

Autodesk Architectural Desktop™ 2007

Feature Based Primer for Basic and Intermediate Classes Part 3b of 4 = Annotation

(ok now there are 5 total parts part 3 is in 2 parts a and b)

...Or...

David's Brain Noise for ADT 2007

Have I said yet,

Use the right mouse button

or

Watch the command line

?

Table of Contents

Annotation and 2D objects	2
DesignCenter Annotation	4
Annotation Scaling.....	5
Scheduling Overview – Terminology.....	6
The Simple Explanation of Schedules:.....	6
Schedules Overview	7
Schedules Functions and Terminology:	8
The Workings	10
Modifying Schedule Information	12
Callouts (sections elevations).....	17
Details:.....	18
Keynotes:.....	23

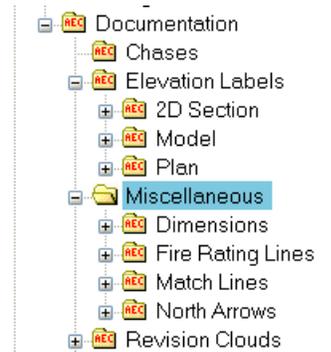
Annotation and 2D objects

Annotation functions of ADT are undergoing transition. Historically, the DesignCenter was the location of not only the multiview blocks, but much of the documentation tools as well. The Documentation folder of the DesignCenter was the location for tags, elevation marks and other pieces of annotation. Each of the pieces of documentation content started with the ACEDB dictionary explained in the environment section. It appears this functionality of the DesignCenter is slowly being phased out. With each release more of the documentation functions accessed from the DesignCenter migrate to be tool specific and rely less on the function of the Dictionary and custom commands written for the DesignCenter use. With 2005, all the callouts (section, elevation and detail marks) integrated into specific tools that rely more on the tool properties than the dictionary for their functions. With 2006, the break marks and schedule tags functions have been rewritten to take advantage of the tool functionality over the dictionary and DesignCenter functionality.

In general, annotation tools are designed to be used in a view drawing if you are using the project navigator. The most of the tools will work in you are not using the project navigator, but many of them are designed specifically for use with the project navigator

- DesignCenter Documentation objects
 - Revision clouds
 - Break marks
 - Elevation tags
 - Chase Blocks
- Dimensions
- Schedules and Tags
- Elevations and Sections
 - Section and Elevation Callouts
 - Section and Elevation Lines
 - Section and Elevation Objects
 - Section and Elevation Styles
- 3D section/elevations = Live Section
- Section/elevation marks
- Keynotes
 - Reference Keynotes
 - Sheet Keynotes

DesignCenter Annotation



Break marks

Break Marks were rewritten for 2006. The new break mark tool relies on the tool properties rather than a custom command. This tool has been removed from the DesignCenter folders.

Revision clouds

Less than documented, or rather I think this was documented in one of the readme's. To keep the settings in the cloud you just drew, hit 'enter' to repeat the last command. To create settings for the current session, use the command line options to set your settings. These should be saved and reused the next time you use the AEC content. I have read that these variables are stored in the drawing, and read that they are not stored in the drawing. I have not had time to check it out, I suspect in ADT1 they were not, and in ADT2 they are. I have not checked ADT3 yet.

Elevation 'tags'

Watch the use of these – they are put in folders 2D/3D for a purpose – they are MultiView blocks that read their Z value into the tag

Chase Blocks

This will place an MV block in the drawing and prompt for an object to interfere with. The function is written so that you can automatically assign interference to the block you are placing.

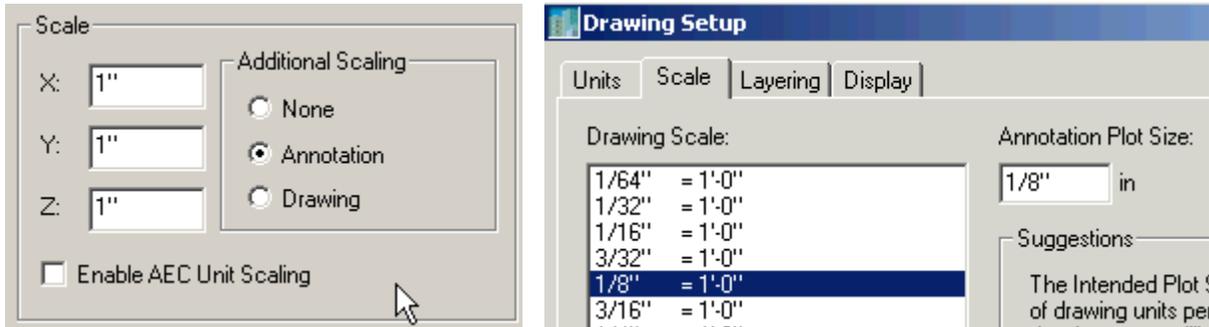
Adding

Not much to add here other than watch the command line. Each of these tools from the documentation folder utilizes the custom command string, this will show up

Annotation Scaling

Content Scaling

In the content wizard you are asked in you want additional scaling for your content. You can choose from None, Annotation or Drawing. You are also given the opportunity to apply an absolute override scale to the content.



None: Just leaves the content as is – MvBlocks and such

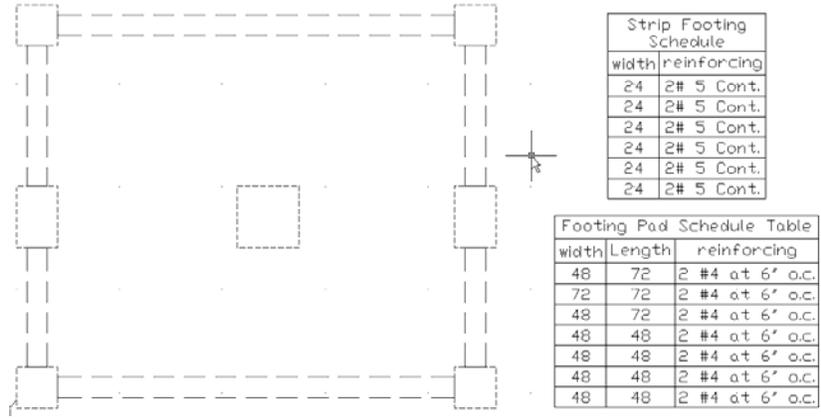
Drawing and Annotation: These options refer to the drawing setup (Desktop>Drawing setup >Scale).

They work in tandem like this: If you create your content annotation text 1" tall

Set the drawing scale to what you will print the sheet at, for example, 1/4". The drawing scale will always be applied to content created with the annotation scaling flag set. At this point the text is now multiplied by what ever factor this is (48) to be plotted at 1/4" so your text is now 48 inches tall. If you just use Drawing scaling, and not Annotation that is the end of the story. If you select Annotation scaling, the content (text is multiplied by the annotation scale - let's use 3/32" in this example. So, the 48" tall text is now multiplied by 3/32. $3/32 \times 48 = 4.5$ the proper size for text to be plotted at 1/4"=1'-0" to appear 3/32" high.

