The CEDRA Corporation's

COMMAND OF THE MONTH

A monthly information bulletin October 2005

FEATURED COMMAND
The Fillet and
Intersection Tool

Application Description

A new tool has been introduced for ArcGIS users in the CEDRA-Intersection-Tools toolbar which should be of interest to those involved in the creation of corner lots. The new tool is the *Trim Elements to Intersection and/or Introduce Fillet* tool shown in Figure 1.

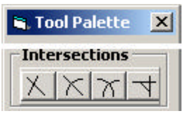


Figure 1
CEDRA Palette Intersection Tools

With this tool, the user can intersect any two primitive elements (two-point lines, polylines and/or curves), and

- Create the point of intersection,
- Create the point of intersection and trim the intersecting primitives to the point of intersection, and/or
- Introduce a linear, or circular arc fillet.

Although this tool can be used as a substitute to the other three intersection tools, its primary function is the introduction of fillets. Regarding the sole creation of intersection points, this tool has certain advantages, and certain disadvantages over the other three intersection tools shown in Figure 1.

Selection Of The Intersecting Elements for Intersection

In selecting the primitives to be intersected, note the following:

- A primitive may be a two-point line (hereinafter referred to as just *line*), a polyline, or a circular curve (hereinafter referred to as just *curve*).
- There is no order as to which element type is to be selected first and which second.
- The tool is not repetitive. That is, you cannot pick one primitive and have it intersect with many others, one after the other.
- A curve to be intersected with this tool cannot be defined by picking a point, and entering a radius.

Command Of The Month bulletin

This month's issue presents the introduction of a new tool for intersecting lines, polylines and curves, and introducing linear or circular fillets..

Depending upon the type of primitives selected for intersection, there are six possible types of intersections, indicated below, with each intersection type yielding one, two, or possibly more intersection points, also as indicated below.

When the possibility exists for an intersection to yield more than one intersection point, regardless of whether a point is to be created at that location, or not, a selection of one of the intersections is made by the program. The basis for this selection is identified in the types of intersections discussion shown below.

Note that the intersection need not be located within the physical extent of either primitive being intersected.

Types of Intersections

1. **Line with another line** yielding only one possible intersection. The intersection point need not be located within the physical extent of either primitive.

2. **Line with a polyline** yielding at least as many intersections as there are crossings of the line with the polyline. If the polyline intersects a line at more than one location, the intersection that is used, is the one nearest to the user's click that selected the polyline. See Figure 2.

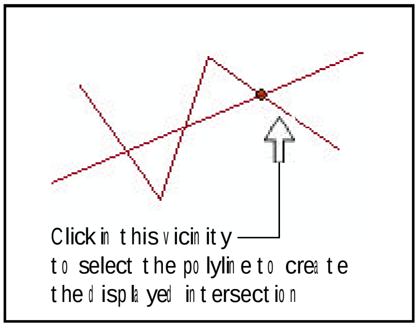


Figure 2 Line - Polyline Intersection

3. **Line with a curve** yielding two possible points of intersection. The intersection that is used is the one nearest to the user's click that selected the curve. See Figure 3.

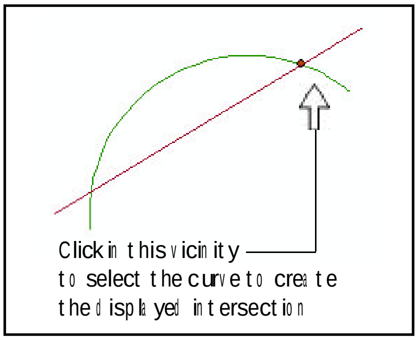


Figure 3 Line - Curve Intersection

4. **Polyline with another polyline** yielding at least as many possible intersections as the number of times the polylines cross each other. The intersection that is used is the one nearest to the user's click that selected the second polyline. See Figure 4. Also refer to the additional comments below regarding polyline intersection rules.

