

# Poglavje 1

## UVOD



## 1.1 Študija primera

Obvladovanje logističnih tokov zajema načrtovanje, izvedbo in kontroliranje premikanja ter razmeščanja blaga in s tem povezanih podpornih aktivnosti v okviru sistema, ki sledi specifičnim ciljem. V današnjem času je razvoj logističnih sistemov napredoval do te mere, da je za obvladovanje celotnega sistema oskrbne verige potrebno upoštevati še ostale vidike (management, informatika, komunikacije, optimizacijske metode...).

V preteklosti so se obvladovanja logističnih tokov lotevali brez sodobnih pristopov - temeljil je zgolj na človeškem umu. Sodobna tehnologija omogoča razvoj informacijske tehnologije in telematike, s podporo katere lahko integriramo celostno oskrbno verigo.

Problema obvladovanja logističnih tokov se lahko študentje in podjetja lotijo na več različnih načinov, z različnimi pristopi in znanji. Uporaba prosto dostopnih programskih orodij in njihova integracija v poslovne procese omogoča sistematično reševanje izbranega logističnega problema, predvsem z vidika optimizacije, vizualizacije, simulacije in projektiranja. Prednost tovrstnih orodij je v njihovi dostopnosti, enostavnosti, funkcionalnosti in uporabi na različnih operacijskih sistemih. Z njihovo pomočjo lahko bistveno izboljšamo obvladovanje notranjih in zunanjih logističnih tokov.

## 1.2 Opis problema

Izvajanje logističnih procesov v oskrbni verigi avtomobilske proizvodnje zahteva znanja iz področja managementa, upravljanja, optimizacije, komunikacije in informacijskih znanj. Integracija vseh deležnikov in področij oskrbne verige rezultira v učinkoviti in optimalni izvedbi vseh procesov. Temelj učinkovitega sodelovanja med njimi je pravilna zasnova procesov, nalog in odgovornosti (npr. informacijski, materialni in finančni tokovi).

Funkcije in naloge so med deležniki oskrbne verige jasno določene in razdeljene. Vsak deležnik mora upoštevati navodila in standarde, in sicer tako, da ne vpliva na ostale procese (zmanjšanje učinkovitosti in zanesljivosti) oskrbne verige. Pri tem mora upoštevati predhodni in naslednji korak. Del učinkovitega izvajanja logističnih aktivnosti predstavlja uporaba informacijske podpore in tehnologije. Izmenjava informacij v realnem času omogoča učinkovito načrtovanje, izvedbo, kontrolo in ukrepanje v primeru pojava nepredvidenih dogodkov.

V celotni oskrbni verigi, kjer sodelujejo ponudniki, dobavitelji, prevozniki, proizvajalci, kupci in potrošniki, se izbrani logistični problem nanaša le na integracijo dobaviteljev, prevoznikov in proizvajalcev. Vključuje obvladovanje in izvajanje logističnih procesov nabave, transporta, prevzema, skladiščenja in odpreme, ki so del celotne oskrbne verige.

Izbran logistični problem v avtomobilski proizvodnji vključuje različna področja in procese:

- planiranje proizvodnih kapacitet glede na trend povpraševanja prodaje novih vozil;
- proces nabave komponent (vijaki, pnevmatike in platišča);
- izbira transporta in transportnih poti (pnevmatike transportiramo z vlakom, paltišča s tovornjakom in vijake s kombijem);
- proces skladiščenja (komponente skladiščimo na treh različnih in ločenih lokacijah v centralnem skladišču);
- proces prevzema in odpreme;
- upravljanje tveganj;
- finančno načrtovanje;
- proces odločanja (večparametrski odločitveni model);
- pregled tveganj lokacijske strategije s pomočjo prostorskega planiranja;
- simulacija materialnih tokov v skladišču;
- statistična analiza podatkov;
- vizualizacija oskrbne verige;
- informacijska tehnologija za označevanje komponent.

### 1.3 Proces aktivnosti

Različni dejavniki, med katere lahko prištevamo gospodarsko stanje države, finančno stanje družbe in ostale družbene vidike, vplivajo na povpraševanje po novih vozilih. Povečanje povpraševanja po novih vozilih posledično vpliva na obseg proizvodnje vozil in povečanju ponudbe na avtomobilskem trgu.

Povečanje obsega proizvodnje vozil sorazmerno spiralno zajame vse ostale logistične procese v makro in mikro okolju. K makro okolju lahko prištevamo globalne vidike (povečanje gostote prometa in števila transportov, negativen vpliv na okolje, gospodarska rast itd.), k mikro okolju pa prištevamo interne procese podjetij (planiranje procesov nabave, proizvodnje in skladiščenja, razširitev skladiščnih in proizvodnih kapacitet, simulacije itd.).

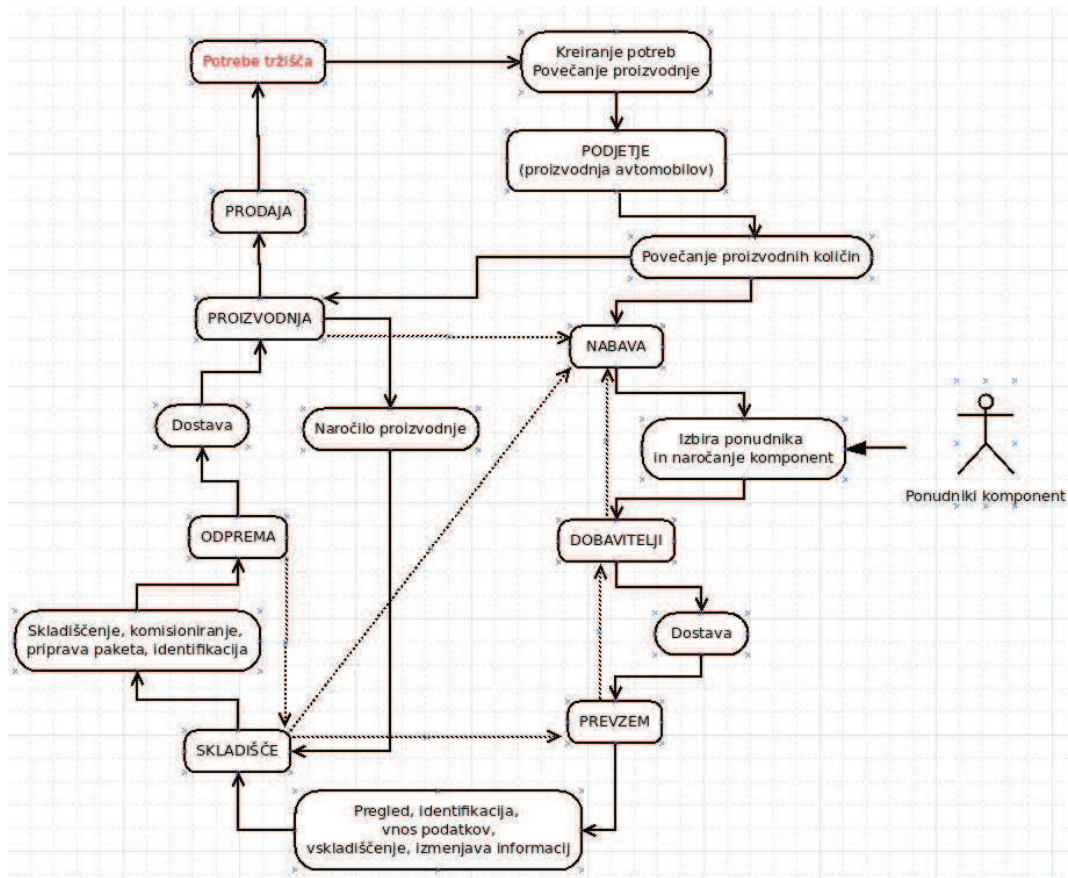
Spremembe na avtomobilskem trgu imajo dokazano zelo velik vpliv na celotno dogajanje v gospodarstvu, saj ta sektor zajema ogromno število podjetij (avtomobilska industrija predstavlja veliko oskrbno verigo s številnimi deležniki). O vplivu avtomobilske industrije na gospodarstvo nazorno priča tudi zadnja svetovna gospodarska kriza - hitro ukrepanje in pomoč avtomobilski industriji je eden izmed izhodov iz krize. Kako vpliva sprememba ponudbe na trg in povpraševanje po

vozilih na poslovanje avtomobilskega podjetja? Povečanje povpraševanja spremeni ravnovesje na trgu in povzroči manjko pri ponudbi, s čimer se poveča tržna cena vozil, kar lahko negativno vpliva na nadaljnjo povpraševanje po vozilih. Podjetja morajo pravočasno ukrepati in povečati ponudbo - povečati proizvodne količine in s tem zagotoviti ravnovesje med ponudbo in povpraševanjem na avtomobilskem trgu.

