfm** name extension.
- Scale Factor represents a combined map projection and sea-level conversion factor, and applies only to horizontal distances (i.e., North and East dimensions).
- Units Factor is for height and distance unit conversion (e.g., feet to metres) so applies to all three coordinate dimensions, and to extended-code sizes (if requested).

## 2. Transforming Coordinates by Control Points

Suppose you have a set of points to transform from one coordinate system, the *source* system, to another, the *target* system. You can estimate a best-fit transformation between specified lists of control points — points whose coordinates are available in both systems — then do the transformation.

The estimation is by least-squares and can be for

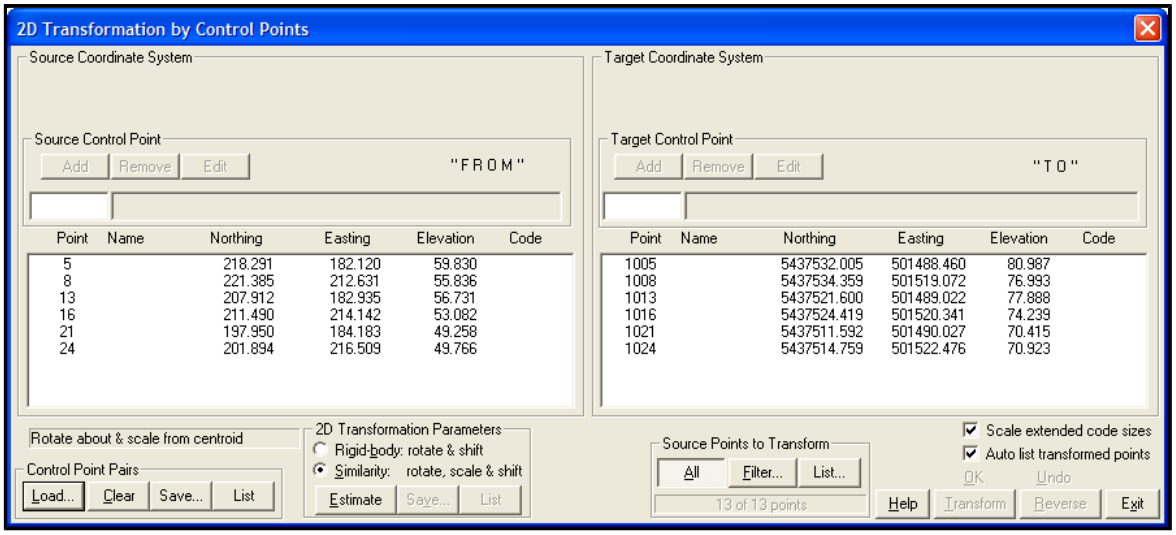
- a 3-parameter *rigid-body transformation*, having a 1D rotation and a 2D shift or
- a 4-parameter *similarity transformation* (or Helmert transformation), having a 1D rotation, a 1D scale, and a 2D shift.

The rotation and scale (if present) are with respect to the arbitrary 2D centroid of the source control points. The least-squares solution is one that minimizes the sum of the squared residuals, where a residual is the difference between a tansformed source point and its corresponding target point.

### To Transform by Control Points

1. **Calculation | Transform Coords... | By Control Points...**
2. Optionally **Load...** a **Control Point Pairs** file.
3. For each **Source Control Point**, enter the # and **Add**.
4. Do the same for each **Target Control Point**, ensuring that corresponding points are in the same order.

5. Select what**Type** of transformation: **Rigid**-body or **Similar**ity.
6. **Estimate** the **Parameters** and examine the listed residuals.
7. Optionally **Save...** the Control Point Pairs for later reuse.
8. Optionally **Save...** the **Parameters** for later use in Transform by Parameters.
9. Choose which **Source points to transform**: **All** of them or **Filter...** a subset (See § Point Filters). Target points are automatically protected so will not get transformed.
10. **Transform** or **OK**.
11. Optionally **Reverse** transform the points. This performs the specified transformation in reverse (so effectively acts as an Undo button).




## The Transformation Control Point Pairs File Format

This plain text (or ascii) file has

- Any name, though names with a **.tcp** extension is advisable.
- One heading line containing “Source” and “Target” (exclude quotes and separate by whitespace).
- Pairs of corresponding source and target point numbers (separated by whitespace), one per line.
- Single-byte character encoding, such as **ANSI**, **DOS** or **UTF-8**. *Note*: multi-byte character encodings (such as Unicode or UTF-16) will *not* work.

**Example: Transformation control point pairs file.**

```
Source  Target
2       1002
3       1003
4       1004
```

## Notes

- The target control points must be in the same coordfile.
- Any point that gets added to the target point list is automatically protected (prevented from being transformed) and remains so, even if subsequently removed from the list.
- If only two pairs of control point are used to estimate a similarity transformation, an exact solution (not an estimate) is determined.
- You can **Remove** items from the control point lists.
- Transformed points are plotted in their new positions.

updated: 24-Mar-2011

**Copan *Lite* for Windows**

# Part IV: Other Functions

- Viewing or Printing Graphics
- Program Settings
- File Locations
- Info Display File
- Point Codes
- Point Filters
- Field Files
- Traverse Processing
- Export to IOB File
- Bearing Expressions
- Distance and Offset Expressions
- Shift Expressions
- Traversing Curves
- Point Renumbering or Replacement

updated: 30-May-2011

updated: 21-Sep-2010

*Copan for Windows*

# Viewing and Printing Graphics

Contents



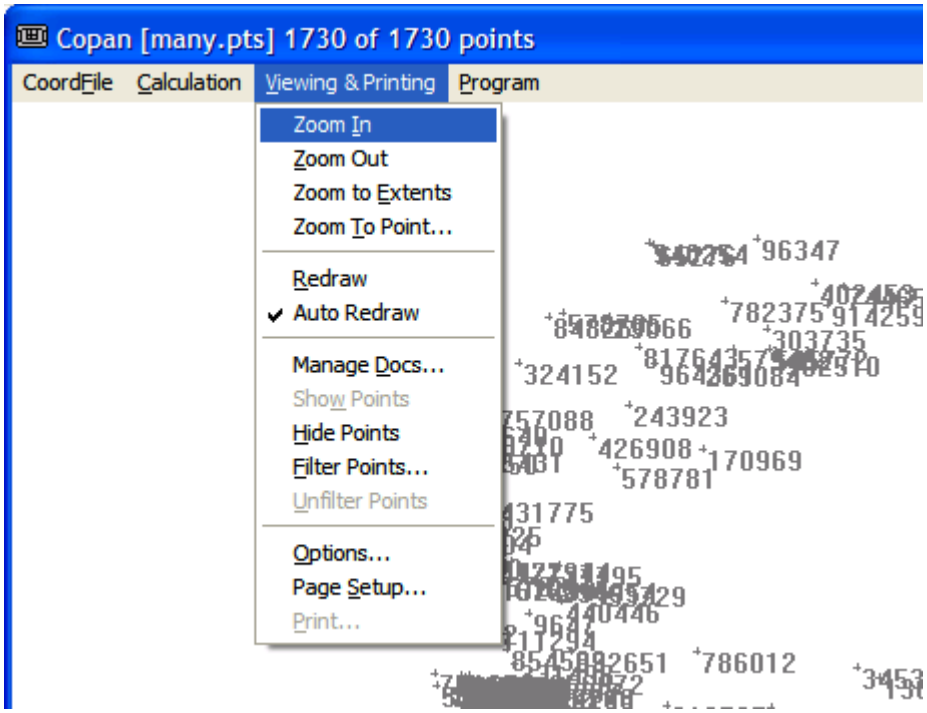
Whenever a coordfile is open in Copan, it is displayed in the main graphic window. You can view — or print out — any portion of this file at almost any scale.

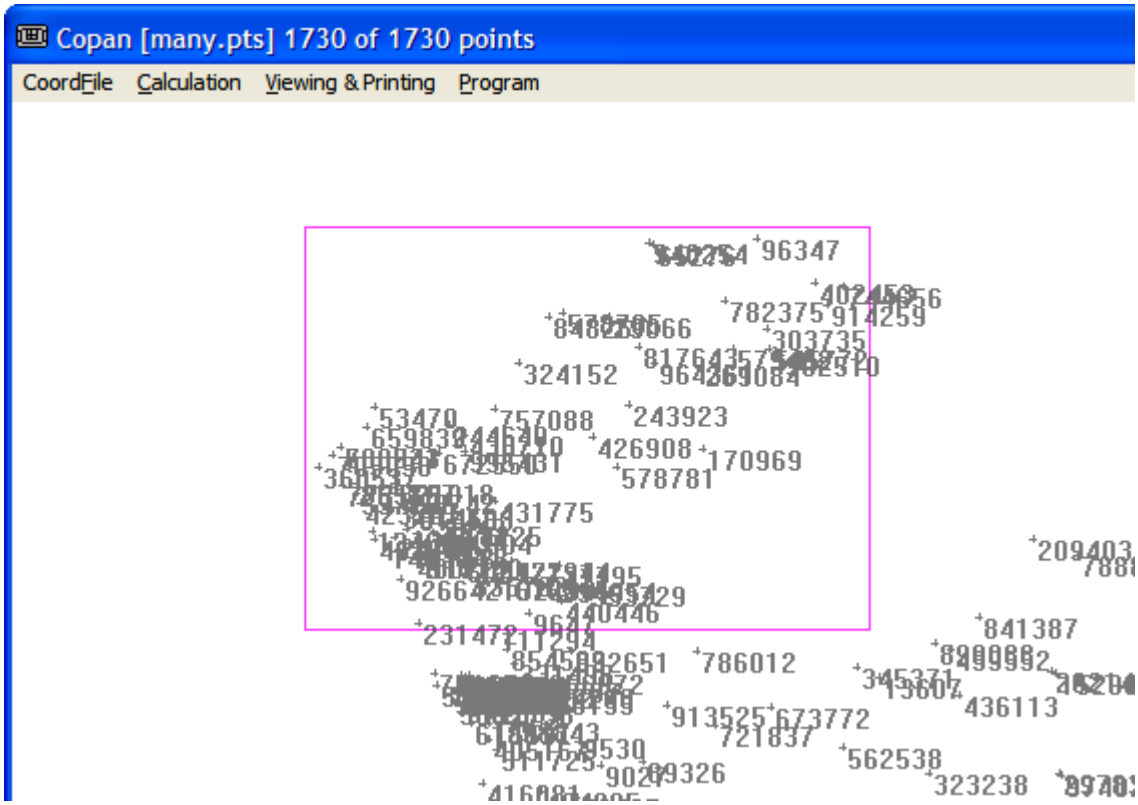
## 1. To Pan Around a Graphic Display

- If you have a *mouse wheel*, hold it down and drag it around to pan around.
- Alternatively, use the *scroll bars* or *scroll arrows* to pan around.


## 2. To Zoom Into and Out of a Graphic Display

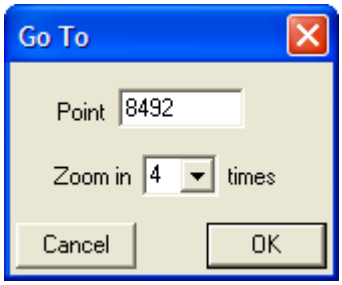
- If you have a *mouse wheel*, spin it to zoom in or out, or double-press it to zoom to the full spatial extent of the file.
- Alternatively, use the **Zoom In**, **Zoom Out** and **Zoom to Extents** items under the **Viewing & Printing** menu for their associated tasks.
  Note that Zoom In requires you to drag a rectangle over your intended target area.


Copan [many.pts] 1730 of 1730 points
CoordFile
Calculation
Viewing & Printing
Program
Zoom In
Zoom Out
Zoom to Extents
Zoom To Point...
Redraw
Auto Redraw
Manage Docs...
Show Points
Hide Points
Filter Points...
Unfilter Points
Options...
Page Setup...
Print...


Copan [many.pts] 1730 of 1730 points
CoordFile
Calculation
Viewing & Printing
Program

### 3. To Pan and Zoom Into a Specific Point

1. **Viewing & Printing | Zoom To Point...**
2. Enter the **Point** number, or the coordinates of the desired location.
3. Optionally, choose how many times you wish to **Zoom in**.
4. **OK**.




The display will then be centered on, and zoomed into, the given point.

### 4. Filtering and Hiding

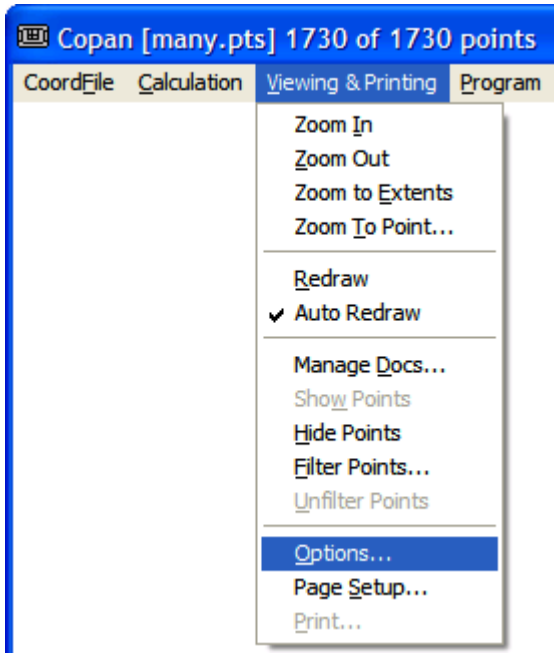
To view a specific subset of points, **Viewing & Printing | Filter Points...** (see § Point Filters). And to view all the points, even when a filter is set for some other purpose, **Viewing & Printing | Unfilter Points**.