Scheduling Overview – Terminology

Scheduling is another one of those automatic functions in Architectural Desktop that if it does things the way you want, life is good. If it doesn't do schedules the way your office does, it will take some setup at the beginning, but should work fine for you after you apply the effort to customize this area of the software to make it work for you. Schedules are a very powerful function of ADT, and very customizable.



Custom Footing Schedules using Polylines and Mass Elements

The Simple Explanation of Schedules:

Schedules have the magic automatic feel of a good demo. You place a door, you tag the door and then you add a schedule table to the drawing. The door shows up in the schedule.

Schedules Overview

Basics

Schedule Hierarchy

1. Schedule Data Format
2. Property Set Definitions
3. Schedule Table Style

Size is determined by table style settings in conjunction with Drawing Scale settings – changed once in the drawing with the properties.

'Tagging' an object automatically attaches the Property Set Definitions to the object tagged, but you may do this manually. (you do not ever need a tag in any drawing to take advantage of automatic scheduling.

Schedule information is held in both the style and the object

Schedule information is held in both the style and the object – but not the schedule. The schedule just displays the information held elsewhere.

Edit the schedule by editing the 'extended data' associated with the AEC object (door) or it's style

Right click edit cell to edit the values in the schedule directly.

It is useful to keep current schedule table styles in your working drawings even if your actual schedule is in a different sheet – this way you can make a 'temporary' schedule in the same drawing as the actual objects to trouble shoot the schedule.

Project based schedules rely on the room. The room is a space object

Etcetera

Mysterious

Project based tags can read objects from an xrefed element (project navigator) – but you must attach the property set manually.

(tagging a door in a construct xref will automatically apply the property set through the xref onto the object in the construct)

Property sets from one drawing to another

Explicitly copying a Property Set from one drawing to another can only be done using the Style Manager.

If a Property Set is copied from a source drawing into a target drawing that already contains a Property Set with the same name, the normal options of Leave Existing, Overwrite, and Rename to Unique are presented. The Overwrite option will replace the Property Set in the target drawing with the copied one, as expected, but there is a potential for a loss of data.

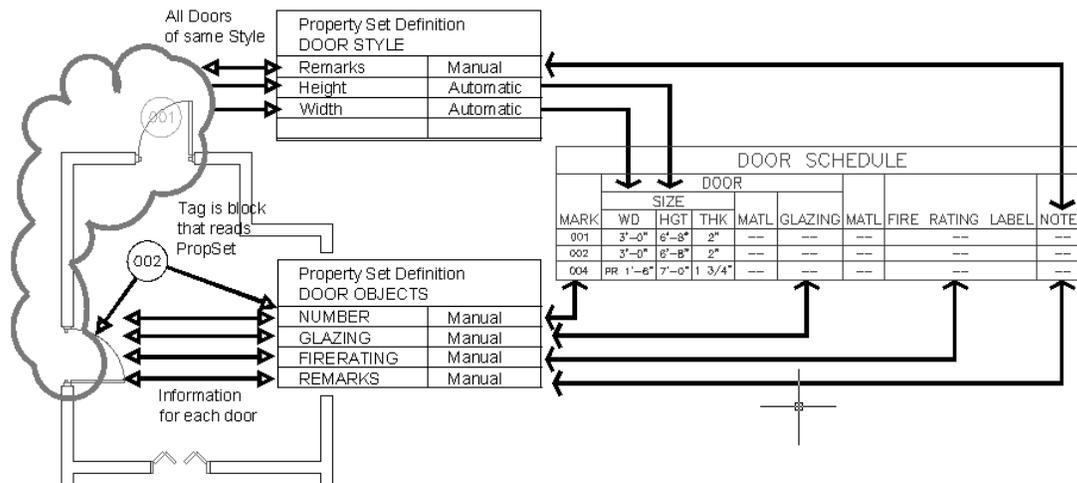
For example: a drawing contains a Property Set that has defined a number of manual properties. This Property Set is attached to a number of doors with its manual property values set, and is used in a Door Schedule Table. If this Property Set is overwritten with another one that has not defined the same manual properties, values associated with the removed properties will be lost. In addition, any columns in the Schedule Table that referenced the removed manual properties will also be removed. Changes to the schedule tables that result from changes to property sets will be displayed the next time the tables are updated manually or automatically.

Schedules Functions and Terminology:

Schedules Functions

Schedules, like Display Representations, have a built in nested complexity. There are styles within styles within styles to get the final Schedule table in your drawing. With Display Configurations, Display Reps are gathered into Sets, which are gathered into Configurations. With Schedules, a format style is nested in a Property Set nested in a Schedule Table Style.

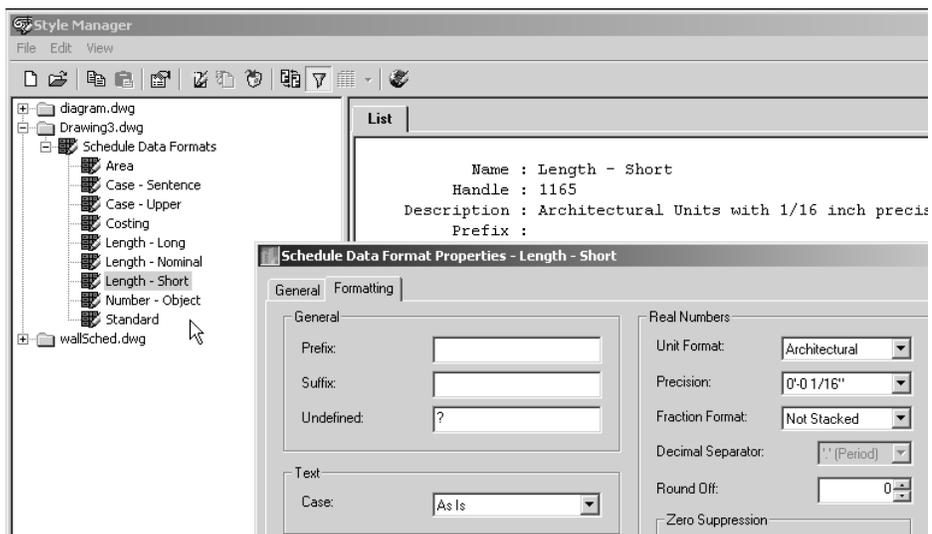
Schedules like Mass Groups are a 'Display Object'. They do not have any meaning by themselves; they only are there to show you information about what is attached to them. However, where Mass groups have a direct attachment to another object, Schedules have a more obscure attachment scheme. Instead of having an attachment directly to an object, the schedule is attached to a Property Set that is attached to an object.



