Fájlba zárt szellem, MS Word dokumentumok kezelése Linux alatt

Mihez kezdjünk újonnan kapott, vagy régen elkészített MS Word dokumentumainkkal Linux alatt? Hogyan olvashatjuk el őket egyszerűen, a karakteres vagy akár a grafikus felületen?

Az alapvető probléma a *doc* formátumú szöveges állományokkal az, hogy a *doc* zárt fájlformátum, vagyis csak a készítők tudják, hogy pontosan hogyan is épül fel a fájl, milyen szimbólumok mit jelentenek, és ez alapján hogyan kell formázni a benne lévő szöveget.
Arról felesleges beszélni, hogy ez kinek jó, és kinek nem jó, mindenesetre az általam írt szöveges fájl tartalma a saját tulajdonom, melyhez bármikor, bárhol hozzá kellene férnem, lehetőségemnek kellene lenni tovább szerkeszteni, még akkor is ha, éppen nem rendelkezem érvényes *Microsoft Office* licencel.
Habár a *Microsoft* honlapján található egy ingyenes *doc* fájl megjelenítő, az sajnos nem támogatja a többi operációs rendszert, így *Linux* alatt sem fog futni. Esetleg valamilyen emulátor használatával még életre lehet kelteni, de a dokumentumot akkor is csak megnézni tudjuk vele, tovább szerkeszteni, módosítani nem.
Vegyük számba tehát, hogy milyen ingyenes és szabad programok állnak rendelkezésünkre, hogy elolvassuk újonnan kapott, vagy régi *doc* fájljainkat, illetve átkonvertáljuk őket valamilyen nyílt formátumra. De melyik a legjobb eszköz erre?
Az *OpenOffice*, vagy az *Abiword*, esetleg a *KOffice*? Vagy van még egyéb lehetőségünk is? Mit csináljunk akkor, ha csak egy karakteres terminál áll rendelkezésünkre? Egyszerű, olvassuk tovább a cikket!

Word dokumentumok olvasása a karakteres felületen

Amennyiben a dokumentum nem tartalmaz képet, vagy a kép számunkra nem fontos, akkor arra is lehetőségünk van, hogy a fájlt a karakteres felületen olvassuk el. Erre is több lehetőségünk van, nézzük meg először a *catdoc* programocskát, ez tulajdonképpen hasonló elven működik, mint a sokak által ismert *cat*, de a bemenete egy *doc* formátumú fájl, a kimenete pedig minden formázás nélküli szöveg. Sokszor a célnak ez is megfelel. Használata közel sem bonyolult, alapvetően két választási lehetőségünk van, aszerint, hogy a kimenet *ascii* vagy *tex* formátumú legyen.
Lássunk, egy példát az *ASCII* formátumú kimenetre:

```
catdoc -fascii valami.doc
```

ekkor a szöveg a terminálra íródik ki, természetesen át tudjuk irányítani a kimenetet fájlba is, például

```
catdoc -fascii valami.doc
➥> olvashato.txt.
```

A `catdoc` programocskával, és némi *Bash* szkripttel pedig akár az összes *doc* formátumú fájlunkból készíthetünk egy egyszerűen olvasható szöveges verziót. A program kimenete lehet *Latex* is, ekkor a *-t* opciót használjuk a `-fascii` helyett.
Lehetőségünk van a kimenet karakterkódolásának megváltoztatására is, ami alapértelmezett esetben *utf-8*. Az elérhető karakterkódolások listáját a

```
catdoc -l
```

