

## Data Granularity in the Storage and Reporting of Soil Exploration Information

"Granularity" is a term used extensively in the software programming arena. It also has applicability to the storage of data. Coarse-grained data structures generally store results exactly how the designer wants to report the information on a particular report or reports. This can combine a number of pieces of information into one piece of data.

Data that has "coarse" granularity gives less information than a fine "grained" data. Coarse data hides its origins. Coarse-grained database designs are more difficult to check. They also lose information that may be useful in some context in the future or to some other party.

Fine-grained data structures are generally easier to input, afford better error checking, and expand the possibilities for reporting and querying. The degree of fineness implemented depends on the needs of each organization.

## Samples of Coarse and Fine-Grained Data Structures

Following is a simple a log report:

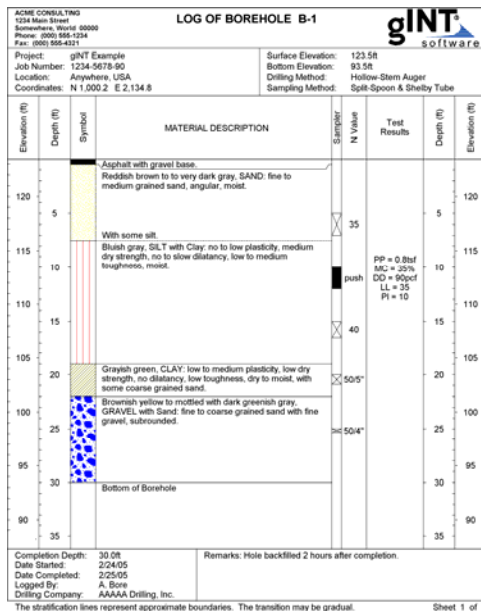


Figure 1. Simple Log.

You cannot tell by the output whether the data are coarse or fine-grained. With either method, the results would look the same. The following sections describe both formats.

## Coarse-Grained Data Design Samples

What is typically done in designing a database is to use the desired output (the log report in Figure 1 in this case) as the design specification and to set up the database to match. Figures 2 and 3 show the designs that can result for the sample and stratigraphy data:

Depth (ft)	Length (ft)	Type	N Value	Test Results
5	2	SS	35	
10	2	SH	push	PP = 0.8tsf MC = 35% DD = 90pcf LL = 35 PI = 10
15	1.5	SS	40	
20	0.9	SS	50/5"	
25	0.33	SS	50/4"	

Figure 2. Coarse Sample Data.

Depth (ft)	Bottom (ft)	Graphic	Description	Layer Line Type
0	0.5	ASPHALT	Asphalt with gravel base.	
0.5	7.5	SP	Reddish brown to to very dark gray, SAND: fine to medium grained sand, angular, moist.	Dash
6.5			With some silt.	
7.5	19	ML	Bluish gray, SILT with Clay: no to low plasticity, medium dry strength, no to slow dilatancy, low to medium toughness, moist.	
19	22	CL	Grayish green, CLAY: low to medium plasticity, low dry strength, no dilatancy, low toughness, dry to moist, with some coarse grained sand.	
22	30	GWS	Brownish yellow to mottled with dark greenish gray, GRAVEL with Sand: fine to coarse grained sand with fine gravel, subrounded.	

Figure 3. Coarse Stratigraphy Data.

These designs are easy to set up and reflect the exact requirements of the desired report.

## Fine-Grained Data Design Samples

Following are the fine-grained equivalents of the tables in Figures 2 and 3:

Depth (ft)	Length (ft)	Type	Blows 1st	Blows 2nd	Blows 3rd	Pocket Pen (tsf)	Dry Unit Wt (pcf)	Moisture Content (%)	Liquid Limit	Plastic Limit
5	2	SS	15	20	15					
10	2	SH	push			0.8	90	35	35	25
15	1.5	SS	10	18	22					
20	0.9	SS	15	50/5"						
25	0.33	SS	50/4"							

Figure 4. Fine Sample Data.

Depth (ft)	Bottom (ft)	Graphic	Color	Additional Color	USCS Name	Grain Size	Angularity	Plasticity	Dry Strength	Dilatancy	Toughness	Moisture	Additional Description	Layer Line Type
0	0.5	ASPHALT											Asphalt with gravel base	
0.5	7.5	SP	reddish brown	to very dark gray	SAND	fine to medium grained sand	angular					moist		Dash
6.5													with some silt	
7.5	19	ML	bluish gray		SILT with Clay			no to low	medium	no to slow	low to medium	moist		
19	22	CL	grayish green		CLAY			low to medium	low	no	low	dry to moist	with some coarse grained sand	
22	30	GWS	brownish yellow	mottled with dark greenish gray	GRAVEL with Sand	fine to coarse grained sand	subrounded							

Figure 5. Fine Stratigraphy Data (adapted from ASTM D-2488).