Zagotovitev ravnovesja na avtomobilskem trgu sproži spremembo pri obsegu proizvodnih količin. Management podjetja je odgovoren, da nastalo situacijo temeljito preuči in poda navodila za nadaljnje delo. Želja po zagotovitvi in zadovoljitvi potreb kupcev po novih vozilih sproži signal po povečanju proizvodnih količin. Signal "potuje" v nabavni oddelek podjetja, kjer je potrebno zagotoviti ustrezno količino komponent. Pri izbiri dobavitelja komponent je potrebno upoštevati načela cene, kakovosti, dostavnega časa in ostalih pogojev. Način dostave naročenih komponent je stvar dogovora med dobaviteljem in kupcem. Transport z različnimi prevoznimi sredstvi omogoča optimizacijo stroškov, zmanjšanje tveganja in odvisnosti od vrste transportnega sistema. Izbira med železniškim ali cestnim transportom je odvisna od naročenih količin, stanja infrastrukturne, prevoznih stroškov itd. Dobavljene komponente se na podlagi ustrezne dokumentacije kvalitativno in kvantitativno pregledajo, prevzamejo, uskladiščijo in vnesejo v informacijski sistem. Informacije o dostavljenih komponentah se sporočijo nabavnemu oddelku, ki o dostavi komponent obvesti proizvodni oddelek. Komponente se nato skladiščijo na ločenih lokacijah. Upravljanje z zalogami je pomembno, saj le te uravnavajo nihanja v proizvodnji. S tem zmanjšamo tveganje morebitnih zastojev proizvodne linije.

Upravljanje skladiščenja je umetnost uravnavanja informacijskih in materialnih tokov. Planiranje skladiščnih kapacitet je eno izmed težjih nalog, saj so lahko nihanja količin precejšnja. V praksi se teorija mnogokrat izkaže v negativni podobi. Polna skladišča vsekakor zagotavljajo nemoteno proizvodnjo in zmanjšujejo tveganje zastoja, vendar pa imajo tudi negativno stran. Zaloge so vezani kapital, ki predstavlja dodaten strošek, zato je potrebno poiskati optimalen nivo zalog, ki upošteva tako finančni kot tudi vidik tveganja.

Pomemben vidik v celotni oskrbni verigi je tudi upravljanje s tveganji, ki lahko bistveno pripomore k učinkoviti realizaciji vseh logističnih procesov.



Slika 1.1: Opis procesa dela

## 1.4 Zbrani podatki

Na Sliki 1.2 so zbrani podatki o namišljenem podjetju OpenStorage, kjer se ukvarjamo s skladiščenjem komponent za proizvodnjo osebnih avtomobilov znamke X.

PODJETJE OpenStorage				
Naslov	Triglavska cesta X, 6264 Bohinjska Bistrica, Občina Bohinj			
DEJAVNOST				
Skladiščenje komponent za proizvodnjo avtomobilov (pnevmatike, platišča in vijaki)				
Količina komponent	4x pnevmatike, 4x platišča, 16x vijak; razmerje 1:1:4			
Vrsta transporta	Vlak, tovornjak in kombi			
Dobavitelji	Italija, Francija			
KOMPONENTA		NABAVNA CENA		
Platišča				35 €/kos
Pnevmatika				60 €/kos
Vijak				3 €/kos
DOSTAVA				
Časovni interval dostave	7200 minut = 120 ur = 5 dni			
Frekvenca dostave	Urejena z naključno porazdelitvijo			
Komponente	Sredstvo	PRIHODI	Interval	Odstopanje +/-
Platišča	Tovornjak		180 min	10 min
Pnevmatika	Vlak		360 min	20 min
Vijak	Kombi		170 min	10 min
ODPREMA				
Komponente	Količina	ODHODI	Interval	Odstopanje +/-
Platišča	20 kosov		8 min	1 min
Pnevmatika	20 kosov		8 min	1 min
Vijak	80 kosov		8 min	1 min
<i>Opomba: skupaj sestavlja 5 avtomobilov</i>				
MANIPULACIJE				
Praznjenje viličarja v proizvodnji			Polnjenje viličarja pri prevzemu	
	Trajanje	Odstopanje +/-	Trajanje	
Platišča	6 min	0 min	6 min	
Pnevmatika	6 min	0 min	6 min	
Vijak	6 min	0 min	6 min	
Skladiščne kapacitete		max	Začetno stanje ob simulaciji	
Platišča	800 kosov		200 kosov	
Pnevmatika	800 kosov		200 kosov	
Vijak	2600 kosov		800 kosov	
KAPACITETA prevoznega sredstva	Optimalno	Investicija	Cena viličarja	
Tovornjak	390 kosov/platišča	Nakup viličarja	20.000 €	
Vlak	800 kosov/pnevmatika			
Kombi	1450 kosov/vijakov			

Slika 1.2: Zbrani podatki

## 1.5 Cilji

Cilj je prikaz uporabe različnih odprtokodnih in prosto dostopnih programskih orodij na primeru reševanja izbranega logističnega problema. Prikazati želimo pomen in smiselnost uporabe programskih orodij z dodanim teoretičnim pogledom na izbrani problem. Bodočim in že izkušenim logistikom želimo pokazati, kako se lotiti reševanja logističnega problema z različnih vidikov (načrtovanje, optimizacija, finance idr.). Želimo jim približati uporabnost tovrstnih programskih orodij, saj menimo, da v praksi niso dovolj prepoznavni. Zavedamo se potreb po povezanosti logističnega znanja z informacijsko tehnologijo, zato spodbujamo k raziskovanju in reševanju praktičnih problemov, z namenom dosega širšega pogleda na določeno problematiko.

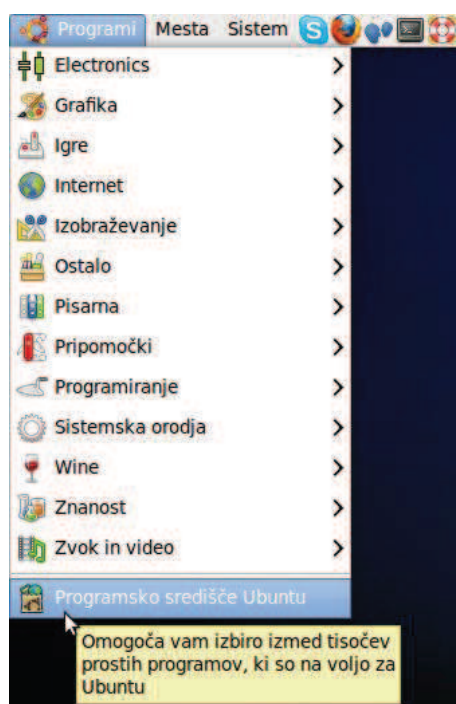
## 1.6 Metodologija

Pri pripravi knjige uporabimo različno metodologijo dela. *Deduktivno metodo* uporabimo na področju, kjer iz splošnih stališč izvedemo posebne in posamične postavke iz katerih nato prehajamo do konkretnih posamičnih zaključkov. Z *metodo analiziranja* razčlenimo posamezna programska orodja na enostavnejše sestavne dele in prvine in se tako lotimo preučevanja vsakega dela. *Metoda sinteze* omogoča, da na podlagi raziskovanja in pojasnjevanja določene procese in postopke povežemo v smiselno celoto, v kateri so njeni deli vzajemno povezani. S pomočjo *statističnih metod* analiziramo posamezne probleme s prikazom grafikonov, tabel, slik. Pri enem izmed programskih orodij se lotimo osnovnih faz *metod modeliranja*, kjer izdelamo program za reševanje konkretnega problema v praksi. *Metoda proučevanja primerov* je metoda, ki jo uporabimo za preučevanje posamičnega primera iz izbranega področja. Z uporabo te metode na podlagi rezultatov opazovanj in zapisanih primerov izvedemo zakonitosti oz. zaključke.

### Kaj pričakujemo od uporabnikov programskih orodij?

Od uporabnikov pričakujemo, da bodo z zanimanjem pristopili k branju knjige in jo morda, kot pripomoček uporabili pri reševanju problemov v praksi. Pričakuje se, da jih dana tematika odprtokodnih in prostodostopnih programskih orodij, navezujoča na logistične probleme, zanima ali pa so zgolj radovedni, kaj jim ponujamo. Želimo, da jim bo raziskovanje skozi posamezna področja prikazano na lahkotnejši in enostavnejši način ter da se pričnejo zavedati, da le ni vse plačljivo, ampak da lahko marsikateri problem razrešimo s pomočjo brezplačnih programskih orodij. Priročnik je namenjen študentom, kakor tudi ostalim uporabnikom, ki jih dana problematika zanima.





Slika 1.3: Zagon programskega središča Ubuntu

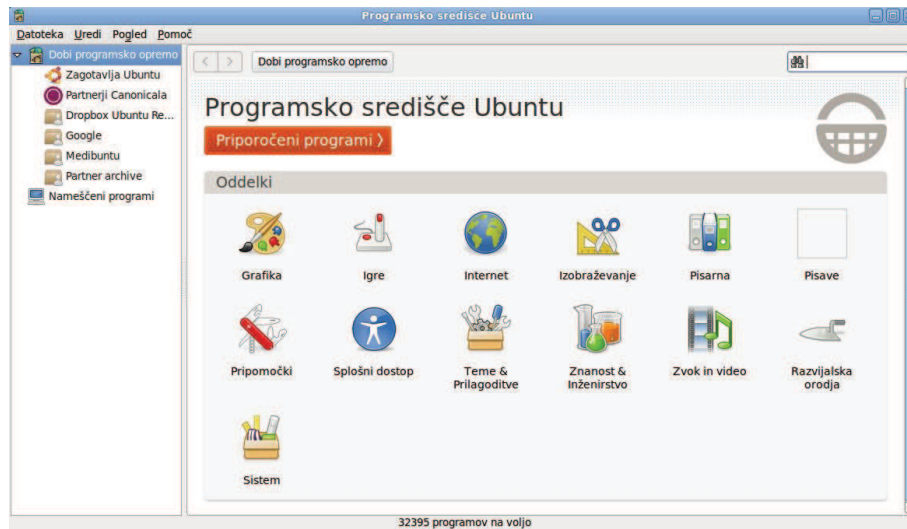
## 1.7 Kako do programskih orodij?