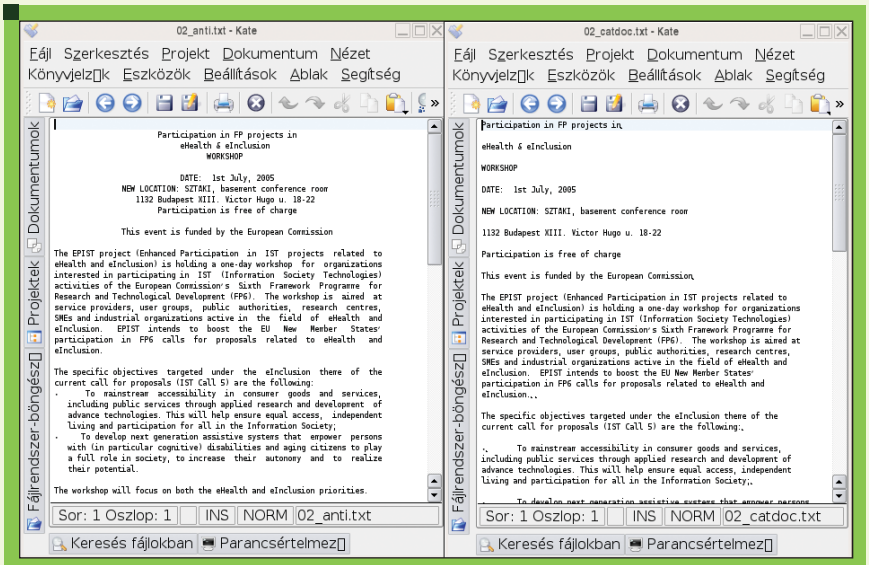
paranccsal kérdezhetjük le, és a `-d` opcióval használhatjuk, például:

```
catdoc -fascii -d8859-2
valami.doc.
```

A `catdoc` tényleg csak arra használható, hogy a fájlból kiolvassuk a szöveget. Amennyiben ennél valamivel nagyobb igényeink vannak, akkor próbáljuk ki az *antiword* programot. Az *antiword* már jóval kifinomultabb program, és még a karakteres felületen is láthatóan szebb eredményt produkál, mint az előző (*1. ábra*). Az eredeti szöveg sorkizárt, és az *antiword* a sorkizárást jól jeleníti meg a karakteres kimeneten, ellentétben az *catdoc* programmal. Különösen tetszett, hogy a sorkizárt szövegeket ténylegesen sorkizártként jelenítette meg a karakteres felületen is. Az *antiword* már használható *PostScript* fájlok készítésére is, ami teljes mértékben megfelel arra, hogy az eredeti *doc* tartalmát bárhol megjelenítsük, akár egy *PostScript* nyomtatónak utána átadjuk, de sajnos tovább szerkeszteni nem lehet. Az

```
antiword valami.doc
```

parancs szintén az alapértelmezett kimenetre rakja ki a `valami.doc` tartalmát *UTF-8* karakterkódolással, lehetőségünk van a megadni, hogy az elkészült szövegfájlban hány karakter lehet maximum egy sorban, a `-w X` opcióval, ahol `X` egy pozitív szám.



■ *1. ábra* Baloldalon látható az antiword és jobb oldalon a catdoc kimenete

Word dokumentumok konverziója

Az `antiword` programmal *PostScript* fájlt, a következő paranccsal készíthetünk:

```
antiword -p a4 -m 8859-2.txt
➥valami.doc > valami.ps
```

ahol `-p a4` jelenti, hogy *A4*-es oldalakra szeretnénk formázni a szöveget, `-m 8859-2.txt` pedig a *PostScript* fájl karakterkódolása, az *UTF-8* használata itt nem támogatott.
Mivel az `antiword` más formátumban történő exportálást nem támogat, ezért érdemes még tovább keresni hasonló programok iránt. Elég ígéretesnek tűnik a *wvWare* program, mely talán az eddigiek közül a legtöbb szolgáltatást nyújtja. Lehetőségünk van ugyanis, *HTML*, *PDF*, *Latex*, *ABW (Abiword)*, vagy akár *RTF* formátumra is átkonvertálni az eredeti *MS Word* fájlt.
A *antiworddel* ellentétben a *wvWare* a *PostScript* fájlokat nem egyből a *doc*-ból készíti el, hanem először *Latex* fájlt készít belőle, és abból készít *ps*, illetve *pdf* fájlokat. Így ha *ps* fájlt szeretnénk kimenetként kapni, akkor feltétlenül szükségünk lesz a *tetex* csomagra.
Bizonyos átalakításoknál a *wvWare* a fejlesztők szerint szebb kimenetet ad, amennyiben a *links*, *elinks*, és *lynx* programok fel vannak telepítve.
A *wvWare* képes *HTML* formátumba is konvertálni, apró csellel persze ezt is megtekinthetjük a karakteres terminálon, mégpedig úgy, hogy a *wvWare* kimenetét beleirányítjuk a *w3m* bemenetébe:

```
wvWare -x /usr/share/wv/
➥wvHtml.xml valami.doc | w3m
➥-T text/html
```

