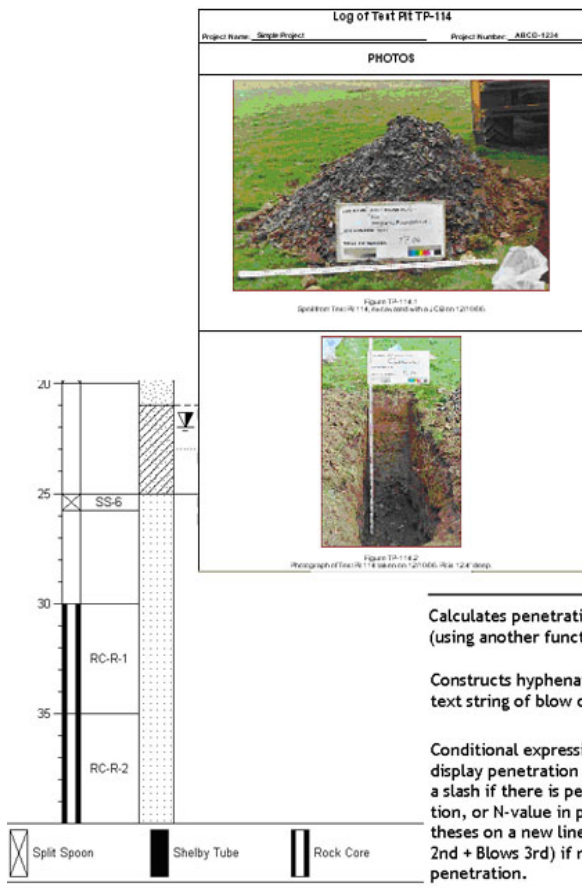
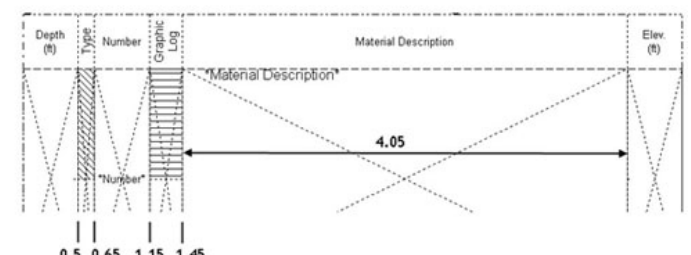


gINT Tutorial

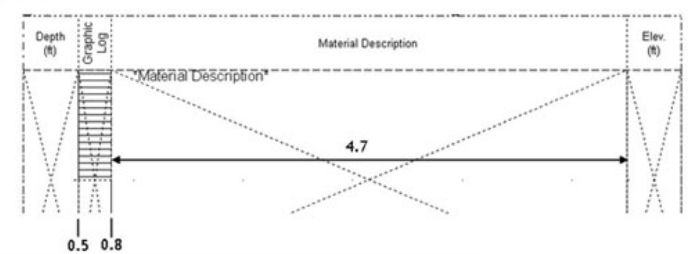
Advanced Log Report Design



Sample Data Present



No Sample Data



Calculates penetration (using another function) →

Constructs hyphenated text string of blow counts →

Conditional expression to display penetration after a slash if there is penetration, or N-value in parentheses on a new line (Blows 2nd + Blows 3rd) if no penetration. →

```

Data or Expression
<<Let(Refusal Pen = <<User System Data.Refusal Penetration>>>>_
<<ListBuildSepIn(
  "
  <<SAMPLE.Blow 1st>>_
  <<SAMPLE.Blow 2nd>>_
  <<SAMPLE.Blow 3rd>>_
  <<SAMPLE.Blow 4th>>_
  )>>_
<<If(Refusal Pen >> 0,
  <<DelimitData(
    <<cr>>{
      <<Calc(<<SAMPLE.Blow 2nd>> + <<SAMPLE.Blow 3rd>>>>}_
    }>>_
  )>>_
  )>>_
  )>>
  
```

If penetration

If no penetration

The information in this publication is subject to change without notice and does not represent a commitment on the part of gINT Software. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software must be used or copied only in accordance with the terms of the agreement.

Printed November 7, 2008

Disclaimer: Every effort was made to ensure accuracy. However, gINT Software makes no warranty as to the correctness of this information or the supplied files.

All rights reserved worldwide. No part of this publication may be reproduced in any form or by any means without the prior written consent of gINT Software. Comments are welcome and become the property of gINT Software. gINT® is a registered trademark of gINT Software. All other products mentioned are trademarks or registered trademarks of the respective producers.

© Copyright 1985-2007 by gINT Software, Inc.
Printed in the United States of America

Web Site: www.gintsoftware.com
E-mail: support@gintsoftware.com
Sales: sales@gintsoftware.com

Table of Contents

- Using this Tutorial..... 1
 - Setting up Sample Files 1
- Smart Report Forms 2
 - Copying the FINAL SIMPLE LOG Report for Exercises..... 2
 - Conditional Visibility of a Header Text Item 3
 - Conditional Visibility of N-Value and Refusal Penetration 5
 - Conditional Font Changes to Data..... 7
 - Conditional Visibility of Log Columns..... 8
- Variable Depth Scales on Logs 12
 - Default Depth Scale Behavior 12
 - Customized Depth Scale Behavior with a Calculated Expression..... 13
 - Specifying Depth Scale on a Per-Project or Per Borehole Basis 15
 - Using Depth Scales other than Depth-Per-Page 16
 - Using Depth Units per Page Unit (Feet per Inch)*..... 16
 - Using Numeric Ratios* 18
- Test Pit Log -- Using Photo Fields and Multiple-Page Setup 20
 - Creating the Borehole Photos Table 20
 - Creating the Test Pit Report 23
 - Creating and Configuring Private Blocks 30
- Variable Graphic Legends on Reports 33
 - Log Report Legend..... 33
 - Fence Report Legend..... 36
 - Graphic Text Doc Legends..... 39
 - Symbol Legend Positions..... 45
- Appendix A -- Sorting Boreholes using a GintPointSort Field 47

Using this Tutorial

This tutorial is designed for intermediate-to-advanced level gINT users, and is intended for self-study. It covers the options for creating various kinds of Smart Report features, such as report columns that disappear when they have no data, with other columns enlarging to take up the space. Another topic explored in this tutorial covers various ways to customize depth scales on a report-by-report, project-by-project or borehole-by-borehole basis. Finally, a complex log report with standard column entities on the first page followed by photos from the database is created, demonstrating Bitmap File type fields and Private Blocks.

You should be familiar with log report design before working through this tutorial. We recommend you work through the *Log Report Design* tutorial or take the gINT University class *gINT 004 - Log Report Design* if you do not have that familiarity.

Setting up Sample Files


Before starting the tutorial, you need the following:

- gINT Version 8 installed on your computer
- Sample files installed in the appropriate subfolders of the \gINT\ installation folder (usually C:\Program Files\gINT\)

To obtain and install the sample files, do the following:

1. Go to www.gintsoftware.com/support_doc.html.
2. Click on the link 'Data Files for Advanced Log Report Design (006) Tutorial only'.
3. Extract the following files to the indicated locations:

File	Destination	Comments
training.glb	\gINT\libraries\	
training.gpj	\gINT\projects\	
test pit data.xls	\gINT\projects\	
<i>all files in \photos folder</i>	\gINT\photos\	You will have to create this subfolder of \gINT\

 **Note:** If these files are already present in the indicated locations because of working on another tutorial, you do not need to replace them.

Smart Report Forms

A smart report form automatically adjusts its format based on the data, utilizing decision logic in properties in the report design and its entities. Some examples:

- Having a surface elevation label and data appear in the report header for boreholes that have elevation data, and not for those without it.
- On a log, if a sample has refusal, displaying the refusal penetration in inches after the final blow count, separated from it by a slash, as in '25-55/3'. If no refusal, displaying the blow counts and N-value.
- Highlighting data that is outside tolerances with bold or colored font.
- On a log, including or omitting specific columns depending on whether data for them is present, and enlarging the other columns to take up any vacant space.
- Setting the depth displayed per page based on a value in the borehole record, if present, and using defaults if not.

This section will demonstrate how the above smart report features are implemented, but this should be considered only a sampling of the possible smart report features. Most entities are configurable to appear or not (or to move) depending on the result of a calculated expression. Also, most entity properties can hold expressions in place of explicit values, and these expressions can use specialized functions—both system and user-defined—and programming logic to arrive at a result. This makes it possible for gINT reports to be heavily manipulated at output time by the data they are reporting.

To find out additional ways to configure smart report features, we recommend that you study the reports provided in the `training.glb` library and custom libraries that you can download from the [Downloads](#) page on the gINT website. We will also use some smart report features later in the course when we build the test pit log with photos, and implement legends.

Copying the FINAL SIMPLE LOG Report for Exercises

You may have created the SIMPLE LOG report during the *Log Report Design* module. In case you have not created this report, or there are errors or omissions, we'll replace it with a final version from the library. Do the following:

1. Go to **REPORT DESIGN ► Logs**.
2. Select 'FINAL SIMPLE LOG' in the object selector.
3. Select **File ► Copy Page**.
4. In **Page Name to Copy To**, enter 'SIMPLE LOG', and click **OK**. This creates a copy of FINAL SIMPLE LOG, and calls it SIMPLE LOG (replacing the existing SIMPLE LOG, if present).

Conditional Visibility of a Header Text Item

You can set up a specific text entity to appear only on the pages that meet the criteria you define. For example, if you are recording surface elevation data on your reports, but not all test sites have surface elevation, you do not want the text to print on the forms without the data, so you would configure the text entity as conditionally visible.

First we will add a Surface Elevation entity to the header, and then make it conditional. Do the following:

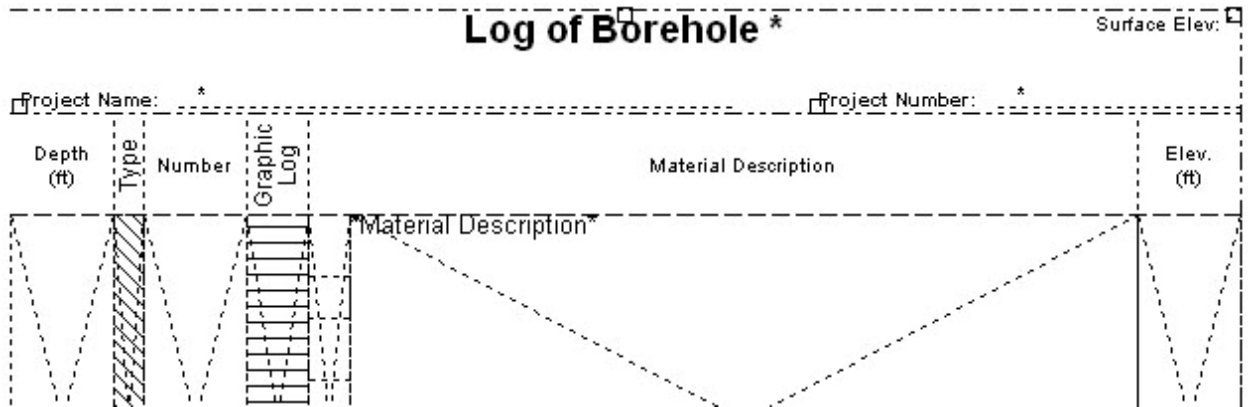
1. Go to **INPUT**.

! 2. Ensure that the **training.glb** library is active (**File ► Change Library**) and the current project is **training.gpj** (select **File ► Open Project** and browse to the project).


3. Select the **Borehole** tab. Notice that all points have an **Elevation** value except 'TP-1'.
4. Go to **REPORT DESIGN ► Logs** and open the 'SIMPLE LOG' report.
5. Create a text (**A**) entity with the following properties in the **Main** tab:

Field Name	Value
Text Expression	Surface Elev: <<POINT.Elevation>>
Height	0.08
Horz Align	Right
Vert Align	Top
X	5.97
Y	8.97


6. Click **OK** to place the entity on the form.



7. Go to **OUTPUT** and preview the report.

- Using the object selector, go to page 1 of the report (borehole 'B-1') then page 8 ('TP-1'). You'll need to zoom in on the upper right corner using the Quick Zoom  tool. Notice that the B-1 page displays 'Surface Elev: 126' and the TP-1 page displays 'Surface Elev:' followed by a blank.

- Close the preview and return to **REPORT DESIGN**.


- Double-click the entity to re-open the Text Properties dialog box and expand the **Text Expression** field by clicking on the Expand  button. Initially the following appears:

```
Surface Elev: <<POINT.Elevation>>
```

- Highlight the contents of the line, select 'Functions' in the **System** list in the Data Tool and 'HasData' in the **Items** list. Then right-click the **Paste** button above **Items**. Right-clicking the **Paste** button inserts the highlighted expression into the selected function:

```
<<HasData(Surface Elev: <<POINT.Elevation>>)>>
```

The HasData() function needs two arguments to achieve the desired behavior—the field or expression to test for the presence of data, and the text to print if the expression has data. We have the second argument, which is the expression to print. What we need to add is the field to test for the presence of data.

 **Note:** The **Output Condition** field in the **Configuration** tab represents an alternative to using HasData() in an expression. See **Help ► Index ► Output Condition (Property)**.

- Click to the immediate left of the word 'Surface', select 'POINT' from the **Table** list and 'Elevation' from the **Field** list, then click the corresponding **Paste** button. Enter a comma following the field expression you just pasted. Your complete expression now looks like this:

```
<<HasData(<<POINT.Elevation>>,Surface Elev: <<POINT.Elevation>>)>>
```

- To improve the formatting, add the continuation character (the underline symbol, '_') followed by a line break (press **Enter**) after the comma, and another one just before the final ')>>', and indent the second line:

```
<<HasData(<<POINT.Elevation>>,_  
    Surface Elev: <<POINT.Elevation>>_  
)>>
```

This expression uses the HasData() function to say, if there is data in the **Elevation** field in the **POINT** table, then print the label 'Surface Elev:' followed by the contents of that field.

- Click the **Configuration** tab. Set the **Design Mode Text Display** property to 'Surface Elev'.

When you create a text entity with just an expression, you need to set a **Design Mode Text Display** or you will not see the entity displayed on the form in **REPORT DESIGN**.

- Click **OK** to save the entity. Notice that '* Surface Elev' appears in the upper right corner.


- Go to **OUTPUT ► Logs** and preview the report. View the contents of the upper right corner on pages 1 and 8, using the Quick Zoom tool to zoom in. Notice that page 1 has the same text as before, but the surface elevation text is missing from page 8, which is the desired behavior.


- Close the preview.