Dostop do programskih orodij je odvisen od izbranega operacijskega sistema, ki ga uporabljamo. Postopek dostopa v sistemu Ubuntu Linux je drugačen, kakor je le ta v okolju Windows. Za pričetek dela je potrebno namestiti operacijski sistem Ubuntu, ki je dostopen v več kot 25-ih različnih jezikih, tudi slovenščini. Pri dostopu do programskih orodij imamo na voljo dve različni možnosti. Prva omogoča, da v meniju *Programi* izberemo možnost *Programsko središče Ubuntu* (glej Sliko 1.3).

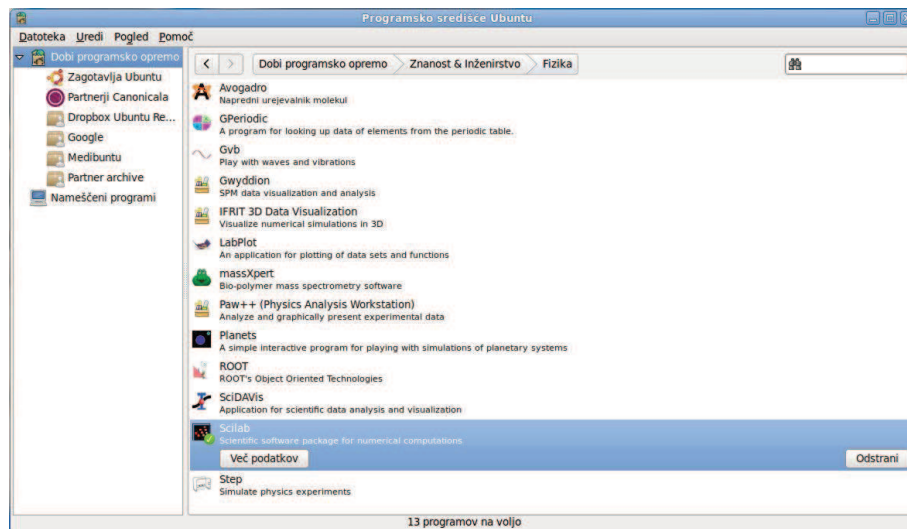
S klikom na dani meni se prikaže okno s priporočenimi programskimi orodji, razvrščenih v trinajst oddelkov. Gre za enega izmed načinov namestitve, ki je preprost in enostaven za uporabnika (glej Sliko 1.4).

Kot primer prikažemo namestitev programskega orodja Scilab. Programsko orodje poiščemo v oddelku *Znanost & Inženirstvo*, pododdelek *Fizika*. S klikom na gumb *Namesti* operacijski sistem namesti program na ustrezno mesto v osnovnem meniju *Programi*. V primeru, če bi želeli določen program odstraniti, ponovimo postopek iskanja določenega programskega orodja, ki ga odstranimo s klikom na *Odstrani* (glej Sliko 1.5).

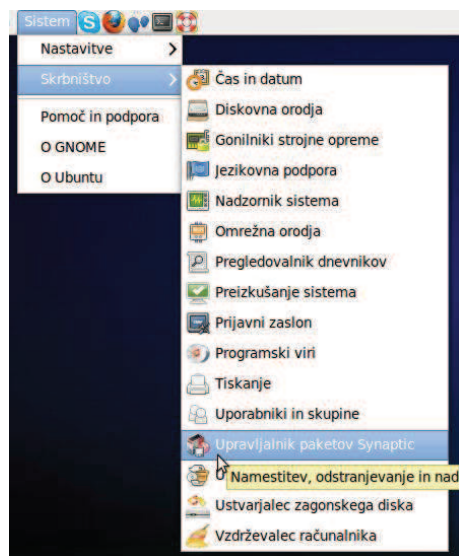
V primeru, da zelenega programa ne najdemo v izbranih oddelkih, so na voljo tudi druge možnosti namestitve. V orodni vrstici *Sistem* izberemo možnost *Skrbnišтво* in v



Slika 1.4: Programsko središče Ubuntu



Slika 1.5: Nameščanje ali odstranjevanje programskega orodja



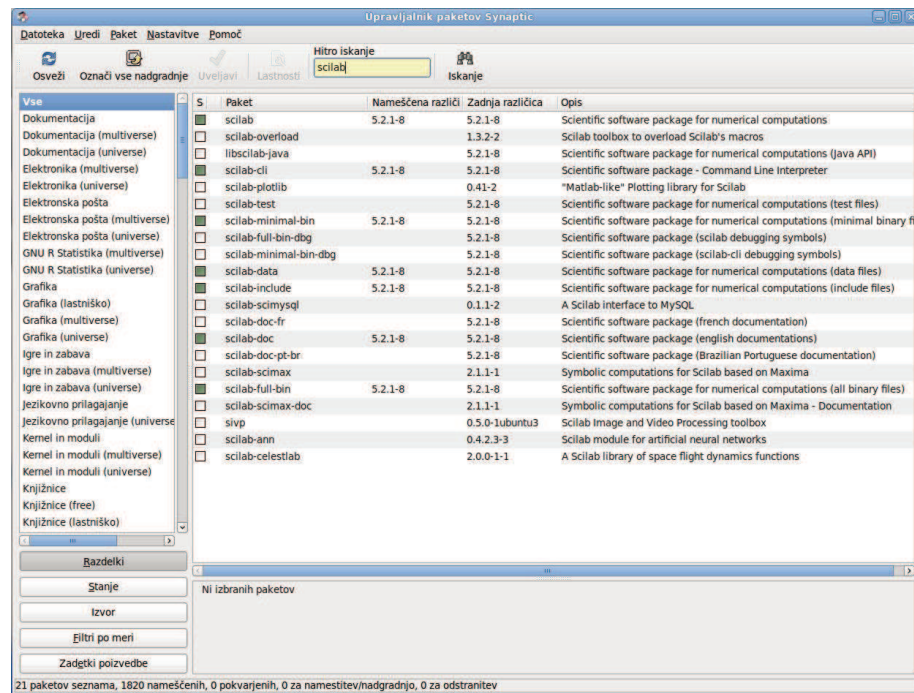
Slika 1.6: Upravljanje paketov Synaptic

nadaljevanju *Upravljanje paketov Synaptic* (glej Sliko 1.6).

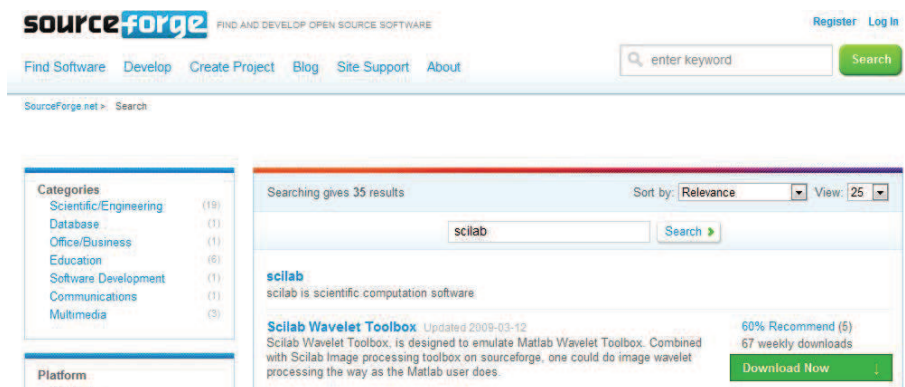
Z izbrano možnostjo lahko dostopamo do številnih drugih programskih orodij, datotek idr. Za pravilno delovanje Ubuntu-ovega sistema je potrebno namestiti številne datoteke povsem samostojno. Za iskanje programskih orodij v okence *Hitro iskanje* vnesemo naslov orodja, ki ga želimo namestiti. V izbranem primeru preverimo, ali lahko programsko orodje Scilab poiščemo še na drug način. V primeru, če še ni nameščen, ga namestimo s klikom na *paket Scilab*. Pozorni moramo biti predvsem, da se program namesti v celoti, brez prekinitev in sočasnega izvajanja posodobitev (glej Sliko 1.7).

V operacijskem sistemu Windows poteka nameščanje programskega orodja na drugačen način. Običajno si posamezna programska orodja namestimo preko njihove uradne spletne strani ali pa preko t.i. "source forge" spletne strani. Vse kar potrebujemo za namestitev programskega orodja, je torej dostop do interneta (glej Sliko 1.8).

Verzije programskih orodij se razlikujejo, odvisno kateri operacijski sistem uporabljamo in katero pot namestitve izberemo. V praksi se dogaja, da uporabniki operacijskega sistema Ubuntu ali drugih Linux operacijskih sistemov, ki namestijo programsko orodje s pomočjo upravljanja paketov Synaptic, dobijo starejšo a delujočo verzijo, ki se razlikuje od tistih, ki jih namestijo uporabniki operacijskega sistema Windows s spleta. V kolikor želimo imeti enake verzije programskega orodja v obeh operacijskih sistemih Ubuntu ali Windows, moramo preveriti verzijo, ki jo je namestil upravljalca paketov Synaptic in jo poiskati na spletu.