Előfordulhat, hogy nekünk máshol helyezkedik el az adott fájl. Ez a megoldás azért jobb, mint ha simán csak a `text` kimenetet választottuk volna, mert így a fájlban található linkekre azonnal rákattinthatunk. Az előbb a `-x` kapcsolót akár el is hagyhattuk volna, mert azt csak akkor kell megadni, ha nem az alapértelmezett *HTML* formátumba történik a konverzió.
A *HTML* formátum nagy előnye, hogy a képek nem a fájlban vannak, hanem csak a hivatkozás található meg rájuk, és mindegyik külön fájlban tárolódik. Ezért ha egy *doc* fájlból ki akarjuk nyerni az összes képet, akkor csak konvertáljuk át a *wvWare* segítségével *HTML* formátumúra, mellesleg a program támogatja azt is, hogy az egy *doc* fájlban lévő képek neve elé valamilyen megkülönböztető karaktersorozatot tegyünk. Ezáltal rögtön tudni fogjuk, hogy melyik kép melyik *HTML* fájlhoz tartozik.
Talán az egyik leghasznosabb konverzió, amit a *wvWare* tud, az a *Latexre* történő átalakítás, és rögtön kétféle *Latex* kódot is képes generálni, attól függően, hogy mennyi formázást kívánunk megtartani az eredeti fájlból.

Amennyiben az `-x` kapcsolót */usr/share/wv/wvCleanLaTeX.xml* paraméterrel hívjuk meg, akkor a kimeneti fájl csak minimális formázást tartalmaz, de ha a */usr/share/wv/wvLaTeX.xml* paraméterrel, akkor a kimeneti *Latex* fájl megpróbál minél jobban hasonlítani a *Word* által megformázott *doc*-ra. Ekkor azonban olyan formázások kerülnek bele a fájlba, melyeket utána kézzel elég nehéz lesz szerkeszteni.
Az egyszerűség kedvéért *wvWare* programhoz tartozó segédprogramokkal (*wv**) nem kell megkeresnünk az adott *xml* fájlt, hanem csak ki kell adnunk a

```
wvLatex valami.doc valami.tex
```

vagy

```
wvCleanLatex valami.doc
valami.tex
```

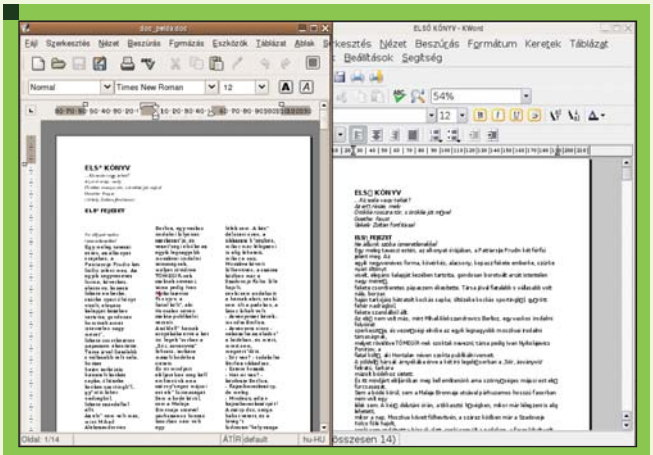
parancsot, és máris rendelkezésünkre áll a kész *Latex* kód.

Grafikus felületű programok

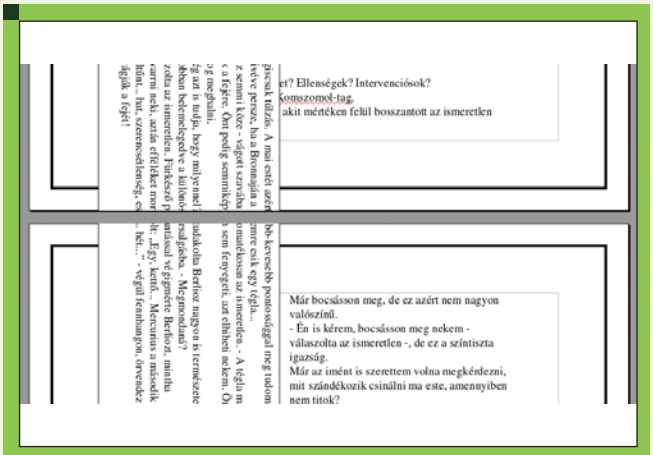
Miután megismerkedtünk a karakteres felületen keresztüli *MS Word* dokumentumok olvasásával, konverziójával, térjünk át egy sokkal izgalmasabb, és valószínűleg többet is használt területre. Ezek pedig a grafikus felületen futó programok. Bizonyára mindenki ismer ezek közül legalább egyet, de valószínűsítem, hogy sokan akár az összessel találkoztak már.
Most nézzük meg ezeket a szövegszerkesztőket abból a szempontból, hogy mennyire értik meg az *MS Worddel* készített fájlokat. A *KOffice* hátránya, hogy csak *Linux* alatti verzió van belőle, míg az *OpenOffice* és az *Abiword* rendelkezik *Windows* vagy *Macintosh* alatt is használható verziókkal.

