Auxiliary File Format Overview

- Has two types of “Sections”.
  - There is no limit to the number of sections in a file

SCRIPT Section

```
SCRIPT ScriptName1
{
script_statement_1;
    :
script_statement_n;
}
```

DATA Section

```
DATA DataName1(object_type, [list_of_fields], file_type_specifier)
{
data_list_1
    :
data_list_n
}
```

DATA Section

```
DATA DataName2(object_type, [list_of_fields], file_type_specifier)
{
data_list_1
    :
data_list_n
}
```

SCRIPT Section

```
SCRIPT ScriptName2
{
script_statement_1;
    :
script_statement_n;
}
```

©2008 PowerWorld Corporation

Auxiliary File DATA Sections

This was discussed in a previous section

- Start with the word DATA
- An optional data name may follow
  - For use with the LoadData Action
- Following this is a list of parameters enclosed in parenthesis
  - **`(object_type,[list_of_fields],file_type_specifier)`**
    - **`object_type`**
    - **`[list_of_fields]`**
    - **`file_type_specifier`**

```
DATA DataName1(object_type, [list_of_fields], file_type_specifier)
{
data_list_1
   :
data_list_n
}
```

©2008 PowerWorld Corporation

Auxiliary File SCRIPT Sections

- Start with the word SCRIPT
- An optional script name may follow
  - For use with the LoadScript Action
- Then a block of script actions follow enclosed in curly braces { }
- Each script statement must end in a semicolon ;
- All the script actions allowed will be covered in a later set of slides.

```
SCRIPT ScriptName1
{
script_statement_1;
   :
script_statement_n;
}
```

Modes and Submodes

- Simulator has an Edit Mode and a Run Mode
- For scripting, submodes must also be obeyed.
- Edit Mode Submode
  - Case Submode
- Run Mode Submodes
  - PowerFlow Submode
  - Contingency Submode
  - ATC Submode
  - Fault Submode
  - PV Submode
  - QV Submode

©2008 PowerWorld Corporation

Modes and Submodes

- Many script actions can only be performed in one submode.
  - This usually mimics the interaction you have with a particular dialog
- To switch between Submodes, use the EnterMode(**) script action.
- Slides at the end of this section will list the script commands that are available in each submode
  - For detailed information on the use of each script command, see the program help.
  - http://www.powerworld.com – Go to Downloads → Simulator Help Files → PowerWorld Simulator Auxiliary File Format

Script Command Execution Dialog

- To open go to the **Tools** or **Add Ons** ribbon tab and select **Script**
- Manually enter script commands
  - Useful for testing scripts
- Load auxiliary files
  - Validates and applies
- Validate auxiliary files
  - Receive messages in the message log if anything is incorrect in an auxiliary file before applying
- Quick Aux
  - Set up list of auxiliary files that are used frequently
  - Quickly reference a selected auxiliary file
  - Apply group of auxiliary files in a specified order

