Digital Mapping Processes and Property Maintenance Procedures

in the Johnson County Clerk's Office

Office of the County Clerk
Johnson County, Kansas

Property Ownership Data

- History
  - Maintaining Property coverages since 1996
  - Used to create:
    - Property Ownership maps
    - Taxing Unit maps
    - School & Fire District maps
    - Myriad of additional applications & maps in other depts.

Property Ownership Data

- **Previously**
  - **Property splits and combinations using COGO and a few custom AMLs**
  - **Entire subdivision plats input using COGO alone**

Property Ownership Splits

- Property Information used in mapping
  - Original property identification number
  - New property identification number
  - Copy of document used to split property:
    - Deed
    - Annexation
    - Sewer Boundary
    - Street Dedication

Property Ownership Splits

- INFO files
  - Transfer File
    - Date received
    - Date due
    - Section (Tile) number
    - Order type
    - Date finished
  - Real Estate File
    - Order number
    - Property real estate number

Property Ownership Splits

- INFO Files
  - AMLs used in INFO files
    - &r transout - checks out transfer order file
    - &r transadd - adds split order information
    - &r transupd - updates transfer order file when complete
    - &r transin - checks in transfer order file

Property Ownership Splits

- Processing Property Split Orders
  - Create ‘Before’ property file
    - &r todobef - uses INFO file information
  - Open ARCEDIT session
    - &r <order number> - opens AE session & zooms to ‘Before’ parcel property

Property Ownership Splits

- Processing Property Split Orders
  - Splitting Property
    - Property lines
    - Polygon labels
    - Update .PAT
    - Add and change annotation

Property Ownership Splits

- Processing Property Split Orders
  - Create ‘After’ property file
    - &r todoaft - uses INFO file information
    - Create Appraiser List
    - &r getout - ‘after’ coverage used for QC’ing
    - Split is QC’ed and put into current coverage
    - INFO file updated
    - Completed order sent to Appraiser’s Office

Digital Platting Process

- Submittal
  - Received
    - Digital
      - Disk
        - Sent to Public Works and passed on to County Clerk
      - E-mail
        - Sent to both Public Works and County Clerk
    - Hardcopy
      - Sent to Public Works, Register of Deeds, and County Clerk

Digital Platting Process

- Submittal
  - Reviewed - Check List
    - County Clerk
    - Public Works
    - Register of Deeds
  - Recorded
    - Register of Deeds
      - After Public Works and County Clerk sign hardcopy, Register of Deeds signs and records the plat.
    - County Clerk
      - Receives hardcopy of recorded plat, COGO can now be officially started on the plat.

Digital Platting Process

- COGO
  - AutoCAD
    - Create DXF file from AutoCAD DWG file
    - Obtain layer names from DWG file
    - Import DXF file into ARC/INFO

Digital Platting Process

COGO

- Combine layers in COGO
  - VI file - joco.dxfspec
    - Boundary
    - Lot Lines
    - Right of Ways
    - Street Centerlines
  - Check layers
    - Closure
    - Bearings
    - Distance
    - Coding

Digital Platting Process

- COGO
  - Annotation
    - Dimensions
      - Boundary
      - Lots
      - Plot Coverage
        - Compare dimension annotation w/ actual length
        - Correct errors

Digital Platting Process

- COGO
  - Annotation
    - Streets
      - Names
      - Centered properly
    - Lots
      - Subcode number
      - Block number
      - Lot number
    - Plat name / Subcode number

Digital Platting Process

- COGO
  - Individual QC
    - Line coding
    - Label points
    - Annotation
    - Subdivision/Lot numbers
  - Move plat coverage to server for processing and merging

Digital Platting Process

- Processing/ Quality Control
  - Processing
    - Assign KUPN numbers to lots
    - Assign Parcel numbers to right of ways
    - Move plat coverage into the correct coordinate space of the existing tile property coverage

Digital Platting Process

- Processing/ Quality Control
  - Quality Control
    - Checkpoly (AML)
      - Insures all polygons are closed for not just the plat, but for the entire tile
    - Mfile
      - Updates the tile's file with regards to arcs, polygons, nodes and annotation
    - Lslivepc
      - Merges the tile property coverage with the County's 'live' property coverage

AMLs

- **MULTDXFARC.aml**
  - **Main Module**

```
&args dxf_filename
&call init1
&call open_dxfspec_file
&call dxfarc_session
&call close_dxfspec_file
&return
```