Az Abiword

Talán erről a programról lehet hallani a legkevesebbet, az összes közül.
A többivel ellentétben az *Abiwordnek* csak szövegszerkesztő funkciói vannak, míg a *KOffice*, és az *OpenOffice* tartalmaz táblázatkezelő, és prezentáció készítő programokat is.
Az *Abiword* előnyére válik ugyanakkor, hogy nagyon kis méretű alkalmazás, a legutóbbi verzió telepíthető csomagja körülbelül 3,5 Mb. Ennek megfelelően kevesebb erőforrásra van



■ *2. ábra* Az Abiword egyszerűsége ellenére jobban megjeleníti az MS Word fájlokat a KWord-nél



■ *3. ábra* Az OpenOffice 1.x sem képes tökéletesen megjeleníteni mindent

szüksége, mint a másik két programnak, és a doc formátumú szöveges fájlokat is érezhetően gyorsabban nyitja meg. A konverzió még nagy dokumentumok esetén is pillanatok alatt megtörténik. Ebből természetesen nem csak azért van, mert a program kicsi, hanem mert az ***Abiword*** fájlszűrője nem tökéletes, és bizonyos formázásokat figyelmen kívül hagy.
Az oldalszegélyek egyáltalán nem jelennek meg, ahogyan néha a képek sem a dokumentumban, a ***Wordart*** szövegek pedig egyáltalán nem támogatottak. A program jól kezeli ellenben a karakter formázásokat, a táblázatokat, a hasábokat, és a címjegyzékkel is megbirkózik, habár az utóbbi esetében az egyes al-al fejezetek számozásánál oda nem illő karakterek jelentek meg.
Az ***Abiword*** nem támogatja a hierarchikus felsorolást, ezért a konverziónál bár a behúzások megmaradtak, de a kis piktogrammok már nem tesznek különbséget az egyes szintek között.
Az ***MS Word*** dokumentumok beolvasása során az ***Abiword*** jól megjelenítette a tabulátorokat, de a tabulátorok közötti kitöltést nem. A lábjegyzet, élőfej, élőláb szintén támogatott a programban, de az élőfejben, élőlábban lévő dátum, és idő mezők többször jelentek meg egymás után egy oldalon, így a felesleges mezőket kézzel kell kitörölnünk.
Az ***Abiwordben*** továbbá egy oldalon nem lehetnek különböző oszlopszámú hasábok sem.
Ne felejtsük el ugyanakkor, hogy egy programot nem feltétlenül az határoz meg, hogy milyen hűen kezeli az ***MS Word*** dokumentumokat. Egyszerűbb szöveges állományok megnyitására azonban jól használható, platform független szövegszerkesztő.

