

Golden software

记录一些 Holz 自己写的或整理的 [Golden Software](#) 系列软件相关的文章。

Golden Software 系列软件包括 Surfer、Grapher、Digger、Mapviewer 和 Strater，几乎是地质工作者必备的工具软件，可用于各种数据分析、数据可视化和专题图制作。

转载请注明出处。

Didger 3 教程

Holz 将以 Didger 3 英文版的依据写一些教程。

Didger 3 入门教程

这个入门教程向大伙介绍 Didger 的部分功能，当然是最基本的那些。我没有数字化仪、没有 GPS 之类，所以许多高级的 Didger 功能我都从来用不上。所以这个教程是一个数字化的基础教程，您看完这个教程，应当能够使用 Didger 建立自己的工程了。

这个教程是很浅薄的，学好点 E 文准备看 Golden Software 给您提供的用户手册罢，据说您的问题通常都能在那上面找到。当然您还有问题就去骚扰他们的技术支持罢，不用问我，我的水平差着呢~~~

下面说说课程安排：

[第一课](#) - 学习如何校准一个光栅图像，不要问我关于如何校准数字化仪、GPS 之类的，我没有，我不知道。

[第二课](#) - 学习如何数字化一个点，如何设置点的属性和数字化线条。

[第三课](#) - 学习如何保存您的 Didger 工程并且将数据输出以供其他程序使用。

第四课 - 学习如何建立多边形，如何从多段线建立多边形等。(没啥重要的，完全可以忽略不看)

第五课 - 学习光栅图像的处理、输出和再次利用等。(也没啥重要的，也完全可以忽略不看)

这个入门教程其实比 Didger 3 本身提供的要简陋，因此如果您的 E 文比较好，不用在此流连，直接看帮助罢。

第一课 – 校准(文字版)

当你要数字化之前，首先要做的是什么呢？校准！

无论你使用的是扫描的光栅图片还是放在数字化仪上的图纸，首先都要选 3-256 个点进行校准。校准点不能都在一条直线上，应该是均匀分布在图纸上的。

我们来看看如果对一个光栅图片进行校准。

为了校准一个光栅图片，首先要输入这个图片：

1. 启动 Didger。Didger 启动时默认建立一个空白的工程，如果您的 Didger 已经在运行了，那么选择 **File | New | Blank Project** 新建一个空白工程。
2. 单击 **File | Import Bitmap | Into Raster Project** 表示要输入一个光栅图片到一个光栅工程中。
3. 在弹出的 **Import Raster File** 对话框中，浏览到 Didger 文件夹下的 **Samples** 文件夹，单击 **TUTORIAL MAP 1.JPG** 文件。如果您安装默认选项安装，那么 **Samples** 文件夹应该位于 `\program files\golden software\didger3`。
4. 单击对话框中的 **OK** 开始输入光栅图。

现在就可以进行图片校准了：

1. 选择 **Image | Calibrate Image** 菜单命令。
2. 在弹出的 **Initial Calibration Settings**（初始校准设置）对话框中，选择 **Cartesian Coordinates**（笛卡尔坐标）。**X Axis Type** 和 **Y Axis Type** 应该设置为 **Linear**，**Calibration Units**（校准单位）应该设置为（不指定）。
3. 单击 **Next** 按钮转到 **Raster Calibration**，这里要为校准点输入坐标。
4. 用鼠标单击 **Point ID** 列的第一行并输入 **CP1**，这就是第一个校准点的 ID。
5. 单击 **World X** 单元格，输入 **CP1** 的 X 坐标 (1200)。按 **TAB** 键移动到 **World Y** (也可用鼠标单击) 然后输入 **CP1** 的 Y 坐标 (4000)
6. 将鼠标移动到地图的左下角的那个 **CP1** 位置上。如果你需要放大地图以更精确地数字化，按住 **SHIFT** 键然后在地图上单击即可；按住 **CTRL** 键并在地图上单击是缩小。当放大显示时还可以通过滚动栏移动查看。单击鼠标左键得到该点的 **Source X** 和 **Source Y** 坐标。
7. 单击 **Add Point** 按钮
8. 重复 4-7 步骤就可以完成四个校准点，如果你点错地方，还可以选中该行然后重新数字化该点。

Point ID	World X	World Y
CP1	1200	4000
CP2	8500	14500
CP3	16300	7100
CP4	19600	12600

9. 单击 Next 按钮来到 RMS Calibration Settings 对话框。这里有 RMS 报告，是关于前面的校准点的测量精度的。如果您发现精度太低，误差太大，可以选择重新校准。
10. 单击 Next 按钮来到 Calibration Settings 对话框。我们将 Vertex Tolerance 和 Snap Tolerance 设置为 200
11. 单击 Finish 按钮就完成了校准工作。

第二课 – 数字化(文字版)

作为教程使用的 Tutorial Map 1 文件中包含一些点和多段线。

当您完成了 [第一课](#) 中的校准工作后，您就可以进行数字化点和线的工作了。

数字化点

本教程中使用的光栅图片包含一系列的点。在 Digger 中，点是通过符号类表现的，因此您可以指定符号的类型、大小、颜色以及关联到每个点的四个 ID。这些 ID 或者 ID 组合可以作为点的标注使用。

在本次教程中，出现了两种类型的点。第一种类型是以 MW- 开头的技术孔，包含地下水位数据；另外一种 is 鉴别孔。

首先我们会数字化技术孔，在数字化的时候，可以输入孔号和地下水位数据，也可以将这些数据作为地图标注。

自动增加 ID

为什么将点分为两种不同的类型呢？因为技术孔的编号是序列化的，因此可以使用一个序列化的 ID，在数字化的时候可以让程序自动增加。

我们来看看如何操作：

1. 选择 Digitize | Point 命令或者单击工具栏上的按钮，弹出 Digitize Points 对话框。
2. 单击 Clear Data 按钮清除所有设置。
3. 选择 Auto Increment Primary ID 复选框。
4. 在 Starting Value 字段中，输入 1，表示从 1 开始。
5. 在 Ending Value 字段中，输入 7，表示最大序列号为 7。
6. 在 Increment Value 字段输入 1，表示每次增加 1。
7. 在 ID Prefix 字段中输入 MW- 表示所有 ID 的前缀是 MW-，这个设置将应用到所有的 Primary ID 中。
8. 选中 Enter Data After Creating Object 复选框，这样每数字化一个点就提示您输入必要的数据。

在数字化点之前我们还要设置符号的属性和标注的属性，因此暂时不要点击 OK 按钮。

设置符号属性

单击 Digitize Points 对话框的 Symbol Properties 标签，这里可以设置符号类型、颜色和大小。

首先选择 Symbol Set 为 GSI Default Symbols。在 Symbol Set 名称右边点击下拉箭头可以选择列表中的符号集。

选择好 GSI Default Symbols 后，选择技术孔的符号：

1. 单击符号右边的下拉箭头，找到技术孔所使用的符号。
2. 单击该符号。

接下来设置符号的颜色：

1. 单击 Color 右边的按钮打开颜色调色板。
2. 单击蓝色，调色板自动关闭并且选中了蓝色作为符号颜色。

最后设置符号的大小：

1. 选中 Size 编辑框。
2. 输入 0.20 作为符号的大小。

至此我们设置了符号的各项属性，接下来要设置标注属性，所以暂时不要点击 OK 按钮。

设置标注属性

单击 Label Layout 标签设置标注属性。

1. 在 [自动增加 ID](#) 部分我们设置了孔号数据保存在 Primary ID 中。在 Available Fields 中选择 Primary ID 并单击 Add 按钮。
2. 在数字化的时候我们会输入地下水位数据，并将这些数据保存在 Secondary ID 字段。因此单击 Available Fields 列表的下拉箭头，选择 Secondary ID，然后单击 Add 按钮。
3. 在 Label Field Name 列中单击 Primary ID，然后单击右边的 Font 按钮，在弹出的 Font Attributes 对话框中，选择 Arial 字体并设置大小为 10。单击 OK 返回 Digitize Points 对话框。
4. Secondary ID 的字体设置与 Primary ID 一样。
5. 默认的情况下，标注是居中显示的，由于我们有两个标注，所以要改变标注的显示位置。在 Label Field Name 列中单击 Primary ID，然后单击 Label Position 的上箭头。

头，使得 Primary ID 在符号的上方显示。然后在 Label Field Name 列中单击 Secondary ID，然后单击 Label Position 的下箭头，使得 Secondary ID 也就是地下水位在符号的下方显示。

数字化技术孔

在前面我们设置了 [自动增加 ID](#)、[符号属性](#)、[标注布局](#)，现在就可以点击 Digitize Points 对话框中的 OK 按钮进行数字化了。

这时光标会变成一个准星样式。

1. 移动鼠标使得准星对准 MW-1 并单击鼠标左键。
2. 此时弹出 Enter Object Data 对话框，我们发现孔号已经自动出现在 Primary ID 字段。
3. 用鼠标单击 Secondary ID 字段然后输入该孔的地下水位数据 (88.36)。
4. 单击 OK 按钮，第一个孔数字化完毕。
5. 在地图上找到 MW-2，重复前面的数字化操作。
6. 依次数字化到 MW-7。

数字化一个点是一个很简单过程。Isn't it?

数字化鉴别孔

本教程使用的光栅图片 Tutorial Map 1 中那些鉴别孔是无序的点，没有序列化的 ID，但他们都有相同的符号属性。对于这样的点，我们可以一个一个使用 Digitize | Point 菜单命令进行数字化，也可以一次性数字化全部，不过与前面那些有序的点稍微不同。

1. 选择 Digitize | Point 菜单命令或者单击工具栏上的符号按钮，弹出 Digitize Points 对话框。
2. 在 Data Attributes 标签页，单击 Clear Data 按钮清除前面设置的参数。
3. 选中 Enter Data After Creating Object 和 Create Several Objects 复选框。
4. 单击 Symbol Properties 标签选择符号类型、颜色、大小。设置大小为 0.5 cm。
5. 单击 Label Layout 标签。我们将发现有两个标注。
6. 在 Label Field Name 选中 Secondary ID，单击 Remove 按钮。
7. 单击 OK 按钮则光标变成准星样式，移动鼠标瞄准要数字化的点，单击鼠标左键。
8. 在弹出的 Enter Object Data 对话框中输入孔号到 Primary ID 字段并点 OK。
9. 继续点击其他要数字化的点并输入相应的数据。当所有的点数字化完毕，按 ESC 键退出数字化模式。

数字化等值线

地图上的多段线可以用来表现道路、等值线、水系、工区范围等等。多段线由一个或多个首尾相连的直线段组合而成,那些多段线的开始和结束点我们称之为节点;而中间那些线段的端点称之为顶点。

在本教程中,我们要数字化的多段线是地下水位等高线。

多段线的数字化可以通过点击线条上的点或者沿着线条进行追踪的方式进行。因为我们要数字化的是光滑的曲线,所以追踪方式非常适合我们。

先使用缩放工具使得工作窗口中可以看见要数字化的线条。

1. 选择 **Digitize | Polyline** 菜单命令或者点击工具栏上的折线工具按钮,显示 **Digitize Polylines** 对话框。
2. 单击 **Data Attributes** 标签,勾选 **Enter Data After Creating Object** 和 **Create Several Objects** 复选框。
3. 单击 **Line Properties** 标签,设置等值线条的属性。
4. 单击 **Label Layout** 标签,设置等值线条标注属性。
 - 在 **Available Fields** 的下拉列表中选择 **Primary ID**,单击右边的 **Add** 按钮。
 - 单击 **Font** 按钮将字体设置为 **Arial**,大小设置为 **10**,单击 **OK** 返回 **Label Layout** 页面。
 - 在 **Label Position** 组选择 **Position Along Line** 这个选项。
 - 设置 **Label Alignment On** 为 **On** 并设置 **Label Position Along** 为 **Middle**。
5. 单击确定按钮则光标变成准星,提示您进入了数字化模式。
6. 将准星移动到 **88.30** 等值线的一个端点上,按住鼠标左键并沿着等值线移动鼠标,您会看见一条多段线绘制在图形窗口。
7. 按 **ENTER** 键结束该线条的数字化。
8. 在弹出的 **Enter Object Data** 对话框中,单击 **Primary ID** 字段并输入 **88.30**,即您刚数字化完的线条的等值线等级。
9. 单击 **Enter Object Data** 对话框的确定按钮继续数字化下一条等值线,直到所有等值线数字化完毕。
10. 按 **ESC** 键结束数字化,退出数字化模式。

提示:在数字化过程中,如果发现错误可以点鼠标右键撤销,甚至随时可以按 **ESC** 中断数字化然后重头再来。

第三课 – 保存和输出(文字版)

当我们完成了对光栅图 **Tutorial Map 1** 的数字化之后,就可以将他保存为一个 **Didger** 工程文件 **[.PJT]**,也可以按要求输出数据以供其他应用程序使用。

如果保存为 **Didger** 工程文件 **[.PJT]**,就能够保存所有的校准信息、数字化的 **XY** 坐标、投影信息、对象属性、**ID** 等。虽然 **[.PJT]** 文件并不能直接为其他应用程序调用,但由于该类型的文件保存了所有必要的信息,并且可以为 **Didger**

再次利用，他还允许您添加额外的数据、再次编辑已有的数据、输出需要的数据为其他应用程序使用等。

保存为工程文件的过程非常简单，选择 File | Save 菜单命令，在弹出的 Save As 对话框中，输入文件名并回车即可。

默认情况下，文件后缀 PJT 会自动添加。

Didger 的输出有许多选项，我们以输出点数据到一个 dat 文件为例，简单说明。

1. 选择 Edit | Criteria Select 菜单命令，因为工程中有点、线等对象，而我们只需要输出点数据，因此要使用标准选择来筛选。
2. 在 Criteria Select 对话框中，在 Selection Type 字段选择 Object Type，在 Object Type 字段选择 Point。Condition 选项保持默认的 None。然后单击 Select 按钮，我们会在主窗口的左下方的状态栏上发现已经选定了 11 个点。此时单击 Close 按钮退出标准选择。
3. 选择 File | Export 菜单命令。
4. 在弹出的 Export 对话框中选中 Selected objects only 复选框。
5. 选择 Golden Software Data (*.dat) 文件格式。
6. 输入一个文件名，不用输入后缀，因为后缀会自动添加。
7. 单击 Save 按钮来到 Export Options 对话，这里可以设置 [.DAT] 文件的一些选项，我们勾选 Quote text 和 Tabs as separators 这两个选项即可。
8. 单击 OK 按钮就建立了一个新的 Golden Software Data [.DAT] 文件，他是一个 ASCII 文本文件，用记事本就可以打开。

第四课 – 建立多边形

多边形是由若干线段收尾相连并且封闭组成的一个对象。在 Didger 3 中多边形类似于多段线，区别在于开始和结束的顶点坐标相同，由此形成一个封闭的对象。多边形可以使用填充图案。

Didger 3 建立多边形的方法有多种：

- 一个独立的多边形可以通过使用 Digitize | Polygon 命令建立，数字化的过程类似于对多段线的数字化，只不过当您完成的时候，开始和结束的顶点会重合。
- 还可以通过转换功能将一个多段线转换为多边形，这个命令在 Tools | Polyline to Polygon。首先选中要转换的多段线对象，然后选择 Polyline to Polygon 命令，则多段线的结束点会自动连接到开始点，由此形成一个封闭的多边形对象。
- 还可以使用 Tools | Create Polygons by IDs 或者 Tools | Create Polygons by Locator 命令来获得多边形对象。这两个命令使得我们可以建立那些有公共边的多边形，因为一般而言我们是很难数字化那些有公共边的多边形并保持这些公共边精确匹配的。Create Polygons by IDs 和 Create Polygons by Locator 命令正是为了这个目的设计的，使用这两个命令，我们只要对那些公共边数字化一次就可以了。

一些数字化技巧

- 1、对于那些比较长直的线段，点击线段的拐点就可以了，拐点之间会自动以直线连接。
- 2、曲线段可用流线模式数字化，如果数字化一个光栅位图，那么可以按住鼠标左键并沿着曲线移动光标即可。
- 3、那些既有较长直线段也有弧段的线条，可以用组合的方法进行数字化。

如果数字化的一线段讲来要用于 Create Polygons by IDs 命令，那么它必需有 Primary Group - Left Side 和 Secondary Group - Right Side IDs 属性。这些 ID 就是用于定于该线段在所建立的多边形中属于哪一边。要判断哪个是“左”哪个是“右”，想象您站在多段线的开始顶点并且向着结束点的方向，则 left ID 就是在您左侧的多边形，right ID 就是您右侧的多边形。