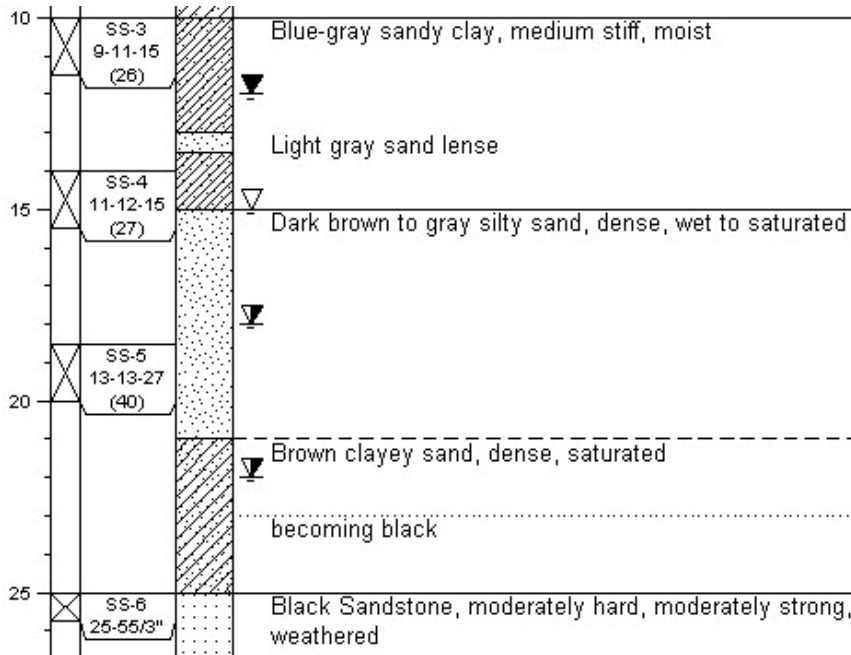
Conditional Visibility of N-Value and Refusal Penetration

Several standard gINT log reports indicate the blow counts for each sample. Refusal is calculated, and when there is no refusal, the N-value is presented. When there is refusal, the refusal penetration in inches appears after the final blow count, separated from it by a slash, as in '25-55/3"'.

The library has a user data item called 'Blows and N Value' that creates the sequence of hyphen-separated blow count values, and appends either the N-value or the calculated refusal. To see this in use, do the following:

1. Go to **REPORT DESIGN** ► **Logs**.
2. Double-click the column entity under the 'Number' heading (double-click in the column, near the diagonal broken lines, not the heading).
3. Click the Expand  button to the right of the **Text Expression** property. Notice that there is already an expression present for assembling the sample type and sample number into a hyphen-separated text string.
4. Click just past the end of the existing expression, then press the **Enter** key to start a new line.
5. In the **Table** list, select 'User System Data', then in the **Field** list select 'Blows and N Value'. Click **Paste**. The completed expression should look like this:


```
<<ListBuildSepTrim(-,<<SAMPLE.Type>>,<<SAMPLE.Number>>)>>
<<User System Data.Blow s and N Value>>
```
6. Click **OK** to save the expression.
7. Check the **Block Align** checkbox. This is used because the text to appear for each sample is now potentially three lines long, and needs to be centered vertically to look acceptable.
8. Click **OK** to save the changes to the column entity.
9. Go to **OUTPUT**, select borehole 'B-1' in the **Borehole ID** field, and preview the report. Use the Quick Zoom  tool to focus on the samples near the bottom of the report.



10. Notice that the samples above 25' show the hyphenated blow counts on one line, followed by the N-value in parentheses on a second line. However, the sample at 25' omits the N-value and adds the refusal penetration in inches. This is conditional behavior due to the fact that the sample at 25' has refusal.
11. To see the expression that generates two possible outcomes (N-value or penetration), close the preview, go to **DATA DESIGN ► User System Data**, and select 'Blows and N-Value' in the object selector.

The expression in the 'Blows and N-Value' user data item is explained as follows.

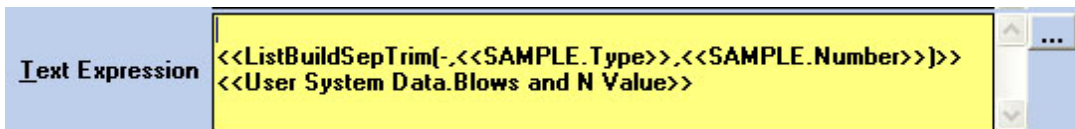
<p>Calculates penetration (using another function)</p> <p>Constructs hyphenated text string of blow counts</p> <p>Conditional expression to display penetration after a slash if there is penetration, or N-value in parentheses on a new line (Blows 2nd + Blows 3rd) if no penetration.</p>	<pre> Data or Expression <<Let(Refusal Pen = <<User System Data.Refusal Penetration>>)>>_ <<ListBuildSepTrim(<<SAMPLE.Blow 1st>>,_ <<SAMPLE.Blow 2nd>>,_ <<SAMPLE.Blow 3rd>>,_ <<SAMPLE.Blow 4th>>_)>>_ <<HasData(<<Get(Refusal Pen)>>),_ / <<Get(Refusal Pen)>>">>_ ← If penetration <<DelimitData(<<cr>>(_ <<Calc(<<SAMPLE.Blow 2nd>> + <<SAMPLE.Blow 3rd>>)>>,_)>>_)>>_ ← If no penetration </pre>
---	--

Conditional Font Changes to Data

You can set up a text entity to appear in a normal font on output when the data is within tolerances, but highlighted with bold or colored font when outside the acceptable range. This makes the data stand out on the report, calling attention to the situation.

We will make the output sample number bold, italic, and red if the sample type is 'RC', to call attention to the rock core samples.

1. Go to **REPORT DESIGN ► Logs**.
2. Double-click on the graphic column under the 'Number' heading. The **COLUMN - TEXT vs DEPTH PROPERTIES** dialog box appears.
3. Create a blank line at the beginning of the **Text Expression** by clicking at the left of the first line and pressing **Enter**.



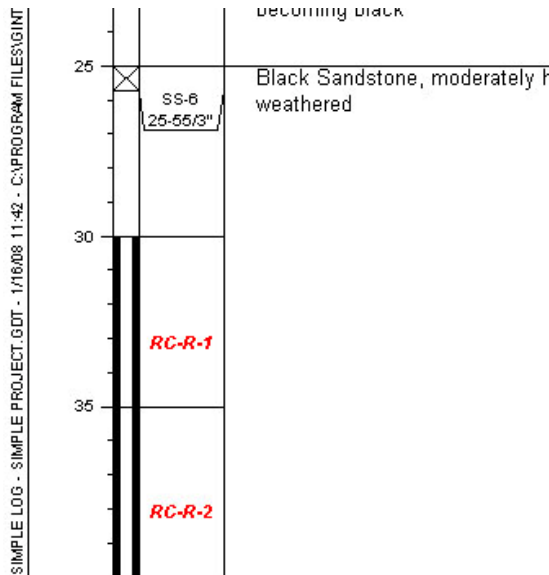
4. Enter the following text on the first line:

```
<<IIf( <<SAMPLE.Type>>="RC" , <<B>><<I>><<CLR!255>> )>>
```


To create this, do the following:

- a. In the Data Tool, select 'Functions' in the **System** list, then 'IIf()' in the **Items** list. Click the corresponding **Paste** button.
 - b. Select 'Sample' in the **Table** list, then 'Type' in the **Field** list, and click the corresponding **Paste** button.
 - c. Enter the text '= "RC", ' (an equals sign, then RC in quotes, then a comma and space).
 - d. Select 'Formatting Codes' in the **System** list, then 'B' in the **Items** list, and click **Paste**.
 - e. Select 'I' (the letter I) in the **Items** list, and click **Paste**.
 - f. Select 'CLR!' in the **Items** list, and click **Paste**.
 - g. Type the number '255'.
5. Click **OK** to save the revised text-vs.-depth entity.

Note: The color code 'CLR!255' corresponds to 'Very Light Red'. This can be seen in **SYMBOL DESIGN ► Colors** at the very bottom of the colors list.
 6. Go to **OUTPUT ► Logs**, select borehole 'B-1', and preview the report.
 7. Zoom in on the bottom left corner of the report. You will see that the numbers of the rock core samples are now in bold italic red, whereas non-rock-core samples are unaffected.



8. Close the preview.

 **Note:** Most gINT formatting codes are toggles that are turned off and on by the same code; but if you are using the CLR! code, you need one code to turn the color on and a different code to turn it off.

 For further study, go to the Help ► Index ► Formatting Codes topic.

Conditional Visibility of Log Columns

One of the most useful smart report features is the ability to make columns disappear on pages where they contain no data, and adjust the surrounding columns to take up the available space. A sophisticated example of a report using this feature is the 'GENERAL BH/TP/WELL' report in your library. It will suppress 'Recovery', 'Blow Count', 'Test/Remarks', 'USCS', 'Environmental', and 'Well' columns in any combination, relocating the other columns and widening the Material Description column to compensate. To view this report in action, do the following:

1. In **OUTPUT**, select all boreholes (clear the selection from the **Borehole ID** field) and choose the 'GENERAL BH/TP/WELL' report in the object selector.
2. Preview the report, and look at each page in sequence (selecting each page in the object selector). Notice that only 'B-1' and 'B-2' have well columns, 'CPT-1' lacks recovery and blow count columns, and 'TP-1' has only four columns of the eleven in the report design.

We will create a much simpler smart log: a modified version of 'SIMPLE LOG' in which we will implement conditional display of the two Sample columns, and enlargement and movement of the Lithology columns when Sample data is not present. However, the principles are the same, and afterwards you should be capable of analyzing how the design of 'GENERAL BH/TP/WELL' works.

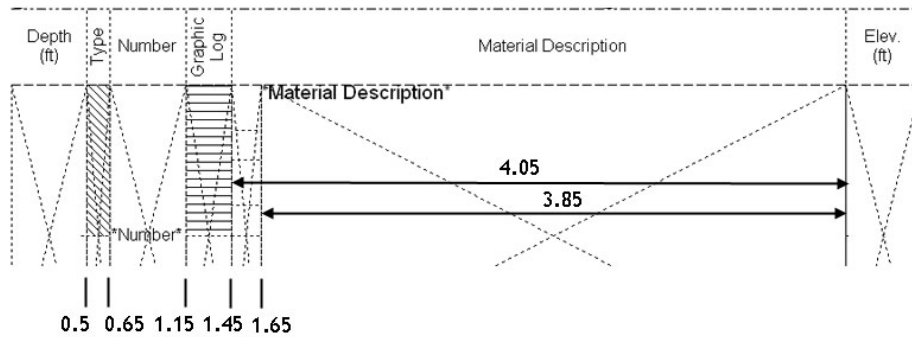
The design approach typically will include the use of the following elements:

- **Private blocks** to organize related entities into groups that appear/disappear and move together.

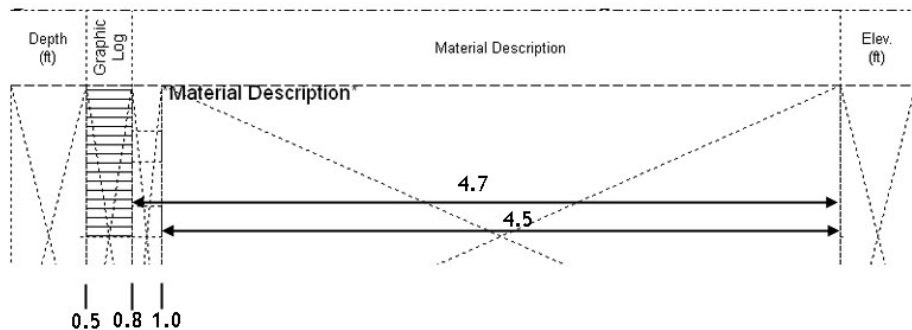
- The **Output Condition** in a private block to turn on or off the visibility of its component entities in response to an expression.
- The **Offset X** property in a private block that moves the block's entities to the left or right based on a calculation.
- **Report variables** for storing expressions that are used in calculating offsets and presence of data. User system data could be used for this purpose, but report variables are local to a specific report, rather than the entire library, and can be customized to the report.

We will set up the report to fluctuate between the following two formats:

Sample Data Present



No Sample Data



Note that we also need to move the water level symbols column 0.65" to the left (from 1.45 to 0.8) when there is no sample data.



We'll create two private blocks:

- A 'SAMPCOLS' block, containing the 'Sample Type' and 'Number' columns.
- A 'LITHCOLS' block, containing the 'Graphic Log' and 'Material Description' columns, and the three water level symbols.

To create a version of 'SIMPLE LOG' with conditional column visibility, do the following:

1. Go to **REPORT DESIGN** ► **Logs** and select 'SIMPLE LOG' in the object selector.
2. Select **File** ► **Copy Page**. The 'Copy Page' dialog box appears.
3. Enter the following values:


Field	Value	Comments
Page Name to Copy To	SIMPLE SMART LOG	Name of new report
Set Current Page To	First in Copy To List	Opens the duplicate report form rather than the original.

- Click the Properties  icon. The LOG REPORT PROPERTIES dialog box appears.
- Click the Expand  icon to the right of User Report Variables. Enter the following using the Data Tool:

Name	Value
HasSamples	<<CBool(<<Count(<<SAMPLE.Depth>>>>>>>>
OffsetSample	<<If(<<Report Var.HasSamples>>,0,0.65)>>

HasSamples is a True/False (boolean) variable that uses the Count() function to count the number of SAMPLE records for the borehole with data, and CBool() converts this to True if there is a positive number of records, or False if zero records.

The **OffsetSamples** variable returns the amount to shift the Material Description columns to the left. If **HasSamples** is False (there are no samples), the value 0.65 is returned as the distance to shift; if **HasSamples** is True (there are samples), zero is returned.


 **Note:** A report variable only prompts for a value on output when there is no **Value** property specified for it in Report Properties. Report variables with expressions do not appear to the user.

- Click OK to save the report properties.
- Select **Blocks ► Create Block**, enter the following values, then click OK.

Name	Value
Type	Private (current page)
Name	SAMPCOLS
Design Mode Color	Very Light Red

- In response to the **Select Entities** command prompt, click the 'Type' column, then the 'Number' column, then click OK.
- Select **Blocks ► Create Block**, enter the following values, then click OK.

Name	Value
Type	Private (current page)
Name	LITHCOLS
Design Mode Color	Very Light Blue

- In response to the **Select Entities** command prompt, click the 'Graphic Log' column, the water level symbols column, and the 'Material Description' column, then click OK.
- Click the Zoom Paper Extents  icon to see the entire report, including the two block inserts at left.

12. Double-click the red block insert for SAMPCOLS. The **BLOCK PROPERTIES** dialog box appears.
13. Click the **Configuration** tab, then enter the following:

Name	Value	Comments
Output Condition	<<Report Var.HasSamples>>	Select 'Report Var' from the Table list in the Data Tool, then 'HasSamples' from the Field list, then click Paste.

