

# Fogyókúra: átállítás KDE-ről Mate-ra

PCLinuxOS Magazine – 2015. június

Írta: Peter Kelly (critter)

Az előző hónapban a KDE-ről Mate asztali környezetre váltásról írtam. Ennek során próbáltam az xdotool-lal elérhető leghasznosabb KDE-ben elérhető tulajdonságokat Mate-ban elérhetővé tenni. A KDE egyik tulajdonsága, amit a legjobban szerettem, az a képesség, hogy egy adott alkalmazást előre meghatározott helyen és méretben, akár meghatározott munkaterületen is, megnyithatott.

A Mate asztali környezetben az ablakkezelő csak egy elem. Lehet eltérő, sokkal erőteljesebb ablakkezelőt is használni az alából beállított Macro-n túl, de „felkapott” kezelők hajlamosak kevésbé gazdaságosan bánni a forrásokkal, így ütközhetnek az eredeti céloommal, amiért Mate-ra váltok. Végül is a Macro jó, könnyűsúlyú ablakkezelő, a Gnome metacity-től eredeztethető tiszteletreméltó örökséggel, amire a Mate végül is épült. A kis Macro csupán némi segítségre szorul.

Az adott tulajdonságot sikerült elérnem az xdotool használatával, de az eredmény nem volt éppen kielégítő, mivel a kívánt paraméterek szerinti mozgatás és átméretezés előtt rövid időre megjeleníti az ablakot. Gondoltam, csak kell legyen ennél jobb eljárás és ennek okán ismét az Internethez fordultam.

Valami olyasmit kerestem, ami a Macro ablakkezelővel együttműködik (a Wikipedia szerint a macro spanyolul keretet jelent, ami ablakkezelő esetén értelmezhető is\*\*). Bármilyen eszköz is legyen, annak a PCLinuxOS tárolójában elérhetőnek kell lennie.

\*\*Azon Mate használók számára, akik rácsodálkoznak a Mate néhány alkalmazásának fura elnevezésére:

Amikor a gnome-asztal 2-esről 3-asra váltott, sok felhasználó, közöttük Linus Torvalds is, nem volt meglepődve az új rendszerrel és elkezdtek alternatívákat keresni. A kiábrándult felhasználók közül egy Perberos nevű argentin felhasználó elhatározta, hogy másképpen közelíti meg a kérdést és a gnome 2-t egy új, mate-nak (népszerű dél-amerikai ital) elnevezett asztali környezetté (DE) alakítja át (leszármaztatja).

A mate-eszközöket, amiket a gnome-tól vett át, át kellett nevezni és erre a célra anyanyelvéből vett nevek használata mellett döntött. A Wikipedia lefordítja nekünk ezeket:

The mate utilities that he forked from gnome had to be re-named and for this he chose to use names from his native Spanish. Wikipedia translates them thus:

Caja (doboz) – fájlkezelő (Nautilus)

Pluma (tollszár) – szövegszerkesztő (Gedit)

Eye of MATE – képmegjelenítő (Eye of GNOME)

Atril (lectern) – olvasópolc (Evince)

Engrampa (kapocs) – archívumkezelő (File Roller)

MATE Terminal – terminál-emulátor (GNOME Terminal)

Marco (keret) – ablakkezelő (Metacity)

Mozo (pincér) – menüszerkesztő (Alacarte)

## wmctrl

Elsőként a **wmctrl**-t néztem meg, ami hasznos eszköz lehet, de makacsul megtagadta néhány alkalmazás felismerését, különösen a LibreOffice és Firefox-ét. Talán abból a tényből fakad, hogy a PCLinuxOS tárolójában található változat az 1.07-es, ami az alkalmazás honlapjának tanúsága szerint 2005-ből származik. Megemlíti az 1.08-as verziót is, de az szintén 2005-ben jelent meg, 10 év ablakkezelő-fejlesztés maradt feldolgozatlanul. Az, hogy képtelen az összes ablakkezelőt alkalmazni, alkalmatlanná teszi az adott, megoldandó feladatra.