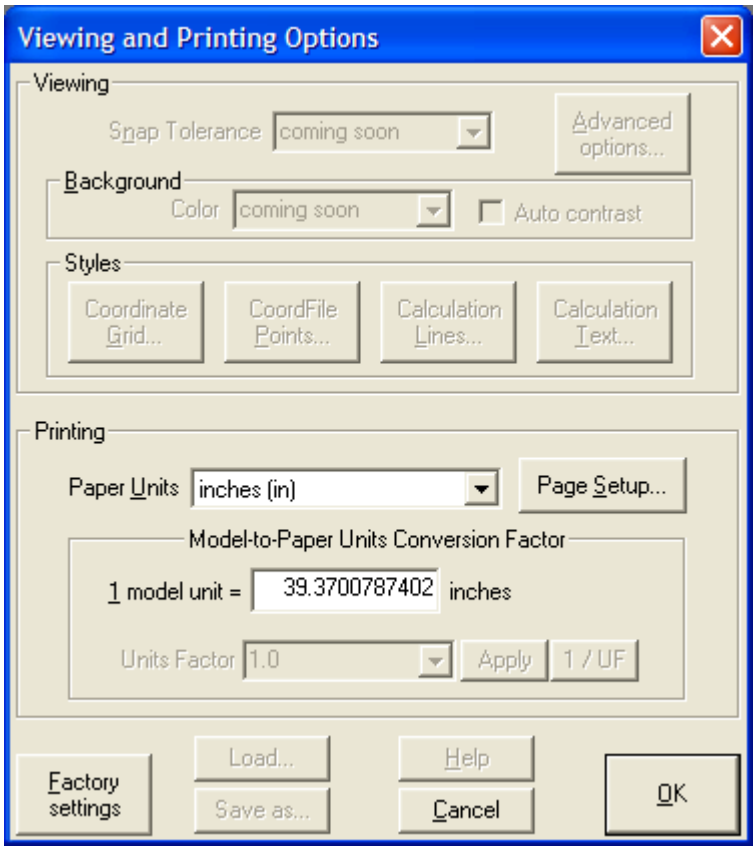
To always view all points when a filter is set for another purpose, **Program | Settings... | General...** (see § Program Settings) and check **Ignore filter when viewing points** under **Logistics**.

To turn off all points temporarily (perhaps when they are obscuring some linework), **Viewing & Printing | Hide Points**, and to turn them on again, **Show Points**.

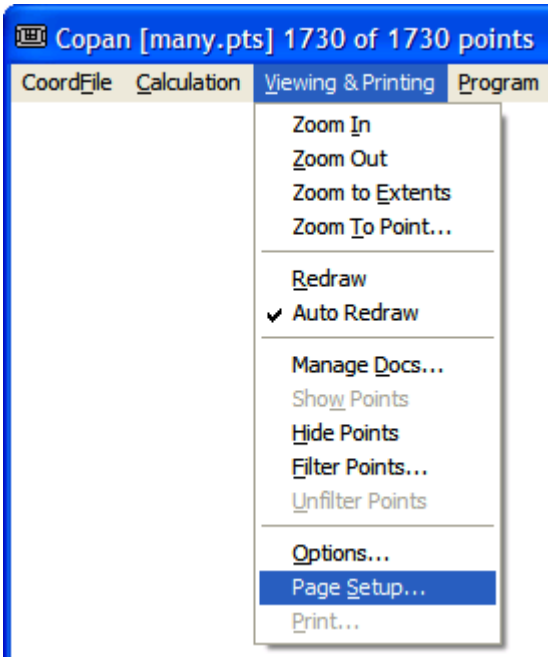
### 5. To Print a Graphic Display

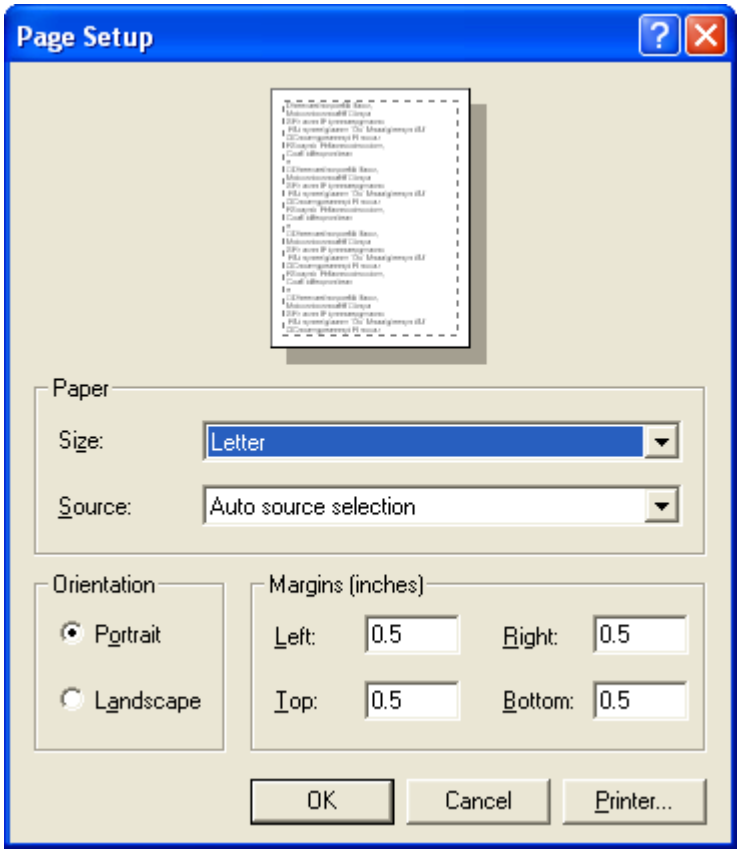
Assuming you have the desired graphics within view, and assuming your computer has access to a printer...


Copan [many.pts] 1730 of 1730 points
CoordFile
Calculation
Viewing & Printing
Program
Zoom In
Zoom Out
Zoom to Extents
Zoom To Point...
Redraw
Auto Redraw
Manage Docs...
Show Points
Hide Points
Filter Points...
Unfilter Points
Options...
Page Setup...
Print...

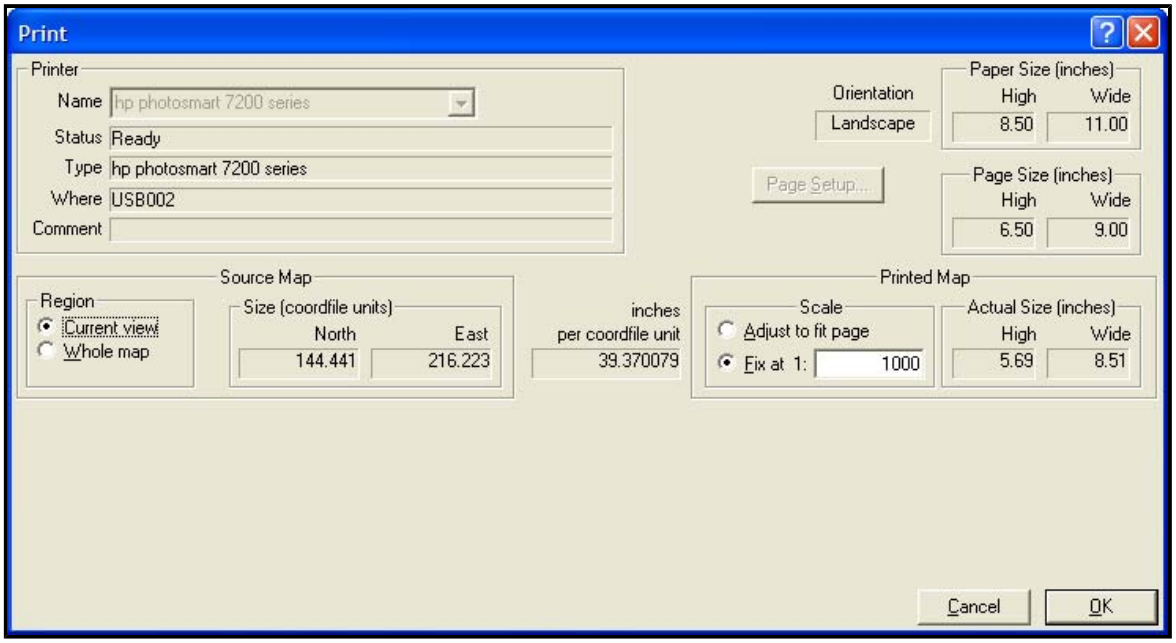

Viewing and Printing Options
Viewing
Snap Tolerance
coming soon
Advanced options...
Background
Color
coming soon
Auto contrast
Styles
Coordinate Grid...
CoordFile Points...
Calculation Lines...
Calculation Text...
Printing
Paper Units
inches (in)
Page Setup...
Model-to-Paper Units Conversion Factor
1 model unit =
39.3700787402
inches
Units Factor
1.0
Apply
1 / UF
Factory settings
Load...
Save as...
Help
Cancel
OK

1. Choose **Viewing & Printing | Options...** to open a Viewing and Printing dialog.
2. Select your preferred **Paper Units** — millimetres, centimetres, or inches — and enter the appropriate **Model-to-Paper Units Conversion Factor**. This pair of choices should ensure that graphics are printed true to any specified scale. For example:
   - if coordfile units are *metres* and paper units are *centimetres*, the conversion factor should be 100 (1m = 100cm)
   - if coordfile units are *feet* and paper units are *inches*, the conversion factor should be 12 (1ft = 12in)
   - if coordfile units are *metres* and paper units are *inches*, the conversion factor should be 39.37008 (1m = 39.37008in), the Factory default
3. Click **OK**.
   Unless your coordfile or paper units change, you'll never need to repeat that part of the process again.






4. Choose **Viewing & Printing | Page Setup...** to open a standard Page Setup dialog.
5. Select various **Paper**, **Orientation**, **Margins**, and **Printer** options.
6. Click **OK**.
   Unless your Page Setup choices change, you only need to repeat that part of the process once each program session.




7. Choose **Viewing & Printing | Print...** to open a special Print dialog.
8. Select which **Region** of the **Source Map** to print — just the **Current View** or the **Whole Map** — and select the **Scale** of the **Printed Map** — **Adjust to fit page** or **Fix at 1:** whatever scale you enter.
9. Click **OK**.

## 6. Notes

- The first time you wish to print any graphics after starting Copan each time, you must choose Page Setup... and click OK before you actually choose Print... — even if you don't change anything.
- You can obtain the best fit (i.e., the largest map for a given page) if you manually create a match between the shape of the main Copan program window and the orientation of the page in Page Setup — both landscape-shaped or both portrait-shaped.

updated: 21-Sep-2010

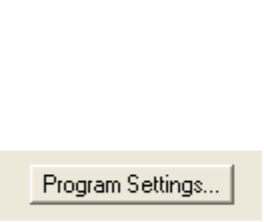
updated: 21-Sep-2010

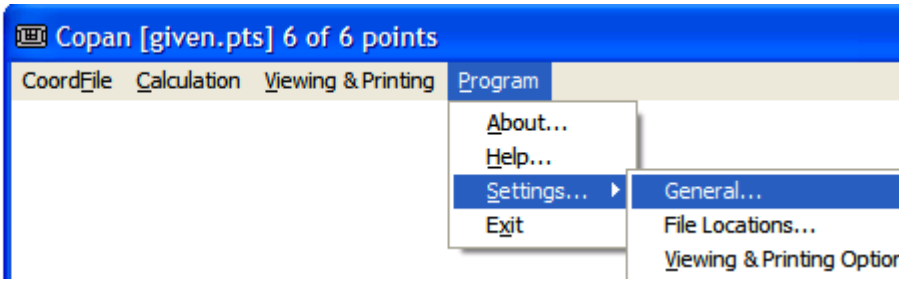
*Copan for Windows*

# Program Settings

There are various general settings which control program behavior. All of these are optional choices you can make.

## To Change General Program Settings




1. Either **Program | Settings... | General...**
   Or click the **Program Settings...** or **Settings...** button from within any module that has one.

### Misc. Settings

2. Enter up to three lines of a **Company** name and enter a **Surveyor** (or user) name.
3. Check **Include in each listing** for the Company or Surveyor name to be listed at the head of the Info Display File each time it is created.

### Logistic Settings

4. Check **Auto create project folder for new coordfile** to have a folder created for each new § Coordinate File created.
5. Check **Always initialize filter to All Points** to have all points selected each time you close a dialog (as opposed to having the current § Filter remain in effect).
6. Check **Auto copy auxiliary files to new project folder** to have the contents of the Auxiliary Folder copied to the Project Folder, automatically, each time a new coordfile is created.
7. Check **Ignore filter when viewing points** to always see all points in the display window, even when a § Filter is in effect.

8. Check **Auto export coordfiles to Ascii on closing** to have a coordinate file exported to a text file, automatically, each time it is closed, and if so, choose its **format**, **decimals** precision and its file name **ext**ension.
9. Choose the **Tool** for displaying § **Info Display** Files:
   - **Notepad**, or
   - **Other**, then **Browse...** to the location of your preferred text editor's executable file.