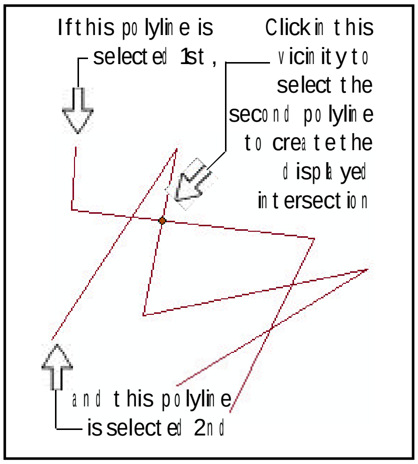


Figure 4
Polyline - Polyline Intersection

5. **Polyline with a curve** yielding at least as many possible intersections as the number of times the polyline crosses the curve. The intersection that is used is the one nearest to the user's click that selected the second primitive, regardless whether that primitive is a curve or a polyline. See Figure 5. Also refer to the additional comments below regarding polyline intersection rules.

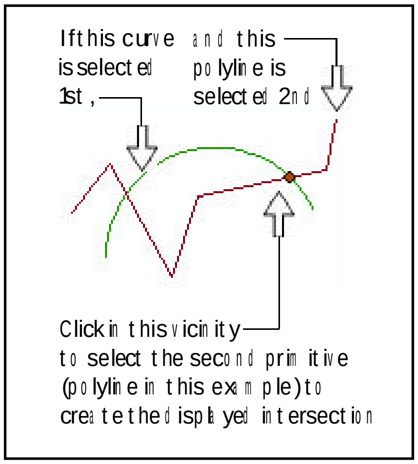

If this curve
is selected
1st,
and this
polyline is
selected 2nd
Click in this vicinity
to select the second primitive
(polyline in this example) to
create the displayed intersection

Figure 5
Polyline - Curve Intersection

6. **Curve with another curve** yielding two possible intersections. The intersection that is used is the one nearest to the user's click that selected the second curve. See Figure 6.

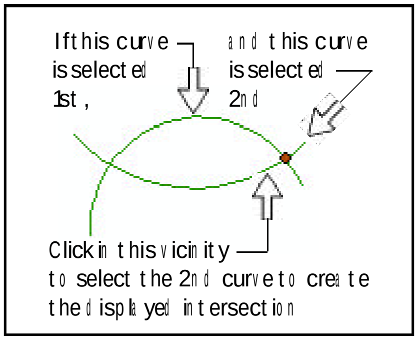


Figure 6
Curve - Curve Intersection

Polyline Intersection Rules

Special attention needs to be paid to the intersection of polylines with lines, curves and other polylines. This is necessary because many a time a polyline is not a design polyline, but a digitized one, and as such it may be comprised of an extremely large number of very small segments that create a saw tooth effect when looked at microscopically.

Furthermore, even design curves could be imported from sources that treat them as polylines without regard to a corresponding radius as an accessible attribute for recomputation. With regards to this issue, it is noted that although other tools of CEDRA-AVcad, CEDRA-AVcogo and CEDRA-AVparcel do force a circular curve to a polyline under certain conditions, this specific tool does not. As such, *a polyline that does not have a radius associated with it is treated as a polyline. A polyline to be treated as a curve must have*

- *A radius value other than zero in the RAD attribute, and*
- *A curve type value of 1, 2, 3 or 4 in the CTP attribute.*

It has been noted above that primitive lines and curves will create valid intersections outside their physical limits of graphic extent. Polylines will also produce such intersections, but due to their potential configuration, they are subject to the conditions, or limitations identified below.

The following comments and figures are provided to better demonstrate the intersection selection process by the program.

1. **Polyline-Line Intersections:** Shown in Figure 7 are four potential polyline-line intersection conditions that could arise.

   In the top of Figure 7, there are two conditions of a line being intersected by two polylines, neither of which crosses the line. The left polyline progresses to the top and left (NW) and away from the line, while the right polyline progresses to the SW and towards the line. The indicated corresponding intersections will be created regardless of where the click on either polyline is made to select such polyline.

