HOW TO USE COMPARE
AUGUST 2011

INTERINDUSTRY FORECASTING PROJECT

ECONOMICS DEPARTMENT

UNIVERSITY OF MARYLAND

COLLEGE PARK, MD 20742

(301) 405-4609

www.inforum.umd.edu

Introduction

The *Compare* program is a general table making program that can be used to make tables from various source Inforum data bank formats. As the name suggests, it particularly is adapted to comparing results of several data banks, but it can also list the contents of a single data bank or the results of a single run of a model. When being used to show a base case and several alternatives, it can show the alternatives as actual values, or as deviations from the base, or as percentage deviations from the base. The results are written to a file that can be printed, viewed as text, viewed in a spreadsheet program, or read by other software programs. The INFORUM databank formats that *Compare* supports include normal *G* databanks ("workspace banks"), *G* compressed banks, *G* hashed banks, Interdyme Vam files, and LIFT/SLIMFORP Dirfor files. Up to 10 banks of different types can be compared using *Compare*.

Compare contains several features for working with the Interdyme system for building input-output models. For example, *Compare* can do matrix listings for an Interdyme model, showing intermediate and final demand flows that comprise the total output for each industry. Matrix listings can be done for either buyers or sellers, and in either flow or coefficient form. The dates for a table or a section of a table should be specified in a \dates command. If no \dates command is given, dates will be requested from the user at runtime. Using the \dates command the same table can have different dates in its various sections. Also, multi-period growth rates can be displayed.

This document will first describe the construction of the stub file, which is the key to specifying the contents of a table, and how they will be displayed. Next, the operation of the program is reviewed. Finally, special uses for Compare are described. These cases include making a matrix listing for Interdyme using *Compare*, and how to make tables using Dirfor files.

I.   *Preparing the Stub File*

To use *Compare*, one first prepares a "stub" file, which is a text file that provides a description of the series to be printed, the format for printing, titles, dates and other information. Our stub files usually have the extension ".stb", but this is not necessary.

The following is a sample of a basic stub file. All commands in the sample, and many others, will be explained in this section.

```
\dates 2007.0 2008.0 2007.1 2007.2 2007.3 2007.4 2008.1 2008.2 2008.3 2008.4
;
;                      THE AMI MODEL
;
&
;
gnp$       ;Gross national product
c$         ;Personal consumption
pibg$      ;Pers. income w/o gov't
pidis$     ;Personal disposable income
taxrate*100 ;taxrate*100 *
gtnis$     ;Gov trans, net int, sub.
cpc$       ;consumption per capita
pop*1000 ;population (in millions) *
100*(gnp$-gnp$[1])/gnp$[1] ;GNP growth rate
```

Choosing a File Type

Compare can print tables in spreadsheets, word processor files, or as plain text files. Some formatting commands are unique to a particular file type.   Enter one of the following commands near the top of a stub file to instruct Compare on what type of file to create. If a file type is not specified, Compare will create an output file with special codes that are interpreted by the printer.

\xls

> This option makes all output go to a Microsoft Excel spreadsheet file. The default file extension and file type is .XLS, though other options are available.   The Microsoft Excel program must be installed for this feature to work. Supported versions include Excel 2000, 2003, and 2007. Unlike the following \wk1 feature, printing to multiple worksheets is supported, with new sheets created as with other new pages. See the \wstitle command for specifying worksheet names. Special formatting commands for Excel spreadsheets are presented in the "Formatting Commands for Excel Documents" section.

\wk1

> This option makes all output go directly to a Lotus .WK1 file. Note that no matter what you give for the name of the output file, it will have the file extension .WK1. You still are limited to the maximum rows and columns (4096 x 256) when using a .WK1 file. To create larger spreadsheets and modern formats, use the \xls option.

\prn   **<y|n>**

This option sets printing to a .PRN file, in other words, with all text in quotes. A .PRN file can be opened by any text editor or Microsoft Excel. It can also be imported into Lotus with the /fin command ({File}{Import}{Numbers}).

\rtf

Using this command will instruct Compare to create a Rich Text Format (.RTF) file. This file type can be opened with Microsoft Word, OpenOffice Writer and most other document editing programs. Support for this file type is limited.

\gdata [<ObsPerLine>]

This command is used to generate output as a G "data card" file. In other words, for all series names or expressions in the file, output is written in a format usable as input for G. The optional parameter <ObsPerLine> specifies how many observations will be written on each line. Note that the "\field" and "\decs" commands also apply to determining the field width and number of decimal points printed. By default, series will be printed to G data cards at 5 observations per line, and the current field width and decs. Also, titles, subtitles, comment lines (with a ';' and text, but no series) will be printed to the file as G-style comments (i.e., lines starting with a '#'). If you want to shorten the file, and include only data cards, use the "\nocomment" option. Note that series that are all zeroes will not be printed.

\oneperline

This is a command, like "\gdata" which specifies that output is not a table, but rather a data dump. This command prints out series to the output file one data observation per line, with whatever is specified on the title field to the left of the data. Formatting is controlled by current width, decimals and line title width settings. Often this command could be used to print the data with series codes in the line title, the codes passed as arguments to "\fadd" commands.

The Most Commonly Used Commands

*   Go to the top of a new page and print the titles of the alternatives runs as given in the ".fix" file. In Excel spreadsheets, a new worksheet will be created.

* m   If there is not room for m more items on the page, go to a new page and print the dates across the top.

* m n If there is not room on the page for m more items plus n lines, go to a new page and print the names of the base and alternative runs.

;   Print the line just as it stands.

&   Print the dates across the page above the appropriate columns. Note that this has changed from the '@' used in older versions! This is so that the '@' can be used for function names.

Any line beginning with this character as the first non-blank character will be treated as a comment. This allows you to selectively remove portions of a table by commenting them out, and then including them later by removing the comment characters.

\dates Provide the dates for the run. Use either 2-digit (deprecated) or 4-digit (prefered) digit dates. Compare can understand either. To specify the number of digits in which dates are printed, see the \yearformat command below. Annual dates are printed as integers, or with a '.0'. Quarterly dates are printed with 1 decimal point, and monthly dates with 3 decimal points. Examples: 92 or 1992 means 1992 at an annual rate. 92.2 means 1992, 2nd quarter. 93.011 means November 1993.

Note that growth rates can also be specified with the \dates command. To request growth rates, enter two numbers separated with a dash, such as "92-95". Growth rates and levels can be mixed on the same display.

