ODS Styles – Breaking the Rules and Creating New Ones

THE POWER TO KNOW®

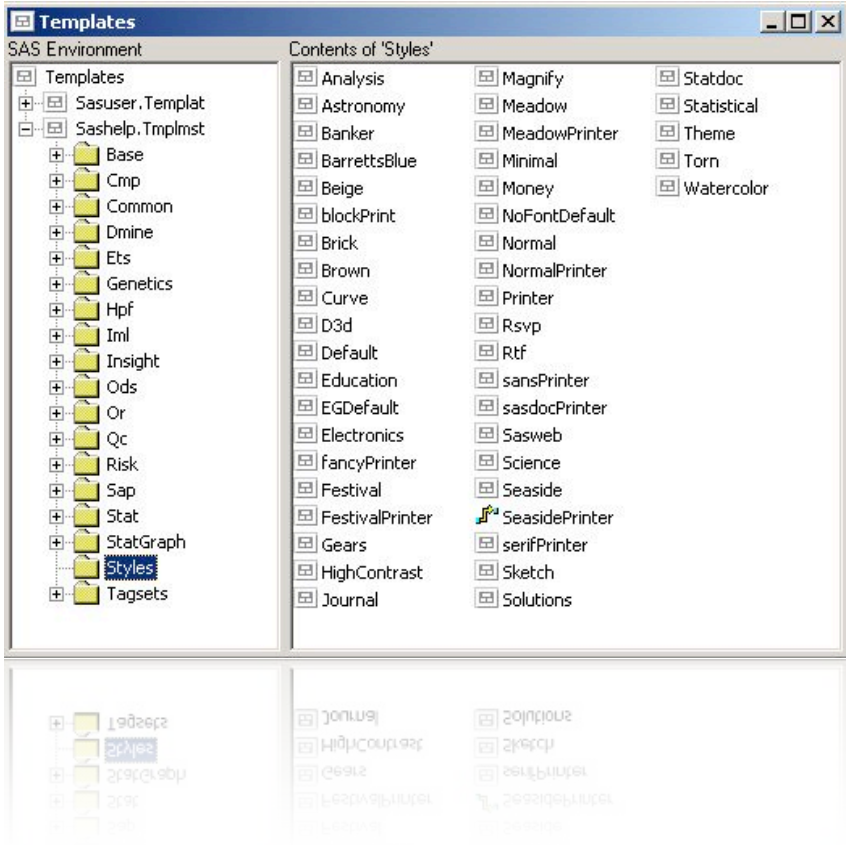
Kevin D. Smith
Base SAS Research & Development
SAS, Inc.

Copyright © 2006, SAS Institute Inc. All rights reserved.

- Most people think Proc Template styles are like an onion
    - Seem innocent enough at first glance
    - Get deeper and you see that it's full of layers, makes you want to cry, and it stinks
- But onions are very versatile, you can do lots of things with onions just like Proc Template styles

Exploring Templates

- Template Browser
- Proc Template
  List and Source
  Statements



Copyright © 2006, SAS Institute Inc. All rights reserved.

- The 'odstemplates' command opens template browser, works just like Windows Explorer
- From SAS code, use Proc Template
- ODS only uses what's in the ODS PATH

Exploring Templates

- Template Browser
- Proc Template
  List and Source
  Statements

ods path show;

Copyright © 2006, SAS Institute Inc. All rights reserved.

- The 'odstemplates' command opens template browser, works just like Windows Explorer
- From SAS code, use Proc Template
- ODS only uses what's in the ODS PATH

Copyright © 2006, SAS Institute Inc. All rights reserved.

- Look in Styles template store for SAS supplied styles
- Use STYLE= on ODS statement to choose a style

Copyright © 2006, SAS Institute Inc. All rights reserved.

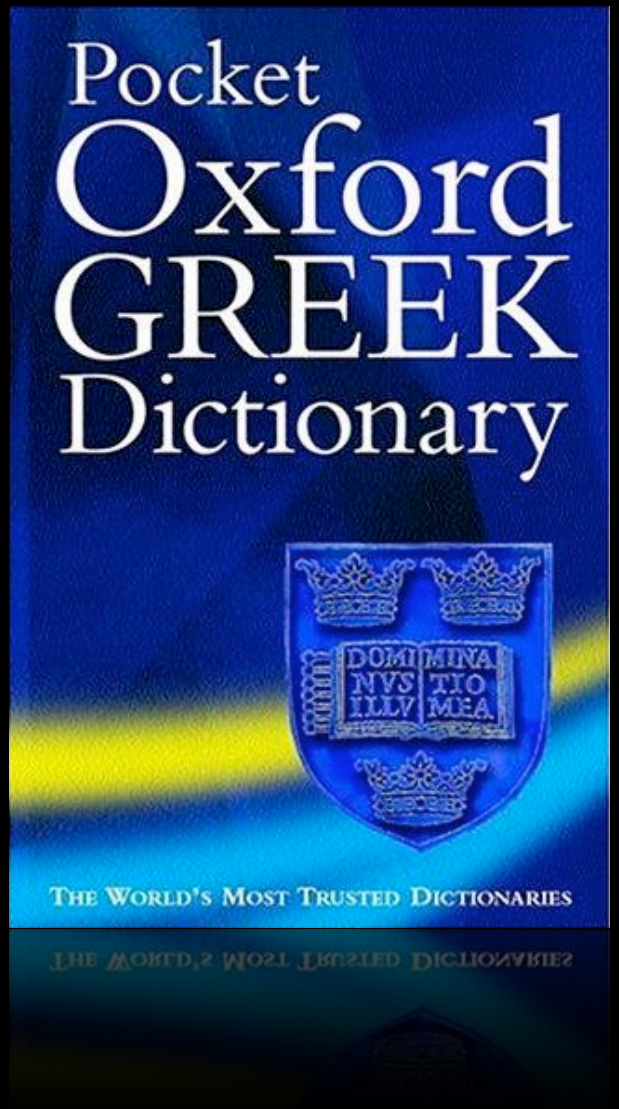
- Look in Styles template store for SAS supplied styles
- Use STYLE= on ODS statement to choose a style

ods pdf **style = styles.fancyprinter ... ;**

Obs	*Name*	*Sex*	*Age*	*Height*	*Weight*
1	Alfred	M	14	69.0	112.5
2	Alice	F	13	56.5	84.0
3	Barbara	F	13	65.3	98.0
4	Carol	F	14	62.8	102.5
5	Henry	M	14	63.5	102.5
6	James	M	12	57.3	83.0
7	Jane	F	12	59.8	84.5
8	Janet	F	15	62.5	112.5

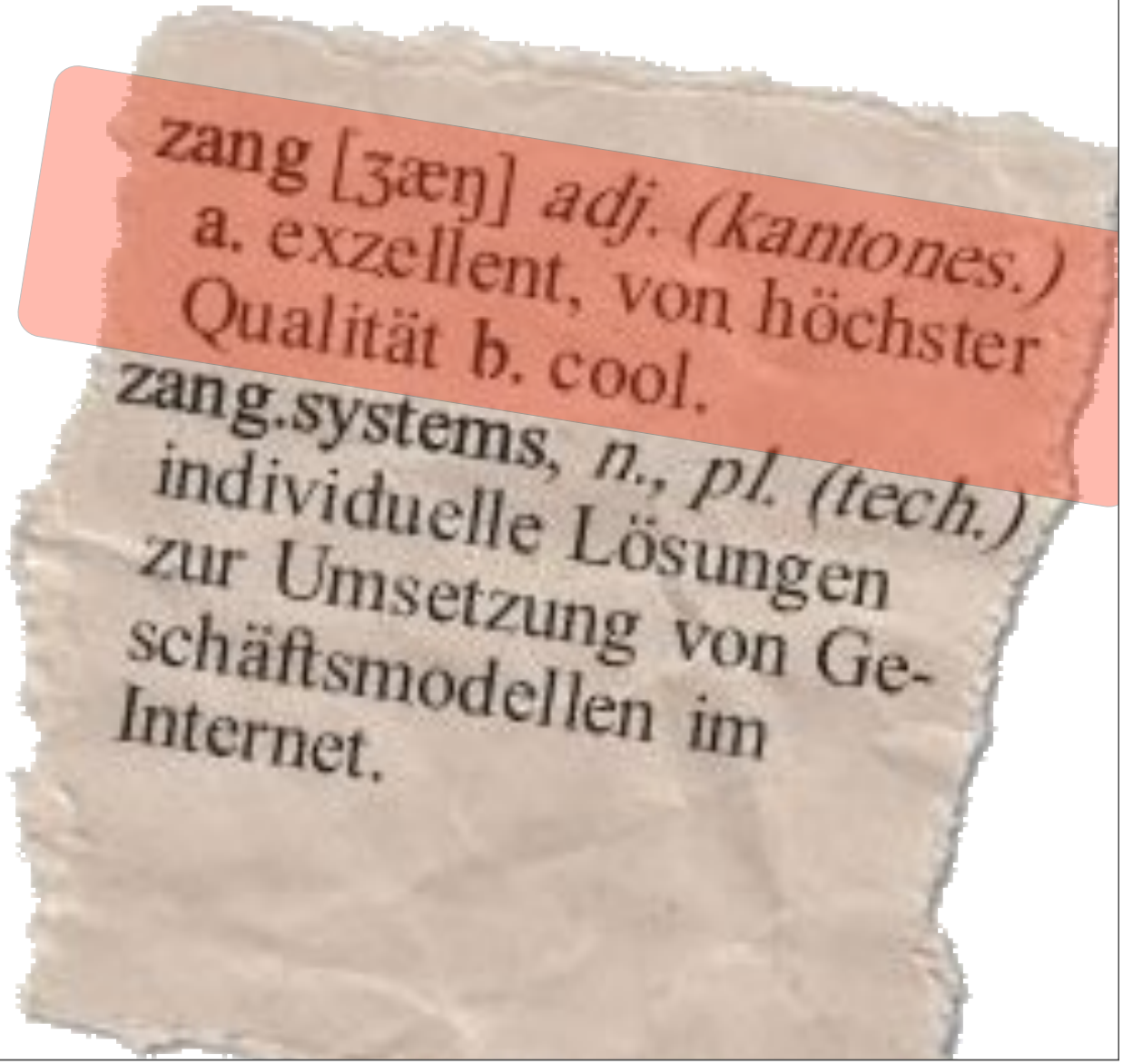
- Need to know the structure of a style to do really fun stuff