   In the bottom of Figure 7, a polyline crosses a line, and then doubles back to the left and up from the left endpoint of the line. The same polyline extends to the right and up from the right endpoint of the line, and continues to the right and up.

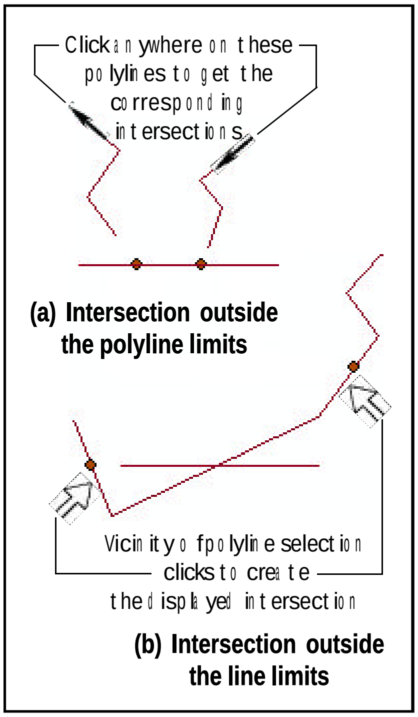

Click anywhere on these polylines to get the corresponding intersections
(a) Intersection outside the polyline limits
Vicinity of polyline selection clicks to create the displayed intersection
(b) Intersection outside the line limits

Figure 7
Potential Line - Polyline Intersections

Note that, if the click to select the polyline is made in the vicinity indicated by the open arrows in said figure, the corresponding intersection as indicated will be obtained. However, if the click to select the polyline is made somewhere around the actual crossing of the polyline with the line, then the intersection would be at said crossing.

2. **Polyline-Polyline Intersections:** Shown in Figure 8 are two potential polyline-polyline intersection conditions that could arise.

   - One of these conditions is the case in which either or both of the end segments of a polyline does not cross the other polyline, as shown by the arrows at the top and bottom of the figure.

   - The other condition involves all in between polyline crossings, as shown by the arrow in the middle of the figure.

   The general rule presented before applies to all polyline-polyline intersections. *The click that selects the second polyline dictates the selection of the point of intersection*, be it an actual polyline crossing, or the extension of one polyline to meet the other.

   Since a polyline may at times extend beyond the extent of the current map display, it is possible that the point of intersection may be well off the map display extent.

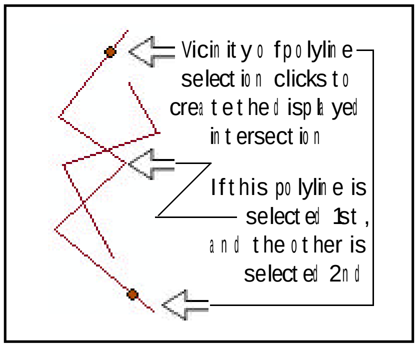

Vicinity of polyline selection clicks to create the displayed intersection
If this polyline is selected 1st, and the other is selected 2nd

Figure 8
Potential Polyline - Line Intersections

3. **Polyline-Curve Intersections:** Shown in Figure 9 are three potential polyline-curve intersection conditions that could arise.

   - The first of these conditions is the case in which either or both of the end segments of a polyline would cross the curve only if the polyline was to be extended.

   - The second condition is that in which either or both of the end segments of a polyline would cross the curve, if the curve was extended.

   - The third condition is that which involves all in between crossings of a polyline with a curve.

   As of this version of the program, the first condition will not yield an intersection. That is, a polyline will not be extended to meet a curve.