数字化重合边界

在 Didger 中多段线的开始点和结束点可看作是一个节点，当我们要使用 Create Polygons by IDs 命令的时候，用于建立多边形的相邻多段线上的节点必需重合。

这一点我们可以通过校准时设置吸附公差来确保，当然我们也可以通过 File | Project Settings | Tolerance Settings 来设置。当两个不同的多段线节点位于吸附公差限定的范围内，表示这两个节点重合（自动吸附），并使用相同的 XY 坐标。

第一课 - 校准(演示版)

本演示使用 Flash 格式，代码只考虑了 Firefox，所以使用 IE 浏览器可能会显示不了。

实在不行就 [下载](#) 来看罢，我不会为IE编写javascript脚本。

第二课 - 数字化(演示版)

为了保证文件大小，分为三个部分。

第一部分：数字化序列点

代码只考虑了 Firefox，在 IE 中可能不能显示，您可以 [下载](#) 到本地观看。

第二部分：数字化无序点

代码只考虑 Firefox，使用 IE 浏览器可能不能显示，不过可以 [下载](#) 到本地观看。

第三部分：数字化等值线

代码只考虑 [Firefox](#)，使用 IE 可能无法显示，您可以 [下载](#) 到本地观看。

Didger 3 的介绍

一句话介绍：Didger 3 是一个拥有所有高级数字化特性的多功能程序。

复杂一点的：Didger 3 是一个高精度的数字化程序，Didger 能够精确地将纸介质的地图、图表、航空照片、扫描的光栅图像、输入的矢量文件和 GeoTIFF 图像中的点、线、面转换为各种不同的数字格式，从而为其他软件提供服务，Didger 可以快速、精确和有效地进行数字化处理。

如果您还不过瘾，还希望看诸如“一个程序，无数用途”、“超级程序 - Didger”之类的牛皮广告，完全可以访问 [Didger 3 的产品介绍](#)。

Grapher 5 的一些常见问题

您在用 [Golden Software Grapher 5](#) 吗？您在使用过程会遇到[许多问题](#)吗？你不喜欢看英文？那我这里整理了很少一部分常见问题，希望有用。什么？您早就用上 [Golden Software Grapher 6](#)？Grapher 5 已经过时？当我没说过。我麻，还没钱去买，Grapher 6 阿，那真是“幸福可望而不可及”

1. 现在 PDF 和 EPS 文件很流行阿，Grapher 5 能生成这样的文件吗？可以，使用 **FILE -> EXPORT** 命令，可以选择 PDF 和 EPS 格式，效果还不错。
2. 我没有 didger，可以用 Grapher 5 对位图进行数字化吗？当然可以，不过只支持二维线性坐标。方法是先输入光栅图片文件，用 File -> Import 命令，然后选中该图片，用 Graph -> Digitizing -> Assignment Coordinate 命令给图片指定坐标，然后就可以数字化了。
3. Grapher 5 支持中文吗？Grapher 5 并不能直接支持中文 以下为官方说明：
Grapher was designed to use English characters. Double-byte and 16-bit Unicode international characters are not supported in Golden Software products.
4. 那我如何在 Grapher 5 中输入中文？有许多方法在 Grapher 5 中输入中文，可以将中文变成图片再用插入的方式；可以插入 OLE 对象，只要该对象支持双字节编辑，就能输入中文，例如 Word；可以导入以前版本的文本对象，例如 Grapher 1.25/1.32 版本中可以输入中文，将该文本块导入即可重复利用；可以使用其他交换各式输入

中文，例如 [PlotCall](#) 格式，DXF 格式等；还可以用插入时间格式间接输入中文；以及等等。

5. 一张图内有 N 多的曲线，如何选中要修改的那条？ 用 Object Manager，单击要编辑的曲线名称。
6. 如何为每条曲线制定一个独特的名字？ 还是用 Object Manager，Rename Object 即可。
7. 每次都要设置坐标等等琐碎的属性，有没有偷懒的方式？ 有！存成 .GRT 文件。
8. 我画了一张图，但图形没有显示？ 用来绘图的数据一定是数值或日期类型，如果是文字，那就没了。
9. 有没有类似 Surfer 8 那样的标注编辑？ Grapher 5 有了这个功能了，先选中图形，用 Graph -> Move Plot label 命令。
10. 以前版本的自动化脚本还可以用吗？ 我很烦了。 其实可以直接运行，如果哪个地方不再可用，按照提示和帮助改了就好了。 估计问这个的都没有自己试验过~~~
11. 我的客户没有 Grapher 5，他怎么看我的图件？ 可以输出 PDF 等格式给客户，也可以让客户安装 Grapher 5 Demo。
12. 我有 Grapher 3、4，怎么看 Grapher 5 的图件？ 如果你研究 Grapher 5 的另存为对话框，就会发现可以存成 Grapher 3 4 的格式了。
13. 我还有更多的问题？ 要么去看 Golden Software 的网站，要么自己看帮助，不过我得回了

Surfer 6 入门

作者：Holz (hemengui [at] gmail [dot] com)

参考：Surfer 6 英文版教程

图片来自 Holz 汉化的 Surfer 6。

Surfer 6 是比较古老的版本，但看起来应用的还比较多，因为可以随意处理中文，并且据说有古老的工作站没有升级，只能用 Surfer 6。

这个入门文章是我多年前依据 Golden software 英文版指南写的，前几天无意中翻了出来，整理了一下，放这里充充门面。

Surfer 指南分为若干课程。在您至少学习了一遍 [第一课- 创建一个网格文件](#) 后，您可以在后续的课程中将它作为一个单独的样本。在第一课中创建的网格文件将在第二课和第三课中使用。

准备工作

理解如何使用 Surfer

Surfer 最常用的一个应用就是使用一个 XYZ 数据文件来创建一个等值线图或表面图。网格 菜单下的 数据 命令就是用来将一个 XYZ 数据文件转换为一个网

格文件[.GRD] 的。当您创建了一个网格文件后，就可以通过 等值线 命令来创建一个等值线图或通过 Surface 命令来生成一个表面图。

下图举例说明了 XYZ 数据文件、网格文件、等值线图和表面图之间的关系。



启动 Surfer

在您安装完 Surfer 后，在 Windows 的程序管理器中你会发现一个 Golden Software 组。

要启动 Surfer 程序，可以这样：

1. 点击 开始 菜单，鼠标移动到程序，在出现的分级菜单中选择 **Golden Software**，点击 **Surfer**。
2. 现在我们看到 **Surfer** 已经启动了，并且我们看到了一个空白的图形窗口，这是我們用来创建网格文件、等值线图、表面图或任何其他地图类型的工作区域。
3. 如果您希望 **Surfer** 充满整个屏幕，点击 **Surfer** 窗口右上角的最大化按钮。同样的道理您也可以将 **Surfer** 内的图形窗口最大化。

使用在线帮助

Surfer 帮助系统使你非常容易的获得任何菜单或对话框项目的信息，有几种方法获得帮助信息：

- 你可以从 帮助 菜单选择一个命令，例如 内容 命令显示帮助主题。
- 你可以点击工具栏上的 帮助 工具按钮，同样显示帮助主题。
- 当你需要一个指定命令的信息的时候，你可以按 **SHIFT+F1** 然后指针出现一个问号标记，当你选择一个菜单命令的时候，**Windows** 帮助系统被启动，你所选择的命

令的帮助信息被显示出来。举个例子，如果你需要查找关于 覆盖地图 命令的信息，按 **SHIFT+F1**，选择 地图 菜单下的 覆盖地图 命令，帮助系统将显示 覆盖地图 主题。

- 在一个对话框中按 **F1** 将显示该对话框相关的帮助主题，该对话框中所有的参数都包含在该会话中。举例来说，当 等值线图 对话框显示的时候，你可能需要 等值线等级 组中的 载入 按钮的信息，如果你按 **F1**，将显示 等值线图 主题，滚动该主题就可以找到 载入 按钮的信息。

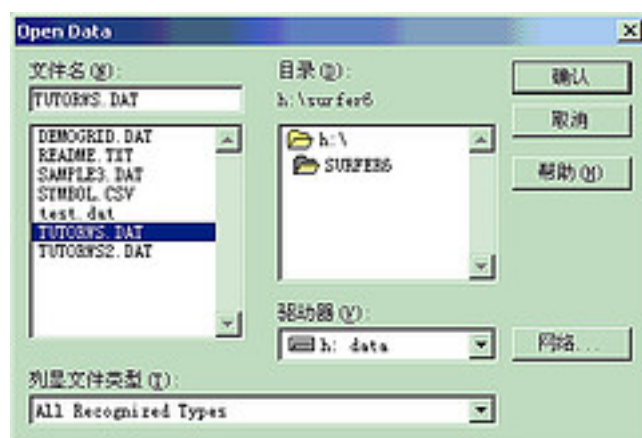
第一课 - 创建一个网格文件

第一课 - 创建一个网格文件

绘制一个等值线图或表面图就需要网格 [.GRD] 文件。网格文件通过 网格 菜单的数据 命令来创建， 数据 命令需要一个 XYZ 数据文件。Surfer 的安装已经附带了一些示范的数据文件，例如 TUTORWS.DAT，因此我们可以用它来示范如何通过一个XYZ 数据创建一个网格文件。

本课的练习使用数据文件 “TUTORWS.DAT”：

1. 从 网格 菜单选择 数据 命令，出现一个 Open Data 的对话框，如下图。在这个对话框中允许我们选择一个用来创建网格文件的 XYZ 数据文件。



在 Open Data 对话框中我们选择 Surfer 目录下的 “TUTORWS.DAT” 文件。

2. 在文件列表中用鼠标单击 “TUTORWS.DAT” 文件，我们看见文件名出现在文件名编辑框中，点击 确认 将出现 离散数据插值 对话框。当然，你也可以通过直接双

击文件名的方式来显示 离散数据插值 对话框。



离散数据插值 对话框允许我们指定创建网格文件的参数。

3. 离散数据插值 对话框允许我们控制网格化参数。花点时间来研究这个对话框中各个选项吧，这些时间是值得的。这一次我们什么都不用改变，我们直接用默认的参数设置来创建一个网格文件。

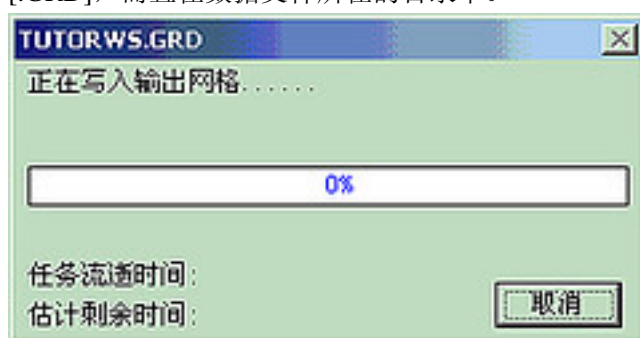
数据列 组用来指定数据文件中包含了X 和 Y 坐标和 Z 值的数据列。

网格尺寸参数 组用来设置网格文件中的 XY 网格限制、网格间距和网格行数(网格行数与网格间距是相关的)。

网格化方法 组指定网格化用的插值方法以及插值方法的控制参数。

输出网格文件 组指定创建的网格文件的路径和文件名。

4. 点 确认，就出现一个 状态 对话框，告诉你网格化的进度。当网格化完成后，网格文件“TUTORWS.GRD”将被创建，并且 Surfer 会通过声音提示我们。在使用默认设置的情况下，网格文件的名称是与数据文件相同的，只不过文件后缀变成了 [.GRD]，而且在数据文件所在的目录中。



状态 对话框指示网格化的进度。

第二课 - 创建一个等值线图

第二课 - 创建一个等值线图

使用 地图 菜单下的 等值线 命令可以创建一个基于网格[.GRD] 文件的等值线图。

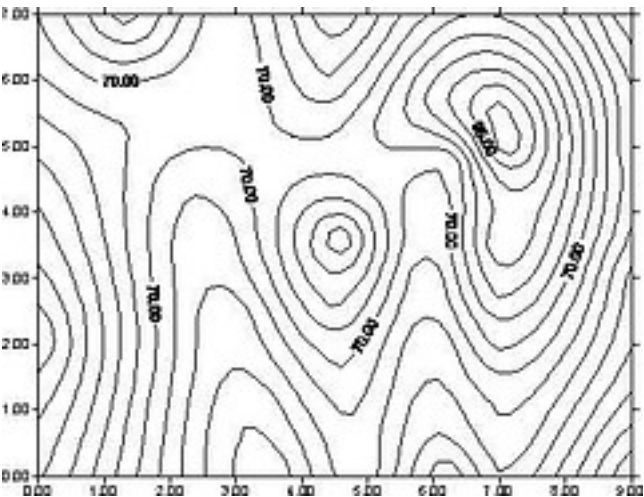
我们使用上一课创建的“TUTORWS.GRD” 文件生成一个等值线图：

1. 从 地图 菜单选择 等值线 命令，或者从工具栏点击等值线图工具 。 在出现的 Open Grid 对话框中，我们发现文件名编辑框中自动出现了我们刚创建的网格文件（TUTORWS.GRD）。直接点击 确认 就出现了 等值线图 对话框，如果你的网格文件是以前创建的，那么在文件列表中选中，并点击 确认。



等值线图 对话框允许我们控制等值线图的等值线线条和线条填充的显示效果。

2. 暂时我们能够接受 等值线图 对话框中默认参数设置下生成的等值线图，我们将在后面重新访问这个对话框。现在我们只要点击 确认，就生成了一个由“TUTORWS.GRD” 创建的等值线图。



由“TUTORWS.GRD”文件创建的等值线图

3. 如果你希望等值线图充满整个窗口，那么在 视图 菜单中选择 适应窗口 命令，我们马上就看见等值线图放大到充满了整个 图形 窗口。

改变等值线等级

在创建了一个等值线图之后，我们可以非常容易的修改这个等值线图的特征。举个例子来说，我们可能需要改变等值线图的等值线等级，以达到我们要求的显示要求。为了改变一个等值线图的任何特征，我们可以通过双击地图的方式打开 等值线图 对话框。

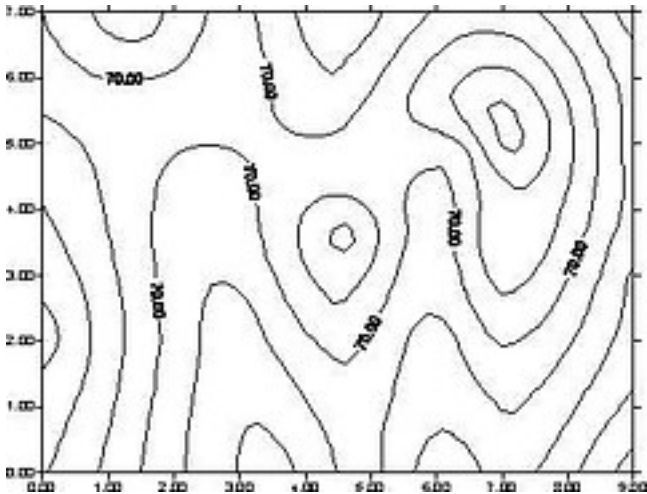
1. 将鼠标指针移动到等值线图范围内，双击鼠标左键，弹出 等值线图 对话框。
2. 在对话框中我们看见显示等值线等级和等值线条属性的 等值线等级 组，我们来示范任何让等值线等级从 $Z = 20$ 开始。如果你点击右边的滚动条，你会发现地图的最大等值线等级是 $Z = 100$ ，而等值线间距是 5。



等值线等级 对话框

3. 我们可以非常容易的改变等值线的范围和间距，点击 等级 按钮将显示 等值线等级 对话框，在这个对话框中显示了地图的最小和最大的等值线等级以及等值线的间距。
4. 双击 间距 编辑框就将该数值加亮，输入 10 表示改变本地图的等值线等级间距为 10。点击确认 就看见 等值线等级 列表已经根据我们的改变而更新了。

5. 点击 确认 关闭 等值线图 对话框，我们看见工作区中地图自动刷新了并且按照新的等值线等级重新绘制了一次。



等值线图按照等值线间距为 10 的设置重新绘制了。

改变等值线条属性

在 等值线图 对话框的 等值线等级 组中，我们还可以直接双击任意元素来单独修改一个元素的参数。例如我们可以双击一个 Z 值来修改某个等级的 Z 值，也可以双击一个线条样本来修改某个等级的线条属性，通过这种方式我们就可以绘制出多姿多彩的等值线图了。

你可以双击一个填充样本来改变某个等级的填充属性。

你也可以双击这些来改变是否在等值线上加标注和影线。

等级 (L)	线条 (L)	填充 (F)	标注 (A)	影线 (S)
20			Yes	No
25			No	No
30			No	No
35			No	No
40			No	No
45			Yes	No
50			No	No
55			No	No
60			No	No
65			No	No

你可以双击一个等级值为该等级输入新的 Z 等级值。

你可以双击线条样本来改变某个等级的线条属性。

删除 (D)

添加 (A)

载入 (L)...

