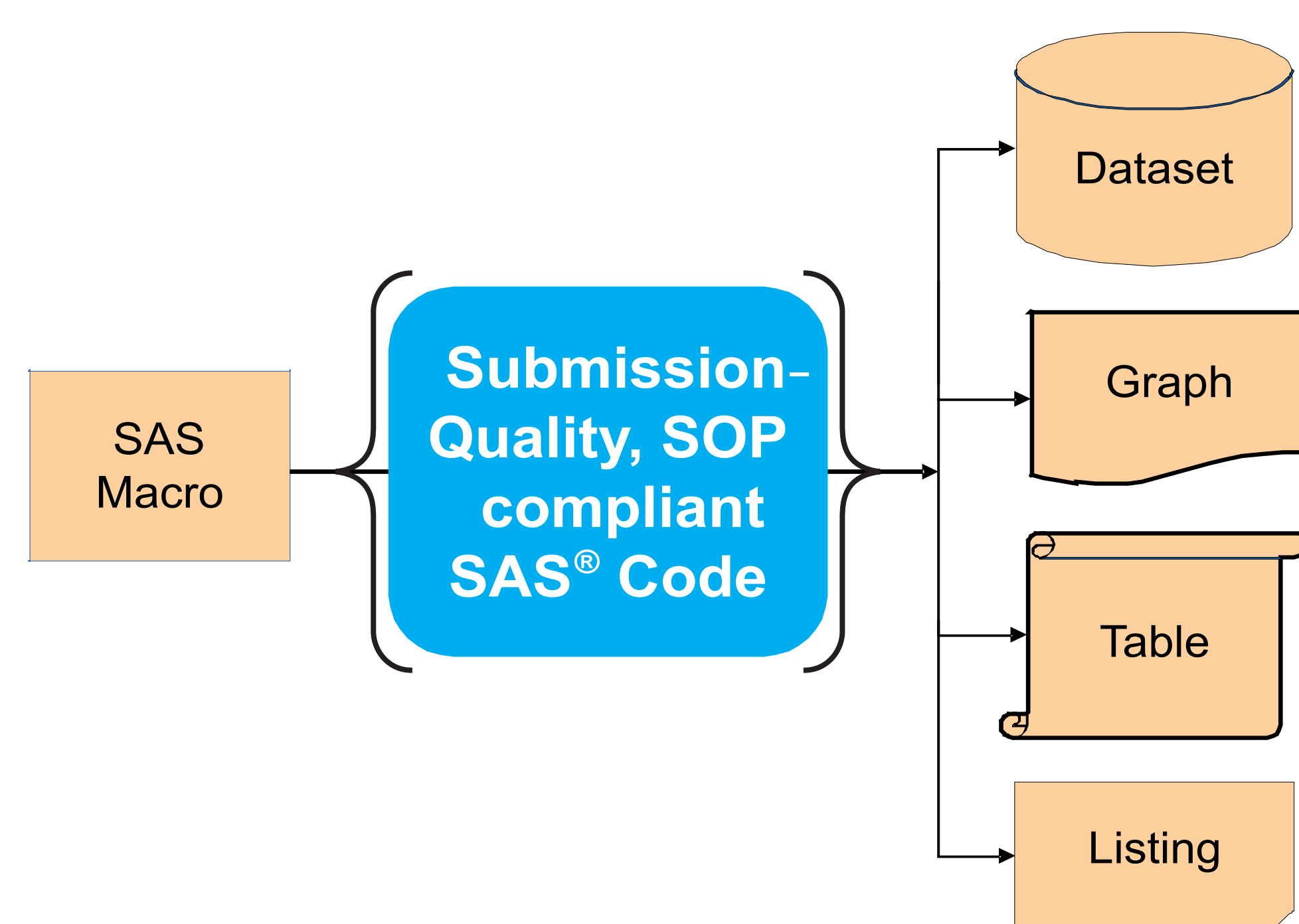


Metaprogramming in SAS

Arvind Chaudhari, Cytel Statistical Software & Services Pvt. Ltd., Pune, India

Parameterized **Macro** to **generate SAS Code** (using **MPRINT** and **MFILE** and other techniques)
Generated SAS code in turn produces outputs such as Safety TLFs.