Style Definition



- Style definition is like a dictionary (in Greek)
- Collection of definitions looked up by name
- Only need to know the names that ODS uses

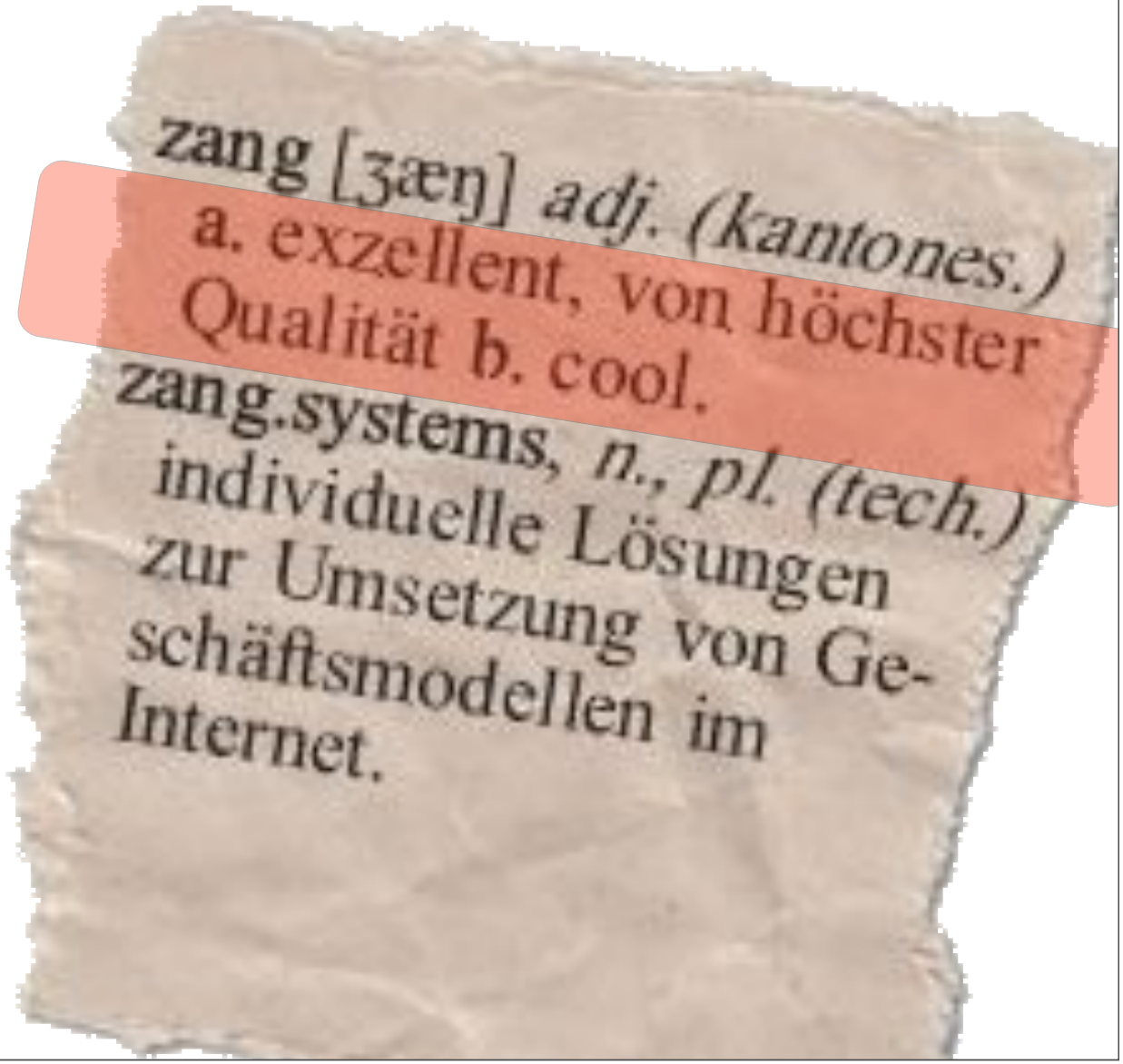
Style Element



Copyright © 2006, SAS Institute Inc. All rights reserved.

- Style element is a name and a description
- Names looked up by ODS

Style Attributes



Copyright © 2006, SAS Institute Inc. All rights reserved.

- Style attributes describe what at element looks like
  - colors
  - fonts
  - borders

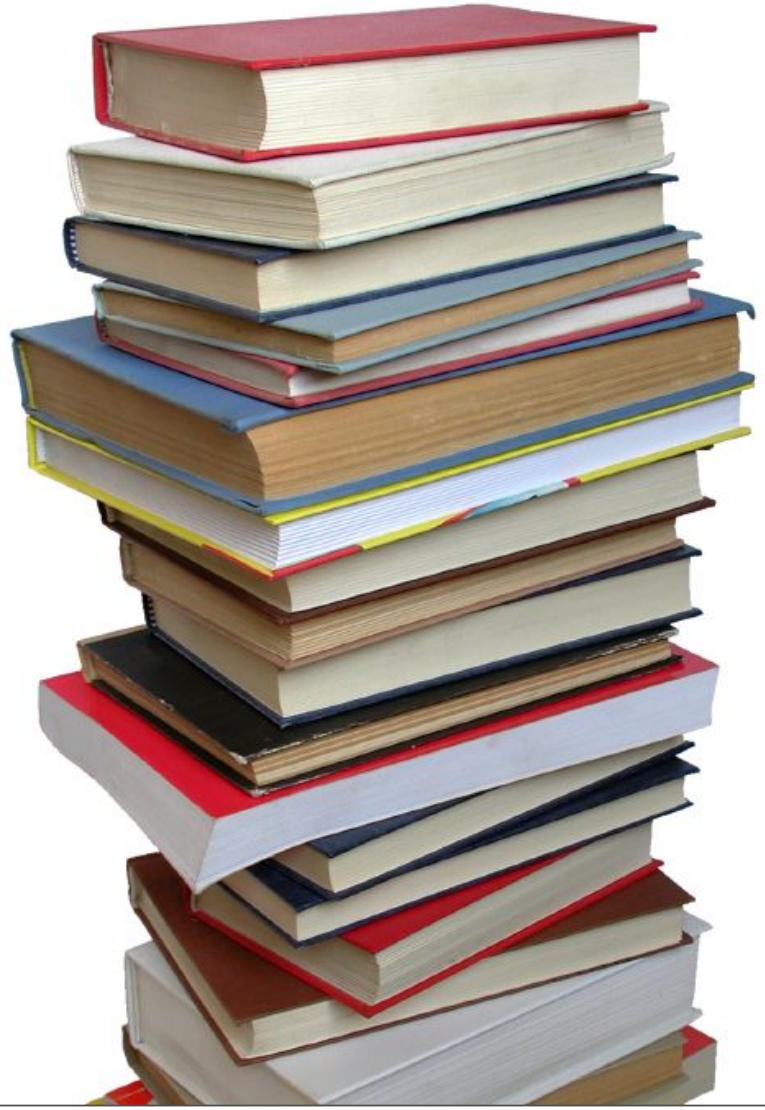
```
proc template;
define style styles.beachball;
   style data /
      background = red
      foreground = white
   ;
   style header /
      background = yellow
      foreground = purple
      font_face = Times
   ;
end;
run;
```

Copyright © 2006, SAS Institute Inc. All rights reserved.

