
EGYSZERŰ AUTOCAD DXF FÁJL SZERKEZETE

FÁJL STRUKTÚRA

A DXF az AutoCAD egy sima ASCII szöveges állománya, meghatározott szerkezet szerint felépítve, ahol két egymást követő sor mindig összetartozik, az első sor egy típuskódot ad meg, a második az adott típushoz/elemhez tartozó értéket.

Alap felépítés:

A DXF fájl különböző szekciókból állhat, de ebből csak az ENTITIES kötelező, utána egy fájl vége jel van (EOF – end of file).

1. HEADER szekció – Fejléc. Általános információk, változók – opcionális
2. CLASSES szekció – opcionális
3. TABLES szekció – Különböző definíciók – opcionális
 - a. Linetype table (LTYPE)
 - b. Layer table (LAYER) – itt lehetne rétegeket definiálni, de új réteget az entitás szekción belül is létre lehet hozni, csak a réteg kóddal (8) és a réteg név megadásával, alap beállításokkal
 - c. Text Style table (STYLE)
 - d. View table (VIEW)
 - e. User Coordinate System table (UCS)
 - f. Viewport configuration table (VPORT)
 - g. Dimension Style table (DIMSTYLE)
 - h. Application Identification table (APPID)
4. BLOCKS szekció – Blokk definíciók – opcionális (ha nincs blokk a fájlban, akkor nem kell)
5. ENTITIES section – Rajzi elemek (pontok, vonalak stb.) – **KÖTELEZŐ!**
6. OBJECTS szekció – opcionális

Szekciók előtt meg kell adni, hogy most egy új szekció jön (0 kód, SECTION), majd utána a szekció nevét (2-es kód, név). Majd a tartalom után jön a szekció vége (0 kód, ENDSEC), legvégül fájl vége jel (0 kód, EOF). Pl.

```
0  
SECTION  
2  
ENTITIES  
...  
0  
ENDSEC  
...  
0  
EOF
```

További kódokat lásd az 1. táblázatban a szöveg után, néhány gyakran használt kódot vastaggal kiemelve. Néhány rajzi elem leírását az ENTITIES szekción belül lásd a 2. táblázatban. Egyszerű példák DXF fájlra:

PONT MEGADÁSA DXF FÁJLBAN

A következő példa egy pont rajzol a 35.23, 12.89 koordinátákhoz.

Simple_point.dxf:

| | |
|-----------------|----------------------------------|
| 0 | Szekció eleje |
| SECTION | |
| 2 | Rajzi elemek szekció következnek |
| ENTITIES | |
| 0 | Pont típusú elem |
| POINT | |
| 8 | Rétegmegadás kódja |
| pontok | 'pontok' a réteg neve |
| 10 | X koordináta |
| 35.23 | |
| 20 | Y koordináta |
| 12.89 | |
| 0 | Szekció vége |
| ENDSEC | |
| 0 | Fájl vége |
| EOF | |

A fenti fájl egy pontot tesz le, de a pont stílusa az alap beállításként használt kicsi pötty, ha ezt meg akarjuk változtatni, akkor kell a HEADER szekció is, és be kell állítani a pont stílusát a PDMODE változóban. PDMODE értéke lehet pl. 2 (+), 3 (X), 34 (⊕). Ehhez az előbbi előtt a HEADER szekciót is definiálni kell.

| | |
|-----------------|-----------------------------------|
| 0 | Szekció eleje |
| SECTION | |
| 2 | Fejléc szekció következnek |
| HEADER | |
| 9 | Változó definiálás jön |
| \$PDMODE | PDMODE változó |
| 70 | Változó értékadása (egész típusú) |
| 34 | értéke: 34 (⊕) |
| 0 | Szekció vége |
| ENDSEC | |

Ha különböző színekkel akarunk rajzolni, akkor a POINT, LINE vagy POLYLINE szekcióban is adhatunk meg színeket. A 62-es kód jelenti a színt, utána pedig az adott szín kódja pl az AutoCAD-ben: 1-piros, 2-sárga, 3-zöld, 4-világoskék, 5-sötétkék, 6-lila, 7-fekete/fehér, 8-szürke.