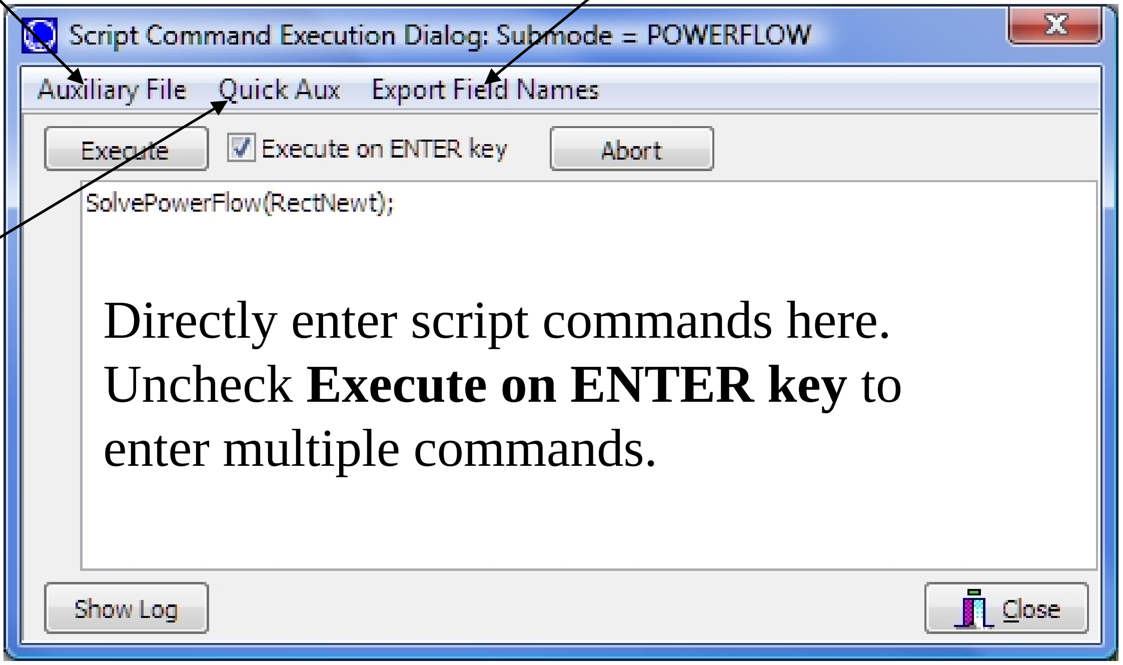
©2008 PowerWorld Corporation

Script Command Execution Dialog

Load or validate auxiliary files

Get list of fields for each object type

Define list of frequently used auxiliary files for easy access or apply list of auxiliary files in a specified order



```
SolvePowerFlow(RectNewt);
```

Directly enter script commands here. Uncheck **Execute on ENTER key** to enter multiple commands.

General Actions

- The following actions are available in all Modes and Submodes

```
RenameFile(“oldfilename”, “newfilename”);
CopyFile(“oldfilename”, “newfilename”);
DeleteFile(“filename”);
LoadAux(“filename”, CreateIfNotFound);
LoadScript(“filename”, ScriptName);
LoadData(“filename”, DataName, CreateIfNotFound);
SelectAll(objecttype, filter);
UnSelectAll(objecttype, filter);
Delete(objecttype, filter);
DeleteIncludingContents(objecttype,filter);
SaveData(“filename”, filetype, objecttype,
          [fieldlist], [subdatalist], filter, [SortFieldList]);
SaveDataWithExtra(“filename”, filetype, objecttype,
          [fieldlist], [subdatalist], filter, [SortFieldList],
          [Header_List], [Header_Value_List]);
```

General Actions

- The following actions are available in all Modes and Submodes

```
SetData(objecttype, [fieldlist], [valuelist], filter);
CreateData(objecttype, [fieldlist], [valuelist]);
WriteTextToFile(“filename”,”text…”);
SetCurrentDirectory(“fielddirectory”,CreateIfNotFound);
ExitProgram;
NewCase;
OpenCase(“filename”, OpenFileType);
SaveCase(“filename”, SaveFileType);
EnterMode(mode or submode);
OpenOneline(“filename”,”view”,FullScreen);
ExportOnelineAsShapeFile(“filename”, “oneline”,
                         “shapefileDOC”,UseLonLat);
LogClear;
LogSave(“filename”, AppendFile);
LogAdd(“string...”);
```

General Actions

- The following actions are available in all Modes and Submodes

```
LogAddDateTime("label", includedate, includetime, includemilliseconds);
CaseDescriptionClear;
CaseDescriptionSet(“text”,Append);
SaveYbusInMatlabFormat("filename", IncludeVoltages);
SaveJacobian("JacFileName", "JIDFileName",FileType, JacForm)
SetParticipationFactors(Method, ConstantValue, Object);
GenForceLDC_RCC(filter);
CalculateRXBGFromLengthConfigCondType(filter);
DirectionsAutoInsert(Source, Sink, DeleteExisting,
                      UseDisplayFilters, Start, Increment);
DetermineShortestPath([start], [end], BranchDistMeas, BranchFilter,
                      Filename);
DeterminePathDistance([start], BranchDistMeas, BranchFilter, BusField);
```

