

III dalis. Atvirieji sprendimai ir globalieji tinklai

Šioje dalyje aptarsime populiarią interneto tarnybinių stočių programinę įrangą. Pastaruoju metu Lietuvoje ir pasaulyje greitai plinta informacinės sistemos, prie kurių prisijungiama interneto naršyklėmis. Tokiuose serveriuose paprastai veikia laisvoji programinė įranga.

Taip pat išsiaiškinsime, kaip skirtinga įranga besinaudojantys vartotojai gali sėkmingai naudotis minėtų informacinių sistemų paslaugomis, kaip atvirieji standartai užtikrina skirtingų kompiuterių ir kitų įrenginių sąveiką ir kaip tai lemia greitą tinklų augimą. Kodėl visą pasaulį apraizgė interneto tinklai, vėliau internetą užpildė ir jo plėtrą dar paskatino žiniatinklis.

12. Atvirieji standartai – interneto sėkmės šaltinis

Panagrinėkime, kaip universalios viešai paskelbtos bei atviros taisyklės leidžia atsirasti ir sėkmingai veikti sudėtingam tinklui, apimančiam visą pasaulį.

Internetas (tarptautinis tinklas, angl. *inter + net*) – pasaulinė kompiuterių tinklų sistema, jungianti visuotinius ir vietinius kompiuterių tinklus. Kompiuterių tinklai jungiami įvairiomis ryšio linijomis ir maršrutizatoriais, kurie iš vieno tinklo gautus duomenų paketus nukreipia į kitą tinklą. Ryšiui tarp kompiuterių naudojamas *TCP/IP* protokolas. Naudotojo kompiuteris prie interneto gali būti jungiamas naudojant modemą, *ISDN* įrangą, *DSL* įrangą, skirtinę arba optinę liniją, kabelinį modemą (kai interneto paslaugą teikia kabelinė televizija), radijo ryšį ar kitais būdais.

Sunku rasti žmogų, kuris nebūtų girdėjęs apie internetą, tačiau XX a. paskutiniame dešimtmetyje kai kurie ekspertai prognozavo komercinio uždaro tinklo, veikiančio pagal X.25 protokolą, plėtrą ir veikiančio pagal laisvą atvirą *TCP/IP* protokolą interneto mirtį. Tačiau nutiko atvirkščiai, nors internetas buvo kaltinamas nesaugumu, mėgėjiškumu ir kitais subjektyviais trūkumais, jis plėtėsi eksponentine progresija, nes buvo laisvas, visus priimančias ir nieko nediskriminuojantis. Tai buvo naujas verslo modelis, pasaulinis tinklas, neturintis savininko, nereguliuojantis kainų, neparduodantis licencijų, nereikalaujantis specialios įrangos, egzistuojantis pagal bendruomenės kuriamas taisykles, laisvas (nemokamas) tinklas, apie kurį sukasi didžiausi pasaulio pinigai.

12.1. Militaristinis poreikis lėmė laisvo tinklo atsiradimą

Interneto protokolo ištakos siekia laikotarpį, kai iškilo poreikis ginkluotę (raketas) valdyti nuotoliniu būdu, bet sistema turėjo išlikti veiksminga ir pažeidus vieną iš valdymo centrų ar ryšio linijų. Paaiškėjo, kad tai galima pasiekti sukūrus nepriklausomų serverių sistemą ir juos visus sujungus tarpusavyje. Kiekvienas mazgas veikia savarankiškai ir tinklas lieka veiksmingas tol, kol yra bent viena grandinė, jungianti komunikuojančius mazgus.

Tokia sistema lengvai plečiasi, o kai jos pagrindu tampa atvirojo standarto protokolai, susikuria galimybė atsirasti naujam fenomenui.

Esant aprašytam ir viešai prieinamam sąveikos taisyklių rinkiniui (protokolui), bet kas gali sukurti sistemą, kuri gali sąveikauti su kito sistema, veikiančia pagal tą patį protokolą. Tada informacinės sistemos kūrėjas neturi derintis prie kiekvienos anksčiau sukurtos sistemos, jam užtenka pasirinkti paplitusį protokolą, kurį naudoja kitos sistemos. Dėl šios priežasties interneto plėtrai naudojamos skirtingų gamintojų sistemos, kurios anksčiau nebuvo suderinamos tarpusavyje.