There is also a code that can be given in the \dates line of a stub file, which provides finer control over the column layout of the table. The "s" code is followed by an integer. The integer tells how many columns to skip before printing the next column, if the file format is text or printer mode, or it provides the column width if the \xls option is employed. This may be useful in quarterly or monthly printouts, where you want to clearly mark out where the year begins and ends. For example:

\dates 93.1 93.2 93.3 93.4 s4 94.1 94.2 94.3 94.4 s4 95.1 95.2

Other uses of the "\dates" command are for calculating sums or averages of variables over a number of periods. If you specify the column heading "95+97", then Compare will calculate the sum over the periods from 1995 to 1997, and print it in that column. If you specify a column heading such as "95|97", then

Compare will calculate the average over that interval and print it in that column. Finally, you should note that a shorthand version for specifying a continuous interval of dates exists. For example, if you want to make a large table with all series from 1950 to 1997 quarterly, you could specify your dates as:

\dates 50.1:97.4

\ti <title>

This command can be used to specify a title, which will be centered on each subsequent page. It is put as many lines down as specified by the top margin

\head <header>

This is different than a title. The header is put at the top left margin, on the first line of each page. It is directly to the left of the page number. This is useful as the header for an entire set of tables, where each table has its own title given by the \ti command. Note: if the <header> is the word "title", the title of the first bank will be used as the header.

\center <text to center>

This command inserts centered text into the table. The centering is done not with respect to the current page width, but rather to the "right end" of the table, which is defined as the end of the last column of data.

\line <linechar>

This command inserts a line of <linechar> characters, out to the "right end" of the table. (The right end is the end of the last column of data.) If no character is supplied, it defaults to the dash ('-') character.

\add <stubfilename> [<arg1>] [<arg2>] ...

This command allows you to include the contents of another stub file into the current stub file. This is convenient when you want to make a large table which is an agglomeration of other tables. For example, with a large model such as LIFT, you may keep separate special interest tables around, but then you may want to add them together for one large table. Note that command line arguments work just as in G, where arguments are replaced by "%1" for the first argument, "%2" for the second, etc.

\fadd <stubfile> <argfile>

This command is also patterned after (stolen from) the corresponding *G* command. The <stubfile> is a stub file with arguments, that could be used with the "\add" command. The <argfile> is a file containing a list of arguments. The <stubfile> is called once, for each line of the <argfile>, just as in *G*.

\mode

This command allows you to change the mode of comparison in selected areas of the table. "\mode a" means that data are to be compared as levels ("actuals"). "\mode d" means that data are to be compared as first differences. "\mode p" means that data are to be compared in terms of percentage differences from the

base. Note that this same information is requested from the user in the interactive running of the *Compare* program. The \mode command in the stub file overrides this default mode of comparison.

\printbase <true | false>

Set \printbase to "true" (or "1" or "on" or "yes") to print the baseline levels. Set \printbase to "false" (or "0" or "off" or "no") to print only deviations from baseline levels. Deviations may be presented in differences or percentage differences, and this is controlled with the \mode command.

\gt   (or \growthtype)

This command is only relevant if you have multi-period growth rates, such as 95-98. The \gt command can take 3 possible arguments. "\gt e" specifies exponential growth, "\gt c" specifies a compound growth rate, and "\gt p" specifies percentage growth. The relevant formulas used are displayed in the table below:

Growth Type	Formula
e	$g = 100 * \log(x[T]/x[0])/T$
p	$g = 100 * ((x[T]/x[0])-1/T)$
c	$g = 100 * \{\exp[\log(x[T]/x[0])/T]-1\}$

where:

g is the growth rate
$x[T]$ is the ending period data point
$x[0]$ is the starting period data point
T is the number of periods
$exp()$ and $log()$ are the exponential and natural logarithm functions

Note that the default growth rate type is exponential ('e'), if none is specified.

\g

Show the growth rates in percent per year for the variables named on the following lines. Annual dates with a quarterly file will give growth of one year over the previous year. Quarterly dates with quarterly data will give quarter to quarter growth and this will be at an annual rate if the \ar command is in effect. Be careful not to use this command if you already have growth rates (such as "92-95") specified in your \dates command.

\n   Cancels a previous \ g.

\div <number>

This divides every table entry in levels following this command by the specified factor. This is helpful if you would like to convert an entire table from millions to billions, for example. To turn off the effect of this command, merely issue the command "\div 1".

Commands That Control The Table Format

\ fw dp pl tm bm tw   (the general formatting command)

> where fw dp pl tm bm and tw are all numbers. This is an optional line to set the field width of each printed number to fw, the number of decimal places to be printed to dp, the page length to pl lines, the top margin to tm lines, the bottom margin to bm lines, and the width of the titles to tw. The last three may be omitted. The default values are 7 1 60 3 9 32, which will be in effect if this command is not given. Note that the title width will establish the column width for the first row of a spreadsheet when the \xls option is employed.

\decs <number>

> This command changes only the number of decimal places printed, and so is simpler than the general formatting command, described above. Note that this command overrides the second argument of the general formatting command. Note also that this setting controls the precision when the \xls option is employed.

\gd (or \growthdecs) <number>

> Using this command, you can override the number of decimal points specified in the \fw etc., command, so that growth rates have their own number of decimal points displayed. For example, you may want level variables to have one decimal point, but growth rates to have 3. In this case, issue the command "\gd 3".

\growthwidth (gw) <number>

> This command enables you to have a different width for the growth rate columns than for the other columns. Note that the width has to be at least as big as the date expression ("00-09") plus 1, so that 6 would be the smallest width allowed. This feature is handy if you're printing lots of growth rates, and need to conserve space.

\field <number>

> This command can be used to specify the column width of the data in the table. Give the column width as the argument. Note that this overrides the first argument in the general formatting command.

\pw (or \pagewidth) <number>

> Use this command to set the page width of the table to something other than the default of 132. This controls how the title is centered, and where the page numbers are printed. You may create a table that is wider than this page width, put the page number will not be at the right margin of the data.

\yf   **(or \yearformat)**

> This command lets you specify how you want dates printed. "\yf 4" means to make all dates 4 digit. "\yf 2" means to print all dates as 2 digit; use of 2-digit dates is not recommended. Finally "\yf m" means "mixed", i.e., to print levels as 4 digit dates, and growth rates as 2 digit dates. Note that mixed format is the default if none is specified.

\pp (or \pageprefix)

> The default leading text for page numbers is "Page ". If you would like to change it to something like "F -", give the command "\pp F -"

\under <character>

> This command specifies the underlining character for dates. For example, for single-line underlining, give the command "\under -". For double-line underlining, give the command "\under =". Any character is legal for underlining, except for 'n', which turns it off.

\pages <on|off>

> This command can be used to turn page numbering on or off.

\bz (or \blankzeroes) <y/n>

> This command allows you to print zeroes as blank fields. This sometimes makes a table less busy, and easier to read. The default is 'n', which means that zeroes will be printed. To enable this option, put "\bz y" in the stub file.

\nzr (or \nozerorows)

> The idea for this command is similar to the previous one. Sometimes we just don't want to clutter up the page with zeros. Even with zeros blank, we don't want to waste paper printing out blank space. After issuing this command in the stub file, series that are all zero will be skipped. In this way, you can create a stub file for, say, all 360 sectors of the *Iliad* model for Producers' Durable Equipment, and be confident that only the relevant PDE sectors will show up in the table.