14. Click **OK** to save the changes to the block properties.
15. Double-click the blue block insert for LITHCOLS. The **BLOCK PROPERTIES** dialog box appears.
16. Enter the following in the Main tab:

Name	Value	Comments
Offset X	<<Calc(-1 * <<Report Var.OffsetSample>>)>>	Select 'Functions' from the System list in the Data Tool, then 'Calc()' from the Item list, then click Paste. Enter the '-1 *' expression manually, then paste <<Report Var.OffsetSample>> from the left half of the Data Tool.

17. Click **OK** to save the changes to the block properties.
18. Double-click the 'Material Description' column entity. The **COLUMN - TEXT vs DEPTH PROPERTIES** dialog box opens.
19. Enter the following:

Name	Value	Comments
Column Width Expression (in Main tab)	<<Calc(3.85 + <<Report Var.OffsetSample>>)>>	Adds either zero or 0.65" to the column width (3.85"), depending on the offset.
Column Display Width (in Configuration tab)	3.85	If not specified, the column entity shrinks horizontally in REPORT DESIGN (no effect on output)

20. Click **OK** to save the changes to the column entity.
21. Go to **OUTPUT**, and preview the report.
22. Page through the report pages using the object selector, and notice that the reports for all boreholes except 'CPT-1' have sample data columns, but 'CPT-1' has no sample columns and has widened and adjusted lithology columns.


Variable Depth Scales on Logs

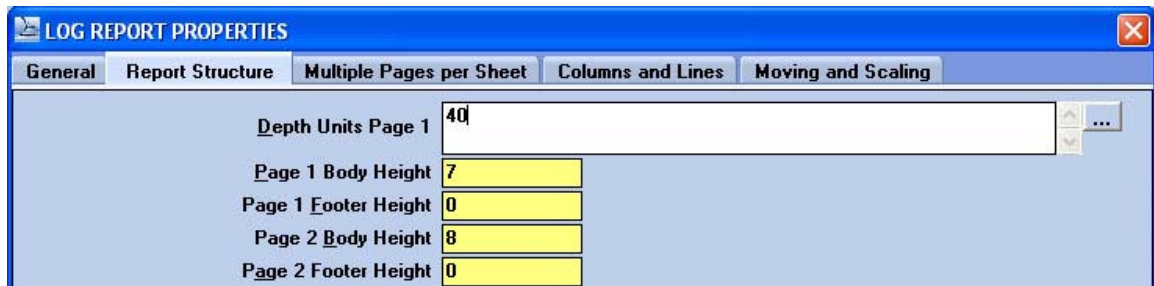
The *depth scale* of a log report is the way you specify how many *page units* (vertical inches or mm measured on the printed page) are used to represent some number of *depth units* (vertical feet or meters of borehole depth in the field) on output. There are certain default behaviors of the software that generate an appropriate depth scale in response to you entering a fixed depth scale value in **REPORT DESIGN** or **OUTPUT** (or leaving this blank). However, users often want more control of the depth scale, such as varying it between projects and/or boreholes, calculating it based on the total depth of the borehole, and so on. We will demonstrate the default behavior first, then explain various ways to customize it.

Default Depth Scale Behavior

In **LOG REPORT PROPERTIES**, there is a property called **Depth Units Page 1** that interacts with the **Depth Units Page 1** field on the **OUTPUT** tab. You can set a numeric value in the report property, and this specifies the depth scale for all boreholes, regardless of their depths. For example, if you set a **Depth Units Page 1** value of 30, the depth scale on the first page of the printed report will range from 0 to 30 feet, the second page (if needed) from 30 to 60 feet, and so on. The value is overrideable at output time, using the field in the **OUTPUT** tab, but always specifies a fixed depth scale.

Alternately you can set the **Depth Units Page 1** report property to blank, which calculates a best fit for each borehole that reduces or enlarges it to fit on a single page. To demonstrate this, do the following:


1. First we will demonstrate Fixed Depth Scale behavior. Go to **REPORT DESIGN** ► **Logs**, select 'SIMPLE LOG', and click on the Properties  icon. The **LOG REPORT PROPERTIES** window opens.
2. Click the **Report Structure** tab.
3. Notice that the **Depth Units Page 1** property has a value of '40'. Click **OK**.



4. Go to **OUTPUT** ► **Logs**. Notice that the **Depth Units Page 1** field appears at center left, and displays a default value obtained from the report property.



5. Preview the report. On page 1, notice that borehole 'B-1' (which has a hole depth of 45') extends all the way to the bottom of the page, and the depth scale extends to 40' (you may need to zoom to the lower left corner with the Quick Zoom tool to see this).

6. Select page 2 in the object selector. Notice that borehole 'B-1' continues for another 5 feet, and that the bottom of the depth scale is 80'.
7. Select page 5 in the object selector to view borehole 'B-3', which has a 25' hole depth. Notice that the log covers only a little more than the top half of the page, with blank space at the bottom. Notice also that the bottom of the depth scale (at lower left) is 40'.
8. Close the preview. Change the **Depth Units Page 1** value (under 'LOG OPTIONS') to '30', and preview the report again.
9. Notice on page 1 that the report is cut off at 30', on page 2 that an additional 15' is shown, and on page 5 that the 25' deep borehole 'B-3' covers most of the 30' deep page.
10. Close the preview.
11. Next we'll demonstrate automatic Best Fit depth scale behavior. Go to **REPORT DESIGN** ► **Logs**, and click the Properties  icon.
12. Click the **Report Structure** tab, and delete the value from the **Depth Units Page 1** property. Click **OK**.
13. Go to **OUTPUT** ► **Logs**. Preview the report.
14. Click through each of the five report pages listed in the object selector, and notice that each fills up one page, in spite of their varying hole depths ('B-1' and 'B-2' are 45', 'B-3' is 25', 'CPT-1' is 45' and 'TP-1' is 13').



When a best-fit page depth is calculated, this is computed as the hole depth plus 10%, rounded to the next whole depth unit. For example, the three 45' boreholes are placed on a 50' depth scale, the 25' borehole is on a 28' scale, and the 13' test pit is on a 15' scale.
15. Close the preview.

Customized Depth Scale Behavior with a Calculated Expression

The Fixed Depth Scale and Best Fit behaviors may be adequate for your needs. However, you may find these approaches too limiting, especially if your borehole depths range widely. Fixed Depth Scale will potentially spread your deepest boreholes over multiple pages while using a fraction of a page for a shallow hole like a test pit. Best Fit will compress deep holes into too short a log. Manually changing the fixed depth value on the **OUTPUT** tab and printing one borehole at a time is highly inefficient.

One remedy is to make use of a calculated expression in the **Depth Units Page 1** report property. The expression can utilize the hole depth of each borehole (which is available from a system variable), and generate a different depth-per-page for each range of hole depths. We'll demonstrate how to set different depths-per-page for three ranges of hole depths: less than 20', between 20' and 30', and greater than 30'.

Do the following:

1. Go to **REPORT DESIGN** ► **Logs**.
2. Click the Properties  icon, then click the **Report Structure** tab.
3. Click the Browse  icon to the right of the **Depth Units Page 1** property.

4. Enter the following:

```
<<Case(<<Max(<<LITHOLOGY.Bottom>>>>>> , _
  < 20,30,_
  < 30,40,_
  Else,50,_
)>>
```

To create this expression, do the following:

- a. Select 'Functions' from the **System** list, select 'Case()' from the **Items** list, then click the right-hand **Paste** button.
- b. Without moving the cursor (leaving it between the parentheses after 'Case'), select 'Max()' from the **Items** list, then click the right **Paste** button.
- c. Select 'LITHOLOGY' from the **Table** list, select 'Bottom' from the **Field** list, then click the left **Paste** button.

This inserts '<<Max(<<LITHOLOGY.Bottom>>>>>>' at the start of the Case() statement.

<<Max(<<LITHOLOGY.Bottom>>>>>> determines the maximum value in the **Bottom** field in the LITHOLOGY table for each borehole, which is the same as the hole depth.

- d. Move the cursor three characters to the right, so that it is between the ')>>' that closes the Max() function and the one that closes the Case() function. Enter a comma followed by an underscore, and press **Enter** twice. The result should look like this:

```
<<Case(<<Max(<<LITHOLOGY.Bottom>>>>>> , _
) >>
```

- e. In between the two lines of the expression, you add three lines, each of which specifies a condition for the hole depth (less than 20', less than 30') and a resulting depth-per-page. These are the following:

```
  < 20,30,_
  < 30,40,_
  Else,50,_
```



Be sure to indent each of these lines by two characters, and terminate it with a comma and an underscore. The comma separates clauses, the underscore (continuation character) unites the multiple lines into a single expression.

5. Click **OK** to save the expression. Click **OK** to save and close the **LOG REPORT PROPERTIES** window.
6. Go to **OUTPUT ► Logs** and preview the report. Notice that 'B-1' (page 1) has a 50 foot depth-per-page, which is consistent with the fact that its hole depth (45') is greater than 30 feet.
7. Select page 3, and notice that 'B-3', at 25' hole depth, has a 40' depth-per-page.
8. Select page 5, and notice that 'TP-1', at 13' hole depth, has 30' depth-per-page.
9. Close the preview.

Specifying Depth Scale on a Per-Project or Per Borehole Basis

By creating appropriate database fields in conjunction with the **Depth Units Page 1** property, you can set things up for a data entry person to specify a default depth scale value for each project, then override this value where desired for particular boreholes. Once the database fields and report property are set up, considerable depth scale control would be provided to persons with no report design knowledge.

gINT typically creates a field called **Depth Log Page** in both the **PROJECT** and **POINT** tables for this purpose. This field already appears in both tables in **training.gpj**, although it would be simple to create if it did not already exist. The **Depth Units Page 1** property in **REPORT DESIGN** is then configured with an expression to use the **POINT** table **Depth Log Page** value first, if present, otherwise use the **PROJECT** table value, and if neither is present, use a fixed-value default. Do the following:

1. Go to **DATA DESIGN ► Project Database**, open the current database, and select the **PROJECT** table from the object selector drop-down list.
2. Click **Depth Log Page** in the **Fields** list, and notice that it has a **Type** of 'Single' (floating point decimal), **Units** of 'ft', and an explanation in the **Description** property.
3. Select the **POINT** table from the object selector drop-down list, and notice that a **Depth Log Page** field is provided here as well. Click on the field in the **Fields** list and notice the property settings.
4. Go to **REPORT DESIGN ► Logs**, click the Report Properties  icon, then click the **Report Structure** tab.
5. Click the Expand  button to the right of the **Depth Units Page 1** field. An expression builder window appears.
6. Enter the following expression in the window, using the **Table** and **Field** drop-down lists in the **Data Tool** to specify the fields, and the **System** and **Items** drop-down lists to specify the **FirstData()** function.

```
<<FirstData(_
  <<POINT.Depth Log Page>>,_
  <<PROJECT.Depth Log Page>>,_
  40_
)>>
```

The **FirstData()** function returns the data from the first non-empty argument. The arguments are deliberately ordered from the most specific (use the per-borehole value) to the most general (the fixed value default of 40), stopping as soon as data is encountered. Notice that instead of 40 you could specify a different default depth-per-page.

7. Click **OK** to save the expression and dismiss the expression builder window. Click **OK** to save and close the report properties.
8. Go to **INPUT ► Project**. Enter a value of '30' for **Depth Log Page**.
9. Go to **OUTPUT ► Logs**. Preview the report. Notice that all the report pages have a depth-per-page of 30'. Close the preview.
10. Go to **INPUT ► Borehole**.

11. Scroll all the way to the right to display the **Depth Log Page** column. In this column, enter '40' for 'B-1' and '20' for 'TP-1'.
12. Go to **OUTPUT ► Logs**. Preview the report. Notice that 'B-1' (pages 1 and 2) has a depth of 40' (which was specified in POINT), 'B-2' (pages 3 and 4) has a depth-per-page of 30' (from the PROJECT table default), and 'TP-1' (page 8) has a depth-per-page of 20' (from POINT).

Using Depth Scales other than Depth-Per-Page

The specification of depth scale values on a per-project basis with a per-borehole override does not have to be restricted to a depth-per-page in feet. Two other possibilities are the following:

- Entering a depth-units-per-page-unit default. This is for users who want a physically measurable scale on their reports, such as X feet per inch, but want to vary the scale by borehole.
- Entering a numeric ratio such as 100:1, also with a per-borehole override. This option is often preferred in regions using metric units.

The implementation of these scenarios is described in the following subsections.

Using Depth Units per Page Unit (Feet per Inch)

Perhaps you want to be able to set a scale that is physically measurable on the page, such as 10 feet per inch, but want the ability to set a different scale from the default for especially shallow or deep holes.

Setting this up is similar to using **Depth Log Page** fields, but uses a different field name and different calculations. Again a field will be created the PROJECT and POINT tables, this time named **Log Scale**. You will set the **Depth Units Page 1** value to a calculation that derives the log depth from one of the **Log Scale** values and the height of the report page (specified in the **Page 1 Body Height** field in Report Properties, and accessed via the system variable **LogBodyHtPg1**).



To set up these fields and this calculation, perform the following steps:

1. Go to **DATA DESIGN ► Project Database** and select the PROJECT table from the object selector drop-down list. Click the **New** button beneath the field list and enter the following (the **Description** is optional):

Field Name	Type	Units	Description
Log Scale	Double	ft	Default is 10. Can be overridden for specific boreholes using the Log Scale field in the POINT table.

2. Select the POINT table from the object selector drop-down list. Click the New button beneath the Fields list and enter the following:

Field Name	Type	Units	Description
Log Scale	Double	ft	Overrides the Log Scale field in the PROJECT table. If blank in both the PROJECT and POINT tables, defaults to 10.

3. Replace the data template (File ► New Data Template ► Clone From Current, and specify data entry.gdt).
4. Open REPORT DESIGN and click the Report Properties  icon. Click the Report Structure tab.
5. Click the Expand  button to the right of the Depth Units Page 1 field.
6. Enter the following expression in the window, using the Table and Field drop-down lists in the Data Tool to specify the fields, the System (with a value of 'Functions') and Items drop-down lists to specify the Calc() and FirstData() functions, and the System (with a value of 'Data Items') and Items drop-down lists to specify LogBodyHtPg1.

```
<<Calc(_
  <<LogBodyHtPg1>> * _
  <<FirstData(_
    <<POINT.Log Scale>>,_
    <<PROJECT.Log Scale>>,_
    10_
  )>>_
)>>
```

What this expression does is the following: it 1) checks for a Log Scale value in the POINT table, and uses it if found, or if no POINT table value is present, uses the Log Scale value in the PROJECT table, or 10 if neither is present, 2) multiplies the Log Scale value by LogBodyHtPg1, which is the height of the log report body in page units. This converts page units to depth units using your specified Log Scale scale—for example, 10 ft/in x 8 inches = 80 feet of depth represented on the page.