## devilsPie



A következő kis ékkő, amin elmerengtem a devilsPie szokatlan nevet viseli. Ez egy másik eszköz, ami nem tűnik karbantartottnak, 2007-es amit elmondható róla, vagyis elég öreg. Van egy frissebb változata a devilsPie2, de ez nincs a tárolókban, így

a listából kikerül. A fejlesztő, Ross Burton weblapjáról való az idézet:

*„Ablakmanipuláló eszköz, amit a Sawfish's "Matched Windows" opciója és a Metacity-ből hiányzó ilyen lehetőség inspirált”*

Kora ellenére ez sokkal biztatóbbnak tűnik.

A Devilspie úgy működik, hogy az ablakkezelő által kezelt ablakokat figyeli, és ha azok bármelyike megfelel egy előre meghatározott feltételnek, a leírt műveletet hajtja rajta végre az adott ablakon. A szabályok „.ds” kiterjesztésű szöveges fájlokban vannak a felhasználó home-jának „.devilspie” könyvtárában, amit létre kell hozni (a név előtti pontról ne feledkezz el). A szabályok egy adott, a lisp programnyelv számára kialakított „s-expression” formátumot követnek, de ettől nem kell megijedni, a formátum elég egyértelmű.

A dokumentáció elég szegényes, de az elérhető műveletek, parancsok stb. elég jó leírása olvasható itt <http://www.foosel.org/linux/devilspie>. Az adott weblapon hivatkozásokon keresztül további információk érhetők el.

Ahhoz, hogy a pluma szövegszerkesztő mindig a második virtuális asztalon a bal felső sarokban jelenjen meg 700 px széles és 500 px magas formában, létre kell hoznom egy fájlt a pluma ablakán alkalmazandó szabályokkal a következő szöveggel:

```
(if
  (is (application_name) "Pluma")
  (begin
    (set_workspace 2)
    (geometry "700x500+0+0")
  )
)
```

A fájlnev lényegtelen, de .ds végződésű kell legyen és a home könyvtár **.devilspie** nevű könyvtárába kell menteni.

vagyis

```
/home/én/.devilspie/pluma.ds
```

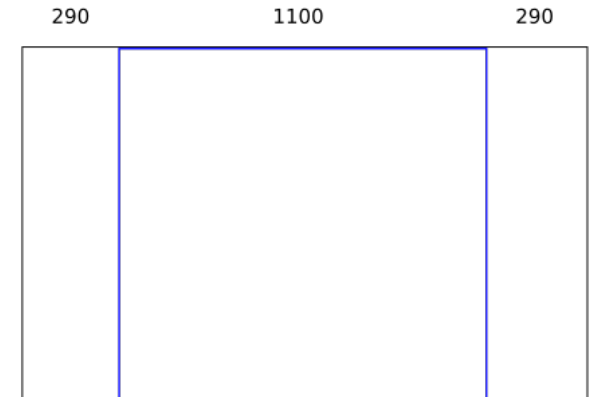
Én így neveztem el a fájlt. A fájl először „Pluma” nevű ablakot keres és ha megtalálta, akkor az indító kifejezést követő két szabályt alkalmazza. A munkaterületek számozása egytől indul, ellentétben az xdotool-lal, ami nullától kezd számolni. Ez az eljárás jobban működik, mint a korábbi kísérletem, de továbbra is röviden felvillan a képernyőn, mielőtt a szabályokat alkalmazná, noha egyelőre elégedett vagyok, tovább keresek.