- Sample style definition
- How do I know what names to use? That's coming up...

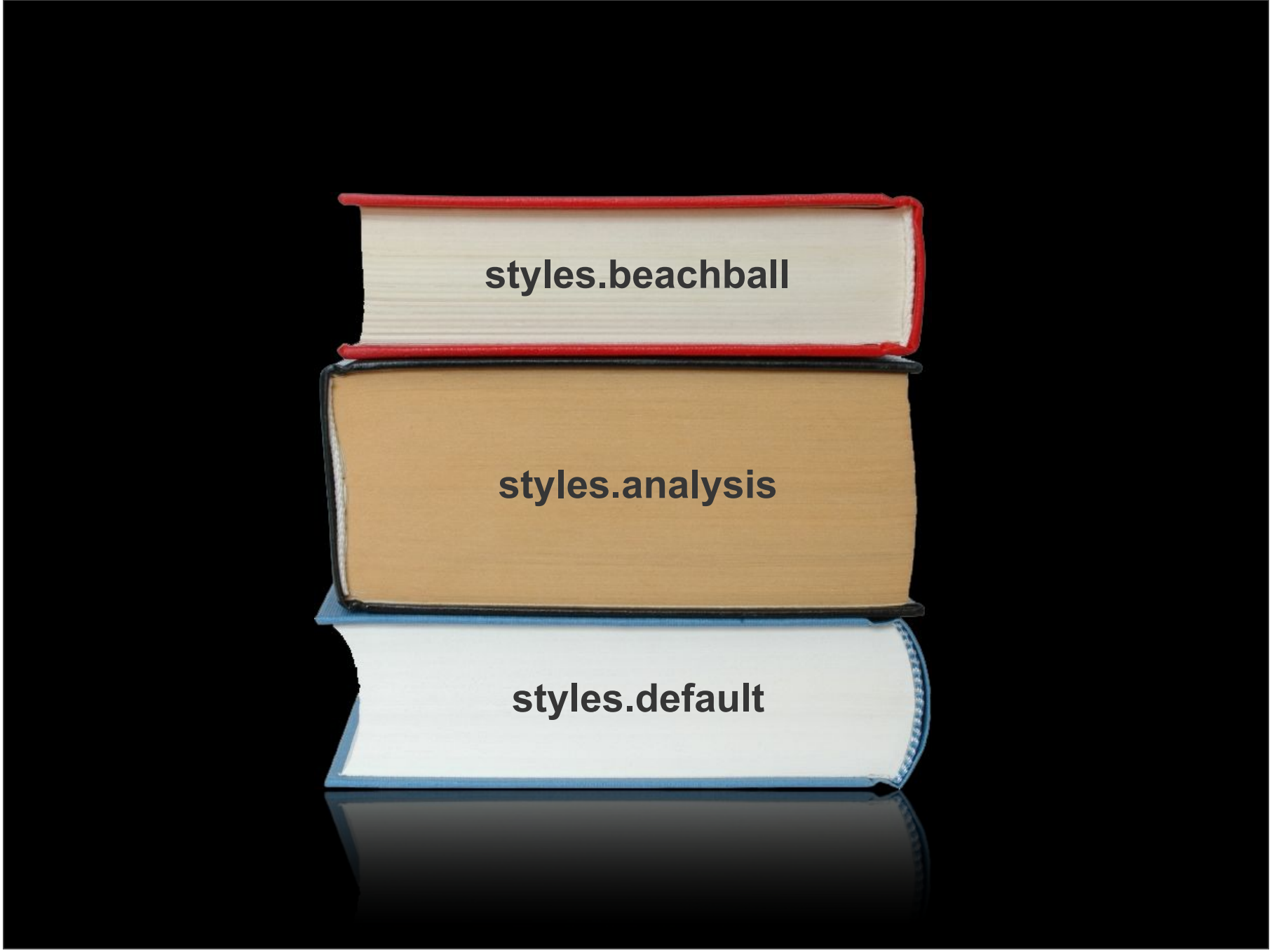
Style Definition Inheritance

```
define style styles.beachball;
   parent = styles.analysis;
   style data / ... ;
   style header / ... ;
end;
```



Copyright © 2006, SAS Institute Inc. All rights reserved.

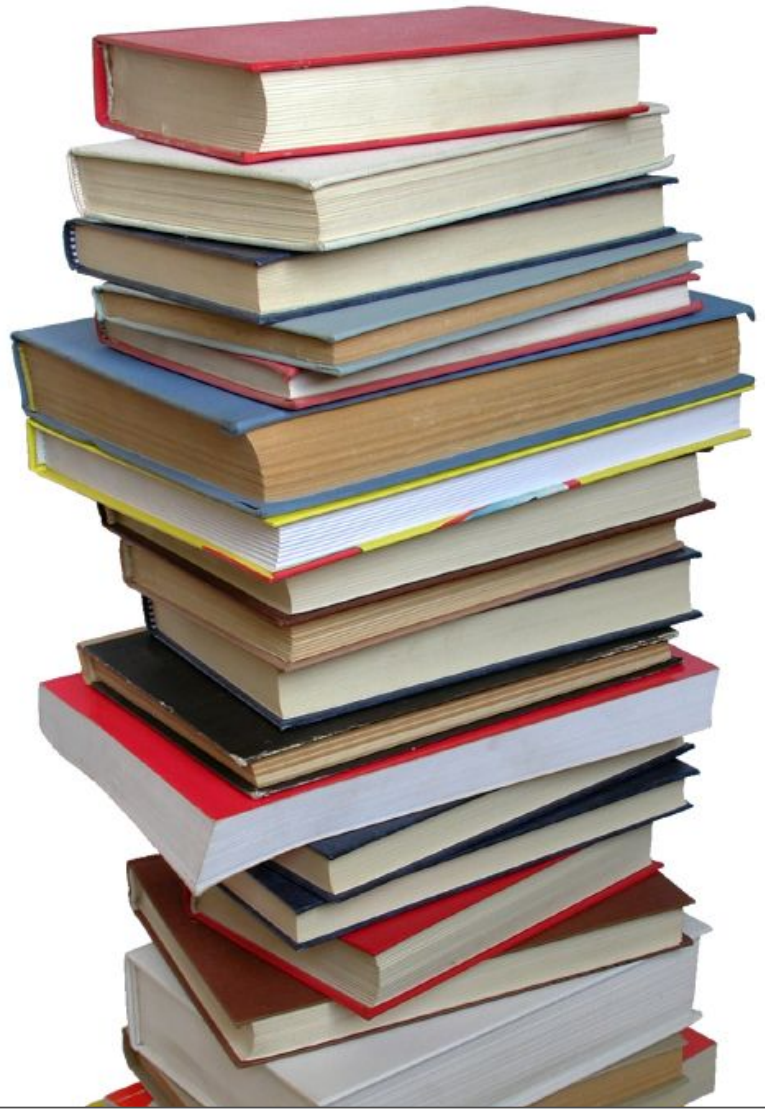
- Definition inheritance is like a stack of dictionaries where each dictionary is missing words
  - Look in the top dictionary
  - Continue popping off the stack until the element is found