AMLs

MULTDXFARC.aml

- Main Module

```
&args dxf_filename
&call init1
&call open_dxfspec_file
&call dxfarc_session
&call close_dxfspec_file
&return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine init1

  &severity &error &fail

  &if [show program] <> ARC &then
    &return &error Must be in ARC ....

  &if [null %dxf_filename%] = .true. &then
    &sv dxf_filename = [getfile *.dxf -file 'Select DXF file']

  &sv dxfspec_filename = [getfile *.dxfspec -file 'Select DXFSPEC file']
/*&sv dxfspec_filename = [before %dxf_filename% .].dxfspec

  &sv carriage_return = ' '

  &return
```

AMLs

MULTDXFARC.aml

- Main Module

```
&args dxf_filename
&call init1
&call open_dxfspec_file
&call dxfarc_session
&call close_dxfspec_file
&return
```

```
/*   Subroutine: Open File
/*-----------------------------------------------------------------
  &routine open_dxfspec_file
  &sv fileunit = [open %dxfspec_filename% status -read]
  &if %status% <> 0 &then
    &return &error Error opening %dxfspec_filename%
  &return
```

AMLs

MULTDXFARC.aml

- Main Module

```
&args dxf_filename
&call init1
&call open_dxfspec_file
&call dxfarc_session
&call close_dxfspec_file
&return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine dxfarc_session

/*  &tty dxfarc %dxf_filename% %temp_cvg%

  &sv eof = 0
  &do &until %eof% = 1
    &sv record = [unquote [read %fileunit% readstatus]]
    &select %readstatus%
      &when 0
        &do
        &sv current_time = [after [date -ftag] .]
        &sv temp_cvg = xx%current_time%
        dxfarc %dxf_filename% %temp_cvg%
        %record%
        %carriage_return%
        y
        y
        &if [exists %temp_cvg% -cover] = .true. &then
          &do
          /* If an acode INFO file exists for the coverage, then
          /* add COGO attributes ...
          &if [exists %temp_cvg%.acode -info] = .true. &then
            add_cogo_att %temp_cvg%
          &call arcedit_session
          &end
        &end
      &when 102
        &sv eof = 1
      &otherwise
        &return &error Error reading record from fileunit %fileunit%
    &end
  &end

  &return
```

AMLs

MULTDXFARC.aml

- Main Module

```
&args dxf_filename
&call init1
&call open_dxfspec_file
&call dxfarc_session
&call close_dxfspec_file
&return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine arcedit_session

  display 9999 size frame 1000 800
  arcedit

  editcoverage %temp_cvg%
  drawenvironment all
  drawenvironment tic off
  drawenvironment node errors
  backcoverage [getcover # # 'Select a back coverage' -none] 2
  backenvironment all
  backenvironment tic off
  backenvironment node errors
  draw

  &type
  &type %record%
  &type
  &type Type &RETURN to continue (Do not QUIT arcedit)
  &type
  &tty
```

AMLs

MULTDXFARC.aml

- Main Module

```
&args dxf_filename
&call init1
&call open_dxfspec_file
&call dxfarc_session
&call close_dxfspec_file
&return
```

```
&if [query 'Do you want to keep this coverage (y/n)' .false.] = .true.
&then
  &do
  &sv cvg = [response 'What do you want to name the coverage']
  &if [exists %cvg% -cover] = .true. &then
    &do
    &type --> A coverage by that name already exists.
    &type --> Do you want to overwrite (o) or append (a)?
    &if [getchar ' '] = a &then
      &do
      editfeature arcs
      select all
      put %cvg%
      y
      quit yes
      kill %temp_cvg% all
      &end
    &else
      &do
      quit yes
      kill %cvg% all
      rename %temp_cvg% %cvg%
      &end
    &end
  &else
    &do
    quit yes
    /* If the coverage was killed without the all option then its
    /* acode info file will still be hanging around ... so delete it
    &if [exists %cvg%.acode -info] = .true. &then
      &sv delstat = [delete %cvg%.acode -info]
    rename %temp_cvg% %cvg%
    &end
  &end
&else
  &do
  quit no
  kill %temp_cvg% all
  &end
&return
```

AMLs

MULTDXFARC.aml

- Main Module

```
&args dxf_filename
&call init1
&call open_dxfspec_file
&call dxfarc_session
&call close_dxfspec_file
&return
```

```
/*   Subroutine: Close File
/*-----------------------------------------------------------------
  &routine close_dxfspec_file
  &if [close %fileunit%] <> 0 &then
    &return &error Error closing fileunit %fileunit%
  &return
```