A devilspie képességeibe kicsit belemerülve az eszköz alkalmazásának több módja tárul elénk. A LibreOffice-t elég gyakran használom és amikor a szövegszerkesztőt elindítom szeretem, ha a szöveg az oldalpaneleket nem bántva teljes szélességben és a képernyő teljes magasságában jelenik meg. Másrészt, a táblázatkezelőt külön virtuális asztalon mindig maximalizálva jelenítem meg. Itt az alkalmazás neve mindkét esetben „LibreOffice 4.4”. (A 4.4-es a rendszeremen az aktuális verzió.) Ugyanakkor az ablak osztálya alkalmazásonként más és más, „libreoffice-writer”, vagy „libreoffice-calc”. Tudván ezt, képes vagyok eltérő utasítás-csomagot írni az egyes alkalmazásoknak (lent látható, hogyan nyerhető ki ez az információ).

writer.ds

```
(if
  (is (window_class) "libreoffice-writer")
  (begin
    (geometry "1100x800+290")
    (maximize_vertically)
  )
)
```

Először az ablakot vízszintesen az előírt és függőlegesen egy tetszőleges méretre állítja. Ezután függőlegesen maximalizálja. A geometriai kifejezésekhez



magyarázat: az értékeket az indokolja, mert két monitoron dolgozom, 1680x1050 és 1920x1080 felbontással, ez 3600x1080 hatásos felbontás. Mivel a szövegszerkesztő 1100 px szélesen az első monitor közepén kell, az  $x = (1680 - 1100) / 2 = 290$  kell legyen.

Egy monitornál, a számítás elhagyható, a +290-es eltolás nélkül egyszerűen középre rendezésre utasítanám, ami kiszámítaná a megfelelő pozíciót. A középre rendező utasítás alkalmazásának hatása több monitor mellett jó esetben is kiszámíthatatlan.

A táblázat, mindkét irányban maximalizált lévén, sokkal egyszerűbb.

calc.ds

```
(if
  (is (window_class) "libreoffice-calc")
  (begin
    (set_workspace 2)
    (maximize)
  )
)
```

A szabályok leírásához szükséges információk kinyeréséhez legegyszerűbb megnyitni az adott alkalmazást, majd terminálban futtatni a devilspie debug-ot. Lehet még létrehozni egy debug.ds fájlt a .devilspie könyvtárban, ami csak a debug szót tartalmazza. Ha a továbbiakban nincs szükség

információra, csak nevezd át debug-ra a fájlt. Utóbbi helyett a parancs kiadása

## devilsjie debug

*Nyomj Ctrl+C-t a kilépéshez*

A következő információkat nyomtatja ki az egyes ablakokról:

Ablak neve, Alkalmazás neve, Osztály és Méretek. Az első három azonos a megfelelő window\_name, application\_name and window\_class sztringgel.

A megfelelő ablak keresésében nagy szabadság van. Az is teszt használata helyett

```
(is string_a string_b)
```

ami a fenti példában alkalmazva az egyezést ellenőrzi, használható a contains

```
( contains string_a string_b )
```

ami a string\_a-n belül keresi a string\_b-t (string\_a alárendet sztringje).

Alternatívaként itt van a **matches**

```
( matches string_a minta )
```

ahol a minta egy szabályos kifejezés. Ez nagyon megnöveli a keresési rugalmasságot és lehetővé teszi ablakcsoportok megfeleltetését. Emellett az ellenőrzések kombinálhatóak az úgynevezett boolean műveleti jelek (and, or és not) alkalmazásával. Használatuk elég egyszerű, ha azonnal nem is látszik annak. Az általános formátum így néz ki.

```
(begin # innen indul
(if # ha és csak ha
(is (némi teszt) érték) # ez igaz
(and # és
(is (más teszt) értékek) #ez is
```

```
(not (is (másik teszt) érték))# de ez
hamis
(begin # akkor és csak akkor
(action 1) # ezt csinálja
(action 2) # és ezt csinálja
... # és ezeket
) # második begin rész vége
) # az and lezárása
) # az if lezárása
) # az els begin lezárása
```