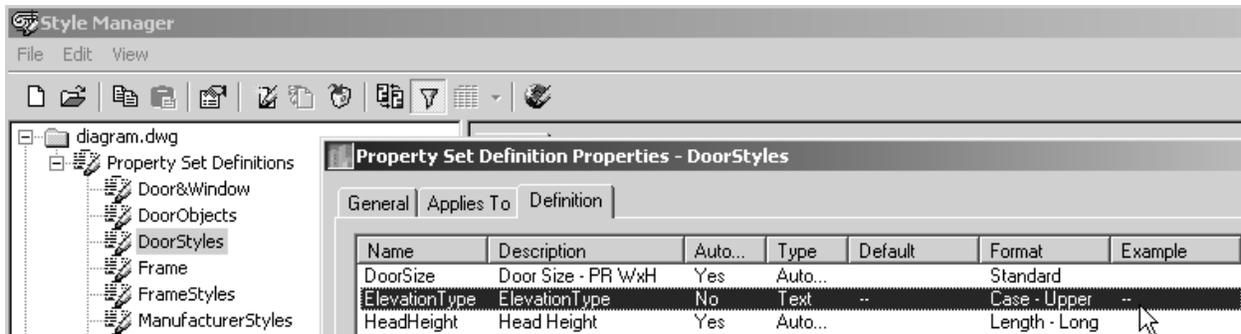
Schedule Data Formats:

Schedule Data Formats hold the variables for the format of the schedule table entries i.e. show me the length in 12' - 9" rather than 118 (inches). The templates have some of the common formats already in place.

Each Property within a PropSet will get assigned a default data format. You can override this data format when you create the columns in the schedule table style.



Property Set Definitions:



The working part of a schedule is the Property Set Definition or PropSetDef for short. They are a group of individual properties of an object or style. Individual properties may either be automatic or manual.

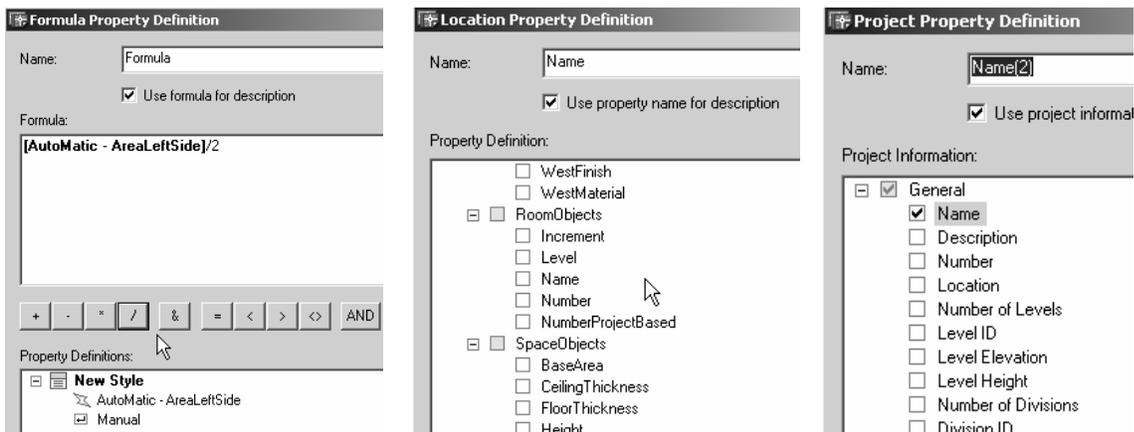
Automatic properties are read directly from the object itself i.e. the width or height of a door. Manual properties rely on you to supply their values i.e. the manufacturer or remarks for a given door.

- Within ADT, Property Set Definitions and the term Schedule Data are used synonymously.
- The PropSet is invisible.
- PropSets are assigned either to a style or to an individual object.

Automatic Properties may not be edited in the schedule. You must change the object itself to change the information in the schedule.

Introduced in ADT2004/5, you have many more types of properties available for you to use – see Schedule Addenda #1 Scott Arvins Brain dump for an extended discussion of the 04/05/06 property sets.

Formula, Location and Project Information



These types of properties allow you to combine and perform calculations on properties based on the project data.

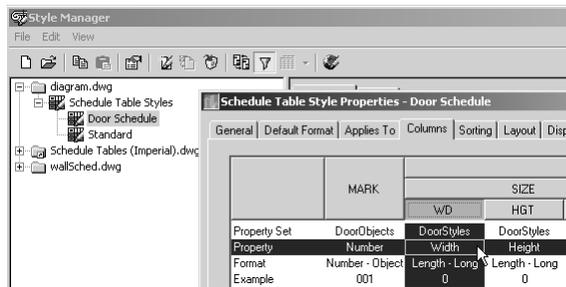
Circular Properties

Classification, Material

You can add a property that points to a different property in another set.

In order to establish a Classification or Material property, you must have defined a Property set that applies to a classification style or a material style first.

Schedule Tables



A schedule table style is just the formatting of the Property Sets in the drawing. Each Column in the Schedule displays an individual property in a property set.

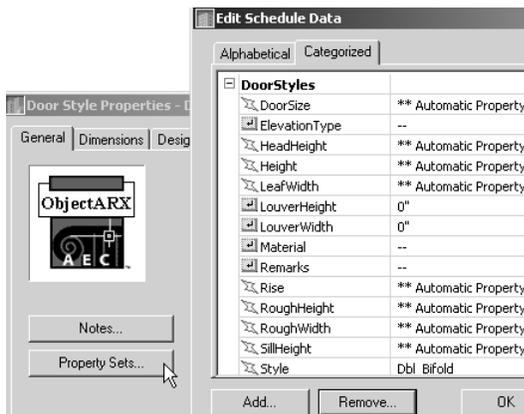
The column “WD” looks at the individual Property “Width” stored in the DoorStyles PropertySetDefinition and uses the Schedule Data Format Length-Long

The Workings

When you place a schedule in a drawing, it looks for the information it needs to fill itself in. This information is stored in the PropSets attached to the styles and objects in the drawing (see diagram first page). If this link is broken (the object or style exists without a PropSet attached) then the schedule will fill up with questions marks. The Property Sets must be attached to the objects and styles for the schedule to fill itself out.

Attaching Schedule data (Property Set Definitions):

PropSets are assigned either to a style or to an individual object. The process of getting the schedule information attached to a style differs from the process of attaching it to an object.



Attaching property sets to Styles

The property set must be placed within the style definition.

Edit the Style, and on the General tab there is a Property Sets button that open up the “Edit Schedule Data” dialog Box. Here you can ADD (attach) property sets or remove them from the style definition.

Note: Almost all of the door styles that exist in the templates already have the schedule data attached. If you create a new door style, (or use one created in ADT 2) you will have to go through this process to assign a PropSet to the style.

Attaching property sets to individual doors (objects):

The schedule data is either attached manually in the same fashion as noted above for the style – Right Click>Edit Schedule Data, pick the Add button and select the property sets to apply – or you can tag the door. The process of adding a Tag to a door adds the schedule data as well as the tag. Which Property Sets are added is held within the definition of the tag itself.

Tags:

A tag is a special form of MultiView Block. Held within the Definition of the MvBlock is a standard AutoCAD block. The AutoCAD Block has attributes within it. The attributes have a special format which is <PropertySetName>:<PropertyName>. When the tag is placed from the DesignCenter, it launches a command AecAnnoScheduleTagAdd. This command does a number of things.