Slika 1.7: Nameščanje programskega orodja



Slika 1.8: Source forge

## 1.8 Področja raziskovanja

Programska orodja smo povezali v sedem smiselno razdeljenih področij.

### Področje 1 - Procesi

V področje *Procesi* umeščamo tri programska orodja. Prvo programsko orodje *Planner* je v pomoč pri projektnem vodenju, ki opredeljuje prikaz procesa načrtovanja, nadzorovanja in poročanja aktivnosti v podjetju. V fazi načrtovanja je smiselna izdelava Ganttovega diagrama, s pomočjo katerega vizualno prikažemo dogajanje oz. trajanje določenih nalog znotraj projekta.

V nadaljevanju uporabimo programsko orodje *Dia* in izdelamo diagram poteka, ki prikazuje prehode iz ene aktivnosti na drugo aktivnost. Vizualno prikažemo proces oskrbe z izbrano komponento v podjetju (proces naročanja, dostave, izmenjave dokumentov, grobega in finega prevzema platišč, reklamacij in skladiščenja).

Obvladovanje oskrbne verige in zagotovitev sledljivosti komponent na vhodni strani sistema in izdelkov na izhodni strani sistema, predvsem v procesu transporta in skladiščenja, zahteva poznavanje in uporabo standardov označevanja in identifikacije logističnih enot, kar omogoča programsko orodje *ZINT*, s katerim prikažemo implementacijo črtne kode v poslovanju izbranega podjetja. *ZINT* je ustrezen pripomoček za začetnike uvajanja črtnih kod v poslovne procese.

### Področje 2 - Vizualizacija

V drugo področje umeščamo *Vizualizacijo* oskrbne verige. S programskim orodjem *ASDN Logistics Analysis* (v nadaljevanju *ASDN*) predstavimo integracijo oskrbne verige - od dobavitelja komponent do izbranega podjetja, saj je združevanje posameznih segmentov nadvse pomembno. *ASDN* je v pomoč pri oblikovanju industrijskih logističnih mrež, ki prispevajo k izboljšanju industrijskih omrežij z optimizacijo zalog in stroškov.

V vsakdanjem življenju se pogosto srečujemo s problemom mobilnosti in pomanjkanjem časa. Pogosto se zgodi, da se nenadoma odločimo za poslovno potovanje v tujino, čeprav niti ne vemo, kako bomo tja prispeli. V takšnih primerih uporabimo programsko orodje *Google Zemlja*, s katerim prikažemo načrtovanje poti z digitalnimi zemljevidi. Izmed številnih segmentov, ki jih ponuja programsko orodje, se osredotočimo na načrtovanje poti od namišljenega poslovnega partnerja v Italiji do našega podjetja.

V izbrano področje vključimo tudi programsko orodje *Quantum GIS*, ki omogoča ustvarjanje, vizualizacijo, poizvedbe in analizo geoprostorskih podatkov. S *Quantum GIS* na podlagi pridobljenih slojev (karta potresnega območja, karta poplavnega območja, karta hrupa ipd.) vizualno opredelimo primernost lokacije za izgradnjo skladiščnega objekta z upoštevanjem tveganj.

### Področje 3 - Prostorsko planiranje

Tretje področje je področje *Prostorskega planiranja*, kamor umestimo programski orodji *Qcad* in *Simple Warehouse Mapper*. Logistika kot dejavnost se ukvarja z upravljanjem različnih tokov v oskrbni verigi ali znotraj podjetja. Pri tem zajema vse procese, ki se izvajajo in pri obvladovanju le teh potrebuje različna znanja. Mednje umestimo tudi poznavanje in uporabo *Qcad* orodja, ki pripomore k vizualizaciji problemov, prostorov ali procesov. Z njegovo pomočjo izrišemo skladišni prostor v 2-D skici, kjer so uskladiščene komponente za proizvodnjo avtomobila.

Z odprtokodnim programskim orodjem *Simple Warehouse Mapper* na podlagi že izdelanega skladišnega prostora prikažemo enostaven model ureditve in razporeditve skladišnega prostora. Uporablja preprosto 3D simulacijsko tehniko, znano kot izometrični zemljevid. Z njim ustvarimo vizualno podobo izbranega objekta s pripadajočimi podatki in omogočimo predstavitev dejanskega stanja v realnem svetu.

### Področje 4 - Odločitve

*Odločitve* so pomembno področje na vseh področjih poslovanja v podjetju. Znotraj izbranega področja umestimo pet različnih programskih orodij. *Petersen* programsko orodje je praktično, enostavno in brezplačno s pomočjo katerega prikažemo izračun optimalne dostave komponent od dobavitelja do izmišljenega podjetja *OpenStorage*.

Zaradi velikega števila numeričnih operacij s programskim orodjem *Lindo* predstavimo problem izbire transporta platišč, ki jih je potrebno dostavljati v določenem obdobju. Programsko orodje uporabimo za optimizacijo stroškov z linearnim in celoštevilskim programiranjem. Izdelamo model, ki prikazuje optimalno izbiro prevoznega sredstva ob izbranih omejitvah.

V nadaljevanju z uporabo programskega orodja *DEXi* predstavimo odločitveni model, s katerim določimo optimalni izbor nakupa viličarja. Določimo kateri viličar je najboljša rešitev za naše potrebe in zahteve. Enostavna razčlenitev kriterijev, opredelitev različnih variant in njihovo vrednotenje poda rešitev izbire najboljšega. S kakovostno raziskavo in primerjavo lahko uspešno nastopimo pred vodstvom in predstavimo izbiro za nakup novega viličarja, katerega izberemo s pomočjo večparametrskega odločitvenega modela.

Pri poslovanju nikakor ne moremo mimo financ, ki so povezane z različnimi zahtevami, problemi in ovirami. S programskim orodjem *GnuCash* prikažemo vodenje računa pri investiciji v nakup sodobne tehnologije. Doseganje učinkovitosti in pretočnosti materialnega toka je odvisno od celotnega delovanja sistema oskrbne verige. Časovne zamude v fazi skladiščenja bistveno vplivajo na celotno pretočnost in podaljšanje procesov oskrbne verige. Hitro ukrepanje in investicija v nakup sodobne tehnologija, ponudita uporabniku možnost, tako optimizacije poslovanja, kot dosego višje pretočnosti materialnega toka in zmanjšanje stroškov.

Z uporabo *GPSS World* simulacijskega orodja je mogoče predvideti učinke modeliranja in izbire realnega kompleksnega sistema. Gre za celovito orodje za modeliranje simulacij dogodkov, ki ohranja simulacijsko okolje z dodano visoko

stopnjo interaktivnosti in vizualizacije. S simuliranjem materialnega toka treh različnih komponent (pnevmatike, platišča in vijaki) in spreminjanjem vrednosti (čas, količine) prikažemo pravilno izbiro modela oskrbne verige s finančnim vidikom spreminjanja vrednosti. Sistematično planiranje oskrbne verige v fazi izvedbe dostave in skladiščenja potrebnih komponent, zagotavlja učinkovitost z visoko stopnjo odzivnosti vseh deležnikov v oskrbni verigi, kar rezultira večjo pretočnosti materialnega toka.

### Področje 5 - Napovedovanje

Z napovedovanjem ocenjujemo verjetne razvoje v prihodnosti. Planiranje ne oznanja samo verjetnega, ampak na osnovi verjetnega postavlja zaželeno. Peto področje je osredotočeno na *Napovedovanje*, predvsem s programskim orodjem *Scilab*. V namišljenem podjetju *OpenStorage* se ukvarjamo s skladiščenjem treh vrst komponent, namenjenih izdelavi osebne avtomobila. Na podlagi zbranih podatkov o nakupu v preteklih 11 letih, izvedemo napoved prodaje avtomobilov za obdobje 5 let. Ob zapisu danega programa in dobljenih rezultatov preverimo ali je izbrana metoda pravilna oz. ali jo je za nadaljnjo delo potrebno spremeniti.

Skozi drugačen pogled s programskim orodjem *Open.Office.org Preglednica* (v nadaljevanju *OOo Preglednica*) izdelamo model s katerim prikažemo izračun stanja zaloga v skladišču. Upoštevamo začetno stanje zaloga, vrednost posamezne komponente in kapaciteto skladišča. Pri tem tedensko vodimo stanje treh komponent. Izdelan model omogoča prihodnjo napoved potreb po komponentah.

### Področje 6 - Analiza

*Statistična analiza podatkov* je ena izmed pomembnih aktivnosti na vseh področjih raziskovanja. V podjetjih jo pogosto uporabljamo, kadar želimo analizirati določeno količino podatkov, se osredotočiti zgolj na en vidik posameznega procesa ipd. Običajno se izvaja na podlagi že zbranih podatkov ali z zbiranjem podatkov z anketnimi vprašalniki. V šestem področju s programskim orodjem *PSPP* izvedemo analizo podatkov zbranih s pomočjo anketnega vprašalnika. Zanima nas zadovoljstvo zaposlenih v namišljenem podjetju *OpenStorage* z delovnim mestom v skladišču, natančneje na oddelku, ki se ukvarja s postopkom prevzema pnevmatik. Pomembno je, da se pri izbranem problemu ne osredotočimo zgolj na matematične izračune, ampak da pri izboljšanju delovnih navad prisostvuje tudi mnenje zaposlenih.