#### Geomatic Settings

10. Choose the number of **Distance decimals to list/display** to set the precision of listed or displayed distances and coordinates, and likewise for **Area decimals**.
11. Choose the **Angle Units** to use:
    - **Degrees Minutes Seconds** (inlcuding whether to **Show seconds**), or
    - **Gons (Grads)**,

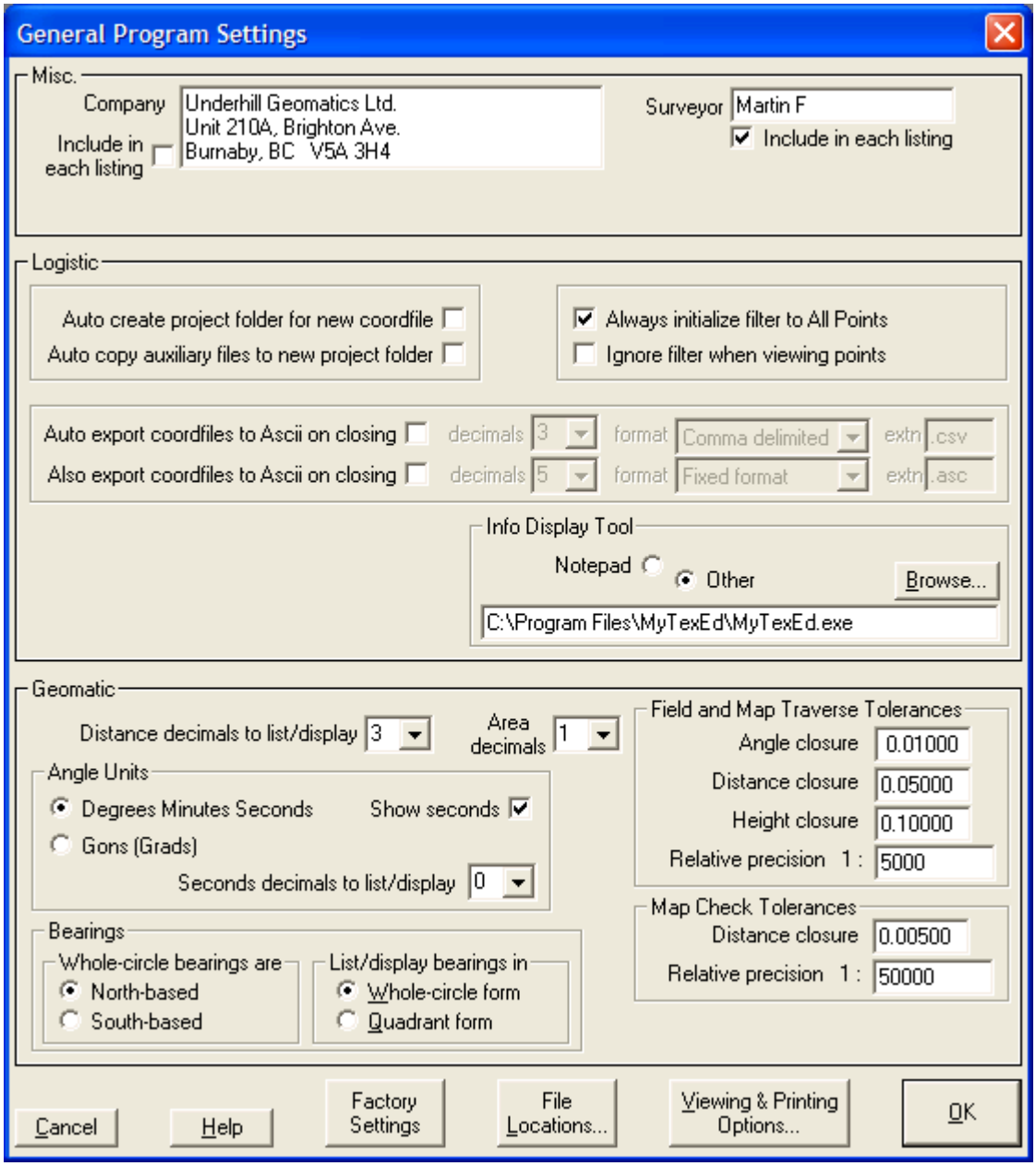
    and the **Seconds** or **Angle decimals to list/display**.
12. Choose whether whole-circle **Bearings** are:
    - **North-based**, or
    - **South-based**.
13. Choose whether **Bearings** should be listed/displayed in **Quadrant** form. Even if you choose the default whole-circle form, you may still use the quadrant form for input (see § Bearing Expressions).
14. Choose the various **Tolerances** for **Field and Map Traverses** and **Map Checks**. (See § Traverse Processing).

#### Other Settings

15. Change other settings, via buttons at bottom, as desired: **File Locations...** (see § File Locations) and **Viewing & Printing Options...** (see § Viewing or Printing Graphics).

#### Apply

16. Click **OK**.




For your information, those and other user settings are stored in the User Settings Folder, which you usually needn't concern yourself with, except possibly when copying settings to or from another user or computer.

updated: 21-Sep-2010

updated: 21-Sep-2010

***Copan for Windows***

# File Locations

The whole Copan package includes a number of different files, some of which are essential and others are supplemental to using Copan. This is an overview of the different kinds of files and their locations.

## Project Files

Each survey project you work on may require a number of different project data files. These include coordinate files, survey drawings, field data, azimuth-distance traverse data, transformations, etc.

All files related to a single job or project should normally be kept in a *project folder*, and each project folder would normally go in your *Initial Project Folder*. By default, your Initial Project Folder is

- `C:\Users\`*`user-name`*`\Documents\Copan\` (Win Vista / Win 7),
- `C:\Documents and Settings\`*`user-name`*`\My Documents\Copan\` (Win XP), or
- `C:\My Documents\Copan\` (Win 98),

but it can be changed to any location you like. (Prior to version 10.08, it was the same as the above but without the “Copan” folder.)

Copan keeps track of what is known as the *current* project folder. When you're about to open or create a project data file, you are shown the current project folder first, and if you choose another location, the new location becomes the current project folder.

## Coordinate Files

Virtually everything in Copan involves a coordinate file, or *coordfile* for short. Even if you'd like only to process coordinates from another program, you must first create a new coordfile and then import them.

A coordfile is a standalone file containing a *head* record and a set of survey *point* records. It is a binary file and has a **.pts** name extension.

See § Coordinate Files on how to do basic coordinate file management and § Point Records on the nature and management of point records.

## Info Display File

Most Copan activities and results are logged to the `info.display.txt` file within the current project folder. The file is overwritten (or emptied) before each activity, so you may wish to Save As or print it regularly from whichever program is displaying it. See § Info

Display File on printing or viewing the file. (Prior to version 10.05, there was only a single info display file, stored in the user settings folder.)

### Auxiliary Files

Some data, that may or may not be specific to a survey project, are used for reference. They include
a *point codes table* (e.g., `codes.txt`), required when using code descriptions or extensions (see § Point Codes). Copan looks for them in your *Auxiliary Files Folder* which starts out being within the Initial Project Folder but it can be changed to, say, a common location on a file server. (Prior to version 10.08, auxiliary files were located in the program folder.)

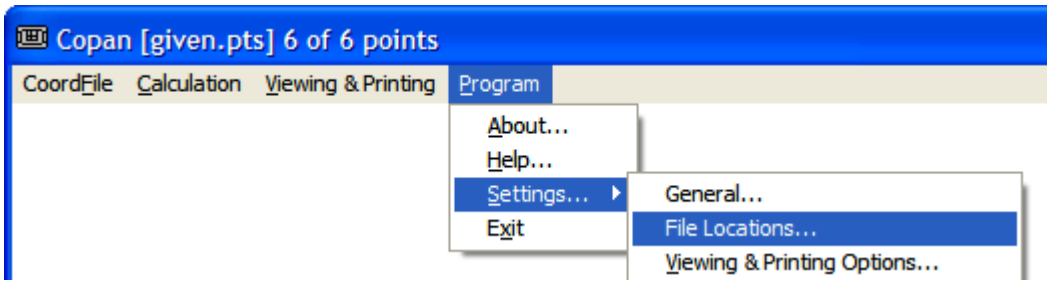
You or your colleagues are responsible for creating your auxiliary files.

Note that you have the option to have separate auxiliary files associated with each project. You might choose to do this if your point codes tend to change from project to project. To do it for particular projects only, check **Copy auxiliary files to project folder** when you create a coordinate file. To do it for all new projects, check **Auto copy auxiliary files to new project folder** in § Program Settings. In either case, Copan will copy the contents of *the* Auxiliary Files Folder to the particular project folder.

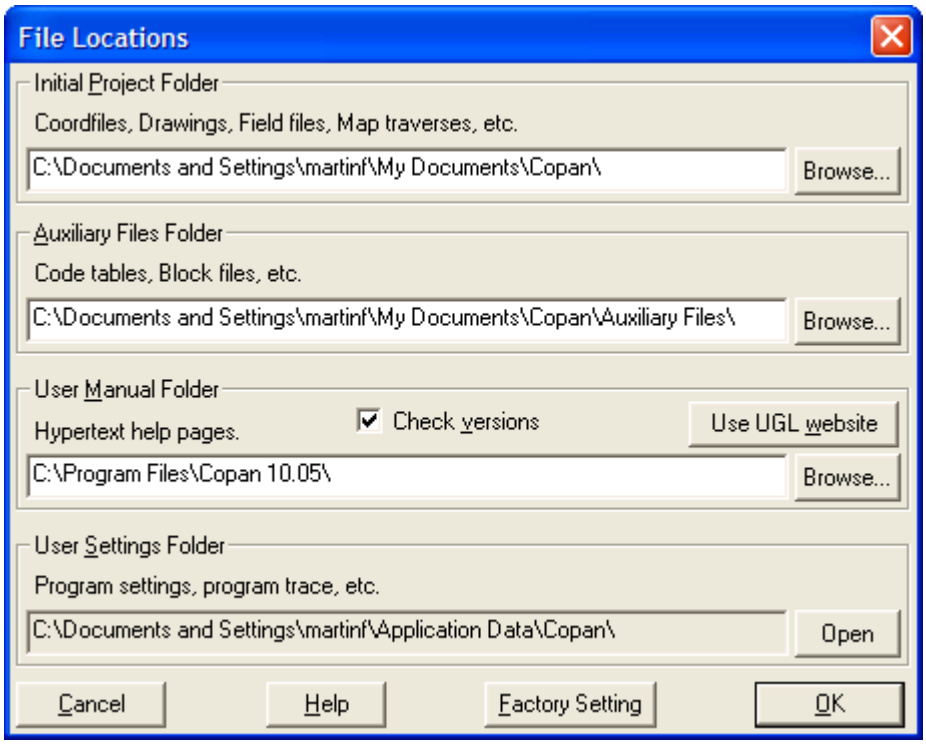
### User Manual Files

Copan looks for hypertext help documents (including this one) in your *User Manual Folder*. This starts out being located inside the Program Folder but can be changed to another location if your user manual is stored there.

### To Change Default File Locations




1. **Program | Settings... | File Locations...**
2. **Browse...** to the desired **Initial Project Folder** location.
3. **Browse...** to the desired **Auxiliary Files Folder** location.
4. **Browse...** to the desired **User Manual Folder** location, or click **Use UGL website** to always use latest user manual on the Underhill web site.
5. Click **OK**.




For your information, there are two other sets of files — User Settings and Program Files — whose locations you cannot change and with which you needn't concern yourself (usually).

### User Settings Files

Most of your preferences, settings, options, etc, for Copan are stored in your *User Settings Folder*. Occaisionally you may like to copy your settings between users or computers. They are here:

- `C:\Users\`*`user-name`*`\AppData\Roaming\Copan\` (Win Vista / Win 7),
- `C:\Documents and Settings\`*`user-name`*`\Application Data\Copan\` (Win XP), or
- `C:\WINDOWS\Application Data\Copan\` (Win 98),

### Program Files

When you install Copan, various files get stored in your *Program Folder*. Usually they are here

- `C:\Program Files (x86)\Copan `*`version-number`*`\` (64-bit Win) or
- `C:\Program Files\Copan `*`version-number`*`\` (32-bit Win).

but you may install them somewhere else. (You have the choice during program installation.)

updated: 21-Sep-2010

updated: 21-Sep-2010

***Copan for Windows***

# Info Display File

Almost all Copan results are written to an `info.display.txt` file in the current project folder and are thus displayed by Copan using your current text editor. To ensure proper alignment of data columns, use a fixed-width (fixed-pitch or monospaced) font setting, such as Courier New, within your text editor. (If you use a variable-width font, output won't quite be arranged as it should.)

## Printing info.display.txt from a text editor

Copan typically outputs lines of up to 100 characters in length to the info.display.txt file. The best settings to use — to get all the data printed across the page — depend on the paper size that your printer is using: North American Letter or International A4.

### Smaller Font, Portrait Orientation

The following are Win Notepad terms and may be slightly different in another editor:

**Format | Font...**

| | |
|---|---|
| Font: | Courier New |
| Style: | Regular |
| Size: | **9** |

**File | Page Setup...**

| | | |
|---|---|---|
| Paper Size: | **Letter** (N. America) | **A4** (International) |
| Left and Right Margins: | **0.45** inch (or **11** mm) | **8** mm (or **0.3** inch) |
| Orientation: | **Portrait** | **Portrait** |

You can use another font family, such as Lucinda Console, but you may need to adjust its size or the page margins slightly. That's because each font family has slightly different widths, even fixed-width fonts of the same nominal size.

If you need to use a larger font size, or larger page margins, you'll need to use Landscape **Orientation**, else you'll probably get messy looking printouts.

### Larger Font, Landscape Orientation

Again, these are Notepad terms and may be different in another editor:

**Format | Font...**

| | |
|---|---|
| Font: | Courier New |
| Style: | Regular |
| Size: | **11** |

**File | Page Setup...**

| | | |
|---|---|---|
| Paper Size: | **Letter** (N. America) | **A4** (International) |
| Left and Right Margins: | **0.75** inch (or **19** mm) | **28** mm (or **1.1** inch) |
| Orientation: | **Landscape** | **Landscape** |

## Alternatives to Notepad

Your default text editor is Notepad. If you prefer to use something else (assuming you have it installed), there is a way of telling Copan to use that editor instead:

1. Open either the Points List dialog (via **CoordFile | List Points...**) or the General Program Settings dialog (via **Program | Settings... | General...**).
2. Choose the **Other** button under **Info Display Tool**
3. Enter the full path to or **Browse...** to the desired text editor's executable file. For example, `C:\Program Files\MyTexEd\MyTexEd.exe` is the location of a hypothetical program called MyTexEd.

updated: 25-Mar-2011

**Copan *Lite* for Windows**

# Point Codes

Contents



A Copan *point code* (or *feature code*) is intended as an abbreviated yet very useful point attribute. Used carefully, it can describe several characteristics of a point. Although any string of 13 printable characters can be used for a point code, for Copan to fully exploit them,

- point codes must conform to a specific type and format, and
- there must be an appropriate external Point Codes Table.

## 1. Point Code Types

There are two kinds of point code — *basic* and *extended* — distinguished by their format and hence the amount of information that they convey about a point:

- **basic** A *basic* point code categorizes a point in a single feature class. It consists of a suitable reference to an entry in an external Point Codes Table and can be either *numeric* or *alphanumeric* in form:
  - A *numeric* code is chosen from the integer Code column of the codes table. For example, 2 for “power pole”, 14 for “coniferous tree”, or 34 for “top of wall”.
  - An *alphanumeric* code is chosen from the Desc column of the codes table. For example, “PP” for “power pole”, “CT” for “coniferous tree”, or “TW” for “top of wall”. Actually, an alpha code need only be the first sequence of characters of a Desc that uniquely distinguishes it, or even just the first character of the Desc if that particular Desc is the first one in the table. For example, if the following are all the Descs beginning with the letter T appearing in the table, “TL”, “telMH”, “tank cap” and “TW”, then they can be referenced merely by the following, “T”, “te”, “ta” and “TW”, respectively.