A KOffice

A ***KOffice*** programcsomag szövegszerkesztője a ***KWord*** program, ahogyan már ezt megelőzően említettem a ***KOfficenak*** nem létezik ***Windowsos***, vagy ***Mac OS X*** verziója, ellenben a ***Fink*** projekt (➲ ***http://fink.sourceforge.net***) keretében elérhető a ***Mac OS X*** verzió.
A ***KOffice*** jól együttműködik a ***KDE*** ablakkezelő rendszerrel, sőt szüksége is van a ***kdebase*** csomagra, és a program felülete a szabványos ***KDE*** programokhoz illeszkedik.
Sok különböző fájlon végzett konverziók után inkább csalódott voltam a programmal kapcsolatban, mintsem megelégedett. Szignifikánsabb jobb eredményt nem sikerült vele előállítani, mint amit az ***Abiworddel*** is sikerült. Ami nekem nagyon hiányzott, az a hasábok támogatása volt. Tulajdonképpen a táblázatok, szövegformázások, felsorolások kezelése jól meg volt oldva a programban. Az ***MS Word*** dokumentumba beágyazott grafikákat sem jelenítette meg, ami kissé érthetetlen, hiszen a ***Kivio*** is a ***KOffice*** része, ami egy ***MS Visio*** klón, ezért jogosan vártam volna el, hogy ilyen téren maga mögé utasítsa az ***Abiword*** programot.
Az ***Abiworddel*** ellentétben a ***Kword*** jól kezeli a tabulátorokat, és az előtte való kitöltést is képes megjeleníteni.
Az élőfej, és élőláb rendesen működik a szövegszerkesztőben, illetve a dátum, oldalszám, és idő mezők is jól jelennek meg.
Durva hibának találtam azt, hogy a szövegdobozban lévő szöveget a ***KWord*** képtelen volt kiolvasni a fájlból, ezért az gyakorlatilag elveszett. Míg az ***Abiword*** legalább kiolvasta és a dokumentum végére fűzte a szövegdoboz tartalmát, habár az a szöveg közepén helyezkedett el eredetileg.

Az OpenOffice.org 1.x

Minden bizonnyal az összes eddig megvizsgált program közül a legismertebb irodai alkalmazás, a legtöbb funkcionalitással, és természetesen az ezzel együtt járó kicsit lassabb működéssel. A szoftver fejlesztésekor kifejezett hangsúlyt fektettek arra, hogy amennyire csak lehet kompatibilis legyen a ***Microsoft*** hasonló célokra készült termékével, amely érezhető is a használat során.
Tulajdonképpen ez az egyetlen alkalmazás, amely szinte mindennel megbirkózott. A dokumentumba beszúrt összes képet megjelenítette, ügyelve arra, hogy a kép tulajdonságait is megtartsa, attól függően, hogy a kép a szöveg előtt, vagy mögött helyezkedett-e el. Az egyetlen probléma talán az volt, hogy a beágyazott grafikákat tovább nem lehetett szerkeszteni, mert az egy dobozba tartozó összes objektum egy képként jelent meg az ***OpenOffice.org***-ban.
Az ***OpenOffice.org*** még olyan kevésbé fontos részleteket is precízen jelenített meg, mint az oldal szegélyek, a szövegdobozban a szöveg irányultsága, a ***WordArt***-tal készített feliratok, és tulajdonképpen még sorolhatnám azon formázási elemeket, melyeket sikerrel átvett a ***doc*** formátumból.

1. táblázat *Konklúzió*

Funkció	Abiword	KWord	OO.o 1.x	OO.o 2.x
Karakter formázások	✓✓	✓✓	✓✓	✓✓
Hasábok	✓	✗✗	✓✓	✓✓
Címjegyzék	✓✗	✓✓	✓✓	✓✓
Beszúrt képek	✓✗	✗✗	✓	✓✓
Tabulátorok	✓✗	✓✓	✓✓	✓✓
Felsorolás, számozás	✓	✓✓	✓✓	✓✓
Táblázatok kezelése	✓	✓	✓✓	✓✓
Szövegdoboz	✗	✗✗	✓	✓✓
Élőfej, élőláb (oldalszám, dátum mező)	✓	✓✓	✓✓	✓✓
Rajzok kezelése	✗✗	✗✗	✓	✓
WordArt	✗✗	✗✗	✓	✓✓
Oldalszegélyek	✗✗	✗✗	✓	✓
Lábjegyzet	✓✓	✓✓	✓✓	✓✓
Összhatás	✓	✗	✓✓	✓✓

Magyarázat: ✓✓ nagyon jól importál; ✓✗ bizonyos esetekben jól importál; ✗✗ egyáltalán nem támogatott; ✓ jól importál; ✗ inkább rosszul importál

Kisebb problémák itt is voltak a megjelenítéssel, például a szövegdoboz (*3. ábra*) két oldal közé került, és kézzel kellett áthelyezni egy olyan helyre, hogy az oldal határain belülre kerüljön.