## Why Fine-Grained Design?

Fine-grained databases store the collected data, not necessarily a particular presentation of the data. It requires more set-up time but pays large dividends in terms of time saving, data quality and consistency, reporting, and querying. A close look at the three log report columns shown in Figure 1 illustrate some of the advantages.

### N Values

A database can be designed with a field just for N Value or separate fields for each individual 6-inch segment. The former has the advantage of easier input and requires no manipulation for output. The latter reflects the actual collected data and avoids a person performing the N Value calculation. Although the calculation is simple, this is one more opportunity to generate an error. Calculations are best left to the software.

Recording individual blows can reflect information that is lost with storage of just the N Value. Let's say the individual blows were 10, 40, 8. The N Value of 48 loses the dramatic reduction of blows in the third interval, which probably means a layer change was encountered.

## Test Results

Following are some advantages to having individual fields for each test instead of one field for all the tests:

Requires less typing	The test labels and units are not needed in the database with the fine-grained model. The software supplies these items and arranges the order and formatting.
Allows calculation of PI	Like N Value, this is a trivial calculation but is one more opportunity for a mistake to be made.
More reporting capabilities	Any or all of the results can be reported elsewhere. For example, plots of moisture content versus depth or elevation and PI vs. LL could be made
Query opportunities	Database queries can be performed easily on any or all of the results.
Validation	Allowable minimum and maximum value rules can be established for each of the test types.

## Material Descriptions

The coarse-grained, single field approach to material descriptions we term the "blob" model. The fine-grained approach, we term the "component" model. There are many advantages to the component model:

Consistency	Consistently structured descriptions are always produced.
Enforced attribute selection	Most of the components would have valid data lists associated with them from which the user chooses. Such data lists aid in data accuracy and consistency. All of the above components, except for "Additional Description" are such lookup lists.
No formatting	Formatting is removed from the data. The burden of the formatting is taken from the user and imposed on the software. Further, the formatting can then be altered as needed without changing the data. Note that in Figure 5 above there is no punctuation. Punctuation is inserted by user-defined rules. Other component formatting, such as bold, italic, underline, color, etc., are also possible. Finally, the order of the components can be changed at will. User-definable software rules determine the final look of the description.
Selective reporting	Not all components may be desirable under some situations. On a log form the full description is usually shown. On a fence diagram showing a number of borehole sticks there is less room so perhaps only the "USCS Name" would be shown. To do this with the coarse-grained design would require creation of a new field to store the abbreviated description. Data redundancy now exists in the database, a condition that needs to be avoided.
Query opportunities	Queries can be run on any combination of components. For example, one could ask for N Value results within layers whose "USCS Name" contains "sand".
Validation	Data validation becomes possible. Simple rules could disallow toughness and dry strength from granular materials.

The level of granularity depends on how the data will be used (what information is important to the final usage of the data) and on how much the organization prizes consistency. We have seen component models as small as three components (principal material, strength, and additional description) and as large as 40. The sample above shows 11 soil components. Another table is also required for rock descriptions. As an example of how fine this process can go, seven component color models are sometimes used:

<b>Intensity 1</b>	<b>Qualifier 1</b>	<b>Color 1</b>	<b>Color Conjunction</b>	<b>Intensity 2</b>	<b>Qualifier 2</b>	<b>Color 2</b>
<b>light</b>	<b>greenish</b>	<b>red</b>	<b>to</b>	<b>dark</b>	<b>bluish</b>	<b>yellow</b>
<b>medium</b>	<b>pinkish</b>	<b>brown</b>	<b>and</b>	<b>light</b>	<b>grayish</b>	<b>green</b>
<b>very light</b>	<b>reddish</b>	<b>yellow</b>	<b>mottled with</b>	<b>very dark</b>	<b>yellowish</b>	<b>blue</b>

*Figure 6. A fine-grained color model.*

Each of these components would have a look up list of allowable choices associated with them.

## SUMMARY

The coarse-grained approach is adequate if your ultimate and only goal is to produce a specific report. The data structures are easy to implement and the data entry "logical" as far as the data entry person is concerned since there is a one-to-one relationship between the database fields and the report. This approach is can be looked at as being the electronic equivalent of entering data in a form on paper.

The main features of the fine-grained approach are that it:

- better reflects the collected data;
- eliminates data redundancy;
- removes the formatting function from the data entry process, thereby reducing data entry time and allowing formatting appropriate to each report;
- enforces consistency;
- expands the range of reporting dramatically;
- opens the database up to querying possibilities; and
- makes data validation possible.

As more data are moved from paper to electronic storage, it is crucial that granularity be addressed in the design of the storage database for the best short-term and long-term use of the data and to help ensure that the data are consistent and valid.