  Thus the basic code could be either 2 or PP for “power pole”; 14 or CT for “coniferous tree”; and 34 or TW for “top of wall”.

- An *extended* point code both categorizes a point in a feature class and conveys further information. It is in the form `B.E` where
  - `B` is the basic point code (as above), and
  - `E` is a string of extra alphanumeric characters, containing further coded information and depending on the type of extended code (see below).

The total character length of an extended code must still be 13 or less.

The above point code formatting is the minimum requirement in order for the List Points, Export to Ascii, etc, modules to correctly show the Desc or Description attributes of any points.

## 2. Extended Codes

In order for the List Points, Export to Ascii, etc, modules to correctly show the Extension attribute of a point, one of three kinds of *extended* code types must be distinguished and used: *hole*, *culvert* and *tree*.

- **hole** A *hole* point code is of the form `B.D` where
  - `B` is the basic point code (described above), and
  - `D` is the *hole depth*, in hundredths of a distance unit (e.g., if coordinates are in metres then depth is in cm).
- **culvert** A *culvert* point code is of the form `B.C` where
  - `B` is the basic point code (described above), and
  - `C` is the *culvert diameter*, in hundredths (cm).
- **tree** A *tree* point code is of the form `B.T.S` where
  - `B` is the basic point code (described above),
  - `T` is the *trunk diameter*, in hundredths (cm), and
  - `S` is the *spread diameter*, in distance units (m).

## 2. Point Codes Table

A codes table is a user-managed, tab-delimited, Ascii-based, single-byte encoded, lookup file, normally found in the Auxiliary Files folder.

The first line should contain 5 or 6 tab-separated headings (the last one being optional):

```
Code  Description  Type  Block  Layer  [Desc]
```

Each subsequent line describes a feature type, and must contain 5 or 6 tab-separated fields:

- an integer **Code** — a required, non-negative, whole number, to uniquely identify the feature class;
- a code **Description** — up to 31 characters — that names or describes the feature class, and can be used by List Points and Export to Ascii;
- a code **Type** (i.e., “basic”, “hole”, “culvert”, or “tree”), that can be used by List Points, and Export to Ascii;
- blank (for future use);
- blank (for future use); and
- a code **Desc** — up to 13 characters — that can serve as both a unique feature class identifier and an abbreviated form of description, and can be used by Edit Point, List Points, Export to Ascii, and COGO.

Note that

- There *must* be 4 or 5 tabs on every line.
- Although the fifth column is optional, the Desc attribute could be used in many places, so you're advised to fill it in.
- While a Codes Table can have any name, Copan looks for one named `codes.txt` until told otherwise.
- Single-byte character encoding, such as **ANSI**, **DOS** or **UTF-8** must be used. *Note*: multi-byte character encodings (such as Unicode or UTF-16) will *not* work.

## 3. Extended Code Examples

Suppose this is an excerpt of your point codes table:

```
Code    Description     Type     Block   Layer    Desc
0       control point   basic    CONT    survey   CP
1       lamp stand      basic    LAMP    util     LS
8       catch basin     hole     CATCH   util     CB
13      deciduous tree  tree     DTREE   veg      DT
36      culvert         culvert  CULV    topo     culv
```

The following could be an excerpt from List Points (option 7):

```
Code         Desc        Extension
------------ ----------- -----------------
0            CP
0.7          CP          (type 7)
1            LS
1.3          LS          (type 3)
8.125        CB          1.25 deep
8.85         CB          0.85 deep
13.150.10    DT          1.5 trnk 10 sprd
13.90.15     DT          0.90 trnk 15 sprd
36           culv
36.35        culv        0.35 diam
```

Note that

- The first and third point codes are basic so have no Extension.
- The "`.7`" and "`.3`" in the second and fourth point codes are merely user-determined codes that might be used to sub-classify the point. For example, 1, 2, or 3 could be used to specify wood, metal, or concrete as the type of lamp stand.

## 4. Codes Table Example

This is a portion of a typical Codes Table. (See the `\Sample Aux Files\sample_codes.txt` file.) Note that because it is tab-delimited, if you look at an actual file, the columns may not appear as neatly aligned as they do here. Also note that the codes need not be consecutive (there are no codes 23-26 or 30-35).

| Code | Description | Type | Block | Layer | Desc |
|---|---|---|---|---|---|
| 0 | control point | basic | CONT | survey | CP |
| 1 | lamp stand | basic | LAMP | util | LS |
| 2 | power pole | basic | POWPL | util | PP |
| 3 | guy wire | basic | POINT | util | GW |
| 4 | traffic light | basic | TRAFL | util | TL |
| 5 | utility box | basic | UTILB | util | UB |
| 6 | transformer box | basic | POINT | util | TB |
| 7 | electrical box | basic | ELECB | util | EB |
| 8 | catch basin | hole | CATCH | util | CB |
| 9 | sanit manhole | hole | SANMH | util | sanMH |
| 10 | storm manhole | hole | STMMH | util | stmMH |
| 11 | power manhole | hole | PWRMH | util | pwrMH |
| 12 | telephone manhole | hole | TELMH | util | telMH |
| 13 | deciduous tree | tree | DTREE | veg | DT |
| 14 | coniferous tree | tree | CTREE | veg | CT |
| 15 | fire hydrant | basic | FIREH | util | FH |
| 16 | water valve | basic | WATRV | util | WV |
| 17 | gas valve | basic | GASV | util | GV |
| 18 | sign post | basic | SIGNP | util | SP |
| 19 | parking meter | basic | PARKM | util | PM |
| 20 | guard post | basic | GARDP | util | GP |
| 21 | tank cap | basic | POINT | util | tank cap |
| 22 | monitoring well | basic | MNWELL | topo | MW |
| 27 | manhole | hole | MANHL | util | MH |
| 28 | bore hole | basic | BH | topo | BH |
| 29 | steps | basic | POINT | topo | steps |
| 36 | culvert | culvert | CULV | topo | culv |

updated: 25-Mar-2011

updated: 21-Sep-2010

*Copan for Windows*

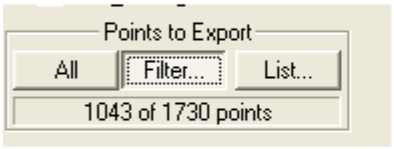
# Point Filters

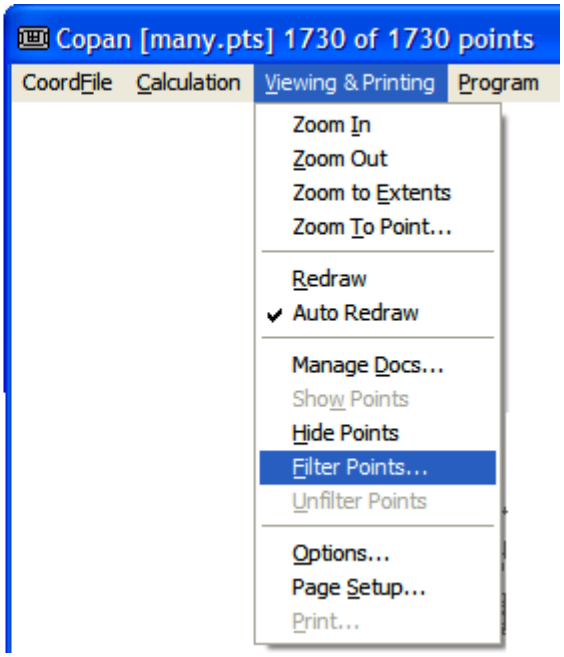
Contents



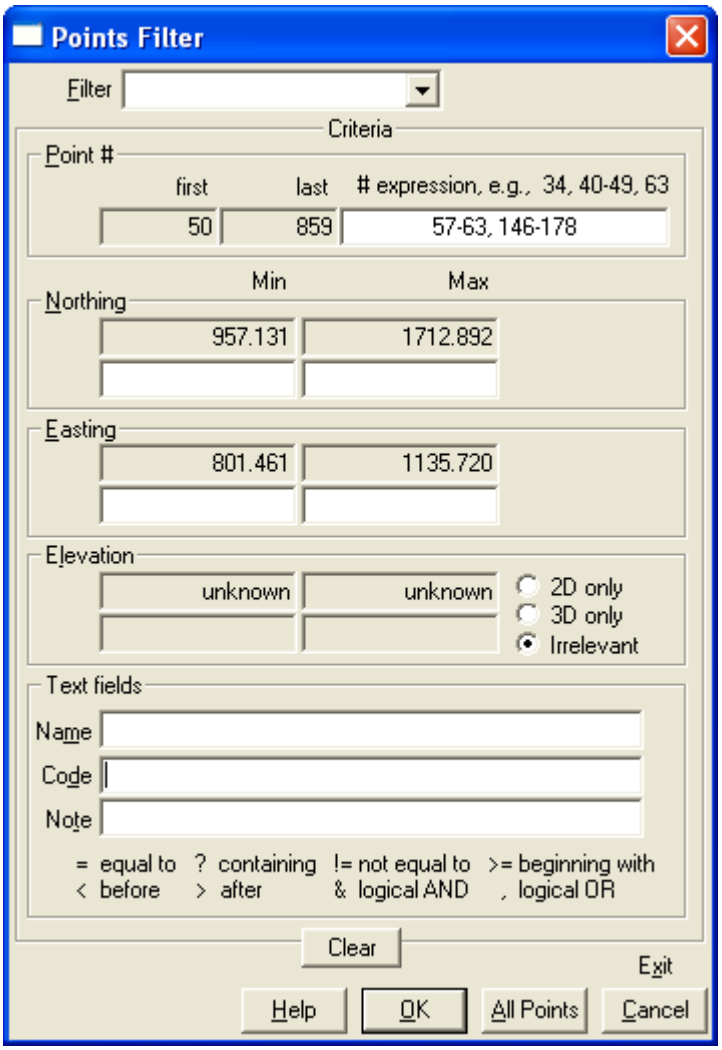
At times you may wish to view or work with a specific subset of points from an open coordfile, rather than the whole file. You can establish such a subset by specifying a *filter* (or *query*). Typically, to specify a filter, you just identify sequences or ranges of point numbers. You can, however, also identify ranges of coordinate values, as well as conditions for the three alphanumeric fields.

## 1. To Filter Points




1. Click the **Filter...** button from whichever module is processing a group of points or choose **Viewing & Printing | Filter Points...** to restrict the viewing of points.
2. Optionally, select an existing filter from the **Filter** list box.
3. Enter or edit the details of the filter (see Filter Criteria below).
4. Optionally, enter a new name for the filter in the **Filter** list box.
5. Click **OK**.




## 2. Filter Criteria

- A **Point # expression** lists individual and ranges of point numbers. The syntax is just like the way page ranges in many print dialogs are specified:

  *point-expression*
  *point-range*
  *point-range* , *point-expression*
  *point-range*
  *point-number*
  *point-number* - *point-number*

  For example, a Point # expression of `1-99, 110, 250-349, 900` means "Points 1 to 99, 110, 250 to 349, and 900."

- The **Min** and **Max Northing** and **Easting** values are *inclusive* extremes. For example,
  - A Min and Max Northing of `290` and `380` means "290 <= northing <=380."
  - A Min Easting of `150.2` means "easting >= 150.2."

- Because points can have null or missing elevations, there are three mutually exclusive **Elevation** options:
  - If you only want 2D points, choose **2D only**.
  - If you only want 3D points, choose **3D only**.
  - If you want 2D and 3D points, choose **Irrelevant**.
- To enter **Elevation** extremes, you must choose the **3D only** option.
  - A Min and Max Elevation of `10` and `30` means "10 <= elevation <=30."
- Criteria based on **Name**, **Code**, or **Note** fields require *text expressions*.

## 3. Text Expressions

### Syntax

In these quasi-formal syntax definitions, a defined term (outdented) is followed by its definition (indented), with alternative definitions separated by vertical lines or on separate lines. Puzzled? See the Examples.

*text-expression*
    *predicate*
    *text-expression* *logical-operator* *text-expression*
    ( *text-expression* )
*predicate*
    *relational-operator* *value*
*relational-operator*
    = | ? | != | <> | < | > | <= | >=
*value*
    any text | " any text containing a special character "
*logical-operator*
    & | ,

### Relational and Logical Operators

| Operator | Meaning |
|---|---|
| = | equal to |
| ? | containing |
| != or <> | not equal to |
| < | alphabetically *before* |
| > | alphabetically *after* |
| <= | alphabetically *before* or equal to |
| >= | alphabetically *after* or equal to |
| & | logical *and* |
| , | logical *or* |
| ( ) | logical *group*, required when logical *or* must be evaluated before logical *and* |

**Examples**

| Expression | Meaning |
|---|---|
| `=""` | is null (blank) |
| `<> ""` | is not null (not blank) |
| `= 13, =14` | equal to “13” or “14” |
| `<> 12 & <> 16` | equal to anything but “12” or “16” |
| `? O & ? P` | containing “O” and “P” |
| `>= G & < K` | beginning with letters “g” to “j” |
| `? BC & (? "?", ? "!")` | containing “BC” and containing “?” or “!” |

## 4. Notes

- If criteria are entered for more than one attribute, the conjunction between them is assumed to be logical *and*. Thus, for example, if both Point # and Code expressions are entered, the filter will allow only points simultaneously meeting both sets of criteria.
- There is currently no way of specifying a logical *or* between different field criteria.
- There is currently no way of saying “not containing”.
- There is currently no way of specifying criteria for other attributes (e.g., Locked, Modified, etc).
- The full syntax for the criteria expressions may evolve. Give us your suggestions.
- By default, after you filter some points for whatever reason, only those points passing the filter are immediately viewable. To make all points viewable, regardless of the current filter, do one of two things:
  - either select **Viewing & Printing | Ufilter Points**, each time you need to,
  - or, for a lasting choice, select **Program | Settings... | General...** and check **Ignore filter when viewing points** under **Logistics**.

updated: 21-Sep-2010

updated: 24-Mar-2011

***Copan for Windows***

# Field Files

Contents



To Copan, field survey data consist of measured or reduced *angles* and *distances*, and come in *field files*. The "field modules" — § Field Data Processing, § Field Bearings Processing, and § Field Resections Processing — can read field files.

Note that coordinates — however they've been collected or derived — are not considered to be field data and can only be imported to Copan via the generic import facility (see § Import and Export). If coordinates *are* in a field file, they will be displayed as comments and will not processed.