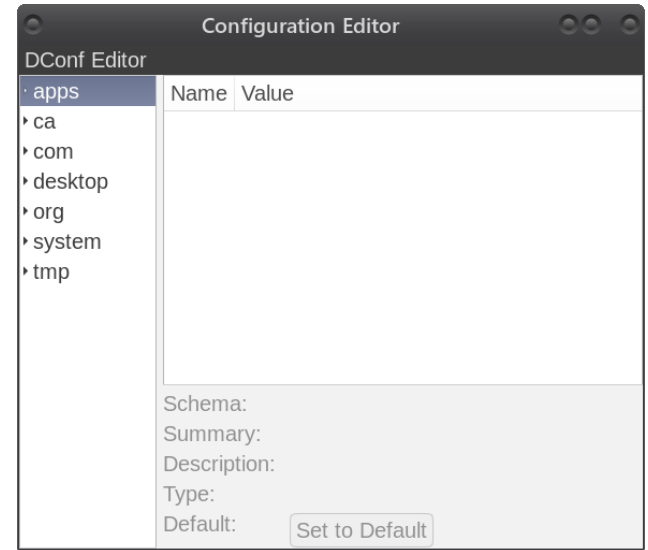
A zárójelek rendes lezárására ügyelj és használj behúzást az olvashatóság érdekében. A korábban megadott hivatkozás további információkkal szolgál és leírja a következő, a devilsjie által végrehajtható műveleteket:

**debug, print, println, str, hex, geometry, fullscreen, focus, center, maximize, maximize\_vertically, maximize\_horizontally, unmaximize, minimize, unminimize, shade, unshade, close, pin, unpin, stick, unstick, set\_workspace, set\_viewport, skip\_pager, skip\_tasklist, above, below, undecorate, wintype, opacity, spawn\_sync and spawn\_async.**

Ez egy elég ekintélyes lista, ami lehetővé kell tegye számomra a KDE-től való eltávolodás okozta ür kitöltését.

## A dconf editor

Szintén az elmúlt hónap cikkben említettem a dconf editor-t is. Ez nem kimondottan Mate eszköz és más asztalok, mint az XFCE is használják. Egy grafikus felület, amivel néhány alacsonyabb szintű rendszerbeállítás érhető el. Ellentétben más Linux beállításokkal ezeket nem egyszerű, szövegszerkesztővel módosítható szöveges fájlok tárolják, hanem adatbázis-kulcsok – értékpárok és a dconf editor lehetővé teszi az értékek könnyebb változtatását. Az erre szolgáló parancssori eszköz a gsettings. A parancssori eszközt nem könnyű

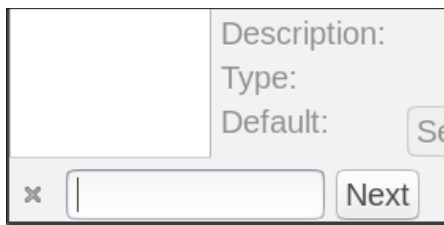


használni és ezért örülünk a grafikus szerkesztőnek, bár az egy kicsit szokatlan.