Interneto standartus kuria nemažai organizacijų, bet aktyviausiai tuo užsiima *IETF (The Internet Engineering Task Force)*³³. Tai didelė atvira tarptautinė bendruomenė, kurios nariai yra tinklų kūrėjai, operatoriai, įrangos gamintojai ir tyrėjai, kuriems rūpi sklandus tinklų veikimas bei interneto evoliucija.

Bendruomenė susibūrė kurti interneto standartus ir yra atvira kiekvienam suinteresuotam asmeniui. *IETF* misija aprašyta *RFC 3935*³⁴. Visi *IETF* standartai yra atviri ir kiekvienam prieinami *IETF RFC* serveryje.

12.2. Kas yra *RFC*

Requests for Comments (RFC) yra techninių ir organizacinių pastabų apie internetą rinkinys. Jos pradėtos kurti 1969 m. ir tada buvo skirtos *Arpanet* tinklui. *RFC* kalbama apie įvairius tinklų aspektus – protokolus, procedūras, programas, koncepcijas. Rečiau *RFC* pateikiamos susirinkimų pastabos, nuomonės ar netgi humoras. Kai kurie *RFC* yra standartų projektai arba patvirtinti standar-



I E T F®

12.1 pav. *IETF* ženklas

³³ <http://www.ietf.org/overview.html>

³⁴ <http://tools.ietf.org/html/rfc3935>

tai. Kiti *RFC* yra informaciniai, eksperimentiniai, geriausios praktikos aprašai.

12.3. Interneto protokolai

Interneto protokolas – taisyklės, kurios pateikia duomenų mainų tarp dviejų kompiuterinių sistemų internete būdą. Pagrindinis protokolas, užtikrinantis interneto veikimą, yra *TCP/IP*.

Tačiau būtina aprašyti ir daugiau kompiuterių sąveikos tinkle aspektų. Kiekvienam iš jų turi būti aprašomas protokolas. Kompiuteriams susijungiant vieni protokoliai sudaro sąlygas veikti kitiems protokolams. Tokiai sudėtingai sistemai aprašyti sukurtas *OSI* modelis (angl. *Open Systems Interconnection Reference Model*).

12.4. Reikšmingos interneto istorijos datos

- 1957 m. – SSRS paleidžia pirmąjį dirbtinį Žemės palydovą *Sputnik*. Atsakydamas į tai JAV Gynybos departamentas įkuria Perspektyviųjų gynybos tyrimo projektų agentūrą, angl. *Advanced Research Projects Agency (ARPA)*, kuri padėtų JAV pirmauti taikant mokslo ir technologijų sritis karo pramonei.
- 1958 m. – *Bell Telephone* išleidžia pirmąjį modemą, kuris telefono linijomis perduoda skaitmeninius duomenis iki 300 bitų per sekundę.
- 1960 m. – JAV Gynybos departamentas iškelė idėją sujungti kompiuterius į tinklą, kuriam neturėtų įtakos atsitiktinės klaidos ir kuris būtų tinkamas ryšiams, ypač karo atveju (pažeidus bet kurią ryšio liniją, tinklo darbas nenutrūktų). Šį projektą turėjo įgyvendinti *ARPA*.
- 1961 m. – keletas nepriklausomų JAV tyrimo grupių pradėjo tobulinti informacijos persiuntimo paketais technologijas.
- 1962 m. – psichologas ir informatikas *Joseph Carl Robnett Licklider* (1915–1990 m.), dirbantis *ARPA*, pasiūlė įrengti

kompiuterių tinklą, kad būtų vystomas žmonių mąstymas ir vyktų intensyvesnis pasikeitimas informacija tarpusavyje.