## 1. Field File Types

- Copan can *read* field data in various different file formats. While the files can have any names, certain extensions are advisable:
  - Copan field file — **.fld** or **.f***
  - Geodimeter file — **.job** or **.raw**
  - TDS RAW/RW5 file — **.raw** or **.rw5**
  - Nikon RAW file — **.raw**
  - Leica GSI-8 file — **.gsi**
  - Leica GSI-16 file — **.gsi**
  - Wild GRE file — **.gre**
- Copan does not communicate with total stations themselves, only their downloaded files.
- Copan only *saves* field data in its *own* format.
- Field data files are plain text (or Ascii) so can be edited outside of Copan (but be careful to follow the relevant syntax).
  *Single-byte* character encoding, such as ANSI, DOS or UTF-8 *must* be used.
  *Multi-byte* character encodings (such as Unicode or UTF-16) will *not* work.
- Note that while both topographic and geodetic variations of Leica GSI-16 data can be read, some manual editing of the loaded data may be required before calculating. This is because GSI files often do not contain setup data (ref. and HI) even though such data are in the instrument!

Let us know ( at support@underhill.ca ) if you'd like Copan to read field data in other formats. Please note the software version and release date (see § Software Version) in all communications.

## 2. Copan Field File Format

- A Copan field file usually consists of numerous *Setup* and *Observation* data lines. It may also contain *comment* lines, as well as *Project*, *Instrument*, or *Scale* data lines, if required.
- Each data line consists of various tab-separated *attribute-key* = *value* pairs (i.e., a tab character must separate each *attribute-key*=*value* pair). Suitable combinations of these attributes depend on the particular situation and are specified in the relevant Field modules chapter.
- Spaces may be placed either side of equals signs (e.g., `HS = 1.56`) if desired.
- A *Project* data line consists of a combination of `job`, `dat`, `tim`, `svr`, `FB`, or `pgs` attributes.
- An *Instrument* data line consists of a combination of `instr`, `VCC`, `DC`, or `DF` attributes.
- A *Scale* data line consists of `SF` or `UF` attributes.
- A *Setup* data line consists of a combination of `at`, `ref/az`, `HC`, or `HI` attribute-keys.
- An *Observation* data line consists of a combination of `to/az`, `HC`, `VC`, `SD`, `HS`, `cod`, `not`, `nam`, `dis/bas`, or `end` attribute-keys.
- Typically, Observations are indented from Setups for readability, but that is not a requirement.
- A valid *comment* line begins with zero or more spaces or tabs, then has two commas (,,), periods (..), slashes (//), semi-colons (;;), or backslashes (\\), which are followed by any character string. Comments are not processed or stored as point notes (but are saved within the field file). Use them to temporarily remove observations, to make remarks, or to log relevant events.

## 3. Field Attribute Keys

Here are the currently recognized key words representing attributes in the *attribute-key* = *value* pairs. Note that correct case *is* important.

| Attribute key | Attribute meaning | Ignored during calculation |
|---|---|---|
| `at` | setup (or occupied) point number | |
| `az` | azimuth (or bearing) | |
| `bas` | base point number for a REM point | |
| `cod` | point (or feature) code | |
| `dat` | date | * |
| `DC` | distance correction constant | |
| `DF` | distance correction factor (related to PPM) | |
| `dis` | distance point number for a REM point | |
| `end` | previous line is last leg of traverse or last shot of resection | |

| | | |
|---|---|---|
| `FB` | field book name or number | * |
| `HC` | horizontal circle reading (or direction) | |
| `HI` | height of instrument | |
| `HS` | height of signal (reflector, prism, or target) | |
| `instr` | instrument name or number | * |
| `job` | job/project name or number | * |
| `not` | point note | |
| `pgs` | field book pages | * |
| `ref` | reference (or backsight) point number | |
| `SD` | slope distance | |
| `SF` | (combined) scale factor (i.e., projection and elevation factor) | |
| `svr` | surveyor/operator name or number | * |
| `tim` | time | * |
| `to` | target (or foresight) point number | |
| `UF` | units conversion factor | |
| `VC` | vertical circle reading (or zenith angle) | |
| `VCC` | vertical circle correction | |

Let us know ( at support@underhill.ca ) if you'd like Copan to recognize other attributes. Please note the software version and release date (see § Software Version) in all communications.

## 4. Remote Elevation Measurement

A *remote elevation measurement* (or *REM*) is an indirect observation of height and position of a topographic point whose slope distance is unobservable. If you can observe a vertical angle to an otherwise inaccessible point (such as the top of a mast) and you cannot get a reflectorless slope distance, you can determine the 3D coords for that point *indirectly*, via another point that you can shoot to normally and that is either vertically below the inaccessible point or at the same horizontal distance away.

Let's call the high inaccessible point the “REM” point, an accessible point vertically below it a “base” point, or an accessible point at the same horizontal distance away a “dist” point. To insert a REM observation, reference must be made to either a base point or a dist point:

- If referencing a “base” point
    1. Enter the observation via the Observation boxes along the bottom, ensuring the HC and SD boxes are blank. This is often also known as a “vertical angle offset” observation.
    2. Insert the observation, noting that `bas= prev` is added to the observation automatically.

3. If the previous side-shot is not the base point, edit the `prev` with the relevant point number, which can be any normally observed side-shot — even one from an earlier or later setup.

- If referencing a "dist" point
  1. Enter the observation via the Observation boxes along the bottom, ensuring the SD box is blank.
  2. Insert the observation, noting that `dis= prev` is added to the observation automatically.
  3. If the previous side-shot is not the dist point, edit the `prev` with the relevant point number, which must be from the same setup.
- As an alternative, add the observation as usual but use *dummy* values for SD (and possibly HC). Then edit the line as appropriate. That is, *delete* the `SD=` (and possibly `HC=` ) portions and *add* the `bas=` or `dis=` portion.

updated: 24-Mar-2011

updated: 21-Sep-2010

*Copan for Windows*

# Traverse Processing

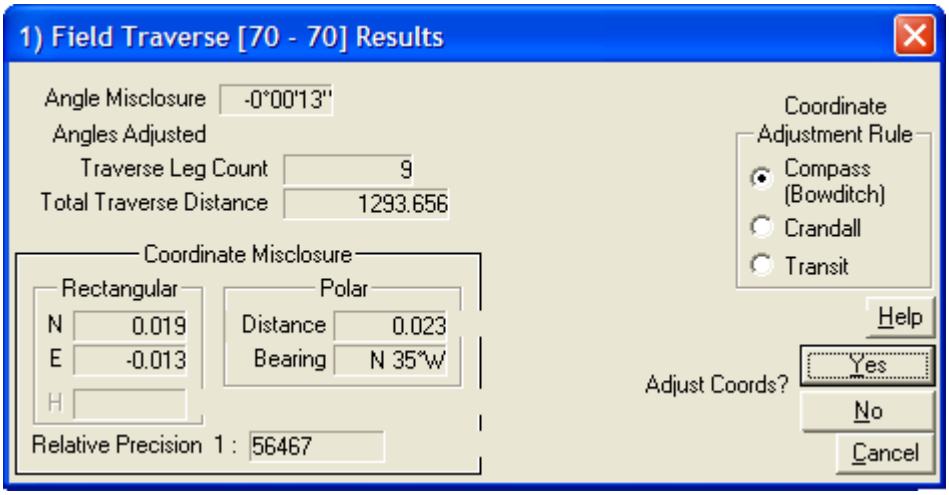
Contents



Various Copan modules allow you to process sets of field or map traverses. When you choose **Calculate** (or **OK**), each traverse is summarized in a Traverse Results window or its detailed results are listed in an info display file. In many cases, the traverse can be adjusted for coordinate misclosure.

## 1. Field Traverse Results

Two "field" modules — § Field Data Processing and § Field Bearings Processing — can process and adjust raw field traverses via Traverse Results dialogs:

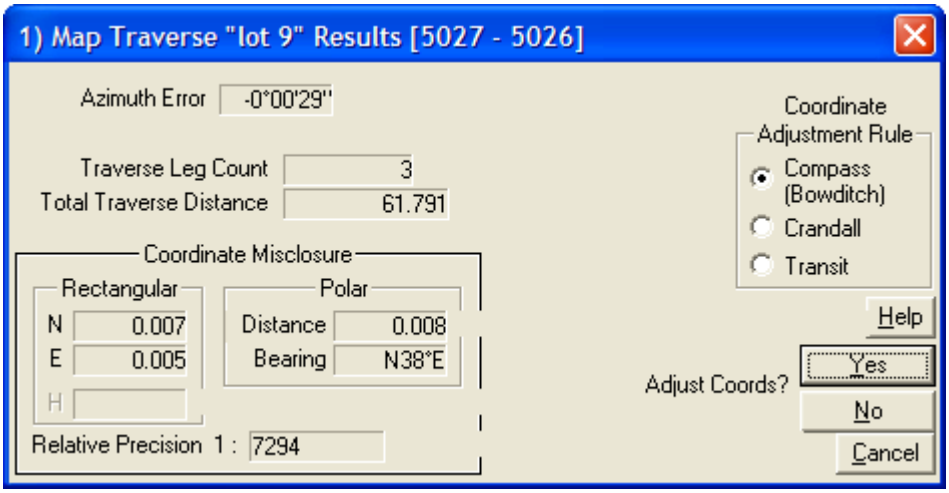



- The first number in the title bar is simply an auto-generated index for the traverse (if there are *N* traverses, they'll be numbered 1 - *N*).
- The square bracketed numbers in the title bar are the start and end point numbers for the traverse (if it is a loop, they'll be the same).
- If a closing (or redundant) angle is present, the **Angle Misclosure** will be shown, along with a note saying the Angles have been Adjusted. The overall angular correction is divided equally among the traverse angles.
- The Traverse Leg Count and Total Distance of the traverse are shown next.

- The large frame shows the **Coordinate Misclosure** — its Rectangular and Polar components, and the Relative Precision.
- If any misclosure is larger than a certain tolerance, as specified in § Program Settings, it is marked with a warning.

## 2. Map Traverse Results

Although two “map trav” modules — § Map Traverses and § Map Checks — can process reduced map traverses, only the former can adjust traverses, via Traverse Results dialogs:




- The first number in the title bar is simply an auto-generated index for the traverse (if there are *N* traverses, they'll be numbered 1 - *N*).
- The quoted name in the title bar is the optional traverse label.
- The square bracketed numbers in the title bar are the start and end point numbers for the traverse (if it is a loop, they'll be the same).
- If it is a non-looping, closing traverse, the overall **Azimuth Error** will be shown.
- The Traverse Leg Count and Total Distance of the traverse are shown next.
- The large frame shows the **Coordinate Misclosure** — its Rectangular and Polar components, and the Relative Precision.
- If any misclosure is larger than a certain tolerance, as specified in § Program Settings, it is marked with a warning.

## 3. Traverse Adjustment

- To adjust the traverse (i.e., eliminate the coordinate misclosure), choose which **Coordinate Adjustment Rule** to use
  - **Compass** or (**Bowditch**), which allocates the overall coordinate correction to each point in proportion to its distance along the traverse;
  - **Crandall**, which adjusts the traverse distances but not the bearings; or

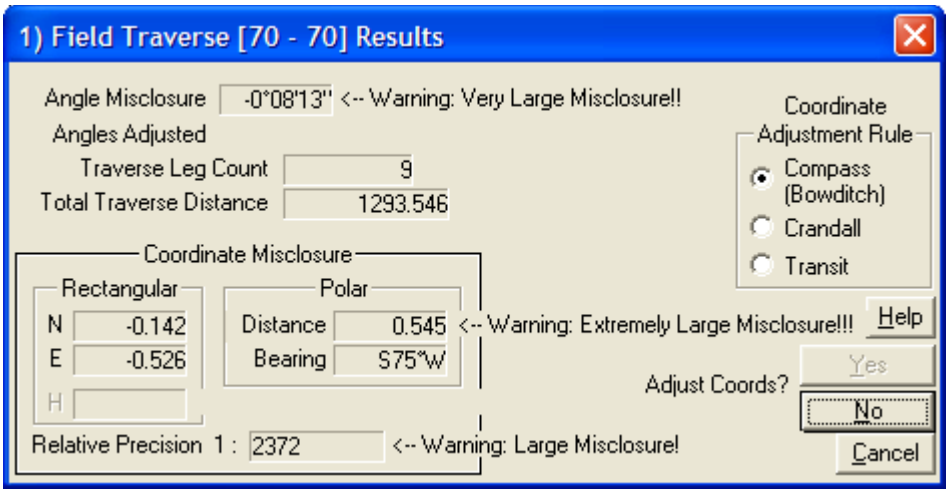
  - **Transit**, which adjusts the traverse distances more than the bearings (but which depends on the traverse's orientation);

  then **Yes**. If there are subsequent traverses, the next one will be presented automatically.

- To leave the traverse unadjusted and to see the next traverse, choose **No**.
- To cancel this and subsequent traverse processing, choose **Cancel**.

Note, whether or not a traverse adjustment is made, a sum-of-angles imbalance, if present, has already been balanced prior to presentation of the coordinate misclosure.

### 4. Warnings




- If any misclosure is larger than a certain tolerance, it is marked with a "Warning: Large Misclosure!"
- If a misclosure is much (over 5 times) larger than the tolerance, it is marked with a "Warning: Very Large Misclosure!!"
- If a misclosure is very much (over 10 times) larger than the tolerance, it is marked with a "Warning: Extremely Large Misclosure!!!" and subsequent adjustment is disabled.
- You can modify the tolerances via the Geomatics section of the General Program Settings window. (Open it via the **Settings...** button on the Field Data Process or Map Traverse Adjustment windows or via the **Program | Settings... | General...** menu.)

### 5. Traverse Listing

Both the field traverse processing modules *can*, and both the map traverse processing modules *do*, list the traverses in an info display file. A traverse listing includes the results shown in the Traverse Results dialog as well as the bearing and distance of each leg in the traverse.

The “inverse map traverse” module — § Area and Perimeter Calculations — processes sequences of points and also lists the above results, except for misclosures (as there are none), in an info display file.