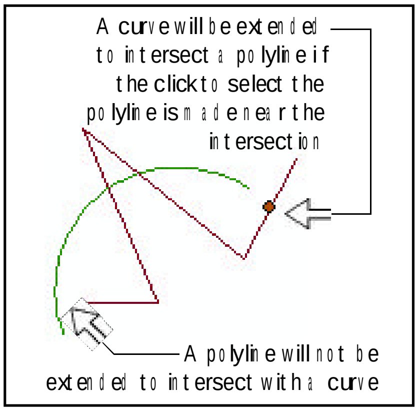
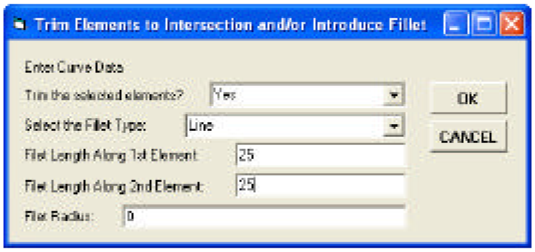

A curve will be extended to intersect a polyline if the click to select the polyline is made near the intersection
A polyline will not be extended to intersect with a curve

Figure 9
Potential Polyline - Curve Intersection

The other two conditions will generate an intersection point depending upon the click that selects the polyline.

The User's Input Specifications

Once the primitives to be intersected have been selected, the user is presented with the input dialog box of Figure 10 to specify whether primitive trimming and fillet creation are desired or not.


Trim Elements to Intersection and/or Introduce Fillet

Enter Curve Data

Trim the selected elements? Yes

Select the Fillet Type: Line

Fillet Length Along 1st Element: 25

Fillet Length Along 2nd Element: 25

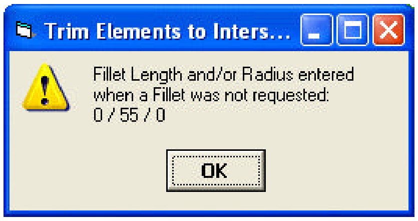
Fillet Radius: 0

Figure 10
Trim And Fillet Specifications Dialog Box

There are five data fields in this dialog box, and they all must be entered appropriate data. If any inappropriate or invalid data is entered, a message box similar to that of Figure 11 is displayed, and the dialog box of Figure 10 is displayed again.

The five data fields to be populated are:

- **Trimming of the selected primitives -** This is a Yes/No choice data field, with "Yes" being the default.



Fillet Length and/or Radius entered when a Fillet was not requested: 0 / 55 / 0

Figure 11
User Input Error Message Box

If the user selects " No", the primitives selected for intersection will be left unaltered. If the option "Yes" is selected, the primitives will be modified as per the Fillet Type parameter setting.

- **Fillet Type selection -** This is a choice data field, with the choices being "Line", "Arc" and "None". The default choice is "Line". Note that fillets may be introduced only when the intersecting primitives are lines or curves. *If one or both of the selected primitives is a polyline, a fillet will not be introduced.*

  The selection of the "Line" option requires an entry in the subsequent two data fields, and the entry of a zero (0) in the next data field (Fillet Radius). This option *will not generate a point* at the desired intersection of the two selected primitives.

  The selection of the "Arc" option requires an entry of zero (0) in the subsequent two data fields, and the entry of the fillet radius in the next data field. This option *will not generate a point* at the desired intersection of the two selected primitives.

  The selection of the "None" option requires an entry of zero (0) in the subsequent three data fields. This option is the one that *will generate a point* at the desired intersection of the two selected primitives.

- **Fillet Length Along the First Selected Primitive -** This data field is reserved for a "Line" type of fillet. Enter in this data field the distance from the desired point of intersection to the fillet end point along the first selected primitive. If the primitive is a curve, the specified distance will be treated as an arc, and not as a chord.

- **Fillet Length Along the Second Selected Primitive -** This data field is reserved for a "Line" type of fillet. Enter in this data field the distance from the desired point of intersection to the fillet end point along the second selected primitive. If the primitive is a curve, the specified distance will be treated as an arc, and not as a chord.

- **Fillet Radius -** This data field is reserved for an "Arc" type of fillet. Enter in this data field the radius of the fillet. The fillet to be created will be tangent to both of the selected primitives.