7. Click OK to save the expression. Click OK to save and exit the report properties.
8. Go to INPUT ► Borehole.
9. Scroll to the far right to show the Log Scale column. For borehole 'B-2', specify a Log Scale value of '5'.
10. Go to OUTPUT ► Logs and preview the report.
11. Notice that borehole 'B-1' (page 1) is very compressed, and has a depth-per-page of 80'. This is due to relying on the default Log Scale (from the expression) of 10 ft/in.
12. Select 'B-2' (page 2) in the object selector. Notice that the depth-per-page is 40', due to the specification of a Log Scale of '5'.
13. Close the preview, and go to REPORT DESIGN ► Logs.
14. Open the properties window, click the Report Structure tab, replace the Depth Units Page 1 value with '40', and click OK. This restores typical depth scale behavior for future exercises using the SIMPLE LOG report.

Using Numeric Ratios

Particularly in countries where metric units are used, you may wish to use a scale ratio such as 100:1 or 50:1, such that the measurements on the page are a specific fraction of the measurements in the field. Setting this up is very similar to the previous two approaches. A **Scale Ratio** field is created in each of the PROJECT and POINT tables. You will set the **Depth Units Page 1** value to a calculation that derives the log depth from one of the **Scale Ratio** values and the system’s log body height value.

1. Go to **DATA DESIGN ► Project Database** and select the PROJECT table from the object selector drop-down list. Click **New** beneath the field list and enter the following:

Field Name	Type	Units	Description
Scale Ratio	Single	(blank)	Numerator of the log scale ratio. For example for 100:1 scale, Scale Ratio would be 100. Default is 100. Can be overridden for specific boreholes using the Scale Ratio field in the BOREHOLE (POINT) table.

2. Select the POINT table from the object selector drop-down list. Click **New** beneath the field list and enter the following:

Field Name	Type	Units	Description
Scale Ratio	Single	(blank)	Overrides the Scale Ratio field in the Project table. If blank in both the Project and Point tables, defaults to 100.

3. Open **REPORT DESIGN ► Logs** and click the Report Properties icon. Click the **Report Structure** tab.
4. Click the Expand **...** button to the right of the **Depth Units Page 1** field. An expression builder window appears.
5. Enter the following expression in the window, using the **Table** and **Field** drop-down lists in the Data Tool to specify the fields, and the **System** and **Items** drop-down lists to specify the FirstData() function.

```
<<Calc(_
  <<LogBodyHtPg1>> * _
  <<FirstData(_
    <<POINT.Scale Ratio>>,_
    <<PROJECT.Scale Ratio>>,_
    100
  )>> / 1000_
)>>
```

What this expression does is the following: it 1) checks for a **Scale Ratio** value in the POINT table, and uses it if found, or if no POINT table value is present, uses the **Scale Ratio** value in the PROJECT table, 2) multiplies the **Scale Ratio** value by **LogBodyHtPg1**, which is the height of the log report body in page units (millimeters), then 3) divides the result by 1000 to convert from millimeters to meters. This converts page units to depth units using your specified scale ratio—for example, a scale ratio of 100:1 = 18 meters of depth represented on the page, since the log body height is 180 mm.

☞ **Note:** This example is intended to show how a user with metric page units and metric depth units would specify a typical scale ratio. In the provided sample files, however, you have English/Imperial page units, so the above example won't actually work on your system. If you want to convert depth units of feet to page units of inches with a scale ratio, you'll use 12 instead of 1000 in the above expression.

6. Click **OK** to save the expression. Click **OK** to close the report properties.

We will not preview the report.

Test Pit Log -- Using Photo Fields and Multiple-Page Setup

In this exercise we take the SIMPLE REPORT log form and convert it to a test pit log that displays photos on the pages following each log. We will set this up so that if there are no photos, no additional pages will print, but if photos are referenced by the borehole, they will be printed two to a page. We will take the following approach:


- Create a new table to reference multiple photos for each borehole, with a **Bitmap File** field in the photos table
- Add two bitmap symbol entities to the side of the report to display the up to two photos per page
- Move the column entities to a block, and set a condition to suppress the printing of this block after page 1
- Set an output override X value for the photos so they move over to the left to replace the column entities on page 2 and beyond.

Creating the Borehole Photos Table

The first step is to set up the database to hold references to multiple photos. If you knew you only wanted a fixed number of photos per test pit log, you could place a specific number of **Bitmap File** fields directly in the borehole record. However, to have the flexibility of including as many photos and diagrams as exist for the test pit, the best approach is to create a child table. We will call the table PHOTOS, and each record will hold the link to one photo, along with its description.

To create the PHOTOS table, do the following:

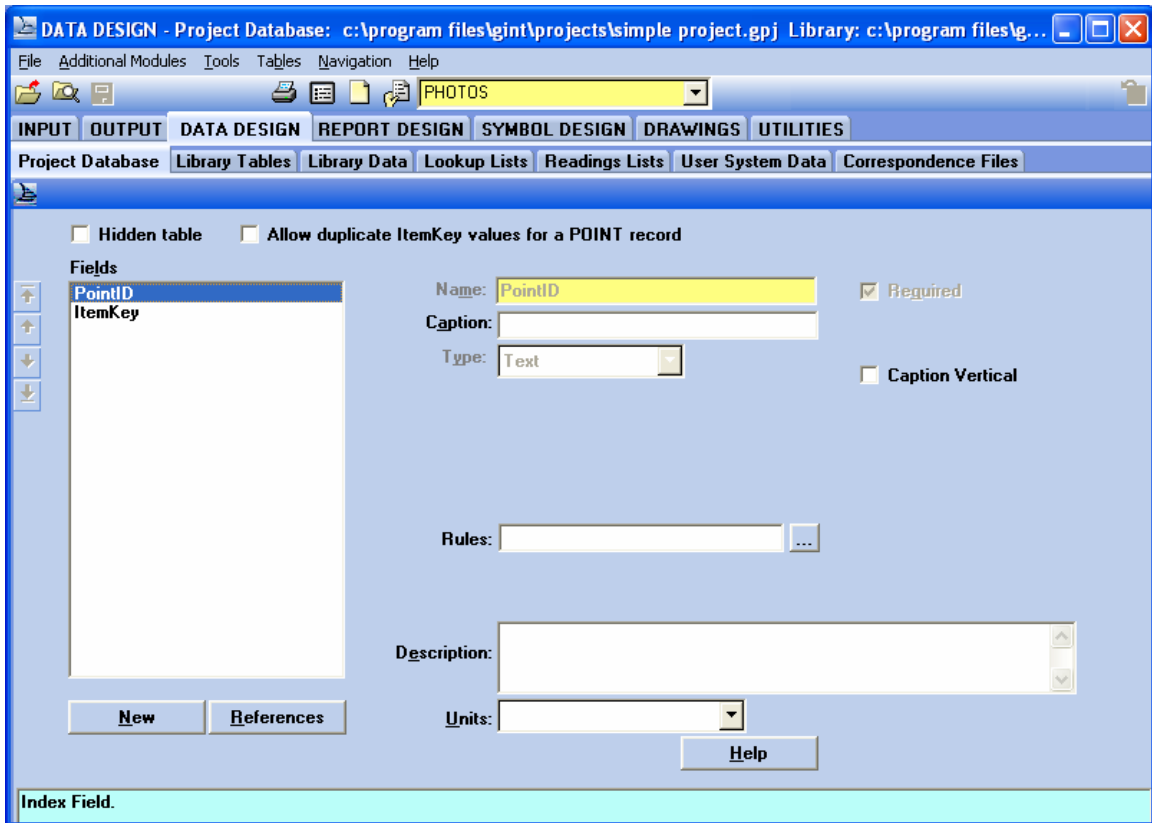
1. Go to **DATA DESIGN ► Project Database**. Open the current project (**training.gpj**).

2. Select **Tables ► New** or the New  button in the toolbar. The New... dialog box appears:

3. For the Name, enter 'PHOTOS'.
4. For the Key Set, select 'PointID, ItemKey'.

To establish a one-to-many relationship between the POINT table and the PHOTOS table requires that the PHOTOS table be keyed on a combination of the POINT table's key (PointID, or 'Borehole Number') and a key of its own. The name for a child table's key is always ItemKey (or ItemKey2), unless Depth, Reading, or DateTime is used as the child key. We will caption ItemKey shortly with a more memorable name.

- Click OK. The field list and properties appear.



- With PointID highlighted, specify a **Caption** of 'Borehole Number', and click OK.
- Click on ItemKey in the field list. Specify a **Caption** of 'Photo ID', and click OK.
- Click the **New** button beneath the field list. Specify a **Name** of 'Photo File' and a **Type** of 'Bitmap File', and check the **Required** checkbox. Click OK.
- Select **File ► New Data Template ► Clone From Current** and specify 'data entry.gdt' for the filename.

☞ Remember, always save to the corresponding template when you make changes to tables or fields in a project!

The **Photo File** field will hold two pieces of data for each photo record: the name and location of the bitmap (JPG, BMP, TIF, etc.) file, and descriptive text. We'll see how this works in **INPUT**.

- Go to **INPUT**. Notice that there is now an additional tab, **Photos**.
- We need to import data for a new test pit. Select **File ► Import/Export ► Import From Excel File**. In the **IMPORT FROM EXCEL FILE** dialog box, specify an Excel File of 'test pit data.xls' from the \gINT\projects\ folder and click OK.
- Close the import status window. The data entry grid should now have data for test pit 'TP-114' in addition to the pre-existing boreholes.
- Click the **Photos** tab. Drag the right edge of the header of the **Photo File** column so that the column extends to the right edge of the application window.


- Select 'TP-114' in the object selector. Enter the following in the empty bottom row (you will need to click the Expand  button in the Photo File column to specify the photo's filename and description, then click OK):

Photo ID	Filename (Load Raster File)	Description
P05	\Photos\TP 114 bottom.jpg	Bottom of Test Pit 114.

Notice how the combined file path and description are stored to a single grid cell, separated by the vertical line character.

Photo ID	Photo File
P01	C:\Program Files\gINT\photos\TP 114 spoil.jpg Spoil from Test Pit 114, excavated with a JCB on 12/10/06.
P02	C:\Program Files\gINT\photos\TP 114.jpg Photograph of Test Pit 114 taken on 12/10/06. Pit is 12.4' deep.
P03	C:\Program Files\gINT\photos\TP 114 spoil alternate.jpg Alternative photo of spoil from pit 114, excavated on
P04	C:\Program Files\gINT\photos\TP 114 alternate.jpg Alternative photograph of Test Pit 114 taken on 12/10/06. Pit
P05	C:\Program Files\gINT\photos\TP 114 bottom.jpg Bottom of Test Pit 114.
*	

A Bitmap File field does not contain the bitmap graphic, only a link to where it resides on your computer or network. If you move or rename the file, you will need to update the link in INPUT.

Creating the Test Pit Report


The test pit report will be a customized version of the SIMPLE LOG report that was previously created. To create a copy, do the following:

- Ensure that you are in REPORT DESIGN and 'SIMPLE LOG' appears in the object selector.
- Select File ► Copy Page. The 'Copy Page' dialog box appears.
- Enter the following values:

Field	Value	Comments
Page Name to Copy To	SIMPLE PHOTO TEST PIT	Name of new report
Set Current Page To	First in Copy To List	Opens the duplicate report form rather than the original.

- Click OK. The SIMPLE PHOTO TEST PIT report has been created and appears in REPORT DESIGN.

We are going to force a specific number of "additional pages" to print for each borehole past the number of pages required for the log. We want up to two photos to print per additional page. So if there are 2 photo records for the borehole, we want 1 additional page. If there are 5 photos we want 3 additional pages. To accomplish this, we create an expression in the **Number of Additional Pages Expression** property.

- Click the Properties  icon. The LOG REPORT PROPERTIES dialog box appears.
- Select the Report Structure tab.
- In the **Number of Additional Pages Expression** property at the bottom, enter the following:


```
<<Calc(( <<Count( <<PHOTOS.ItemKey>> ) >> + 1) \ 2) >>
```

The '<<Count(<<PHOTOS.ItemKey>>>)>>' sub-expression counts the number of PHOTOS records for the borehole being output. The backslash symbol performs "integer division", which rounds off the remainder from the result, so the calculation adds 1 to the number of photos, divides by 2, then discards the remainder.


4. Click OK.

Next, we will add two bitmap symbol entities to the report, each one for display of a photo, along with their descriptive text blocks and a divider line. In the report design the bitmap symbols and their associated text and line will be placed to the right of the right margin rather than at the X coordinates where they will print at output time. This is for ease of report form maintenance (it would be too messy if we overlaid the photos on the column entities). We will create and configure blocks shortly that will cause the column entities to print only on page 1, and on subsequent pages for these to be suppressed and the photos, descriptions, title and line to print at the correct distance to the left.

We'll include a subtitle, 'PHOTOS', to appear on the photo pages beneath the main heading.