- Simple example

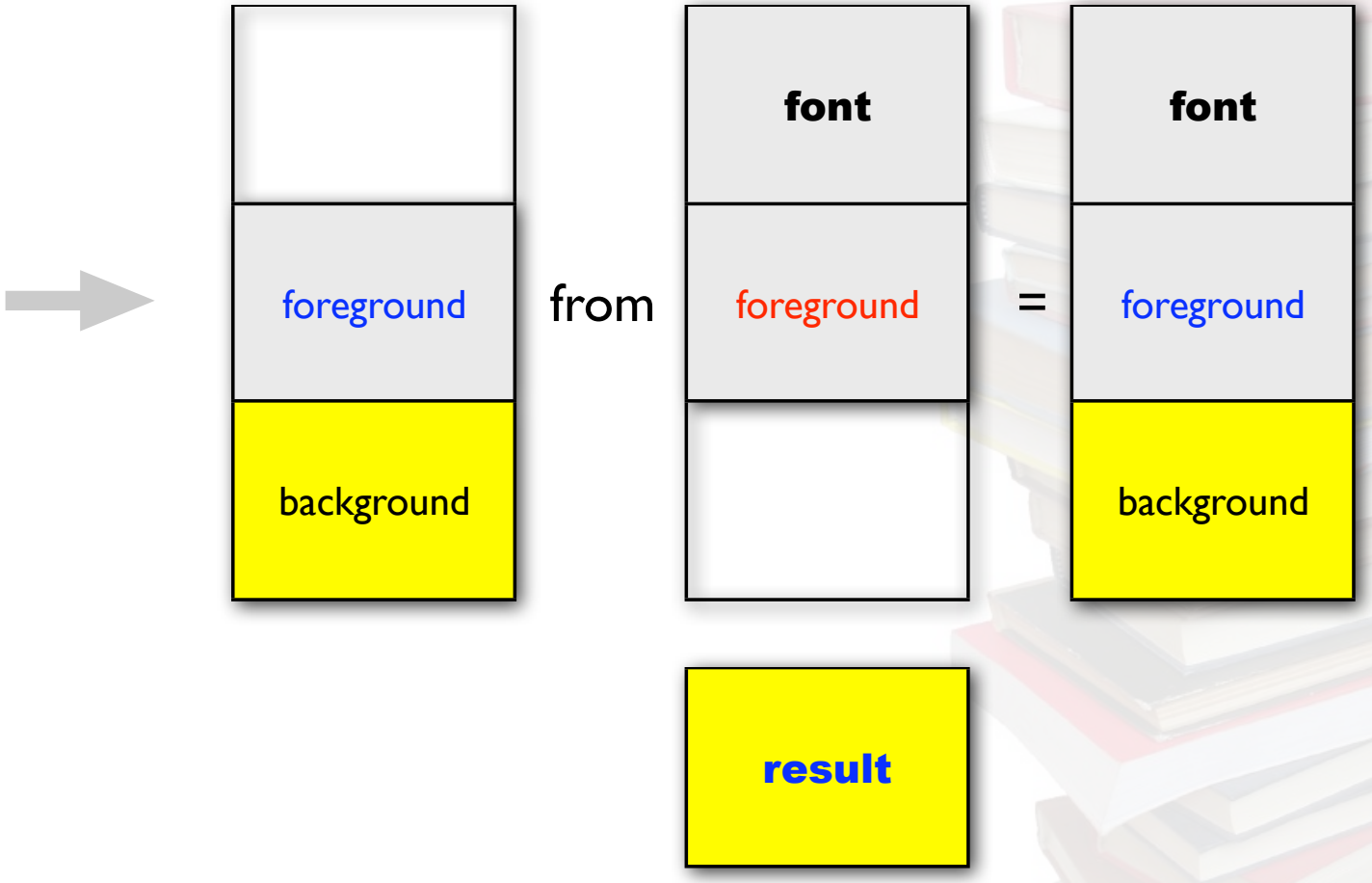
Style Element Inheritance

```
define style styles.beachball;
    parent = styles.default;
    style data from data /
        ...
    ;
    replace header from
        headersandfooters /
        ...
    ;
end;
```

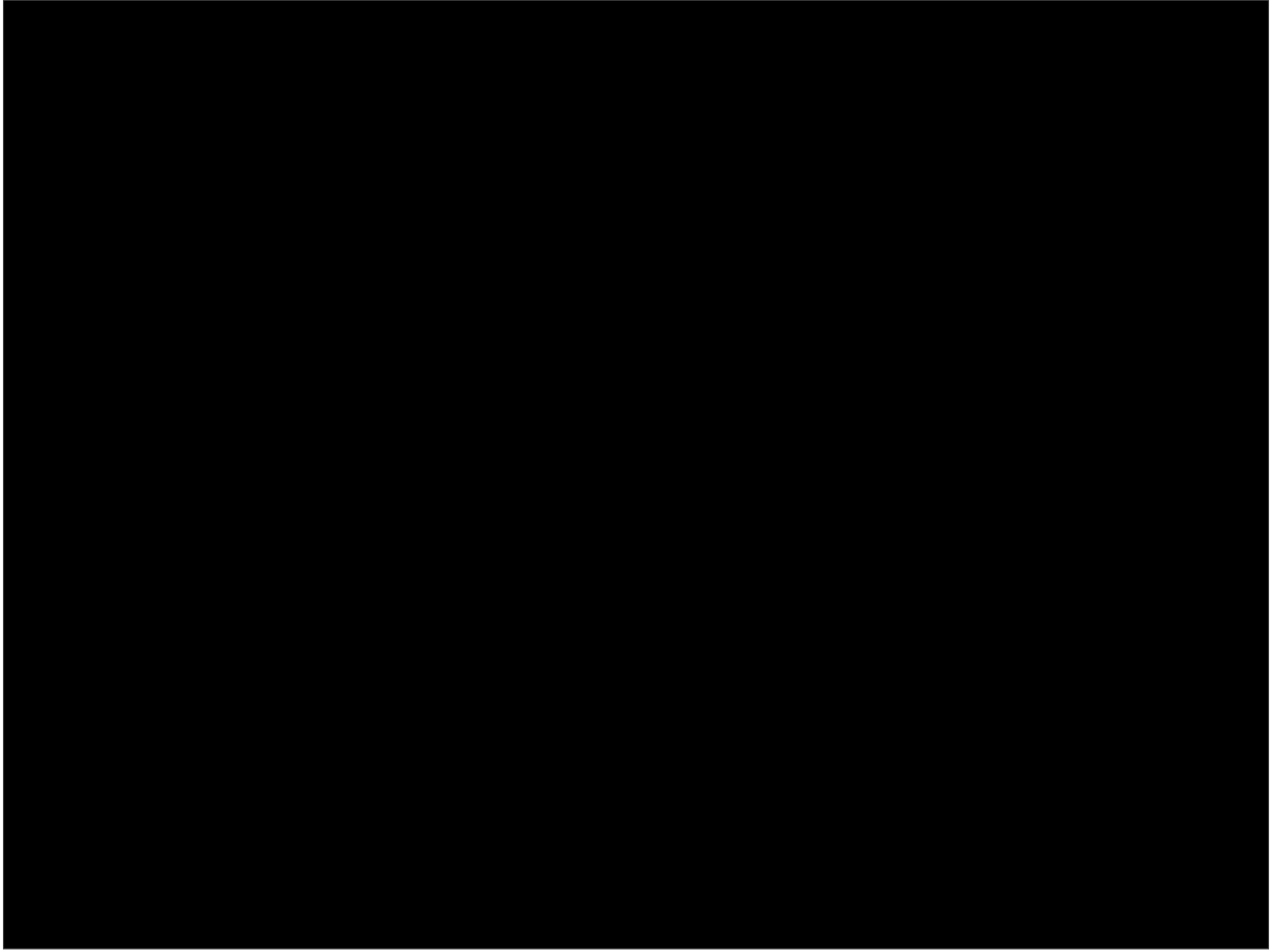


Copyright © 2006, SAS Institute Inc. All rights reserved.

- Style element inheritance is like a “see also”
  - Once you find the element, look for the “from” element in the same dictionary!
- Replace allows you to insert an element in a dictionary higher in the stack
  - Needed for things like the “colors” and “fonts” elements (if inheriting from styles.default)