保存 (S)...

等值线图 对话框中的等级列表。你可以双击一个独立的元素以设置相应的参数。

为了改变等值线的线条属性，我们可以：

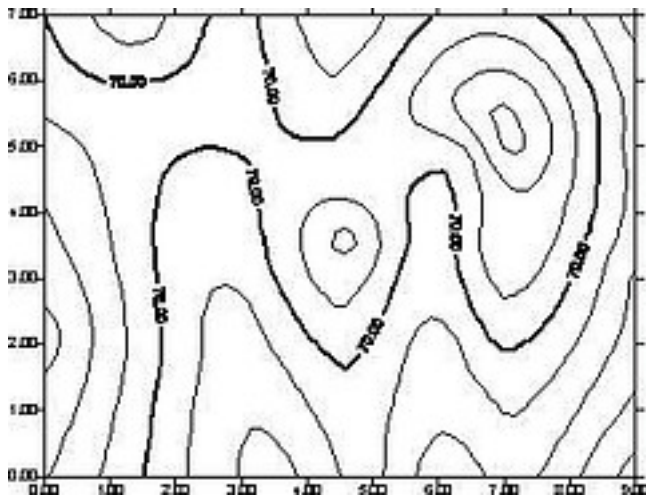
- 1. 双击等值线图，弹出 等值线图 对话框。

2. 在 等值线等级 组中双击 $Z = 70$ 的线条样本，出现 线条属性 对话框。



线条属性 对话框允许你设置指定线条的属性。

3. 在这里你可以选择任何线条颜色、样式或宽度，在 宽度 编辑框中，点击向上箭头使宽度值变为 0.076 cm，点击 确认 使得 等值线等级 列表更新。



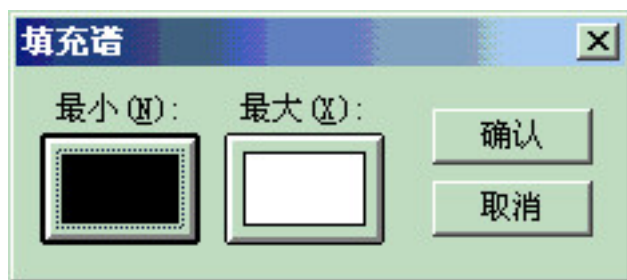
地图中 $Z = 70$ 的等值线被加粗。

4. 点击 等值线图 对话框中的 确认 按钮，我们看见地图自动的刷新了， $Z = 70$ 的等值线线条被加粗了。

在等值线线条之间添加颜色填充

我们确实可以在等值线线条之间添加颜色填充。我们可以像改变等值线线条属性一样方便的改变不同等级之间的颜色填充，当然我们也可以通过指定一个基本的颜色谱来定义梯度变化的颜色填充。

1. 双击等值线图出现 等值线图 对话框。



填充谱 对话框允许我们指定颜色填充谱。

2. 在 等值线等级 组中，单击 填充 按钮出现 填充谱 对话框。 最小 按钮显示的是最小等值线等级对应的填充颜色，而 最大 按钮显示的是最大等值线等级对应的填充颜色。

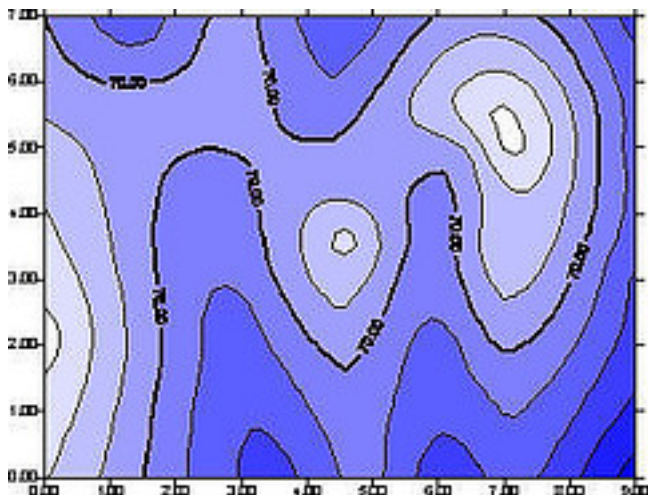
3. 单击 最小 按钮出现 填充属性 对话框。在这个对话框中我们可以指定对应于地图中最小等值线等级的填充颜色和填充图案。我们来选择颜色表中的蓝色“Blue”，然后点击 确认， 填充谱 对话框中的 最小 按钮现在变成蓝色了。点击 确认 就发现 等值线等级 列表中的颜色发生了相应的变化。



填充属性对话框。

4. 在 填充等值线 组中，单击 填充等值线 复选框。

5. 单击 确认，等值线图就按照我们选定的颜色进行填充了。



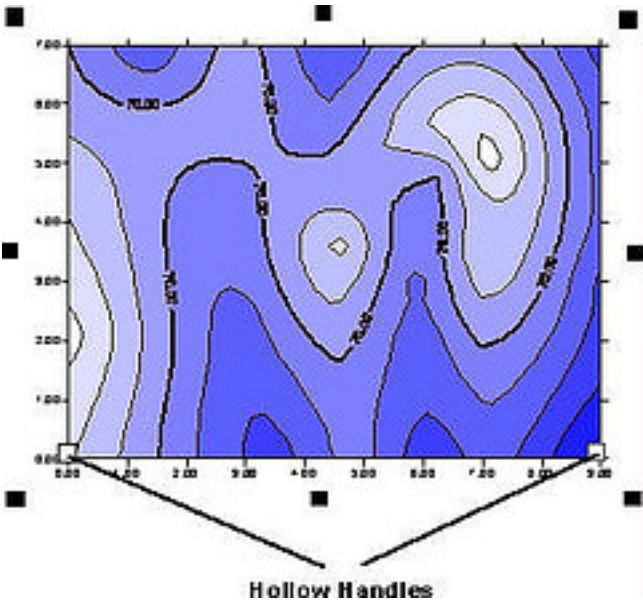
一个进行了颜色填充的等值线图。

修改一个坐标

一个等值线图有四个坐标。本图中每个坐标都添加了标题以说明不同的坐标。

每个等值线图都包含四个坐标：Bottom, Right, Top 和 Left 坐标。我们可以单独的控制地图中的任何一个坐标的显示。在我们示范的地图中 Bottom 坐标的主间距是 1.0，而标注使用了 2 位小数，现在来看看如何修改这两个参数，并且添加一个坐标标题。

1. 移动鼠标指针到底部 X 坐标的刻度标注上并且点击，在图形窗口底部的 状态栏 将显示关键字” Map : Bottom Axis”，这表示已经选定的等值线图的底部坐标。 另外，坐标两端将出现空心的框，整个地图的四周则出现包围的黑色实心框，表示所选中的坐标是整个地图的一个”子对象”。



八个实心的句柄包围选定的地图，空心的句柄出现在选定的子对象的末端，在这个例子中，是 Bottom 坐标。

2. 当鼠标指针在坐标刻度标注上面的时候，双击将出现 Bottom Axis 对话框，在这里可以控制选定的坐标的显示。



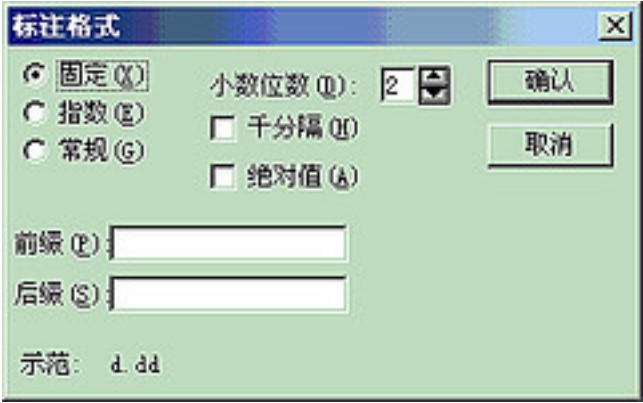
Bottom Axis 对话框控制了底部坐标的显示，如果你需要修改其他坐标的显示，

双击该坐标。



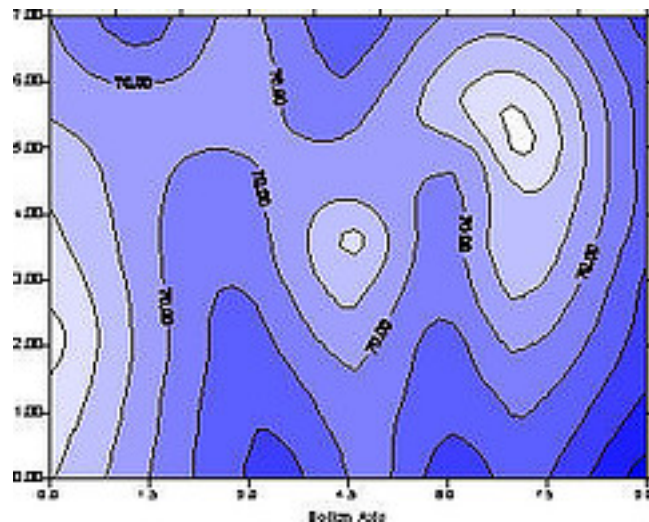
坐标比例 对话框允许你指定坐标限制、主刻度间距和位置。

3. 在 标题 编辑框中，键入“Bottom Axis”，这样就为选定的坐标添加了一个标题。
4. 点击 比例 按钮显示 坐标比例 对话框，在 主间距 编辑框中，键入 1.5，表示改变选定坐标主刻度间距。点击确认 返回 Bottom Axis 对话框。
5. 点击 标注 组的 标注格式 按钮，显示 标注格式 对话框，点击向下箭头改变 小数位数 编辑框的值为 “1”。这个设置表示坐标的刻度标注使用一位小数，点击 确认 返回 Bottom Axis 对话框。



标注格式 对话框

6. 点击 Bottom Axis 对话框的 确认 则地图被刷新，坐标刻度的间距和标注改变了，地图的下方放置了一个坐标标题。



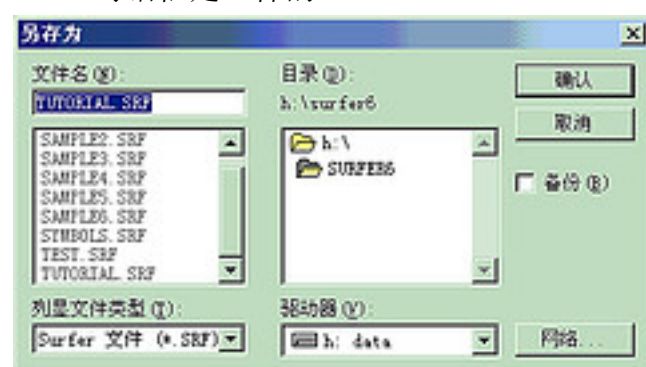
坐标刻度标注和间距已经修改，并且添加了一个坐标标题。

保存一个地图

当你绘制完一个地图后，你可以保存这个地图为 Surfer [.SRF] 文件格式，它将包含所有用于绘制此地图的必须的信息。当你保存一个地图的时候，所有的比例、格式和地图的各项参数都将保存在 [.SRF] 文件中。

要保存一个地图：

1. 从 文件 菜单选择 保存 命令，或者点击工具栏上的 保存 工具 。如果你是第一次保存地图，将出现一个 另存为 对话框，其中的驱动器和目录与 Open Grid 对话框是一样的。



这就是 另存为 对话框

第 3 课 - 创建一个表面图

第 3 课 - 创建一个表面图

表面图是一个网格文件的三维表现，表面图是通过绘制网格 X 和 Y 线条（网格列和行）生成的块状图表。在每个列和行的相交点，表面的高度是该点相应的网格 Z 值，网格文件中列和行的数目决定了表面图 X 和 Y 线条的数目。

要在一个新的、空白的图形窗口创建一个表面图：

1. 选择 文件 菜单下的 新建 命令或者点击工具栏上的 新建 工具，在显示的新建窗口 对话框中，选择 图形 选项，点击 确认，一个新的、空白的图形窗口出现了。



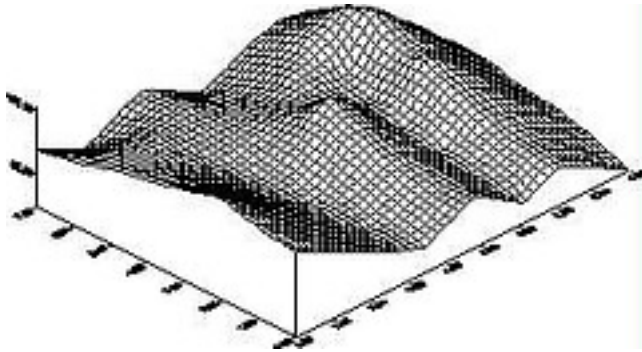
新建窗口 对话框允许你指定创建的窗口类型。

2. 从 地图 菜单选择 表面 命令，或者点击工具栏上的 表面图 工具，出现 Open Grid 对话框，选择网格文件 “TUTORWS.GRD” 然后点击 确认，出现 表面图 对话框。



表面图 对话框。

3. 点击 确认 接受所有默认设置,一个基于“TUTORWS.GRD”文件的表面图就绘制成功了,表面图比其他图形多了一个 Z 坐标。

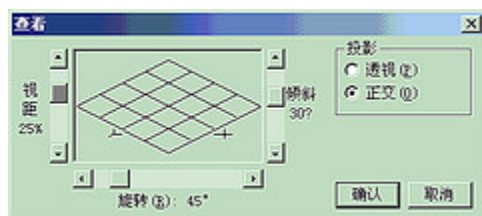


一个基于“TUTORWS.GRD”文件的表面图。

改变一个表面图的方向

你可以使用任何的方向或者倾斜来显示任何地图,虽然这个例子使用的是一个表面图,但是你可以调整任何类型的地图的方向。

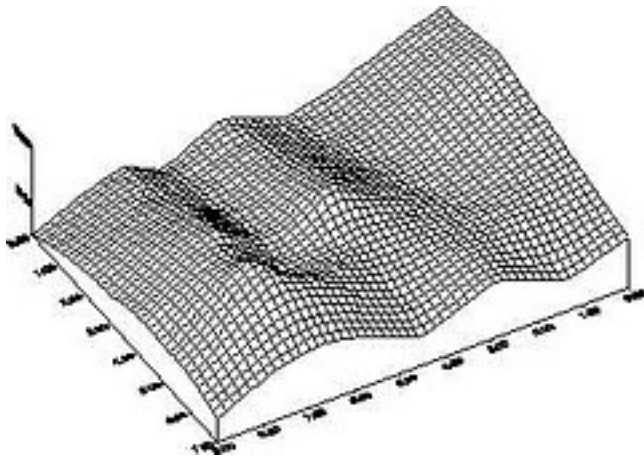
1. 点击表面图,选择句柄出现在表面图四周。
2. 选择 地图 菜单下的 3D 查看 命令,显示 查看 对话框,这里你可以调整选定地图的倾斜、旋转和投影类型。



查看 对话框允许你指定倾斜角度、旋转角度和地图的投影类型。

3. 使用 Rotation 滚动条改变旋转角度为 210°,在你改变角度的时候,对话框中的图形即时显示了这种改变。
4. 使用 Tilt 滚动条改变角度为 40°。