\thresh (or \threshhold) <number>

> This is one more wrinkle on the same general idea. If you not only want to avoid looking at zeroes, but want to avoid looking at tiny numbers, you can set a threshhold. Any lines of data with no numbers above this threshhold will then not be printed.

\commas <on | off>

> Giving this command with the "on" argument, or with no argument, tells *Compare* to print numbers with commas separating thousands, millions, etc. It takes effect at the point in the table where the \comma command is given. "\comma off" turns commas back off again. This command helps with the formatting of very large numbers. However, remember that *Compare* and *G* work in single-precision floating point arithmetic. Therefore, only 7 digits are significant.

\spacing (sp) <number>

> This command changes the default spacing between variables or expressions in the table. The notation is similar to that of word-processors, where "1" means single spacing. "2" means to insert one blank line between variables or expressions. If you are comparing multiple banks or simulations, the spaces aren't inserted between each simulation, but only between separate variables. The purpose of this command is to allow space for notes in "scratch" tables, or to make lines of the table stand out when doing multiple simulations.

\missing <text>

> This command supplies text to be printed in the table for missing values of variables, or for expressions that default to missing values, because they are based on variables which have missing data. For example: "\miss N/A".

Commands That Control The Printer

The following commands relate to the control of the printer, and perform their functions by inserting various printer control strings into the Compare table file:

\landscape

> Issue this command if you want your table to print out in landscape mode on the laser printer. You can go back to portrait mode in the same table file, by giving the command "\landscape off". Note that you may want to increase the pagewidth with the "\pagewidth" command. For example:

> \pw 170   # set up for wide printing

> \landscape

> You may also want to set the page length to a shorter page. Landscape printing makes it possible to print up to 15 or 16 columns on the same page.

\portrait

> Issue this command if you are already in landscape mode, and want the rest of the pages of the table to be in portrait mode.

\lm <columns>

> This command is to prevent the left margin of tables from being too close to the side of the page. The number that you supply as <columns> for this command will be the number of columns on the left margin that the table will be indented. This makes it easier to put the tables into books. Note that spaces are not inserted into the table, but rather a printer code is added at the top of each page, that specifies how many columns to skip over before printing. This left margin can be changed dynamically within the document.

\lpp <lines>

> This command allows you to specify the number of printer lines per page. This command causes an HP LaserJet printer code to be printed at the top of each page of the table, which controls vertical lines per inch. Standard settings would be "\lpp 72" or "\lpp 77".

\pcontrol or \pc <control string or name>

> This command allows you to send a printer control string directly to the Compare table. Printer control strings are used in the Hewlett Packard printer control language (PCL5) to control the formatting and appearance of printouts. Many of the printer functions that can be accessed from the buttons on the printer can also be used by sending printer control strings. In fact, the other Compare commands "\landscap", "\lpp", "\lm" work by sending a printer control string in the header line of each page. Printer control strings appear somewhat strange to the unititiated. They always begin with the {ESC} character (ASCII 27). For example, the string for setting the printer into portrait mode is {ESC}&l1O (That's "el" "one", "capital O"). (I type in the {ESC} here because I can't in my word processor.) To type an {ESC} in the SEE editor, while in Edit mode, type "\", then "27". You can refer to the Laser Jet manual for other printer control

strings. Since these strings are not easy to remember, we have also created names for commonly used features.

These are listed below:

For use of bold, and italic typefaces:

>   \pc bold
>   \pc italic

For fonts, so far we have courier, line printer, pica and gothic:

>   \pc courier
>   \pc lineprinter
>   \pc pica
>   \pc gothic

And for sizes, so far one additional font size:

>   \pc large - 10.00 pitch.

However, note that any printer control string that works on your printer can be inserted into the file with this command.

\noformat

Compare achieves features such as "\lpp" (lines per page), "\landscape" (landscape mode), "\lm" (left margin) by insertting HP LaserJet printer codes in the top of the table file. Sometimes, such as when printing to a 9-pin printer, you want to force these codes off. Use the "\noformat" command to do this.

\nocomment

Issue this command when using the "\gdata" option to inhibit printing of comment lines in a file.

Commands to Build Tables in Excel Format

Many commands in this document apply regardless of the file format for the document that is being printed. Several commands apply only to the creation of spreadsheets with the \xls command. Use of these features requires that Microsoft Excel is installed on the user's machine.

\xls [<filename>]

> Issue this command to create spreadsheet documents. If a filename is specified, then it will be employed to name the document. If no name is given, then the name will be specified according to the *.IN configuration file.

\filetype <type>

> Issue this command if a file format other than the default XLS is desired. Other available file types include AddIn (.xlam), CSV (.csv), CSVMac (.csv), CSVMSDOS (.csv), CSVWindows (.csv), DBF2 (.dbf), DBF3 (.dbf), DBF4 (.dbf), DIF (.dif), Excel2 (.xls), Excel2FarEast (.xls), Excel3 (.xls), Excel4 (.xls), Excel5 (.xls), Excel7 (.xls), Excel9795 (.xls), Excel4Workbook (.xls), IntlAddIn (.xla), IntlMacro (.xlsm), WorkbookNormal (.xls), SYLK (.slk), Template (.xltx), TextMac (.txt), TextMSDOS (.txt), TextPrinter (.txt), TextWindows (.txt), WK1 (.wk1), WK1ALL (.wk1), WK1FMT (.wk1), WK3 (.wk3), WK4 (.wk4), WK3FM3 (.wk3), WKS (.wks), WQ1 (.wq1), UnicodeText (.txt), Html (.html), and XLS (.xls).

\wstitle <worksheet name>

> This command names the current worksheet when printing to a Microsoft Excel spreadsheet. New worksheets are created in the standard fashion; that is, the '*' command creates a new worksheet.

Compare allows the user to format specific parts of an Excel spreadsheet by inserting commands into a stub file. The font in the header, title, dates, banks, series names, and data can changed. Specifically, the type face, size, color, vertical and horizontal alignment, bold status, italic status, and underline status and type can be controlled. The number and order of settings does not matter. the type face is specified by giving the name surrounded by double quotes. Size is adjusted by entering "size" followed by an integer. Horizontal alignment is controlled by entering "justify", "center", "right", or "left" for the horizontal alignment. Vertical alignment can be specified by "top", "vcenter", "bottom", or "vjustify". A complete listing of Excel font settings are listed in Appendix A at the end of this document. To remove a formatting option for any area of an Excel spreadsheet, enter "clear" as the setting.

\fonthead <settings>

This command will control formatting for the \head output.

\fonthead "arial" size32 bold underline

\fonttitle <settings>

This command will control formatting for the \title output.

\fonttitle "times new roman" size24 bold

\fontdate <settings>

This command will control formatting for \date output.

\fontdate "times new roman" size14 underline center

\fontbank <settings>

This command will control formatting for the printed banks names.

\fontbank "courier new" size10

\fontname <settings>

This command will control formatting for the bank series names.

\fontname "verdana" size12 italic left

\fontdata <settings>

This command will control formatting for the numerical data output.