Edit Mode: Case Submode

- The following actions are only available in the Case Submode of Edit Mode.

```
Equivalence;
DeleteExternalSystem;
SaveExternalSystem(“Filename”, SaveFileType, WithTies);
Scale(scaletype, basedon, [parameters], ScaleMarker);
Move([elementA], [destination parameters]);
Combine([elementA], [elementB]);

InterfacesAutoInsert(Type, DeleteExisting, UseFilters,”Prefix”,Limits);
SplitBus([element], NewBusNumber, InsertBusTieLine, LineOpen);
MergeBuses([element], Filter);
TapTransmissionLine([element], PosAlongLine, NewBusNumber,
                      ShuntModel, TreatAsMSLine);
```

Run Mode

- The following actions are available during any of the Run mode Submodes.

```
Animate(DoAnimate);
CalculatePTDF([transactor seller], [transactor buyer], LinearMethod);
CalculatePTDFMultipleDirections(StoreForBranches, StoreForInterfaces,
                                LinearMethod);
CalculateLODF([BRANCH nearbusnum farbusnum ckt], LinearMethod);
CalculateTLR([flow element], direction, [transactor], LinearMethod);
CalculateTLRMultipleElement(TypeElement, WhichElement, direction,
                            [transactor], LinearMethod);
CalculateVoltSense([BUS num]);
CalculateFlowSense([flow element], FlowType);
CalculateLossSense (FunctionType);
CalculateVoltToTransferSense([transactor seller], [transactor buyer],
                             TransferType, TurnOffAVR);
CalculateVoltSelfSense(filter);
SetSensitivitiesAtOutOfServiceToClosest;
ZeroOutMismatches;
```

©2008 PowerWorld Corporation

Run Mode: Powerflow Submode

- The following actions are only available in the Powerflow Submode of Run Mode.

```
DoCTGAction([contingency action]);
SolvePowerFlow(SolMethod, “filename1”, “filename2”,
               CreateIfNotFound1, CreateIfNotFound2);
ResetToFlatStart(FlatVoltagesAngles, ShuntsToMax,
                 LTCsToMiddle, PSAnglesToMiddle);
SolvePrimalLP(“filename1”, “filename2”,
              CreateIfNotFound1, CreateIfNotFound2);
InitializePrimalLP(“filename1”, “filename2”,
                 CreateIfNotFound1, CreateIfNotFound2);
SolveSinglePrimalLPOuterLoop(“filename1, “filename2”,
                 CreateIfNotFound1, CreateIfNotFound2);
SolveFullSCOPF(BCMethod, “filename1”, “filename2”,
               CreateIfNotFound1, CreateIfNotFound2);
DiffFlowSetAsBase;
DiffFlowClearBase;
DiffFlowMode(diffmode);
OPFWriteResultsAndOptions(“filename”);
```

Run Mode: Contingency Submode

- The following actions are only available in the Contingency Submode of Run Mode.

```
CTGSolveAll;
CTGSolve(“ContingencyName”);
CTGSetAsReference;
CTGRestoreReference;
CTGProduceReport(“filename”);
CTGWriteResultsAndOptions(“filename”);
CTGAutoInsert;
CTGCalculateOTDF([transactor seller], [transactor buyer],LinearMethod);
```

Run Mode: ATC Submode

- The following actions are only available in the ATC Submode of Run Mode.

```
ATCDetermine([transactor seller], [transactor buyer],
             ApplyTransfer);
ATCRestoreInitialState;
ATCIncreaseTransferBy(amount);
ATCTakeMeToScenario(RL, G, I);
ATCDetermineFor(RL, G, I);
ATCWriteResultsAndOptions(“filename”);
ATCWriteToExcel(“worksheetname”);
```

Run Mode:
Fault Submode

- The following actions are only available in the Fault Submode of Run Mode.

```
Fault([Bus num, faulttype, R, X]);
Fault([BRANCH nearbusnum farbusnum ckt],
       faultlocation, faulttype, R, X]);
```

Run Mode:
PV Submode

- The following actions are only available in the PV Submode of Run Mode.

```
PVCreate(“name”, [element source], [element sink]);
PVSetSourceAndSink(“name”, [element source], [element sink]);
PVRun(“name”);
PVClearResults(“name”);
PVStartOver(“name”);
PVDestroy(“name”);
PVWriteResultsAndOptions(“filename);
RefineModel(objecttype, filter, Action, Tolerance);
```