- 1966 m. – *Larry Roberts* (g. 1937 m.), dirbdamas *ARPA*, tapo pagrindiniu tinklo *ArpaNet* architektu.
- 1969 m. – *ArpaNet* tinklu, kuris laikomas interneto prototipu, buvo sujungti 4 kompiuteriai.
- 1970 m. – *ARPA* sukūrė interneto protokolą *IP*.
- 1971 m. – *Ray Tomlinson* (g. 1941 m.) iš *BBN* sukuria elektroninio pašto programą žinutėms siūsti paskirstytame tinkle. Ji parašyta naudojantis programų *SNDMSG* (programa žinutėms siūsti vieno kompiuterio terpėje) ir *CPYNET* (bandomoji failų perdavimo programa) patirtimi. *Ray Tomlinson* išsiuntė pirmąjį elektroninį laišką į kitą kompiuterį savo biure (manoma, kad tai buvo tekstas *QWERTYUIOP*). Jis pasirinko ženklą *@* adresui ir gavėjo vardui atskirti. Tuo metu *ArpaNet* tinkle buvo 15 mazgų ir 23 kompiuteriai.
- 1972 m. – sukurta *Telnet* specifikacija (*RFC 318*).
- 1973 m. – sukurta *FTP* specifikacija (*RFC 454*). Pirmieji tarptautiniai prisijungimai prie *ArpaNet* – Londono universiteto koledžas ir Norvegijos Karališkoji radarų bendrija. *Bob Mctcalfe* disertacijoje pateikia *Ethernet* idėją. *Vinton Gray Cerf* (1943 m.) San Francisko viešbutyje antroje voko pusėje greitomis nubraižo tinklų sietuvo (*gateway*) architektūrą.
- 1974 m. – *Vinton Gray Cerf* (g. 1943 m.) ir *Bob Kahn* (g. 1938 m.) išspausdina *A protocol for packet Network inter-communication*, kur pateikia detalią *TCP* (*Transmission Control Program*) aprašą.
- 1976 m. – *AT&T Bell Labs* sukuria failo kopijavimo iš kompiuterio į kompiuterį programą *UUCP* (*Unix to Unix cp*, kur *cp* yra *Unix* sistemos failų kopijavimo komanda). *UUCP* po metų išplatinama su *Unix* sistema. Vėliau *UUCP*

skirta programinė įranga kuriama ir kitoms sistemoms, *UUCP* tampa protokolu (*Unix to Unix Copy Protocol*).

- 1977 m. – sukurta el. pašto specifikacija (*RFC 733*).
- 1979 m. – *Tom Truscott*, *James Tice Ellis* (1956–2001 m.) ir *Steven M. Bellovin* panaudoję *UUCP* įkuria diskusijų grupių sistemą *UseNet*.
- 1982 m. – *DCA* ir *ARPA ArpaNet* tinkle įdiegiamas *TCP* ir *IP* junginys (*TCP/IP*) ir JAV gynybos departamentas paskelbia jį standartu. Sukuriama *EGP* (*External Gateway Protocol*) specifikacija (*RFC 827*).
- 1983 m. – *Paul Mockapetris* sukuria domeno vardų serverį (*DNS*), *RFC 882* ir *RFC 883*. *Tom Jennings* sukuria *FidoNet*.
- 1984 m. – išigali *DNS* sistema. Mazgų skaičius viršija 1 000. Amerikos gynybos departamentas paskelbia *TCP/IP* standartiniu visų karinio pobūdžio tinklų protokolu. *Usenet* tinkle atsiranda moderuojamų diskusijų grupių.
- 1986 m. – *Craig Partridge* sukuria *MX* įrašo formatą – tai leidžia ne *IP* tinklams turėti domenų vardus. Sukurtas *NNTP* protokolas.
- 1987 m. – *Bill Atkinson* pritaikė hipertekstą *Apple* kompiuteriams.
- 1988 m. – interneto tinklu prašliaužia „kirminas“ ir užkrečia 6 000 iš tuo metu veikusių 60 tūkst. mazgų. Sukurta pirmoji elektroninio pašto programa *Eudora*, pirmą kartą pritaikyta Iliojaus universitete.
- 1989 m. – Europos interneto tinklams administruoti įkuriamas *RIPE*. Mazgų kiekis viršija 100 tūkst. *ARPANET* veikla nutrūko, susikūrė dabartinis interneto tinklas.