Az OpenOffice.org 2.0 beta

Mivel az ***OpenOffice.org 2.0*** nagyon sok újítást tartalmaz az előző verzióhoz képest, de a cikk írásának pillanatában nincsen belőle stabil verzió, ezért úgy gondoltam, hogy jobb, ha mindkét változatot külön-külön veszem górcső alá.
Az előző verzió összes jó tulajdonsága tulajdonképpen igaz a 2.0-ra, azért csak azokat a területeket említem meg, ahol a változás volt tapasztalható.
Első ránézésre nagyon hasonló volt az összes megjelenített dokumentum, és csak egyetlen hiányosságot találtam benne. Viszont egy kisebb beállítás után az ***MS Word*** és ***OpenOffice 2.0*** alatt pontosan ugyanúgy nézett ki a dokumentum.
Ez a hiányosság mégpedig az oldalszegélyek kapcsán került bele a dokumentumban, ugyanis az ***OpenOffice*** és az ***MS Word*** alapértelmezésként másképpen állítja be az oldalszegély helyzetét. Míg az ***OpenOffice*** alapbeállításként a szegélyt szorosan a betűk mellé helyezi, addig az ***MS Word*** a szegély és a betűk között hagy némi távolságot. Ettől a kis különbségtől a megnyitás után egy kicsit másképpen néz ki a dokumentum, de átállítás után teljesen azonos kép fogad minket.
A fejlesztők a szoftver esetében még arra is figyeltek, hogy a címjegyzékben lévő hivatkozások is működjenek. Tulajdonképpen itt sorolhatnám azokat az apró finomításokat, amit az ***OpenOffice 2.0***-át tökéletesítik az elődjéhez képest, de minden apró részletre hely hiányában nem térnék ki.

Konklúzió

A táblázatban megpróbáltam összefoglalni, körülbelül 20 ***Word*** dokumentum konvertálása közben történt tapasztalataimat. A ***Word*** dokumentumok között vegyesen voltak különböző verziójú ***Word***del készített fájlok. Ez a szám tulajdonképpen kevés arra, hogy azt merjem állítani, hogy bizonyos funkciók minden esetben működnek, úgyhogy a valós életben történt felhasználás során eltérések természetesen lehetségesek.

Mivel tulajdonképpen a ***doc*** formátum egy fekete doboz, és a fejlesztők mindent megtettek annak érdekében, hogy a belső struktúráját visszafejteni a lehető legnehezebb legyen, ezért elképzelhető az, hogy egyes esetekben mégsem tudjuk jól megjeleníteni az adott fájlt. Mellesleg ez sokszor még a Microsoft termékek egyes verziói közötti váltás során is problémákat okozott, hogyan várhatnánk el akkor, hogy a tulajdonképpen a sötétben tapogatózó fejlesztők ennél jobbat alkossanak?!
Mindenkinek azt ajánlom, aki ***doc*** fájlokkal foglalkozik, hogy mindenképpen szerezze be az ***OpenOffice 2.0*** béta verzióját, mert sokkal megbízhatóbban kezeli az ***MS Word*** formátumát mint az előző verzió. Sajnálatos módon a két verzió nem fér meg egymás mellett, így nem lehet párhuzamos használni mindkettőt. Ezért amennyiben a 2.0 nem tűnik elég megbízhatónak számunkra ahhoz, hogy mindenhol lecseréljük vele az ***OpenOffice 1.x***-et, legalább néhány helyen érdemes ezt megtenni, főleg ott, ahol sok ***doc*** formátumú bejövő anyagra kell számítanunk.

Az optimista jövő

Amennyiben hihetünk az Interneten terjengő híreszteléseknek, és a piacot uraló termék fejlesztőinek, akkor a következő irodai programcsomag már olyan fájlformátumot fog használni, amelyik teljesen nyílt lesz. Ez nem csak azért nagyszerű, mert ezentúl ezzel a termékkel készült fájlokat gond nélkül lehet majd kezelni ***OpenOffice*** vagy akár ***Abiword*** alatt, hanem azért mert végleg megoldódnak a konverziós problémáink, és az összes eddigi ***doc*** formátumban tárolt adatunkat könnyedén konvertálhatjuk át ebbe az új, nyílt formátumba.

Horváth Ernő
ernohorvath@gmail.com
24 éves, műszaki informatikus. Három évvel ezelőtt ismerkedett meg komolyabban a Linux rendszerekkel és emellett érdeklődik még a robotika és a biztonságtechnika iránt is. Ha lenne szabadideje sokat kirándulna, biciklizne és filmeket nézne.