Trimming Of The Primitives

If the user selects the "Yes" option in the first data field in the input dialog box, both of the selected primitives will be trimmed to the:

- Fillet endpoints, if fillets are being introduced, or

- Point of intersection of the two primitives.

It is noted that when a primitive is trimmed, the primitive is divided in two portions at the point of trimming. Of these two portions, the one that is kept is always the longer one. If this is not desirable for either or both of the primitives, then the subject primitive should be broken, or cutoff prior to the invocation of this subject tool.

Introduction Of Fillets

Even in the simplest condition of primitive selection, that of the Line-Line intersection, there are four possibilities as in which intersection quadrant a fillet is to be introduced. The general premises of fillet insertion assume that:

- Either of the primitives to be intersected could be a line in any direction, or a curve turning clockwise or counterclockwise, and that:

- There is no restriction as to which element is to be selected first or second.

These premises or assumptions provide room for various options.

Shown in Figure 12 are two line primitives to be selected for intersection. Note the following:

- It is arbitrarily decided to select (pick) as the first primitive (element) the line that happens to run from NW to SE, and as the second primitive the line that happens to run from NE to SW.

- As each primitive is picked, it is highlighted, but after both primitives have been picked, only the second primitive remains highlighted as a reminder to which primitive was selected first and which second.

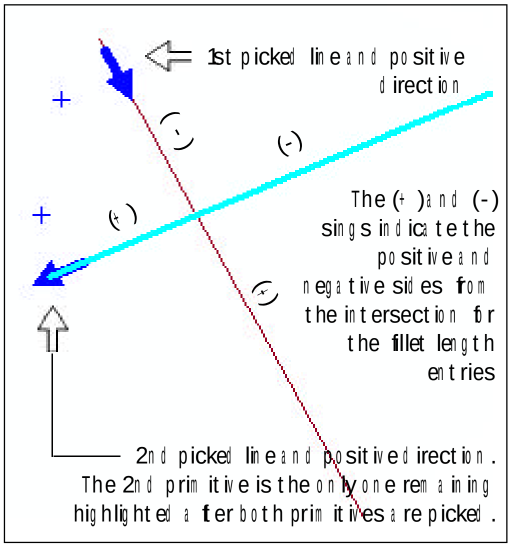

1st picked line and positive direction
(-)
(-)
(+)
(+)
The (+) and (-) sings indicate the positive and negative sides from the intersection for the fillet length entries
2nd picked line and positive direction. The 2nd primitive is the only one remaining highlighted after both primitives are picked.

Figure 12
Selection Of Line Primitives

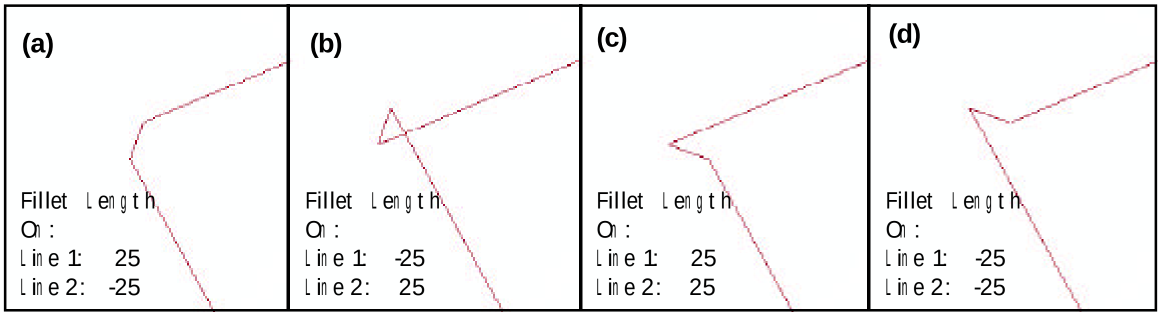


Figure 13 Fillets And Trimming As A Result Of Fillet Length Input

- Also, after both primitives have been picked, a blue arrow and a plus (+) sign to its right are displayed near the pick point that selected each of the two primitives. These markers indicate the primitive's positive direction.