- 1991 m. – *Philip Zimmerman* išleidžia *PGP (Pretty Good Privacy)*, skirtą kompiuterių tinklais perduodamiems duomenims šifruoti.
- 1992 m. – interneto mazgų kiekis viršijo milijoną. Lietuvai suteikta .lt aukščiausiojo lygio adresų sritis. Lietuvoje pradėjo veiklą pirmieji interneto paslaugų tiekėjai UAB „Elneta“ ir KŪB „Aiva sistema“.
- 1993 m. – į areną įsiveržia naršyklė *Mosaic*. Žiniatinklis (*WWW*) vystosi nematytu tempu.
- 1994 m. – internete pasirodo el. parduotuvės ir pirmasis virtualusis bankas. Arizonoje įsikūrusi advokatų kontora „prišiukšlina“ internetą „Žalios kortos“ reklama (*spam*). internautai atsako keiksmiais (*flame*).
- 1995 m. – *http* tapo pirmaujančiu tinklo protokolu pagal perduodamą duomenų srautą. Paskelbta *Apache 1.0* versija. Įsisteigė *Netscape* kompanija. Domenų vardai tapo mokami.
- 1996 m. – internetinė telefonija patraukė JAV ryšio bendrovių dėmesį ir jos paprašė uždrausti šią technologiją. Priimtas JAV Ryšių padomumo įstatymas (*Communications Decency Act*), sukėlęs internautų protestą ir vėliau atšauktas. Vasario 23 d. buvo išleistas pirmojo lietuviško interneto žurnalo „Vartiklis“ pirmasis numeris.
- 1998 m. – du Stenfordo studentai įkūrė paieškos sistemą *Google*.
- 2003 m. – pirmas lietuviškas straipsnis Vikipedijoje, lietuviškos laisvosios enciklopedijos kūrimo pradžia. Sukurta pelno nesiekianti organizacija *Mozilla Foundation*.
- 2007 m. – mobiliojo ryšio bendrovė „Omnitel“ save ir Lietuvą pristato beveik 10 mln. vartotojų turinčiame virtualiame pasaulyje *Second Life*.

- 2008 m. – Ministro Pirmininko ir ministrų atsakymai į žurnalistų klausimus bei LR Vyriausybės spaudos konferencijos skelbiamos vaizdo įrašų publikavimo sistemoje *Youtube*.

12.5. Klausimai ir uždaviniai

1. Kas yra internetas?
2. Kas yra *RFC*?
3. Kodėl interneto plėtrai svarbus standartų atvirumas?

13. Duomenų aprašymo kalbos *HTML* ir *XHTML*

HTML yra puikus pavyzdys, kaip laisvi atvirieji standartai gali pakeisti pasaulį. Šiame skyriuje apžvelgsime *HTML* istoriją, *HTML* kalbos elementus, *HTML* kalbos ir žiniatinklio tendencijas. Pamatysime, kaip kyla nesutarimai tarp laisvųjų standartų kūrėjų ir kaip tai gali veikti standarto vystymąsi ir naujų standartų atsiradimą.

13.1. Dokumentų formato ir perdavimo standartai

Žiniatinklis – kitaip pasaulinis voratinklis – angl. *World Wide Web (WWW)* – tarpusavyje susietų *HTML* dokumentų ir juos papildančių kitokių failų, prieinamų internete *http* protokolu, visuma.

HTML – angl. *Hypertext Markup Language* – teksto su nuorodomis žymių kalba. Jos paskirtis – aprašyti dokumento struktūrą.

HTTP – angl. *Hyper Text Transfer Protocol* – hiperteksto persiuntimo protokolas – taisyklių ir susitarimų rinkinys, skirtas hipertekstinių dokumentų persiuntimo būdui apibrėžti, kai informacija siunčiama iš tarnybinės stoties į vartotojo kompiuterį. Kai vartotojas savo interneto naršyklėje nurodo adresą ar paspaudžia interneto puslapyje nuorodą, jo kompiuteris susijungia su tinklo tarnybine stotimi ir siunčia į ją pranešimą. Tada iš tarnybinės stoties siunčiami failai, reikalingi interneto svetainės puslapiui vaizduoti.

13.2. Žiniatinklio atsiradimas

Žiniatinklis (*World Wide Web*) yra prieinamų ir tarpusavyje susietų *HTML* kalba aprašytų dokumentų visuma. Tam, kad taptų prieinami, jie turi būti pateikti tam skirtuose serveriuose. Be to, jie gali būti papildomi kitų formatų dokumentais ar grafinėmis iliustracijomis. Yra kelios genialios idėjos, siejamos su žiniatinklio gimimu.