### Področje 7 - Upravljanje vsebin

*Sistemi za upravljanje vsebin* so bili ustvarjeni za lažje urejanje spletnih vsebin. Koncept teh sistemov je, da uporabnik, ki ni več v programskem jeziku HTML ureja in vzdržuje spletne strani. Uporabniku takšen sistem omogoča postavitev spletne strani, dodajanje novih strani, urejanje ipd. Ko upravljamo s podjetjem je pomembno, da znamo nastopati ne le lokalno, temveč tudi na globalnih trgih. S programskim orodjem *WordPress*

izdelamo spletno stran, ki bo za podjetje OpenStorage omogočila večjo prepoznavnost. Ažurnost podjetij je v današnjem času nujno potrebna za digitalni obstoj.

## 1.9 Nadomestljivost plačljivih programskih orodij

Prikazati želimo, da se je moč posluževati številnih odprtokodnih in prosto dostopnih programskih orodij povsem brezplačno, čeprav pri študentih še niso pogosto v uporabi. Prosto dostopna so že dlje časa na spletnih straneh, vendar so kljub temu mnogim še nepoznana. Bralca želimo skozi posamezna poglavja popeljati v določen segment logističnega problema, katerega z ustreznim programskim orodjem razrešimo in v nadaljevanju prikažemo rezultate, jih natančneje opišemo ter smiselno povežemo v zaključeno celoto.

Na spletu so dostopna številna programska orodja, ki jih uporabimo v študijske namene in delo na projektih. Pogosto se zgodi, da potrebujemo določeno programsko orodje, katerega cene so študentom nedosegljive. S strani izobraževalnih ali drugih inštitucij je moč pridobiti študentske verzije, ki so pogosto necelostne, kar pomeni, da je programsko orodje moč uporabiti le z z določenimi omejitvami.

Tabeli 1.1 in 1.2 prikazujeta uporabljena programska orodja in primerjalna (nadomestna) programska orodja. Prikazuje še nekaj drugih programskih orodij, ki jih lahko preizkusite sami. Brezplačne programske rešitve so v praksi specializirane predvsem za določena področja in ne zajemajo širših področij, kot plačljive, zato v priročniku povežemo različna brezplačna programska orodja in se tako skušamo približati plačljivim.



Programska orodja				
Področje	Uporabljena	Druga brezplačna	Plačljiva	Uporaba
Procesi	Planner 0.14.4	Open Workbench	Microsoft Project	Načrtovanje projektov
	Dia 0.97.1	OOo Impress, Violet, StarUML	Microsoft Visio, Gliffy, SmartDraw	Načrtovanje diagramov
	ZINT 2.3.2	Online Barcode Generator	DRPU Barcode Label Maker Software	Generiranje črtne kode
Vizualizacija	ASDN 1.217	/	SAP	Integracija oskrbne verige
	Google Zemlja 5.1	TangoGPS, GoogleMaps, Viking, Marble, Earth 3D	/	Načrtovanje poti z digitalnimi zemljevidi
	Quantum GIS 1.01	Grass GIS, Saga GIS	GIS	Geografsko informacijski sistem
Prostorsko planiranje	Qcad 2.0.5	Archimedes, BRL-CAD	Autocad, ProEngineering, Archicad	Projektiranje objektov, lokacij v 2D dimenzijah
	Simple Warehouse Mapper	/	ArchiCAD, AutoCAD	Vizualizacija tokov v 3D dimenzijah

Tabela 1.1: Programska orodja – 1. del

Programska orodja				
Področje	Uporabljena	Druga brezplačna	Plačljiva	Uporaba
Odločitve	Petersen 3.2.3	GraphThing	Matlab 7.0.1	Odločanje na podlagi teorije grafov
	Lindo 6.1	/	Lingo	Optimizacija stroškov z linearnim in celoštevilskim programiranjem
	Dexi 3.02	JDexi	Decision Pad	Odločitveni modeli
	Gnucash 2.2.9	Grisbi, Eqonomize	Quicken, AceMoney	Računovodsko orodje za finančno načrtovanje
	GPSS 5.2.2.	Simula, JGPPS	SIMAN, Arena	Simulacija "dogajanje" v skladišču in opredelitev prihodov
Napovedovanje	Scilab	Maxima	Matlab	Numerično reševanje in analiziranje podatkov
	OOo Calc 3.2.0	/	Microsoft Excel	Uporaba preglednic za prikaz gibanja zalog
Analiza	PSPP 0.602	Google Docs	SPSS	Statistična analiza podatkov
Upravljanje	WordPress	Drupal, Joomla, Xoops, SilverStripe, Cushy CMS, Frog CMS	Bitrix, way4CMS	Postavitev spletne strani

Tabela 1.2: Programska orodja – 2. del

## 1.10 Kategorije programske opreme

Programsko opremo delimo v več različnih zvrsti - od povsem osnovnih do precej kompleksnih, ki so izdelane s pomočjo diagramov za večjo preglednost. Slika 1.9 prikazuje delitev programske opreme na vizualni podlagi.

### 1.10.1 Prosta programska oprema

Prosto programsko opremo definiramo kot opremo, ki jo lahko zastonj ali proti plačilu vsakdo uporablja, kopira in dopolnjuje. Prost program je potencialno mogoče vključiti v prost operacijski sistem, kot je GNU, ali proste različice sistema GNU/Linux. Besedo "prost" je potrebno strogo ločiti od besede "brezplačen" (ang. *Freeware*) - tovrstna oprema je zelo podobna odprtokodni programski opremi.

### 1.10.2 Odprtokodna programska oprema

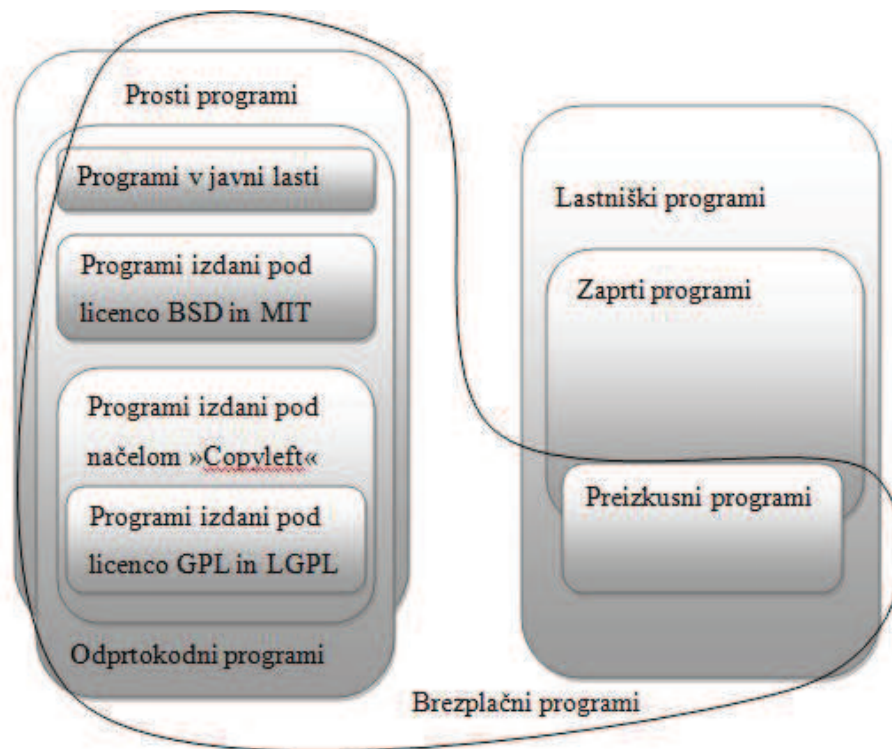
Razvoj odprtokodne programske opreme (*Open source software* - OSS) se prične kot marketinška kampanja za prosto programiranje. Lahko jo definiramo, kot programsko opremo, za katero je izvorna koda izdana pod zaščitno licenco (ali aranžma, kot npr. javna domena). Odprta koda dovoljuje vsakršno uporabo in spreminjanje programske opreme, njeno razširjanje v spremenjeni ali nespremenjeni obliki. Definiciji proste programske opreme in odprtokodne programske opreme sta skoraj identični. Bistvena razlika je v tem, da prva poudarja svobodno uporabo programske opreme, druga pa dostop do izvorne kode programske opreme [61].

Projekte OSS postavljajo in vzdržujejo prostovoljni programerji. Primeri odprtokodnih produktov so Apache HTTP Server, Internet Protocol in internetni brskalnik Mozilla Firefox. Med najuspešnejšimi programi je operacijski sistem Linux ter odprtokodni operacijski sistem Unix-like [61] (glej Sliko 1.10).

V praksi obstaja veliko splošno znanih ovir za prehod na odprtokodno programiranje. Podjetja so prepričana, da odprtokodne licence vsebujejo viruse, nimajo zadostne formalne podpore in izobraževanj, se hitro spreminjajo in ne sovpadajo z dolgoročnimi načrti. Večina teh ovir je povezanih s stopnjo tveganja. Potrebno je poudariti, da tudi lastniški projekti ne vsebujejo natančnih načrtov za prihodnost, vse odprtokodne licence niso enako virusne, veliko OSS-jevih projektov (še posebej operacijski sistemi) pa prinašajo dobiček iz plačane podpore in dokumentacije. Pogosto uporabljena poslovna strategija komercialnih odprtokodnih programskih podjetij je dvojno-licenčna strategija (*Dual-Licence Strategy*), katero uporabljajo tudi MySQL, Alfresco in drugi.