Run Mode: QV Submode

- The following actions are only available in the QV Submode of Run Mode.

```
QVRun(“filename”, InErrorMakeBaseSolvable);
QVWriteResultsAndOptions(“filename”);
RefineModel(objecttype, filter, Action, Tolerance);
```

©2008 PowerWorld Corporation

Auxiliary Files and Difference Cases

- The Difference Case Tool may be used to build an auxiliary file that shows the topology difference between 2 cases
- Uses
  - Document topology changes in cases
  - Modify a case with Simulator GUI and capture changes in auxiliary file; the auxiliary file can be run on a different copy of the Base Case to produce the Changed Case

Auxiliary Files and Difference Cases

- Example
  - Open “Case A” and choose Tools ➔ Difference Flows ➔ “Set Present Case as Base Case”
  - Open “Case B” and choose Tools ➔ Difference Flows ➔ “Present Topological Differences From Base Case”

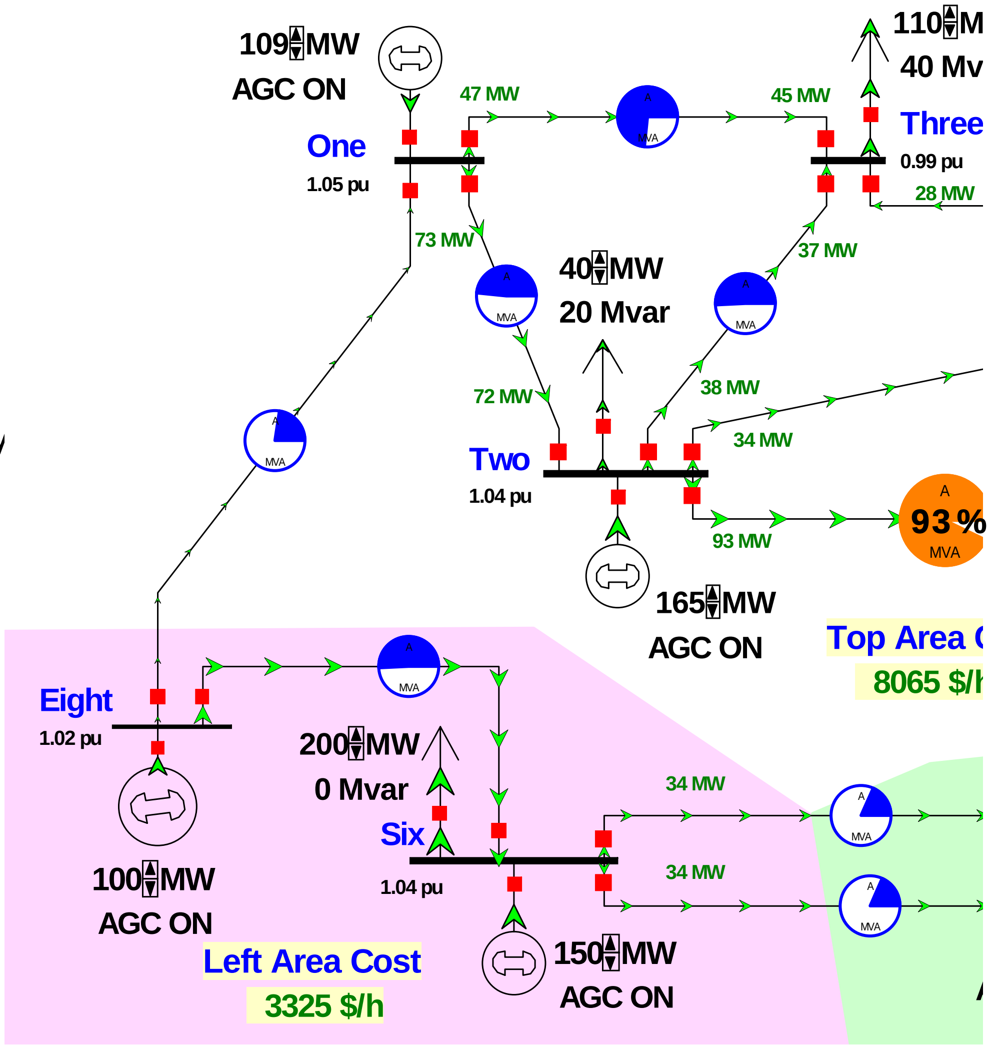
©2008 PowerWorld Corporation

Use Difference Cases

- First, save new items (“Only Elements Added”) to auxiliary file
- Then save removed items (“Only Elements Removed”) items: append to the same auxiliary file if desired
- Add a script statement at the beginning of the newly created file to switch to edit mode so that objects may be added and deleted