VONAL RAJZOLÁSA

A következő dxf fájl egy vonal a (0.0,0.0) kezdőpontból a (100,200) pontba.

| | |
|----------|-------------------------------|
| 0 | Szekció eleje |
| SECTION | |
| 2 | Rajzi elemek szekció következ |
| ENTITIES | |
| 0 | Vonal típusú elem |
| LINE | |
| 8 | Rétegmegadás kódja |
| vonal | 'vonal' a réteg neve |
| 10 | X koordináta (kezdőpont) |
| 0.0 | 0.0 |
| 20 | Y koordináta (kezdőpont) |
| 0.0 | 0.0 |
| 11 | X koordináta (végpont) |
| 100 | 100 |
| 21 | Y koordináta (végpont) |
| 200 | 200 |
| 0 | Szekció vége |
| ENDSEC | |
| 0 | Fájl vége |
| EOF | |

EGYSZERŰ SZÖVEG ÍRÁSA

írjuk ki a 'Haliho' szöveget a (3,8) pontba, szöveg rétegbe, 1 egység magassággal.

| | |
|----------|-------------------------------|
| 0 | Szekció eleje |
| SECTION | |
| 2 | Rajzi elemek szekció következ |
| ENTITIES | |
| 0 | Szöveg típusú elem |
| TEXT | |
| 8 | Rétegmegadás kódja |
| szöveg | 'szöveg' a réteg neve |
| 10 | X koordináta |
| 3.0 | 3.0 |
| 20 | Y koordináta |
| 8.0 | 8.0 |
| 40 | Szöveg magassága |
| 1.0 | 1 rajzi egység |
| 1 | Szöveg tartalma: |
| Haliho! | 'Haliho' |
| 0 | Szekció vége |
| ENDSEC | |
| 0 | Fájl vége |
| EOF | |

VONALLÁNC RAJZOLÁSA

Vonallánc a (0.0,0.0) kezdőpontból a (34.5,23.2) ponton át a (56.0,16.2) pontba.

| | |
|-----------------|---------------------------------|
| 0 | Szekció eleje |
| SECTION | |
| 2 | Rajzi elemek szekció következik |
| ENTITIES | |
| 0 | Vonallánc típusú elem |
| POLYLINE | |
| 8 | Rétegmegadás kódja |
| vonal | 'von' a réteg neve |
| 66 ¹ | Csomópontok következnek |
| 1 | értéke mindig 1 |
| 0 | Csomópont következik |
| VERTEX | |
| 8 ² | Rétegmegadás kódja |
| vonal | 'von' a réteg neve |
| 10 | X koordináta (kezdőpont) |
| 0.0 | 0.0 |
| 20 | Y koordináta (kezdőpont) |
| 0.0 | 0.0 |
| 0 | Csomópont következik |
| VERTEX | |
| 8 | Rétegmegadás kódja |
| vonal | 'von' a réteg neve |
| 10 | X koordináta (kezdőpont) |
| 34.5 | 34.5 |
| 20 | Y koordináta (kezdőpont) |
| 23.2 | 23.2 |
| 0 | Csomópont következik |
| VERTEX | |
| 8 | Rétegmegadás kódja |
| vonal | 'von' a réteg neve |
| 10 | X koordináta (kezdőpont) |
| 56.0 | 56.0 |
| 20 | Y koordináta (kezdőpont) |
| 16.2 | 16.2 |
| 0 | Csomópontok vége |
| SEQEND | |
| 0 | Szekció vége |
| ENDSEC | |
| 0 | Fájl vége |
| EOF | |

¹ A 66-os kód jelöli, hogy csomópontok következnek, ez nem minden CAD programnál kötelező elem, de a tanszéki AutoCAD MAP 2008 változatnál igen.

² Lásd mint előbb. Nem minden CAD programnál kötelező minden csomópontnál megadni a réteg nevét is, de az AutoCAD MAP 2008 esetében igen.

