

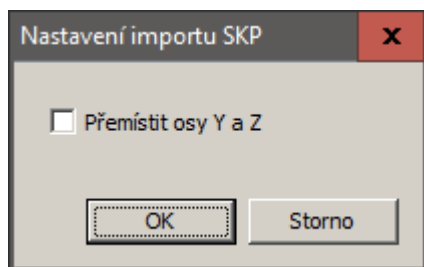
DesignCAD 3D Max 2016 Release Notes

20. červenec, 2016

Díky za nákup programu DesignCAD 3D Max 2016. Zde je detailní pohled na to, co je v této verzi nového.

Nová funkce Import/Export (New Import/Export feature)

Import souborů SketchUp (SketchUp Import) – DesignCAD 2016 nyní může importovat soubory výkresů a modelů programu SketchUp. Podpora je až do verze SketchUp 2015. Při importu souboru se každý samostatný předmět importuje do DesignCADu jako samostatná entita **Povrch tělesa** (Solid Surface). Export byl možný již dříve.



Dialog importu souborů SketchUp pro nastavení záměny souřadných os zahrnuje nyní volbu **Přemístit osy Y a Z** (Transpose Y and Z axis). Pokud je políčko zaškrtnuto, je umožněno objekty SketchUp importovat do DesignCADu "pravou stranou nahoru". Tato možnost je k dispozici proto, neboť svislá osa u programu DesignCAD je Y, zatímco SketchUp má svislou osu Z. Nebude-li tato volba zaškrtnuta, budou importované objekty programu SketchUp pootočeny.

Správce vložených prvků (Insert Manager)

Soubor / Správce vložených prvků (File/Insert Manager) je nový příkaz, který umožňuje uživateli kontrolovat a spravovat všechny **bloky, symboly a obrazové soubory**, na které jsou reference, nebo jsou vloženy ve výkresu.

Jestliže příkaz spustíte, zobrazí se dialogové okno se třemi záložkami, samostatně pro bloky, obrázky a symboly.

Bloky (Blocks)

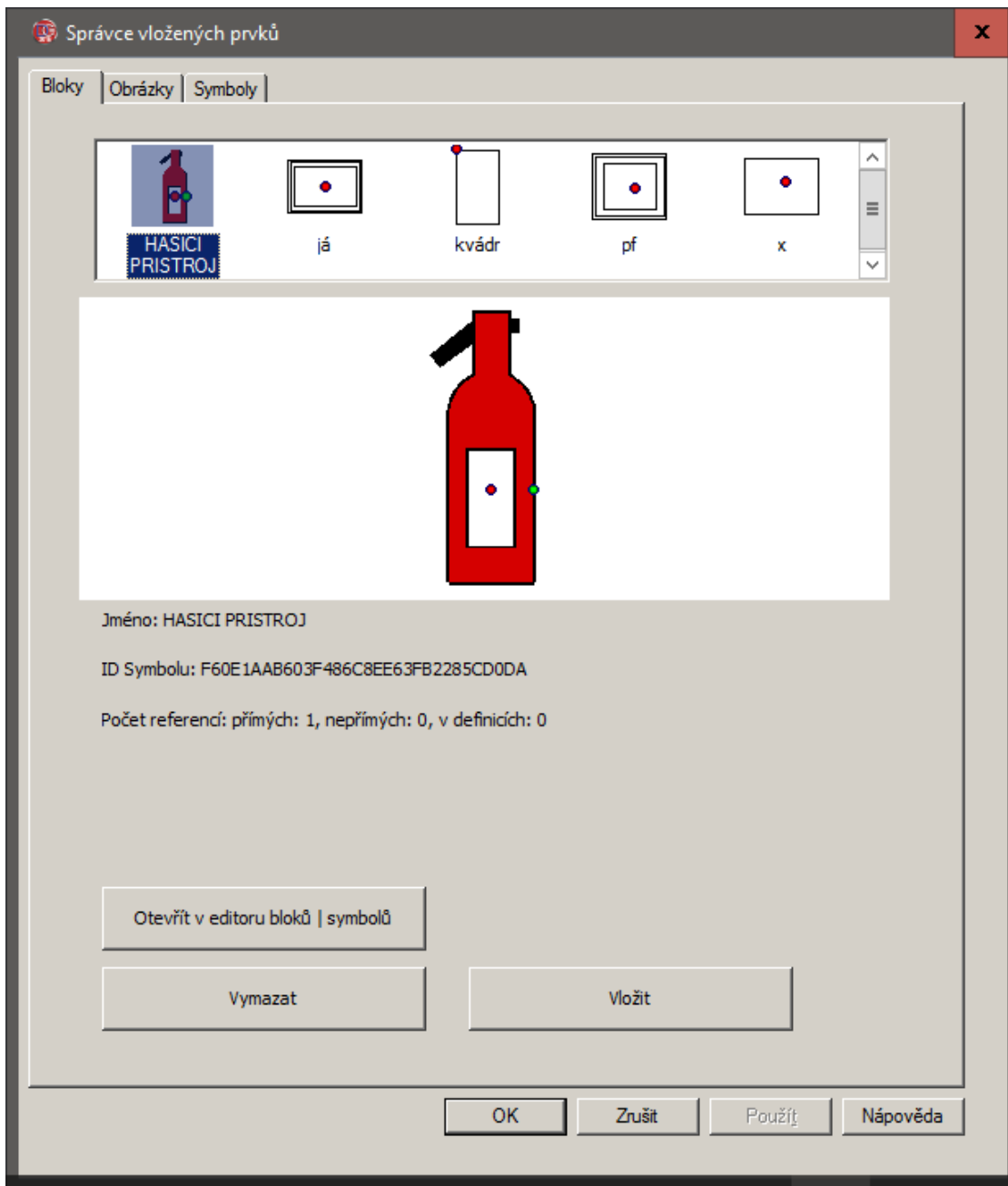
Záložka **Bloky** (Blocks), **Správce vložených prvků** (Insert Manager) zobrazuje vizuální seznam všech bloků, které jsou obsaženy ve výkresu. Je-li jeden blok zvýrazněn kliknutím na něj, zobrazí se větší náhled pod seznamem. Pod náhledem je více informací o bloku, ... jeho jméno, identifikační číslo a počet referencí.