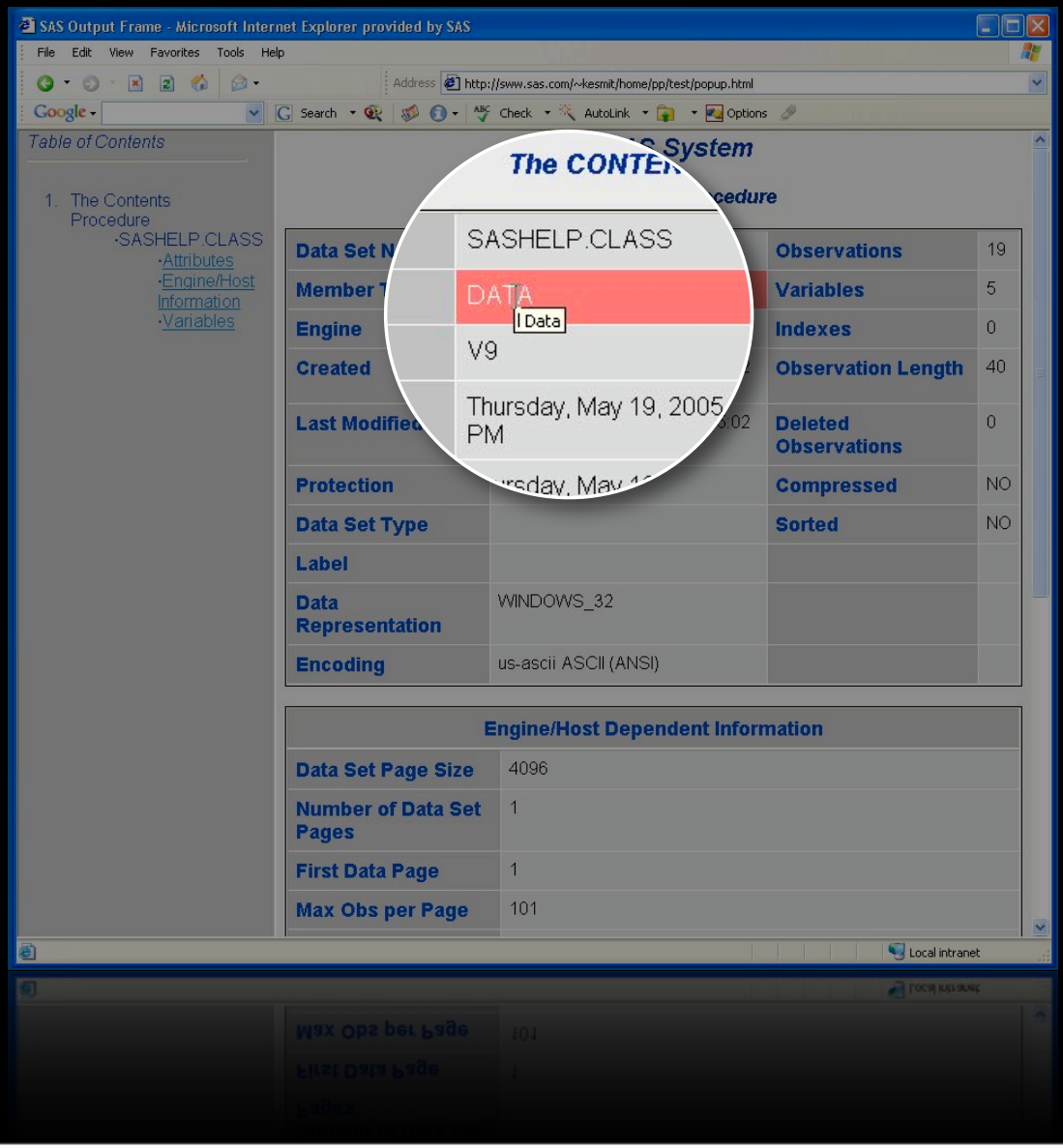


Copyright © 2006, SAS Institute Inc. All rights reserved.

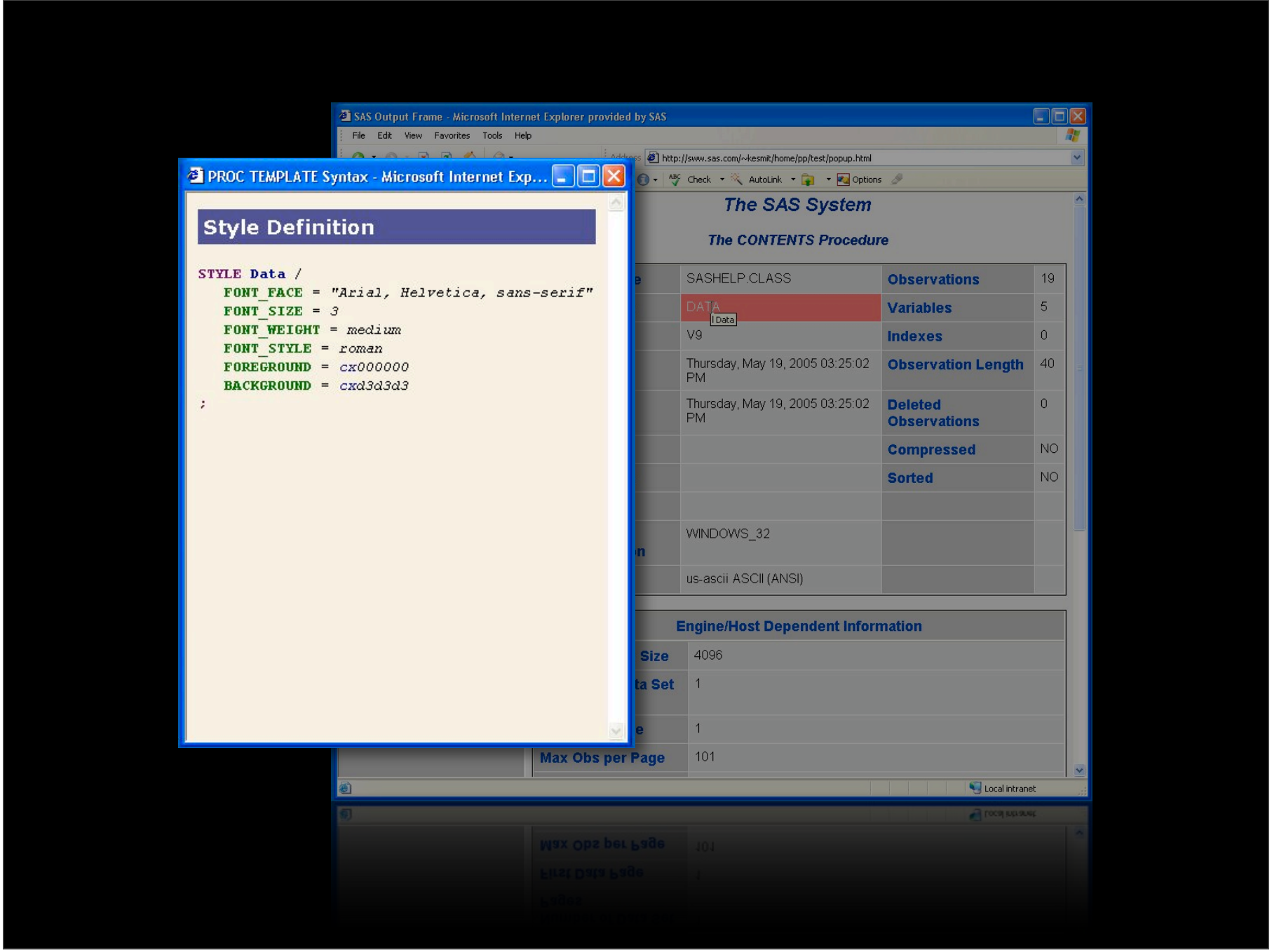


- How do we determine which style elements apply to which areas of a report?

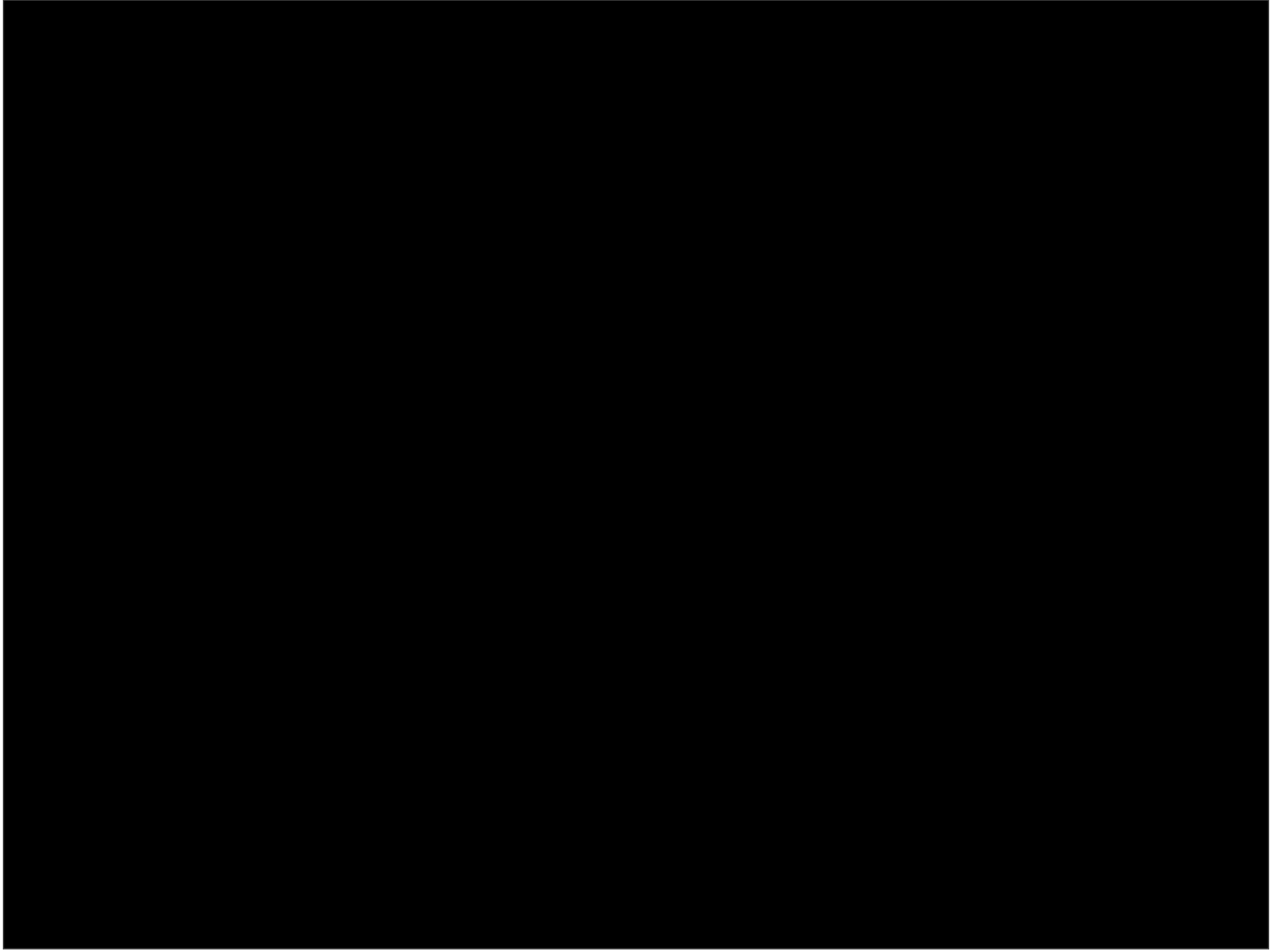
ods **tagsets.style_popup** file="test.html" ... ;



- Tooltips give style element
- Click to see the Proc Template syntax



- Not actual definition, extracted from CSS
- Close enough...