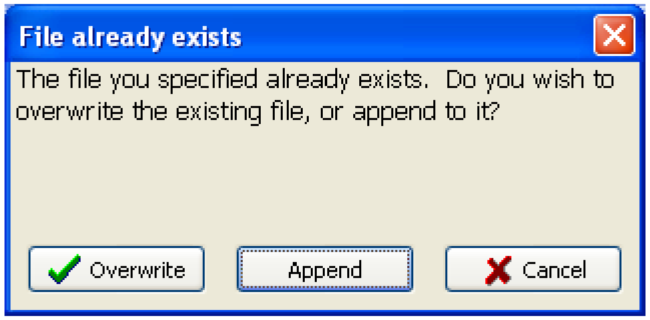
Example

- Open B7FLAT.pwb
- Set Present Case as Base Case
- Add new Bus 8, new Gen at Bus 8, and nev Branches connecting Bus 8 to Bus 1 and Bus 6
- Remove Branch between Bus 2 and Bus 6



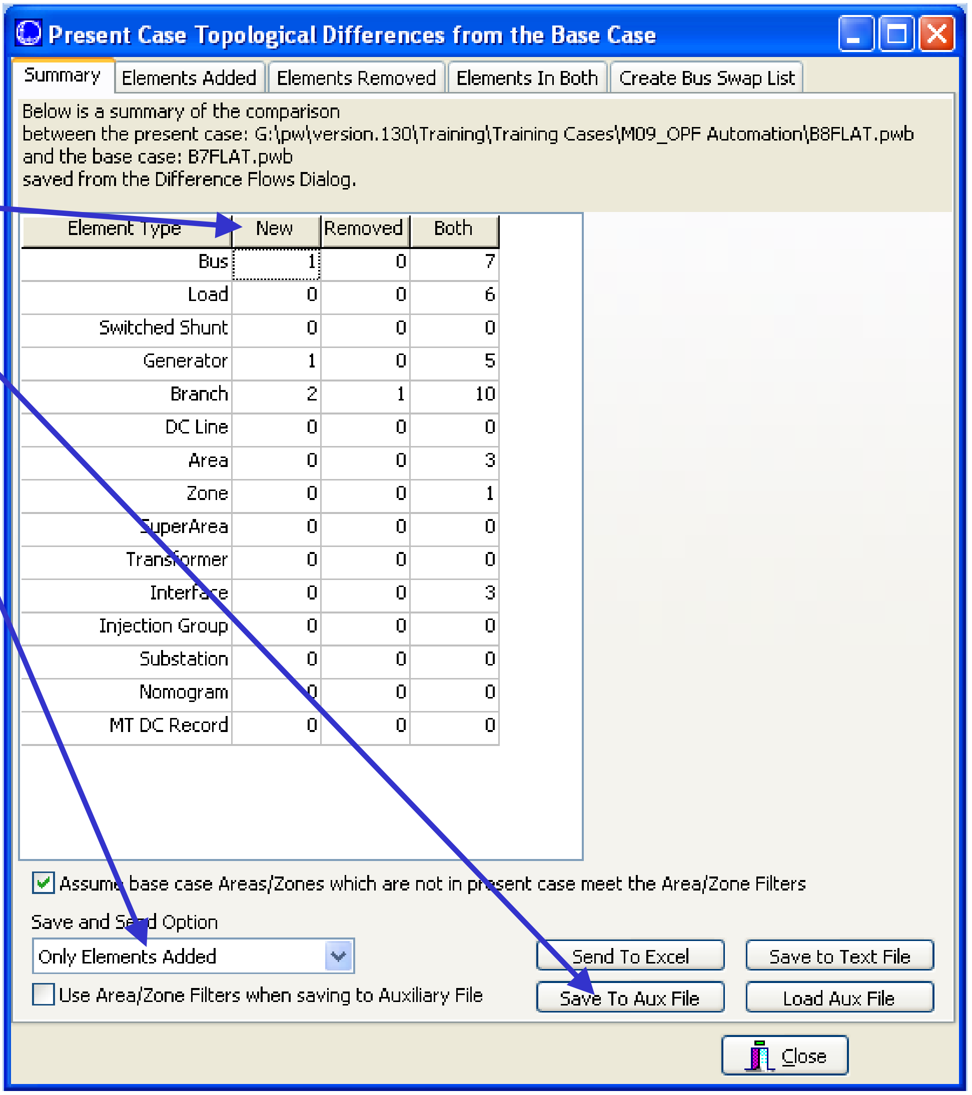
Present Topological Differences from Base Case