### Linux

Operacijski sistem Linux je ustvaril Linus Torvalds na finski univerzi v Helsinkih. Zanimal se je za Minix, majhen UNIX sistem, in se pri tem odločil, da razvije sistem,



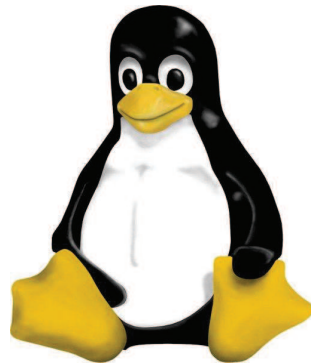
Slika 1.9: Povezave med kategorijami programske opreme

Vir: [112]



Slika 1.10: Odprtokodni produkti

Vir: [3] [36]



Slika 1.11: Odprtokodni produkti

Vir: [49]

ki bo presegel Minix-ove standarde. Z delom je pričel leta 1991 in v letu 1994 izdal prvo verzijo 1.0. Izdana je bila pod GNU licenco, izvorna koda pa je prosto dostopna. V vsakdanjem pogovoru se je ustalilo poimenovanje "Linux" za celoten operacijski sistem plus, kar ni povsem točno, saj se poimenovanje Linux nanaša le na Linuxovo jedro. Samega jedra kot takega ni nemogoče s pridom uporabljati. Uporabno je le kot del celotnega operacijskega sistema, zato se Linux običajno uporablja kot sestavni del operacijskega sistema GNU. Tako govorimo o GNU/Linux, katerega maskota je sedeči pingvin z imenom Tux (Slika 1.11), ustvarjalca Larryja Ewinga [21].

Pridobitev operacijskega sistema Linux je povsem brezplačna. Vsakdo si ga lahko enostavno namesti preko spleta. V svetu so poznane številne distribucije Linuxa, ki so navedene v naslednjih odstavkih.

Ubuntu Linux – Beseda "Ubuntu" je starodavnega afriškega izvora in pomeni "človečnost za druge". Gre za dovršen operacijski sistem, temelječ na distribuciji Debian. Ubuntu skupnost želi, brezplačen dostop do vse programske opreme, uporabnost programskih orodij v lokalnem jeziku in svobodo za prilagajanje programske opreme na kakršen koli ustrezen način.

Pingo Linux - slovenska distribucija Linux. Gre za preprost, poslovenjen in uporabniku prijazen operacijski sistem, ki združuje poslovenjeni namizji KDE in Gnome, poslovenjeno pisarniško zbirko OpenOffice.org, spletni brskalnik in poštni program Mozilla, poslovenjen namestitveni program, multimedijske aplikacije in orodja za upravljanje s sistemom.

Ferix GNU/Linux – slovenska distribucija Linux namenjenega izobraževanju. Združuje poslovenjeni namizji KDE in Gnome, poslovenjeno pisarniško zbirko OpenOffice.org, spletni brskalnik in poštni program Mozilla, poslovenjen namestitveni program, multimedijske aplikacije in orodja za upravljanje s sistemom ter seveda izobraževalne programe za vse starostne skupine. Druge Linux distribucije so še: *Slax*,



Slika 1.12: Ubuntu

Vir: [26]

*Konppix, Mandriva Linux in SuSE Linux.*

V slovenskem jeziku je izšla prosto dostopna knjiga Linux na namizju, katero uporabniki Linux brezplačno prenesejo iz spletnega naslova: Linux na namizju [30]. Linux na namizju je v pomoč pri preučevanju Linux operacijskega sistema in sami uporabi Ubuntu Linux.

## Ubuntu Linux

Kot smo že omenili je Ubuntu Linux ena izmed številnih Linux distribucij. Temelji na znani Linux distribuciji Debian, ki slovi predvsem po varnosti in hitremu sistemu za nameščanje programskih paketov. Uporablja namizno okolje Gnome, na voljo je tudi posebna različica za KDE. Na sistem je kasneje možno namestiti tudi poljubno namizno okolje. Ubuntu Linux je na voljo za 32 in 64-bitne sisteme. Izgled sistema lahko povsem prilagodimo operacijskemu sistemu Windows. Na voljo je tudi v slovenskem jeziku.

V primeru, da ima uporabnik na računalniku že nameščen operacijski sistem Windows (oz. kakšen drug sistem), ga Ubuntu ne "povozi", pač pa omogoči, da na istem računalniku vzporedno uporabljamo več operacijskih sistemov.

## 1.11 Vrste licenc

### 1.11.1 Licenca GNU GPL

Kadar govorimo o Linux je potrebno poudariti še t.i. licenco GNU GPL, ki dovoljuje prosto razmnoževanje in urejanje ter izboljševanje programske kode. Uporaba Linux zato pomeni uporabo licenčne in povsem zakonite programske opreme. GNU GPL licenca uporabniku omogoča poganjanje programa za kakršenkoli namen - uporabnik

lahko preučuje, kako program deluje in program prilagaja svojim potrebam, prosto razširja kopije programa, program izboljšuje in daje svoje izboljšave na voljo javnosti.

Običajno so licenčne pogodbe zasnovane tako, da uporabniku preprečujejo njegovo svobodno razdeljevanje in spreminjanje. Za razliko od teh *Splošno dovoljenje GNU* (angl. *GNU General Public License, GPL*) jamči svobodo pri razdeljevanju in spreminjanju prostega programiranja in s tem zagotavlja, da programi ostanejo prosti za vse uporabnike. Dovoljenje GNU uporabniku zagotavlja pravico razširjati kopije prostega programiranja (in zaračunavati za to storitev, če tako želi).

### Določitve in pogoji za razmnoževanje, razširjanje in spreminjanje

Licenca je povzeta po spletnem naslovu [Lugos.si](http://Lugos.si), prevajalec Roman Maurer [31].

(0) "Licenca se nanaša na vsak program ali drugo delo, ki vsebuje obvestilo lastnika avtorskih pravic... z izjavo, da se lahko distribuira pod pogoji Splošnega dovoljenja GNU... Ta licenca ne pokriva nobenih drugih aktivnosti razen razmnoževanja, razširjanja in sprememb; ostale so izven njenega dometa. Dejanje poganjanja programa ni omejeno in izhod programa je zajet le, če njegova vsebina sestavlja delo, iz katerega je izpeljan program (ne glede na to, da je bil narejen s poganjanjem programa)... " [31].

(1) "Razmnožujete in razširjate lahko dobesedne izvode izvirne kode programa v enaki obliki, kot jo dobite, preko kateregakoli medija, če le na vsakem izvodu razločno in primerno objavite obvestilo o pravicah razširjanja in zanikanje jamstva... Za fizično dejanje prenosa kopije lahko zaračunavate in po vaši presoji lahko ponudite garancijsko zaščito v zameno za plačilo" [31].

(2) "Spreminjati smete vaš izvod ali izvode programa ali katerikoli njegov del, in tako narediti delo, ki temelji na programu, ter razmnoževati in razširjati takšne spremembe ali dela pod pogoji zgornjega razdelka 1, če zadostite tudi vsem naslednjim pogojem: (a) Zagotoviti morate, da spremenjene datoteke nosijo vidna obvestila o tem, da ste jih spremenili in datum vsake spremembe. (b) Zagotoviti morate, da je vsako delo, ki ga razširjate ali izdajate in ki v celoti ali deloma vsebuje program ali katerikoli njegov del ali pa je iz njega izpeljano, licencirano pod pogoji te licence kot celota brez plačila katerikoli tretji osebi (c) Če spremenjeni program ob zagonu navadno bere ukaze interaktivno, morate zagotoviti, da se ob najbolj običajnem zagonu za takšno interaktivno uporabo izpiše ali prikaže najava, ki vključuje primerno sporočilo o pravicah razširjanja in sporočilo, da jamstvo ni zagotovljeno... " [31].

(3) "Program (ali delo, ki temelji na njem, pod razdelkom 2) lahko razmnožujete in razširjate v objektni kodi ali izvedljivi obliki pod pogoji zgornjih razdelkov 1 in 2, če izpolnite tudi kaj od tega: (a) Opremite ga s popolno in ustrezno izvirno kodo v strojno berljivi obliki, ki mora biti razširjana pod pogoji zgornjih razdelkov 1 in 2 na mediju, ki se navadno uporablja za izmenjavo programja; ali, (b) Opremite ga z napisano ponudbo, veljavno vsaj tri leta, da boste katerikoli tretji osebi, za plačilo, ki ne bo presevalo vaših stroškov fizičnega izvajanja izvirne distribucije, dali popoln izvod ustrezne izvirne kode v strojno berljivi obliki, ki bo razširjana pod pogoji zgornjih razdelkov 1 in 2 na mediju, ki se običajno uporablja za izmenjavo programja; ali, (c) Opremite ga z

informacijo, ki ste jo dobili vi, kot ponudbo distribucije ustrezne izvorne kode. (Ta alternativa je dovoljena le za nekomercialne distribucije in le, če ste dobili program v obliki izvorne kode ali izvedljivi obliki s takšno ponudbo, glede na podrazdelek b, zgoraj.)" [31].