Kliknutím pravým tlačítkem myši na seznam ikon se zobrazí kontextové menu pro třídění bloků. Bloky mohou být setříděny podle názvu, ID nebo počtu referencí.

Kliknutím pravým tlačítkem myši na řádek **Jméno** (Name) vám bude umožněno zkopírování jména do schránky.

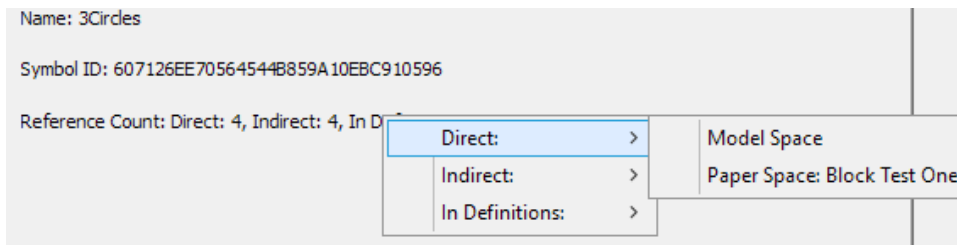
Kliknutím pravým tlačítkem myši na řádek **ID Symbolu** (Symbol ID) vám bude umožněno kopírování ID do schránky (například při použití v makru nebo COM programu).

Počet referencí (Reference Count): řádek ukazuje počet referencí tři různých hodnot: **přímých** (Direct), **nepřímých** (Indirect), a **v definicích** (In Definitions). Počet **přímých** je počet, kolikrát je blok součástí výkresu jako samostatný blok (tj. ne jako součást jiného bloku nebo symbolu). Počet **nepřímých** je počet, kolikrát je blok obsažen v jiných blocích nebo symbolech, které jsou ve výkresu. Počet **v definicích** je počet, kolik definic bloků nebo symbolů patří k bloku jako součást této definice.



Klepněte pravým tlačítkem myši na **Počet referencí** (Reference Count); zobrazí se kontextové menu s detailnějším zobrazením počtu referencí dle kategorie: **přímých** (Direct), **nepřímých** (Indirect), a **v definicích** (In Definitions).

Rozšíření kategorie **přímých** (Direct) uvádí, zda blok je používán v **Modelovém prostoru** výkresu a nebo jedním nebo více rozvržení v režimu **Šablony papíru** (Paper space). Rozšíření kategorie **nepřímých** (Indirect) ukáže, které symboly či bloky, které obsahují tento blok, jsou aktivní. Rozšíření kategorie **v definicích** (In Definitions) budou uvádět, které symboly či bloky obsahují tento blok nebo zda existují nějaké aktivní odkazy na tyto symboly / bloky.



Záložka **Bloky** (Blocks), **Správce vložených prvků** (Insert Manager) nabízí tři editační operace: **Otevřít v editoru bloků | symbolů** (Open in Block Editor), **Vymazat** (Delete) a **Vložit** (Insert).

Kliknutím na tlačítko **Vložit** (Insert) bude spuštěn příkaz Vložit blok a ukáže se původní dialogové okno **Vložit blok** (Block Insert) s již vybraným označeným blokem.

Kliknutím na tlačítko **Vymazat** (Delete) vám umožní odstranit všechny výskyty zvýrazněného bloku z výkresu, stejně jako definici bloku.

Kliknutím na tlačítko **Otevřít v editoru bloků | symbolů** (Open in Block Editor) se otevře speciální pracovní plocha s omezenou nabídkou příkazů. V Editoru bloků můžete provést téměř veškeré změny v definici bloku, které si přejete, bez porušení jiných prvků výkresu. Viz **Editor bloků** níže.

Obrázky (Images)