A FENTI RAJZI ELEMÉK EGY FÁJLBAN

| | |
|-------------------------------|----------|
| <i>simple_dxf_example.dxf</i> | 8.0 |
| | 40 |
| 0 | 1.0 |
| SECTION | 1 |
| 2 | Haliho! |
| HEADER | 0 |
| 9 | POLYLINE |
| \$PDMODE | 8 |
| 70 | vonal |
| 34 | 66 |
| 0 | 1 |
| ENDSEC | 0 |
| 0 | VERTEX |
| SECTION | 8 |
| 2 | vonal |
| ENTITIES | 10 |
| 0 | 0.0 |
| POINT | 20 |
| 8 | 0.0 |
| pontok | 0 |
| 10 | VERTEX |
| 35.23 | 8 |
| 20 | vonal |
| 12.89 | 10 |
| 0 | 34.5 |
| LINE | 20 |
| 8 | 23.2 |
| vonal | 0 |
| 10 | VERTEX |
| 0.0 | 8 |
| 20 | vonal |
| 0.0 | 10 |
| 11 | 56.0 |
| 100 | 20 |
| 21 | 16.2 |
| 200 | 0 |
| 0 | SEQEND |
| TEXT | 0 |
| 8 | ENDSEC |
| szöveg | 0 |
| 10 | EOF |
| 3.0 | |
| 20 | |

1. TÁBLÁZAT ÁLTALÁNOS KÓDOK JELENTÉSE A DXF FÁJLBAN³

| | |
|-----|---|
| 0 | Text string indicating the entity type (fixed) |
| 1 | The primary text value for an entity |
| 2 | Name (attribute tag, block name, and so on) |
| 3-4 | Other textual or name values |
| 5 | Entity handle. Text string of up to 16 hexadecimal digits (fixed) |
| 6 | Line type name (fixed) |
| 7 | Text style name (fixed) |

³ lásd részletesebben: <http://www.autodesk.com/techpubs/autocad/acadr14/dxf/index.htm>

| | |
|---------------|--|
| 8 | Layer name (fixed) |
| 9 | Variable name identifier (used only in HEADER section of the DXF file) (Example: \$PDMODE – point style) |
| 10 | Primary point. This is the start point of a line or text entity, center of a circle, and so on. DXF: X value of the primary point (followed by Y and Z value codes 20 and 30) |
| 11-18 | Other X coordinates |
| 20 | Primary Y coordinate. 2n values always correspond to 1n values and immediately follow them in the file |
| 21-28 | Other Y coordinates |
| 30 | Primary Z coordinate. 3n values always correspond to 1n and 2n values and immediately follow them in the file |
| 31-37 | Other Z coordinates |
| 38 | DXF: entity's elevation if nonzero. |
| 39 | Entity's thickness if nonzero (fixed) |
| 40-48 | Floating-point values (text height, scale factors, etc.) |
| 49 | Repeated floating-point value. Multiple 49 groups may appear in one entity for variable-length tables (such as the dash lengths in the LTYPE table). A 7x group always appears <i>before</i> the first 49 group to specify the table length. |
| 50-58 | Angles |
| 62 | Color number (fixed) |
| 66 | "Entities follow" flag (fixed) |
| 67 | Space--that is, model or paper space (fixed) |
| 68 | Identifies whether viewport is on but fully off screen, is not active, or is off |
| 69 | Viewport identification number |
| 70-78 | Integer values such as repeat counts, flag bits, or modes |
| 210, 220, 230 | X, Y, and Z components of extrusion direction (fixed) |
| 999 | Comments |
| etc. | |