- Defining a complete style

You Are Here

- Breaking the rules

Usual Method

```
proc template;
define style styles.beachball;
    parent = styles.default;
    replace fonts /
        ...
    ;
    replace colors /
        ...
    ;
    style data from cell /
        ...
    ;
end;
run;
```

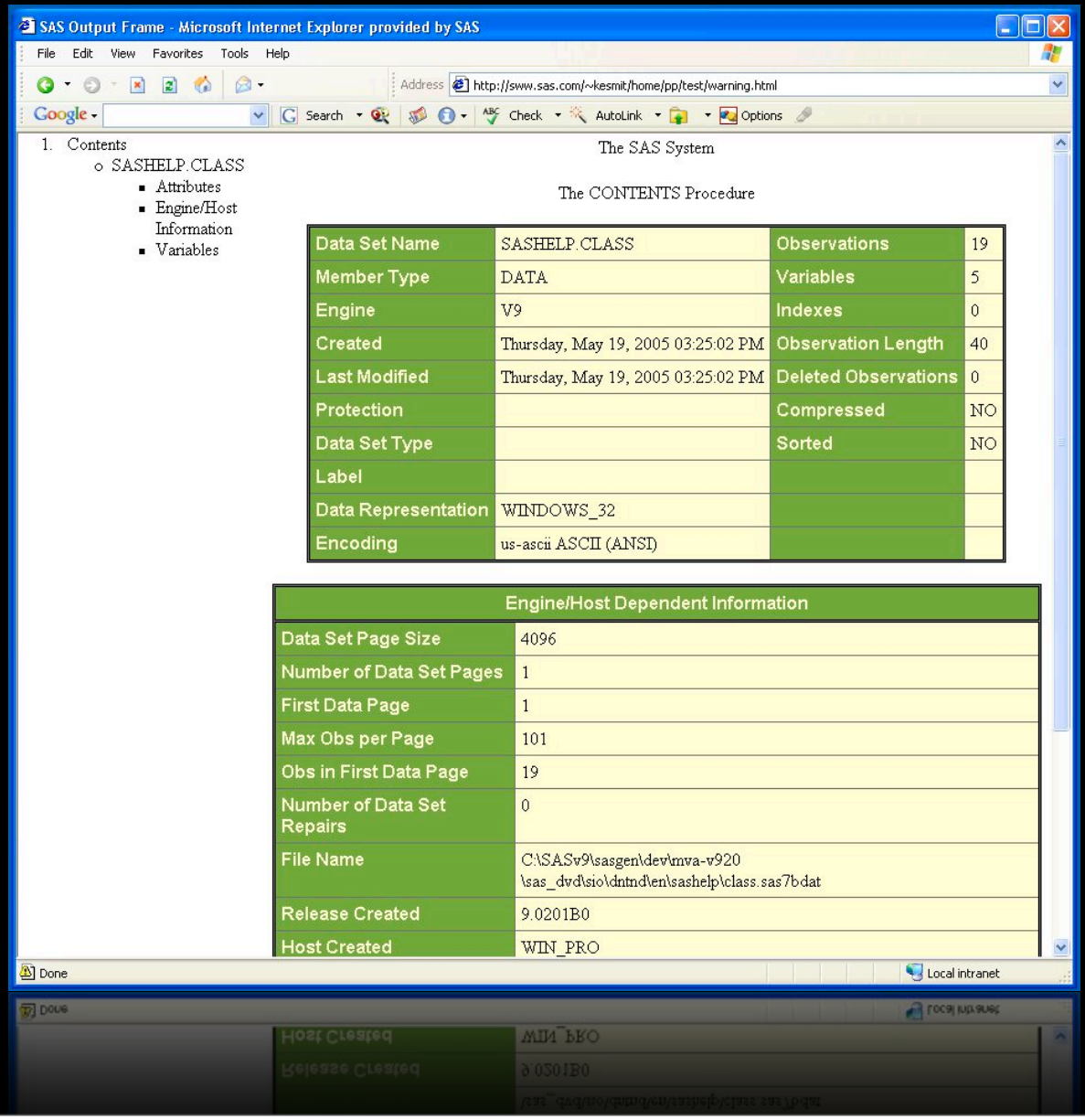
Copyright © 2006, SAS Institute Inc. All rights reserved.

- This method is difficult
  - definition inheritance
  - element inheritance
  - REPLACE statement
  - Have to know style element names
- Use tagsets.style_popup instead

```
proc template;
define style styles.python;

style data /
      font_face = Times
      foreground = black
      background = #ffffcc
;

style header /
      font_face = Arial
      foreground = #ffffcc
      background = #669933
;

style table /
      ...
;

end;
run;
```

Copyright © 2006, SAS Institute Inc. All rights reserved.

- Let's start fresh
- Just define a couple of elements, and work from there
- Build from the ground up



- Report with only a couple elements defined
- Easy to see result, no confusing inheritance

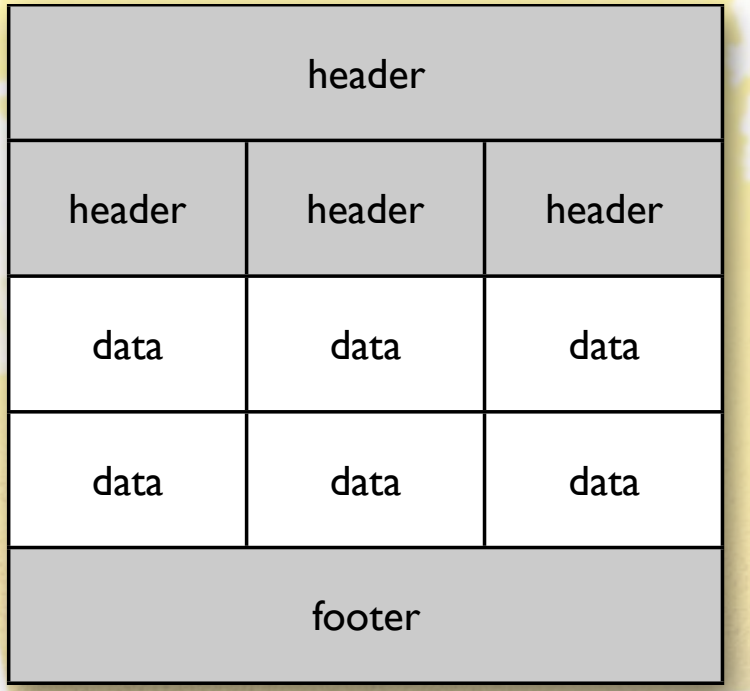
Reporting Procedures

- Proc Print
- Proc Report
- Proc Tabulate
- Proc Template
  Table Templates

Copyright © 2006, SAS Institute Inc. All rights reserved.

Proc Template Table Templates

```
proc template;
    define table survey;
        style = table;
        define header stitle;
            style = header;
        end;
        define column age;
            style = data;
        end;
        ...
    end;
run;
```



Copyright © 2006, SAS Institute Inc. All rights reserved.

- If a STYLE= is supplied, ODS uses a default

Proc Print Styles

```
proc print style(...) = table ...;
     var / style(...) = data;
     id / style(...) = rowheader;
     sum / style(...) = header;
run;
```

obsheader	header	header
obs	data	data
obs	data	data
total	total	total
obs	data	data
obs	data	data
total	total	total
grandtotal	grandtotal	grandtotal

Copyright © 2006, SAS Institute Inc. All rights reserved.