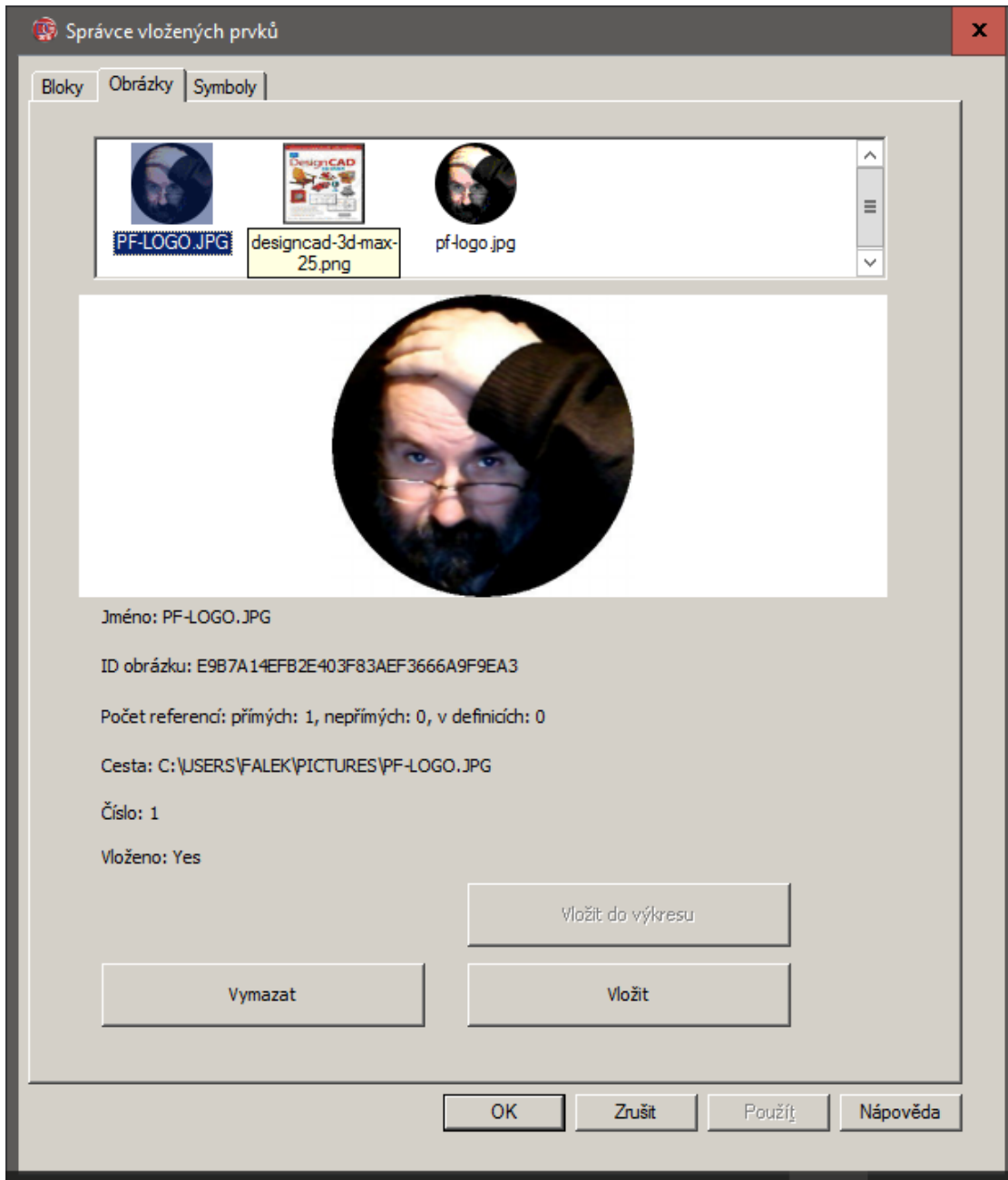
Záložka **Obrázky** (Images) okna **Správce vložených prvků** (Insert Manager) je podobná záložce **Bloky** (Blocks), a zobrazuje všechny rastrové obrázky, které byly do výkresu vloženy. Většina z možností pro **Obrázky** (Images) je stejná jako pro **Bloky** (Blocks). Obrázky navíc obsahují pole **Cesta** (Path), pole **Číslo** (Index) a pole **Vloženo** (Embedded). To jsou další možnosti třídění, pokud klepnete pravým tlačítkem myši na miniaturu v horní části okna.

Pole **Cesta** (Path) označuje úplnou cestu a název původního obrázku. Pro vložené obrázky platí, že se původní soubor již nemusí nacházet na odkazovaném místě. Kliknutí pravým tlačítkem myši na pole Cesta umožňuje zkopírovat úplnou cestu a název souboru do schránky pro pozdější použití.

Pole **Číslo** (Index) udává index obrázku v seznamu obrázků, tj. relativní pořadí, ve kterém byl obrázek přidán do výkresu.

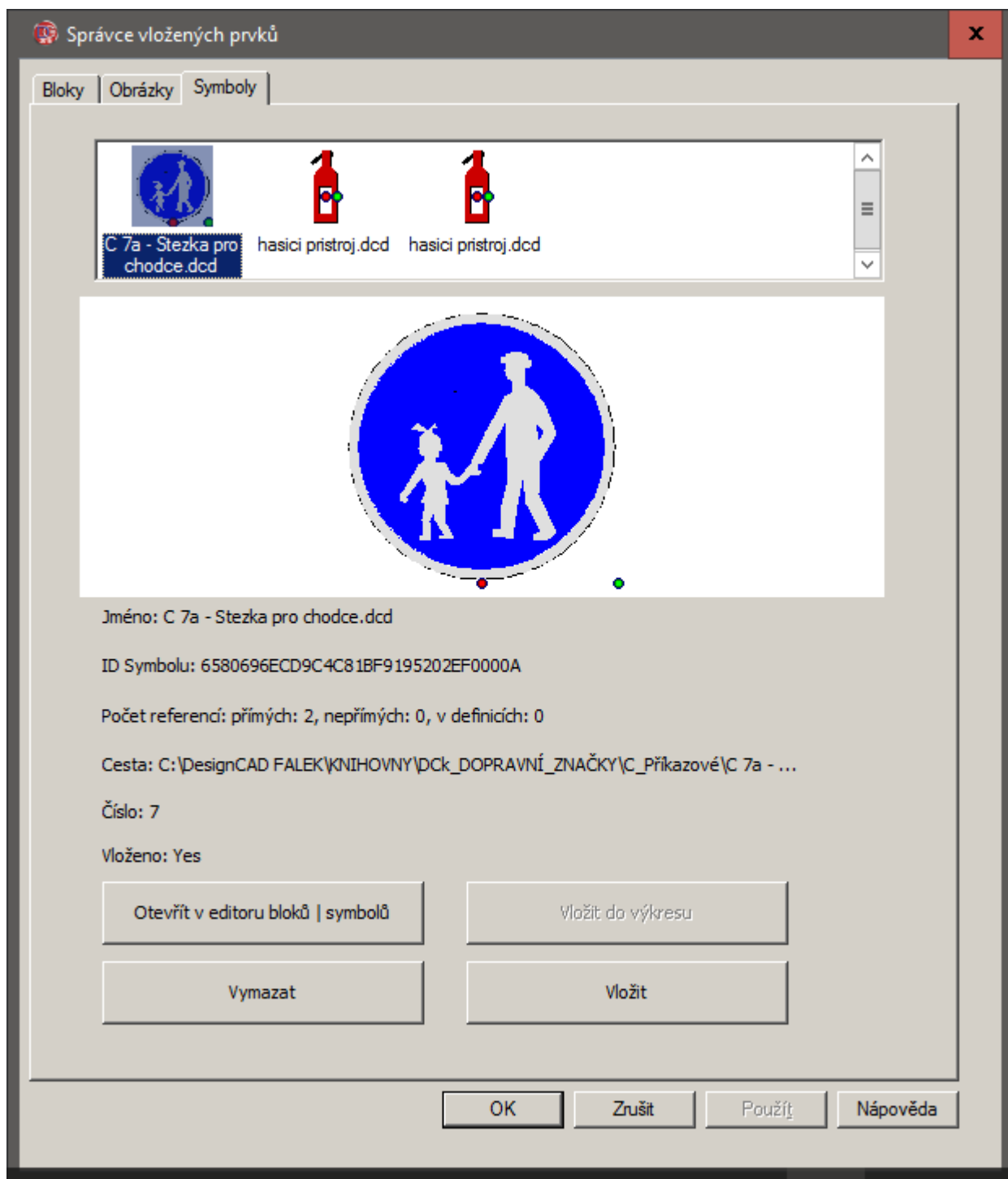
Pole **Vloženo** (Embedded) označuje, zda je obrázek součástí výkresu – **Yes** (Ano), nebo je pouze odkazem na externí soubor **No** (Ne). (PF: Tuto odpověď uvádí program anglicky, nelze provést překlad). *Bloky jsou vždy vloženy, takže toto tlačítko je na stránce bloků skryté.*