Save New items to Aux File, then Save Removed items to same Aux File (append)



File already exists

The file you specified already exists. Do you wish to overwrite the existing file, or append to it?



Present Case Topological Differences from the Base Case

Below is a summary of the comparison
between the present case: G:\pw\version.130\Training\Training Cases\M09_OPF Automation\B8FLAT.pwb
and the base case: B7FLAT.pwb
saved from the Difference Flows Dialog.

Element Type	New	Removed	Both
Bus	1	0	7
Load	0	0	6
Switched Shunt	0	0	0
Generator	1	0	5
Branch	2	1	10
DC Line	0	0	0
Area	0	0	3
Zone	0	0	1
SuperArea	0	0	0
Transformer	0	0	0
Interface	0	0	3
Injection Group	0	0	0
Substation	0	0	0
Nomogram	0	0	0
MT DC Record	0	0	0

☑ Assume base case Areas/Zones which are not in present case meet the Area/Zone Filters

Save and Send Option: Only Elements Added

☐ Use Area/Zone Filters when saving to Auxiliary File

Edit the Resulting Aux File

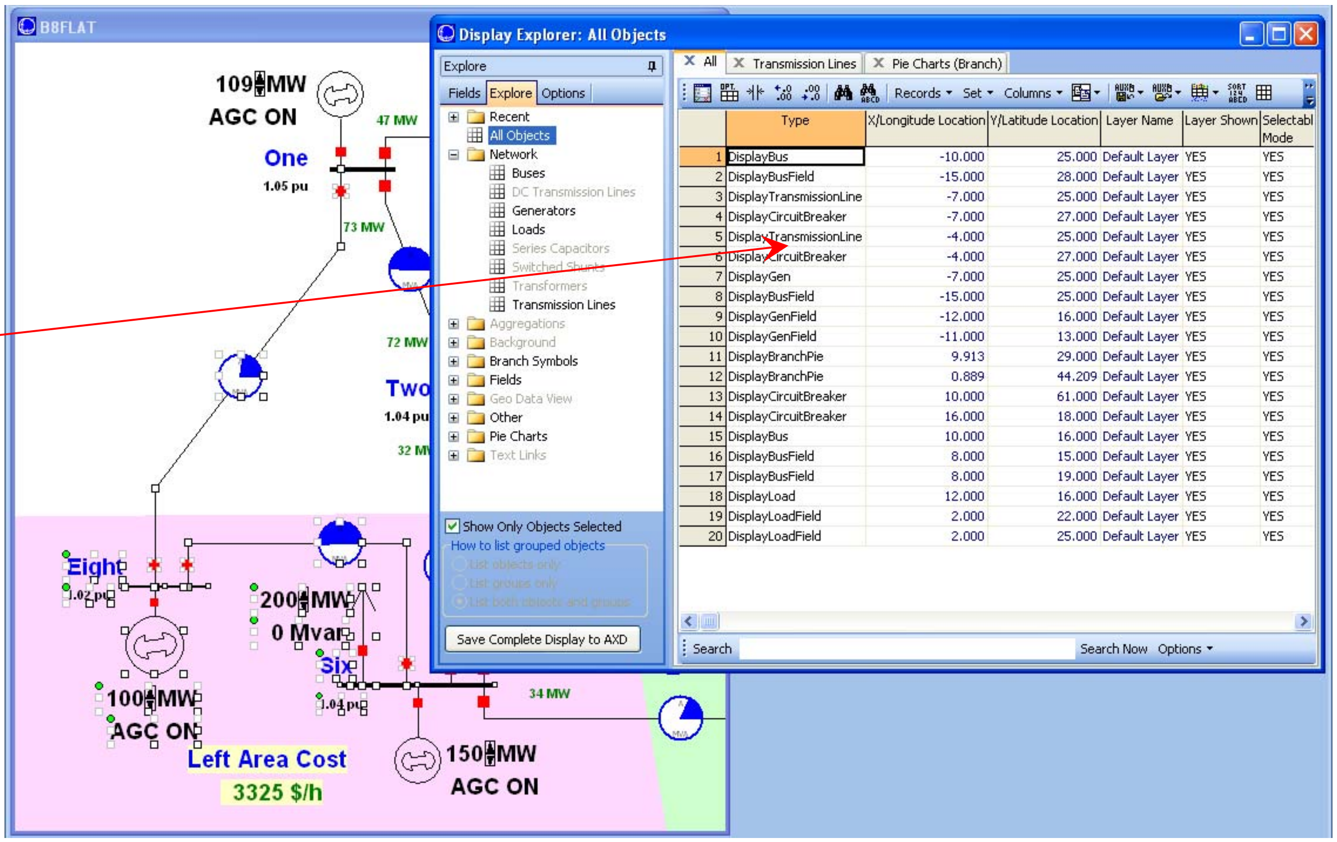
- Optionally add SCRIPT statement to change to EDIT mode to ensure that objects may be created and removed
- Then re-open B7FLAT.pwb and load the aux file

```
SCRIPT
{
EnterMode(EDIT);
}

//--------------------------------------------------------------------------------
// THE FOLLOWING SECTION CONTAINS THE ELEMENTS ADDED IN PRESENT CASE
//--------------------------------------------------------------------------------
DATA (BUS,
[BusNum,BusName,BusNomVolt,BusPUVolt,BusAngle,BusG:1,BusB:1,AreaNum,ZoneNum,
