Outline

1. Dynamic Data Exchange
   - Dynamic Data Exchange
   - Resources
   - Using DDE
What is DDE?

DDE, Dynamic Data Exchange, is a built-in protocol in Microsoft Windows that allows applications to share data (chat among applications, that is)
Use DDE with Excel in SAS

- can dump SAS data directly into cells in Excel
- can easily format cells (colors, fonts, outlines, etc)
- can write formulas to cells
- can create/kick off macros from within SAS
- can issue system commands such as saving workbook to files, rename worksheets, etc.
Use DDE with Excel in SAS

- can dump SAS data directly into cells in Excel
- can easily format cells (colors, fonts, outlines, etc)
- can write formulas to cells
- can create/kick off macros from within SAS
- can issue system commands such as saving workbook to files, rename worksheets, etc.
Use DDE with Excel in SAS

- can dump SAS data directly into cells in Excel
- can easily format cells (colors, fonts, outlines, etc)
- can write formulas to cells
- can create/kick off macros from within SAS
- can issue system commands such as saving workbook to files, rename worksheets, etc.
Use DDE with Excel in SAS

- can dump SAS data directly into cells in Excel
- can easily format cells (colors, fonts, outlines, etc)
- can write formulas to cells
- can create/kick off macros from within SAS
- can issue system commands such as saving workbook to files, rename worksheets, etc.
Use DDE with Excel in SAS

- can dump SAS data directly into cells in Excel
- can easily format cells (colors, fonts, outlines, etc)
- can write formulas to cells
- can create/kick off macros from within SAS
- can issue system commands such as saving workbook to files, rename worksheets, etc.
Use DDE with Excel in SAS

- can dump SAS data directly into cells in Excel
- can easily format cells (colors, fonts, outlines, etc)
- can write formulas to cells
- can create/kick off macros from within SAS
- can issue system commands such as saving workbook to files, rename worksheets, etc.
Besides SAS help documentation for Windows, many SUGI papers covered this topic, two important ones:

- “Using Dynamic Data Exchange to Export Your SAS Data to MS Excel — Against All ODS, Part I —” SUGI 26 (Paper 11-26) by Koen Vyverman.
Besides SAS help documentation for Windows, many SUGI papers covered this topic, two important ones:

- “Using Dynamic Data Exchange to Export Your SAS Data to MS Excel — Against All ODS, Part I —” SUGI 26 (Paper 11-26) by Koen Vyverman.

Besides SAS help documentation for Windows, many SUGI papers covered this topic, two important ones:

- “Using Dynamic Data Exchange to Export Your SAS Data to MS Excel — Against All ODS, Part I —” SUGI 26 (Paper 11-26) by Koen Vyverman.

Using DDE

Prerequisite for using DDE: both client and server applications are up and running. A client/server DDE communication link is initiated by means of a special form of the SAS *filename* statement in two ways:

- **DDE triplet:**
  
  ```
  FILENAME fileref DDE '<server app>|<topic>!<item>’;
  ```

- **DDE ‘doublet’:**
  
  ```
  FILENAME fileref DDE '<server app>|system’;
  ```
Using DDE

Prerequisite for using DDE: both client and server applications are up and running. A client/server DDE communication link is initiated by means of a special form of the SAS `filename` statement in two ways:

- **DDE triplet**:  
  ```
  FILENAME fileref DDE '<server app>|<topic>!<item>';
  ```

- **DDE ‘doublet’**:  
  ```
  FILENAME fileref DDE '<server app>|system';
  ```
Using DDE

Prerequisite for using DDE: both client and server applications are up and running. A client/server DDE communication link is initiated by means of a special form of the SAS `filename` statement in two ways:

- **DDE triplet:**
  
  `FILENAME fileref DDE ’<server app>|<topic>!<item>’;`

- **DDE ‘doublet’:**
  
  `FILENAME fileref DDE ’<server app>|system’;`
DDE Triplet Example

- `<server app>` = excel
- `<topic>` = `[myXcel.xlsx]sheet1, [myXcel.xlsx]my data sheet`
- `<item>` = r1c1:r10c3
DDE Triplet Example

- `<server app> = excel`
- `<topic> = [myXcel.xlsx]sheet1, [myXcel.xlsx]my data sheet`
- `<item> = r1c1:r10c3`
DDE Triplet Example

- `<server app>` = excel
- `<topic>` = `[myXcel.xlsx]sheet1, [myXcel.xlsx]my data sheet`
- `<item>` = R1C1:R10C3
Dynamic Data Exchange

DDE Triplet Example

- `<server app>` = excel
- `<topic>` = `[myXcel.xlsx]sheet1, [myXcel.xlsx]my data sheet`
- `<item>` = `r1c1:r10c3`
Examples of Writing to Excel
Using DDE doublet

in the form of (in DATA _NULL_):

PUT '[[system-command]]';

- PUT '[[save.as("f:\myXcel.xlsx")]]'; — save excel
- PUT '[[format("Courier", 200, true, false, false, false, 3, false, false)]]'; — change font
  “font name”, font size, bold, italic, underline, strike-through, color, outline, shadow
- PUT '[[workbook.name("sheet1","my SAS data")]]'; — rename sheet
Examples of Writing to Excel
Using DDE doublet

in the form of (in DATA _NULL_):

PUT ‘[system-command]’;

- PUT ‘[save.as("f:\myXcel.xlsx")]' — save excel
- PUT ‘[format("Courier", 200, true, false, false, false, 3, false, false)]’ — change font
  “font name”, font size, bold, italic, underline, strike-through, color, outline, shadow
- PUT ‘[workbook.name("sheet1","my SAS data")]' — rename sheet
Examples of Writing to Excel
Using DDE doublet

in the form of (in DATA _NULL_):

```
PUT '[system-command]';
```

- PUT '[save.as("f:\myXcel.xlsx")]; — save excel
- PUT '[format("Courier", 200, true, false, false, false, 3, false, false)]]; — change font
  “font name”, font size, bold, italic, underline, strike-through, color, outline, shadow
- PUT '[workbook.name("sheet1","my SAS data")]; — rename sheet
Examples of Writing to Excel
Using DDE doublet

in the form of (in DATA _NULL_):

PUT '[system-command]';
- PUT '[save.as("f:\myXcel.xlsx")]' — save excel
- PUT '[format("Courier", 200, true, false, false, false, 3, false, false)]' — change font
“font name”, font size, bold, italic, underline, strike-through, color, outline, shadow
- PUT '[workbook.name("sheet1","my SAS data")]' — rename sheet