AMLs

DIGPLAT_DRIVER.aml

- Main Module

```
 &call init1
  &sv cvg = %.sy$per%
  &call create_perimeter

/*If menu passes a non-null right-of-way coverage and that coverage is
/*named something other than none, then process it ...
  &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
    &do
    &sv cvg = %.sy$row%
    &call create_right_of_way
    &end

  &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
    &do
    &sv cvg = %.sy$lot%
    &call create_lot_lines
    &end

  &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
    &do
    &sv cvg = %.sy$cl%
    &call create_centerlines
    &end

  &call create_plat
  &return
```

AMLs

DIGPLAT_DRIVER.aml

- Main Module

```
&call init1
 &sv cvg = %.sy$per%
 &call create_perimeter

/*If menu passes a non-null right-of-way coverage and that coverage is
/*named something other than none, then process it ...
 &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
   &do
   &sv cvg = %.sy$row%
   &call create_right_of_way
   &end

 &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
   &do
   &sv cvg = %.sy$lot%
   &call create_lot_lines
   &end

 &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
   &do
   &sv cvg = %.sy$cl%
   &call create_centerlines
   &end

 &call create_plat
 &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine init1
  &severity &error &fail
  &if [show program] <> ARC &then
    &return &error Must be in ARC ....

/*If menu passes a null perimeter coverage or one named none, then
/*don't go any further ...
  &if [null %.sy$per%] = .true. or %.sy$per% = none &then
    &return &error Must select a perimeter coverage ...

/*Check plat coverage name ...
  &sv plat_cvg = %.sy$subcode%%.sy$page%
  &if [exists %plat_cvg% -cover] = .true. &then
    &do
    &type --> A coverage by the name of %plat_cvg% already exists.
    &type --> Do you want to overwrite (y/n)?
    &if [query ' '] = .true. &then
      kill %plat_cvg% all
    &else
      &return &error Process aborted ...
    &end

  &return
```

AMLs

DIGPLAT_DRIVER.aml

- Main Module

```
&call init1
 &sv cvg = %.sy$per%
 &call create_perimeter

/*If menu passes a non-null right-of-way coverage and that coverage is
/*named something other than none, then process it ...
 &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
   &do
   &sv cvg = %.sy$row%
   &call create_right_of_way
   &end

 &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
   &do
   &sv cvg = %.sy$lot%
   &call create_lot_lines
   &end

 &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
   &do
   &sv cvg = %.sy$cl%
   &call create_centerlines
   &end

 &call create_plat
 &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine create_perimeter
/*&sv cvg = [response 'Name of perimeter coverage']
/*&sv cvg = [entryname [getcover # # 'Select perimeter coverage']]
  build %cvg% line
  &call cogo_attributes
  build %cvg% node
  renode %cvg%
  addxy %cvg% node
  snap %cvg%
  &run additems %cvg% aat
  &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine cogo_attributes

/*If the coverage does not have COGO attributes (which is indicated
/*by the lack of an item named angle) and an .acode file exists (thus
/*making it possible to add COGO attributes), then add COGO attributes.

  &if [iteminfo %cvg% -arc angle -exists] = .true. &then
    &type --> %cvg% already has COGO attributes
  &else
    &do
    &if [exists %cvg%.acode -info] = .false. &then
      &type --> Unable to add COGO attributes to %cvg% due to missing acode file
    &else
      add_cogo_att %cvg%
    &end

  &return
```

AMLs

DIGPLAT_DRIVER.aml

- Main Module

```
&call init1
 &sv cvg = %.sy$per%
 &call create_perimeter

/*If menu passes a non-null right-of-way coverage and that coverage is
/*named something other than none, then process it ...
 &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
   &do
   &sv cvg = %.sy$row%
   &call create_right_of_way
   &end

 &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
   &do
   &sv cvg = %.sy$lot%
   &call create_lot_lines
   &end

 &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
   &do
   &sv cvg = %.sy$cl%
   &call create_centerlines
   &end

 &call create_plat
 &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine create_right_of_way
/*&sv cvg = [response 'Name of right-of-way coverage']
/*&sv cvg = [entryname [getcover # # 'Select right-of-way coverage']]
  build %cvg% line
  &call cogo_attributes
  build %cvg% node
  renode %cvg%
  addxy %cvg% node
  snap %cvg%
  &run additems %cvg% aat
  &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine cogo_attributes

/*If the coverage does not have COGO attributes (which is indicated
/*by the lack of an item named angle) and an .acode file exists (thus
/*making it possible to add COGO attributes), then add COGO attributes.

  &if [iteminfo %cvg% -arc angle -exists] = .true. &then
    &type --> %cvg% already has COGO attributes
  &else
    &do
    &if [exists %cvg%.acode -info] = .false. &then
      &type --> Unable to add COGO attributes to %cvg% due to missing acode file
    &else
      add_cogo_att %cvg%
    &end

  &return
```