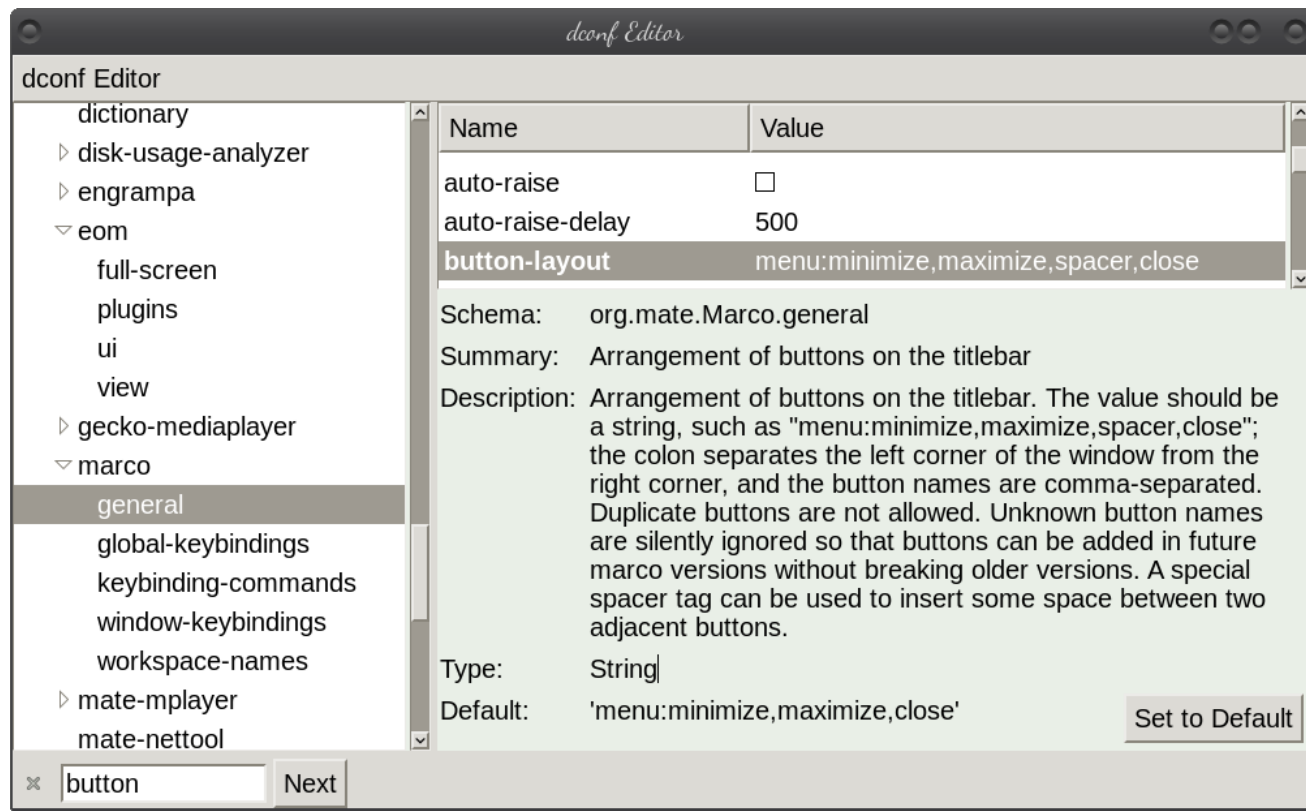
Mivel ezek az értékek az asztal megjelenítése és működése szempontjából kritikusak, bölcs dolog mentést készíteni az aktuális beállításokról, mielőtt változtatni kezdenél, amiket később megbánsz. A mentés fájlja a ~/.config/dconf/user-ben lesz.



Menüje mint olyan, ahol szokásosan megtalálható a mentés és a megnyitás nincs, de ha rákattintasz a bal felső sarokban a Dconf Editor-ra, háromelemű lenyíló listát kapsz benne Find, About és Quit (Keresés, Névjegy és Bezár) opciókkal.



A Find opciót kiválasztva megnyílik egy kis szövegbeviteli ablak balra lent, itt rákereshetsz dolgokra, amiket vagy megtalál neked, vagy sem. Alternatívaként mozoghatsz úgy, hogy a bal oldali fa



struktúra szakaszait igény szerint, a kis háromszögekre kattintva, kinyitod.

A kereső ablakba beírtam, hogy button és szerencsésen, elég gyorsan megtaláltam, amit kerestem. Ezek azok a gombok, amiket az ablak címsora tartalmaz, ahol pl. maximalizálhatod, vagy bezárhatod az ablakot. A dconf editor ad némi hasznos segítséget a kulcsok lehetséges értékeiről, azok hatásairól és az alapbeállítás szerinti értékről. Ennek a gombelrendezési kulcsnak számos előfordulása van, de ami a számomra Mate alatt működött, a macro → general → button-layout rész alatt található.

Szeretem, ha a bezáró gomb egy kicsit elkülönül a minimalizáló és maximalizáló gomboktól, hogy elkerüljem a véletlen bezárást. Ennek érdekében

## Fogyókúra: átállítás KDE-ről Mate-ra

beillesztettem a „spacer” értéket vesszőkkel elválasztott felsorolásba. A hatása fenti képen látható. Lehet, hogy ki kell jelentkezned, vagy akár újraindítani, hogy a változtatásaid hatását lásd.