- Anchors the tag to a specific door
- Assigns a property set to the door if none exist i.e. tagging a door, automatically does what is discussed in the above section “Attaching property sets to individual doors (objects)”.

Annotation

- Creates a link between the attribute in the tag block and the Property Set assigned to the door. This link enables the tag to read the property set and use the schedule data to fill itself in. Tags can be created to display either automatic or manual types of properties.
- If the PropSetDef the tag needs does not exist in the drawing, it will look to C:\Documents and Settings\All Users\Application Data\Autodesk\Autodesk Architectural Desktop 2004\R16.0\enu\Shared Content 4 for the PropSet and import it from there.

For a door tag, the attributes read DOOROBJECTS:NUMBER. So when the tag is placed, it at least attaches the property set DoorObjects to the door, then it looks within DoorObjects PropSet for the single property Number, and fills in this value into the block.

See the separate Content chapter for more on tags and how to create them.

Modifying Schedule Information

Unless you have modified your right click menu, most of what you need to edit a schedule table are available here. Select the schedule table and right click. Here you can edit individual cells, show the object in the drawing the cell is referring to and more. However; often schedule sheets are in a different drawing than the objects themselves. Multiple floors are externally referenced into the Schedules.dwg to allow the schedules to show all the building doors for example. This creates an Xref barrier between the schedule and the information (Property sets assigned to the door) the schedule is displaying. Many (most) of the right click functions of the schedule do not work through this xref barrier. I work around this by having the same schedule styles in my 'model' drawings as I have in the schedule sheet. If there is a problem with a door on the second floor, I open the second floor plan 'model' dwg, place a temporary door schedule in the drawing, and then use this right click functions on the schedule to:

Edit Table Cell: Individual value

Pick on a single cell in the schedule to edit its value. If the property of that cell is a manual property you are allowed to change the value for that instance. You are changing the value of a property in a property set that is attached to the object (door). The new value is stored in the 'schedule data' in the 2nd floor plan.dwg and will appear when you save and open the schedules drawing.

Edit Table Cell: All values in a column

Pick on a line between two cells to edit this value for all instances in the drawing.

Selection>show

Pick a cell in the schedule. The associated object in the drawing will be highlighted.

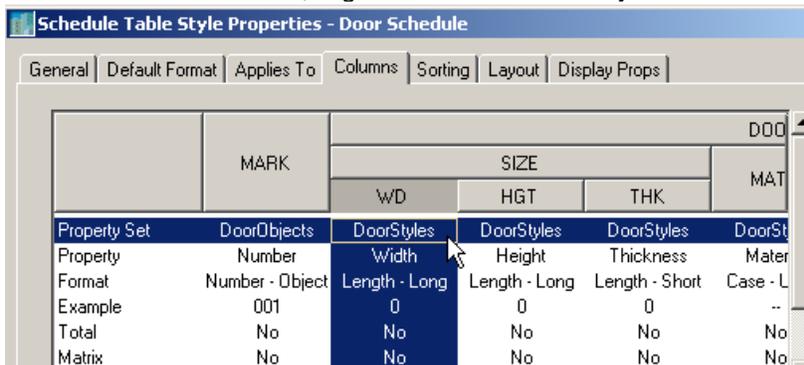
When I am done making my changes in 2nd floor.dwg, I erase the temporary schedule.

Remember the schedule is just a display object showing information in the drawing. The information is displayed and accessed by the schedule, but not stored in the schedule.

Of course, to change the schedule information, you can always go directly to the source. Select the offending door and right click> Edit Schedule Data. Or edit style and then on the general tab of the style, select Property sets.

How do I know if the schedule information is held in the style or in the object itself?

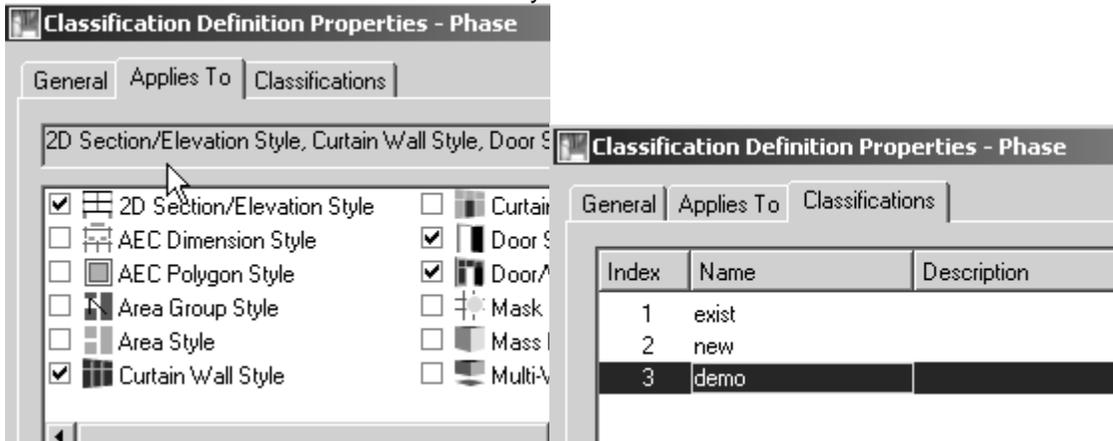
Select the schedule table, Right Click> Edit Table Style. Pick the Columns tab.



Every column in the table is a single property in a property set. You can tell if the value is from a style or object from the name of the Property Set used by that column – in this case the Mark is attached to the door itself, while the width, height and thickness are held by the door style.

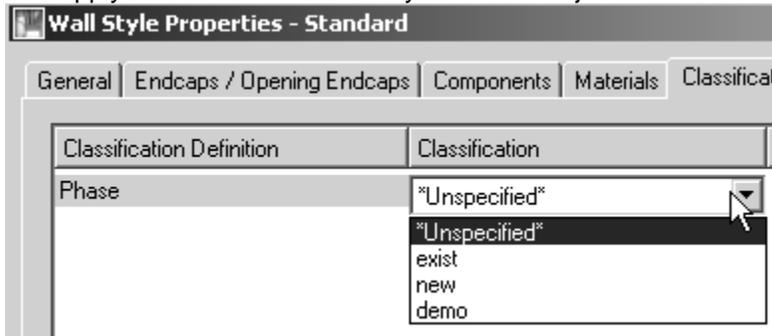
Classifications

Classifications added to ADT04/05/06 allow you to do many thing, while not an in-depth coverage by any means this section is an overview of how they work.



Classifications allow you to add additional information to the styles in the drawing.