## 6. Traversing Curves

A map traverse (i.e., sequences of bearings and distances) can define curves if the curves' radials are included as traverse legs. The same is true of inverse map traverses (i.e., sequences of points). See § Traversing Curves for details on how curve radials are defined as traverse legs.

updated: 21-Sep-2010

updated: 30-May-2011

**Copan *Lite* for Windows**

# Export to IOB File

While Copan does not currently process arbitrary survey networks, for those of you who have access to and know how to use the BitWise Ideas least-squares adjustment program, GeoLab, Copan is your friend.

Copan can export field data to IOB format files. Note: We do not provide help on the GeoLab software!

## User-control of IOB Output

You have some control over IOB sigma records and horizontal angle output.

1. Find your `program.settings.ini` file (it is in the same location as the `program.trace.txt` files, see User Settings Files in § File Locations).
2. Open the file in a text editor.
3. Copy and paste these lines to the bottom of the file.

```
[Export-to-IOB]
    reducing = 1
    instrument = TS
    iob_HA_key = A
    iob_VA_key = Z
    iob_SD_key = D
```

4. Edit as explained below.
5. Save.

(You needn't worry about the presence/absence of blank lines or presence/absence of leading spaces or spaces either side of "=".)

### Angles or Direction Sets

Do you want reduced or raw horizontal angles?

- `reducing = 1` (the default) means output ANGL records
- `reducing = 0` means output DSET and backsight and foresight DIR records

### Sigma IDs

Would you like to change the “Sigma IDs”?

Three-character Sigma IDs are used to cross-reference the instrument's horizontal and vertical angle variances and distance variance. Edit the

- two characters after "`instrument =`" to your preferred instrument ID, but use exactly two characters.

- single character after "`iob_HA_key =`" to your preferred horizontal angle variance ID, but use exactly one character.
- single character after "`iob_VA_key =`" to your preferred vertical angle variance ID, exactly one character.
- single character after "`iob_SD_key =`" to your preferred slope distance variance ID, one character.

**Example field data**

```
at= 1  ref= 2  HC=  0.0000                        HI= 1.617
  to= 2  HC=  0.0000  VC= 89.5735  SD=  13.231  HS= 1.584
  to= 4  HC= 92.2849  VC= 89.4558  SD=   2.349  HS= 1.509
  to= 3  HC= 10.2706  VC= 90.2528  SD=  13.270  HS= 1.641
```

**Sample of output using above data and program.settings**

```
 ANGL TSA 1           2           4                92 28    49
 ZANG TSZ 1           4                            89 45    58
 DIST TSD 1           4                                  2.349
 HT       4                1.509
```

**Using same data and settings, except `reducing = 0`**

```
 DSET TSA
 DIR      1           2                             0 00    00
 DIR      1           4                            92 28    49
 ZANG TSZ 1           4                            89 45    58
 DIST TSD 1           4                                  2.349
 HT       4                1.509
```

**Future Improvements**

In time, we may provide a GUI for editing the settings and adding sigma values. We may allow other IOB observation records.

updated: 30-May-2011

updated: 21-Sep-2010

***Copan for Windows***

# Bearing Expressions

A fairly sophisticated set of options exists for entering *bearings* (*headings* or *azimuths*) into Copan.

You can enter bearings *explicitly*, in **whole-circle** format or in **quadrant** format, or *implicitly* as being the bearing between two specified coordfile points. You can even include a "+" or "–" operator in an expression to add a correction or turned angle to, or subtract from, a bearing. A correction, turned angle, or whole-circle bearing is always positive clockwise and negative anticlockwise. Finally, unless the appropriate Program Settings are made, angle units are *degrees-minutes-seconds*, and bearings are north-based (i.e., north=0).

(Note that while lay people may know bearings as *directions*, in surveying, a "direction" usually means an arbitrary angle reading, where an angle is the difference between two directions. Even so, we use "cardinal direction" below to have the common meaning.)

## Syntax

If you measure angles in **degrees-minutes-seconds** (i.e., 1 circle = 360°), when entering bearings or angles, always use the format `dd.mmss`. For example, you must enter 24°52'10.5" as `24.52105` and you must enter 30' as `0.30` (or just `.3`).

If you measure angles in **gons** or **grads** (i.e., 1 circle = 400$^{g}$), just enter bearings or angles *as is* (e.g., enter 24.82105$^{g}$ as `24.82105`).

In these quasi-formal syntax definitions, a defined term (outdented) is followed by its definition (indented), with alternative definitions on separate lines or separated by vertical bars. Puzzled? See the Examples.

*bearing-expression*
    *bearing*
    *bearing operator angle*
    *implied-bearing* [**.** 1 | 2 | 3] (where .1, .2 or .3 means add ¼, ½ or ¾ circle)
*bearing*
    [*operator*] *angle*
    *quad-bearing*
    *implied-bearing*
*angle*
    *ddd.mmsss*
    *ggg.ggggg*
*quad-bearing*
    N | S | E | W | NE | SE | NW | SW
    N | S *acute-angle* E | W (where *acute-angle* is ≤ ¼ circle)
*operator*
    + | –

*implied-bearing*

*from-point-number* , *to-point-number*

, *to-point-number* (valid if the *from-point-number* is already entered in an appropriate from-point field)

## Examples

In the following tables, note that case doesn't matter and that spaces are not required. Also, the bearings are north-based.

**Table 1: Degrees-Minutes-Meconds Bearing Expressions.**

For the lower half of the table, assume that point 102 is at a bearing of exactly 035° from point 101.

| Expression | Bearing | Explanation |
|---|---|---|
| `210.5613` | 210°56'13" | explicit whole-circle bearing |
| `-10` | 350° | negative bearing, normalized |
| `24 - 2` | 022° | bearing, plus correction or turned angle |
| `s` | 180° | cardinal direction |
| `E - 10` | 080° | cardinal direction, plus correction or turned angle |
| `w+5` | 275° | cardinal direction, plus correction or turned angle |
| `n 35 W` | 325° | explicit quadrant bearing |
| `s 3 e +0.38` | 177°38' | quadrant bearing, plus correction or turned angle |
| `s 4 w` | 184° | quadrant bearing |
| `sw` | 225° | cardinal direction |
| `101, 102` | 035° | implied bearing from point 101 to point 102 |
| `101, 102 .1` | 125° | implied bearing from point 101 to point 102, plus 090° |
| `101, 102 .2` | 215° | implied bearing from point 101 to point 102, plus 180° |
| `101, 102 .3` | 305° | implied bearing from point 101 to point 102, plus 270° |
| `101, 102 + 10` | 045° | implied bearing from point 101 to point 102, plus 10° |
| `101, 102 - 15` | 020° | implied bearing from point 101 to point 102, minus 15° |
| `, 102` | 035° | implied bearing from point 101 to point 102 |
| `, 102 .2` | 215° | implied bearing from point 102 to point 101 |

For the last two expressions, assume that “101” is already in an appropriate from-point field.

**Table 2: Gons (or Grads) Bearing Expressions.**

For the lower half of the table, assume that point 102 is at a bearing of exactly 035$^{g}$ from point 101.

| Expression | Bearing | Explanation |
|---|---|---|
| 210.5613 | 210.5613$^{g}$ | explicit whole-circle bearing |
| -10 | 390$^{g}$ | negative bearing, normalized |
| 24 - 2 | 022$^{g}$ | bearing, plus correction or turned angle |
| s | 200$^{g}$ | cardinal direction |
| E - 10 | 090$^{g}$ | cardinal direction, plus correction or turned angle |
| w+5 | 305$^{g}$ | cardinal direction, plus correction or turned angle |
| n 35 W | 365$^{g}$ | explicit quadrant bearing |
| s 3 e +0.38 | 197.38$^{g}$ | quadrant bearing, plus correction or turned angle |
| s 4 w | 204$^{g}$ | quadrant bearing |
| sw | 250$^{g}$ | cardinal direction |
| 101, 102 | 035$^{g}$ | implied bearing from point 101 to point 102 |
| 101, 102 .1 | 135$^{g}$ | implied bearing from point 101 to point 102, plus 100$^{g}$ |
| 101, 102 .2 | 235$^{g}$ | implied bearing from point 101 to point 102, plus 200$^{g}$ |
| 101, 102 .3 | 335$^{g}$ | implied bearing from point 101 to point 102, plus 300$^{g}$ |
| 101, 102 + 10 | 045$^{g}$ | implied bearing from point 101 to point 102, plus 10$^{g}$ |
| 101, 102 - 15 | 020$^{g}$ | implied bearing from point 101 to point 102, minus 15$^{g}$ |
| , 102 | 035$^{g}$ | implied bearing from point 101 to point 102 |
| , 102 .2 | 235$^{g}$ | implied bearing from point 102 to point 101 |

For the last two expressions, assume that “101” is already in an appropriate from-point field.

updated: 21-Sep-2010

updated: 21-Sep-2010

***Copan for Windows***

# Distance and Offset Expressions

A fairly sophisticated set of options exists for entering *distances* (*radii* or *lengths*) into Copan. A similar set of options also exists for entering *offsets*, where an offset is a relatively short distance left or right of, and perpendicular to, a particular bearing or line.

You can enter distances and offsets *explicitly* as values or *implicitly* as being the distance between two specified coordfile points. You can even include a simple arithmetic operator in an expression to add, subtract, multiply or divide a distance or offset by another value. An offset is always prefixed with an “L”, “R”, “–” or “+” character.

## Syntax

In these quasi-formal syntax definitions, a defined term (outdented) is followed by its definition (indented), with alternative definitions on separate lines or separated by vertical bars. Puzzled? See the Examples.

*offset-expression*
    *offset-prefix* *distance-expression*
*distance-expression*
    *distance*
    *distance* *operator* *quantity*
*distance*
    *quantity*
    *implied-distance*
*operator*
    + | – | * | /
*implied-distance*
    *from-point-number* , *to-point-number*
    , *to-point-number* (valid if the *from-point-number* is already entered in an appropriate from-point field)
*offset-prefix*
    L | R | – | + (the prefix is always required)

## Examples

In the following tables, note that case doesn't matter and that spaces are not required. Also, assume that point 102 is at a distance of exactly 25 from point 101.

**Table 1: Distance Expressions.**

| Expression | Distance | Explanation |
|---|---|---|
| `10.2` | 10.2 | explicit distance |
| `10 + 2` | 12 | distance plus distance |
| `10-2` | 8 | distance minus distance |
| `10*3.5` | 35 | distance times factor |
| `10/2.5` | 4 | distance divided by divisor |
| `101, 102` | 25 | implied distance from point 101 to point 102 |
| `101,102+6` | 31 | implied distance, plus distance |
| `101,102/5` | 5 | implied distance, divided by divisor |
| `, 102` | 25 | implied distance from point 101 to point 102 |
| `, 102 -4.5` | 20.5 | implied distance, minus distance |

For the last two expressions, assume that “101” is already in an appropriate from-point field.

**Table 2: Offset Expressions.**

| Expression | Offset | Explanation |
|---|---|---|
| `+ 3.5` | R 3.5 | offset right, explicit distance |
| `L 5` | L 5 | offset left, explicit distance |
| `-5` | L 5 | offset left, explicit distance |
| `R6.4/2` | R 3.2 | offset right, distance divided by divisor |
| `r101, 102` | R 25 | offset right, distance implied (from 101 to 102) |
| `l 101,102/2` | L 12.5 | offset left, implied distance divided by divisor |
| `r , 102` | R 25 | offset right, distance implied (from 101 to 102) |
| `+ , 102 / 5` | R 5 | offset right, implied distance divided by divisor |

For the last two expressions, assume that “101” is already in an appropriate from-point field.

updated: 21-Sep-2010

updated: 21-Sep-2010

***Copan for Windows***

# Shift Expressions

It is easy to enter an implied shift (translation or slide) from one point to another.

## Syntax

In these quasi-formal syntax definitions, a defined term (outdented) is followed by its definition (indented), with alternative definitions on separate lines.

*shift-expression*
    *point-number* *separator* *point-number*
    *point-number*
*separator*
    **,** | **–**

The implied shift is always *from* the first point *to* the second point. A single point number implies a shift from it to the origin and is rarely used. Puzzled? See the Examples.

## Examples

Assume that these points exist:

| Point | Northing | Easting | Elevation |
|---|---|---|---|
| 1 | 100 | 100 | 10 |
| 2 | 120 | 110 | 12 |
| 3 | 130 | 90 | |

The following shift expressions would yield these shift vectors (the first four all being equivalent):

| Expression | Northing shift | Easting shift | Elevation shift |
|---|---|---|---|
| `1,2` | 20 | 10 | 2 |
| `1 , 2` | 20 | 10 | 2 |
| `1-2` | 20 | 10 | 2 |
| `1 -2` | 20 | 10 | 2 |
| `2 - 1` | -20 | -10 | -2 |
| `2 - 3` | 10 | -20 | 0 |
| `3 ,1` | -30 | 10 | 0 |
| `3` | -130 | -90 | 0 |
| `2` | -120 | -110 | -12 |

updated: 21-Sep-2010

updated: 21-Sep-2010

***Copan for Windows***

# Traversing Curves

Contents



Copan's Map Check and Map Traverse modules allow you to calculate *circular curve parameters*, given certain inputs. If you have the *azimuth* (or *bearing*) and *distance* (or *radius*) from the beginning of the curve in to the *centre* of curvature, and the bearing out from the centre to the end of curve, Copan will calculate the curve's *arc length*, *chord length*, and *arc angle* (*deflection* angle or *delta* angle). Curves may be *tangential* or not and may be combined in sequences to form *compound* or *reverse* curves. The Map Traverse module will also calculate the centre and end of curve coordinates.

The Area and Perimeter Calculation module also can calculate curve parameters, if the key curve points are already in place. In this case, because it is an inverse traverse, you don't enter the bearing and distance of the curve radials; they are part of the output.

## 1. Curve Codes

Curves are assumed to be “traversed” in this sequence: beginning of arc, centre of circle, and end of arc. If it defines part of a curve, a point must be given an appropriate *curve code*:

| Curve code | Meaning |
|---|---|
| BC | beginning of curve |
| TBC | beginning of tangential curve |
| EC | end of curve |
| TEC | end of tangential curve |
| POC | point of connection (end of one curve and beginning of another) |
| TPOC | tangential point of connection |
| C | centre of clockwise (right) curve |
| CC | centre of counterclockwise (left) curve |

Note that it is the “to point” end of a traverse leg which must be given the relevant curve code.

