

# XFCE4: a General Monitor (Genmon) használata

PCLinuxOS Magazine - 2016. szeptember

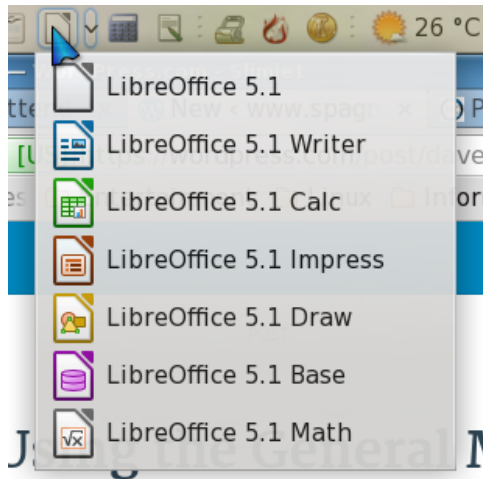
Írta: Dave Spagnol [DaveCS]

A [blogjából](#) engedélyével publikálva

**[Megjegyzés: a következő példákban az összes egyes, vagy dupla idézőjel, függetlenül attól, hogy jobbos-e, vagy balos-e, mind egyenes.]**

Az XFCE-ről köztudott, hogy könnyű, gyors, funkcionális, de mégis funkciókban gazdag. Noha a beállítója (Desktop Stting Manager), amivel számos kisalkalmazást lehet meghívni, nem olyan komplex, vagy nagy mint a KDE Setting Manager, de tudja, amire a legtöbbeknek szüksége lehet. Sok panel alkalmazást lehet indítani. A Launcher-e (indító) a legrugalmasabb, amit eddig láttam, lehet egyetlen ikon, egy alkalmazás indítására, vagy ikon kis nyíllal lenyíló menü számára, és ez úgy érhető el, hogy egynél több programot adunk a Launcher-hez. Az első indító ikonja lesz a fő ikon, de ha szeretnéd a menüben is megjeleníteni, akkor csak add hozzá még egyszer.

Miként itt a képernyőképen is láthatod, a LibreOffice összes elemét egyetlen indítóra raktam ki.



Még van más is. Ellentétben a többi programindítóval, nem vagy kizárólag a menüelemekhez kötve. Amikor új elemet adsz hozzá, az lehet a szokásos indító menüből, de készíthetsz magad is teljesen újat, ha tudod, hol található a futtatható fájl, az ikon és szövegszerkesztőt használva nevet és leírást készíthetsz. Akár sajátot is készíthetsz a szabványos bejegyzést kiegészítve.

## Mi a helyzet a GenMon-nal?

Az Xfce4 teljesítménye mellett az is lenyűgöző, hogy milyen szépen fut. De egy panelalkalmazás mégis hiányzik, a CPU-hőfok figyelő. Tehát, most a GenMon segítségével készíték egyet.

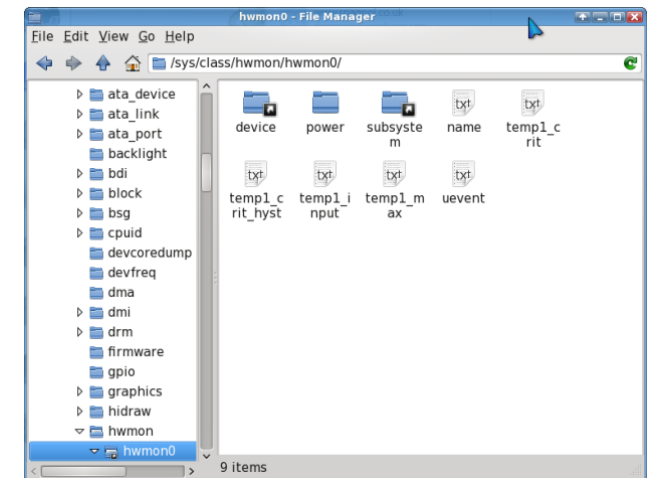
Amit most írni készülök, nem biztos, hogy nálad is működni fog, de az eljárás a lényeg. Telepíteni kell a **psensor**-t, a PCLinuxOS-ben Synaptic-kal elérhető. Emellett rendszergazdaként be kell lépned a **/sys/class/hwmon/**-ba, ami egy „virtuális könyvtár”, ahonnan a számítógép hardverével kapcsolatos dolgokat lehet kiolvasni.