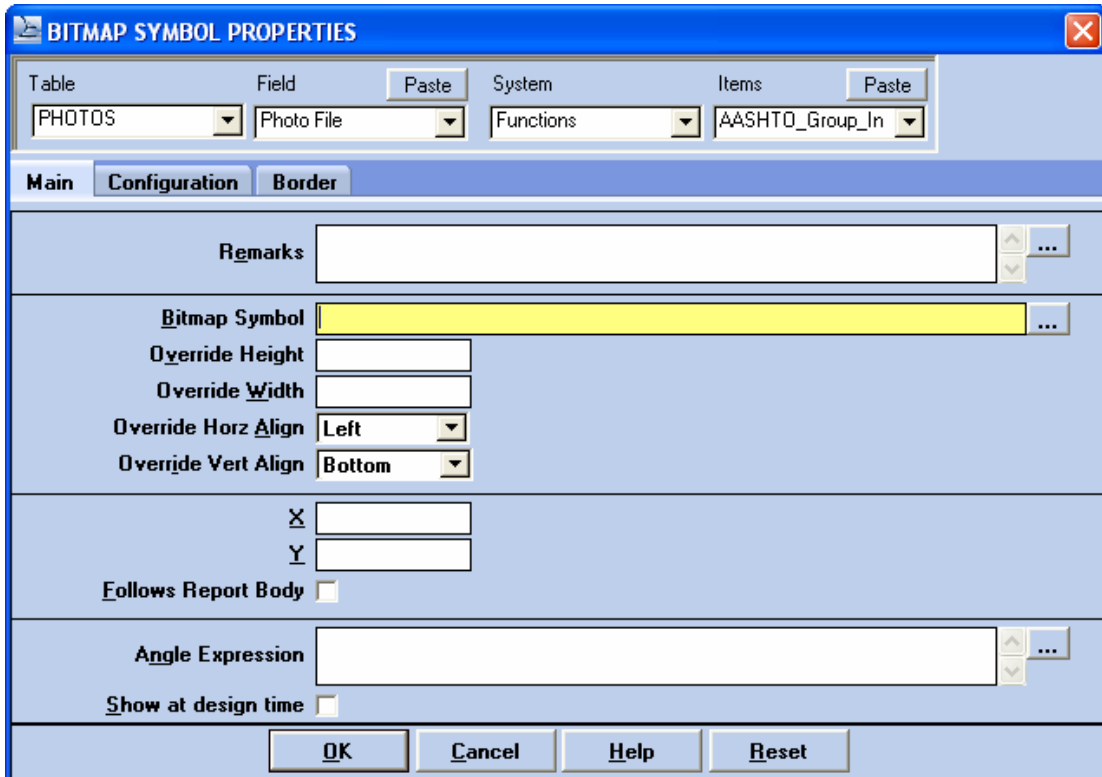
1. Select Draw ► Text ► Text or click the Text  icon. The TEXT PROPERTIES window appears.
2. Click Reset, then enter the following values:

Field	Value	Comments
Text Expression	PHOTOS	Text to print.
Height	0.14	Text size.
Horz Align	Center	Center-aligns at X coordinate vertically
Vert Align	Middle	Center-aligns at Y coordinate vertically.
X	11	
Y	8.25	

3. Click OK.
4. Click the Zoom To Extents  button. Notice that the PHOTOS text entity is at the far right.

To place the upper bitmap symbol, do the following:

1. Select Draw ► Graphics ► Bitmap Symbol. The BITMAP SYMBOL PROPERTIES window appears.



2. Enter the following values, then click OK:

Field	Value	Comments
Bitmap Symbol	<<User System Data.Photo File Page Top>>	Do not click the Browse icon – click in the field, then use the Data Tool to select 'User System Data' from the Table list and 'Photo File Page Top' from the Field list.
Override Height	3.3	Sets the height of the bitmap. Otherwise may be too small or too large for the space
Override Width	6.0	Sets the width of the bitmap. Note that whichever of Override Height or Width results in the smaller figure is the one used. Both are not used, because the figure would distort.
Override Horz Align	Center	Photo is center-aligned to X coordinate
Override Vert Align	Top	Photo is top-aligned to Y coordinate
X	11	Actual X coordinate plus 8" offset
Y	7.9	Height of photo alignment point

Notice that a small double-triangle marker symbol appears beneath the PHOTOS text heading.

The expression in the **Bitmap Symbol** property points to a user system data item called **Photo File Page Top**. You would normally create such a user data item yourself, or merge it from another library then customize it, but we have provided these in the library to save time.

The **Photo File Page Top** item has the following definition:

```
<<Let(PhotoNum = <<User System Data.Photo Number Top>>)>>_
<<User System Data.Photo File>>
```

which in turn references the **Photo Number Top** item:

```
<<Calc(((<<PgAdd1>> - 1) * 2) + 1)>>
```

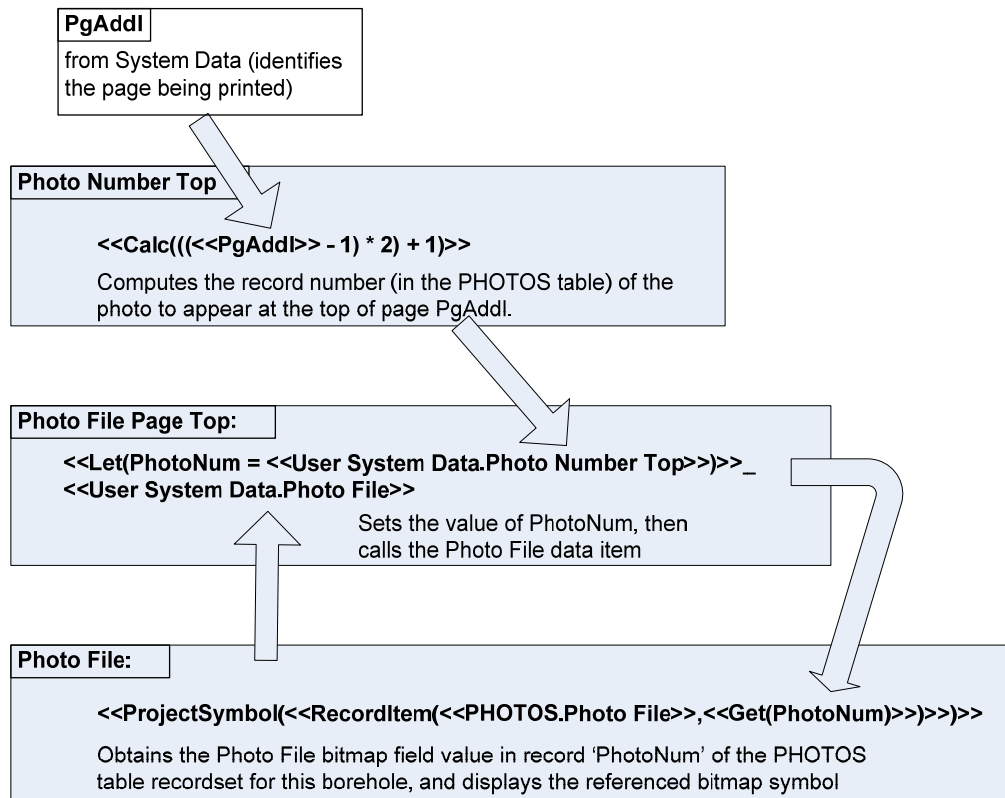
and the **Photo File** item:

```
<<ProjectSymbol(<<RecordItem(<<PHOTOS.Photo File>>,<<Get(PhotoNum)>>)>>)>>
```

So, what these calculations collectively do is the following:

- o Determine from the **PgAdd1** system data item which page is being printed, and compute from this the record number of the photo that should display at the top of this page. This calculation is performed in the **Photo Number Top** data item and is assigned to the temporary variable **PhotoNum** (in the Let statement in the **Photo File Page Top** data item).
- o Get the value of **PhotoNum**, and obtain the **Photo File** bitmap field value in record 'PhotoNum' of the PHOTOS table for the current borehole. Extract the bitmap symbol reference from it using the **ProjectSymbol()** function.

This is illustrated below:



- Click OK to close the dialog box and place the entity.

To create the first photo's description entity, do the following:

- Select **Draw ► Text ► Text**. The TEXT PROPERTIES dialog box appears. Enter the following values in the **Main** tab:

Field	Value	Comments
Text Expression	<<User System Data.Photo Description Page Top>>	See discussion below
Height	0.08	
Horz Align	Center	
Vert Align	Top	
X	11	
Y	4.5	

The **Photo Description Page Top** user data item extracts the description portion of the Photo File bitmap field from the appropriate PHOTOS record to display at the top of the current page. The logic is similar to **Photo File Page Top**, and uses the following user data items:

- Photo Description Page Top:**

```
<<Let(PhotoNum = <<User System Data.Photo Number Top>>)>>_
<<User System Data.Photo Description>>
```

- Photo Number Top:**

```
<<Calc(((<<PgAdd1>> - 1) * 2) + 1)>>
```

- Photo Description:**

```
<<DelimitData(_
  Figure <<POINT.PointID>>.<<Get(PhotoNum)>><<cr>>,_
  <<ProjectSymbolDesc(<<RecordItem(<<PHOTOS.Photo File>>,<<Get(PhotoNum)>>)>>)>>_
)>>
```

The **Photo Description** data item generates, prior to the description, a figure number from the borehole number and the record number. For example the third photo in test pit 12 would be labeled 'Figure 12.3'.

- Enter the following value in the **Configuration** tab:


Field	Value	Comments
Design Mode Text Display	photo description	Labels the text entity in REPORT DESIGN.

- Enter the following value in the **Wrap** tab:

Field	Value	Comments
Column Width Expression	6	Maximum width of the text.

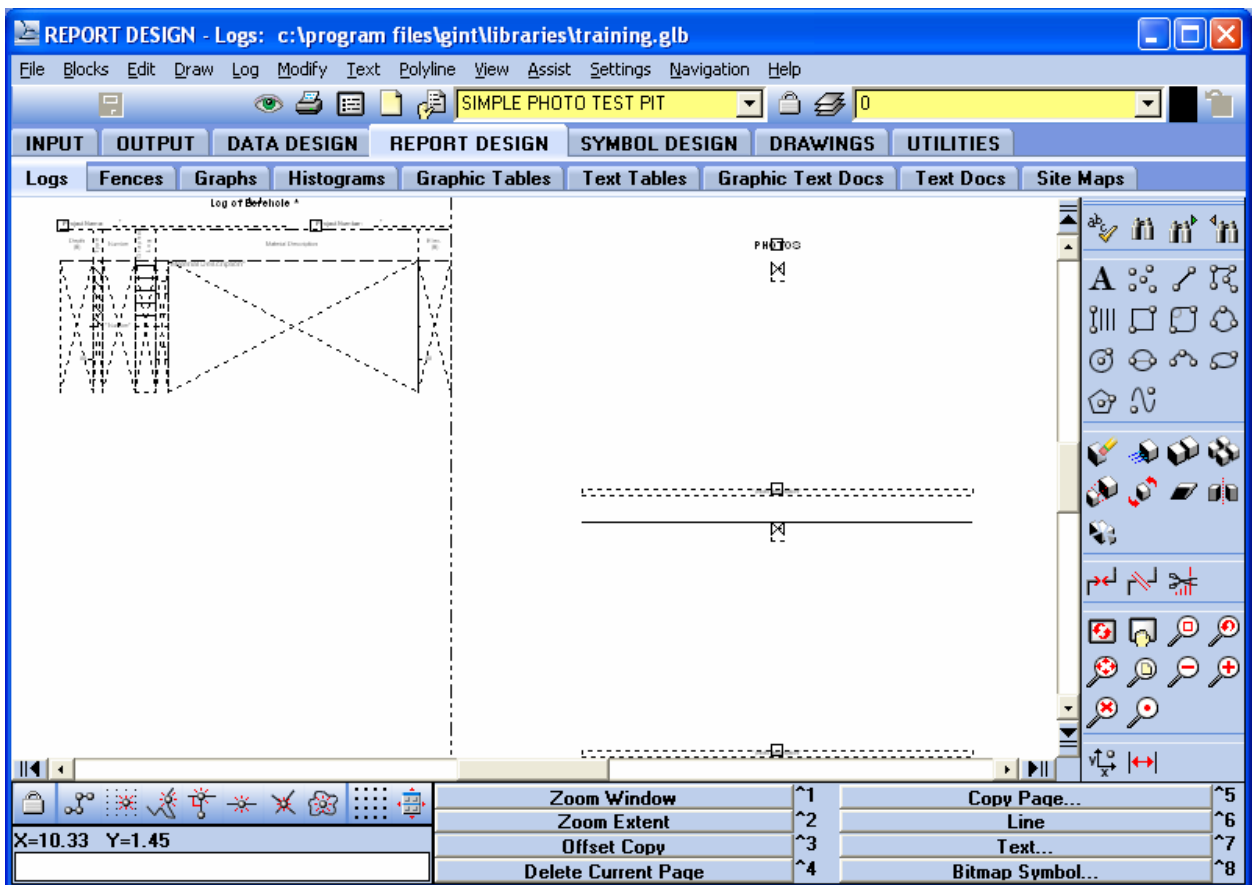
- Click OK to close the dialog box.

Next we create a dividing line between the upper and lower photos.

1. Click the Line  tool or select **Draw ► Points and Lines ► Lines**.
2. In response to the 'First Point' prompt, enter '8, 4' in the coordinates box at lower left, then press **Enter**.
3. In response to the 'Second Point' prompt, enter '14, 4' and press **Enter**. Then press **Esc** to get out of line entry mode. The dividing line will appear.

Now we duplicate the upper bitmap symbol and text entity using **Offset Copy** and modify the properties in the duplicates.

1. Select **Modify ► Offset Copy**. The 'Select Entities' prompt appears in the Command Box.
2. Click the graphic symbol entity to select it.
3. In the yellow **Chg X,Y** box at lower right, enter '0, -4', then click **OK**. A copy of the graphic symbol entity appears beneath the dividing line. The 'Select Entities' prompt reappears.
4. Click the photo description text entity to select it. Click **OK**. You do not need to change the **Chg X,Y** value, which is still correct. A copy of the text entity appears near the bottom. The report design appears as shown:



- Double-click on the lower bitmap symbol entity. The BITMAP SYMBOL PROPERTIES window appears. Change the following properties in the **Main** tab:

Field	Value	Comments
Bitmap Symbol	<<User System Data.Photo File Page Bottom>>	Extracts the file reference portion of the bitmap field in the appropriate PHOTOS record. Use the left half of the Data Tool.

The **Photo File Page Bottom** user data item extracts the bitmap portion of the Photo File bitmap field from the appropriate PHOTOS record to display at the bottom of the current page. The logic is similar to **Photo File Page Top**, and uses the following user data items:

- Photo File Page Bottom:**

```
<<Let(PhotoNum = <<User System Data.Photo Number Bottom>>)>>_
<<ProjectSymbol(<<RecordItem(<<PHOTOS.Photo File>>,<<Get(PhotoNum)>>)>>)>>
```

- Photo Number Bottom:**

```
<<Calc(<<PgAdd1>> * 2)>>
```

- Click OK to close the dialog box.
- Double-click on the lower description text entity. The TEXT PROPERTIES window appears.
- Change the following property setting in the **Main** tab:

Field	Value	Comments
Text Expression	<<User System Data.Photo Description Page Bottom>>	Extracts the description portion of the bitmap field in the appropriate PHOTOS record.

The **Photo Description Page Bottom** user data item extracts the description portion of the Photo File bitmap field from the appropriate PHOTOS record to display at the bottom of the current page. The logic is similar to **Photo Description Page Top**, and uses the following user data items:

- Photo Description Page Bottom:**

```
<<Let(PhotoNum = <<User System Data.Photo Number Bottom>>)>>_
<<User System Data.Photo Description>>
```

- Photo Number Bottom:**

```
<<Calc(<<PgAdd1>> * 2)>>
```

- Photo Description:**

```
<<DelimitData(_
  Figure <<POINT.PointID>>.<<Get(PhotoNum)>><<cr>>,_
  <<ProjectSymbolDesc(<<RecordItem(<<PHOTOS.Photo File>>,<<Get(PhotoNum)>>)>>)>>_
)>>
```

- Click OK to close the dialog box.

- Click the Save  icon.