AMLs

DIGPLAT_DRIVER.aml

- Main Module

```
&call init1
 &sv cvg = %.sy$per%
 &call create_perimeter

/*If menu passes a non-null right-of-way coverage and that coverage is
/*named something other than none, then process it ...
 &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
   &do
   &sv cvg = %.sy$row%
   &call create_right_of_way
   &end

 &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
   &do
   &sv cvg = %.sy$lot%
   &call create_lot_lines
   &end

 &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
   &do
   &sv cvg = %.sy$cl%
   &call create_centerlines
   &end

 &call create_plat
 &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine create_lot_lines
/*&sv cvg = [response 'Name of lot lines coverage']
/*&sv cvg = [entryname [getcover # # 'Select lot lines coverage']]
  build %cvg% line
  &call cogo_attributes
  build %cvg% node
  renode %cvg%
  addxy %cvg% node
  snap %cvg%
  &run additems %cvg% aat
  &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine cogo_attributes

/*If the coverage does not have COGO attributes (which is indicated
/*by the lack of an item named angle) and an .acode file exists (thus
/*making it possible to add COGO attributes), then add COGO attributes.

  &if [iteminfo %cvg% -arc angle -exists] = .true. &then
    &type --> %cvg% already has COGO attributes
  &else
    &do
    &if [exists %cvg%.acode -info] = .false. &then
      &type --> Unable to add COGO attributes to %cvg% due to missing acode file
    &else
      add_cogo_att %cvg%
    &end

  &return
```

AMLs

DIGPLAT_DRIVER.aml

- Main Module

```
&call init1
 &sv cvg = %.sy$per%
 &call create_perimeter

/*If menu passes a non-null right-of-way coverage and that coverage is
/*named something other than none, then process it ...
 &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
   &do
   &sv cvg = %.sy$row%
   &call create_right_of_way
   &end

 &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
   &do
   &sv cvg = %.sy$lot%
   &call create_lot_lines
   &end

 &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
   &do
   &sv cvg = %.sy$cl%
   &call create_centerlines
   &end

 &call create_plat
 &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine create_centerlines
/*&sv cvg = [response 'Name of centerlines coverage']
/*&sv cvg = [entryname [getcover # # 'Select centerlines coverage']]
  build %cvg% line
  &call cogo_attributes
  build %cvg% node
  renode %cvg%
  &run additems %cvg% aat
  &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine cogo_attributes

/*If the coverage does not have COGO attributes (which is indicated
/*by the lack of an item named angle) and an .acode file exists (thus
/*making it possible to add COGO attributes), then add COGO attributes.

  &if [iteminfo %cvg% -arc angle -exists] = .true. &then
    &type --> %cvg% already has COGO attributes
  &else
    &do
    &if [exists %cvg%.acode -info] = .false. &then
      &type --> Unable to add COGO attributes to %cvg% due to missing acode file
    &else
      add_cogo_att %cvg%
    &end

  &return
```

AMLs

DIGPLAT_DRIVER.aml

- Main Module

```
&call init1
 &sv cvg = %.sy$per%
 &call create_perimeter

/*If menu passes a non-null right-of-way coverage and that coverage is
/*named something other than none, then process it ...
 &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
   &do
   &sv cvg = %.sy$row%
   &call create_right_of_way
   &end

 &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
   &do
   &sv cvg = %.sy$lot%
   &call create_lot_lines
   &end

 &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
   &do
   &sv cvg = %.sy$cl%
   &call create_centerlines
   &end

 &call create_plat
 &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
 &routine create_plat

 &sv cvg = %.sy$per%
 &call copy_perimeter

 &call start_ae

 &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
   &do
   &sv cvg = %.sy$row%
   &call merge_right_of_way
   &end

 &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
   &do
   &sv cvg = %.sy$lot%
   &call merge_lot_lines
   &end

 &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
   &do
   &sv cvg = %.sy$cl%
   &call merge_centerlines
   &end

 &call quit_ae
 &return
```

AMLs

DIGPLAT_DRIVER.aml

▼Subroutine - create_plat

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine create_plat

  &sv cvg = %.sy$per%
  &call copy_perimeter

  &call start_ae

  &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
    &do
    &sv cvg = %.sy$row%
    &call merge_right_of_way
    &end