\fontdata "arial" size11 right

\font <settings>

This command is used to set the default font formatting and will override all other font commands.

\font "courier new" size12 left

Calculation of Data Within Compare

The "\f" command works like the "f" command in *G*. The original motivation for the "\f" command was to create variables that could be used later in the *Compare* table, or even intermediate variables to be used in further "\f" calculations.

\f <name> = <expression>

> This command is parallel to the "f" command in *G*. The format is exactly the same as in *G*. The "\f" command works by creating a "Compare workspace bank" for each simulation or databank that is being compared. The name of each bank is of the form "CWSx.BNK", where "x" is the number of the simulation, starting at zero. Therefore, after doing a Compare table of 4 simulations, and using the "\f" command in the table, there should be banks CWS0.BNK, CWS1.BNK, CWS2.BNK and CWS3.BNK in your directory. They will contain any series created with the "\f" command. When *Compare* is looking for a series, it checks first in the Compare workspace bank to see if the series has been put there by a "\f" command. Then it checks the corresponding simulation bank.

At present, the *Compare* workspace banks are not created until the first use of the "\f" command, and remain in use after *Compare* is finished. Of course, they can be assigned by *G* just like any other workspace bank, so you may find further uses for them beyond what was originally intended.

Calculation and Display of Running Totals

The next three commands work together, and are used for the calculation and display of running totals.

\starttotal (\st)

> This and the two following commands are used together, for calculating running totals. The "\st" command should be placed before the first variable you want included in the running total.

\endtotal (\et)

> This command should be placed after the last variable you want included in the running total.

\printtotal (\pt) <series title text>

> When you finally want to print the total in the table, issue the "\pt" command. The <series title text> is whatever text you want to appear to the left of the data series, such as "Total for All Industries", or something of this sort.

Commands for Sorting or Ranking

These three commands work together, and enable rankings to be done with Compare. They enable you to rank by levels, sums, growth rates or averages. The "startsort" command is given before the range of lines that you want to sort, and "endsort" is given after that range. Printing of those lines will be suppressed until the first "printsort"

command, and printing of the rest of the table should resume after "printsort". Note that for these features to work, all three commands must be in the same file. In other words, do not put "add" or "fadd" commands between the "startsort" and "endsort"

\startsort (ss) <date expression> [<size to allocate>]

> Put the "startsort" command before the lines in the stub file that you would like to sort. The date expression represents the number that you will use as the criteria for the sort. This can be a value for single year, or a range of years indicating a growth rate (e.g. 90-97), a sum (90+97) or an average (90|97). Note that the date expression used for the sorting does not have to actually be a date expression used in the table, but can be any valid date expression which makes sense for the bank you are using. Note that if you are comparing several banks, the sorting will be done based on the value of the variables for this date expression in the first simulation bank.

> The last argument for "startsort" is optional. By default, enough space is set up to sort up to 500 lines. However, if you want to save memory, you may reduce this number, or if you need more than 500 you need to increase it. Note that Compare will only sort lines that would normally print out values, i.e., either variables or expressions. All other printing lines such as "\center", "\line", ";", etc. will be skipped.

> Example:

> > \startsort 90-95 1000

\endsort (es)

> The "endsort" command marks the end of the range of the table that will be sorted. Note that all table lines between the "startsort" and "endsort" commands will not be printed until a "printsort" command is given.

\printsort (ps) [<direction>] [<numtoprint>]

> The "printsort" command is the place where all the work is done. Both arguments are optional. The first, the sort direction is 'd' (descending) by default. If you want the sort to be in ascending order, give an 'a'. The second option is the number of items you would like to print. For example, if you want to print the 10 largest items, you would use:

> > \printsort d 10

Miscellaneous Commands

\toc

This command turns on table of contents generation. The table of contents will be generated as the last page of the table. The page heading will be the title of the entire table, as given in the \head command. Each entry will be triggered by the presence of a \ti command either in the main stub file, or in any \add'ed stub files. If table of contents generation is turned on, each table title will be preceeded by the text: "TABLE 1.", "TABLE 2.", etc. Page numbers in the table of contents will have the page prefix tacked on, as given by the \pageprefix command.

\timestamp or \ts

This command will specify the printing of a date and time, centered at the bottom of each page. This is the date and time that the table was generated.

\mem

This command reports the amount of memory currently available. If you are working with large Interdyme matrices or long @csum() commands, you can bump into memory limits.

\tell

This command tells the time and date on the screen, to aid in timing table creation.

\announce <announcement>

This command is useful if you want to announce when a certain portion of a stub file is being processed, for debugging reasons, or just to let yourself know how far you are in the process. If you are using a lot of \add files, you may want to put a \announce command near the top of each one, so you can trace when each one is done.

\at (or \aggtype)

This command is useful to change the way that data is calculated "at annual rates". This is a subject that is clouded by the fact that people don't agree what annual rates are. The Inforum approach, which is the default ("\at i") yields annualized growth rates which are exactly the frequency of the data times the period growth rates. This is internally consistent, and gives annualized growth rates that yield the same numbers as the corresponding annual growth rates. The BEA approach, which seems to have been adopted by just about everyone else (the CEA, the Fed and the Washington Post, to name a few), is to take the period to period ratio to the n'th power (where n is the frequency), then subtract 1.0 and multiply by 100. In other words, where xt is the current quarter, and x0 is the previous quarter, this method is calculated by the following formula:

$$100.0*(exp(4.0*log(xt/x0))-1.0)$$

It is not clear what is meant by "annual rates" in this sense, but this is what everybody publishes. To use the BEA approach, give the command "\at b". Take your pick.

\ar

> Use if monthly or quarterly data is given at annual rates (the default). (ar = annual rates)

\pr

> Use if monthly or quarterly data is given at monthly or quarterly rates (pr = period rates). Note that it only makes sense to compare period and annual rates if the growthtype is exponential ('e').

Groups in Compare

Compare has been able to use group expressions in @csum() functions for some time now. However, the newest version has adopted the feature used in G7 of named groups. Named groups are created by running the *Fixer* program, which creates a groups file called GROUPS.BIN. If a GROUPS.BIN file is in the current directory, *Compare* will now read it and store the group definitions in that file. For example, if the *Fixer* input file was VECFIX.VFX, and it had as input:

> group Manufacturing
>
> > 9-58
>
> the group called "Manufacturing" would be created and stored in GROUPS.BIN. This could then be used in the Compare @csum() function in a stub file:
>
> @csum(emp,:Manufacturing) ; Manufacturing employment

Note that the group name is prefixed by a colon (':').

Some other commands that can be used for checking the groups defined in the GROUPS.BIN file are taken from Vam:

\listgroups
> This will print out a list of the currently defined groups to the screen.

\glist <groupname>
> If <groupname> is one of the groups printed in the list above, then this command will print out those sectors or categories comprising the group.

Note that neither of these commands affects the output of the table, but are only for providing infomation to the screen.