5. 点击 确认 则表面图按照新的方向和倾斜角度重新绘制。



一个新的旋转角度和倾斜角度的表面图。

改变一个表面图的比例

你可以任何地图的比例，在这里我们示范改变表面图的 Z 比例。

1. 点击表面图，选择句柄出现在表面图四周。

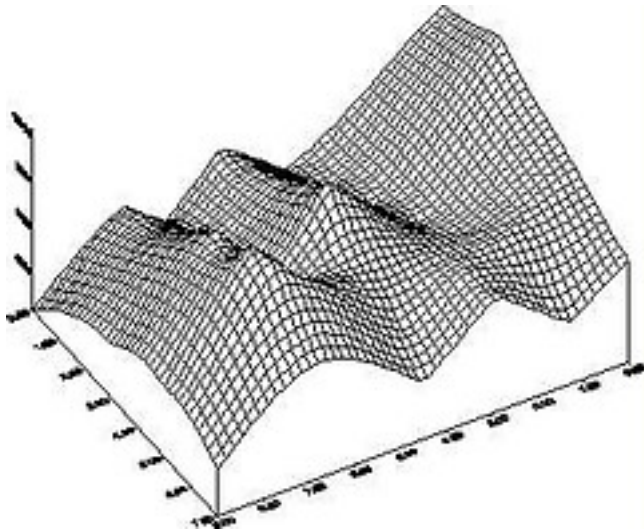


比例 对话框指定选定地图的比例。

2. 选择 地图 菜单下的 比例 命令，显示 比例 对话框，这里允许你改变选定地图的比例，可以键入地图的长度值，也可以键入比例值。

3. 在 Z 比例 组的 长度 编辑框中，键入 7， 地图单位 值自动改变。

4. 点击 确认 则表面图按照新的 Z 比例重新绘制。



表面图的 Z 比例改变，Z 方向被放大。

第 4 课 - 张贴数据点并使用覆盖

第 4 课 - 张贴数据点并使用覆盖

张贴图用来放置数据点到一个地图上。包含了 XY 坐标的数据文件用于定位地图上数据点的位置，同时也可以对每个点进行标注。张贴数据点到地图上，以及放置数据或文本信息到指定点上，对于检查数据点的分布是非常有用的。

当你张贴数据点到一个等值线图或表面图上的时候，你“覆盖”该张贴图到等值线图或表面图上。地图覆盖并不限制于等值线图和表面图，你可以覆盖等值线图、基面图（包含边界的地图）和张贴图，你也可以覆盖这些类型的地图到一个表面图上，但是在一个覆盖中，只能有一个表面图。

当你创建一个张贴图的时候，你在当前的绘图窗口创建了一个完全独立于其他地图的一个新的地图。当两个地图同时显示的时候，你会发现两个坐标集，每个坐标集对应一个地图。当你使用 地图 菜单下的 覆盖地图 命令的时候，两个地图被组合为一个单独的地图，拥有且仅仅有一个坐标集。

1. 选择 文件 菜单下的 打开 命令显示 打开文件 对话框，选择 “TUTORIAL.SRF” 并点击确认显示 TUTORIAL.SRF 中包含的地图。
2. 选择 地图 菜单下的 张贴 命令，或者点击工具栏上的 张贴 工具 ，显示 Open Data 对话框，这里允许你选择用于生成张贴图的数据文件。
3. 在文件列表中，点击 “TUTORWS.DAT”，我们看到文件名出现在 文件名 编辑框中，点击确认就生成了一个 张贴图。



粘贴图对话框

4. 点击 默认符号 按钮显示 符号属性 对话框。



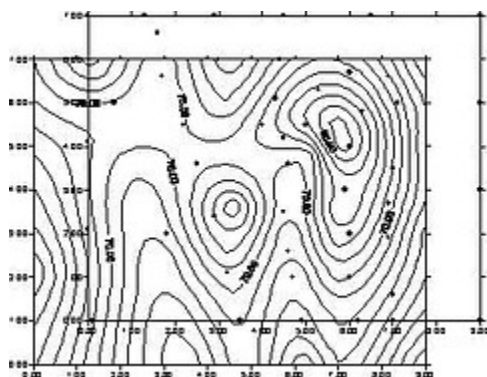
符号属性 对话框。

5. 从 符号 面板中选择填充的圆符号，点击确认，该符号将显示在 符号 按钮上，此符号将用于显示地图上的数据点。

6. 在 符号尺寸 组的 固定尺寸 编辑框，指定 0.25 cm。

7. 点击确认则粘贴图绘制在等值线图上方，等值线图和粘贴图是两个完全独立的地图，如果你仔细看 X 坐标，你会发现两个地图有两个坐标刻度标注集。

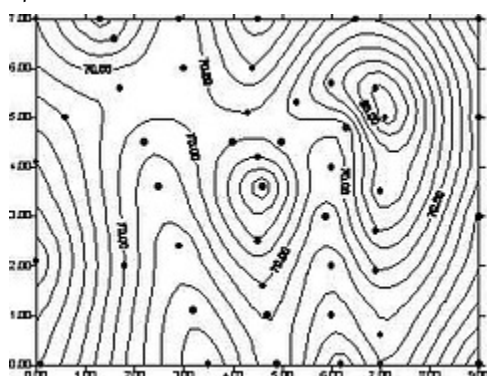
8. 为了看清楚这是两个独立的地图，将鼠标光标移动到地图中间并点击，按住鼠标左键然后向任意方向稍微移动鼠标，松开鼠标左键，两个地图间有了相应的位移，你可以移动等值线图或粘贴图。



两个地图稍微错位以显示等值线图和张贴图是独立的

9. 按键盘的 F2 键，选择两个地图。

10. 选择 地图 菜单下的 覆盖地图 命令，两个地图被组合到一个单独的覆盖中。



等值线图和张贴图被组合到一个覆盖中，这将在每个数据位置放置点符号

从覆盖中选择一个地图并指定一个对象 ID

当你创建了地图覆盖后，你仍然可以修改覆盖中的每个地图，有两个方法可以从一个覆盖中选择一个地图：你可以双击地图或者你可以选择覆盖，然后选择 地图 菜单下的 编辑覆盖 命令。

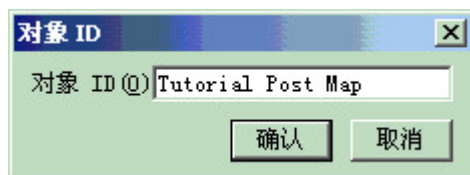
要用鼠标选择交迭的地图，你经常会需要 CTRL 键来配合。实际上，任何时候当图形窗口中存在多个对象的时候，你都可能需要 CTRL 键来配合选择你想选择的对象。CTRL 键允许你循环遍历交迭的对象，例如，如果你要选择一个矩形下面的文本框，或者一个等值线图后面的表面图，按住 CTRL 键并点击鼠标左键，直到你选择了需要的对象，你可以使用 状态栏 来帮助选择对象。



Surfer 窗口底部的 状态栏 指示了图形窗口中选中的对象，本例中，状态栏 报告一个张贴图被选中了

要从一个覆盖中选择一个地图，然后为该地图指定一个对象 ID:

1. 移动鼠标光标到覆盖上，点击，覆盖中被选中的地图将显示在 状态栏 上，例如，如果选中的地图是等值线图，那么 状态栏 将显示 “地图: Tutorial 等值线图”。
2. 要选择覆盖中的其他地图，保持光标在地图上，按下 CTRL 键，然后点击，检查 状态栏，这时覆盖中的另一个地图应该被选中了，例如，状态栏 现在应该报告 “地图: Post”。
3. 当 状态栏 指示张贴图被选中的时候，选择 编辑 菜单下的 对象 ID 命令。



对象 ID 对话框为选中的对象指定一个 ID

4. 在 对象 ID 对话框中，键入名称 “Tutorial Post 地图” 并点击确认，状态栏 立刻显示了这个改变。

添加张贴标注到覆盖中的张贴图中

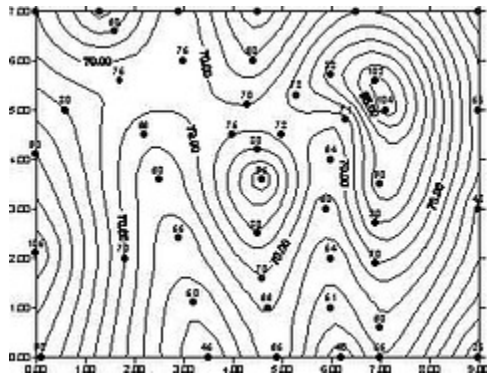
你可以添加标注到覆盖中张贴图的数据点上，在本例中，你将使用 编辑覆盖 命令选择一个张贴图。此外你还可以直接在覆盖中选择 “Tutorial Post 地图”，然后双击显示 张贴图 对话框。



编辑元件 对话框列出了覆盖包含的所有对象

1. 选择覆盖，然后选择 地图 菜单下的 编辑覆盖 命令，显示的 编辑元件 对话框中有一个包含所有覆盖对象的列表。
2. 从 覆盖 列表中选择 Tutorial Post 地图，然后点击 编辑 按钮，显示 张贴图 对话框。
3. 在 工作表列 组中，点击 标注 下拉列表，显示 “TUTORWS.DAT” 文件中所有的列。

4. 从列表中选择 Column C: Elevation, 此列包含了 "TUTORWS.DAT" 文件中的 Z 值。
5. 点击 格式 按钮显示 标注格式 对话框。修改 小数位数 为 0, 因为海拔高度的数据点精度到米就可以了, 点击确认返回 张贴图 对话框。
6. 点击确认则覆盖被重新绘制并且每个数据点显示了标注。



覆盖中的张贴图被修改以显示每个数据点的值

第 5 课 - 创建一个 XYZ 数据文件

第 5 课 - 创建一个 XYZ 数据文件

	A	B	C
1	X Data	Y Data	Z Data
2	0.1	0	90
3	9	3	48
4	1.3	7	52
5	4.7	1	66
6	1.7	5.6	75
7	6	1	50
8	2.5	3.6	60

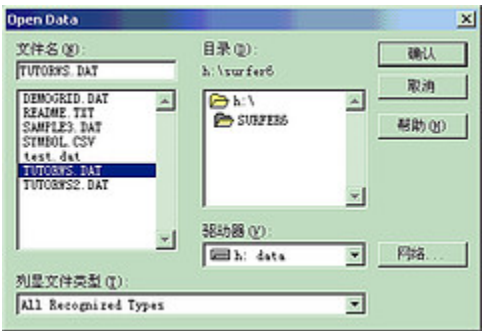
一个简单的 XYZ 数据文件, 注意到 X, Y 和 Z 数据被相应的放在 A, B 和 C 列

一个 XYZ 数据文件是一个至少包含三列数据值的文件, 开始的两列分别是用于定位数据点位置的 X 和 Y 坐标, 而第三列是 XY 点上的 Z 值。将 X 坐标放在 A 列、Y 坐标放在 B 列、Z 值放在 C 列是一个良好的习惯, 虽然并不是强制要求, 也不是必须如此。因为 Surfer 在默认的情况下就是按照这样的次序搜索这些数据的。

如果你的数据已经存放到一个文件中了, 你就不需要象本课讲述的一样创建一个新的 XYZ 数据文件了, 而且你也不需要在一个工作表窗口中显示你的数据。然而, 如果你的数据并没有形成一个文件, 你必须首先生成一个数据文件, 然后才能对该数据文件进行其他的操作。

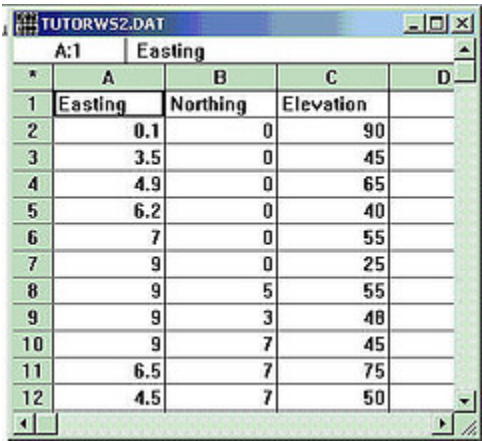
要查看一个示范的 XYZ 数据文件，你可以在一个工作表窗口中打开“TUTORWS2.DAT”：

1. 从 文件 菜单中选择 工作表 命令，或者点击工具栏上的 工作表 工具按钮，一个新的空白的工作表出现了。此外你也可以选择 文件 菜单下的 新建 命令，出现 新建窗口 对话框，选择 工作表 选项并点击确认，一个新的空白的工作表出现了。
2. 选择 文件 菜单下的 打开 命令，或者点击工具栏上的 打开 工具按钮，显示 Open Data 对话框，这里允许你选择一个显示在工作表窗口的 XYZ 数据文件。



在 Open Data 对话框中，选择 Surfer 附带的示范文件“TUTORWS2.DAT”

3. 在文件列表中，点击“TUTORWS2.DAT”，则文件名显示在 文件名 编辑框中，点击确认，文件内容显示在工作表窗口中。



	A:1	Easting		
*	A	B	C	D
1	Easting	Northing	Elevation	
2	0.1	0	90	
3	3.5	0	45	
4	4.9	0	65	
5	6.2	0	40	
6	7	0	55	
7	9	0	25	
8	9	5	55	
9	9	3	48	
10	9	7	45	
11	6.5	7	75	
12	4.5	7	50	

当你显示一个数据文件的时候，文件名显示在工作表窗口的标题栏。在本文件中，第一行包含了每列数据的说明信息。

4. 注意 X 坐标 (Easting) 在 A 列，Y 坐标 (Northing) 在 B 列，Z 值 (Elevation) 在 C 列，标题文本（第一行中的文本）对于识别列中的数据是非常有帮助的，在对话框中也用于指定工作表列。

你可以使用 Surfer 的工作表来创建一个数据文件，要打开一个工作表并开始输入数据：

1. 选择 文件 菜单下的 新建 命令,或者点击工具栏上的 新建 按钮 ,点击新建窗口 对话框中的 工作表 选项,点击确认显示一个新的空白的工作表。
2. 数据被输入到活动单元格中,活动单元格通过鼠标点击选中,或者使用光标键在单元格之间移动。活动单元格有一个加粗的边框表示,其内容显示在 活动单元格 编辑框中。
3. 当一个单元格被激活的时候,你可以键入一个值或文本,而你所键入的信息同时显示在活动单元格和 活动单元格 编辑框中。
4. 在键入或编辑数据的时候可以使用 BACKSPACE 和 DELETE 键。
5. 按 ENTER 键则数据被输入到单元格中。
6. 为了保存键入的数据到活动单元格并移动到一个新的单元格中,你可以直接点击一个新的单元格,或者按光标键。



数据被输入到工作表中的活动单元格,活动单元格有指示并且该活动单元格的内容显示在活动单元格编辑框中。

保存数据文件

当你完成数据输入的时候:

1. 选择 文件 菜单下的 保存 命令或者点击工具栏上的 保存 工具按钮 ,如果你之前还没有保存过数据文件则显示 Save As 对话框。
2. 在 文件保存类型 下拉列表框中,选择 ASCII Files (*.DAT) 选项。
3. 在 文件名 编辑框中键入名称,点击确认则文件保存为 ASCII [.DAT] 格式,文件名显示在工作表窗口的标题栏上。

Surfer 8 教程

使用 Flash SWF 格式制作一些简单的 Surfer 8 教程,使用 Holz 汉化的 8.05 版本。

Surfer 8 的一些问题通过升级到 8.05 版以后会自动消失，我已经接到数个报告，都是低版本有问题而到了高版本没有的。

无论如何，现阶段应该升级到最新的版本。

01. 建立数据

Surfer 8 是基于网格的科学绘图软件，网格来自原始的数据文件。

最原始的数据文件可以通过手工输入，假如我们还在使用 DDC-2B 在野外采集直流电法数据，那么，我们总是需要手工输入数据的。

02. 打开数据文件

如果使用现代的设备，那么我们就可以直接由仪器得到数据文件，例如 WDC-2B、WGMD-1 等仪器。

但一般仪器采集的数据是否要进行预编辑呢？是不是要进行一些统计呢？是不是要剔除一些坏点呢？

用 Surfer 8 的工作表打开数据文件就可以对数据进行编辑、统计、变换等操作。

03. 数据网格化

采集到的数据输入到工作表，编辑，统计，处理完毕后，就可以对数据进行网格化了。

04. 建立等值线图

数据网格化之后得到一个网格文件，这个网格文件就是绘制各种地图的基础。

看看地图菜单，有许多图件的建立都是需要一个网格文件的。

我们先来看看如何建立一个简单的等值线图。

05. 打开地图属性

默认建立的等值线图往往是不能满足我们的要求的，所以我们需要改变地图的属性。

要打开一个等值线图的属性设置对话框，有许多方法。

可以用主菜单、可以用上下文菜单、可以用对象管理器、甚至可以用键盘（通过移动光标到等值线图上连续按两次空格即可访问属性对话框，按 ESC 相当于取消）。

06. 改变等级

改变等值线的等级有三种方法，可以线性改变，只要修改最小值、最大值、间距；可以单独改变某个等级数值；可以载入一个编辑好的 [等级文件](#)。

07. 改变等值线线条

等值线线条属性的修改也有三种方法，可以通过等级中线条按钮线性改变；可以单独改变；可以在等级文件中定义，然后导入。

这里只演示单独改变线条属性的方法，其余两种方法类似 [等级的改变](#)。

08. 添加等值线填充

在 Surfer 8 中，启用等值线填充和修改填充属性分别处于不同的页面。

09. 编辑等值线标注

可以手工编辑等值线标注，随意添加、删除、移动等值线标注，是 Surfer 8 中一个迷人的特性，对于最终生成一个美观的地图非常有用。

10. 修改坐标

修改坐标，就象修改等值线属性一样，有许多中途径访问其属性对话框。

在许多时候，坐标是不得不进行修改的，然而每个地图容器中，坐标对象就有四个，而 surfer 8 没有 Grapher 4/5/6 那样的模板特性，所以许多时候不得不使用脚本来加速我们的工作（是懒人的做法）。