Solution :



Code Generation

SAS Macro

```
%inc "E:\Project\test_macro.sas" ;
```

Options Mprint Mfile

```
options mprint mfile;
Filename mprint "E:\Project\test_code.sas";
```

SAS Macro Call

```
%Report_Macro (Parameter1= X,
                Parameter2= Y,
                Parameter3= Z ) ;
options nomprint nomfile;
```

Submission Ready Code

Include Setup Macro

```
data _null_ ;
file mprint ;
put " %inc 'E:\setup.sas' ; " ;
put " %nrquote(%)setup (Path=E:\ ) ; " ;
run;
```

Comments

Commenting style used in the Macro	Result
*Comment;	Passed through as it is
*Comment1 *Comment2;	Text will flow to long lines: *Comment1 *Comment2;
/*Comment*/	Ignored
%"Comment";	Ignored

Indentation

```
%indentcode (
  programpath = &FolderPath ,
  programname = &Filename ,
  filepath = &FolderPath.\keywords.csv,
  space = 2 ,
  linelen = 90 ) ;
```

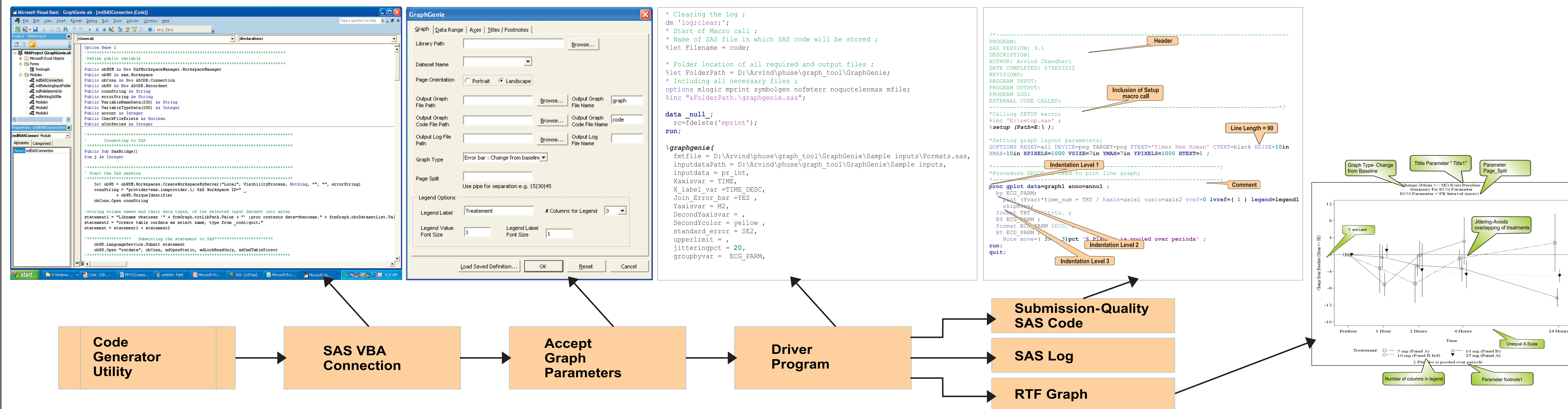
Header

```
/*Printing header in the output sas file*/
data _null_;
file "&programpath.\&programname..sas" ;
put "/*-----*/";
put "PROGRAM:";
put "SAS VERSION: &sysver.";
put "DESCRIPTION:";
put "AUTHOR: &author.";
put "DATE COMPLETED: &sysdate9.";
put "REVISIONS:";
put "PROGRAM INPUT:";
put "PROGRAM OUTPUT:";
put "PROGRAM LOG:";
put "EXTERNAL CODE CALLED:";
put " ";
put "PROGRAM ALGORITHM:";
put " ";
put " ";
put " ";
put "-----*/";
run;
```

SAS Log

```
proc printto
log="&OutFilePathLocation.\&Filename..log" new ;
run;
%inc mprint;
proc printto;
run;
```

Illustration :



Conclusion : Submission ready SAS code can be generated from Macro, instead of hand-programmed.
Advantages : productivity↑ consistency↑ accuracy↑ skill dependency↓