Commands Used With Interdyme Vam Files

Note that when printing a table of series from Interdyme, for vectors it is usually advantageous to issue the "\load" command before printing the vector. This loads the data for the entire vector into memory, and results in much faster table printing. Also, note that you can print time series of matrix elements now in Compare. To print out the values of the A-matrix row 1, column 1, you would insert the series name "am1.1", if am

were the name given to the A-matrix in the Interdyme VAM.CFG file. Note the dot that separates the row and the column.

\load \<vectorname\>

> This command can be useful when using *Compare* to print tables of vector variables, for example if you are printing a table of many industries or sectors for the same vector. Use of this command may results in a significant increase in speed.

\matlist \<groupdef\>

> This is the command to ask *Compare* to print a matrix listing. Note that this command only works on a vam file, and the program will complain about any other type of file. The group definition can be a simple list of sectors, or it can include ranges of sectors specified by two sector numbers separated by a dash. Sectors or intervals in parentheses will be excluded.

> Example:

> \matlist 20 24 30-50 (41 45 46-48)

> would perform a matrix listing for sectors 20,24, 30 through 50, with 41,45 and 46 through 48 excluded.

\row

> This command is to be given before requesting the matrix listing. This specifies that a row listing is to be printed.

\column

> This command specifies that a column listing is to be printed, and also must be given before the \matlist command.

\cutoff \<cutoff ratio\> [\<year\>]

> This command specifies the cutoff, in terms of a ratio in terms of output, on what flows to print. For example "\cutoff .005" says to print all flows that are greater than .5% of output. You can specify the cutoff year (optional), and then the data for that year will be used to determine if the line should be printed in the matrix listing or not.

\cd (or \coefdecs) \<number\>

> This command is really only relevant for matrix listings, as it allows you to specify how many decimal places I-O coefficients will be displayed at.

\nst

> This command is used to turn off "seller titles" in matrix listings. In some cases, the matrix listing is used to print out detailed flows from a matrix identity other than the typical buyer or seller listing. An example would be a table of flows of employment by occupation.

\nosubtot

> This command, also for matrix listings for Interdyme, will suppress the printing of subtotals at the bottom of a matrix listing section. This is useful when the items

of the matrix already include aggregates. Calculating a subtotal would imply double counting.

\matcfg (\mc) <filename>

This option allows you to specify a different configuration file for a matrix listing besides the normal "matlist.cfg". The matlist configuration file specifies the matrix identity that the matrix listing will summarize. In a complicated model, with many matrix identities, such as a regional model, you may need many alternative configuration files. This command allows you to switch easily between these files.

Variables, Expressions and Functions

A line beginning in any other way will be presumed to begin with a variable name, such as gnp$, or an expression. For the expressions, all the functions available in *G* are also available in *Compare*. In addition, *Compare* has a useful function for sectoral models, called @csum(), which is used to sum up a specified group of industries for a given data concept. For example, suppose you wanted a certain line in your table to contain the sum of exports ("exp") for sectors 1 to 10. Then the formula to use in the name section of the stub file would be:

@csum(exp,1-10)

The expression:

@csum(exp,1-12 (4-7) 15 18)

would include exports of sectors 1 to 12 inclusive, except for sectors 4 to 7 inclusive, and then sectors 15 and 18 in addition.

Note that when working with Interdyme Vam files, a series name is followed by the vector name and the sector. Therefore, output of sector 10 would be "out10", if the output vector is named "out". To print a matrix element, use the matrix name, plus the row index, then a period ("."), and finally the column index. For example, "am1.1" is am(1,1).

After the name or expression comes a ';'and after the ";" is a line title, which will be printed at the left side of the line. The length of this line title is specified by the formatting command described above. Leading blanks -- blanks between the ";" and the first visible letter on the line -- will be printed and can be used to indent the titles.

II. _Running the Compare Program_

Once the ".stb" file is ready, one can run Compare by typing:

> compare

at the DOS prompt.

As soon as _Compare_ starts, it asks you how many alternatives you want to see in the table (see the sample session in the box below). Of course, respond with '1' if you are just going to make a table from 1 bank. You may have up to 10 alternatives. Next, for each alternative, you are asked to specify what type of bank this alternative should be read, and then the rootname of the data bank file.   Answer this first question with one character, and then hit {ENTER}. Type 'w' if this is a normal G bank, or workspace type bank (.bnk), 'c' if this is a compressed G bank (.cbk), 'h' if this is a hashed G bank (.hbk). 'd' if this is a dirfor file (.dfr), and 'v' if this is a vam file (.bin). Note that if you want to use a dirfor file, you must have the dirfor.dat corresponding to that file in the current directory in which you are running Compare. Make sure that the MACRONAM.BIN specified in that file points to a valid location. (See the LIFT manual, in the Display chapter for more details on DIRFOR.DAT.) If you are assigning a vam file, note that Compare automatically assumes that each VAM file has a "sister" G bank, with the same root name, same directory.   It will try to open this file to find series such as macrovariables which may not be in the VAM file. See the InterDyme manual for more details on vam files. Remember with all databank file names, to give only the root name of the file (no file extension.).

Next, only if you are printing from more than one bank, you are asked whether you would like to see the alternatives in actual values ('a'), as differences ('d') from the base run, as percentage differences ('p') from the base. Type the indicated letter and press {Enter} to show your choice. (If you are listing only 1 data bank or a single run of a model, it does not matter how you answer this question.)
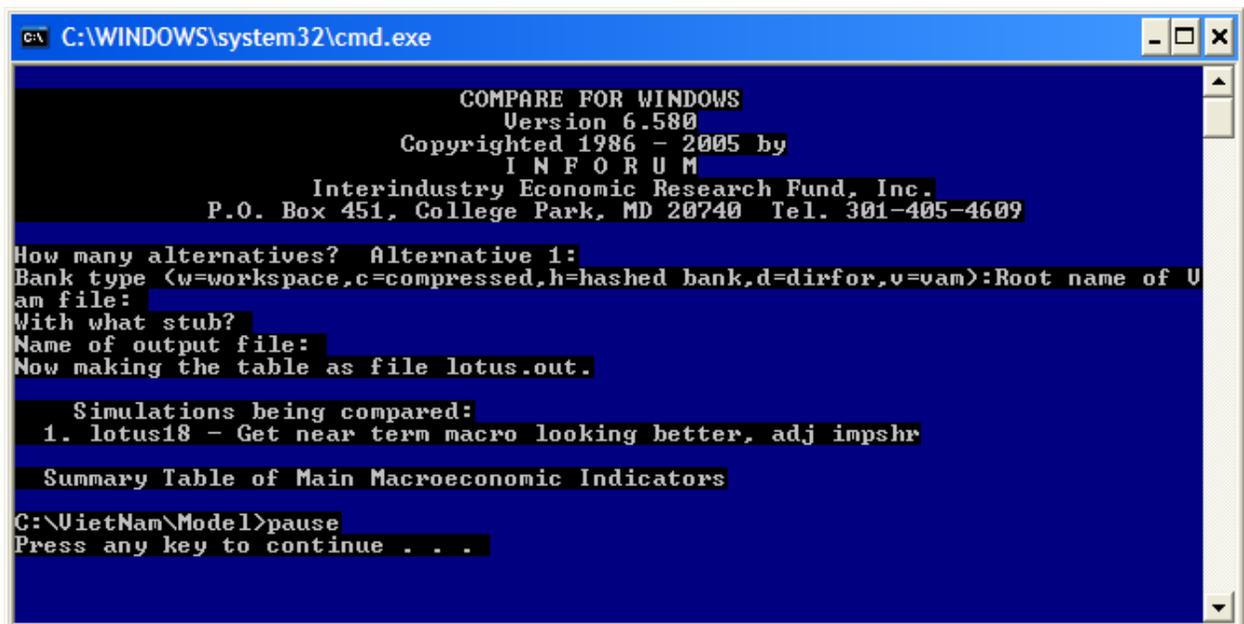
```
C:\WINDOWS\system32\cmd.exe                                    _ □ ×

                      COMPARE FOR WINDOWS
                         Version 6.580
                    Copyrighted 1986 - 2005 by
                         I N F O R U M
              Interindustry Economic Research Fund, Inc.
          P.O. Box 451, College Park, MD 20740  Tel. 301-405-4609

How many alternatives?  Alternative 1:
Bank type (w=workspace,c=compressed,h=hashed bank,d=dirfor,v=vam):Root name of V
am file:
With what stub?
Name of output file:
Now making the table as file lotus.out.

    Simulations being compared:
  1. lotus18 - Get near term macro looking better, adj impshr

   Summary Table of Main Macroeconomic Indicators

C:\VietNam\Model>pause
Press any key to continue . . .
```