  &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
    &do
    &sv cvg = %.sy$lot%
    &call merge_lot_lines
    &end

  &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
    &do
    &sv cvg = %.sy$cl%
    &call merge_centerlines
    &end

  &call quit_ae
  &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine copy_perimeter
  copy %cvg% %plat_cvg%
  &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine start_ae
  display 9999 size frame 1000 800
  arcedit
  editcoverage %plat_cvg%
  drawenvironment arc node errors
  setdrawsymbol 2
  editfeature arc
  select all
  calc line = 13
  colorit
  draw
  &pause
  &return
```

AMLs

DIGPLAT_DRIVER.aml

▼Subroutine - create_plat

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine create_plat

  &sv cvg = %.sy$per%
  &call copy_perimeter

  &call start_ae

  &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
    &do
    &sv cvg = %.sy$row%
    &call merge_right_of_way
    &end

  &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
    &do
    &sv cvg = %.sy$lot%
    &call merge_lot_lines
    &end

  &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
    &do
    &sv cvg = %.sy$cl%
    &call merge_centerlines
    &end

  &call quit_ae
  &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine merge_right_of_way
/*&sv cvg = [response 'Name of right-of-way coverage']
/*&sv cvg = [entryname [getcover # # 'Select right-of-way coverage']]
  &sv arcsnap_dist = .00001
  &sv nodesnap_dist = .000005
  &sv line_value = 12
  &sv extend_dist = .1
  &sv delete_if_lt_dist = .1
  &call merge_common
  &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine merge_lot_lines
/*&sv cvg = [response 'Name of lot lines coverage']
/*&sv cvg = [entryname [getcover # # 'Select lot lines coverage']]
  &sv arcsnap_dist = .025  /*.05
  &sv nodesnap_dist = .025 /*.05
  &sv line_value = 10
  &sv extend_dist = .05  /*.2
  &sv delete_if_lt_dist = .05  /*.1
  &call merge_common
  &return
```

DIGPLAT_DRIVER.aml

▼Subroutine - create_plat

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine create_plat

  &sv cvg = %.sy$per%
  &call copy_perimeter

  &call start_ae

  &if [null %.sy$row%] = .false. and %.sy$row% <> none &then
    &do
    &sv cvg = %.sy$row%
    &call merge_right_of_way
    &end

  &if [null %.sy$lot%] = .false. and %.sy$lot% <> none &then
    &do
    &sv cvg = %.sy$lot%
    &call merge_lot_lines
    &end

  &if [null %.sy$cl%] = .false. and %.sy$cl% <> none &then
    &do
    &sv cvg = %.sy$cl%
    &call merge_centerlines
    &end

  &call quit_ae
  &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine merge_centerlines
/*&sv cvg = [response 'Name of centerlines coverage']
/*&sv cvg = [entryname [getcover # # 'Select centerlines coverage']]
  &sv arcsnap_dist = .05
  &sv nodesnap_dist = .05
  &sv line_value = 17
  &sv extend_dist = .1
  &sv delete_if_lt_dist = .1
  &call merge_common
  &return
```

AMLs

DIGPLAT_DRIVER.aml

▼Subroutine - merge_common

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine merge_centerlines
/*&sv cvg = [response 'Name of centerlines coverage']
/*&sv cvg = [entryname [getcover # # 'Select centerlines coverage']]
  &sv arcsnap_dist = .05
  &sv nodesnap_dist = .05
  &sv line_value = 17
  &sv extend_dist = .1
  &sv delete_if_lt_dist = .1
  &call merge_common
  &return
```

```
/*   Subroutine:
/*-----------------------------------------------------------------
  &routine merge_common
  arcsnap on %arcsnap_dist%
  snaporder segment vertex
  nodesnap closest %nodesnap_dist%
  intersectarcs all
  get %cvg%
  calc line = %line_value%
  reselect dangle
  drawselect
  &if [show number selected] > 0 &then
    &do
    extend %extend_dist%
    reselect dangle
    drawselect
    &if [show number selected] > 0 &then
      reselect length < %delete_if_lt_dist%
      &if [show number selected] > 0 &then
        delete
    &end
  colorit
  draw
  &type --> Just finished adding arcs from %cvg%
  &pause
  &return
```

Contacts

- **Office of the County Clerk**

111 S. Cherry St.

Suite 1200

Olathe, Kansas 66061

(913) 715-0775

www.jocoks.com/countyclerk

Email:

Janet.Jacobs@jocoks.com

Rodney.Odom@jocoks.com

Steven.Tolman@jocoks.com