Je-li obrázek zvýrazněný ve vizuálním seznamu vložený pouze jako odkaz na externí soubor, je zvýrazněno tlačítko **Vložit do výkresu** (Embed in Drawing). Kliknutí na toto tlačítko umožňuje, aby odkazovaný obrázek byl vložen do výkresu, takže externí soubor s obrázkem již není nutný. Tím se změní i pole **Vloženo:** (Embedded:) z No (Ne) na Yes (Ano). Všimněte si, že původní soubor s obrázkem musí být přítomen v zadané cestě, aby to fungovalo. *(Asi tu je drobná chyba. Ono to vypadá když změna proběhne, No se změní na Yes a tlačítko Vložit do výkresu se znepřístupní, ale ve skutečnosti se to nestane. Po novém otevření okna Správce vložených prvků je vidět původní stav. Doufám, že časem to výrobce opraví a objeví se informace, že to není možné a k vizuální změně nedojde.) V originál textu je to dokonce popsáno jaksí obráceně!*



Symboly (Symbols)

Záložka **Symboly** (Symbols) **Správce vložených prvků** (Insert Manager) má stejné vlastnosti jako záložka **Obrazy** (Images) a také společné tlačítko **Otevřít v editoru bloků | symbolů** (Open in Block Editor), stejně jako na záložce **Bloky** (Blocks).



Editor Bloků | Symbolů (Block Editor)

Po označení symbolu nebo bloku v okně **Správce vložených prvků** (Insert Manager) klepněte na tlačítko **Otevřít v editoru bloků | symbolů** (Open in Block Editor). Blok či Symbol se zobrazí v samostatném okně bez ostatních objektů původního výkresu. Pracovní plocha tohoto režimu má poněkud omezenou sadu příkazů; není přístup k načítání symbolů nebo bloků přes menu Soubor (File), chybí příkaz pro Tisk (Print), nejsou žádné možnosti uložení symbolu nebo bloku jako externí soubor. Není přístup do speciálních režimů, jako je animační režim, režim průchodu a režim šablony papíru. Chybí přístup k příkazům v nabídce Nápověda (Help) nebo Okno (Window) a příkaz Nastavit / Možnosti (Options / Options) též chybí. Kromě těchto výjimek je většina ostatních kreslicích a editačních příkazů v Editoru bloků k dispozici.

Pokud potřebujete přidat symbol či blok k tomu, který chcete upravovat, nelze to provést příkazem z menu Soubor v **Editoru bloků | symbolů** (Block Editor). Místo toho nejprve vyberte blok nebo symbol, který má být přidán v hlavním okně výkresu a zkopírujte ho do schránky. Potom můžete otevřít symbol či blok v **Editoru bloků | symbolů** a přidejte k němu obsah schránky (Ctrl+V). *Je třeba dbát na to, abyste nepřidávali do bloku či symbolu jeho vlastní kopii - to způsobuje cyklický odkaz a může vést k řadě složitých problémů.*

Editor bloků | symbolů opustíte příkazem přes menu Soubor. Chcete-li změny provedené v Editoru bloků | symbolů zrušit, zvolte Soubor / **Zrušit změny, ukončit Editor bloků | symbolů a návrat do Správce ...** (File/Cancel Changes and Exit). Chcete-li změny provedené v Editoru bloků | symbolů uložit, zvolte Soubor / **Uložit změny, ukončit Editor bloků | symbolů a návrat do Správce ...** (File/Save Changes and Exit). Příkazem se vrátíte do otevřeného okna Správce vložených prvků.

Správce vložených prvků (Insert Manager) a **Editor bloků | symbolů** (Block Editor) jsou k dispozici jak v režimu **Modelového prostoru** (Model Space) i režimu **Šablony papíru** (Paper Space). Může však nastat případ, kdy v režimu **Šablony papíru** (Paper Space) půjdou některé bloky či symboly upravovat obtížně, vzhledem k omezením kreslicí plochy v tomto režimu. Pokud budete mít problémy upravit konkrétní blok či symbol v režimu **Šablony papíru** (Paper Space), zkuste se přepnout do **Modelového prostoru** (Model Space) a upravit jej tam. *(PF doporučuje vytvářet projekt zásadně v režimu Modelového prostoru (Výkresového prostoru; běžná pracovní plocha) a v režimu Šablony papíru případně přidávat pouze rámečky s nedefinovanými vlastnostmi (měřítka, stínování, ...).*