Creating and Configuring Private Blocks

We will place the “page 2 and beyond” entities (everything associated with the photos) in one private block, and the “page 1 only” entities (all of the column entities and the water-level indicators) in another. Private blocks are very useful for displaying or suppressing a group of entities together, as well as for moving a group of entities a specified distance in response to a specified condition.

To create the two blocks, do the following:

1. Select **Blocks ► Create Block**. The **CREATE BLOCK** dialog box appears.
2. Enter the following values:

Field	Value	Comments
Type	Private (current page)	A private block.
Name	PHOTO PAGE	Block’s name.
Page Range	Additional Pages	Restricts the output of this block to page 2 and beyond of the report.
Design Mode Color	Very Light Magenta	The identification color for entities in the block.

3. Click **OK**. The ‘Select Entities’ prompt appears in the Command Box.
4. Draw a crossing rectangle from the upper right of the group of page 2+ elements to lower left. This should include the two graphic symbol entities, the two description text entities, and the dividing line.
5. Click **OK**. A magenta block insert tag appears, and the contents of the block are also magenta.
6. Double-click the block insert. The **BLOCK PROPERTIES** dialog box appears. Enter the following property setting:

Field	Value	Comments
X	11	Places the block insert directly above the block in REPORT DESIGN. No impact at output time.
Y	10	Same as current value.
Offset X	-8.0	Moves the block 8" to the left on output.

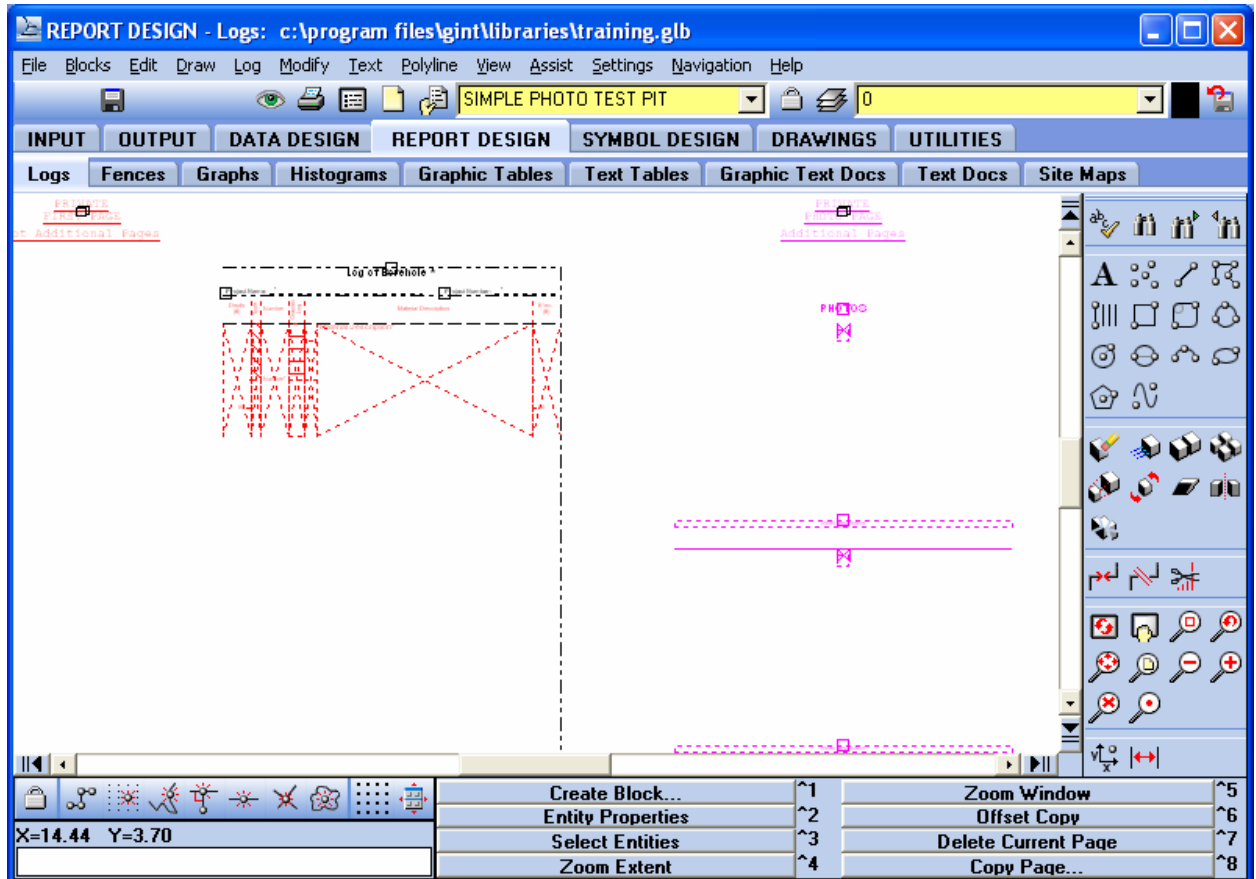
7. Click **OK** to close the dialog box.

Now we create the page 1 block.

1. Select **Blocks ► Create Block**. The CREATE BLOCK dialog box appears.
2. Enter the following values:


Field	Value	Comments
Type	Private (current page)	
Name	FIRST PAGE	
Page Range	Not Additional Pages	Restricts the output of this block to the test pit data, typically just page 1.
Design Mode Color	Very Light Red	

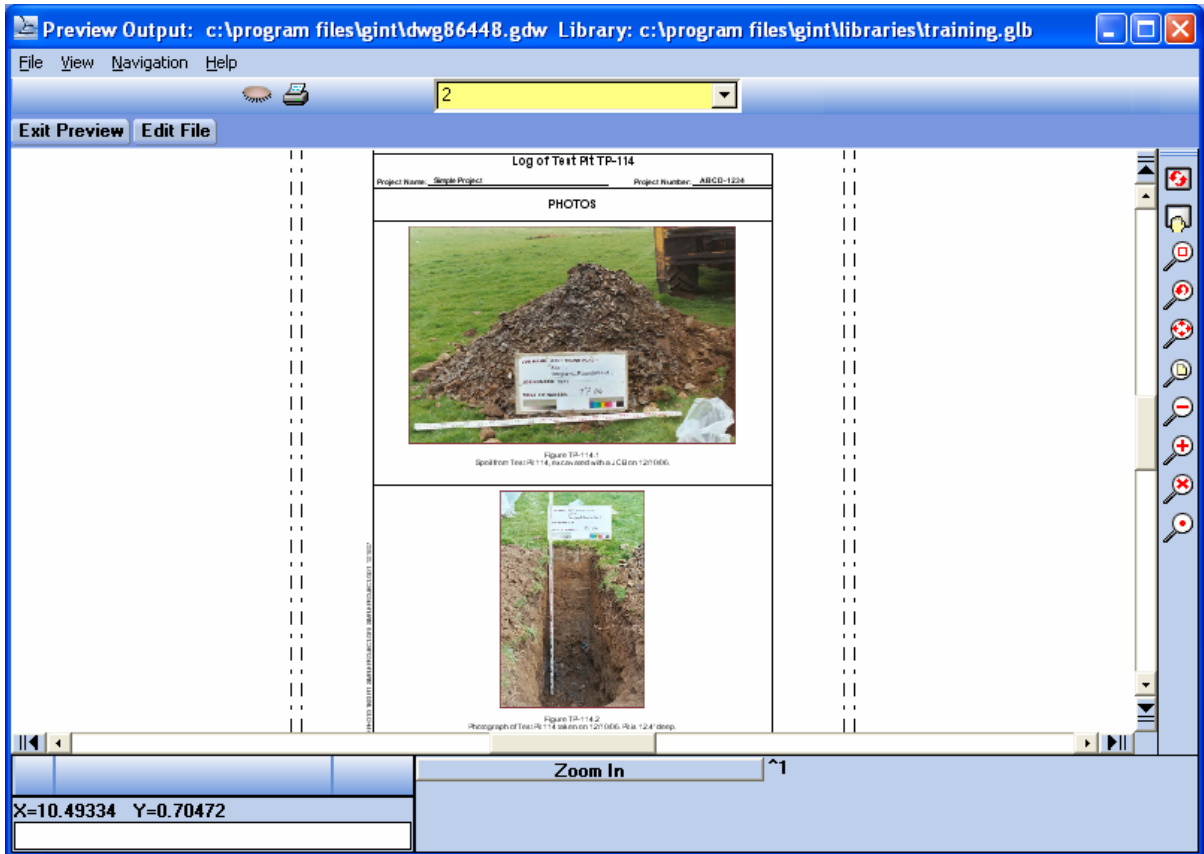
3. Click **OK**. The 'Select Entities' prompt appears.
4. Draw a crossing rectangle from the upper right of the group of page 1 elements to lower left. This should include the seven column entities, but not the header.
5. Click **OK**. A red block insert tag appears, and the contents of the block are also red.



Everything is in place for the report to generate additional pages of photos. Do the following:

1. Go to **OUTPUT**.
2. Click the **Browse** button to the right of **Borehole Number**. Select borehole (test pit) TP-114 and click **OK**.

- Click the Preview  button.
- Notice the borehole log on page 1 of the report. Select page 2 (and subsequent pages) in the object selector, and note the photo pages.



- Zoom in on one of the photo descriptions. Notice that the figures have been successively numbered using the borehole number as a prefix, and the description has been extracted from the **Bitmap File** field for the correct photo.



Figure TP-114.3
Alternative photo of spoil from pit 114, excavated on 12/10/06. Very hard ground.

Variable Graphic Legends on Reports

A variable legend entity creates a legend displaying the material, sampler, well or other symbols in use in a single log, all boreholes on a fence report, or throughout a project (displayed in a graphic text doc). This is a useful visual aid on a report. The legend is called variable because the set of symbol boxes that appears will vary from log to log or fence to fence, depending on which symbols are in use.


We will create variable legends on a log and a fence, and create a graphic text document showing all material and sampler symbols for the project.

Log Report Legend

First we create the log, which will have a legend for sampler graphics.

1. Ensure that the current library is **training.glb**.
2. Go to **REPORT DESIGN ► Logs**. In the object selector select **SIMPLE LOG** (or **FINAL SIMPLE LOG** if **SIMPLE LOG** is missing or incomplete).
3. Select **File ► Copy Page**. Enter the following, then click **OK**:

Field	Value
Page Name to Copy To	SIMPLE LOG WITH LEGEND
Set Current Page To	First in Copy To List

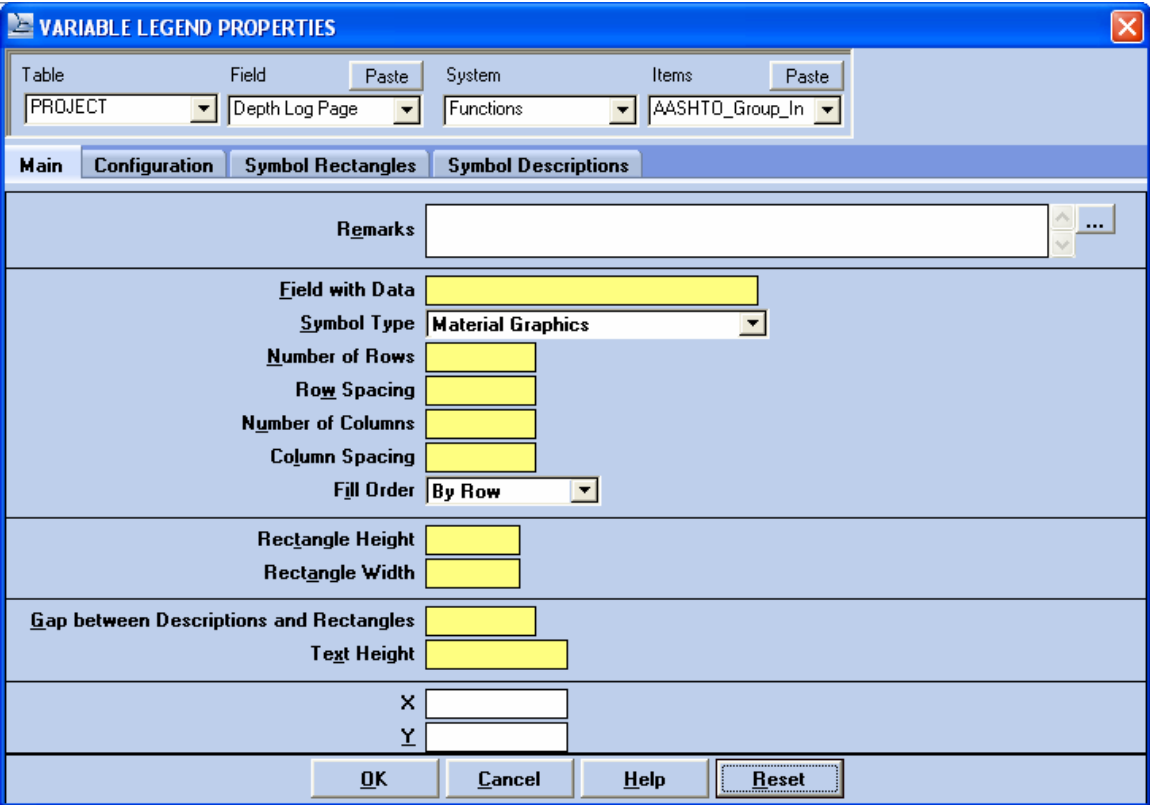
4. We need to create a footer for the legend, and shorten the body height (where column entities reside) to make room for the footer. Click the Properties  icon, then select the **Report Structure** tab. Enter the following:

Field	Value
Page 1 Body Height	7.5
Page 1 Footer Height	0.5

We only make these changes for Page 1. On subsequent pages the legend will not appear.

5. Select the **Columns and Lines** tab. Check the **Bottom of Body** checkbox.
6. Click **OK** to close the Properties window.

7. Select Log ► Variable Legend. The VARIABLE LEGEND PROPERTIES window appears.



8. Enter the following in the Main tab:

Field	Value	Comments
Field with Data	<<SAMPLE.Type>>	The field containing symbol names. Its values determine which symbol graphics print. Use the Data Tool.
Symbol Type	Sampler Graphics	Determines which kind of symbol is displayed in the legend.
Number of Rows	1	One row of symbols will appear in the legend (we have limited vertical space, and there will be few sample graphics per log).
Row Spacing	0.4	Ignored in a 1-row legend.
No. of Columns	5	Maximum number of columns of symbols displayed. Note that num_of_columns x column_spacing should be the width of the report body, 6" in this case.
Column Spacing	1.175	Distance from the start of one symbol to the start of the next. Can also be thought of as column width.
Rectangle Height	0.4	Rectangle height of each symbol.
Rectangle Width	0.15	
Gap Between Descriptions and Rectangles	0.05	Distance from the right edge of each rectangle to the left edge of its description.
Text Height	0.08	Character size of description text.
X	0.05	X coordinate of legend alignment point (which is at upper left corner of legend)
Y	0.45	Y coordinate of alignment point.