A GenMon egy paneli alkalmazás, ami általad írt szkriptet futtat. A szkript 4, megjelenítendő (nyomtatandó) sor kell legyen. A sorok ilyen formátumúak:

```
<txt>Text</txt>
<tool>Tooltip</tool>
<img>image.png</img>
<click>program</click>
```

Bármelyikük lehet változó, tehát számos tényező mérhető és csak döntés kérdése, melyik tartalma jelenjen meg a képernyőn. Az „image” egy ikon, a „click” egy program, ami csak akkor fut, ha van kattintható ikon hozzá, a „text” bármilyen szöveg és „Tooltip” (elemleírás) pedig szintén bármilyen szöveg lehet.

A komputeremben Athlon FX4100-as processzor van. A hardvermonitora csak két adatot jelez, mások rengeteget adnak! Az első a processzor maghőmérséklete. A második a processzor által a hűtőbordán disszipált teljesítmény. Nyisd meg a fájlkezelővel a **/sys/class/hwmon/hwmon0**-t és ezt láthatod:



A temp1\_input virtuális szövegfájl tartalmazza a CPU maghőmérsékletét, ezred °C-ban megadva. Ez a Mousepad-del (az Xfce alap szövegszerkesztője) nem nyitható meg, de tartalma bármikor kiolvasható. Nyiss terminált és gépeled ezt be:

```
cat /sys/class/hwmon/hwmon0/temp1_input
```

Egy számot kapsz, valószínűleg 10 000 körül. Előfordulhat, hogy máshol kell keresned a fájlt. Most annál maradok, hogy ez a megfelelő. Ahhoz, hogy az adatot egy szkriptfájl változójába kinyerjük, az első sornak valahogy így kell kinéznie:

```
temp1=`cat /sys/class/hwmon/hwmon0/temp1_input`
```

Figyeld meg, hogy a visszafelé álló idézőjelek hatására kapod meg a parancs szöveges kimenetét olyan módon, hogy egy változóba, most a temp1-be rakhasd. A /sys/class/hwmon/hwmon1/ könyvtár tartalmaz egy hasonló fájlt, a „power1\_input” nevűt. Ha terminálba beírod:

```
cat /sys/class/hwmon/hwmon1/power1_input
```

az eredmény 10 000 000-s lesz, jelezvén a hűtőbordán keresztül, hő formájában eldisszipált CPU-teljesítményt. Sok CPU nem ezt az adatot adja meg, hanem a hűtőventilátor fordulatszámát. A hardvermonitor könyvtárában valahol található valami használható. Ugyanakkor számomra ez az érdekes, ezért a szkriptfájlt kiegészíttem egy újabb sorral.

```
power1=`cat /sys/class/hwmon/hwmon1/power1_input`
```

Most a kérdés az, hogyan használjuk fel ezt a két számot, hogy a GenMon alkalmazásod hasznos információt jelenítsen meg a panelen. Ha a fenti „cat” parancs nem ad vissza, számokra hasznos adatot, akkor további dolgokat kell keresned a hardverrel kapcsolatban. Ott van minden, valahol!