Perduodant dokumentą iš vieno kompiuterio kitam nebuvo garantijos, kad gavėjas galės juo tinkamai pasinaudoti. Sudėtingesnio dokumento perdavimas nebuvo įmanomas tuo atveju, jei siun-

tėjas ir gavėjas nesinaudojo ta pačia programine įranga. Net ir naudojantis identiška programine įranga perduodant dokumentą galėjo kilti problemų dėl skirtingos vaizduoklių raiškos ar kitų skirtumų. Akivaizdžiai reikėjo rasti būdą, kaip perduoti dokumento esmę tarp skirtingų sistemų. Užtikrinti technologinį neutralumą, tai yra, kad dokumentu būtų galima pasinaudoti nepriklausomai nuo naudojamo kompiuterio ar operacinės sistemos. Būtent dokumento turinio, o ne jo išvaizdos perdavimas buvo pradinis tikslas.

Labai svarbus *HTML* komponentas yra nuorodos. Ne veltui jos paminėtos netgi kalbos pavadinime (*hypertext*). Buvo svarbu sukurti bendrą adresų sintaksę, kuri leistų nurodyti tikslią vietą konkrečiame dokumente.

Taip atrodo bendra nuorodos struktūra (adresas rašomas be tarpų):

protokolas : serveris / kelias / failas ? parametrai # vieta
--

Pavyzdžiui:

http://www.example.net/darbo/doku.php?id=12#5_sk

Svarbu tai, kad bet kuris elementas gali būti praleistas. Nesant nurodyto protokolo naršyklė pasirenka įprastą protokolą *http*. Nesant nurodyto serverio naršyklė ieškos dokumento tame pačiame serveryje, kuriame buvo nuoroda. Taip gaunamos santykinės nuorodos. Kelias nenurodomas, jei norima atversti serverio šakniniame aplanke esantį dokumentą. Jei nenurodytas nei serveris, nei kelias, tada failo ieškoma tame pačiame aplanke, iš kurio gauta nuoroda. Nenurodžius failo serveris pateikia nustatytą standartinį aplanko failą. Parametrų gali ir nebūti, o nenurodžius konkrečios vietos dokumentas rodomas nuo pradžios.

Nuorodose į kitus resursus tame pačiame serveryje visada įmanomas absoliučių ir santykinųjų nuorodų pasirinkimas.

Pavyzdžiui, absoliutus adresas:

```
http://www.example.net/abc/def.html#antras
```

Tas pats resursas rodant iš kito aplanko tame serveryje:

```
/abc/def.html#antras
```

Tas pats resursas rodant iš kito failo tame pačiame aplanke:

```
def.html#antras
```

Vieta dokumente rodant iš to paties dokumento:

```
#antras
```

13.3. *HTML* elemento konstrukcija

HTML kalba sukurtas dokumentas – tai tekstas su specialiais žymenimis tarp ženklų „<“ ir „>“. Taip tekstas padalinamas į tokio formato elementus:

```
<komanda> turinys </komanda>
```

Turinys gali būti tekstas bei kiti elementai, komanda gali turėti parametrus:

```
<komanda param="reikšmė"> turinys </komanda>
```

13.4. Paprasčiausias *HTML* dokumentas

Visas *HTML* kodas pateikiamas elemente <html>, kuris susideda iš dviejų dalių. Pradžioje techninė informacija pateikiama elemente <head>, paskui dokumento turinys pateikiamas elemente <body>.

```
<html>
  <head>
    <title>Pirmasis HTML pavyzdys</title>
  </head>
  <body>
    <h1>Pirmasis HTML dokumento pavyzdys</h1>
    <p>Pirmojo dokumento pirmoji pastraipa.</p>
    <p>Pirmojo dokumento antroji pastraipa.</p>
  </body>
</html>
```

Be anksčiau paminėtų elementų, čia panaudoti dar trys:

`<title>` – taip antraštėje nurodomas dokumento pavadinimas, rodomas lango antraštėje ir naudojamas automatiškai kuriant nuorodą į dokumentą.

`<p>` – pastraipos elementas.

`<h1>` – skirtas kurti aukščiausio lygio pavadinimams dokumento tekste (*heading 1*). Smulkesnių skyrių pavadinimai žymimi `<h2>`, jų poskyriai – `<h3>` ir taip toliau.

Šis *HTML* kodas atitinka visų *HTML* versijų ir mokyklų reikalavimus. Prieš aptariant kitas komandas būtina paminėti versijas ir mūsų dėl *HTML* kodo vystymo krypčių.