11. 输出等值线

能够单独输出等值线，也是 Surfer 8 一个不错的特性，因为有些时候我们完全不需要任何多余的东西。

输出等值线命令和文件菜单中的输出命令是有区别的，文件菜单中的命令输出所有对象，包括被覆盖而我们不能看见的对象，只要输出就原形毕露了。

12. 建立张贴图

张贴图用符号和标注来表现 XY 位置。张贴图在检查数据的空间分布以及数据密度非常有用。我们还可以用来做一些特殊的数据点位图。

张贴图既可以单独用来表现原始数据的空间分布特征，也可以将张贴图覆盖到一个基于网格的地图（例如等值线图）上，以得到更完美的数据可视化效果。

张贴图的限制是依据所使用数据文件的 XY 坐标的，但在需要的时候随时可以改变。

在 Surfer 8 中，张贴图有两种不同的类型。

13. 改变张贴图属性

在张贴图的属性对话框中有非常详细的参数可以调整一个张贴图的表现，包括符号、大小、颜色、标注等用得最多的属性。

14. 覆盖地图

如果不同的地图有相同的坐标范围，例如使用同一个原始数据建立的张贴图和等值线图、线框图、表面图等，就可以通过覆盖的方式组合到一个地图对象容器内。

可以参加覆盖的地图类型有基面图、张贴图、等值线图、渐变阴影图、影像图、矢量图以及表面图。

使用覆盖的方法只有一种：首先选中要参加覆盖的地图，然后选择 **地图 >> 覆盖地图** 命令。

15. 数字化边界

简单地说，数字化就是将地图的坐标写入一个数据文件中。

在数字化模式中，状态栏显示了鼠标所在位置的 XY 坐标，单击鼠标左键记录下该点坐标值，暂时放在数字化编辑器窗口中。

数字化完毕，您可以将数字化编辑器窗口中的数据保存为边界文件（可用于白化、基面图等），也可以保存为数据文件（可用于张贴图等）。

只有那些水平（倾斜 = 90°）且平面的地图才可以进行数字化。你不能对表面图和线框图进行数字化。

16. 建立基面图

基面图可以方便地表现地理、行政方面的信息，例如道路、水系、湖泊、岛屿、各种行政边界等。

基面图可以包括点、线、面、文本甚至是光栅图像。基面图可以使用众多文件格式，包括 [BLN](#)、[BNA](#)、[PLT](#) 等边界文件。

17. 改变基面图属性

基面图有几个主要的属性参数：线条、填充、文本、符号。

其中线条针对所有以折线组成的边界线如多段线、多边形边框等；填充针对封闭对象如多边形；符号针对点对象。

18. 非模态对话

Surfer 8 一个很吸引人的特性是采用了非模态的属性对话框。你完全可以打开着属性对话框，把它扔一边然后干其他活去，当你选中某个对象后，该对象的属性内容就出现在打开着的对话框上。

当然你在改变你的选择之前，记得点 **应用** 按钮。

新的属性对话框还可以显示多个对象的公共属性。这有几种情况：

1. 如果多个相同类型的对象被选中，如 N 个等值线图，那么对话框显示第一个选中的对象的选项。
2. 如果多个对象被选中，但不是同类，如矩形和等值线图，由于他们之间没有任何相同的属性，所以对话框提示你**他们没有任何公共属性**。
3. 如果多个地图被选中，但不同类，例如等值线图和线框图，那么就显示公共的属性，例如查看和背景。
4. 如果多个同类的地图被选中，例如多个等值线图，你会发现等级不能应用到所有地图上，因为等级是每个地图特定的属性，只能每个地图单独设置。

Surfer 不能的事

虽然 Surfer 8 一直顽固的不升级，就象当年 DOS 下一样，它也一样有各种各样的局限性。

DXF 输出别太长

在 Surfer 的帮助中，DXF 输出选项有许多控制参数，以为可以精细地控制输出的 DXF 文件。

当我用脚本测试的时候，发现这个选项字符串是不能太长的，否则执行不了。

解决办法？Golden Software 说 no way！你还是乖乖将这个选项字符串缩短，缩短，再缩短，直到脚本能执行为止罢。

网格别过万

在 Surfer 8 中，网格 [.GRD] 文件是有限制的，那就是网格密度不能超过 10000×10000 。

也许我们都以为网格文件不应有限制，只有 Windows 的内存限制，但事实就是这么残酷。

也许我们可以期待 Surfer 9。

但目前我们只好在处理超大网格时转用 ArcGIS。

自动化输入 DXF 不能指定层

当我们使用 Golden Software Surfer 8 的时候，如果要用一个 DXF 文件建立基面图，那在输入 DXF 文件的时候，会有一个选项让我们指定要冻结的层。

这个特性有时候很有用，因为我们可能只需要 DXF 文件中的某些实体就可以了。

但如果您自己编写了自动化脚本，希望通过自动化方式输入 DXF 文件，那就好好考虑了。Surfer 8 的自动化接口中并不包括指定 DXF 层的选项。

输出 GIF 动画靠人帮忙

Golden Software Surfer 8 可以输出 GIF 格式的图片，并且通过自动化脚本，还可以使生成的地图活动起来。

例如做一个 3D 表面图，那么要看这个地图的全部，就可以通过自动化脚本控制使得地图旋转运动，就象一个 3D 演示。

当然您可以通过脚本控制输出图片，将旋转的地图输出为 GIF 动画多好，但是慢着，Surfer 8 会生成一张张的 GIF 图片，却不会将这些图片变成动画。

要将一个动画演示过程中输出得到的 gif 图片综合为一个动画 gif 文件，需要 [ImageMagick](#) 的帮忙。

透明图片不透明

知道什么图片会是透明的？GIF、PNG 等等，都有背景透明特性罢？

您希望导入一个透明图片到 Surfer 8 的图形文档中罢？很遗憾，透明的背景在 Surfer 8 中不再透明，而是白色。

如果这个图片在其他图形上面，下面的所有对象将不可见。

是不是觉得支持图片透明其实不难？但是，让我们期待下一个版本罢。

目前的应急办法？将图片放到最底层罢。

有断层的地图不要乱改输入网格

如果你做了一个等值线图并且有断层，没问题，Surfer 很正确的显示了结果。但如果此时打开等值线图的属性，在常规标签下选择另外一个不包含断层的网格文件，应用后，奇怪的问题出现了。此时断层仍在，而等值线却是新网格的。问：有什么解决办法？答：乖乖重新做一个等值线图。

Surfer 快速上手

Golden Software Surfer 的快速上手指南，超级简单，只要三步。

大家念 123，就学会了：

1. 启动 Surfer（废话）。打开工作表窗口，输入 XYZ 三列数据，一般来说 A 列为 X，B 列为 Y，C 列为 Z。将输入的数据保存（还是废话）；
2. 网格化选择 Grid -> Data，选取要网格化的数据文件，网格化完毕后存成.GRD 文件；
3. 选择要画的 MAP 图形，如 Contour Map 就是等值线图（或者叫等高线图），选择一个.GRD 文件就会看到结果。

就是这么简单！ I'm Loving it ！

若您要改变设置，往图形中央双击，即会显示出属性对话框窗口。

交换文件格式

Golden Software 系列软件支持多种交换文件格式，其自身也定义了若干文件格式，如 BNA、BLN、PLT 等。

好用的 LVL 格式

作者: Holz (hemengui [at] gmail [dot] com)

参考: Surfer 8 帮助

LVL 是等级文件格式，一般用于 [Surfer](#)，这种文件定义了特定的等级值及其相应的各项属性。

等级文件是一个明码文件，每行表示一个记录，每行中的各参数之间可以用逗号或者空格来分隔。

等级文件还可以有注释行，在行的前面加一个半角的单引号表示该行被注释。

等级文件有两种不同形式的文件格式，第一种是简单的，只有 Z 值。例如：

```
0.1
1
10
100
1000
```

复杂的等级文件有文件头信息，一般如下：

```
LVL2
'Level Flags LColor LStyle LWidth FFGColor FBGColor FPattern FMode FScale
FType
```

其中第一行 LVL2 是必须的，他是一个标识，说明这是等级文件的类型。第二行其实是注释，说明后面各列数值或字符代表的含义。

我们看 [Surfer 8 教程](#) 中使用的一个等级文件的内容：

```
LVL2
'Level Flags LColor LStyle LWidth FFGColor FBGColor FPattern FMode FScale
FType
0 1 "Cyan" "Solid" 0 "Black" "White" "Solid" 2 1 1
25 0 "Cyan" "Solid" 0 "R1 G1 B1" "White" "Solid" 2 1 1
```

```

50 0 "R1 G254 B255" "Solid" 0 "R2 G2 B2" "White" "Solid" 2 1 1
75 1 "R2 G253 B255" "Solid" 0 "R2 G2 B2" "White" "Solid" 2 1 1
100 0 "R3 G252 B255" "Solid" 0 "R2 G2 B2" "White" "Solid" 2 1 1
125 0 "R4 G251 B255" "Solid" 0 "R3 G3 B3" "White" "Solid" 2 1 1
150 1 "R5 G250 B255" "Solid" 0 "R4 G4 B4" "White" "Solid" 2 1 1
200 0 "R7 G248 B255" "Solid" 0 "R6 G6 B6" "White" "Solid" 2 1 1
250 0 "R9 G246 B255" "Solid" 0 "R9 G9 B9" "White" "Solid" 2 1 1
300 1 "R10 G245 B255" "Solid" 0 "R13 G13 B13" "White" "Solid" 2 1 1
400 0 "R14 G241 B255" "Solid" 0 "R15 G15 B15" "White" "Solid" 2 1 1
500 0 "R18 G237 B255" "Solid" 0 "R18 G18 B18" "White" "Solid" 2 1 1
750 1 "R27 G228 B255" "Solid" 0 "R27 G27 B27" "White" "Solid" 2 1 1
1000 0 "R36 G219 B255" "Solid" 0 "R36 G36 B36" "White" "Solid" 2 1 1
1500 0 "R54 G201 B255" "Solid" 0 "R54 G54 B54" "White" "Solid" 2 1 1
2000 1 "R72 G183 B255" "Solid" 0 "R72 G72 B72" "White" "Solid" 2 1 1
2500 0 "R91 G164 B255" "Solid" 0 "R91 G91 B91" "White" "Solid" 2 1 1
3000 0 "R109 G146 B255" "Solid" 0 "R109 G109 B109" "White" "Solid" 2 1
1
3500 1 "R127 G128 B255" "Solid" 0 "R127 G127 B127" "White" "Solid" 2 1
1
4000 0 "R145 G110 B255" "Solid" 0 "R145 G145 B145" "White" "Solid" 2 1
1
4500 0 "R163 G92 B255" "Solid" 0 "R163 G163 B163" "White" "Solid" 2 1 1
5000 1 "R182 G73 B255" "Solid" 0 "R182 G182 B182" "White" "Solid" 2 1 1
5500 0 "R200 G55 B255" "Solid" 0 "R200 G200 B200" "White" "Solid" 2 1 1
6000 0 "R218 G37 B255" "Solid" 0 "R218 G218 B218" "White" "Solid" 2 1 1
6500 1 "R236 G19 B255" "Solid" 0 "R236 G236 B236" "White" "Solid" 2 1 1
7000 0 "Magenta" "Solid" 0 "White" "White" "Solid" 2 1 1

```

每行数据有 11 列，每列数据均有特定含义，下表：

列	标识	说明
1	Level	等值线的等级值
2	Flags	标注和影线开关， 0=普通（无标注，无影线），1=标注， 2=影线，3=既有标注又有影线
3	LColor	该等级的线条颜色
4	LStyle	该等级的线条风格，实线、虚线等，命名要符号 Golden Software 规则，并且要用双引号括住
5	LWidth	线条宽度，单位是千分之一英寸

6	FFGColor	填充的前景颜色
7	FBGColor	填充的背景颜色
8	FPattern	填充图案，必须与对话框中出现的名称一致并且要用双引号括住
9	FMode	背景模式， 1=透明， 2=不透明
10	FScale	比例系数，仅用于矢量填充图案。
11	FType	填充类型， 0=unknown, 1=stock, 2=bitmap, 3=vector.

注：

1. 其中颜色既可以使用名称，也可以使用 "Rxxx Gyyy Bxxx" 这样的字符串标识。
2. 线条风格、填充图案、颜色的名称列表其实可以查看 Surfer 8 安装目录下的 attrib.ini 文件。

小巧的 BLN 格式

作者：Holz (hemengui [at] gmail [dot] com)

参考：Surfer 8 帮助

BLN 格式的文件称之为白化文件 (Blanking File)，用于表现边界的信息，也可以用于白化网格文件的某个多边形区域。白化文件用逗号分隔，是个明码文件。

白化文件的格式是这样的：

```
length, flag "Pname 1"
x1, y1
x2, y2
...
xn, yn
length, flag "Pname 2"
x1, y1
x2, y2
...
xn, yn
```

其中

- length 是一个大于 0 的整数，表示后面跟着的 X,Y 坐标对数目；

- flag 为 1 时表示白化区域的内部，0 表示区域的外部被白化；
- Pname 是一个可选的参数，一般我们不用，它用于与对象关联的主 ID 名，主 ID 用于将对象与外部数据链接。
- 接着的 X,Y 坐标对用于建立对象，可以是整数或实数，但要保证每行只放一个坐标对。

BLN 文件虽然简单，但一样可以描述点、线和区域对象，这些对象全部通过 length 字段来表示：

1. 当 length 为 1 时，该对象为一个点，因为后面只能有一个坐标对。
2. 当 length 大于 1 时，可以表示一条线，也可以表示一个区域。当第一个坐标位置与最后一个坐标位置不相同，为一线对象；
3. 当 length 大于 1 且第一个坐标位置与最后一个坐标位置相同时，表示一个封闭的区域。

白化文件还可以作为**折断线**使用，但此时需要 Z 值。

折断线用于数据网格化，并不能用于白化。

下面是一个折断线类的 BLN 文件例子：

```
2, 1
3.0, 4.5, 1.0
0.5, 4.5, 2.0
3, 1
4.5, 0.0, 1.1
8.5, 0.0, 1.2
8.5, 1.5, 1.4
```

这种文件能够用于 [Golden Software](#) 系列的软件中，一般作为边界文件使用，只有在 Surfer 中可用于数据网格化和网格白化。

强大的 PlotCall

作者：Holz (hemengui [at] gmail [dot] com)

参考：Surfer for DOS 帮助

以前在DOS 6.22 下，在那台 386SX+4M内存+500M硬盘的便携式电脑上，是我玩 Surfer和Grapher的顶峰时期，三年的黄金时期，以至于Surfer6 和Surfer7 都分别只玩了半年就转 [Surfer 8](#) 了。

很巧合，DOS 下的 Surfer 也是 8 版。

那时候,装备多强阿,有 NC 管理文件,有 QE 全屏编辑文本和 bas 程序,有 basica 执行程序,有 TW 和单文件的 WPS 写报告。在 Dos 下都不能显示汉字,因为内存太少。而绘制的图件却需要中文(图名、责任表、标注……),多亏了 plt 格式是明码文件,多亏了那几个简单而强大的 PlotCall 命令,让我能够为所欲为。

记得写了简单的 bas 程序,可以自动往生成的 plt 文件中加入图框可责任表,记得老总的 bas 程序,可以将美丽的中文字符转为 PlotCall 命令。

可惜转到 Surfer8,老的 386 笔记本退役而不知所终,而里面许多宝贵的东西竟然没有备份出来!虽然那台笔记本没有光驱,软驱也不能使用,但我明明记得有个 113 程序的(多么强大的一个小程序)。当我发现 plt 文件仍然可以在 Windows 下发挥大作用时,我找不到 PlotCall 的详细指令了。