## 2. Example Curves

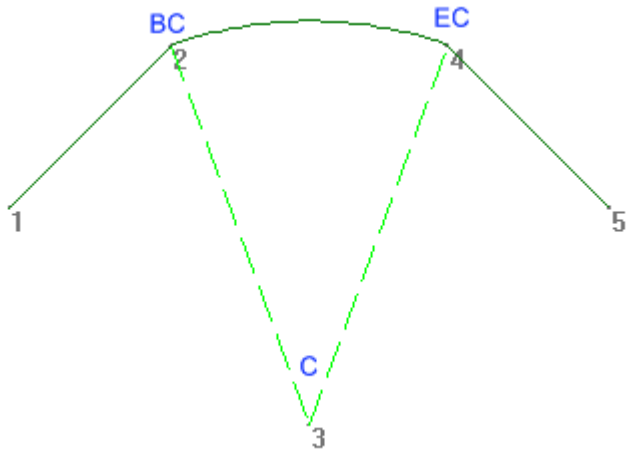
Each of these examples shows

- an annotated sketch of a curve (with points numbered in the order that they're traversed), and
- the data input to, and the results listed by, the Map Traverse or Map Check modules.

In the case of the Inverse Map Trav module, assuming the points are already on file, the input would consist of point numbers and curve codes but not bearings or distances.

Also, note that, although the points are all numbered consecutively for clarity, they need not be so.

**Example 1: Simple Non-tangential Curve.**



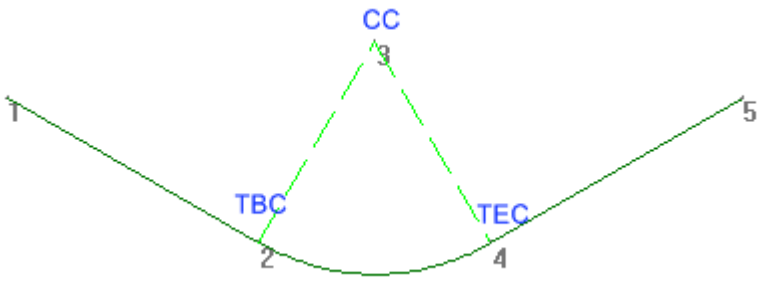

```
Data Input

Point  Bearing  Distance    Curve code
1
2      N45E     50          BC
3      S20E     88          C
4      N20E     "           EC
5      S45E     50


Curve Results

   Beg   Cent    End      Arc   Chord   Radius    Angle
     2      3      4   61.436   60.196  88.000  + 40°00'00"
```

**Example 2: Simple Tangential Curve.**



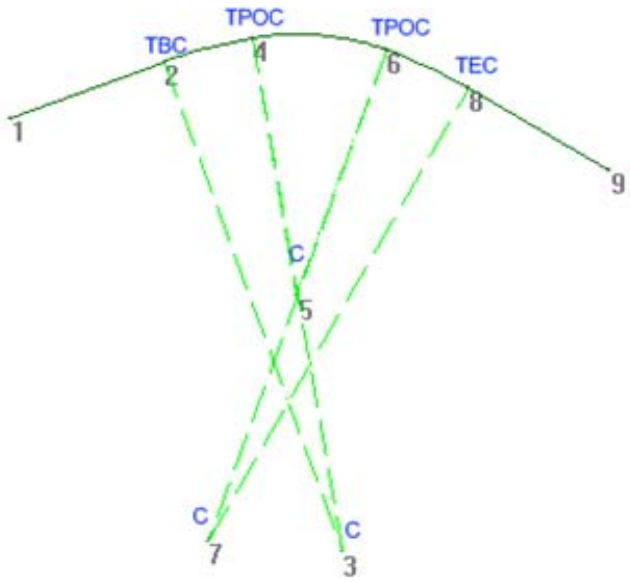

Data Input

| Point | Bearing | Distance | Curve code |
|---|---|---|---|
| 1 | | | |
| 2 | 120 | 50 | TBC |
| 3 | 030 | 40 | CC |
| 4 | 150 | " | TEC |
| 5 | 060 | 50 | |

Curve Results

| Beg | Cent | End | Arc | Chord | Radius | Angle |
|---|---|---|---|---|---|---|
| 2 | T 3 | 4 | T 41.888 | 40.000 | 40.000 | - 60°00'00" |

**Example 3: Compound Tangential Curve.**




Data Input

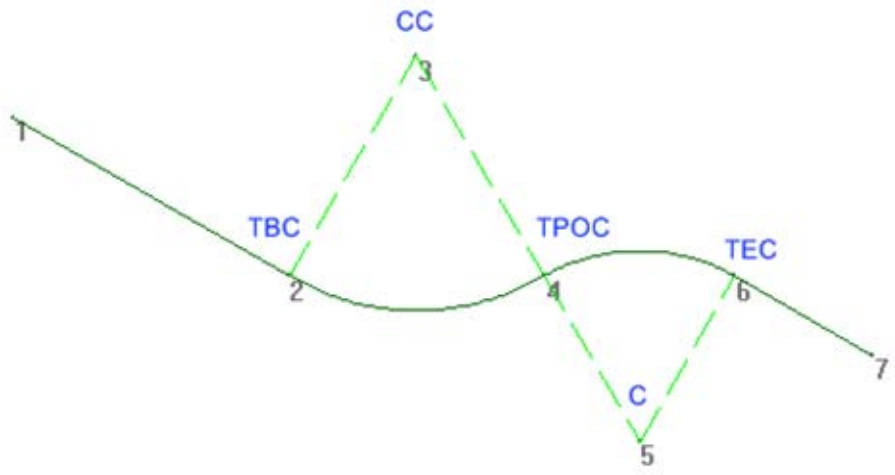
| Point | Bearing | Distance | Curve code |
|---|---|---|---|
| 1 | | | |

| | | | |
|---|---|---|---|
| 2 | 070 | 25 | TBC |
| 3 | 160 | 80 | C |
| 4 | 350 | " | TPOC |
| 5 | 350-180 | 40 | C |
| 6 | 020 | " | TPOC |
| 7 | 020+180 | 80 | C |
| 8 | 030 | " | TEC |
| 9 | 120 | 25 | |

Curve Results

| Beg | | Cent | End | | Arc | Chord | Radius | Angle |
|---|---|---|---|---|---|---|---|---|
| 2 | T | 3 | 4 | | 13.963 | 13.945 | 80.000 | + 10°00'00" |
| 4 | | 5 | 6 | | 20.944 | 20.706 | 40.000 | + 30°00'00" |
| 6 | | 7 | 8 | T | 13.963 | 13.945 | 80.000 | + 10°00'00" |

**Example 4: Reverse Tangential Curve.**




Data Input

| Point | Bearing | Distance | Curve code |
|---|---|---|---|
| 1 | | | |
| 2 | S60E | 50 | TBC |
| 3 | N30E | 40 | CC |
| 4 | S30E | " | TPOC |
| 5 | " | 30 | C |
| 6 | N30E | " | TEC |
| 7 | S60 | 25 | |

Curve Results

| Beg | | Cent | End | | Arc | Chord | Radius | Angle |
|---|---|---|---|---|---|---|---|---|
| 2 | T | 3 | 4 | | 41.888 | 40.000 | 40.000 | - 60°00'00" |
| 4 | | 5 | 6 | T | 31.416 | 30.000 | 30.000 | + 60°00'00" |

### 3. Curve Checks

Copan will check the following before calculating curve parameters:

- The inward and outward radial distances of a curve are equal.
- The inward radial direction of a beginning-of-tangential-curve is perpendicular to the preceding line or parallel to the preceding outward radial.
- The outward radial direction to an end-of-tangential-curve is perpendicular to the following line or parallel to the following inward radial.

### 4. Curve Adjustments

If a traverse that contains curves is adjusted (in Map Trav), its curves are affected as follows:

- For non-tangential curves, the curve chords are treated as traverse legs for the adjustment, and then the curve centres are recomputed based on the original curve radii and new curve begin-end points.
- For tangential-curves, the external tangents are treated as traverse legs for the adjustment. The curve begin-end points are recomputed based on the new curve intersection points and external tangents, and then the curve centres are recomputed as above.

updated: 21-Sep-2010

updated: 21-Sep-2010

***Copan for Windows***

# Point Renumbering or Replacement

You may *renumber* any existing Copan coordfile points. You may also renumber new points being added to a coordfile or *replace* existing points when the numbers match.

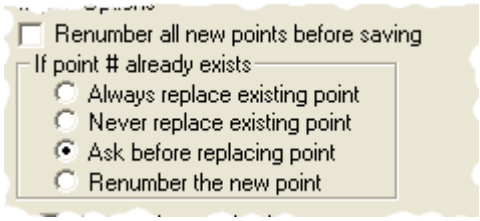
## Renumbering Existing Coordfile Points

To renumber current coordfile points on an *individual* basis, see To Renumber a Point Record in § Point Records.

To delete, renumber, or edit a *group* of existing coordfile points, see § Bulk Point Editing.

## Renumbering New Points

When adding batches of *new* points to a coordfile — such as in Import Ascii, Field Data Process, or Map Traverse Adjustment — what if some of your incoming points are not numbered the way you wish them to be or already exist in the coordfile? You can instruct Copan what to do *before* clicking **Import** or **Save points**.



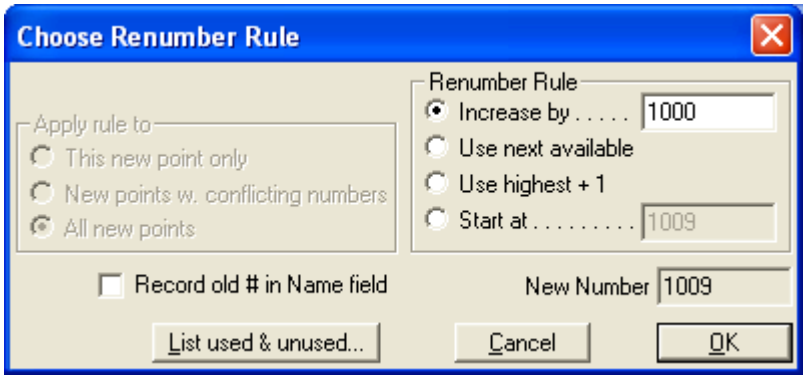

### To automatically renumber every incoming point

- Check **Renumber all new points**.

### When incoming points have numbers matching coordfile points

- Check **Always replace existing point** if you know that new point values should replace existing ones.
- Check **Never replace existing point** if you know that new points should be *ignored* and existing ones should remain unchanged.
- Check **Ask before replacing point** in order to decide what to do on a point-by-point basis.
- Check **Renumber the new point** if you know that new points should be *renumbered* and existing ones should remain unchanged.

## Rules for renumbering




You also have a selection of possible rules for the new number:

- Check **Increase by . . .** for each new point number to be increased by a constant value.
- Check **Use next available** for each new point number to be replaced by the next unused point number.
- Check **Use highest + 1** for the new point numbers to begin after the current highest point.
- Check **Start at . . .** for the new point numbers to begin at a specified value.

If you want each new point to remember what its original number was before being renumbered, check **Record old # in Name field**.

updated: 21-Sep-2010

***Copan for Windows***

# Part V: Appendix

- Installing or Updating
- Uninstalling
- Running via CoordFile Icon
- Bugs and Other Problems
- Support
- News

updated: 21-Sep-2010

updated: 16-Dec-2011

**Copan *Lite***

# Installing or Updating

Contents



Note that the *first* time you set up Copan on any computer, you are said to be ***Installing*** it, and any time you put a *different* version of Copan on the *same* machine, you are said to be ***Updating*** it (in which case, you can skip steps 1 and 4 below).

## How to Install Copan

Please follow these instructions carefully. (Steps 1 and 2 may done in reverse.) If you're having trouble, contact support@underhill.ca with the *exact* details of what works and what doesn't.

### 1. Request a password (or software key)

If you haven't already done so, request a password from www.underhill.ca/software/copan-windows-software-key.

An eight character password (e.g., "secret39") will be emailed to you. You should receive it within a few seconds. If you don't, maybe you didn't enter your correct email address or maybe automatic mail from our mail server to your mail server is blocked.

### 2. Download the installation software

If you haven't already done so, download the latest version of the installer: Visit www.underhill.ca/software/copan-windows-download and download the install package, `CopanLite.setup.exe`, to (anywhere on) your computer.

If your download tool happens to strip off the ".exe" extension from the `Copan.setup.exe` filename, be sure to add it back before you run it. (You may need to change the “show file extensions” setting in your folder browser.)

### 3. Run the installation software

Open or run the `CopanLite.setup.exe` file and follow the simple setup instructions. The installer will place various files in your *program folder* (usually `C:\Program Files\Copan 11.11\`), and will create some shortcuts. If you know what you're doing, during the installation process, you may choose to install Copan in an alternative location or to turn off certain installation options. (If you change your mind, you can always uninstall then reinstall.)

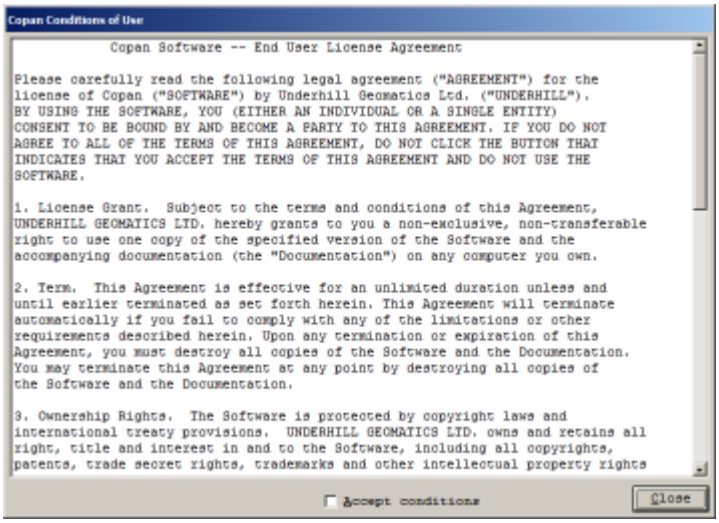
The resulting folder and file configuration should look something like this:

| Folder | Contents |
|---|---|
| `C:\Program Files\Copan 11.11\` | `User Manual` file folder.<br>`User Manual.pdf` adobe document.<br>`Copan.exe` application.<br>`unins000.exe` application.<br>`dd_alloc.dll`, `dockwnd.dll`, `msvcirtd.dll`, `msvcp60d.dll`, and `msvcrtd.dll` application extensions.<br>`CopanKey.bin` file.<br>`EULA.txt` text document. |
| `C:\Program Files\Copan 11.11\User Manual\` | Various other file folders.<br>Various `.html` documents.<br>Various other files. |

The installer may also place various files in your Documents (or My Documents) folder (see § File Locations).