9. Go to the Configuration tab. Enter the following:

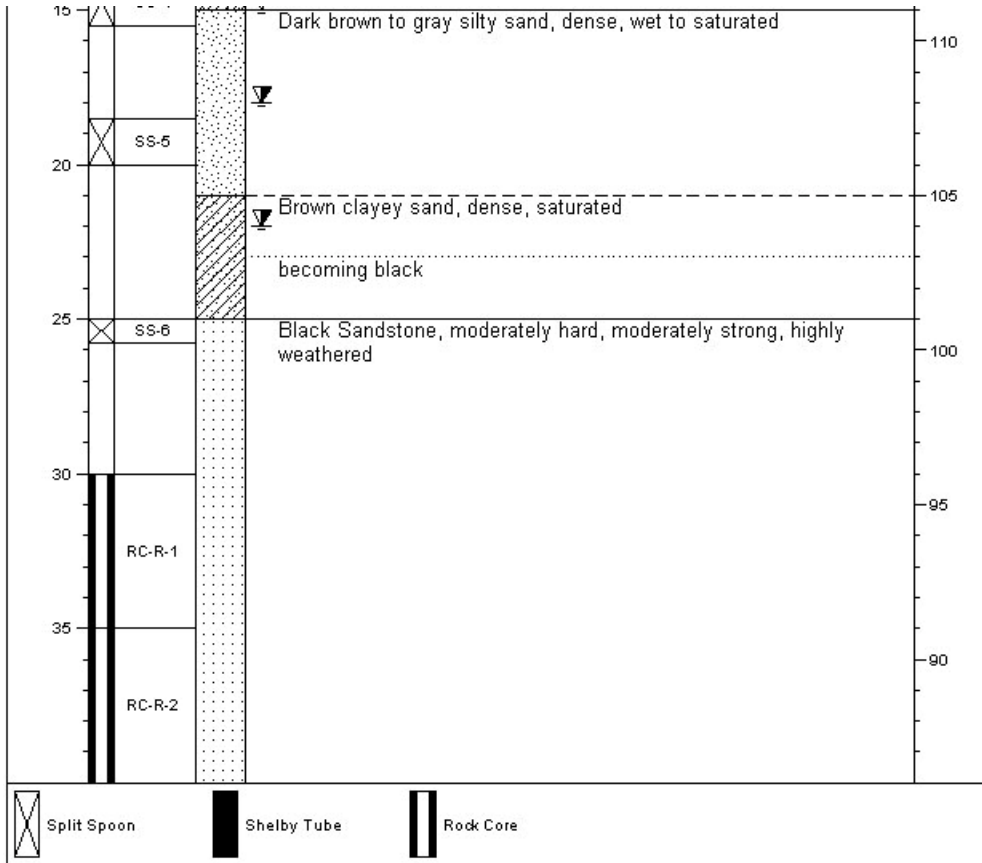
Field	Value	Comments
Output Condition	<<Pg>>=1	Restricts display of the legend entity to page 1. Use the Data Tool (right half) to paste the Pg system data item.

10. Go to the Symbol Descriptions tab. Enter the following:

Field	Value	Comments
Name and/or Description	Description	Specifies what (from the Symbol record) will appear next to each symbol.

11. Click OK. The variable legend entity is added to the report.

12. Go to OUTPUT, and preview the report.



Notice the legend at the bottom.


13. Close the preview.

Fence Report Legend

We will create a material graphic legend for a fence report next. In a fence report, a legend displays all symbols used of the specified type (material in this case) in all boreholes of the report. Do the following:

1. Go to REPORT DESIGN ► Fences.
2. In the object selector select SIMPLE FENCE (or FINAL SIMPLE FENCE if SIMPLE FENCE is missing or incomplete).
3. Select File ► Copy Page. Enter the following, then click OK:

Field	Value
Page Name to Copy To	SIMPLE FENCE WITH LEGEND
Set Current Page To	First in Copy To List

4. Some modifications are required to the data frame layout. Click the Properties  icon. The FENCE REPORT PROPERTIES window appears.

5. In the Data Frame tab, enter the following settings:

Field	Value	Comments
Frame Y	0.95	Elevates the frame bottom by .45" to make room for the legend.
Frame Height	5.25	Shortens the frame by .25" so it doesn't run into the header.

6. Click OK to save the property changes.
 7. Double-click on the 'Distance Along Baseline (ft)' entity. The TEXT PROPERTIES window opens.
 8. In the Main tab, enter the following, then click OK:

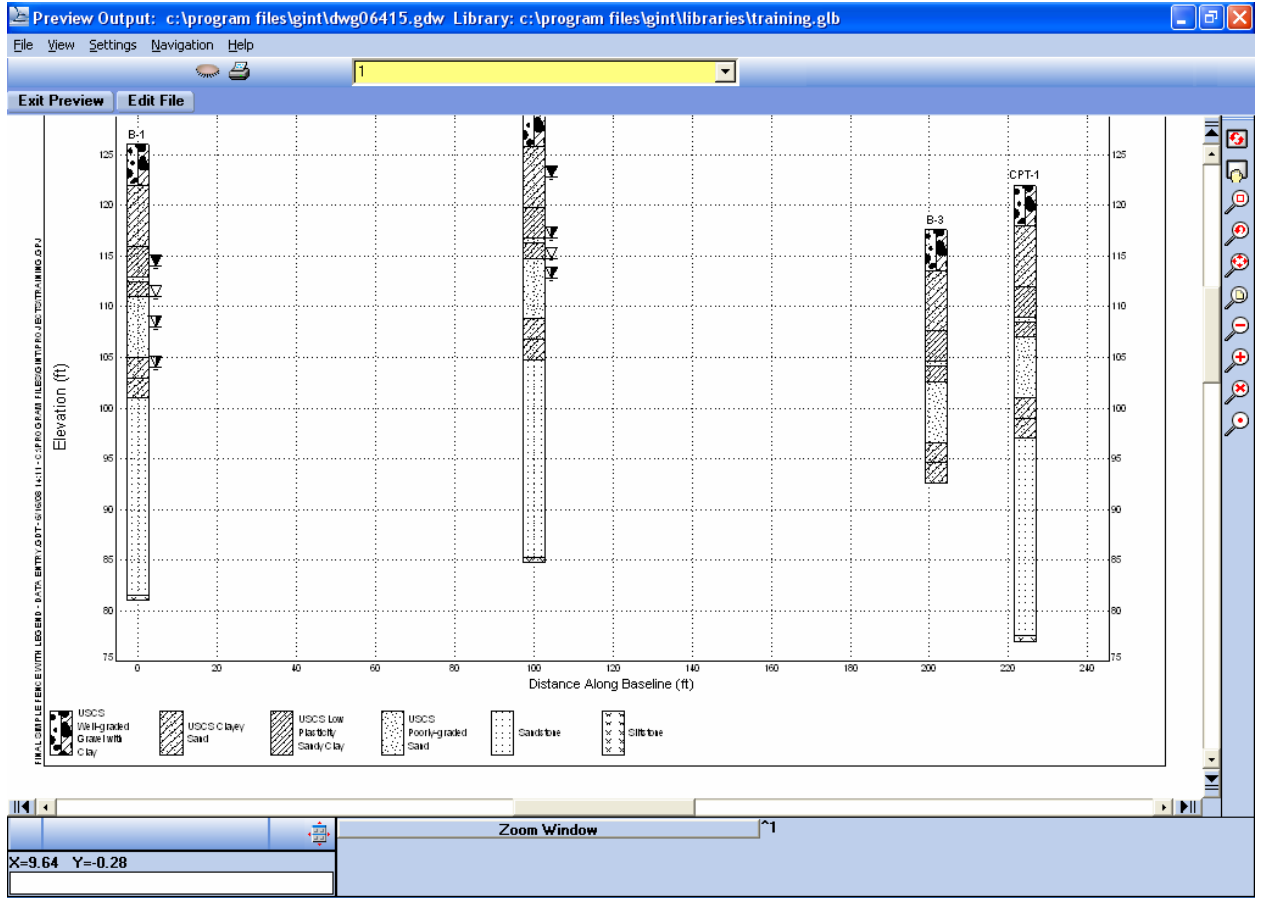
Field	Value	Comments
Y	0.7	Elevates the text entity by .6" to keep it just beneath the elevated data frame.

9. Select Fence ► Variable Legend. The VARIABLE LEGEND dialog box appears.
 10. Enter the following values:

Field	Value	Comments
Field with Data	<<LITHOLOGY.Graphic>>	The data source will be the Graphic field in the LITHOLOGY table. Use the Data Tool.
Symbol Type	Material Graphics	The legend will collect all material symbols used in the fence report.
Number of Rows	1	We use a single-row legend to avoid having to compress the data frame too much.
Row Spacing	0.4	Ignored in a single-row legend.
Number of Columns	10	A maximum of 10 symbols can be displayed. To increase this we would have to narrow the column spacing or add a row.
Column Spacing	1	We use a narrower column than in the log.
Rectangle Height	0.4	Same as the log.
Rectangle Width	0.2	Slightly wider than the log report's .15"
Gap between Descriptions and Rectangles	0.05	Same as the log.
Text Height	0.08	Character height. Same as the log.
X	0.05	Alignment point is .05" to right of the report border.
Y	0.5	Alignment point is .5" above the report border. This allows room for the .4" high graphic rectangles and a reasonable amount of text.

11. Click OK to save the legend properties.


- Go to **OUTPUT ► Fences** and preview the report. (Cancel the User Report Variables window if desired.) The report appears as shown:



- Close the preview.

Graphic Text Doc Legends

A good use of the Graphic Text Doc report type and variable legends is to report all the material, sampler, well, and/or general component graphics in use throughout the project. We will create the report from scratch, and include two legends, one each for material and sampler graphics.


1. In REPORT DESIGN, select the Graphic Text Docs tab.
2. Click the New  icon or select **File ► New**. The 'New...' dialog box appears.
3. In the General tab, click **Reset**, then enter the following:

Field	Value	Comments
Name	SIMPLE LEGEND	
Description	Legend of material and sampler graphics used	
Left Margin	0.6	
Bottom Margin	0.425	
Page Size	Letter	
Orientation	Portrait	
Template	data entry.gdt	
Key Set	Project	Specifies the level at which data is consolidated.


4. Click **OK**. The empty report is created.
5. We will use the TITLE PORTRAIT block from the Drawing Library to create borderlines and header text. Select **Blocks ► Insert Block**. In the **BLOCK PROPERTIES** dialog box, click **Reset** then enter the following:

Field	Value	Comments
Block Source	Dwglib!title portrait	Click the Browse icon, select a Type of 'DWGLIB' and a Block of 'TITLE PORTRAIT'.
Design Mode Color	Very Light Red	Color in REPORT DESIGN for the block insert and contents.
X	-2	X coordinate of alignment point for block insert. No effect on output.
Y	8.5	See X.

6. Click **OK**.

7. The report needs a title. Select **Draw ► Text ► Text** or click the Text  icon, click **Reset**, then enter the following:

Field	Value
Text Expression	PROJECT SYMBOL LEGEND
Height	0.14
Horz Align	Right
Vert Align	Top
X	7.5
Y	10.1

8. Click **OK**.
9. We need a centered title above the first legend, which is for Material graphics. Select **Draw ► Text ► Text** or click the Text  icon, click **Reset**, then enter the following:

Field	Value
Text Expression	Material Graphics
Height	0.14
Horz Align	Center
Vert Align	Top
X	3.8
Y	8.85

10. Click **OK**.


11. Next we create a variable legend for material graphics. Select **Graphic Text ► Variable Legend**, click **Reset**, then enter the following:

Field	Value	Comments
Field with Data	<<LITHOLOGY.Graphic>>	Source field for symbols. Use the Data Tool.
Symbol Type	Material Graphics	Type of symbol to display.
Number of Rows	4	Four rows of symbols.
Row Spacing	0.6	Row height. Needs to exceed rectangle height by some small distance for spacing.
Number of Columns	6	Six columns of symbols.
Column Spacing	1.25	Width of each column.
Fill Order	By Row	Symbols populate the legend from left to right.
Rectangle Height	0.4	Symbol rectangle height.
Rectangle Width	0.4	Symbol rectangle width.
Gap Between Descriptions and Rectangles	0.04	
Text Height	0.08	
X	0.06	
Y	8.6	

12. Go to the **Symbol Descriptions** tab and specify the following:

Field	Value	Comments
Name and/or Description	Name: Description	Specifies whether the Name (symbol code), Description or both appear, and if both, in what format. We're printing the code first, then a colon, then the description.