Having selected the two primitives to be intersected, the aforesaid user input dialog box is displayed.

To introduce a line fillet, we must enter the fillet length (the distance from the intersection to the fillet's endpoint on a primitive) for each of the two selected primitives. A fillet length may be positive or negative, depending upon which intersection quadrant the fillet is desired to be located. Reference is made to the plus and minus signs in parentheses in Figure 12.

Shown in Figures 13(a) through 13(d) are the resultant fillets and trimming of the primitives (if trimming is requested) as a result of positive and negative fillet length entries for the primitives of Figure 12. Regarding the trimming of the primitives, note that it is their longest portion that has been kept.

The introduction of linear fillets as discussed above could be used to introduce linear fillets at corner land parcels, and to generate sight triangles at street intersections.

To introduce an arc fillet for the conditions of the example of Figure 12, we must enter the radius of the fillet (say 50), rather than two fillet lengths, which should be entered as zero (0). *Note that regardless of whether a positive or negative radius is entered, its absolute valued will be used.*

Since there are four possible quadrants into which an arc fillet can be introduced, four blue dots are displayed, as shown in Figure 14(a), at the location of the center of the four possible fillets.

The user should now click at the blue dot denoting the center of the desired fillet, for example the one shown in Figure 14(a). Having done this, the program is faced with the task of selecting one of the two possible tangent arcs. Based on program aspects beyond the scope of this document, the program selects for the subject example the arc shown in Figure 14(b), and asks the user if the 360^{o} supplement is desired. A "Yes" response will display said supplement arc, and pose the same question again. The process is repeated until a "No" response is given. For the sake of this example we will request the first supplement arc, which will cause the display of the arc fillet and trimmed primitive lines as shown in Figure 14(c). Again, with regards to the trimming of the two primitives, note that the longest portion of each primitive has been kept.

We will now consider the occasion when each of the two selected primitives is a curve. The selection process is similar to that of selecting two primitive lines. There are two basic differences:

- The blue arrows and their sidekick blue plus (+) signs are tangential to the curve at the user's selection pick, and
- There are eight potential fillet locations, depending on the magnitude of the radii and location of the centers of the two primitives, and the

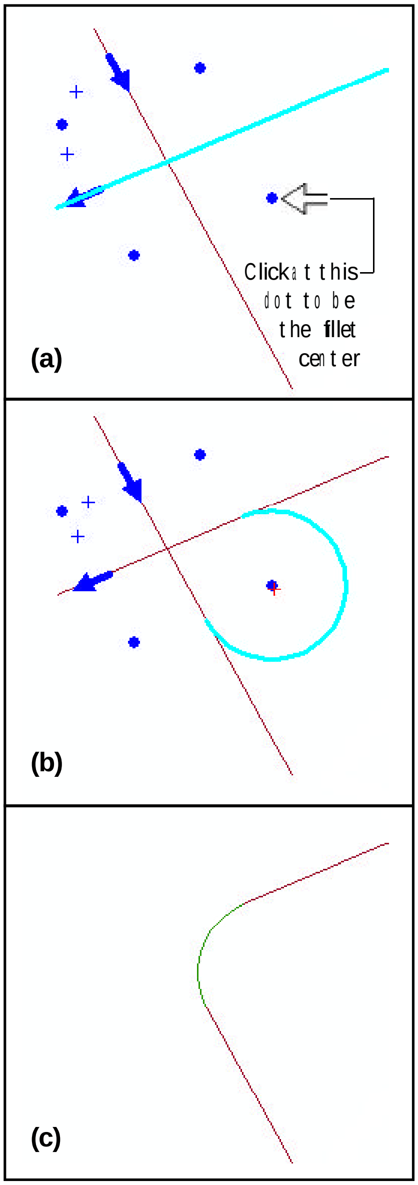


Figure 14 Introduction Of An Arc Fillet

magnitude of the radius of the desired fillet.