- Parentheses are filled with one or more names from the table
- OBS column is also ID column
- STYLE(...) on statements have tighter scope than on Proc Print statement

Proc Report Styles

```
proc report style(...) = table ...;
    define / style(...) = header;
    break / style(...) = rowheader;
run;
```

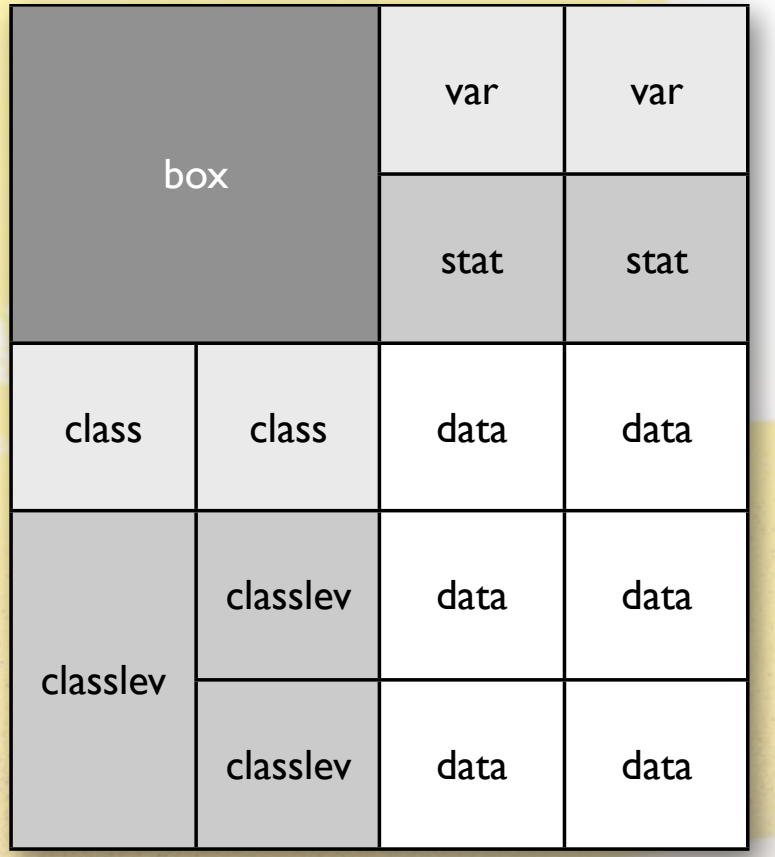
header	header	header
column	column	column
column	column	column
summary	summary	summary
lines		
column	column	column
column	column	column
summary	summary	summary
lines		
lines		

Copyright © 2006, SAS Institute Inc. All rights reserved.

- The 'report' location refers to the table itself
- STYLE(...) on statements have tighter scope than on Proc Report statement

Proc Tabulate Styles

```
proc tabulate style = data;
     class / style = header;
     classlev / style = rowheader;
     var / style = rowheader;
     keyword / style = rowheader;
     table / style = table
          box = {style = rowheader};
run;
```



Copyright © 2006, SAS Institute Inc. All rights reserved.

- Tabulate statement sets data cells
- Keyword sets headings
- Table statement sets entire table

– Evolution and Revolution

- Syntactic sugar features

Replace Statement

v9.2

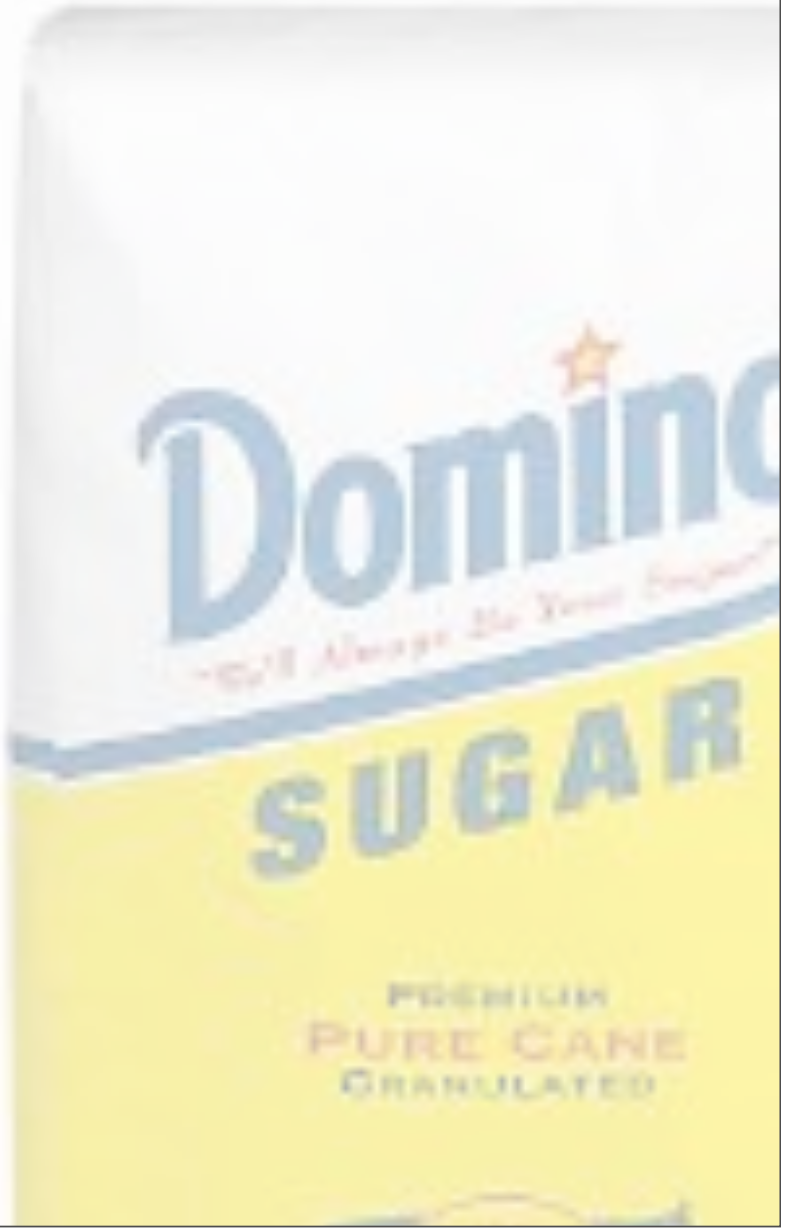
Copyright © 2006, SAS Institute Inc. All rights reserved.

- Replace is dead!
- New element inheritance model obsoletes it
- Backwards compatibility
  - always use ‘from’ (e.g., style fonts from fonts)

Cascading Inheritance

```
define style style1;
    style data / foreground = blue;
    style data / background = white;
end;
```

```
define style style2;
    style data /
        foreground = blue
        background = white
    ;
end;
```



v9.2

Copyright © 2006, SAS Institute Inc. All rights reserved.

- Can use same name more than once, attributes get aggregated
- Why is this useful? See the next slide (simultaneous elements)

Simultaneous Style Elements

```
style data, dataempty, datafixed /
    foreground = blue
    background = white
;

style datafixed /
    fontfamily = “Courier New”
;
```

```
style data, dataempty from _self_ / ... ;
```

v9.2

Copyright © 2006, SAS Institute Inc. All rights reserved.

- Replacement for explicit inheritance
- Use ‘from _self_’ to inherit from same name, or use the CLASS statement...

Class Statement

```
class data, dataempty, datafixed
      / ... ;
```

```
style data, dataempty, datafixed
      from _self_ / ... ;
```

v9.2

Copyright © 2006, SAS Institute Inc. All rights reserved.

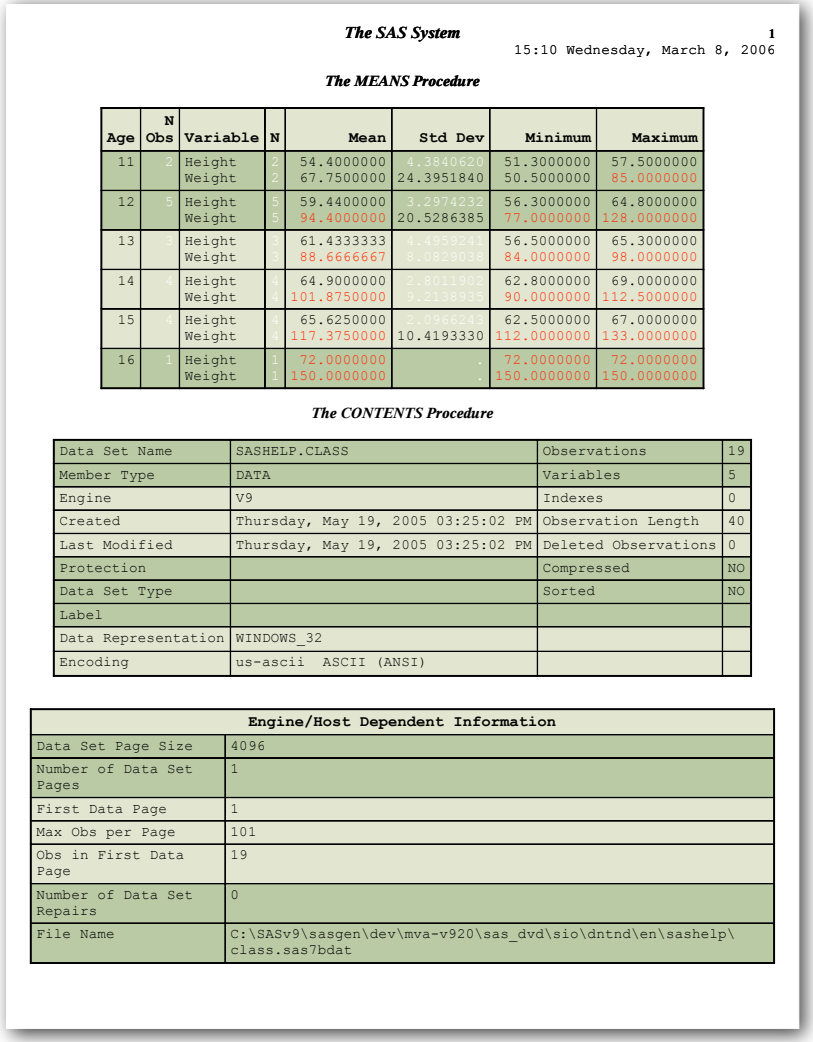
- Easier way of saying 'style foo from _self_'