### 4. Initialise Copan

The *first* time you run Copan, you'll be presented with Conditions of Use which you must accept.




After accepting the conditions of use, enter the eight character password that was included in the email.


Be careful to use it exactly as is, without the enclosing double-quotes. (Copy and paste works best.) If the password is not entered correctly you'll get an Invalid Password message.


If you get that message and you're sure that you're entering it correctly, you probably have an out-of-date version of Copan, or at least a mismatch between your Copan and your key. Try re-downloading it from us and re-installing it (steps 1 and 2 above).

## How to Update Copan

The procedure for updating Copan is the same as for installing, except for the following:

- Unless you are installing on a new computer (i.e., a machine that has not had another version of Copan installed), then you do *not* need to request another password (step 1 above) and you do *not* need to re-initialise Copan (step 4 above).
- During setup, you'll notice the program folder (the place where Copan gets installed), `C:\Program Files\Copan 11.11\`, will be named differently for each version. This allows you to fall back to an earlier version should you have problems with the new one.

### Moving Auxiliary Files

Prior to version 10.08, the default location for the Auxiliary Files folder was the Program Folder, which usually changed with each installation, and the program installer would always suggest that you move your auxiliary files from an earlier program folder to the latest one. Since version 10.08, however, the default location for the Auxiliary Files folder is now in a more stable location, the Initial Project Folder. If you haven't yet done so, consider moving your auxiliary files, one last time, to a more stable location.

updated: 16-Dec-2011

updated: 21-Sep-2010

*Copan for Windows*

# Uninstalling

The following four scenarios for removing Copan refer to a *Copan program folder*. That is where Copan is actually installed and is usually in `C:\Program Files\Copan`.

## How to Uninstall the Only Existing Version

If you only have one version of the Copan, do any *one* of the following:

- Select **Start** | **All Programs** | **Copan** | **Uninstall**.
- Select **Start** | **Control Panel**, open **Add or Remove Programs**, find **Copan** from the list, and click **Change/Remove**.
- Open the Copan program folder and run the **unins000** application.

## How to Uninstall a Previous Version but Keep the Current Version

If you're happy with the latest version of the Copan, and you wish to remove a previous version, you only need to do one thing:

- Manually delete the previous version of the Copan program folder.

If you accidently used Uninstall (as described above), you'll need to

- either reinstall (see § Installing or Updating) the latest one,
- or open the latest Copan program folder, right-click the Copan application icon and choose **Send To** then **Desktop**.

## How to Uninstall the Current Version but Keep a Previous Version

If you're *unhappy* with the latest version of the Copan, and you wish to revert a previous version, do three things:

1. Optionally, let us know what's wrong with the new version.
2. Manually delete the current version of the Copan program folder.
3. Either
    - reinstall (see § Installing or Updating) the old one, or
    - for each Copan-related shortcut (on Quick Launch, Desktop, Start Menu, etc)
        i. right-click the icon and choose **Properties**
        ii. edit the **Target** box so the current Copan program folder name is replaced by the old one
        iii. if the **Start in** box also has the Copan program folder name, edit that too
        iv. click **OK**.

## How to Uninstall All Versions

1. Uninstall the latest version of Copan (as described in How to Uninstall the Only Existing Version above).
2. Manually delete each of the previous versions of the Copan program folder.

updated: 21-Sep-2010

updated: 18-Jun-2009

***Copan for Windows***

# Running via CoordFile Icon

CoordFiles (.PTS files) are automatically associated with the Copan program. That is, in Windows Explorer, a .PTS file is labelled as a Copan CoordFile and will open inside Copan if it is opened. (To open a file, double-click its icon or right-click it then choose Open.)

If that file association is not happening, you can reinstall Copan (version 08.04 or later), ensuring that the **Associate coordinate files (*.pts) with Copan** task is checked.

updated: 18-Jun-2009

updated: 17-Feb-2011

***Copan for Windows***

# Bugs and Other Problems

Copan is a work in progress and there may still be some issues. These may range from mere annoyances to outright crashes-without-warnings. Please be patient and gently report the specifics of any such issues you find.

## Annoyances

The mildest problem is when Copan behaves in a way that is, whether on purpose or by default, part of its design. What may be useful or acceptable behavior to some may be inconvenient to you. Please let us know if you think it can be improved.

## Error Messages

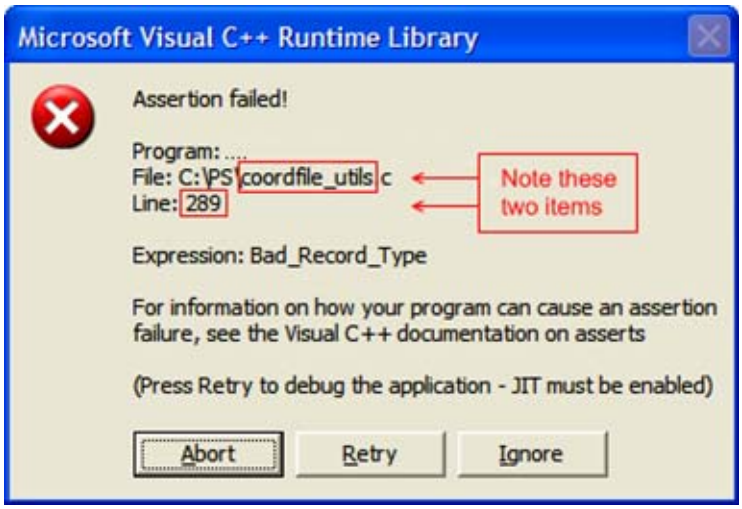
Sometimes you, the user, make mistakes — whether you tried to do something inappropriate for the current situation or you entered some invalid data.

There's a large set of error messages and warnings that Copan can give, and we've tried to make them clear and polite. If you have a problem with any message, please let us know. support@underhill.ca

## Assertion Failed!

Occasionally, a **Microsoft Visual C++ Runtime Library** pop-up appears saying "**Assertion failed!**" It means that something that shouldn't have happened just has. Usually, the information provided can be helpful in pinpointing the problem. Please note and send us this potentially useful data:

- File: C:\PS***source-file-name***.c
- Line: ***line-number***



The Expression that appears may provide a hint as to what's wrong but it's not required.

After noting the file name and line number, try **Ignore**. Sometimes, after you choose Ignore, Copan carries on as if there were nothing wrong, so it's worth trying. Other times it will crash. Either way, please let us know what you did just before the message appeared, what you tried, etc.

## Bugs

A bug in a program is something that doesn't work the way it's supposed to, such as an incorrect calculation result. A small number of bugs in Copan are known and documented in the relevant user manual page and some are not. If you encounter a bug, please check if it is already documented. If it's not, please note the details and send them to support@underhill.ca. Please note the software version and release date (see § Software Version) in all communications.

## Crashes

A crash is when a program shut's down when it's not supposed to, *without any warning*. If you encounter a crash, please check if it is already documented or not. If not, please note the circumstances that led to the crash and keep us informed.

You can help by sending us the relevant *program trace* file — located in the User Settings folder (see § File Locations). Send it to support@underhill.ca.

Since version 11.01, four generations of trace files are kept (`program.trace.0.txt` to `program.trace.3.txt`). The dates and times of the files should indicate which ones are relevant. If in doubt, send them all.

Prior to version 11.01, only one `program.trace.txt` was ever kept, so you were required to send us the file *before starting Copan again*.

updated: 17-Feb-2011

updated: 18-Jun-2009

 ***Copan for Windows***

# Support

Help on using or printing this manual and on basic use of Copan can be found in § Getting Started.

Support for registered users of the Free version of Copan *may* be available via email, depending on demand:

- support@underhill.ca

Suggestions for improvement or reports of bugs are always welcome from all users.

Please note the software version and release date (see § Software Version) in all communications.

updated: 18-Jun-2009

***Copan for Windows***

# News

We improve Copan quite frequently and we announce such updates in our irregular newsletter *Copan News*. In it we also provide occasional tips on using Copan or solicit feedback from users.

Contact us for a subscription: support@underhill.ca. If you requested a key, you subscribed automatically. To unsubscribe: support@underhill.ca.

updated: 21-Sep-2010

# Copan News

### Mar-2011

**Copan** *Lite* for Windows version 11.03, release 2011.03.21, was released at 2pm on 25-Mar-2011, Vancouver, BC.

While there are improvements over version 11.01, the main reason for this software release and news edition is to establish a base release for the official beta test version of Copan CAD. And to do a little bragging...

## 10 000 “Copanians”!

There are now 10 000 Copan users, world-wide!

Among the three different platform versions, the breakdown is Windows 74%, PocketPC 24%, and PalmOS 2%.

The mobile versions of Copan have not been updated for a long time and, with so many newer and more popular hand-held devices coming out, we're unsure where or when the next mobile Copan development will be.

## Copan CAD is still coming

Last time we gave a sneak preview of “Copan Pro” / “Copan CAD”. Our offer of the chance to try it out, to become a beta tester of the pre-release version, is now closed. Thank you to all who filled in the simple questionnaire — we'll be in touch very shortly about getting your test copy.

You can expect the first commercial release of Copan CAD in about six months.

## “Scale extended-code sizes” added to Field family

In the Field family of modules, you can now automatically scale all the extended-code sizes in side-shot observations by the current units factor. For example, if your coordinates are in metres and your field data are in feet, you can now convert, via a UF= 0.3048, your manhole depths and culvert diameters from hundredths of feet to hundredths of metres (along with all the usual conversions of SD, HI, HS from ft to m).

## Other small improvements to Field family

### Load Data

- The chances of a crash happening during the Loading of data from any total station file, particularly Leica GSI data, has been reduced.
- When Loading TDS RAW data, the signal (or target) heights are now created for *each* shot (not just the first shot from a set-up).

### List Data

- If present, a Units Factor is now listed correctly (not as 0.00).
- Vertical circle readings are now listed before any VCC, if present, is applied.

### Calculate

- Correction factors (such as SF, UF, DF) now retain their explicitly set existing values, even if empty values (such as in "UF = ") were accidently subsequently assigned.

### List Calcs

- Side shots are now listed after a traverse has been adjusted, even if the traverse was first *not* adjusted.

## Small improvements to Import Ascii points

- There's now better reporting of any problem records found in an ascii import file.
- Any such problem records found are now no longer added to (and corrupting) the coordfile as bogus points.

## Small bug in curve display

We just discovered a curve display buglet that's been present since release 2010.12:
If you calculate any curves in Map Traverse or Map Check and zoom in a long way, at the end of the displayed arc, you may notice the arc is offset from the actual end-point by a mm or so. The survey calculations and curve results are still correct, only the small details of the display are wrong.

Mar-2011

# Copan News

## May-2011

**Copan** *Lite* for Windows version 11.05, release 2011.05.26, was released at 2pm on 30-May-2011, Vancouver, BC.

While there are *some* improvements over version 11.03, the main reason for this software release and news edition is to update Copan Pro for the beta testers. And to do a little more bragging...

## 200 Countries!

Copan is now used in 200 countries. We recently had a new user from Bermuda!

## More small improvements to Field family

### Load Data

The chance of a crash on Loading data files, especially GSI files, has been further reduced.

### Save Points

The "renumber/replace" option is now always reset to "Ask before saving" for new Field files.

### Export Data

Did you know you can Export field data to GeoLab IOB format files? The "missing foresight DIRs in DSETs" bug is now fixed. Actually, Export now outputs (1-line) ANGLs instead of (3-line) DSETs.

Also, you now have some control over IOB sigma records and horizontal angle output. (See the new **Export to IOB File** section in the User Manual.)

## Small improvements to Map Trav

If you ran a multiple-traverse file, starting from a specified point or label, and then ran a new, single-traverse file not containing that previously specified start point/label, Copan would complain. This is now fixed.

The "renumber/replace" option is now always reset to "Ask before saving" for new Map Trav files.

May-2011

# Copan News

### Dec-2011

Copan *Lite* version 11.11, release 2011.12.15, was released at 2pm on 22-Dec-2011, Vancouver, BC.

Note: It has *no* functional improvements over version 11.05 – only a name change and some updated links to our new web site. You do not need to update your software.

## Name change and new editions

For a long time, we provided three Copan packages and they were named for the different platforms that they ran on: "Copan for Windows", "Copan for Pocket PC" and "Copan for Palm OS". There were also two editions of Windows Copan: a free one for the public and a more advanced one for our internal (UGL) use.

Now, more advanced editions of Copan are available for the public, see below, and as we now have multiple editions, a naming change is order:

**"Copan for Windows"** (free edition) is **now officially called "Copan *Lite*".** It still costs nothing and has no usage restrictions!

## Copan Pro now available

A new, advanced edition of Copan—called "Copan *Pro*"— is now available for the public. It has all of the functionality of Copan *Lite* (formerly Copan for Windows) *plus* numerous **drawing tools**, *and* it **processes Leica digital level data**.

It costs only $650 Canadian, but is available for a free trial. See www.underhill.ca/software/copan-pro for more information.

## Copan Pro+DWG also available

An even more advanced edition of Copan—called "Copan *Pro+DWG*"— is also available for the public. It has all of the functionality of Copan *Pro* (ie, COGO, drafting, etc) *plus* it **exports Copan coordfile points directly to AutoCAD DWG files** *and* it **exports Copan drawings to AutoCAD DWG files**.

It costs only $800 Canadian, but is available for a free trial. See www.underhill.ca/software/copan-pro-dwg for more information.

## New web site

If you've not been to our company web site since June 2011, you might like to visit. It has an enhanced design, more surveying services described, many more survey projects shown, a much-expanded history section, many more photos, and of course, a newer and expanded surveyor/engineer software section. Got to www.underhill.ca and explore.

Dec-2011