Shown in Figure 15(a) are two curve primitives, their direction blue arrows, the eight possible fillet centers, and the hollow arrowhead denoting the center of the purportedly desired fillet. Shown in Figure 15(b) is the program's first selection of the tangent fillet arc, which for the sake of demonstration we will consider as the desirable one. Note in Figure 15(c) the resultant trimming of the primitive curves, and that the longer arc portion of each primitive is the portion that has been maintained.

It is important to note that if a line fillet was to be inserted, rather than an arc fillet, the process would be the same as that of the two primitives being lines. For example, if:

- The curve primitive selection is the same as indicated in Figure 15(a),
- The first fillet length is specified as 25,
- The second fillet length is specified as -25, and
- No trimming of the selected two primitives is requested,

the line fillet created would be located at the inside of the two curve primitives at the SW end of the curves, see Figure 16. Note that the fillet was placed at the SW end of the two curves because the second curve was selected by a click in that vicinity

It is also important to note, that if one of the primitives is a curve, and the other a line, the concepts of fillet insertion and trimming of the primitives, as well as the operational steps would be the same as presented above.

Operational Steps for using this Tool

➤ **1** Select the *Trim Elements to Intersection and/or Introduce Fillet* tool.

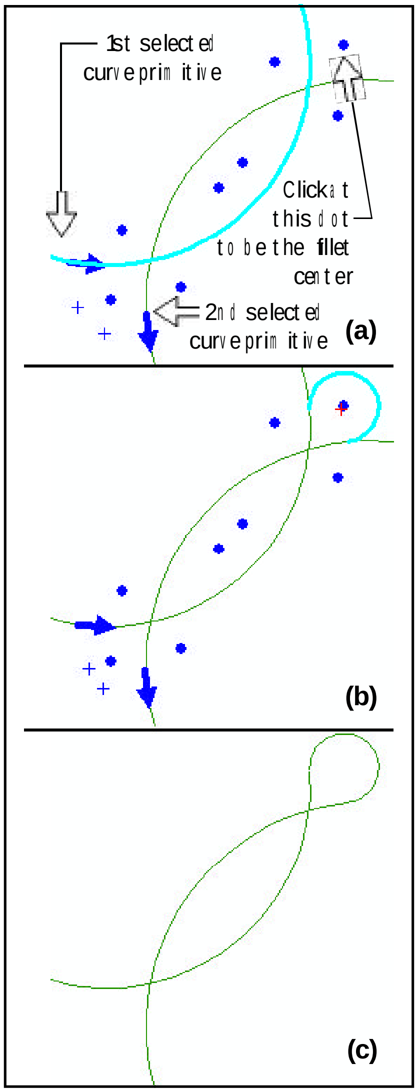

1st selected curve primitive
Click at this dot to be the fillet center
2nd selected curve primitive
(a)
(b)
(c)

Figure 15
Introduction Of An Arc Fillet Between Two Curves

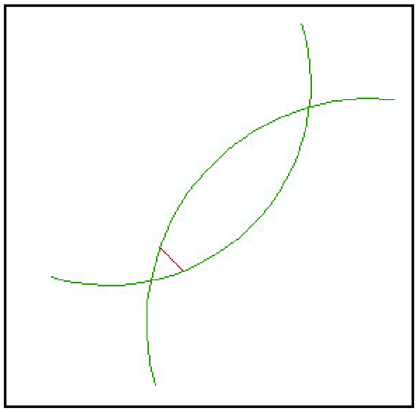


Figure 16
Introduction Of A Line Fillet Between Two Curves

➤ **2** Click at the first line, polyline or curve primitive to be intersected. Keep in mind that if a fillet is to be introduced, then a polyline should not be selected. If one is selected, the command will abort.

➤ **3** Confirm the selection.