今天我又找到了 PlotCall 命令,原来除了我一直使用的 MA, PA, PS, SP 命令,还有其他几个很酷的命令。

=====

PLOTCALL 命令:

命令: 绝对移动

格式: MA X,Y

其中

MA 是命令名称,他将画笔抬起并移动到 (X,Y)

X,Y 是画笔要移动的目标坐标,默认单位为英寸

命令: 绝对绘图

格式: PA X,Y

其中

PA 是命令名称,放下画笔,从当前位置画到 (X,Y)

X,Y 是画笔要移动的目标坐标,默认单位为英寸

命令: 位移

格式: TR DX,DY

其中

TR 这个命令位移所绘制的图形

DX, DY x 和 y 方向的位移量

命令：比例

格式：SC XSC, YSC

其中

SC 这个命令用来缩放图形

XSC X 方向的比例系数

YSC Y 方向的比例系数。

命令：绘制字符串

格式：PS X, Y, HT, AN, STR

其中

X, Y 第一个字符的左下角坐标，如果使用了居中符号集，则 X, Y 是符号的中心

HT 符号的高度，默认单位英寸

AN 字符串的旋转角度

STR 要绘制字符串，要用半角的双引号括住

命令：设置符号集

格式：SS SYMBLSET

其中

SS 这个命令用于设置当前符号集

SYMBLSET 符号集名称，要用双引号括住

命令：旋转

格式：R0 THETA

其中

R0 这个命令用于设置旋转角度，本命令以后的所有命令都经过旋转再执行

THETA 要旋转的角度

命令：极轴

格式：PI XPIV, YPIV

其中

PI 这个命令用于设置要旋转和缩放图形的点

XPIV, YPIV 极轴点的坐标

命令：选择画笔

格式：SP PN

其中

PN 画笔号，每个画笔可拥有独立的颜色。

=====

灵活的 BNA 格式

作者：Holz (hemengui [at] gmail [dot] com)

参考：Surfer 8 帮助

BNA 格式是一种地图边界文件，也是 ASCII 文件。这种文件能够用于存储包括区域、曲线、椭圆和点在内的地质地图信息，是一种相当灵活的交换格式。

BNA 文件用逗号分隔，它的一般格式如下：

"Pname 1", "Sname 1", type/length
x1, y1

```
x2, y2
...
xn, yn
"Pname 2", "Sname 2", type/length
x1, y1
x2, y2
...
xn, yn
```

在上面的文件格式中：

- Pname 是主 ID 的名称，用于链接对象到外部数据；
- Sname 是副 ID 名，是一个可选的参数；
- type/length 是一个不等于 0 的整数，用于表示对象类型，同时其绝对值表示后面跟着的坐标对数目。
- 跟在 type/length 后面的是实际的 X,Y 坐标对，即对象的各拐点位置，可以是整数或实数。要记得每行只放一个坐标对

那么 type/length 参数如何告诉我们将要建立的是一个什么类型的对象呢？有下面几种情况：

1. 当 type/length 是一个大于 2 的正整数时，表示后面的坐标对将建立一个封闭的区域。岛屿和湖泊就是这样的对象。
2. 当 type/length 是一个小于 -1 的负整数时，表示后面的坐标对将建立一个曲线对象。
3. 当 type/length 是 2 时，表示后面的坐标对将建立一个椭圆，其中第一个坐标对表示椭圆的中心，第二个坐标对分别表示椭圆的长轴和短轴。如果短轴为 0，表示一个圆。
4. 如果 type/length 等于 1，该对象为一个点，因为后面只能跟一个坐标对。
5. 如果 type/length 等于 -1，这种情况不会发生，因为这是错误的！记得 type/length 可以是 -1 和 0 以外的任何整数。

为什么说 BNA 格式是灵活的呢？因为它可以表现两种不同类型的区域：**独立区域**和**组合区域**。

独立区域，或者说一个简单的区域，首先有一个起点，然后是一系列拐点，最后终点回到起点；而组合区域，或者说一个复合区域呢，它可以包含若干子区域，例如岛屿、湖泊。BNA 文件使用一个非常特别的技术来表示这种由若干子区域组合得到的一个复合区域。

好吧，上面说的好好像比较晦涩，我们以实例说话。让我们来作一个假设，假设我们有一个复合区域，外面有一个封闭的区域，称之为“A”区，里面有两个小

岛，分别叫 “B” 和 “C”。现在我们来看看 BNA 文件是如何来描述这样一个复合区域的：

```
AX1, AY1      "A" 区的起点
AX2, AY2      "A" 区的另一个拐点
.
.
AXn, AYn      "A" 区的终点
BX1, BY1      "B" 区的起点
BX2, BY2      "B" 区的其他拐点
.
.
BXn, BYn      "B" 区的终点
AX1, AY1      "A" 区的起点(标志点)
CX1, CY1      "C" 区的起点
CX2, CY2      "C" 区的其他拐点
.
.
CXn, CYn      "C" 区的终点
AX1, AY1      "A" 区的起点(标志点)
```

要注意每个区域的终点都必需等于其起点，而 “A” 区的起点作为一个记号（称之为标志点）用于表示每个子区域的结束。这有什么讲究呢？这样意味着 A 区的每个点都必需唯一并且不能出现在任何子区域内，否则这个文件是错误的！

前面我们说了，要每行放一个坐标对，其实，BNA 格式并不限于于此，您完全可以将所有数据放在一行中，用逗号分隔开就可以了，例如这样：

```
"Pname 1", "Sname 1", type/length, x1, y1, x2, y2, x3, y3, ... xn, yn
"Pname 2", "Sname 2", type/length, x1, y1, x2, y2, x3, y3, ... xn, yn
```

前面都是理论，下面我们来看一些实际的数据，首先来看一个多边形，这个多边形同时具有主副两个 ID：

```
"C0", "Colorado", 7
-102.073, 37.0033
-103.015, 37.01
-103.106, 37.01
-109.064, 37.0033
-109.08, 41.0091
```

-102.081, 41.0157
-102.073, 37.0033

再来看一个曲线的对象，这个对象没有主副 ID，因为 Surfer 目前并没有使用到这两个参数，所以只要我们在 Golden Software 系列软件中使用，都不需要这两个 ID：

"" , "" , -10
535.954, 1681.83
541.209, 1241.31
1166.1, 1466.83
672.581, 1723.87
604.268, 1351.66
1260.25, 1089.36
1260.25, 1571.92
535.954, 1792.18
132.202, 1262.33
1260.25, 837.565

然后是一个点对象，此对象拥有主副 ID：

"Well HA-11", "Hansford Field", 1
-104.351, 37.847

BNA 文件可以用在 [Golden Software](#) 系列软件当中，例如 Surfer、Grapher 等。

神奇的 CLR 文件

作者：Holz (hemengui [at] gmail [dot] com)

参考：Surfer 8 帮助

CLR 是一种颜色谱文件格式，用于定义一个谱系，或者说一个连续梯度变化的颜色。CLR 文件本身并不知道所要上色的等级值，但它能够自动将您的地图变得五彩缤纷。

CLR 实际上定义了一系列的颜色锚点和相应的颜色，锚点之间的颜色通过插值的方式自动获得。

CLR 文件是一个文本文件，第一行为文件头，后面各行指定锚点，每个锚点占一行。

颜色谱文件的文件头暂时是固定的，就是 “ColorMap m n”，其中 ColorMap 是颜色谱文件的标识符，是区分大小写的，因此不能写错，写为 colormap 或者 COLORMAP 都是不对的。m 是颜色谱文件的格式版本号，目前应该为 1。n 是节点之间的插值方法，目前也应该设为 1。

文件头后面是具体的数据，有四列：

第一列是锚点的位置，用百分点来表示，因此取值范围从 0 到 100。

第二列是颜色的红色分量，从 0 到 255 之间取值。

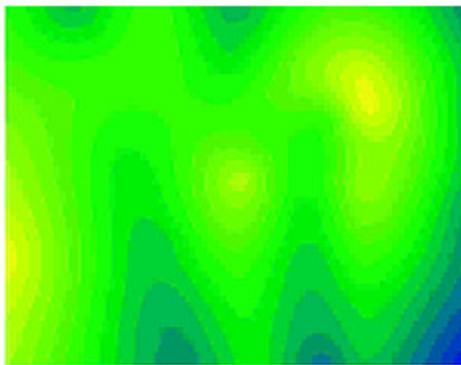
第三列是颜色的绿色分量，从 0 到 255 之间取值。

第四列是颜色的蓝色分量，从 0 到 255 之间取值。

来看一个例子：

```
ColorMap 1 1
0 0 0 255
50 0 255 0
100 255 255 0
```

在这个范例中，0%位置处是蓝色；50%处是绿色；100%处是黄色。
看看这个颜色谱文件生成的效果：

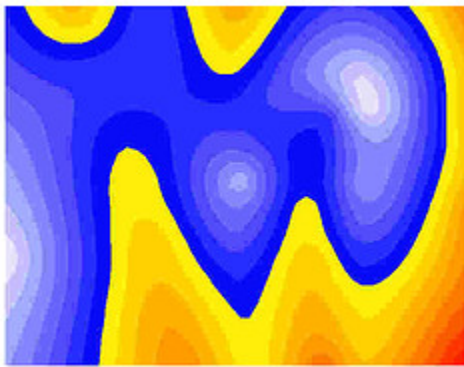


再来看看一个高级一些的例子：

```
ColorMap 1 1
0 255 0 0
50 255 255 0
50 0 0 255
100 255 255 255
```

这个文件告诉我们，要在 50%的地方出现颜色断层，就是在 0-50%之间使用红色和黄色插值，在 50-100%之间用蓝色和白色插值。

看看效果：

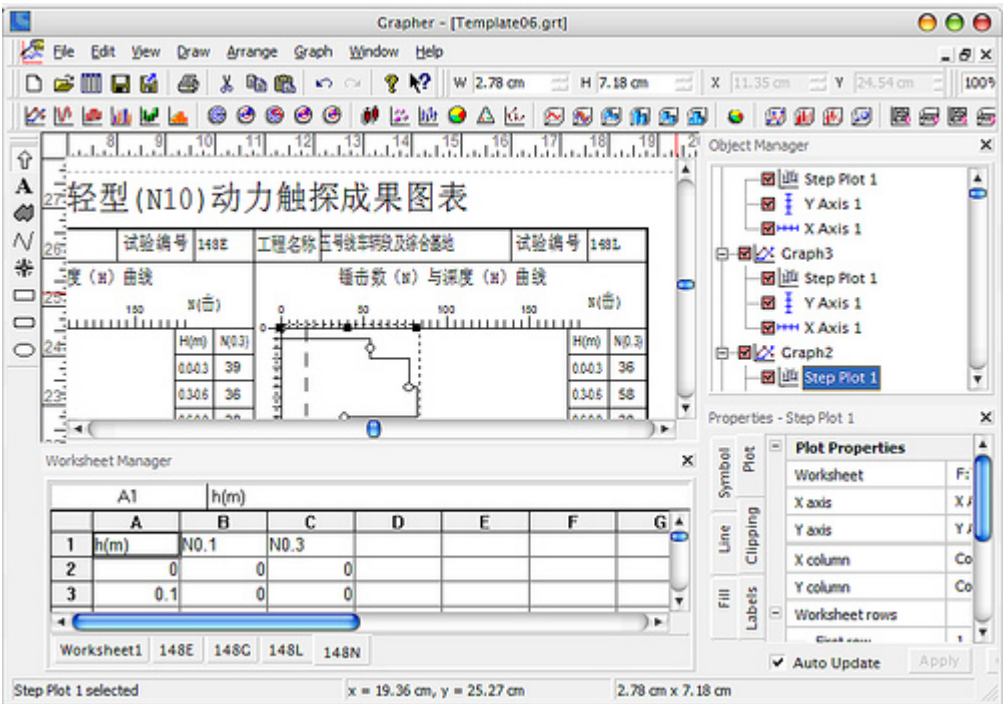


CLR 文件一般用于 [Golden Software Surfer](#) 中。

Golden Software Grapher 6

[Golden Software Grapher 6](#) 是一个高效而强大的图表绘制系统，能够制作大多数简单或复杂的图表。

Grapher 6 也是一个容易理解和学习的学习绘图软件包，能够快速便捷地制作出高精度的图表。



在 Grapher 6 中可以建立超过 三十 种不同的二维、三维图表类型，所建立的图表的任何部分都可以根据需要自行定义参数、属性等。如果一定要将 Grapher 6 的图表类型分类，那么他们可以粗略分了七个大类，看下表。

线性图表	条形图	极坐标图表	专业图表	3D 形态的 XYZ 图表	3D XYZ 图 表	等值线和表 面图
折线图	条形图	极坐标图（折线/散点）	盘高-盘低-收盘图	丝带/墙体图	3D 折线/散点图	XY 等值线数据图
散点图	悬浮条形图	极坐标条形图	饼图	阶梯图	3D 条形图	XY 等值线网格图
阶梯图	直方图	极坐标函数图	三角图	函数图	3D 气泡图	XZ 等值线数据图
函数图		玫瑰图	箱线图	条形图	3D 悬浮条形图	XZ 等值线网格图
		风图	气泡图	悬浮条形图		表面数据图
			矢量图	直方图		表面网格图
				饼图		表面函数图

Grapher 6 中一个快捷的绘图方式是图表向导，是一个相当不错的工具。
Grapher 6 还可以通过对象管理器来更好地组织和管理图形文档中的各类对象。

在这个版本中，终于可以将您绘制的图表保存为之前版本的格式，为文件交流提供了便利，在 Grapher 6 中你可以将一个图表保存为 Grapher 3、Grapher 4、Grapher 5 或 Grapher 6 的 GRF 格式，也可以保存为 Grapher 4、Grapher 5 或 Grapher 6 的 GPJ 格式。

Grapher 6 的输出功能也得到增强，现在不但可以输出 eps，也可以输出为 PDF 格式了，这是个不小的进步！虽然输出 pdf 格式的功能尚待改进。

Grapher 6 的图表类型

如果你手头有一个数据，那么 Golden Software Grapher 6 可以用超过 30 种图表类型中的一种来表现您的数据，还可以用一个全新的方式来组合多种不同类型的图表类型来表现您的数据。

- 折线/散点图可以用折线或者符合来表现数据，这是一种传统的二维数据表现形式。
范例参考：`samples -> lineplot.grf`
- 丝带/墙体图可以说是三维形式的折线/散点图，用于表现数据的空间定位，可以添加后墙投影、投影线等。

- 阶梯图通过阶梯线来表现数据，可以是二维或三维的。范例参考：`samples -> stepplot.grf`
- 气泡图用不同大小的符号来表现数据，符号的大小由某个变量来控制。气泡图的符号是可以任意设置的，在 **Grapher 6** 中，可以用二维或三维气泡图来表示变量的改变，符号可以填充纯颜色、梯度颜色等，从而建立美观的图表。范例参考：`samples -> bubble.grf`
- 条形图有一般的条形图和悬浮条形图，有二维条形图和三维条形图。条形图用条形来表现数据，有点类似直方图。悬浮条形图则还可以用于表现两个变量之间的差异。
- 函数图用来显示函数的特征，这需要你定义函数表达式并指出初始变量，当然函数图也可以是二维或三维的，甚至还可以使用极坐标系统。
- 盘高-盘低-收盘图显示在每个 **X** 值处的 **Y** 范围，最常用的形式是用来表现股票交易。盘高-盘低-开盘-收盘图可以用标准的线条来表示，也可以用条形来表示（此时称之为“烛台”图）。
- 极坐标图显示角度和半径变量，可以用线条表现也可以用条形表现，也可以表示一个极坐标函数特征。
- 玫瑰图用来表现[角度, 半径]或者[梯度, 径向]的变量，有点类似直方图，因为他也是用来表现统计数据的。在玫瑰图中，分组也是完全可以定制的。
- 风图表现频率和方向，分组根据风速建立，频率可以按计数、相对频率、相对频率百分数等形式查看，分组也可以自由定制。
- 饼图很容易理解，总数是一个圆饼，每个数值在总数中所占的比例控制饼块的大小。饼图可以是二维的或者是三维的。
- 三角图如果叫三坐标系图或者叫三分量图可能比较容易理解，叫三角图只是因为一般情况下其形状如此。三角图需要三个相关的分量。
- 直方图是个统计图，**x** 方向是数值的分组，**y** 方向是相对频率，可以添加高斯拟合检查正态分布。
- 箱线图通常用来表现最大值、最小值、中值、下四分位数（第一四分位数）和上四分位数（第三四分位数）。
- 矢量图用来表现变量的起点和终点，例如风、电流等。**Grapher** 的矢量图允许设置开始的 **XY** 位置，而结束的位置可以按角度、幅值或者终点 **XY** 位置确定。
- 等值线图是 **Grapher 6** 新增加的，与 **Surfer** 不同的是，**Grapher 6** 可以直接由数据得到等值线图。不过不要以为从此可以抛弃 **Surfer**，因为目前 **Grapher 6** 的等值线图目前只能使用克里格网格化，这是远远不能与 **Surfer** 相比拟的。
- 表面图也是 **Grapher 6** 新增的，同样可以直接从数据得到图形，缺点同等值线图。

surfer 中文帮助

做点尝试

Surfer 自动化

看看 drupal 书籍的目录结构如何处理。

对象

Surfer 自动化对象。

Application 对象

Application 对象指向 Surfer 程序。通过 Surfer 对象模型进行的所有导航都开始于 Application 对象。Application 对象是 Surfer 的一个独立实例并且是 Surfer 中所有对象的根。外部程序在初始化的时候将典型的创建 Application 对象的一个实例。在 VB 中使用 CreateObject 函数：

```
Set SurferApp = CreateObject("Surfer.Application")
```