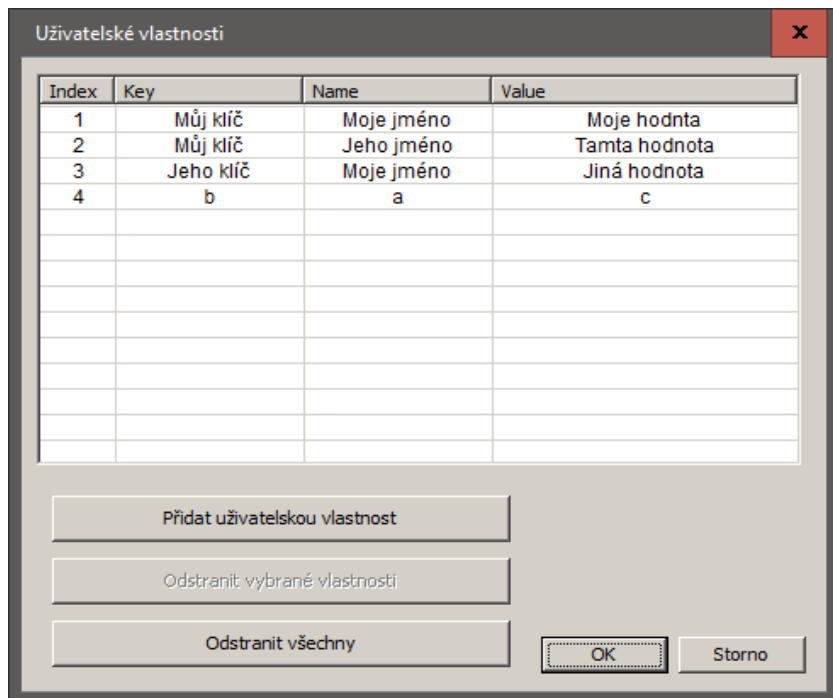
Uživatelské vlastnosti (Custom Properties)

DesignCAD nyní podporuje přidávání **Uživatelských vlastností** celému výkresu i jednotlivým entitám výkresu.

Uživatelské vlastnosti jsou části textu, které jsou identifikovány pomocí kombinace polí **Klíč** (Key) a **Jméno** (Name). Jedná se o způsob přidávání "neviditelné" textové informace k výkresu nebo k entitě výkresu. Tyto uživatelské vlastnosti lze prohlížet a upravovat ručně pomocí dialogu **Uživatelské vlastnosti** (Custom Properties), nebo kódem makra BasicCADu či programu automatizace OLE.

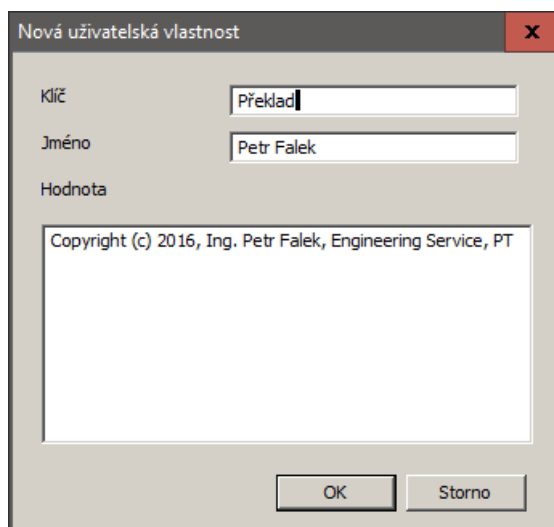
Uživatelské vlastnosti výkresu (Drawing Custom Properties) jsou přístupné z menu **Soubor / Uživatelské vlastnosti výkresu** (File/Document Custom Properties).

Uživatelské vlastnosti entity (Entity Custom Properties) jsou přístupné pomocí tlačítka **Uživatelské vlastnosti** (Custom Properties) v **Info panelu** (Info Box). Oběma způsoby otevřete editor Uživatelské vlastnosti:



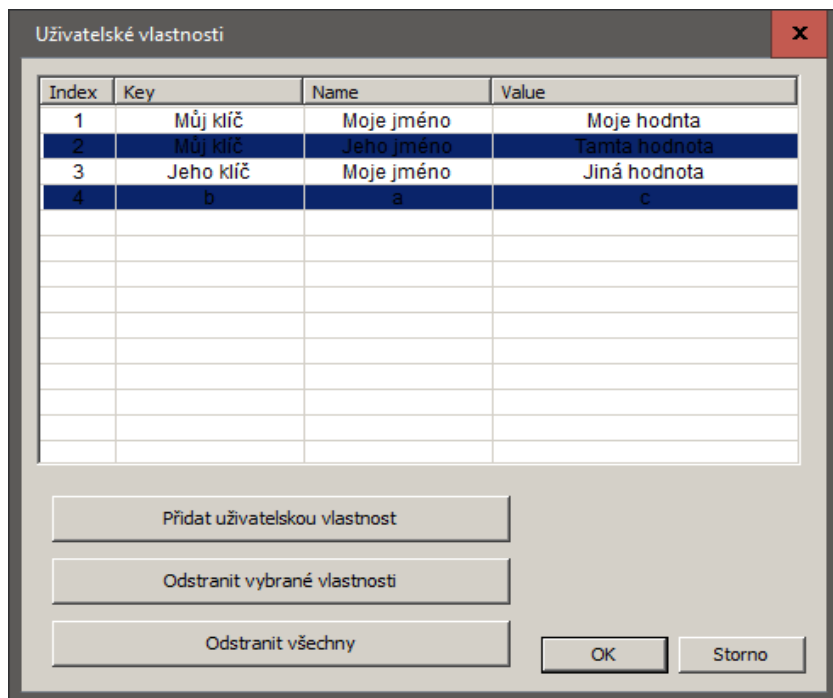
Dialog ukazuje čtyři pole: **Číslo** (Index), **Klíč** (Key), **Jméno** (Name) a **Hodnota** (Value). **Číslo** (Index) je pořadové číslo, které je automaticky přidáno každé uživatelské vlastnosti. **Klíč** (Key) je prvním ze dvou polí, které jednoznačně identifikuje každou uživatelskou vlastnost. **Jméno** (Name) je druhé povinné pole určené k identifikaci uživatelské vlastnosti. Pole **Hodnota** (Value) je "obsah" uživatelské vlastnosti.