You will then be asked for the name of the stub file and you reply with the name of the stub file you have prepared. (Use the full name, including the ".stb".) Finally it asks for the name of the output file. We commonly use the ".out" suffix for these files, but any name is OK. When Compare has finished and given the DOS prompt again, you can use the DOS Print command to obtain a printed copy.

As an alternative to answering the individual questions asked by Compare, you can put the answers into a file such as "compare.in" and start the program with

compare compare.in

Compare assumes that a filename given on the command line is a file containing input responses. The box below shows the response file that would correspond to the example on the previous page.

```
1
v
dyme
mast.stb
tabpf.out
```

A configuration file format also is available, which can be used by specifying the "-f" option on the command line. For example, if your configuration file is named COMPARE.CFG, you can specify using this with: "compare -f compare.cfg". Note the space between the "-f" and the file name. An example config file is in the box below.

```
Number of Simulations; 1
1st bank type; h
1st bank name; h:\ami\quip
Stub file; h:\ami\quip.stb
Output file; quip.out
```

Note that for the IdLift model, a stub file called DYME.STB has already been prepared, which contains most of the macrovariables, and many of the sectoral variables in the model. The QUEST model already has stub files called LONGTERM.STB and SHORTTRM.STB, that produce tables similar to those in the QUEST section of the meeting book. If you would like to make tables containing only portions of these file, you can cut and paste to create new stub files. These files can also be used to print or graph data in *G* using the 'look' command (see the G documentation).

III. *Special Uses with Compare*

This section describes how to make a matrix listing for Interdyme using Compare, and how to make tables using Dirfor files.

Using Compare as a Matrix Lister for Interdyme

A "matrix listing" displays all of the cells in a row or column of an input-output table. The command in the stub file is simply

> \ matlist <sectors>

For example, it could be

> \ matlist 1 5 7 15-30 (21 23-25)

where we have used the now-familiar system of indicating a list of sectors.

The \ matlist command should be preceded by two related commands, \ row or \ column and \ cutoff. Here is a complete example of a stub file for a matrix listing:

```
\date 1987 1988 1989 1991 1993 1995 2000
91-95 90-100 95-2000
\ti MUDAN MATRIX LISTING
\9 1 66 1 3 33
\row
\cutoff 0.02
\matlist 18-25
```

This will produce a row listing; a cell of a matrix must account for at least .02 of the total (two percent of the row total) in order to be listed.   Rows 18 - 25 will be listed.

The \ matlist command causes *Compare* to look for a file which at present must be called "matlist.cfg". Here is an example from Mudan.

```
Matrix listing identity; out = am*out+cr+cu+bmv*capital+vin+exp-
imp+othdm
# Title file name for the rows of out, the lefthand side vector
out; "sectors.ttl"
# Title file names for matrix columns
am; "sectors.ttl"
bmv; "bmv.ttl"
# headers for each term
header for out;        "Output"
header for am*out;   "Intermediate"
header for cr;        "Rural Consumption"
header for cu;        "Urban Consumption"
hdr for bmv*capital; "Investment"
header for vin;       "Inventory"
header for exp;       "Exports"
header for imp;       "Imports"
header for othdm;    "Other demand"
```

In this file, all lines beginning with a # are comments, and anything before a ';' on a line is also a comment. The first line that does not begin with a # must be the matrix listing identity. It specifies, in terms of the particular model, an input-output identity. Usually this identity says, in effect, total output = intermediate demand plus final demand. But other identites are possible. The identity must have a single vector on the left and then on the right an expression that is the sum of a number of terms. Each term may be either a single vector, like cr and cu in the example, or may be a matrix*vector product, such as am*out and bmv*capital. The terms should be joined by + or - signs. The matrix*vector terms result in a display of all flows obtained by multiplying each column of the matrix by the corresponding element of the vector.

Following the identity, the next non-comment line must name the title file name for the vector on the left hand side of the identity. The file name must be enclosed in quotation marks (""). These title files can be the same one used with *G*, but *Compare* skips the 10-letter abbreviation at the beginning of each line which is used by *G* in the show command. Rather, *Compare* looks for a string within " marks, and uses that. There can be other material in front of or after the quoted string. Here are the first few lines of the sectors.ttl file used in the above example.

```
Agricul      ;1   e "Agriculture"
Coal         ;2   e "Coal"
CrudeOil     ;3   e "Crude Oil & Natural Gas "
```

There must also be file names for the titles of the columns of every matrix in a row listing or the row of every matrix in a column listing. In the example they are sectors.ttl and bmv.ttl. These files should be of the form as just explained with the titles between quotes.