CreateObject 函数激活一个新的 Surfer 实例并返回一个指向 Application 对象的参考给你的脚本。如果 Surfer 已经在运行，而你不想创建一个新的 Surfer 实例，使用 VB 的 GetObject 函数代替 CreateObject：

```
Dim srf As Object  
Set srf = GetObject(, "Surfer.Application")
```

GetObject 函数获取 Application 对象自当前运行的 Surfer 实例。如果 Surfer 尚未运行，本函数将失败。调用 Application 对象的 [Quit](#) 方法从一个脚本关闭 Surfer。

当 Surfer 被一个脚本启动时，主窗口最初是隐藏的，为了使 Surfer 窗口可见，你必需设置 Application 对象的 [Visible](#) 属性为 True：

```
Set srf = GetObject(, "Surfer.Application")  
srf.Visible = True
```

大多数的网格相关的操作都是 Application 对象的方法。使用 GridData、GridFunction、GridMath、GridFilter、GridSplineSmooth、GridBlank、GridConvert、GridVolume、GridSlice、GridTransform、GridExtract 和 GridCalculus 方法来创建新的网格和执行网格计算。

其他 Application 对象的方法和属性能够移动和调整 Surfer 窗口，例如调整主窗口状态（最大化、最小化、隐藏）、改变窗口标题以及修改参数设置等。

Application 对象提供两个重要的集合来访问谱系中的下一级对象。使用 [Documents](#) 属性来获取 Documents 集合对象的参考，而 [Windows](#) 属性则可以获取 Windows 集合对象的参考。Documents 和 Windows 集合对象提供了对谱系中其他对象的访问。你可以用 Application 对象的 ActiveDocument 和 ActiveWindow 属性访问当前活动的文档和窗口对象。

设置 [ScreenUpdating](#) 属性为 False 可以禁止更新屏幕并且可以使你的脚本更加快速的执行。 在你执行一系列导致屏幕刷新的操作的时候，关闭屏幕更新可以大大提高自动化性能。

属性

[ActiveDocument](#)
[ActiveWindow](#)
[Application](#)
[BackupFiles](#)
[Caption](#)
[DefaultFilePath](#)
[Documents](#)
[FullName](#)
[Height](#)
[Left](#)
[Name*](#)
[PageUnits](#)
[Parent](#)
[Path](#)
[ScreenUpdating](#)
[ShowStatusBar](#)
[ShowToolbars](#)
[Top](#)
[Version](#)
[Visible](#)
[Width](#)
[Windows](#)
[WindowState](#)

方法

[CrossValidate](#)
[GridBlank](#)
[GridCalculus](#)
[GridConvert](#)
[GridData](#)
[GridExtract](#)
[GridFilter](#)
[GridFunction](#)
[GridMath](#)
[GridMosaic](#)
[GridResiduals](#)
[GridSlice](#)
[GridSplineSmooth](#)
[GridTransform](#)

[GridVolume](#)

[NewGrid](#)

[NewVarioComponent](#)

[Quit](#)

*默认属性

范例 1

本脚本[samples\ApplicationObjectProperties.bas] 在 Scripter 中打开并显示 Application 对象的所有属性。要运行脚本可以：单击 脚本 | 运行；单击绿色箭头；或者在你的键盘按 F5。

范例 2

本脚本[samples\ApplicationObjectMethods.bas] 在 Scripter 打开并显示 Application 对象的所有方法。要运行脚本可以：单击 脚本 | 运行；单击绿色箭头；或者在你的键盘上按 F5。

范例 3

下列脚本示范如何使用 Application 对象。

```
Sub Main
' 定义 SurferApp 为一个对象
Dim SurferApp As Object
' 创建 Surfer 应用程序的实例对象并指定
' 给变量 "SurferApp"
Set SurferApp = CreateObject("Surfer.Application")
' 使 Surfer 可见
SurferApp.Visible = True
End Sub
```

Documents 集合

Documents 集合

Documents 集合包含当前打开的所有文档。Documents 集合提供访问图形和工作表文档的途径。

用 [Add](#) 方法可以创建新的图形或工作表文档。

Surfer 提供了 [预定义值](#) srfDocPlot 和 srfDocWks 来指定要创建的 [文档](#) 类型：

```
Set srf = CreateObject("Surfer.Application")
```

srf.Documents.Add srfDocPlot ' 创建一个新的, 空白的图形文档
srf.Documents.Add srfDocWks ' 创建一个新的, 空白的工作表文档

Add 方法返回一个指向被创建的 Document 对象的参考, 新的网格文档不能由 Documents 集合的 Add 方法创建。你可以通过调用 [Application](#) 对象的网格相关对象来创建。

用 [Open](#) 方法可以打开存在的图形、工作表和网格文件。Open 方法返回一个指向打开的文档的参考。

用 [Item](#) 方法获取一个打开的图形或工作表文档对象的参考。如果在文档的集合中进行循环, 文档的 [Type](#) 和 [Index](#) 属性可以用于检测集合中文档的类型和位置。

属性

[Application](#)

[Count](#)

[Parent](#)

方法

[Add](#)

[CloseAll](#)

[Item](#)*

[Open](#)

[SaveAll](#)

*默认方法

范例 1

本脚本(samples\documentscollection.bas)在 Scripter 打开并显示 Documents 集合的所有属性。要运行脚本, 单击 脚本 | 运行、点击绿色箭头, 或者在键盘上按 F5。

范例 2

下列脚本示范如何使用 [Application](#) 对象的 Documents 集合。

```
Sub Main
    ' 定义 SurferApp 为一个对象
    Dim SurferApp As Object
    ' 创建一个 Sufer 应用程序对象实例并指定
    ' 给变量 "SurferApp"
    Set SurferApp = CreateObject("Surfer.Application")
    ' 使 Surfer 可见
    SurferApp.Visible = True
    ' 定义 Docs 为一个对象
    Dim Docs As Object
    ' 指定 Documents 集合给变量 "Docs"
    Set Docs = SurferApp.Documents
End Sub
```

Windows 集合

Windows 集合

Windows 集合提供对 Surfer 所有窗口的访问。Windows 集合返回的 [Application](#) 对象包含应用程序的所有窗口，包括 [图形窗口](#)、[工作表窗口](#) 以及 [网格编辑器](#) 视图。创建新的 Window 对象可以调用 [Document](#) 对象的 [NewWindow](#) 方法。

Windows 集合通过 [PlotDocument](#) 对象 返回父图形文档中打开的所有窗口。

Windows 集合通过 [WksDocument](#) 对象返回父工作表文档打开的所有窗口。

属性

[Application](#)

[Count](#)

[Parent](#)

方法

[Arrange](#)

[Item](#)*

*默认方法

范例 1

本 脚本（samples\windows.bas） 打开于 **Scripter** 并显示 Windows 集合的所有属性。 要运行脚本， 点击 脚本 | 运行, 点击绿色箭头， 或者在键盘上按 F5。

范例 2

下列脚本示范如何应用 Windows 集合于 [Application 对象](#)。

```
Sub Main
    ' 定义 SurferApp 为一个对象
    Dim SurferApp As Object
    ' 创建一个 Sufer 应用程序对象实例并指定
    ' 给变量 "SurferApp"
    Set SurferApp = CreateObject("Surfer.Application")
    ' 使 Surfer 可见
    SurferApp.Visible = True
    ' 定义 Windows 为一个对象
    Dim Windows As Object
    ' 指定 Windows 集合给变量 "Windows"
    Set Windows = SurferApp.Windows
End Sub
```

范例 3

下列脚本示范如何应用 Windows 于 [PlotDocument 对象](#) 的第一个文档集合。

```
Sub Main
```

```

' 定义 SurferApp 为一个对象
Dim SurferApp As Object
' 创建一个 Sufer 应用程序对象实例并指定
' 给变量 "SurferApp"
Set SurferApp = CreateObject("Surfer.Application")
' 使 Surfer 可见
SurferApp.Visible = True
' 定义 Plot 为一个对象
Dim Plot As Object
' 创建一个 Surfer 图形文档并指定给变量
' "Plot"
Set Plot = SurferApp.Documents.Add(srfDocPlot)
' 定义 Windows 为一个对象
Dim Windows As Object
' 指定 Windows 集合给变量 "Windows"
Set Windows = Plot.Windows(1)
End Sub

```

范例 4

下列脚本示范如何应用 **Windows** 于 [WksDocument](#) 对象。

```

Sub Main
' 定义 SurferApp 为一个对象
Dim SurferApp As Object
' 创建一个 Sufer 应用程序对象实例并指定
' 给变量 "SurferApp"
Set SurferApp = CreateObject("Surfer.Application")
' 使 Surfer 可见
SurferApp.Visible = True
' 定义 Wks 为一个对象
Dim Wks As Object
' 创建一个 Surfer 工作表文档并指定给变量
' "Wks"
Set Wks = SurferApp.Documents.Add(srfDocWks)
' 定义 Windows 为一个对象
Dim Windows As Object
' 指定 Windows 集合给变量 "Windows"
Set Windows = Wks.Windows
End Sub

```

方法和属性

Surfer 自动化对象的方法和属性。

ActiveDocument 属性

ActiveDocument 属性

ActiveDocument 返回活动的文档对象。 返回一个对象。 这是一个只读的属性。

语法

object.**ActiveDocument**

范例

本范例示范如何返回一个活动的文档。

Debug.Print SurferApp.ActiveDocument

用于

[Application 对象](#)

ActiveWindow 属性

ActiveWindow 属性

ActiveWindow 返回活动窗口对象。 返回一个对象。 这是一个只读属性。

语法

object.**ActiveWindow**

范例

本范例示范如何返回活动的窗口。

Debug.Print SurferApp.ActiveWindow

用于

[Application 对象](#)

Application 属性

Application 属性

Application 返回一个 [Application](#) 对象。 这是一个只读属性。

语法

object.**Application**

范例

本范例示范如何返回 Surfer 应用程序对象。

Debug.Print SurferApp.Application

用于

[Application 对象](#), [Axes 集合](#), [ColorMap 对象](#), [Document 对象](#), [Documents 集合](#), [FillFormat 对象](#), [FontFormat 对象](#), [GLFillFormat 对象](#), [Grid 对象](#), [LabelFormat 对象](#), [Level 对象](#), [Levels 集合](#), [LineFormat 对象](#), [MarkerFormat 对象](#), [Overlays 集合](#), [PageSetup 对象](#), [Ruler 对象](#), [ScaleBars 集合](#), [Selection 集合](#), [Shape 对象](#), [Shapes 集合](#), [VarioComponent 对象](#), [Window 对象](#), [Windows 集合](#)

BackupFiles 属性

BackupFiles 属性

BackupFiles 返回或设置全局文件 [备份状态](#)。 返回一个布尔值。

语法

object.BackupFiles ***object.BackupFiles = Backup***

参数	类型	说明
----	----	----

Backup	布尔值	需要
--------	-----	----

范例 1

本范例示范如何返回是否在保存文件之前创建一个备份。

Debug.Print SurferApp.BackupFiles

范例 2

本范例示范如何允许在保存文件之前使用 .BAK 后缀保存前一个版本的一个副本。

SurferApp.BackupFiles = True

用于

[Application](#) 对象

Quit 方法

Quit

Quit 方法终止应用程序。

语法

object.Quit

范例

本范例示范如何关闭 Surfer 应用程序。

SurferApp.Quit

用于

[Application](#) 对象

ScreenUpdating 属性

ScreenUpdating 属性

ScreenUpdating 返回或设置所有视图窗口的 [刷新](#) 标志。 返回一个布尔值。

语法

object.ScreenUpdating ***object.ScreenUpdating = Update***

参数	类型	说明
----	----	----

Update	布尔值	必需
--------	-----	----

备注

在需要执行大量导致屏幕刷新的操作的时候关闭屏幕更新将大大提高自动化性能。可以通过设置 ScreenUpdating 为 false 以达到提高性能的目的，执行各项操作，然后重新打开 ScreenUpdating 。 当 ScreenUpdating 被重新打开，所有的窗口都被刷新到一个新的状态。

范例 1

本范例示范如何返回所有窗口的自动刷新状态。

```
Debug.Print SurferApp.ScreenUpdating
```

范例 2

本范例示范如何自动刷新 **Surfer** 的每个窗口。

```
SurferApp.ScreenUpdating = True
```

用于

[Application](#) 对象

Visible 属性

Visible 属性 [Application]

Visible 返回或设置应用程序窗口可见性。返回一个布尔值。

语法

```
object.Visible object.Visible = bVisible
```

参数	类型	说明
----	----	----

bVisible	布尔值	必需
-----------------	-----	----

范例 1

本范例示范如何返回是否 **Surfer** 应用程序可见。

```
Debug.Print SurferApp.Visible
```

范例 2

本范例示范如何显示 **Surfer** 应用程序。

```
SurferApp.Visible = True
```

用于

[Application](#) 对象

如何在 Grapher 2/3/4/5/6 输入中文

自从 Golden Software 将 Grapher 更新到 2 版以上,到如今已经是 Grapher 6 的天下了。但 Grapher 不能正确输入和显示中文的问题,贯穿了 2-6 的每个版本,如今能正确处理中文的只有 Grapher 1 了。参考 Grapher 4 中文教程：<http://www.xdcad.net/forum/showthread.php?s=&threadid=39437> 参考 PlotCall 文件格式：<http://holz.byethost15.com/PlotCall> 参考 Grapher 版本历史：<http://www.goldensoftware.com/grapherhistory.shtml> 参考 Grapher 常见问题：<http://www.goldensoftware.com/faq/grapher-faq.shtml>
http://holz.byethost15.com/grapher_5_tips_faq

利用旧版 Grapher

既然 Grapher 1 可以正确输入和显示中文,那么我们就在 Grapher 1 中输入中文,然后保存为 grf 文件。到 Grapher 2/3/4/5/6 中打开这旧版本的 grf 文件,发现中文显示正常。

再次编辑

更进一步，发现旧版本中的文本用新版本的 Grapher 打开后，其中的文本对象可以随意复制、再编辑，以及等等，而不会再出现乱码现象。这很好地解决了中文字符问题。

使用 OLE 对象

再继续研究 Grapher，发现可以使用 OLE 对象。那么我们就很容易想到，插入一个可以正确处理中文字符的 OLE 对象，不就解决问题了吗？例如 M\$ Word。不过这种方法会大大增加 grf 文件的尺寸。

使用交换文件格式

如果是从 DOS 下用 Grapher 过来的人，或者还记得强大的 PlotCall 文件格式，这种格式允许你绘制文本，所以只要在 .plt 文件中输入中文，到 Grapher 中输入就可以了。

利用日期时间格式

还有一种无意中发现的，不用的人说傻，要用到的人都说好。原来 Grapher 增加了日期/时间格式，本意是让用户可以方便地输入各类标注格式，不过居然可以自由增加样式。可想而知，某天老夫不小心输入了汉字。