Style Overrides

Age	N Obs	Variable	N	Mean	Std Dev	Minimum	Maximum
11	2	Height	2	54.4000000	4.3840620	51.3000000	57.5000000
		Weight	2	67.7500000	24.3951840	50.5000000	85.0000000
12	5	Height	5	59.4400000	3.2974232	56.3000000	64.8000000
		Weight	5	94.4000000	20.5286385	77.0000000	128.0000000
13	3	Height	3	61.4333333	4.4959241	56.5000000	65.3000000
		Weight	3	88.6666667	8.0829038	84.0000000	98.0000000
14	4	Height	4	64.9000000	2.8011902	62.8000000	69.0000000
		Weight	4	101.8750000	9.2138935	90.0000000	112.5000000
15	4	Height	4	65.6250000	2.0966243	62.5000000	67.0000000
		Weight	4	117.3750000	10.4193330	112.0000000	133.0000000
16	1	Height	1	72.0000000	.	72.0000000	72.0000000
		Weight	1	150.0000000	.	150.0000000	150.0000000

v9.2

- STYLE= now merges with CELLSTYLE-AS
- New variables: _style_, _dataname_, _col_, _row_, _datatype_
- Striped tables are now easier because of new merge rules



The SAS System 1
15:10 Wednesday, March 8, 2006

The MEANS Procedure

Age	N Obs	Variable	N	Mean	Std Dev	Minimum	Maximum
11	2	Height	2	54.4000000	4.3840620	51.3000000	57.5000000
		Weight	2	67.7500000	24.3951840	50.5000000	85.0000000
12	5	Height	5	59.4400000	3.2974232	56.3000000	64.8000000
		Weight	5	94.4000000	20.5286385	77.0000000	128.0000000
13	3	Height	3	61.4333333	4.4955241	56.5000000	65.3000000
		Weight	3	88.6666667	8.0829037	84.0000000	98.0000000
14	4	Height	4	64.9000000	2.8011901	62.8000000	69.0000000
		Weight	4	101.8750000	9.2138931	90.0000000	112.5000000
15	4	Height	4	65.6250000	2.0966241	62.5000000	67.0000000
		Weight	4	117.3750000	10.4193330	112.0000000	133.0000000
16	1	Height	1	72.0000000	.	72.0000000	72.0000000
		Weight	1	150.0000000	.	150.0000000	150.0000000

The CONTENTS Procedure

Data Set Name	SASHELP.CLASS	Observations	19
Member Type	DATA	Variables	5
Engine	V9	Indexes	0
Created	Thursday, May 19, 2005 03:25:02 PM	Observation Length	40
Last Modified	Thursday, May 19, 2005 03:25:02 PM	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label			
Data Representation	WINDOWS_32		
Encoding	us-ascii ASCII (ANSI)		

Engine/Host Dependent Information	
Data Set Page Size	4096
Number of Data Set Pages	1
First Data Page	1
Max Obs per Page	101
Obs in First Data Page	19
Number of Data Set Repairs	0
File Name	C:\SASv9\sasgen\dev\mva-v920\sas_dvd\sio\dntnd\en\sashelp\class.sas7bdat

Implicit Parents

```
Base.Template.Style
Base.Template.Table
Base.Template.Column
Base.Template.Header
Base.Template.Footer
```

v9.2

Copyright © 2006, SAS Institute Inc. All rights reserved.

- Changes all tables in a report with one template
- Stripe all tables in a report easily
- Doesn't work for report, tabulate, print, freq
- Base.Template.Style now includes all inheritance

- CSS is a standard syntax
- Use third-party tools
- Must still obey ODS rules for writing CSS files (i.e., style elements = CSS classes)

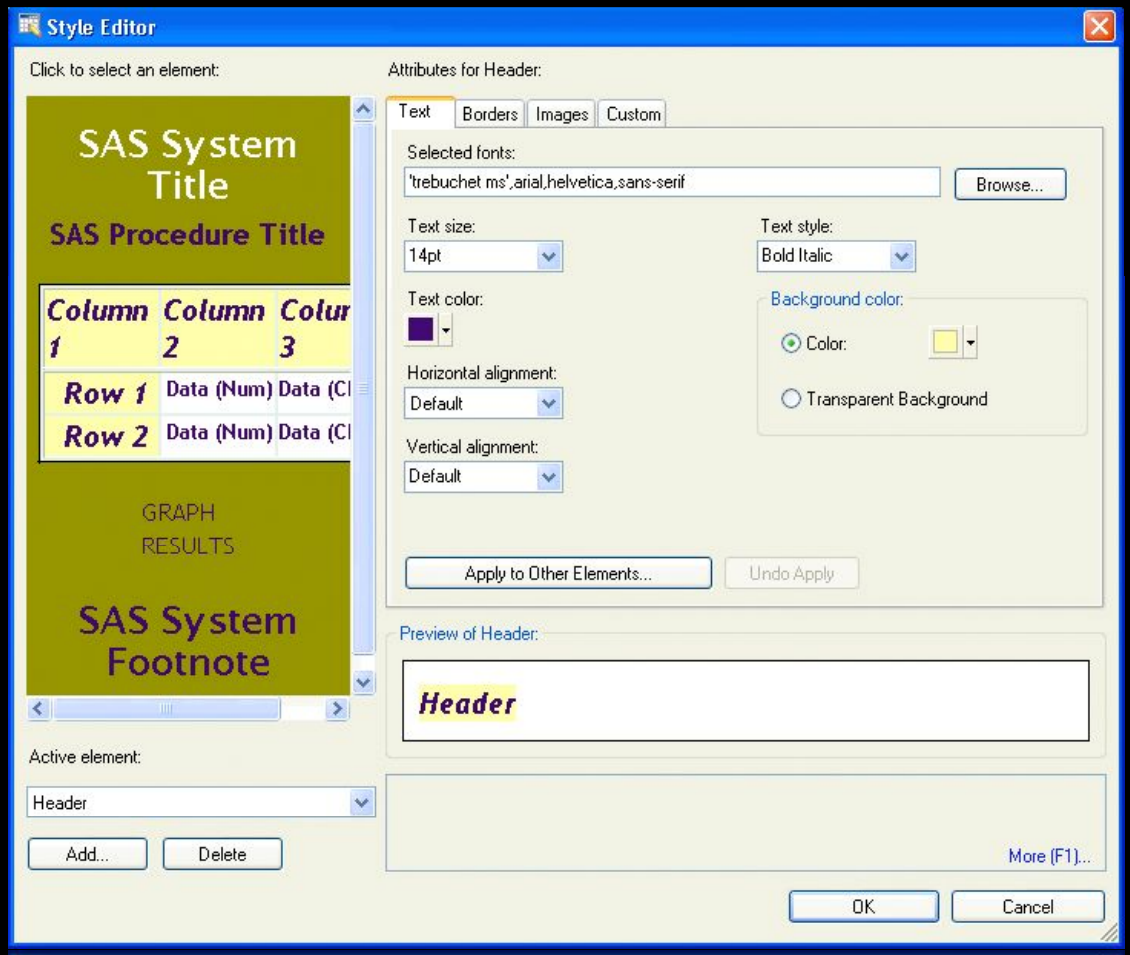
```
define style my-style;
    import my-css-file;
end;
```

```
ods destination cssstyle=my-css-file;
```

v9.2

Copyright © 2006, SAS Institute Inc. All rights reserved.

- Convert a CSS file to Proc Template style definition
  - Import multiple files
  - Imports files, filerefs, and URLs
- Use a CSS file directly with CSSSTYLE=



v9.2

- Enterprise Guide Style Editor
  - Saves CSS files
- Included with Base/SAS

- With a little bit of work, it's possible to turn those layers into a treat

Resources

- Kevin D. Smith - Kevin.Smith@sas.com
- ods@sas.com
- http://support.sas.com/documentation/onlinedoc/

Copyright © 2006, SAS Institute Inc. All rights reserved.