You apply a classification at the style level for objects

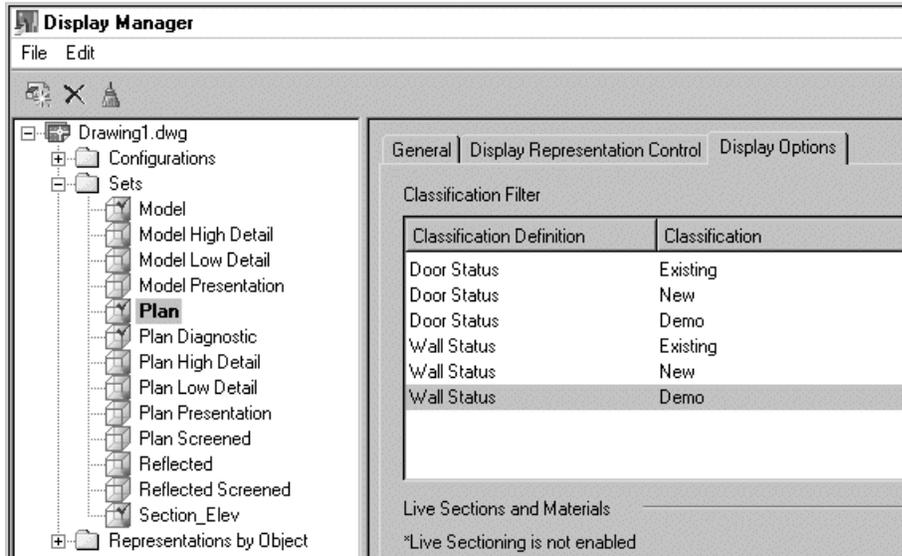


There is a drawing that has Uniformat classifications defined as classification styles for you to use.

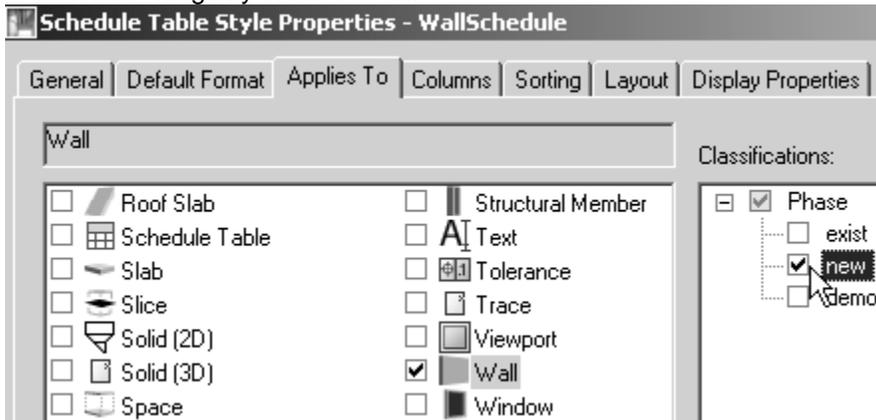


This information can be used then is several places
To Display things by classification

Hiding objects based on their classification

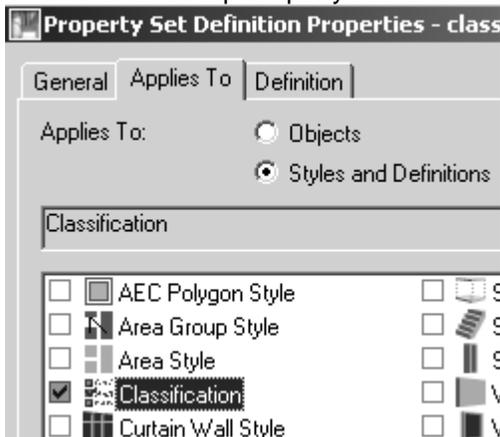


to schedule things by classification:



A Schedule set up with a classification will only show the walls of styles that have the **new** classification attached to their style definitions.

You can also set up Property set definitions that apply to classifications



by applying a property set definition to a classification, you can be sure that the property set will be available to all styles that use that classification.

Callouts (sections elevations)

OK there is not much here but a few notes and reminders of things I said in class....I cover this in class twice – and there are tutorial that cover the topic fairly well...

General Process:

Create model.dwg (project navigator view)

Add section lines or elevation marks

If the result is good enough – fine use it

If you need substantial additions/modifications, then

Put the elevation on a no-plot layer

Polyline the elevation

Move the elevation somewhere else (300 feet to the right)

Associatively hatch the polylines

Be careful with the order that you create the callouts

Be aware that the callouts tools have different section/elevation styles associated with them

Details:

- Setting up the drawing for a detail – remember that the drawing setup scale will determine the annotation and other scale issues including the keynote – set your drawing scale first
- Drawing a detail using the component database
- Editing materials
- Annotation
- Detail Access Database Library
 - Adding new sizes using the “specify” option
 - Adding new sizes by editing the database
 - Adding a new material using blocks
- Detail Management

There are a few key things that we want to illustrate up front to make creating details a little more understandable, and then we will move on.

Four general material placement tools (or jigs) that Details uses again and again to draw material components are:

- Stamping
Used to place a copy of an element in the drawing. It has options for flipping, moving, rotation
- Array
Start like the stamp jig, but then allows for an array of elements in the drawing
- Linear
Draws a line, and an offset side, and uses these to draw a more complex material
- Drawn
You draw the material using a variation of the polyline command.

There are 12 total jigs but the above 4 are the major used

Even if the preview picture is 3D the shipped content is only 2D.

3D Content might be possible for 3rd party developers, i.e. Glass blocks.

Even if it looks like we have parametric objects, the inserted parts are only AutoCad objects.

Detail Components are inserted with “Color By Layer”

Paths to useful items

Help file for detailer – if you are making your own database and xml

C:\Program Files\\Help\dtlauto-reference.chm

Parts of the Detailer:

Database

This is an Access database

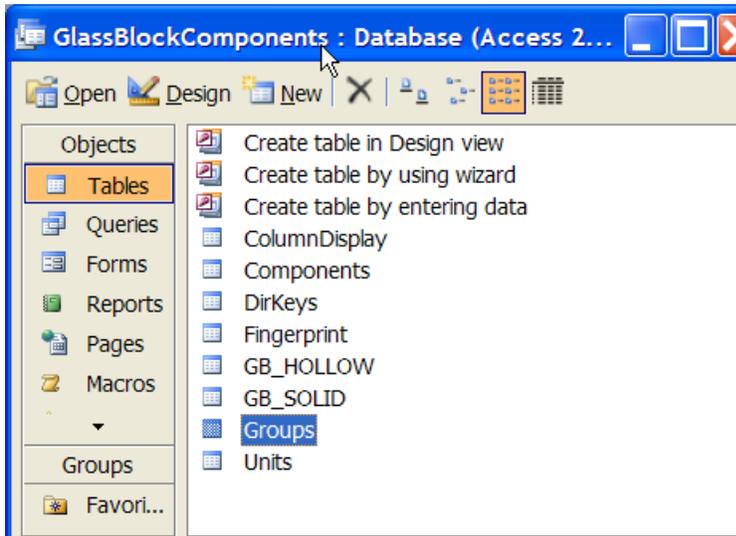
There is a sample database located in

C:\Program Files\\Sample\Sample Details Database - Glass Block

There are details and keynote database template in

C:\Documents and Settings\All Users\Application Data\Autodesk\\enu\Template\Details and Keynotes

This glass block sample database is the most easy to dissect:



The column table will list what columns show up in the detail component manager

ID	TableID	GlobalName	LocalName	Hidden	MODIFIED
1		DESCRIPTION	Description	<input type="checkbox"/>	<input type="checkbox"/>
2		HEIGHT	Height	<input type="checkbox"/>	<input type="checkbox"/>
3		WIDTH	Width	<input type="checkbox"/>	<input type="checkbox"/>
4		THICKNESS	Thickness	<input type="checkbox"/>	<input type="checkbox"/>
5		ID	ID	<input type="checkbox"/>	<input type="checkbox"/>
6		MODIFIED	Modified	<input type="checkbox"/>	<input type="checkbox"/>
7		PICTURE	Picture File	<input type="checkbox"/>	<input type="checkbox"/>
8		KEYNOTE	Keynote	<input type="checkbox"/>	<input type="checkbox"/>
0	0			<input type="checkbox"/>	<input type="checkbox"/>

The component table shows the values for each of the individual components

ID	Group Name	Name	TableName	Keynote	SubComp	RecipeSpec	SearchKey	Units	MODIFIED	PICTURE
1	Glass Block	Hollow core	GB_HOLLOW	04270		GlassBlockHollow.xml	glass block hollow	in	<input type="checkbox"/>	glassblockhollow.png
2	Glass Block	Solid core	GB_SOLID	04270		GlassBlockSolid.xml	glass block solid	in	<input type="checkbox"/>	glassblocksolid.png
0	\ - *								<input type="checkbox"/>	

The most important thing here is the recipeSpec which points to an XML file

DirKeys

ID	Key	Value	Modified
1	ZM_GB	Glass Block	<input type="checkbox"/>
0			<input type="checkbox"/>

To be quite honest I have no clue what this does

Fingerprint

Fingerprint : Table						
	GUID	Author	ID	Type	DisplayName	Description
▶	{AACAD816-B3B7-4700-98C8-37373458538C}	Autodesk, Inc.	1	Component	Glass Block Cor	AEC Detail API sample database
*	{AACAD817-B3B7-4700-98C8-37373458538C}	Autodesk, Inc.	2	Keynote		
*			0			

The GUID for the detail database is blank in the provided template and will be filled in when you link it on the options page in ADT – likewise with the keynote database – once this is filled in for the keynote copy it here.

Groups:

Groups : Table							
	ID	Parent	Group	PathKey	SearchKey	MODIFIED	PICTURE
▶	0 \	\				<input type="checkbox"/>	
*	0 \					<input type="checkbox"/>	

This works just like the keynote groups – see that handout

Sample Size

SampleSize : Table							
	DESCRIPTION	SIZE_DATA_1	SIZE_DATA_2	ID	MODIFIED	PICTURE	KEYNOTE
▶	Textual Description of item	1	text value	1	0		
*							

Mostly if you are creating a detail component it will be block based so this will not apply

Units

Units : Table			
	ID	LookupName	DisplayName
▶	1	in	Inches
	2	mm	Millimeters
*	(AutoNumber)		

Just like it says....used in components table

Jigs

A "Jig" is a programming construct that defines how the user adds objects to the drawing. You can think of it as an "Add" command that gives visual feedback as you are specifying what the object looks like. For instance, the simple Arc Jig in AutoCAD dynamically sketches out the potential arc while you input various values.

Cooking with xml

For each component there is an xml that contains a recipe

From the help file:

Recipe Specification

A Recipe Specification (RecipeSpec) is an XML file that describes the different Recipes that can be used to create a particular component in AutoCAD. For each Recipe, it identifies the types of controls (prompts) that should appear in the Properties palette and specifies how to call the function that implements the given Recipe. The RecipeSpec also provides view-specific information such as the layers and hatch patterns to use when creating the component.

The following is a copy of the xml for the glass block

```

<?xml version="1.0" encoding="UTF-8" ?>
- <AecRecipeSpec defaultUnit="in">
- <ControlDefs>
  - <ComboBox idStr="jig">
    <Label>Placement</Label>
    - <ComboBoxValue idStr="jigStamp">
      <Label>Single</Label>
    </ComboBoxValue>
    - <ComboBoxValue idStr="jigArray">
      <Label>Many</Label>
    </ComboBoxValue>
  </ComboBox>
</ControlDefs>
- <Recipe idStr="top">
  <Label>Top</Label>
  <Description>Top View</Description>
  - <Implementation>
    <VBA projectName="GlassBlock.dvb" macroName="GlassBlockTopView" />
  </Implementation>
  - <Controls>
    <ControlRef idRef="jig" />
  </Controls>
  - <Data>
    - <Item idStr="unit">
      - <Params>
        <Type>Hollow</Type>
      </Params>
    </Item>
  </Data>
</Recipe>
- <Recipe idStr="elevation">
  <Label>Elevation</Label>
  <Description>Elevation View</Description>
  - <Implementation>
    <VBA projectName="GlassBlock.dvb" macroName="GlassBlockFrontView" />
  </Implementation>
  - <Controls>
    <ControlRef idRef="jig" />
  </Controls>
  - <Data>
    - <Item idStr="unit">

```

Below is the complete xml for the wood millwork

Annotation

```
<?xml version="1.0" encoding="UTF-8" ?>
- <AecRecipeSpec defaultUnit="in">
- <Recipe idStr="section">
  <Label>Section</Label>
  - <Implementation>
    <ObjectARX logicalAppName="AecRcpBase40" funcRcp="Stamp" />
  </Implementation>
  - <Data>
  - <Item idStr="block">
    - <AcadBlockProps>
      <BlockName dwgLib="dwgs/AecDtlLib_WoodMillwork.dwg" remapLayers="true"
        ref="dbTable">R_BLOCK</BlockName>
      <Scale>1.0</Scale>
    </AcadBlockProps>
    - <AcadEntityProps>
      <Layer ref="layerKey">MED</Layer>
    </AcadEntityProps>
  </Item>
  </Data>
</Recipe>
</AecRecipeSpec>
```

Basically this tells the ARX AECRCPbase40 to use the jig "stamp" to read the listed drawing and clone the block named into the current drawing.

Keynotes:

This section is definitely a work in progress...

There are two types of keynotes Reference keynotes and sheet keynotes – they can both look at the same database. The Reference keynote will use the number (usually a CSI, or Unifomat number) in the drawing. The sheet keynote will overwrite this number with a ?. The ? will be filled in when you generate the sheet keynote legend.

Here too notice that the keynote tools (on the palette) have different properties that control many aspects of how the tools work (leader types etc)

Creating a keynote dataset

I include this here not because it is basic or intermediate, but because it is probably essential if you want to use the tool. I cover this in the advanced class

Problem:

For any given “sheet type” (plans, RCP’s, building sections, elevations etc) there will be two types of key notes. Those that will “Most likely occur on all sheets of this type”

And those that are unique to one sheet only.

Solution is two part.

One create the “Most likely” notes in a database and use the Reference Key

Create a database for the other notes and use the sheet keynote, or use the text and/or leader commands to pick up this text.