Pomocí tlačítka **Přidat uživatelskou vlastnost** (Add Custom Property) otevřete dialog **Nová uživatelská vlastnost** (New Custom Property), kde zadáváte uvedené hodnoty.



Žádné dvě uživatelské vlastnosti pro daný výkres nebo entitu výkresu nemohou mít stejnou kombinaci **Klíč / Jméno** (Key / Name). Dva různé výkresy mohou mít stejnou kombinaci Klíč / Jméno a výkres a jedna či více entit v něm obsažených mohou mít stejnou kombinaci Klíč / Jméno; omezení platí pouze pro sadu uživatelských vlastností použitých v konkrétním výkresu nebo konkrétní entitě výkresu.

Jednu uživatelskou vlastnost lze vybrat kliknutím levého tlačítka myši na určitý řádek. Více řádků lze vybrat pomocí klávesy Ctrl + levého tlačítka myši, nebo klávesy Shift + levého tlačítka myši, nebo klávesy Shift + šipky dolů/nahoru. Jakmile jsou jedna nebo více uživatelských vlastností vybrány, mohou být odstraněny pomocí tlačítka **Odstranit vybrané vlastnosti** (Delete Selected Properties). Tlačítko **Odstranit všechny** (Delete All) vymaže všechny uživatelské vlastnosti, které byly přidány pomocí tohoto editoru.



Mějte však na paměti, že uživatelské vlastnosti, které byly přidány přes BasicCAD nebo OLE automatizaci, není možné upravovat. Takové uživatelské vlastnosti bude mít šedé pozadí a lze je upravit nebo odstranit pouze pomocí BasicCAD nebo OLE automatizace.

Přístup k uživatelským vlastnostem přes BasicCAD a OLE automatizaci je popsán níže (*pouze anglicky*).

Uživatelské vlastnosti entit výkresu jsou duplikovány, pokud jsou tyto objekty zkopírovány a vloženy přes schránku, kopírovány, kopírovány do pole, rozříznuty nebo rozloženy na více částí. (Poznámka: neupravitelné vlastnosti nejsou tímto způsobem duplikovány.)

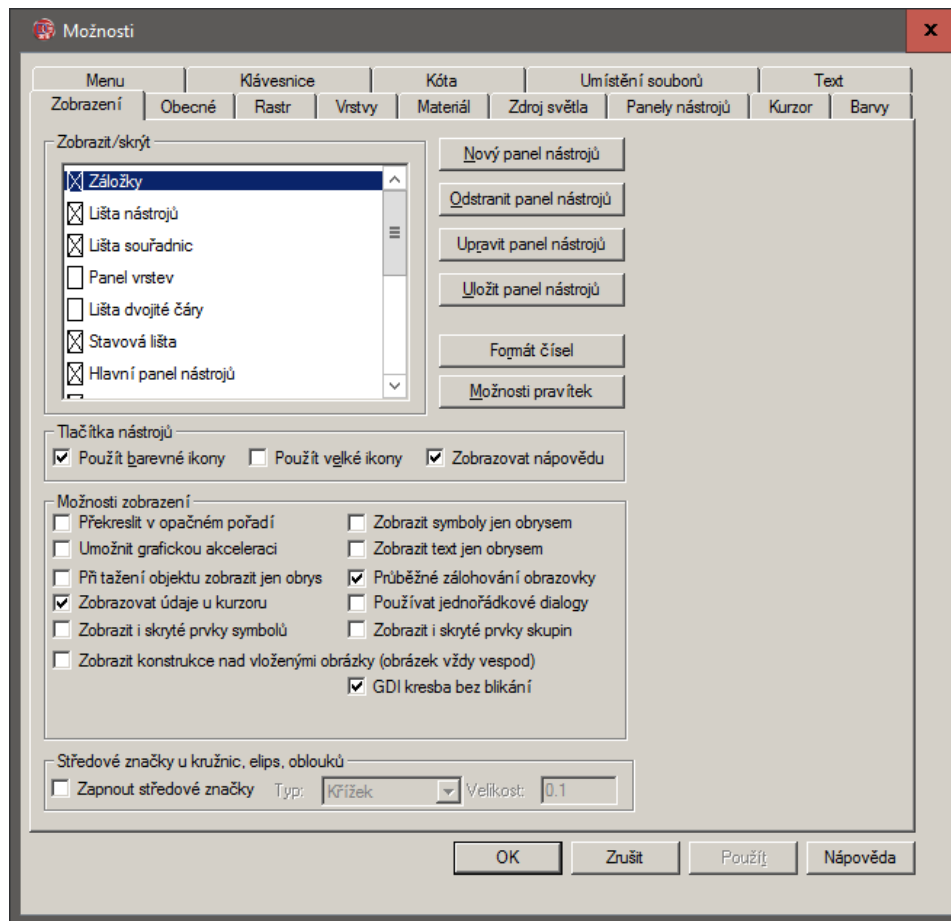
Použitím příkazu **Rozložit** (Explode) na vybranou sloučenou entitu výkresu (například kótu nebo mřížku), která obsahuje uživatelské vlastnosti, nebudou rozložené kusy již obsahovat tyto vlastnosti. Stejně tak příkaz **Spojit čáry** (Combine Lines) způsobí, že výsledný objekt ztratí uživatelské vlastnosti, které byly jejich součástí.