   SubNum,BusSlack,BusStatus,BusNum:1,BusName_NomVolt:1,BusLongName,BusKVVolt,

.
.
.
```

Save New and Modified Display Objects to AXD File

- Select objects on One-line
- Onelines ➔ List Display ➔ Only Selected Display Objects…

Right Click ➔
Save As ➔
Display
Auxiliary File…



	Type	X/Longitude Location	Y/Latitude Location	Layer Name	Layer Shown	Selectabl Mode
1	DisplayBus	-10.000	25.000	Default Layer	YES	YES
2	DisplayBusField	-15.000	28.000	Default Layer	YES	YES
3	DisplayTransmissionLine	-7.000	25.000	Default Layer	YES	YES
4	DisplayCircuitBreaker	-7.000	27.000	Default Layer	YES	YES
5	DisplayTransmissionLine	-4.000	25.000	Default Layer	YES	YES
6	DisplayCircuitBreaker	-4.000	27.000	Default Layer	YES	YES
7	DisplayGen	-7.000	25.000	Default Layer	YES	YES
8	DisplayBusField	-15.000	25.000	Default Layer	YES	YES
9	DisplayGenField	-12.000	16.000	Default Layer	YES	YES
10	DisplayGenField	-11.000	13.000	Default Layer	YES	YES
11	DisplayBranchPie	9.913	29.000	Default Layer	YES	YES
12	DisplayBranchPie	0.889	44.209	Default Layer	YES	YES
13	DisplayCircuitBreaker	10.000	61.000	Default Layer	YES	YES
14	DisplayCircuitBreaker	16.000	18.000	Default Layer	YES	YES
15	DisplayBus	10.000	16.000	Default Layer	YES	YES
16	DisplayBusField	8.000	15.000	Default Layer	YES	YES
17	DisplayBusField	8.000	19.000	Default Layer	YES	YES
18	DisplayLoad	12.000	16.000	Default Layer	YES	YES
19	DisplayLoadField	2.000	22.000	Default Layer	YES	YES
20	DisplayLoadField	2.000	25.000	Default Layer	YES	YES