13.5. Vaizdas ar turinys

Nuo pradžių *HTML* kalba sukurta kaip priemonė turiniui perteikti nepriklausomai nuo naudojamos įrangos. Autorius sužymi dokumento struktūrą, o naršyklė ją atvaizduoja taip, kad dokumento skaitytojui būtų kuo patogiau. Autorius neturėtų rūpintis išvaizda, nes jis nežino, kokios yra dokumentą atverčiančios naršyklės galimybės.

Tačiau *HTML* dokumentus ėmus kurti plačiau visuomenei paaiškėjo, kad daugelis autorių labiau rūpinasi vaizdu negu turiniu. Naršyklių kūrėjai siekė papildyti *HTML* kalbą kuo daugiau išvaizdą aprašančių elementų. Autoriai pamėgo klaidingai naudoti

turinio elementus dokumentui pagražinti. Pavyzdžiui, `<h1>` elemente pateikiamas visai ne pavadinimas, bet sakinio dalis, kurią norima pabrėžti, išskirti. *HTML* kūrėjai pritarė „švariam“ be dizaino priemaišų *HTML*, o naujai atsiradusi ir vis stiprėjanti nekvalifikuotų naudotojų grupė reikalavo galimybių gražinti dokumentą, kištis į jo atvaizdavimą.

Kalba papildyta ne tik išvaizdai skirtais elementais `` (*bold* – pusjuodis šriftas), `<i>` (*italic* – kursyvas – pasvirasis šriftas), bet ir kitiems elementams suteikiami išvaizdai nurodyti skirti parametrai, pavyzdžiui, `<body background=“red“>`.

13.6. Ar toleruotinos *HTML* sintaksės klaidos

Nevienareikšmiška buvo ir bendruomenės pozicija dėl *HTML* taisyklių griežtumo. Kompiuteriams tampant vis galingesniems naršyklės lengvai ištaiso *HTML* netikslumus.

Pavyzdžiui, toks kodas, nors ir yra netikslus, buvo pripažintas teisingu:

```
<p>Miške kur eglės ošia,<p>Po pušim sena
```

Nors nepažymėta pirmosios pastraipos pabaiga, bet akivaizdu, kad jei prasidėjo antroji pastraipa, tai pirmoji pasibaigė.

Naršyklėms nebuvo sudėtinga priimti dokumentą nepriklausomai nuo to, ar didžiosios, ar mažosios raidės naudojamos *HTML* komandoms. Todėl komandas kiekvienas rašė kaip kam patinka. Paplito trys užrašymo sistemos: `<body>`, `<BODY>` ir `<Body>`. Sintaksės reikalavimai leido rašyti ir `<BoDy>` ar dar 12 šios komandos variantų.

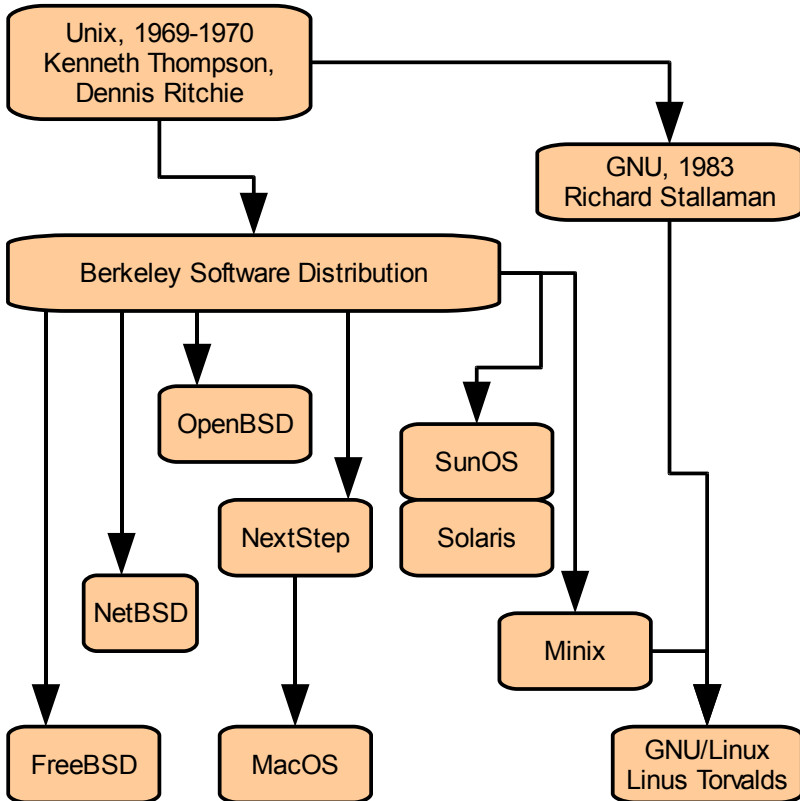
Dėl sintaksės tikslumo nors ir nebuvo tokių didelių ginčų, kaip esmės ir formos kovoje, bet vieni autoriai siūlė praleisti nuspėjamus *HTML* komponentus, taip sutrumpinti dokumento apimtį, sutaupyti vietą serveryje bei krovimosi laiką, kiti siūlė laikytis griežtos *HTML* sintaksės.