(4) "Ne smete razmnoževati, spreminjati, podlicencirati ali razširjati programa drugače, kot to izrecno določa pričujoča licenca. Vsak poskus siceršnjega kopiranja, spreminjanja, podlicenciranja ali razširjanja programa je nič in bo samodejno prekinil vaše pravice pod to licenco. Vendar pa se osebam, ki so svoj izvod ali pravice dobile od vas pod to licenco, licenca ne prekine, dokler se ji popolnoma podrejšo" [31].

(5) "Ni vam treba sprejeti te licence, saj je niste podpisali. Vendar vam razen nje nič ne dovoljuje spreminjanja ali razširjanja programa ali iz njega izpeljanih del. Če ne sprejmete te licence, ta dejanja prepoveduje zakon. Torej, s spremembo ali razširjanjem programa (ali katerikoli dela, ki temelji na programu), pokažete svoje strinjanje s to licenco in z vsemi njenimi določitvami in pogoji za razmnoževanje, razširjanje ali spreminjanje programa ali del, ki temeljijo na njem" [31].

(6) "Vsakič, ko razširjate program (ali katerikoli delo, ki temelji na programu), prejemnik samodejno prejme licenco od izvirnega izdajatelja licence (angl. original licensor) za razmnoževanje, razširjanje ali spreminjanje programa glede na ta določila in pogoje. Ne smete vsiljevati nobenih nadaljnjih omejitev izvajanja prejemnikovih pravic, podeljenih tukaj. Niste odgovorni za vsiljevanje strinjanja tretjih oseb s to licenco" [31].

(7) "Če so vam, kot posledica presoje sodišča ali suma kršitve patenta ali zaradi katerekoli drugega razloga (ne omejenega zgolj na patentna vprašanja), vsiljeni pogoji (bodisi z odlokom sodišča, sporazumom ali drugače), ki nasprotujejo pogojem te licence, vas ne odvezujejo pogojev te licence. . . Ta razdelek namerava temeljito pojasniti, kaj so predvidene posledice nadaljevanja licence" [31].

(8) "Če sta razširjanje in/ali uporaba programa omejena v določenih državah, bodisi zaradi patentov ali vmesnikov s posebno pravico razširjanja (angl. copyrighted interfaces), lahko izvorni lastnik ali lastnica pravic razširjanja, ki postavlja program pod to licenco, doda eksplicitno zemljepisno omejitev razširjanja, ki izključuje te države, tako da je razširjanje dovoljeno le v in med državami, ki niso na tak način izključene. V takem primeru ta licenca vključuje omejitve, kot da so napisane v telesu te licence" [31].

(9) "Ustanova Free Software Foundation lahko od časa do časa izdaja preurejene in/ali nove različice Splošne javne licence (angl. *General Public License*). Nove različice bodo pisane v duhu trenutne različice, vendar se lahko razlikujejo v podrobnostih, ki bodo obdelovale nove težave ali poglede. Vsaki različici je prirejena razločevalna številka različice. Če program določa številko različice te licence, ki se nanaša na njo in "na katerekoli poznejše različice", imate izbiro upoštevanja pogojev in določil bodisi te različice ali katerekoli poznejše različice, ki jo je izdala ustanova Free Software Foundation. Če program ne določa številke različice te licence, lahko izberete katerikoli različico, ki jo je kdajkoli izdala ustanova Free Software Foundation" [31].

(10) "Če želite vključiti dele programa v druge proste programe, katerih pogoji razširjanja so drugačni, pišite avtorju in ga prosite za dovoljenje. Za programje, katerega pravice razširjanja ima Free Software Foundation, pišite na Free Software Foundation;



včasih naredimo izjemo pri tem. Našo odločitev bosta vodila dva cilja: ohranitev prostega statusa vseh izvedenih del iz našega prostega programja in spodbujanje razdeljevanja in ponovne uporabe programja na splošno" [31].

### Brez jamstva

(11) "Ker je program licenciran kot brezplačen, ni nobenega jamstva za program do meje, ki jo določa pristojni zakon. Razen, če ni drugače napisano, imetniki pravic razširjanja in/ali druge osebe ponujajo program "tak kot je", brez zagotovila kakršnekoli vrste, neposrednega ali posrednega, kar vključuje, a ni omejeno na posredna jamstva cenovne vrednosti in primernosti za določeno uporabo. Celotno tveganje glede kakovosti in delovanja programa prevzamete sami. Če se program izkaže za okvarjenega, sami nosite stroške vseh potrebnih storitev, popravil ali popravkov" [31].

(12) "V nobenem primeru, razen če tako pravi veljavni zakon ali je pisno dogovorjeno, ne bo lastnik pravic razširjanja ali katerakoli druga oseba, ki lahko spremeni in/ali ponovno razširja program, kot je to dovoljeno zgoraj, prevzel odgovornosti zaradi škode, najsi gre za splošno, posebno, nenamerno škodo ali škodo, izhajajočo iz uporabe ali nezmožnosti uporabe programa (vključno z, a ne omejeno na, izgubo podatkov ali nenatančno obdelavo podatkov ali izgubo, povzročeno vam ali tretjim osebam ali nezmožnost programa, da bi deloval s kakim drugim programom), četudi je bil tak lastnik ali druga oseba obveščen o možnosti nastanka takšne škode" [31].

## 1.11.2 Druge odprtokodne in prosto dostopne licence

Uporabniki odprtokodnih programov lahko izbirajo še med ostalimi ponudniki programske opreme. Licence z odprto kodo se v veliki meri razlikujejo. Med najbolj uporabljenimi so GPL, katero smo že opisali, BSD, LGPL in MPL.

*Berkeley Software Distribution (BSD)* licenca je preprosta, saj dovoljuje uporabo programske opreme, distribucijo izdelka in izvorne kode. Dovoljuje spreminjanje in vključevanje v drugo programsko opremo brez omejitev. Zahteva, ki jo je potrebno upoštevati je navedba avtorjev v izvorni kodi in dokumentaciji programa. Imen avtorjev ni dovoljeno uporabljati za promocijo izdelka brez predhodnega pisnega dovoljenja [61].

Licenca *Library General Public License (LGPL)* je variacija GPL licence in je namenjena programskim knjižnicam (na primer DLL), torej programom s funkcijami, ki jih je mogoče izkoristiti v drugih programih. Licenca *Mozilla Public License (MPL)* dovoljuje brezplačno uporabo in distribucijo programske opreme. Vsakdo, ki razpečuje spremembe ali dodatke k izdelku, mora dovoliti enake pravice za njegov kos programske opreme in kombinacijo izvirnega dela in njegove rešitve. Pogoj za distribucijo je tudi izvorna koda, ki je lahko vključena v distribucijo ali pa je dostopna na kakšni spletni strani. MPL je dovoljeno vključevati tudi v licenčno programsko kodo lastnih izdelkov, a mora biti za ta del na voljo izvorna koda [61].

BSD licenca temelji na načelih proste programske opreme (*Free Software*) in je prirejena za istega uporabnika. Edini pogoj, ki ga je potrebno vključiti je informacija o

avtorstvu programske opreme in licenciranje vsebin.

*Common Public License* (CPL) licenca je prav tako prosto programska oprema. Leta 1988 je bila odobrena s strani *Open Source* in je podobna opisanemu dovoljenju "GNU *General Public License*".

### 1.11.3 Študentska licenca

Mnogo katero programsko orodje ni prosto dostopno na spletu, vendar ga kljub temu lahko pridobimo, kar omogoča študentska verzija (*Student Version*) licence. Na podlagi pridobljenih t.i. študentskih verzij s strani izbrane fakultete je omogočena enoletna, večletna ali dolgoročna uporaba posameznega programskega orodja. Običajno se pojavi razlika med študentskimi in praviimi verzijami, ki jih je potrebno plačati, v okrnjenosti delovanja programskega orodja. Kot primer navedimo programsko orodje GPSS, namenjeno simulacijam. Študentje imajo možnost pridobitve študentske verzije na spletu ali na fakulteti, pri čemer je število blokov študentske verzije omejeno na 180, plačljiva verzija pa v okviru možnosti vsebuje 2000 blokov.

Študentske verzije so namenjene zgolj k spoznavanju dela z določenim programskim orodjem in k reševanju enostavnih problemov z manjšim številom podatkov. To seveda variira pri vsaki posamezni opremi in lastniškem podjetju posebej. Nekatera podjetja omogočajo pridobitev študentske licence za določene programe zgolj s posredovanjem originalnega potrdila o šolanju, z vsemi vsebovanimi moduli. V določenih primerih je za prijavo potrebno posredovati podatke izbrane fakultete oz. izbrana fakulteta pridobi dovoljenje in nato študentje uporabljajo program s časovno omejenim obdobjem.

### 1.11.4 Licenca za prosto verzijo

V praksi se pogosto znajdemo v situaciji, ko potrebujemo določeno programsko orodje le za kratek čas. Nakup le tega predstavlja prevelik strošek, glede na dejstvo, da ga potrebujemo npr. zgolj za določen projekt. V takšnem primeru so na voljo t.i. licence za poskusno verzijo. Praksa je, da lahko vsi uporabniki prosto uporabljajo program v poskusnem obdobju trideset (30) dni, s pričetkom štetja od prve aktivacije licence za poskusno verzijo na uporabnikovem računalniku.