2. TÁBLÁZAT RAJZI ELEMÉK (ENTITIES SECTION)⁴

| LINE | 10, 20, 30 (start point), 11, 21, 31 (end point). | | | | | | | | | | | | |
|----------------|--|----------------|---------|---|---|---|------------------------------------|---|-------------------------------------|---|-----------------------|----|--|
| POINT | 10, 20, 30 (point), 50 (angle of X axis for the UCS in effect when the Point was drawn -optional 0, for use when PDMODE is nonzero). PDMODE:34 | | | | | | | | | | | | |
| CIRCLE | 10, 20, 30 (center), 40 (radius). | | | | | | | | | | | | |
| ARC | 10, 20, 30 (center), 40 (radius), 50 (start angle), 51 (end angle). | | | | | | | | | | | | |
| POLYLINE | 66 ("vertices follow flag"), 70 (Polyline flags), 40 (default starting width), 41 (default ending width), 71 and 72 (poly- gon mesh M and N vertex counts -optional 0), 73 and 74 (smooth surface M and N densities -optional 0), 75 (smooth surface type - optional 0). The default widths apply to any vertex that doesn't supply widths (see below). The "vertices follow" flag is always 1, indicating that a series of VERTEX entities is expected to follow the POLYLINE, terminated by a sequence end (SEQEND) entity. The "polyline flags" group is a bit-coded field with bits defined as follows: <table border="0"> <thead> <tr> <th>Flag bit value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>This is a closed Polyline (or a polygon mesh closed in the M direction)</td> </tr> <tr> <td>2</td> <td>Curve-fit vertices have been added</td> </tr> <tr> <td>4</td> <td>Spline-fit vertices have been added</td> </tr> <tr> <td>8</td> <td>This is a 3D Polyline</td> </tr> <tr> <td>16</td> <td>This is a 3D polygon mesh. Group 75 indicates the smooth surface type, as follows:</td> </tr> </tbody> </table> 0 = no smooth surface fitted 5 = quadratic B-spline surface | Flag bit value | Meaning | 1 | This is a closed Polyline (or a polygon mesh closed in the M direction) | 2 | Curve-fit vertices have been added | 4 | Spline-fit vertices have been added | 8 | This is a 3D Polyline | 16 | This is a 3D polygon mesh. Group 75 indicates the smooth surface type, as follows: |
| Flag bit value | Meaning | | | | | | | | | | | | |
| 1 | This is a closed Polyline (or a polygon mesh closed in the M direction) | | | | | | | | | | | | |
| 2 | Curve-fit vertices have been added | | | | | | | | | | | | |
| 4 | Spline-fit vertices have been added | | | | | | | | | | | | |
| 8 | This is a 3D Polyline | | | | | | | | | | | | |
| 16 | This is a 3D polygon mesh. Group 75 indicates the smooth surface type, as follows: | | | | | | | | | | | | |

⁴ lásd részletesebben: <http://www.autodesk.com/techpubs/autocad/acadr14/dxf/index.htm>