Finally, for each term in the identity, matlist.cfg must show the name to be listed with the term. These term names are also show as strings within quotes. For a vector term, this name will be listed on the left. For a matrix*vector term, it will be centered above the display of the cells of the matrix. The box below shows a sample of the matrix listing produced by our example.

```
                         MUDAN MATRIX LISTING
                              Seller:  18 Non-electrical Machinery
                       1987    1990    1995    2000    90-95   95-00
                                       Sales to Intermediate
  3 Crude Oil & Natural Gas   20.2    37.1    42.0    45.8    2.49    1.76
 14 Chemicals                 24.8    60.9    93.0   138.4    8.46    7.96
 15 Building Materials        24.7    52.3    77.8   108.8    7.95    6.71
 16 Metallurgy                46.0    91.7   124.0   161.6    6.03    5.31
 18 Non-electrical Machinery 283.6   376.0   538.6   750.3    7.19    6.63
 19 Tranportation Equipment   43.4   102.7   138.3   203.8    5.95    7.76
 20 Electrical Machinery      31.5    74.7   112.7   177.1    8.21    9.05
 25 Construction             109.9   164.3   228.8   290.3    6.63    4.76
 31 Education, Health, and Scienc  19.2  37.7  56.6   89.0    8.14    9.07
SUM: Intermediate            757.4  1314.5  1856.4  2576.7   6.90    6.56
                                       Sales to Other Final Demand
Rural Consumption             80.9    83.3   139.4   205.3   10.30    7.75
Urban Consumption             36.5    43.0    84.5   158.2   13.49   12.56
                                       Sales to Investment
  1 Agriculture etc.          12.3    12.3    20.5    30.8   10.14    8.16
 11 Electricity etc.          39.3    39.7    56.5    86.0    7.07    8.42
 14 Chemical Industry         30.7    25.3    41.1    62.6    9.66    8.42
 16 Metallurgical Indust      32.3    21.3    36.2    54.7   10.61    8.24
 25 Trans.& Communication     42.6    31.0    60.8    92.6   13.48    8.43
 27 Public utility and service 22.4   12.8    22.8    34.7   11.59    8.42
 35 Other state-owned units   33.1    17.2    34.1    56.5   13.66   10.10
 36 Urban collective-owned units 30.4 14.1    31.1    53.3   15.91   10.77
 37 Rural collective-owned units 61.3 31.5    57.3    92.3   11.97    9.54
 38 Rural individuals        117.6    76.1   102.3   126.9    5.92    4.31
 40 Balancing item            27.1    17.4    22.2    33.8    4.82    8.42
SUM: Investment              660.9   458.7   726.4  1094.5    9.19    8.20
Inventory                     70.7    51.4    72.8    93.8    6.94    5.07
Exports                      155.8   357.1   502.8   648.6    6.84    5.09
Imports                      385.9   500.9   771.7  1124.1    8.64    7.52
Other demand                  19.3    42.2    42.2    42.2    0.00    0.00
Output                      1398.6  1854.0  2656.1  3700.2    7.19    6.63
```

Using Dirfor Files with Compare

Although *Compare* was primarily developed first for use with *G* and Interdyme, it can now also be used with the DIRFOR files that are traditionally created by the LIFT and SLIMFORP software. This section will illustrate what needed to be done to get the Japanese model Dirfor's into Compare, but any SLIMFORP model that uses a Dirfor file structure can be adapted for use with the Compare program.

The key to the Dirfor file structure is the DIRFOR.CRD file, found in almost every SLIMFORP model (in LIFT, this is called DIRFOR.DAT). Those of you who work with LIFT or SLIMFORP are already familiar with this file, so it won't be described here in complete detail. However, there is a subtle difference in the format of the LIFT and SLIMFORP formats of this file. The *Compare* program works with the LIFT format as is, but minor changes must be made to the SLIMFORP versions.

```
daf\DIRHIS,
TOTAL NUMBER OF RECORDS
3064          /*   =  1 + 22*67 + 5*25 + 10*26 + 2*67 +10*67 + 1*400
NUMBER OF RECORDS PRECEDING EACH BLOCK    (IGN)
1     1475    1600    1860    1994    2664    3064
BLOCK    1 SERIES 22 SECTOR 67          IO LEVEL DATA
```

The DIRFOR.CRD for the Japanese model will be used in the example discussion below. The first change that needs to be made to the DIRFOR.CRD is in the section describing the blocks of the DIRFOR file. In the usual SLIMFORP format, these lines are as in the box below:

The disadvantage of this traditional format is that, each time that the sector numbering, or number of series in each block changes, these numbers must be hand edited, and new starting record numbers for each block must be calculated. In the LIFT format, these same lines would be written as follows:

```
daf\DIRHIS,
 6 blocks
Sectors in block  67  25  26  67  67 400
Number of series  22   5  10   2  10   1
BLOCK    1 SERIES 22 SECTOR 67          IO LEVEL DATA
```

In this format, the number of blocks is given in the first line, in the format (I2), and the number of sectors and number of series in each block are given in the following two lines, in the format (16X,12I4). The LIFT version of Mkdirfor will calculate the starting record numbers of each block automatically, based on these data, and Compare does the same.

```
BLOCK    1 SERIES 22 SECTOR 67          IO LEVEL DATA
SERIES   1 RDFIL   1 RPOINT  1                CENTRAL GOVERNMENT
IHIS   1history\CEN5584.HIS,
SERIES   2         1         2                LOCAL GOVERNMENT
IHIS   1history\LOC5584.HIS,
SERIES   3         1         3                WATER & SANITATION
IHIS   1history\WAS5584.HIS,
```

The next difference is that each series in the new format has its own mnemonic, by which it can be referred to in making tables. The traditional DIRFOR series format is displayed in the box below:

To change this to the new format, you need only to add mnemonics of up to 4 characters, starting in column 32:

```
BLOCK    1 SERIES 22 SECTOR 67          IO LEVEL DATA
SERIES   1 RDFIL   1 RPOINT  1 CEN            CENTRAL GOVERNMENT
IHIS   1history\CEN5584.HIS,
SERIES   2         1         2 LOC            LOCAL GOVERNMENT
IHIS   1history\LOC5584.HIS,
SERIES   3         1         3 WAS            WATER & SANITATION
IHIS   1history\WAS5584.HIS,
```

Once this is done, Compare can read the SLIMFORP file, using mnemonics such as cen1, loc21, etc. During the interactive running of the program, if you specify file type 'd', for Dirfor, you will also be asked to give the path of the DIRFOR.DAT file. You may want to keep a copy in the traditional format as DIRFOR.CRD, and a copy in the new format as DIRFOR.DAT. Then you can use DIRFOR.DAT when running *Compare*.

Appendix A: Excel Font Format Settings

Underline

Underlining may be specified in several ways in the \font commands. Note that the \under command also may be used set the underline style for printing dates.

underline [-]

UnderlineSingle

-

Single underlining.

underline [=]

UnderlineDouble

=

Double underlining.

underline [_]

UnderlineSingleAccounting

_

Single accounting underlining.

underline [~]

UnderlineDoubleAccounting

~

Double accounting underlining.

underline n

UnderlineNone

No underlining.

Font Alignments

Horizontal alignment is controlled by entering "justify", "center", "right", or "left" for the horizontal alignment. Vertical alignment can be specified by "top", "vcenter", "bottom", or "vjustify".