Én írtam egy többsoros valamit, ami feldolgozza a számokat, szétválasztja a részeket és tizedespontot (ford. nálunk tizedesvessző van) tesz köztük. Azt akartam elérni, hogy 4 számjegyet kapjak egy tizedesponttal úgy, hogy a szám hossza ne változzon, mindig 5 karakter legyen, például 5.030, 54.37, vagy 120.2. Az egyik eljárás, amivel ez elérhető, hogy a temp1-et elosztjuk 1000-rel:

```
#First we divide by 1000, but this loses
any decimal points.
temp2=$((temp1/1000))
#Now we get the remainder.
temp3=$((temp1%1000))
#Now we have to add leading zeroes to the
remainder
temp4=$((1000+temp3))
temp3=${temp4:1:3}
#And finally...
temp4=${temp2}."$temp3
```

**#and to 4 significant figures:**  
**temp1=\${temp4:0:5}}°C”**

Működik, és jól, de én valami sokkal rendezettebbet akartam, ezért körülnéztem a net-en, a mozgó tizedesponos számításra jobb eljárást keresvén. Az awk-ra esett a választásom a bc helyett, mivel a bc z 1-nél kisebb számok esetén a kezdő 0-t nem tesz ki.

Az awk egy gyors és trehány programozási nyelv (minden nap tanulhatunk valami újat). Általában fájlból dolgozik, de képes változókat kezelni. Még nem ismerkedtem meg teljesen vel, de abból, amit olvastam, kidolgoztam a mozgó tizedesponos osztást. Az [Awk Primer -- print and printf\(\) oldal](#) sokat segített.

Minden esetre nekem ezt a sort kellett használnom (egy perc és elmagyarázom, mi mit jelent)

```
temp3=$(awk -v m=$temp1 'BEGIN { printf("%.3f", m/1000
)}')
temp1=${temp3:0:5}
```

Az **awk** parancs ez esetben egy karaktersort ad vissza, a parancsot zárójelbe rakva, amit egy dollárjel követ, lehetővé téve, hogy az eredményt egy karaktersorba ágyazzuk. A „-v m=\$temp1” fontos, mivel definiál egy, a parancs által használt változót. Enélkül egy fájlra lenne szüksége. A „'BEGIN {...} ' ” rész az awk szintaxisának része, nem értem teljesen. A „printf()” egy formázott karakter-sor és a korábban általam hivatkozott lap sokat segít a megértésében. Az első, idézőjeles karakter-sor a zárójelen belül, meghatározza a formátumot, esetünkben a „%- .3f” balra igazítást jelent, változó tizedesjellel, legfeljebb 3 tizedesig. A következő sor az eredményt 4 jegyre vágja (tizedesjellel együtt 5).

Oké, ennek ismeretében meg tudok írni egy szép, rövid szkriptet, a saját gépemre szabva. Megtörtén, de egy nap hibajelzést adott! Miért? Nos, a Linux hardverfigyelési rendszere a hwmon0 és a hwmon1 elhelyezkedését megváltoztatta! Ezért hozzáírtam egy ezt ellenőrző részt előre és íme a végleges szkript.

(Szerk. megjegyzése: mindegyik szövegcsoport egy-egy önálló sor, csak a tisztánlátásért.)

```
#!/bin/sh
```

```
if [ -e /sys/class/hwmon/hwmon1/temp1_input ] ;
```

```
then
```

```
temp1=`cat /sys/class/hwmon/hwmon1/temp1_input`
```

```
power1=`cat /sys/class/hwmon/hwmon0/power1_input`
```

```
else
```

```
temp1=`cat /sys/class/hwmon/hwmon0/temp1_input`
```

```
power1=`cat /sys/class/hwmon/hwmon1/power1_input`
```

```
fi
```

Négy további sort adtam hozzá később, l. lent

```
temp3=$(awk -v m=$temp1 'BEGIN { printf("%.3f", m/1000
)}')
```

```
temp1=${temp3:0:5}}°C”
```

```
power2=$(awk -v m=$power1 'BEGIN {printf("%.3f",
m/1000000 )}')
```

```
power1=${power2:0:5}}°W”
```

```
echo “<img>/usr/share/icons/temperature-s.png</img>
```

```
echo “<txt>$temp1</txt>”
```

```
echo “<tool>Heat Transfer: $power1</tool>”
```

```
echo “<click>psensor</click>”
```