| | <p>6 = cubic B-spline surface 8 = Bezier surface</p> <p>32 The polygon mesh is closed in the N direction</p> | | | | | | | | | | | | | | | | | | | | |
|----------------|--|----------------|---------|---|---------------------------------------|---|---|-------|---------------------------------|---|---|----|-------------------------------------|----|-------------------------|----|--|---|-----------------------------------|---|---|
| VERTEX | <p>10, 20, 30 (location), 40 (starting width -optional, see above), 41 (ending width -optional, see above), 42 (bulge), 70 (vertex flags), 50 (curve fit tangent direction -optional). The bulge is the tangent of 1/4 the included angle for an arc segment, made negative if the arc goes clockwise from the start point to the end point; a bulge of 0 indicates a straight segment, and a bulge of 1 is a semicircle. The meanings of the bit-coded "vertex flags" are shown in the following table.</p> <table> <thead> <tr> <th>Flag bit value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Extra vertex created by curve fitting</td> </tr> <tr> <td>2</td> <td>Curve fit tangent defined for this vertex. A curve fit tangent direction of 0 may be omitted from the DXF output, but is significant if this bit is set.</td> </tr> <tr> <td>4</td> <td>Unused (never set in DXF files)</td> </tr> <tr> <td>8</td> <td>Spline vertex created by spline fitting</td> </tr> <tr> <td>16</td> <td>Spline frame control point</td> </tr> <tr> <td>32</td> <td>3D Polyline vertex</td> </tr> <tr> <td>64</td> <td>3D polygon mesh vertex</td> </tr> </tbody> </table> | Flag bit value | Meaning | 1 | Extra vertex created by curve fitting | 2 | Curve fit tangent defined for this vertex. A curve fit tangent direction of 0 may be omitted from the DXF output, but is significant if this bit is set. | 4 | Unused (never set in DXF files) | 8 | Spline vertex created by spline fitting | 16 | Spline frame control point | 32 | 3D Polyline vertex | 64 | 3D polygon mesh vertex | | | | |
| Flag bit value | Meaning | | | | | | | | | | | | | | | | | | | | |
| 1 | Extra vertex created by curve fitting | | | | | | | | | | | | | | | | | | | | |
| 2 | Curve fit tangent defined for this vertex. A curve fit tangent direction of 0 may be omitted from the DXF output, but is significant if this bit is set. | | | | | | | | | | | | | | | | | | | | |
| 4 | Unused (never set in DXF files) | | | | | | | | | | | | | | | | | | | | |
| 8 | Spline vertex created by spline fitting | | | | | | | | | | | | | | | | | | | | |
| 16 | Spline frame control point | | | | | | | | | | | | | | | | | | | | |
| 32 | 3D Polyline vertex | | | | | | | | | | | | | | | | | | | | |
| 64 | 3D polygon mesh vertex | | | | | | | | | | | | | | | | | | | | |
| SEQUEND | <p>No fields. This entity marks the end of vertices (VERTEX type name) for a Polyline, or the end of Attribute entities (ATTRIB type name) for an INSERT entity that has Attributes (indicated by 66 group present and nonzero in INSERT entity).</p> | | | | | | | | | | | | | | | | | | | | |
| TEXT | <p>10, 20, 30 (insertion point), 40 (height), 1 (text value), 50 (rotation angle -optional 0), 41 (relative X scale factor -optional 1), 51 (obliquing angle -optional 0), 7 (text style name -optional "STANDARD"), 71 (text generation flags -optional 0), 72 (justification type -optional 0), 11, 21, 31 (alignment point -optional, appears only if 72 group is present and nonzero).</p> <p>The "text generation flags" are a bit-coded field with meanings as follows:</p> <table> <thead> <tr> <th>Flag bit value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>Text is backwards (mirrored in X)</td> </tr> <tr> <td>4</td> <td>Text is upside down (mirrored in Y)</td> </tr> </tbody> </table> <p>The "justification type" value (not bit-coded) indicates the text justification style used on this entity, as shown in the following table.</p> <table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Text is left justified</td> </tr> <tr> <td>1</td> <td>Text is centered along its baseline</td> </tr> <tr> <td>2</td> <td>Text is right justified</td> </tr> <tr> <td>3</td> <td>Text is aligned between two points (height varies)</td> </tr> <tr> <td>4</td> <td>Text is "middle" (fully) centered</td> </tr> <tr> <td>5</td> <td>Text is fit between two points (width varies)</td> </tr> </tbody> </table> <p>If the justification is anything other than 0 (left justified), 11, 21, and 31 groups will also appear in the entity to specify the alignment point of the text (center, right-most, or second alignment point).</p> <p>DXFOUT handles ASCII control characters in text strings by expanding the character into a "^" (caret) followed by the appropriate letter. For example, an ASCII Control-G (BEL, decimal code 7) is output as "^G". If the text itself contains a caret character, it is expanded to "^ " (caret, space). DXFIN performs the complementary conversion.</p> | Flag bit value | Meaning | 2 | Text is backwards (mirrored in X) | 4 | Text is upside down (mirrored in Y) | Value | Meaning | 0 | Text is left justified | 1 | Text is centered along its baseline | 2 | Text is right justified | 3 | Text is aligned between two points (height varies) | 4 | Text is "middle" (fully) centered | 5 | Text is fit between two points (width varies) |
| Flag bit value | Meaning | | | | | | | | | | | | | | | | | | | | |
| 2 | Text is backwards (mirrored in X) | | | | | | | | | | | | | | | | | | | | |
| 4 | Text is upside down (mirrored in Y) | | | | | | | | | | | | | | | | | | | | |
| Value | Meaning | | | | | | | | | | | | | | | | | | | | |
| 0 | Text is left justified | | | | | | | | | | | | | | | | | | | | |
| 1 | Text is centered along its baseline | | | | | | | | | | | | | | | | | | | | |
| 2 | Text is right justified | | | | | | | | | | | | | | | | | | | | |
| 3 | Text is aligned between two points (height varies) | | | | | | | | | | | | | | | | | | | | |
| 4 | Text is "middle" (fully) centered | | | | | | | | | | | | | | | | | | | | |
| 5 | Text is fit between two points (width varies) | | | | | | | | | | | | | | | | | | | | |