数学文本指令

数学文本指令集提供了对文本的高级格式化能力，它可以改变字体、大小、颜色、粗细、风格等等。

用数学文本指令可以书写希腊字母和数学公式；它还提供了详细的字符和符号位置控制方式。因此通过数学文本指令可以轻易做出上标、下标的形式。

如何找到需要的符号

如果要使用数学文本指令 \aX 进行符号标注，或者书写数学公式，那么如何知道某个符号对应的字符代码呢？

第一个方法是建立一个三列数据文件，然后用 Surfer 建立一个张贴图，只要将符号列设置为第三列即可。

数据如下，显示第 0 到 255 个字符

x	y	char_code
---	---	-----------

0	19	0
1	19	1
2	19	2
3	19	3
4	19	4
5	19	5
6	19	6
7	19	7
8	19	8
9	19	9
10	19	10
11	19	11
12	19	12
0	18	13
1	18	14
2	18	15
3	18	16
4	18	17
5	18	18
6	18	19
7	18	20
8	18	21
9	18	22
10	18	23
11	18	24
12	18	25
0	17	26
1	17	27
2	17	28
3	17	29
4	17	30
5	17	31
6	17	32
7	17	33
8	17	34
9	17	35
10	17	36
11	17	37
12	17	38
0	16	39
1	16	40
2	16	41
3	16	42
4	16	43

5	16	44
6	16	45
7	16	46
8	16	47
9	16	48
10	16	49
11	16	50
12	16	51
0	15	52
1	15	53
2	15	54
3	15	55
4	15	56
5	15	57
6	15	58
7	15	59
8	15	60
9	15	61
10	15	62
11	15	63
12	15	64
0	14	65
1	14	66
2	14	67
3	14	68
4	14	69
5	14	70
6	14	71
7	14	72
8	14	73
9	14	74
10	14	75
11	14	76
12	14	77
0	13	78
1	13	79
2	13	80
3	13	81
4	13	82
5	13	83
6	13	84
7	13	85
8	13	86
9	13	87

10	13	88
11	13	89
12	13	90
0	12	91
1	12	92
2	12	93
3	12	94
4	12	95
5	12	96
6	12	97
7	12	98
8	12	99
9	12	100
10	12	101
11	12	102
12	12	103
0	11	104
1	11	105
2	11	106
3	11	107
4	11	108
5	11	109
6	11	110
7	11	111
8	11	112
9	11	113
10	11	114
11	11	115
12	11	116
0	10	117
1	10	118
2	10	119
3	10	120
4	10	121
5	10	122
6	10	123
7	10	124
8	10	125
9	10	126
10	10	127
11	10	128
12	10	129
0	9	130
1	9	131

2	9	132
3	9	133
4	9	134
5	9	135
6	9	136
7	9	137
8	9	138
9	9	139
10	9	140
11	9	141
12	9	142
0	8	143
1	8	144
2	8	145
3	8	146
4	8	147
5	8	148
6	8	149
7	8	150
8	8	151
9	8	152
10	8	153
11	8	154
12	8	155
0	7	156
1	7	157
2	7	158
3	7	159
4	7	160
5	7	161
6	7	162
7	7	163
8	7	164
9	7	165
10	7	166
11	7	167
12	7	168
0	6	169
1	6	170
2	6	171
3	6	172
4	6	173
5	6	174
6	6	175

7	6	176
8	6	177
9	6	178
10	6	179
11	6	180
12	6	181
0	5	182
1	5	183
2	5	184
3	5	185
4	5	186
5	5	187
6	5	188
7	5	189
8	5	190
9	5	191
10	5	192
11	5	193
12	5	194
0	4	195
1	4	196
2	4	197
3	4	198
4	4	199
5	4	200
6	4	201
7	4	202
8	4	203
9	4	204
10	4	205
11	4	206
12	4	207
0	3	208
1	3	209
2	3	210
3	3	211
4	3	212
5	3	213
6	3	214
7	3	215
8	3	216
9	3	217
10	3	218
11	3	219

12	3	220
0	2	221
1	2	222
2	2	223
3	2	224
4	2	225
5	2	226
6	2	227
7	2	228
8	2	229
9	2	230
10	2	231
11	2	232
12	2	233
0	1	234
1	1	235
2	1	236
3	1	237
4	1	238
5	1	239
6	1	240
7	1	241
8	1	242
9	1	243
10	1	244
11	1	245
12	1	246
0	0	247
1	0	248
2	0	249
3	0	250
4	0	251
5	0	252
6	0	253
7	0	254
8	0	255

第二种方法使用文本块，新建一个图形文档，使用文本命令，在对话框中输入下面的代码：

```
\a1 \a2 \a3 \a4 \a5 \a6 \a7 \a8 \a9 \a10 \a11 \a12 \a13
\a14 \a15 \a16 \a17 \a18 \a19 \a20
```

\a21	\a22	\a23	\a24	\a25	\a26	\a27	\a28	\a29	\a30	\a31
\a32	\a33	\a34	\a35	\a36	\a37	\a38	\a39	\a40		
\a41	\a42	\a43	\a44	\a45	\a46	\a47	\a48	\a49	\a50	\a51
\a52	\a53	\a54	\a55	\a56	\a57	\a58	\a59	\a60		
\a61	\a62	\a63	\a64	\a65	\a66	\a67	\a68	\a69	\a70	\a71
\a72	\a73	\a74	\a75	\a76	\a77	\a78	\a79	\a80		
\a81	\a82	\a83	\a84	\a85	\a86	\a87	\a88	\a89	\a90	\a91
\a92	\a93	\a94	\a95	\a96	\a97	\a98	\a99	\a100		
\a101	\a102	\a103	\a104	\a105	\a106	\a107	\a108	\a109	\a110	
\a111	\a112	\a113	\a114	\a115	\a116	\a117	\a118	\a119	\a120	
\a121	\a122	\a123	\a124	\a125	\a126	\a127	\a128	\a129	\a130	
\a131	\a132	\a133	\a134	\a135	\a136	\a137	\a138	\a139	\a140	
\a141	\a142	\a143	\a144	\a145	\a146	\a147	\a148	\a149	\a150	
\a151	\a152	\a153	\a154	\a155	\a156	\a157	\a158	\a159	\a160	
\a161	\a162	\a163	\a164	\a165	\a166	\a167	\a168	\a169	\a170	
\a171	\a172	\a173	\a174	\a175	\a176	\a177	\a178	\a179	\a180	
\a181	\a182	\a183	\a184	\a185	\a186	\a187	\a188	\a189	\a190	
\a191	\a192	\a193	\a194	\a195	\a196	\a197	\a198	\a199	\a200	
\a201	\a202	\a203	\a204	\a205	\a206	\a207	\a208	\a209	\a210	
\a211	\a212	\a213	\a214	\a215	\a216	\a217	\a218	\a219	\a220	
\a221	\a222	\a223	\a224	\a225	\a226	\a227	\a228	\a229	\a230	
\a231	\a232	\a233	\a234	\a235	\a236	\a237	\a238	\a239	\a240	
\a241	\a242	\a243	\a244	\a245	\a246	\a247	\a248	\a249	\a250	
\a251	\a252	\a253	\a254	\a255						

上面两种方法都可以方便地得到某个符号集中所有符号，还可以将这些符号打印出来备查。

但上面两种方法有时候得到的结果稍微有点差别，例如对于 Symbol 符号集，用文本块的方法可以得到性别符号 ♂，但用张贴图方式就不会显示，让人误以为该代码处符号为空。这是在使用当中应当注意的地方。

数学文本指令列表

下面是一些基本的数学文本指令，在 Surfer 中用的比较多，非常强大的但是简单的东西。

注意：指令名是区分大小写的（\fs50, \FS50, \Fs50 或者 \fS50 都是非法的）。

1. 基本指令

- \aX 插入一个 ANSI 字符，X 为字符代码 (0 到 255)。
- \b 文本加粗。
- \date 插入日期。
- \dnX 向下移动当前字体大小的 X%。
- \f"X" 使用 X 字体。
- \fsX 改变字体大小为当前的 X%。
- \i 文本倾斜。
- \plain 设置无格式文本 (不加粗、不倾斜、无下划线、无删除线)。
- \rpX 恢复当前位置到第 #X (1-20) 个位置，当前行有效。
- \spX 保存当前位置为第 #X (1-20) 个位置，当前行有效。
- \strike 删除线。
- \time 插入时间。
- \upX 向上移动当前字体大小的 X%。
- \ul 下划线。

[注意：\sp 和 \rp 指令仅在当前行有效，例如，即使您在第一行保存了一个位置，也不能在第三行恢复到该位置。]

2. 下面的指令允许按 RGB (红、绿、蓝) 值设置文本颜色:

- \rgbrX 设置红色分量 (X=0 到 255)。
- \rgbgX 设置绿色分量 (X=0 到 255)。
- \rgbbX 设置蓝色分量 (X=0 到 255)。

3. 下面的指令提供快速的设置文本颜色的途径:

- \black 黑色
- \blue 蓝色
- \green 绿色
- \cyan 青色
- \red 红色
- \magenta 洋红
- \yellow 黄色
- \white 白色
- \gray 灰色

数学文本指令实例示范

首先来一个 Golden Software 的示范，启动 Surfer， 选择绘制菜单下的文本命令，在弹出的文本对话框中，输入下面的代码：

```
\sp1 {\fs200 \f"Symbol" \a229 \sp2 }{\rp1 \dn90 \fs75 i=1\rp1 \up220  
n}\rp2 \up25 X\dn50 {\fs75 i}\up50 Y\dn50 {\fs75 i}\up50 = S{\dn50 {\fs75  
XY}\up50 }
```

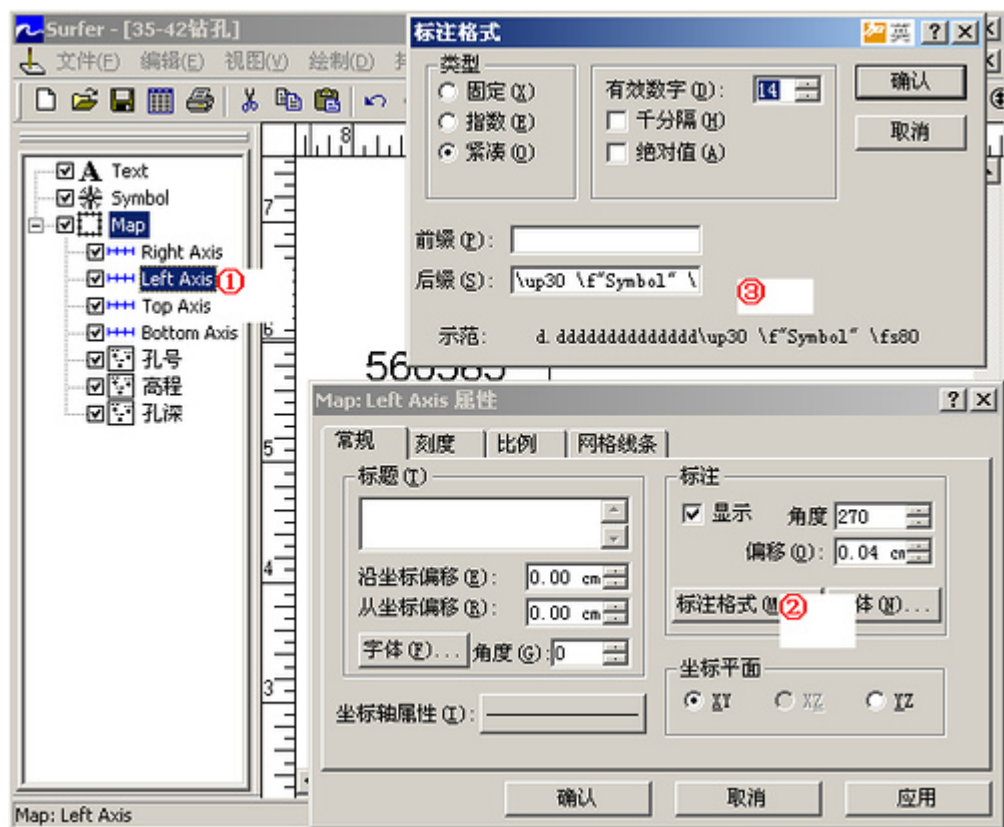
确认以后，得到结果如图所示：

$$\sum_{i=1}^n X_i Y_i = S_{XY}$$

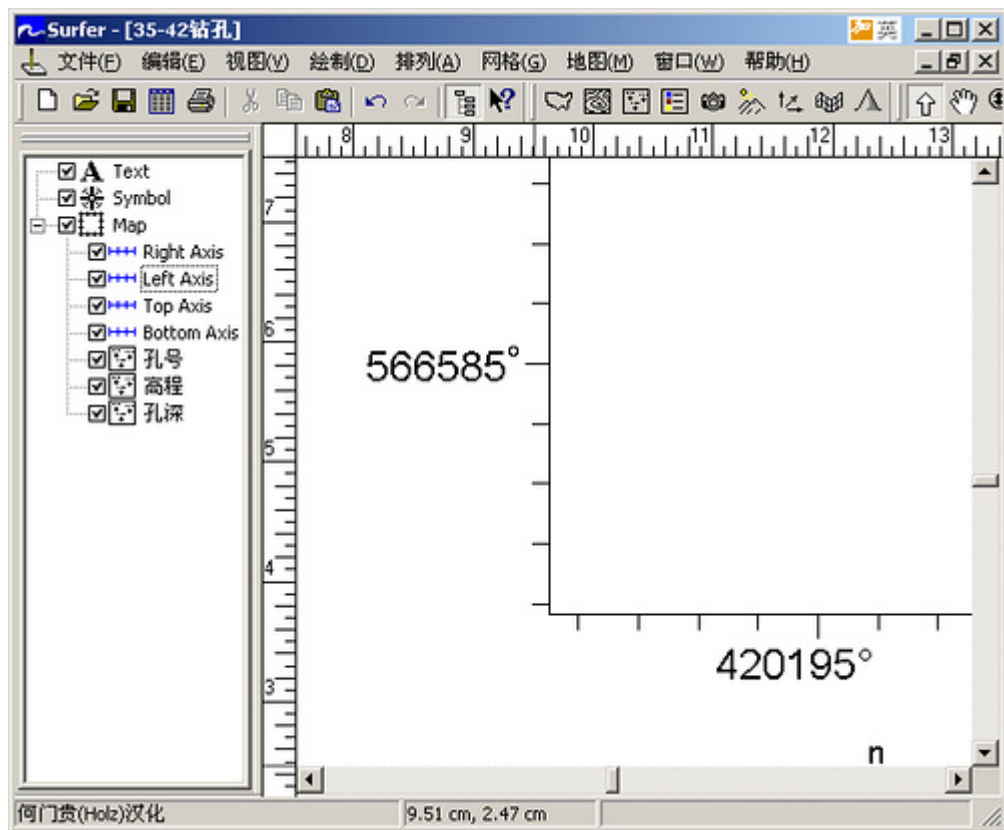
再来看一个坐标标注的例子，假设要为坐标的标注加一些修饰，例如单位，那么首先双击坐标，弹出属性对话框。在常规标签页中单击标注格式按钮，弹出格式对话框。如果要添加单位符号，那么在后缀里面输入

```
\up30 \f"Symbol" \fs80 \a176
```

操作步骤如图所示：



然后一路确认回去，得到结果如图：



数学文本指令的语法

在图形窗口中的所有文本都可以通过数学文本指令来控制，例如，坐标标注、标题、通过文本对话框建立的文本块等。用于标注的工作表单元格可以包含数学文本指令。

除了特别指出，所有数学文本指令都始于反斜杠 (“\”)，并结束于一个空格。例如 “\up50 ” 将文本向上移动 50% 的当前文本高度。所有以反斜杠开始并以空格结束的字符都通过数学文本解析器解析并执行，最后得到结果。

文本块中每行都有默认的属性如字体、大小、颜色、风格（注意一些字体例如 Symbol 是不支持加粗和倾斜的）。文本块中所有文本使用默认的属性，除非发现数学文本指令，在数学文本指令后面的属性将依据指令修改，例如一行的中间通过数学文本指令改变了字体，那么指令后面的文本将使用新的字体，直到行结束或者其他影响到字体改变的指令为止。

如果要数学文本指令不影响整行，那么可以用大括号来进行限制，左边的括号 (“{”) 表示数学文本域的开始，右边的括号 (“}”) 表示结束。但这个限制不包括基线指令 (\dnX 和 \upX) 和位置指令 (\rpX 和 \spX)。此外大括号可以嵌套使用。

数学文本指令带来一个问题，就是如何输入几个特殊的符号，即反斜杠和大括号。我们可以在这些符号前面加一个反斜杠来输入，例如 “\\” 表示 “\” 而 “\{” 表示 “{”。

基于百分数的指令如字体大小，是相互影响的。这意味着第二个百分数指令受到第一个百分数指令的影响，例如第一个指令将字体大小设置为 50%，第二个指令也设置为 50%，那么实际上第二个指令得到的字体大小是当前字体大小的 25%。