Ko se odločamo o bodočem nakupu in želimo preučiti več različnih programskih orodij, lahko uporabimo t.i. *shareware licence*. Dostopne so dovolj časa, da preučimo posamezen program in ga glede na zahtevane značilnosti primerjamo z ostalimi. Gre za licenco s simbolnim plačilom, ki po namestitvi omogoča uporabo vseh funkcij programa za določeno obdobje. V primeru nadalje rabe programskega orodja je po preteku časa potrebno kupiti celotno licenco (npr. protivirusni program NOD32 lahko testiramo 30 dni, nato moramo kupiti polno licenco).

Programska orodja				
Programsko orodje	Vrsta	Licenca GNU GPL	Uporaba	Omejitve
Planner	Odprtokoden	Da	Sistem	Ne
Dia	Odprtokoden	Da	Sistem	Ne
Zint	Odprtokoden	Da	Sistem	Ne
ASDN	Odprtokoden	Da	Interne	Ne
Google Zemlja	Prosto dostopen	Ne	Internet	Ne
Quantum GIS	Odprtokoden	Da	Sistem	Ne
Qcad	Odprtokoden	Da	Sistem	Ne
Simple Warehouse Mapper	Prosto dostopen	Ne	Sistem	Ne
Petersen	Prosto dostopen	Ne	Java	Da
Lindo	Prosto dostopen	Ne	Sistem	Da
Dexi	Prosto dostopen	Da	Sistem	Ne
GnuCash	Odprtokoden	Da	Sistem	Ne
GPSS World	Prosto dostopen	Da	Sistem	Da
Scilab	Odprtokoden	Da	Sistem	Ne
OpenOffice.org Preglednica	Odprtokoden	Da	Sistem	Ne
PSPP	Odprtokoden	Da	Sistem	Ne

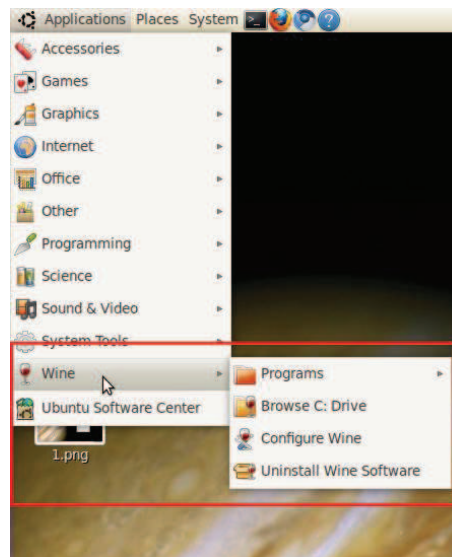
Tabela 1.3: Programska orodja

## 1.12 Kako uporabljati programska orodja?

V nadaljevanju predstavljamo uporabo programskih orodij napisanih za Windows okolje v operacijskem sistemu Ubuntu (glej Tabelo 1.3).

V izbor programskih orodij uvrstimo tudi nekaj programskih orodij, ki so napisana izključno za uporabnike operacijskega sistema Windows. V kolikor je program napisan zgolj za eno okolje in brez odprtokodne licence, potem izgubi pomen odprtokodnega programa v točki delovanja na obeh platformah (Windows ali Linux) in dostopa do izvorne kode. Ko so programska orodja brezplačna in javno dostopna ustrezajo definiciji proste programske opreme, ki se od odprtokodne razlikuje le v dostopu do izvorne kode.

Preizkusimo delovanje vseh izbranih programskih orodij v različnih okoljih, Windows in Ubuntu. Preizkus potrdi, da nekatera programska orodja napisana za okolje Windows ne delujejo v Ubuntu. Za takšna programska orodja poiščemo rešitev - program Wine. Z uporabo programa Wine vsa izbrana programska orodja napisana za okolje Windows delujejo tudi v Ubuntu.



Slika 1.13: Program Wine

### Program Wine

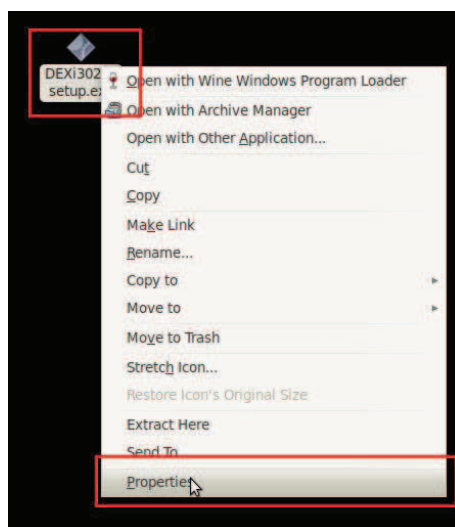
Wine je program, ki ponuja uporabnikom Linux operacijskih sistemov kompatibilnost delovanja programskega orodja napisanega za okolje Windows. Program Wine lahko uporabniki operacijskega sistema Ubuntu enostavno poiščejo, prenesejo in namestijo s pomočjo upravljalca paketov Synaptic ali pa ga poiščejo na spletni strani Wine HQ [51]. Program je odprtokoden in izdan pod GNU GPL licenco [68]. Wine ima tudi svojo knjižnico Winelib, katero razvijalci dopolnjujejo s seznamom delujočih programskih orodij napisanih za okolje Microsoft Windows v Linux okoljih [66] (glej Sliko 1.13).

Uporaba programa Wine je za uporabnike Ubuntu enostavna, vseeno pa opišimo njegovo uporabo pri namestitvi programskega orodja napisanega za okolje Microsoft Windows. V kolikor želimo namestiti takšno programsko orodje, ga s pomočjo spleta ali upravljalca paketov Synaptic poiščemo in prenesemo. Namestitev izvedemo z desnim klikom na *.exe* datoteko, nakar izberemo *Properties* oz. *Nastavitve* (glej Sliko 1.14).

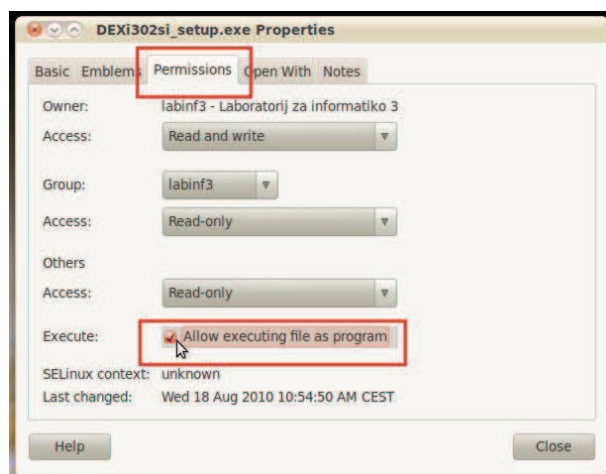
Kliknemo na *Permissions* oz. *Dovoljenja*, kjer izberemo možnost *Execute*, s čimer dovoljujemo namestitev dokumenta kot program (glej Sliko 1.15).

V razdelku *Open With* oz. *Odpri* izberemo *Wine Windows Program Loader* (glej Sliko 1.16).

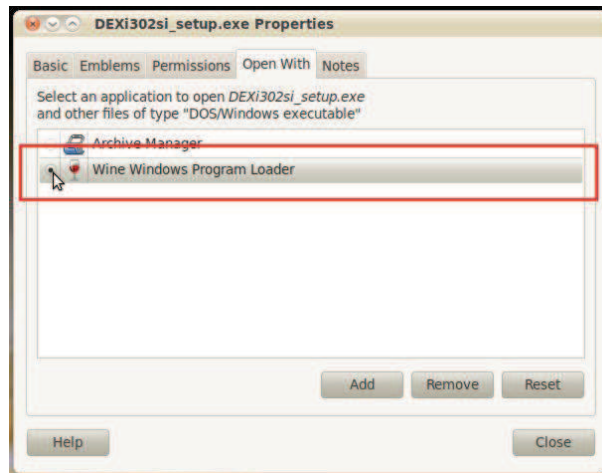
Zapremo in ponovno s desnim klikom kliknemo na *.exe* datoteko. Izberemo možnost *Open with Wine Windows Program Loader*, s čimer se prične namestitev programskega orodja, kakršno smo vajeni v Windows okolju (glej Sliko 1.17).



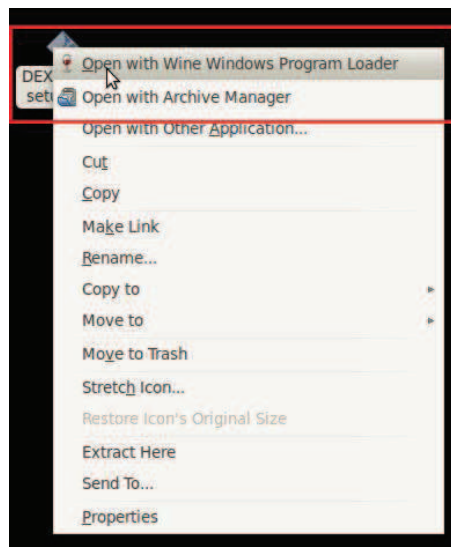
Slika 1.14: Namestitev programskega orodja s pomočjo programa Wine



Slika 1.15: Dovoljenje za odpiranje datoteke kot program



Slika 1.16: Nastavitev za odpiranje z Wine Windows Program Loader



Slika 1.17: Odpiranje z Wine Windows Program Loader