Uživatelské vlastnosti výkresů jsou ukládány s výkresem při použití příkazů **Uložit** (Save), **Uložit jako ..** (Save As), **Uložit kopii ..** (Save a Copy) nebo **Uložit jako symbol** (Save as Symbol); nicméně nejsou převedeny do hostitelského výkresu, pokud použijete příkaz **Načíst symbol** (Load Symbol).

Ostatní změny v DesignCADu 2016

GDI kresba bez blikání (Flicker-Free GDI Draw)

Do panelu **Možnosti / Zobrazení** (Options / View) byla přidána nová volba - **GDI kresba bez blikání** (Flicker-Free GDI Draw). Je-li zvolena tato možnost, DesignCAD použije speciální optimalizaci pro urychlení vykreslování širokých řádků v GDI zobrazení drátového modelu. Tato volba nemá žádný účinek, pokud je aktivní **Režim RedSDK** (RedSDK Mode). Pro přístup k tomuto nastavení stiskněte klávesu "Q", otevře se panel **Možnosti** (Option), aktivujte kartu **Zobrazení** (View). Zaškrtnávací políčko je v pravém dolním rohu skupiny **Možnosti zobrazení** (Display). Rozdíl je zvláště patrný pod systémem Windows 7, pokud máte zapnutou transparentnost.



Obecné opravy a vylepšení:

- Při exportu souborů do formátu DXF nebo DWG byl někdy text pod úhlem převeden na vodorovný text.
- Příkaz **Skenovat obrázek** (Scan Image) byl někdy šedivý - nepřístupný, protože DesignCAD nemohl najít potřebný program. Tento problém byl opraven.
- Při aplikaci materiálu (jako je Oak, Cherry, Pearl, atd.) na entitu **Povrch tělesa** (Solid Surface) nebyl při uložení výkresu uložen s objektem také tento materiál.
- Volba **Uložit jako 2D projekci** (Save as 2D Projection) nepracovala s entitou **Povrch tělesa** (Solid Surface).
- Při práci s výkresem, který obsahoval **multičáry** (multi-lines), docházelo k havárii DesignCADu.

BasicCAD Additions for Custom Properties

STATEMENT: ADDPROPERTY entid, key\$, name\$, value\$, nocopy

Arguments:

- **entid** – a number. If entid is greater than zero it represents the index of the entity to which the new property is to be added. If entid is 0, the property will be added to the drawing instead.
- **key\$** -- a string representing the key to be used in identifying this custom property.
- **name\$** – a string representing the name to be used in identifying this custom property.
- **value\$** – a string representing the contents of the property.
- **nocopy** – a flag that indicates whether this custom property should be propagated from the original entity via Duplicate, Copy/Paste, Array, or other editing operations. Also, if set, does not allow editing or deletion of this property via the Custom Property editor.

The ADDPROPERTY statement adds a new custom property to the specified entity if entid > 0, or to the drawing if entid = 0.

Errors:

- If the key/name pair clashes with an already-defined custom property, the statement triggers Error 114.
- If entid is non-zero and does not specify a legal entity id, the statement triggers Error 60.

Example 1:

```
' add a custom property to the first selected entity
PRECISION 0
ON ERROR GOTO oops
entid = GETSELECT 1, j
key$ = "PART"
name$ = "NUMBER"
value$ = "ZX_500_A12"
nocopy = 0 'false - this custom property can be edited by the drawing's user.
ADDPROPERTY entid, key$, name$, value$, nocopy
END
oops:
e = Error(q) ' get error number
MESSAGE "Error", e, "; ", Err$(e)
RETURN
```

Example 2:

```
' add a custom property to the drawing
entid = 0 ' specifies the drawing instead of an entity
key$ = "DRAWING"
name$ = "START DATE"
value$ = "06/25/2016"
nocopy = 0 'false - this custom property can be edited by the drawing's user.
ADDPROPERTY entid, key$, name$, value$, nocopy
```

STATEMENT: GETPROPERTY entid, propid, key\$, name\$, value\$, nocopy

Arguments:

- **entid** – a number. If entid is greater than zero it represents the index of the entity from which the property is to be retrieved. If entid is 0, the property will be retrieved from the drawing instead.
- **propid** – a number specifying the index of the property to be retrieved. Indexing begins at 1, not 0.
- **key\$** -- a string that will contain the key of this custom property.
- **name\$** – a string representing the name of this custom property.
- **value\$** – a string representing the contents of this custom property.
- **nocopy** – a flag that indicates whether this custom property can be propagated from the original entity via Duplicate, Copy/Paste, Array, or other editing operations. Also, if set, does not allow editing or deletion of this property via the Custom Property editor.

The GETPROPERTY statement retrieves a custom property from the specified entity if entid > 0, or from the drawing if entid = 0.

Errors:

- If entid is non-zero and does not specify a legal entity id, the statement triggers Error 60.
- If propid is not the index of an existing property, the statement triggers Error 25.

Example 1:

```
' get the third custom property from the first selected entity
e = 0
PRECISION 0
ON ERROR GOTO oops
entid = GETSELECT 1, j
propid = 3
key$ = "aaaa"
name$ = "bbbb"
value$ = "cccc"
nocopy = -1
GETPROPERTY entid, propid, key$, name$, value$, nocopy
IF e = 0 THEN
    MESSAGE "Property", propid, " KEY:", key$, "; NAME:", name$, "; VALUE: ",
values$, "; nc: ", nocopy
END IF
END

oops:
e = Error(q) ' get error number
MESSAGE "Error", e, "; ", Err$(e)
RETURN
```

STATEMENT: SETPROPERTY entid, propid, key\$, name\$, value\$, nocopy

Arguments:

- **entid** – a number. If entid is greater than zero it represents the index of the entity for which the property is to be changed. If entid is 0, the indexed drawing property will be changed instead.
- **propid** – a number specifying the index of the property to be changed. Indexing begins at 1, not 0.
- **key\$** -- a string that will contain the new key of this custom property.
- **name\$** – a string representing the new name of this custom property.
- **value\$** – a string representing the new contents of this custom property.
- **nocopy** – a flag that indicates whether this custom property can be propagated from the original entity via Duplicate, Copy/Paste, Array, or other editing operations. Also, if set, does not allow editing or deletion of this property via the Custom Property editor.