Ha az asztali környezeted használja a dconf editor-t, érdemes egy pillantást vetni rá. Sok olyan beállítás található itt, amiket nehéz, vagy majdnem lehetetlen más módon változtatni és olyanok is vannak, amikről biztos nem is tudtál. Miközben nézegettem az org → mate → macro részt, felfedeztem számos engedélyezett billentyűparancsot, amikről nem volt tudomásom és számos esetlegesen hasznosat, amiket csak össze kell rendelni.

Még az hiányzott, ami a KDE-ben van, a kézikönyvek megnyitása Konqueror-ban kioslave segítségével. Ez csak annyit tesz, hogy a szokott borzalmas terminál stílusúnál sokkal civilizáltabb módon jelenítette meg a kézikönyv lapjait.

A Mate help fájl böngészőjét yelp-nek hívják, a dokumentációja szerint képes a man és info fájlokat megjeleníteni. Sajnos úgy tűnik, hogy ez lehetőség a PCLinuxOS Mate változatában nem érhető el.

Ennek a megoldására egy kis bash-függvényt hívok meg, amit régen akkor találtam, amikor a ~/.bashrc fájlomhoz készítettem álnevet (alias-t). Elfogad olyan szekció számokat, mint a **man 8 lsblk** -ban

Íme a függvény, ahogy azt a ~/.bashrc fájlomba beépítettem

```
# a reddit-en találtam nagyon régen
# sledgespread publikálta - köszönet sledgespread-nek
pdfman()
{
    TMPFILE=$(mktemp atril-$USER.XXXXXXX
--tmpdir)
    man -t "$@" | ps2pdf - "$TMPFILE"
    atril "$TMPFILE"
    bash -c "sleep 2 rm $TMPFILE" &
}
```

És természetesen az alias.

```
alias man=pdfman
```

A függvény készít egy időleges fájlt és a ps2pdf eszköz való a kimenet pdf formátumúra konvertálására, ami azután a képernyőn jelenik meg. A konverzió meglepően gyors, késleltetés alig érzékelhető. Pdf megjelenítésére az Atril-t használom, de ezt helyettesítheted a saját kedvenc pdf-olvasóddal.

Amennyiben a kézikönyvoldal nem létezik, akkor egy üres pdf-et kapsz és a terminálban a „No manual entry for...” üzenet jelenik meg. Noha ez bug-nak minősíthető, de úgy gondolom nem szükséges javítani, hiszen az üres fájl jelzi a lényegét. A függvényből kilépéssel az időszakos fájl megsemmisül.

Még ha ez nem is a leghatékonyabb eljárás, jól működik és a erőforrásigénye alacsony, illetve időleges. Tovább dolgozhatnék ezen az info fájlok megjelenítése érdekében – nos, talán valamikor, de én valóban nem sokat használom az info fájlokat, az információk az Interneten sokkal világosabbak.

A ps2pdf eszköz a ghostscript-common csomag része, és ha még nem lenne telepítve, a PCLinuxOS tárlóiban elérhető.

Vagyis nekem Mate-m van, amin a beállítások zöme olyan, hogy úgy tudom használni, mint a KDE-t. Ha valamilyen tulajdonság hiányozna, van róla sejtsem, hogyan állíthatom be, és olyan tulajdonságokat is elérek, amiről a KDE-ben nem tudtam. A Mate-beállításom elegáns és gyors, miközben a hardvert kevésbé terheli. Külön jó, hogy ha másik asztali

## Fogyókúra: átállás KDE-ről Mate-ra

környezetre váltanék, már tudom, hogyan hozhatom ki belőle a legtöbbet.

A Linux mindig is kiemelten jól konfigurálható volt, lehetővé téve a felhasználónak, hogy sokkal pontosabban szabja az igényeire és elvárásaihoz. Ez a néhány bemutatott eszköz még ennél több tulajdonságot tesz elérhetővé.