13. Click **OK**.

14. To add a centered title above the sampler graphics legend, select **Draw ► Text ► Text** or click the Text  icon, then enter the following:

Field	Value
Text Expression	Sampler Graphics
Height	0.14
Horz Align	Center
Vert Align	Top
X	3.8
Y	6.0

15. Click **OK**.

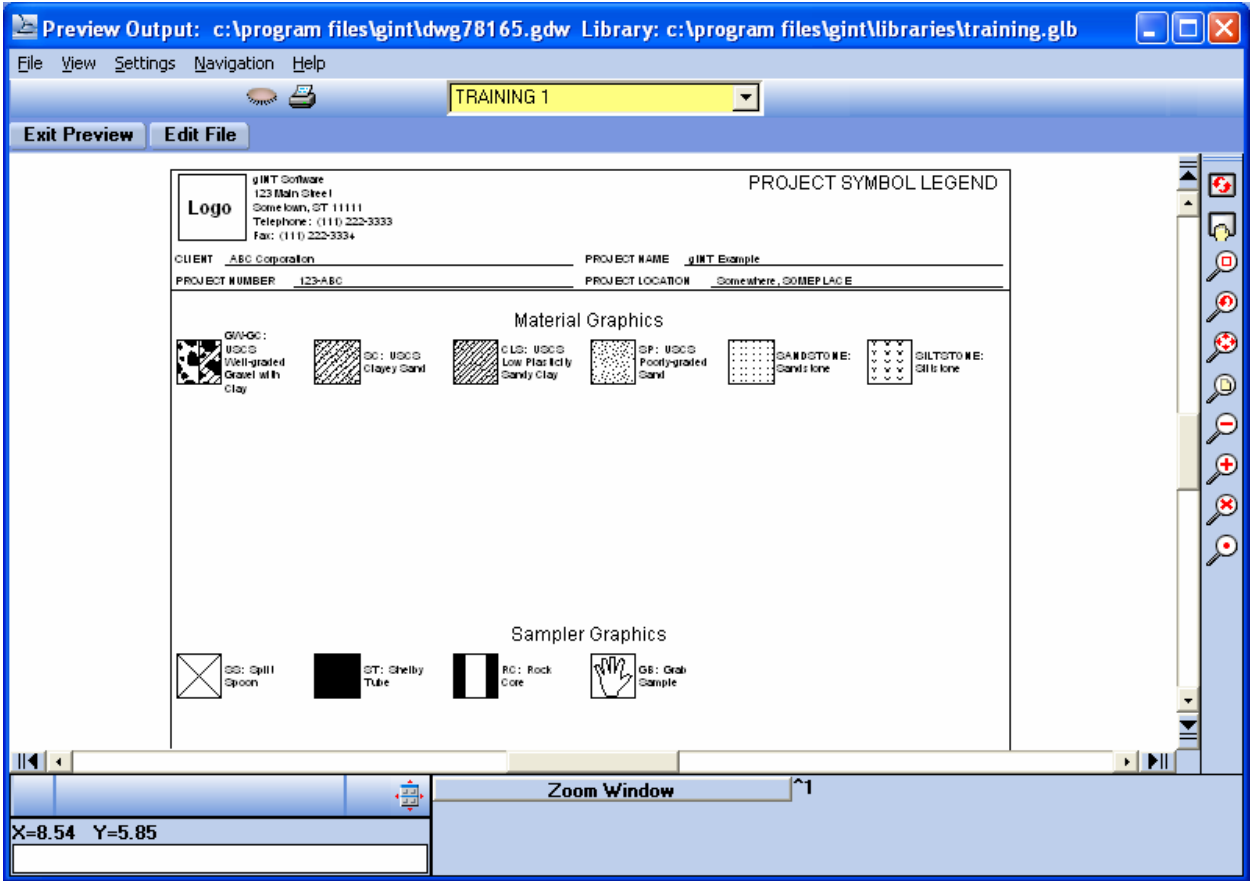
16. To create a variable legend for sampler graphics, select **Graphic Text ► Variable Legend**, click **Reset**, then enter the following in the **Main** tab:

Field	Value
Field with Data	<<SAMPLE.Type>>
Symbol Type	Sampler Graphics
Number of Rows	2
Row Spacing	0.6
Number of Columns	6
Column Spacing	1.25
Fill Order	By Row
Rectangle Height	0.4
Rectangle Width	0.4
Gap Between Descriptions and Rectangles	0.04
Text Height	0.08
X	0.06
Y	5.75

17. Go to the **Symbol Descriptions** tab and specify the following:

Field	Value
Name and/or Description	Name: Description

18. Click **OK**.
 19. Go to **OUTPUT ► Graphic Text Docs** and preview the result.



20. Close the preview.

A useful variant is the vertical project symbol report. To create this, do the following:

1. Go to REPORT DESIGN ► Graphic Text Docs.
2. Select File ► Copy Page. Name the copy 'SIMPLE VERTICAL LEGEND'.
3. Double-click the 'Material Graphics' text entity, change the following values, then click OK:

Field	Value
Horz Align	Left
X	0.06

4. Double-click the 'Sampler Graphics' text entity, change the following values, then click OK:

Field	Value
Horz Align	Left
X	3.8
Y	8.85

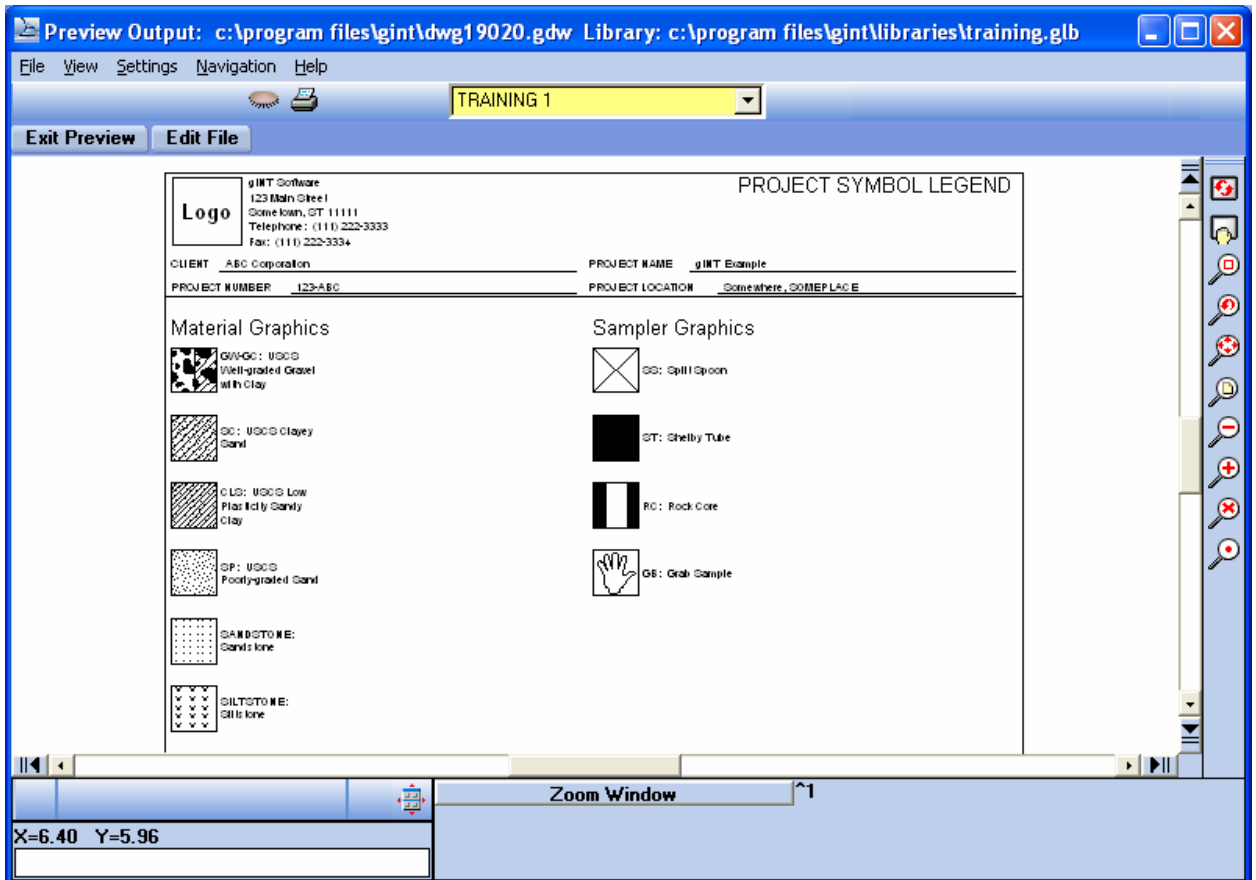
- Double-click the material graphics variable legend entity, change the following values, then click OK:

Field	Value
Number of Rows	14
Number of Columns	2
Column Spacing	1.5
Fill Order	By Column

- Double-click the sampler graphics variable legend entity, change the following values, then click OK:

Field	Value
Number of Rows	14
Number of Columns	2
Column Spacing	1.5
Fill Order	By Column
X	3.8
Y	8.6

- Go to OUTPUT ► Graphic Text Docs and preview the report.



- Close the preview.

Symbol Legend Positions

The Variable Legend entity in logs, fences, and graphic text documents, by default, prints the symbols in the order that they were encountered in the output. You can override this default behavior for sampler, material, well and/or general symbols using the Symbol Legend Positions feature in **SYMBOL DESIGN**, which enables you to assign rankings to symbols for display in legends.

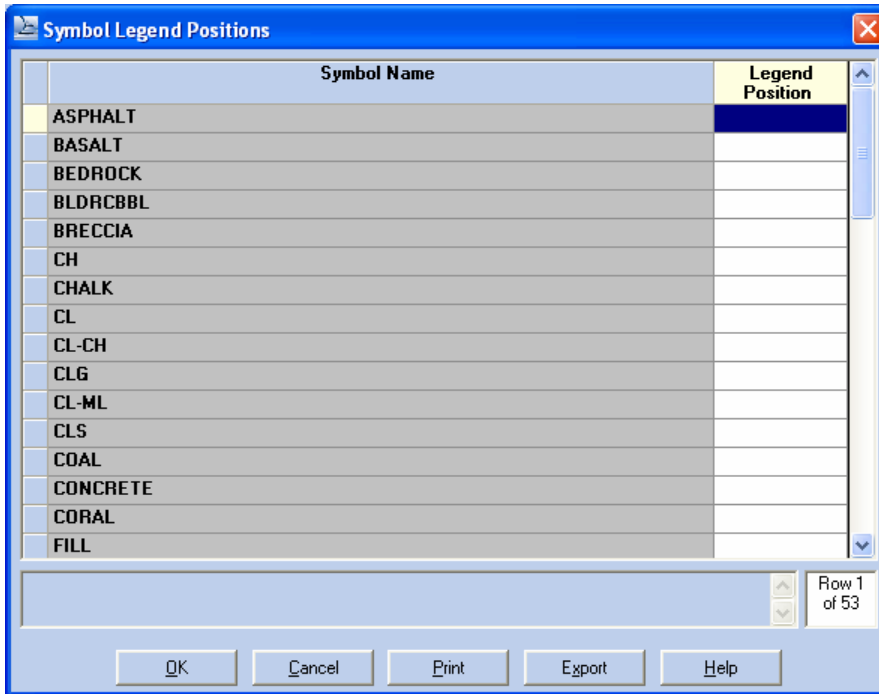
For example, you might want to order material symbols to show gravel symbols first, followed by sand, silt, clay, rock, and all others. To do this you would assign an integer to all gravels you use, a higher integer for all sands, and so on. You can skip values. It is only the relative values that are important, that is, 5 will sort before 8 and it doesn't matter that there are no 6 or 7 values. Also, you can have multiple symbols with the same legend position values, and symbols with unspecified values are placed at the end of the list.

We'll assign the following values to material symbols used in our project:

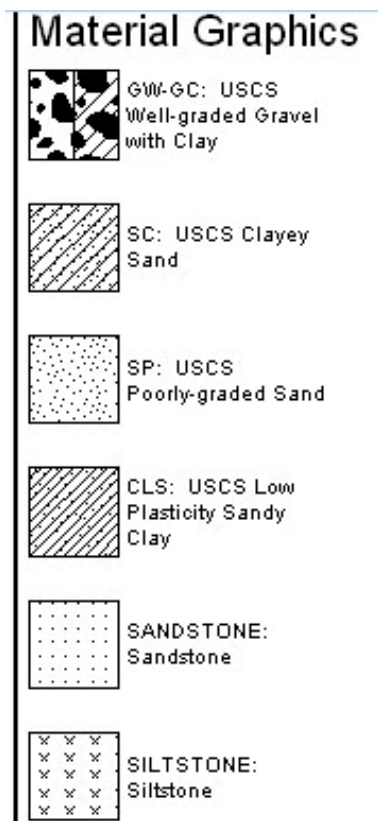
Code	Description	Legend Position
CL	USCS Low Plasticity Clay	4
CLS	USCS Low Plasticity Sandy Clay	4
GW-GC	USCS Well-graded Gravel with Clay	1
ML	USCS Silt	3
SANDSTONE	Sandstone	5
SC	USCS Clayey Sand	2
SILTSTONE	Siltstone	5
SM	USCS Silty Sand	2
SP	USCS Poorly-graded Sand	2

To demonstrate the feature, do the following:

- Go to **SYMBOL DESIGN** and select the **Material** tab.
- Select **File ► Symbol Legend Positions**. The **Symbol Legend Positions** dialog box appears.




3. Enter the legend position values in the corresponding rows from the list above, then click OK.
4. Go to OUTPUT ► Graphic Text Docs. Preview the output. The material symbols are now in gravel-sand-silt-clay-rock-other order.




5. Close the preview.

Appendix A -- Sorting Boreholes using a GintPointSort Field

By default, POINT records are sorted based on the alphanumeric borehole numbers you assign to them in the **PointID** field, and the sorting follows standard character sort order. This affects the order of output of log reports, of records in graphic and text tables, and of boreholes in object selector dropdowns. There are situations in which you need to specify a custom sort order for boreholes so that they output in a different order from the default. This is accomplished by adding a **GintPointSort** field to the POINT table. To demonstrate this (using **training.gpj** and **training.glb** from the [Standard Data Files for All Tutorials](#)), do the following:

1. Go to **INPUT ► Lithology** and click the object selector drop-down arrow. Notice that the order is 'B-1', 'B-2', 'B-3', 'CPT-1', 'TP-1' (followed by 'TP-114', if present). Press Esc.
2. Go to **DATA DESIGN ► Project Database** and open the current project (**training.gpj**). Select 'POINT' in the object selector.
3. Select **Tables ► Point Sort Field**. Notice that **GintPointSort** has been added to the bottom of the **Fields** list.
4. Highlight **GintPointSort** and move it up by repeatedly clicking the Up Arrow  icon until it is between **PointID** and **HoleDepth** in the list.
5. Go to **INPUT ► Borehole**. Enter the following values in the **GintPointSort** column:

Borehole No.	GintPointSort Value
B-1	3
B-2	4
B-3	5
CPT-1	2
TP-1	1
TP-114 (if present)	6

6. Click the **Lithology** tab. Click the object selector drop-down arrow. Notice that the listing order is now 'TP-1', 'CPT-1', 'B-1', 'B-2', 'B-3', 'TP-114'.
7. Go to **OUTPUT ► Logs**. Click in the **Sort 1** field.
8. In the Data Tool under the **FILTER** heading, select a **Table** of 'POINT' and a **Field** of 'GintPointSort', then click **Paste**. '[POINT].[GintPointSort]' appears in the **Sort 1** field.
9. Select 'ENVIRONMENTAL BH' in the object selector and preview the report. Page through the report pages using the object selector, and notice that the order of output is 'TP-1', 'CPT-1', 'B-1', 'B-2', 'B-3', 'TP-114'. Close the preview.
10. Go to **REPORT DESIGN ► Logs**. Select the Properties  icon. Notice the **Sort Field 1** property. You would enter the same value here as you used in **OUTPUT**, namely '[POINT].[GintPointSort]', to make this sort order permanent for the report. We will not do this at this time. Click **Cancel**.

11. Close the preview.

- ☞ **Note:** If you have many reports within a particular report type (such as logs), you can use the **File ► Page ► Copy Page Properties** command to copy the **Sort Field 1** property value to the other reports.

- ☞ **Note:** A gINT Rule can be written to insert the values into **GintPointSort** according to some predefined scheme. Download and install **gra007.zip** from http://www.gintsoftware.com/support_gintrules.html for an example of how this can be done. The **gra007.txt** file provides an explanation and implementation details.