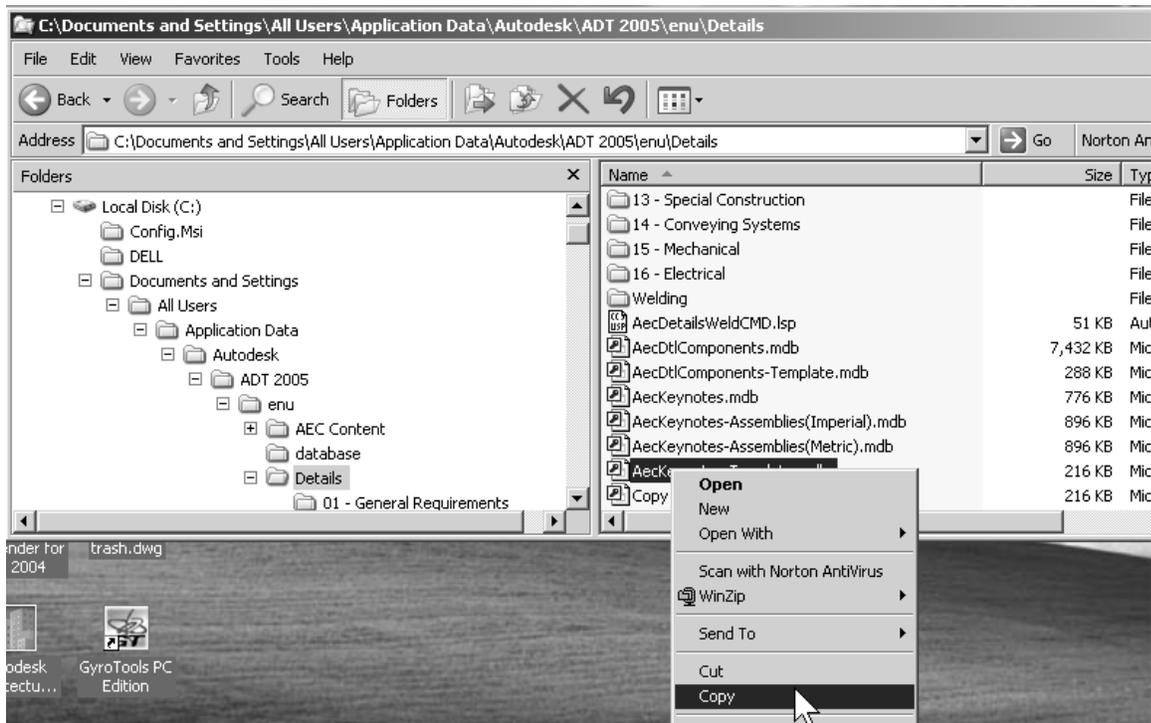
Alternative I have not looked into is to use the scheduling function, make a block that has a property set def – I can see no way to pull a field into a schedule (have not looked hard)...

OK

Key # is simple integer – note is text.

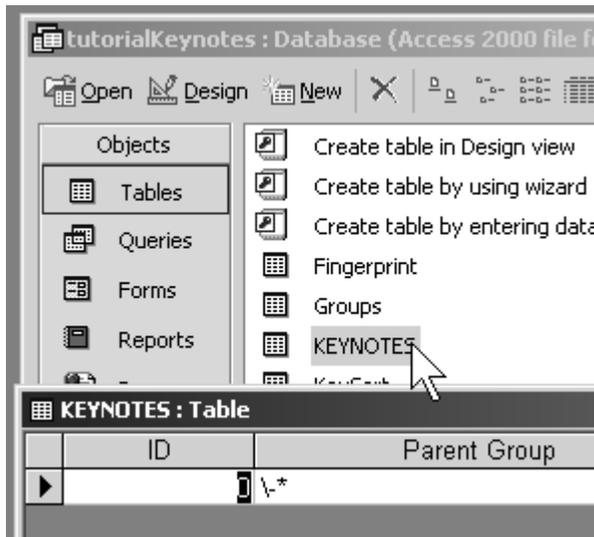
This keynote database will be used for keyed notes that occur on all sheets – for instance this database will be the plan notes – I would set up a different database for elevations and sections notes.

Use windows explorer to copy and past the key template



rename the key template in windows explorer (right click rename) tutorial keynotes

open this file in access



double click on the keynotes table to open it up

add key notes by adding number to the key field – this value will appear as the key # if you use reference key notes – this value will be overwritten if you use sheet keynotes

KEYNOTES : Table			
ID	Parent Group	Key	
1	*	1	5' C
2	*	2	2AI
0	*	1	

Annotation

add an ID to match the key – the ID is not used by ADT (verify), but is used by access to link tables together. This ID must be unique for each line in the database., so as you are entering in values, increment the ID also

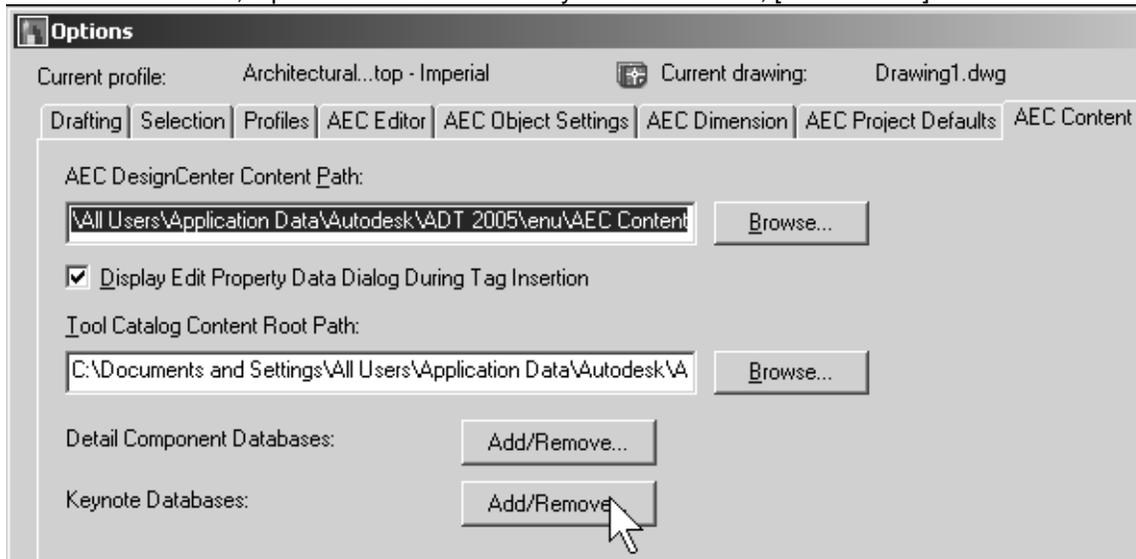
KEYNOTES : Table				
	ID	Parent Group	Key	Not
	1	*	1	5' Clear diameter wheelchair turning radius
	2	*	2	2AIO B.C. Fire Extinguisher in recessed cabi
	3	*	3	Barrier Free Dual Level Water Cooler
	4	*	4	Built in presentation screen
	5	*	5	Plumbing Chase
▶	6	*	6	Mechanical Chase 1 hour rated typical
*	0	*		