Ez további problémát vetett fel. Az Athlon FX4100-as

CPU-ról visszkapott hőmérsékletadat hihetetlenül alacsony. Azért van mindez, mert úgy alakították ki a visszaadott számot, hogy a CPU terhelését jelezze és addig nem lesz pontos, amíg a processzor nem kezd forróvá válni. Ha egy forró nyári napon azt jelzik, hogy a processzor 8 °C-on fut, egyértelműen hibás. Kicsit gondolkodtam és úgy döntöttem, hogy az adat 60 °C-tól lesz pontos és ezért hozzá kell adnom a jelzett hőmérséklet és a 60 közötti különbség egy harmadát. A legtöbb processzor esetén ez nem okozhat gondot, de természetesen, amikor a saját szkriptedet írod, tetszés szerint dönthetsz!

Tehát hozzáadtam ezt a néhány sort a szkriptben a jelzett részhez.

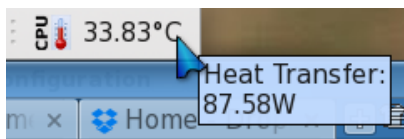
```
tempdiff1=$((60000-$temp1))
```

```
discrim=$((($tempdiff1>0))
```

```
tempdiff2=$((($tempdiff1*$discrim/3))
```

```
temp2=$((($temp1+$tempdiff2))
```

A sorok mérik a különbséget, ha a visszaadott adat 60 alatt van és ekkor elosztják 3-mal, majd azt hozzáadják az érzékelő által adott értékhez. Vedd észre, hogy lett egy „temp2”-d, így át kell írnod a „temp3=” kezdetű sorban az „m=\$temp1”-et „m=\$temp2”-re.

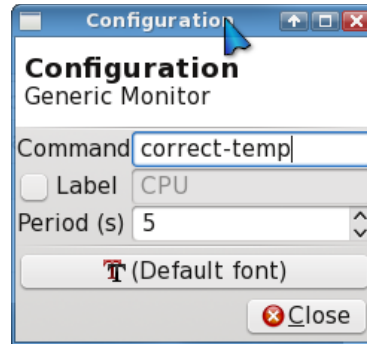


Ha Fahrenheitben szeretsz számolni, akkor még valamit tenned kell, a „temp3=” kezdetű sorban az „m/1000”-et írd át „32+9\*m/5000”-re. A következő sorban pedig a „°C”-t „°F”-re.

Mentsd a fájlt (én „correct-temp”-nek neveztem el az enyémet) a /usr/local/bin könyvtárba és tedd

végrehajthatóvá. Ám mielőtt a GenMon-hoz kapcsolnád, ellenőrizd, hogy működik-e! Ha terminálban futtatod négysoros kimenete kell látnod, ahogy azt a cikk elején mutattam. Emellett kell még egy **temperature-s.png** ikon a /usr/bin/icons könyvtárba. Itt van egy:

<https://dl.dropboxusercontent.com/u/81695069/temperature-s.png>



A szkript GenMon-hoz adásához előbb telepítened kell az „xfce4-genmon-plugin”-t Synaptic-kal, majd hozzá kell adnod a panelhez. Jobb kattintás rajta és válaszd ki a **Tulajdonságokat**, ahol a „Command” címkéjű négyzetbe írd be a szkript nevét, valamint állítsd be az ismétlés időzítését másodpercben.

Remélem, kitalálsz, hogy a genmon-t hogyan használd a az XFCE4 lehetőségeinek fejlesztésére.

Looking for an old article?  
Can't find what you want? Try the

**PCLinuxOS Magazine's  
searchable index!**

The **PCLinuxOS** magazine

Support PCLinuxOS! Get Your Official

**PCLinuxOS**  
Merchandise Today!

**PCLinuxOS**