The SETPROPERTY statement modifies a custom property from the specified entity if entid > 0, or from the drawing if entid = 0.

Errors:

- If entid is non-zero and does not specify a legal entity id, the statement triggers Error 60.
- If propid is not the index of an existing property, the statement triggers Error 25.
- If key\$ and name\$ would clash with an already-existing property, the statement triggers Error 114.

Example 1:

```
' Set the third custom property in the drawing
e = 0
PRECISION 0
ON ERROR GOTO oops
entid = 0 'we're targeting a drawing property instead of an entity property
propid = 3
key$ = "NewKey"
name$ = "NewName"
value$ = "I have changed."
nocopy = 1
SETPROPERTY entid, propid, key$, name$, value$, nocopy
IF e = 0 THEN
    GETPROPERTY entid, propid, nkey$, nname$, nvalue$, nnc
    MESSAGE "Updated Property", propid, " KEY:", nkey$, "; NAME:", nname$, _ "; VALUE:
", nvalue$, "; nc: ", nnc
END IF
END

oops:
e = Error(q) ' get error number
MESSAGE "Error", e, "; ", Err$(e)
RETURN
```

STATEMENT: GETPROPERTYINDEX entid, name\$, startindex, foundindex

Arguments:

- **entid** – a number. If entid is greater than zero it represents the index of the entity to be searched. If entid is 0, the drawing properties will be searched instead.
- **name\$** – a string representing the custom property name to be searched for.
- **startindex** – a number representing which property index to start at.
- **foundindex** – a number representing the index of the next property that matches the specified name. Set to -1 if no matching property is found.

The GETPROPERTYINDEX statement searches the custom properties of the drawing (if entid is 0), or of the specified entity (if entid > 0) for a name field matching name\$. Set startindex to 1 to begin searching at the first custom property. If searching for multiple matches (which is possible, since there may be two or more properties with different key fields but the same name field), search for the second and subsequent matches by adding 1 to the last foundindex.

Errors:

- If entid is non-zero and does not specify a legal entity id, the statement triggers Error 60.
- If startindex is not the index of an existing property, the statement triggers Error 25.

Example 1:

```
' Search the drawing for all custom properties having the name "PARTNO"
entid = 0
name$ = "PARTNO"
startindex = 1
foundindex = 0
PRECISION 0
ON ERROR GOTO oops
DO WHILE foundindex >= 0
    GETPROPERTYINDEX entid, name$, startindex, foundindex
    IF e <> 0 THEN GOTO finish
    IF foundindex <> -1 Then
        MESSAGE "Match at index", foundindex
        Startindex = foundindex + 1
    END IF
LOOP
finish:
END

oops:
e = ERR(q)
RESUME NEXT
```

STATEMENT: DELETEPROPERTY entid, propid

Arguments:

- **entid** – a number. If entid is greater than zero it represents the index of the entity from which a custom property is to be deleted. If entid is 0, a custom property will be deleted from the drawing properties instead.
- **propid** – the index of the custom property to be deleted.

The DELETEPROPERTY statement deletes the custom property specified by propid from the entity specified by entid, or from the drawing if entid is zero.

Errors:

- If entid is non-zero and does not specify a legal entity id, the statement triggers Error 60.
- If propid is not the index of an existing property, the statement triggers Error 25.

Example 1:

```
' Delete the fourth custom property from entity 2
entid = 2
propid = 4
DELETEPROPERTY entid, propid
```

New Sys() functions relating to Custom Properties

Sys(689) – the number of custom properties that the current drawing contains.

Sys(690) – the number of custom properties contained in the entity last loaded by the ENTITY statement.

New BasicCAD Commands and Parameters For Import/Export

SKETCHUP SUPPORT

>SkpOut

```
{
<Type [t] '3 = Sketchup 3.0, 4 = 4.0, 5= 5.0, 6 = 6.0, 7 = SketchUp 7.0, ...
'8 = SketchUp 8, 9 = 2013, 10 = 2014, 11 = 2015.
' Numbers less than 3 default to SketchUp 3, and values greater
' than 11 match SketchUP 2015. (This upper end will 'slide' as newer
' version of SketchUp are supported)
<TransposeYZ [tx] '0 = no, 1 = yes
<Filename [f$]
<NULL 'optional -- if used only updates Setup options for SketchUp export
}
```

OLE Automation Classes, Methods, and Properties for Custom Properties

CustomProperties Class

Methods:

AddProperty(Key\$, Name\$, Value\$, bNoCopy) as Integer

Returns the index of the newly-added property if successful, or -1 in case of failure.

DeleteProperty(iIndex) as Boolean

Returns True if successful or False if the property could not be deleted for any reason.

Item(iIndex) as CustomProperty

Returns the CustomProperty specified by the index.

Properties:

Count as Long

Returns the number of custom properties in the collection

CustomProperty Class

Methods:

SetKey(Key\$) as Boolean

Sets the Custom Property's Key to the value specified in Key\$. Returns False if the key could not be updated to the specified value (typically due to a key/name clash that would result).

SetName(Name\$) as Boolean

Sets the Custom Property's Name to the value specified in Name\$. Returns False if the Name could not be updated to the specified value (typically due to a key/name clash that would result).

Properties:

Key as String (read-only)

Returns the Key for the Custom Property. Read-only. To change the contents of the Key property, use the SetKey method.

Name as String (read-only)

Returns the Name for the Custom Property. Read-only. To change the contents of the Name property, use the SetName method.

Data as String (read/write)

Returns the Data contents for the Custom Property. Can be directly read and written.

NoCopy as Boolean

Returns the NoCopy field of the Custom Property. Can be set directly read and written.