➤ **4** Click at the second line, polyline or curve primitive to be intersected. Keep in mind that if a fillet is to be introduced, then a polyline should not be selected. If one is selected, the command will abort.

In selecting the second primitive, note that if:

- A linear fillet is to be introduced, or if
- A fillet is not to be introduced, but instead the point of intersection of the two primitives is to be created with, or without trimming the primitives,

then the click to select the second element should be made in the vicinity of the desired intersection.

➤ **5** Confirm the selection.

The user's input dialog box of Figure 10 is displayed.

➤ **6** In the data fields of the dialog box indicated below:

- Scroll down in the 1st data field, and select the "Yes" or "No" option denoting whether to trim or not the two selected primitives.
- Scroll down in the 2nd data field, and select the "Line", "Arc" or "None" option denoting the desired type of fillet, or absence thereof.
- Enter the fillet length along the first selected primitive, if a line fillet is to be inserted, or enter zero (0) if an arc fillet, or no fillet is to be introduced.

- Enter the fillet length along the second selected primitive, if a line fillet is to be inserted, or enter zero (0) if an arc fillet, or no fillet is to be introduced.

- Enter the radius of the fillet, if an arc fillet is to be inserted, or enter zero (0) if a line fillet, or no fillet is to be introduced.

➤ **7** Click at the OK button to accept the data, and continue with the operation, or click at the Cancel button to abort the command, and start from the beginning.

Note that if in the input dialog box above the selected fillet type option is:

- "Line" or "None", the command will display the result of its computations and terminate. The command will, however, remain active thereby enabling the user to select two additional primitives for processing.

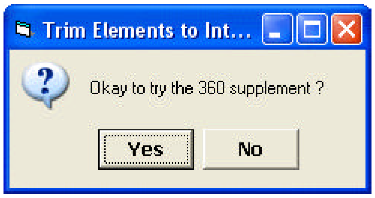
- "Arc", blue dots will be displayed at the centers of the possible fillet solutions. In this case, continue with the instructions below.

➤ **8** Click at the desired blue dot to select the center of the desired arc fillet.

The program selects one of the two possible tangent arcs, displays it, and asks the user (see Figure 17) whether its 360° supplement should be introduced, or not.

➤ **9** Click at the "Yes" button to do so, or click at the "No" button to accept the displayed arc. This step is repeated until a "No" response is given.

The command now terminates, and it remains active to select two additional primitives for processing.



Trim Elements to Int...

Okay to try the 360 supplement ?

Figure 17
Request To Display The 360° Arc Supplement

Summary

Although there are several occasions in which the *Trim Elements to Intersection and/or Introduce Fillet* tool may find use, its primary intent is to:

- Introduce fillets at the street right-of-way intersection of corner lots.

- Create sight triangles at roadway intersections.

- Augment the polyline intersection tools.

Regarding the last intent, it is noted that the other intersection tools require that an intersection be located within the confines of a polyline's graphic extent. This new tool does away with this requirement, and will generate an intersection outside the graphic extent of a polyline by extending the first or last polyline segment. The disadvantage of this new tool for intersecting polylines is that the user is forced to complete the dialog box of Figure 10.

In addition to the above disadvantage, there is one point of which the user should be aware when working with digitized polylines, or curves imported as polylines without a radius as discussed above. For example, if a polyline is to be intersected, and its last segment near the desired intersection is so small that it cannot be detected without zooming closer, it is possible that the resultant intersection point may not be the anticipated one. Furthermore, the direction of said end segment may be such that the intersection point may be created off the map display. In such a situation, the user may wish to create a circular arc using the Arc 3 tool to simulate the polyline, if this is indeed possible, or modify the polyline removing the redundant or extraneous vertices which are causing the problem.

Concluding, the *Trim Elements to Intersection and/or Introduce Fillet* tool is a highly functional command that once the user gets a feel for its operation, this tool will become a vital asset in a user's daily operation.

If you have a request for Command Of The Month, feel free to phone, fax or e-mail your request to The CEDRA Corporation.