Font Colors

"None"

"Aqua"

"Black"

"Blue"

"Cream"

"Dark Gray"

"Fuchsia"

"Gray"

"Green"

"Lime green"

"Light Gray"

"Maroon"

"Medium Gray"

"Mint green"

"Navy blue"

"Olive green"

"Purple"

"Red"

"Silver"

"Sky blue"

"Teal"

"White"

"Yellow"

Font Types

"Agency FB"

"Agency FB Bold"

"Algerian"

"Arial"

"Arial Black"

"Arial Black Italic"

"Arial Bold"

"Arial Bold Italic"

"Arial Italic"

"Arial Narrow"

"Arial Narrow Bold"

"Arial Narrow Bold Italic"

"Arial Narrow Italic"

"Arial Rounded MT Bold"

"Arial Unicode MS"

"Baskerville Old Face"

"Batang"

"Bauhaus 93"

"Bell MT"

"Bell MT Bold"

"Bell MT Italic"

"Berlin Sans FB"

"Berlin Sans FB Bold"

"Berlin Sans FB Demi Bold"

"Bernard MT Condensed"

"Blackadder ITC"

"Bodoni MT"

"Bodoni MT Black"

"Bodoni MT Black Italic"

"Bodoni MT Bold"

"Bodoni MT Bold Italic"

"Bodoni MT Condensed"

"Bodoni MT Condensed Bold"

"Bodoni MT Condensed Bold Italic"

"Bodoni MT Condensed Italic"

"Bodoni MT Italic"

"Bodoni MT Poster Compressed"

"Book Antiqua"

"Book Antiqua Bold"

"Book Antiqua Bold Italic"

"Book Antiqua Italic"

"Bookman Old Style"

"Bookman Old Style Bold"

"Bookman Old Style Bold Italic"

"Bookman Old Style Italic"

"Bradley Hand ITC"

"Britannic Bold"

"Broadway"

"Brush Script MT Italic"

"Californian FB"

"Californian FB Bold"

"Californian FB Italic"

"Calisto MT"

"Calisto MT Bold"

"Calisto MT Bold Italic"

"Calisto MT Italic"

"Castellar"

"Centaur"

"Century"

"Century Gothic"

"Century Gothic Bold"

"Century Gothic Bold Italic"

"Century Gothic Italic"

"Century Schoolbook"

"Century Schoolbook Bold"

"Century Schoolbook Bold Italic"

"Century Schoolbook Italic"

"Chiller"

"Colonna MT"

"Comic Sans MS"

"Comic Sans MS Bold"

"Cooper Black"

"Copperplate Gothic Bold"

"Copperplate Gothic Light"

"courier"

"courierbold"

"courierbolditalic"

"courieritalic"

"Courier New"

"Courier New Bold"

"Courier New Bold Italic"

"Courier New Italic"

"Curlz MT"

"Edwardian Script ITC"

"Elephant"

"Elephant Italic"

"Engravers MT"

"Eras Bold ITC"

"Eras Demi ITC"

"Eras Light ITC"

"Eras Medium ITC"

"Felix Titling"

"Footlight MT Light"

"Forte"

"Franklin Gothic Book"

"Franklin Gothic Book Italic"

"Franklin Gothic Demi"

"Franklin Gothic Demi Cond"

"Franklin Gothic Demi Italic"

"Franklin Gothic Heavy"

"Franklin Gothic Heavy Italic"

"Franklin Gothic Medium"

"Franklin Gothic Medium Cond"

"Franklin Gothic Medium Italic"

"Freestyle Script"

"French Script MT"

"Garamond"

"Garamond Bold"

"Garamond Italic"

"Gigi"

"Gill Sans MT"

"Gill Sans MT Bold"

"Gill Sans MT Bold Italic"

"Gill Sans MT Condensed"

"Gill Sans MT Ext Condensed Bold"

"Gill Sans MT Italic"

"Gill Sans Ultra Bold"

"Gill Sans Ultra Bold Condensed"

"Gloucester MT Extra Condensed"

"Goudy Old Style"

"Goudy Old Style Bold"

"Goudy Old Style Italic"

"Goudy Stout"

"Haettenschweiler"

"Harlow Solid Italic"

"Harrington"

"High Tower Text"

"High Tower Text Italic"

"Impact"

"Imprint MT Shadow"

"Informal Roman"

"Jokerman"

"Juice ITC"

"Kristen ITC"

"Kunstler Script"

"Lucida Bright"

"Lucida Bright Demibold"

"Lucida Bright Demibold Italic"

"Lucida Bright Italic"

"Lucida Calligraphy Italic"

"Lucida Fax Demibold"

"Lucida Fax Demibold Italic"

"Lucida Fax Italic"

"Lucida Fax Regular"

"Lucida Handwriting Italic"

"Lucida Sans Demibold Italic"

"Lucida Sans Demibold Roman"

"Lucida Sans Italic"

"Lucida Sans Regular"

"Lucida Sans Typewriter Bold"

"Lucida Sans Typewriter Bold Oblique"

"Lucida Sans Typewriter Oblique"

"Lucida Sans Typewriter Regular"

"Magneto Bold"

"Maiandra GD"

"Map Symbols"

"Matura MT Script Capitals"

"Mistral"

"Modern No. 20"

"Monotype Corsiva"

"MS Mincho"

"MS Outlook"

"MT Extra"

"Niagara Engraved"

"Niagara Solid"

"OCR A Extended"

"Old English Text MT"

"Onyx"

"Palace Script MT"

"Palatino Linotype"

"Palatino Linotype Bold"

"Palatino Linotype Bold Italic"

"Palatino Linotype Italic"

"Papyrus"

"Parchment"

"Perpetua"

"Perpetua Bold"

"Perpetua Bold Italic"

"Perpetua Italic"

"Perpetua Titling MT Bold"

"Perpetua Titling MT Light"

"Playbill"

"PMingLiU"

"Poor Richard"

"Pristina"

"Rage Italic"

"Ravie"

"Rockwell"

"Rockwell Bold"

"Rockwell Bold Italic"

"Rockwell Condensed"

"Rockwell Condensed Bold"

"Rockwell Extra Bold"

"Rockwell Italic"

"Script MT Bold"

"Showcard Gothic"

"SimSun"

"Snap ITC"

"Stencil"

"Symbol"

"Tahoma"

"Tahoma Bold"

"Tempus Sans ITC"

"Times"

"timesbold"

"timesbolditalic"

"timesitalic"

"Times New Roman"

"Times New Roman Bold"

"Times New Roman Bold Italic"

"Times New Roman Italic"

"Trebuchet MS"

"Trebuchet MS Bold"

"Trebuchet MS Bold Italic"

"Trebuchet MS Italic"

"Tw Cen MT"

"Tw Cen MT Bold"

"Tw Cen MT Bold Italic"

"Tw Cen MT Condensed"

"Tw Cen MT Condensed Bold"

"Tw Cen MT Condensed Extra Bold"

"Tw Cen MT Italic"

"Verdana"

"Verdana Bold"

"Verdana Bold Italic"

"Verdana Italic"

"Viner Hand ITC"

"Vivaldi Italic"

"Vladimir Script"

"Wide Latin"

"Wingdings"

"Wingdings 2"

"Wingdings 3"