save close exit access

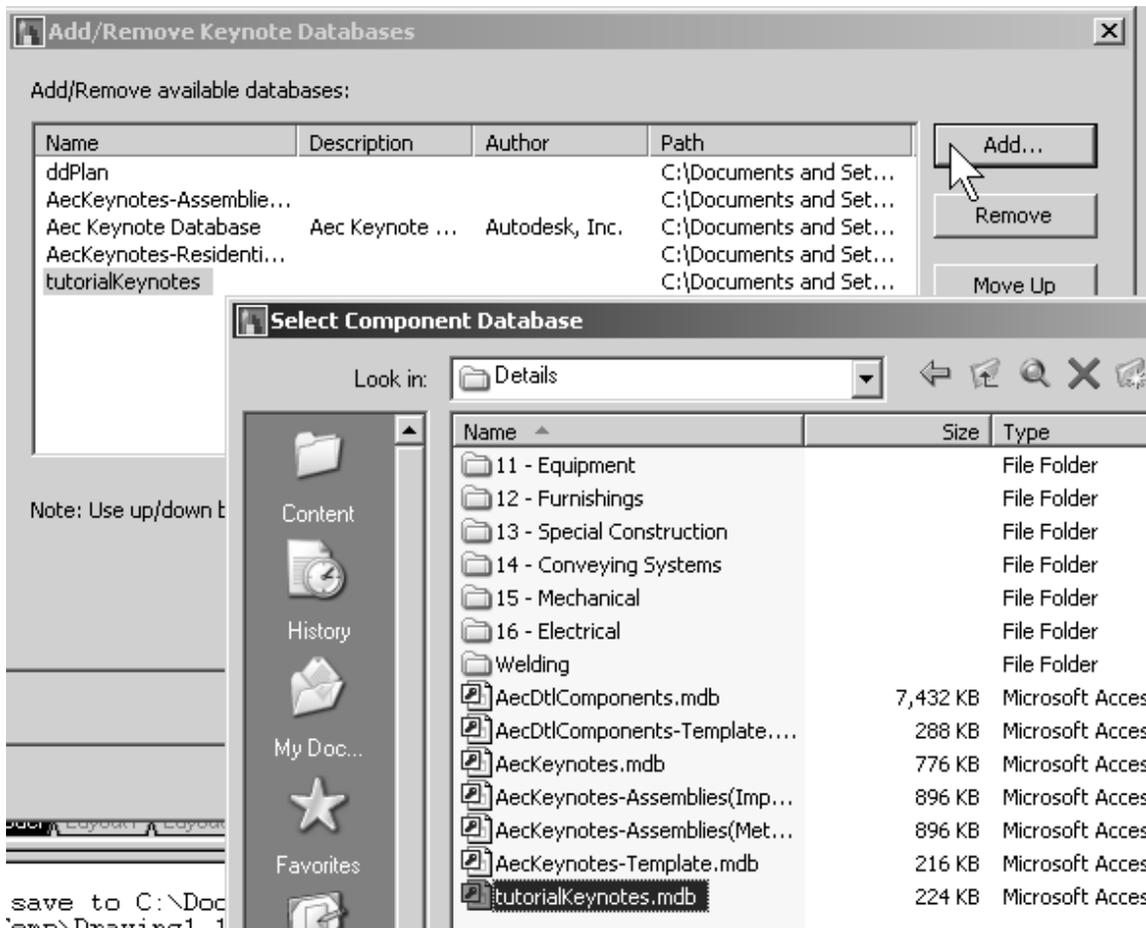
(side note: if you have an existing database and you want to copy and reuse it, copy it with windows explorer and then open the “fingerprint” table of the copied database and erase the value of the GUID)

Fingerprint : Table	
	GUID
▶	{CD349101-ADDC-40FC-A58E-BA4D2B9573CC}
*	

in a session of ADT, Options>AEC Content>Keynote Databases, [add/remove]

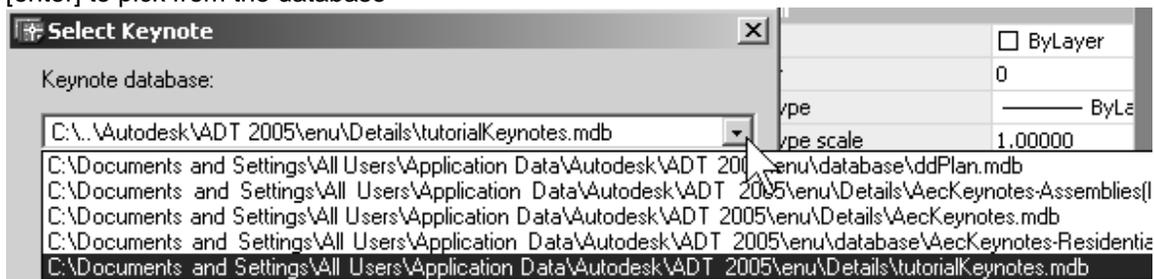


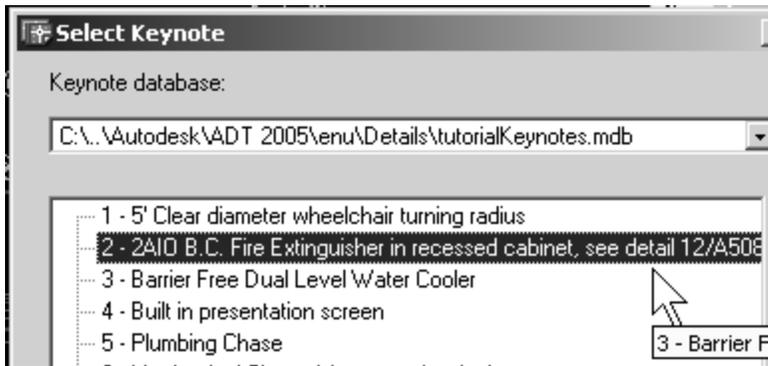
pick the add button and browse to the tutorial keynotes database



the database is now available for you to use

in the dwg you want to keynote, place a reference keynote (use the reference rather than the sheet – the sheet will override the keynote number with it's own numbering system [enter] to pick from the database





place the keynote (change the view to key only)

now if the budget gets slashed and all your fire extinguishers need to be flush mounted, just go change this note one place in the database and all instances of that note will be updated.

Yes, it does not solve the problem of jumped numbers, but I think this is a valid approach that will work for now – will leave look at combining reference and sheet keynoting to dennis and fritz

OK, that's all for the tutorial –

Groups in Keynotes. In the Groups table you can establish the “tree” organization you see when you select a keynote. To continue with the sample above,

Open the keynote database

Open the Groups table

Add a group “ADA Code Note”

ID	Parent	Group	Description
962,	\-*	Notes	Accessible Code Notes

open the keynotes table add

ID	Parent	Key	Note
4502	Notes-Accessible Code Notes	1	5' clear turning radius

- or rather if this key already exists, just change the Parent group from /-*

to

Notes-Accessible Code Notes

then go back and add the key note –

does the same thing – almost – gets the user into the database.