13.7. Klausimai ir uždaviniai

1. Kokia yra *HTML* paskirtis?
2. Sukurti dokumentą tvarkinga *XHTML* kalba, susidedantį bent iš trijų skyrių. Automatiškai patikrinti *HTML* kokybę. Pavyzdžiui: „Biografija“, „Hobis“, „Gimtojo miesto rajonai“.
3. Rasti internete nedidelės apimties netvarkingą *HTML* dokumentą ir iš jo padaryti tvarkingą dokumentą *XHTML* kalba.

14. GNU operacinės sistemos komponentas – *Linux*

Čia trumpai pristatysime *GNU/Linux* istoriją, kodėl kai kas rašo *GNU/Linux*, kas yra *Linux* distributyvas.



14.1 pav. *GNU/Linux* tarp kitų *Unix* tipo operacinių sistemų³⁵

³⁵ http://commons.wikimedia.org/wiki/Image:Unix_history.en.svg

14.1. Operacinės sistemos branduolio *Linux* atsiradimas

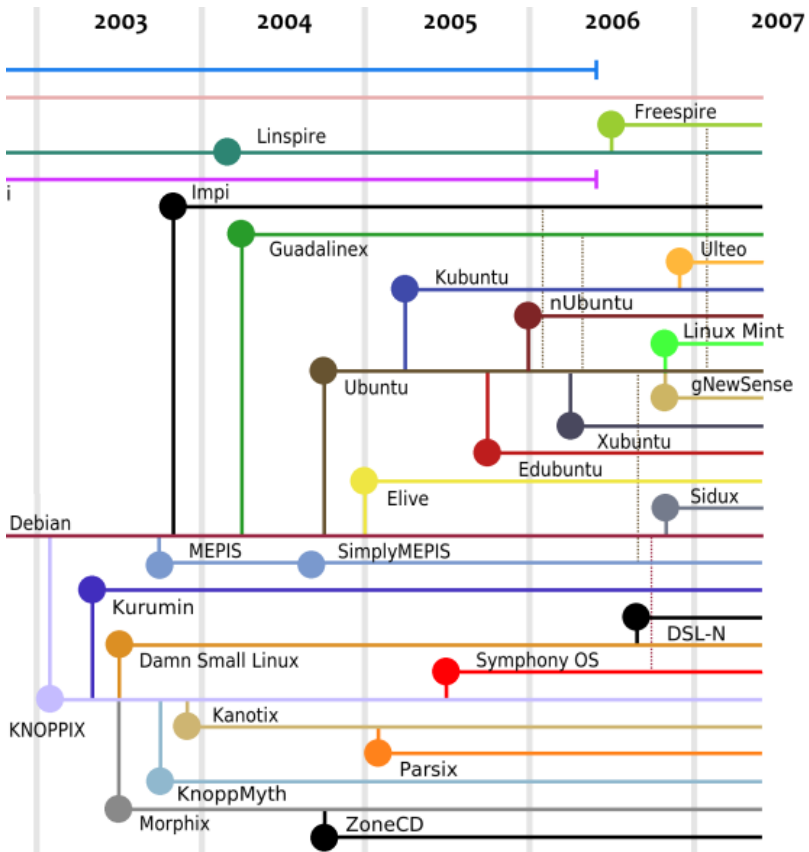
1991 m. Suomijos Helsinkio universiteto studentas Linus Torvalds parašė džiugų laišką į *comp.os.minix* naujienų grupę ir pranešė apie tai, kad jo kuriama operacinė sistema jau „beveik veikia“. Ši data ir šis laiškas oficialiai laikomas *Linux* gimtadieniu.

1992 m. *Linux* kūrėjai paskelbė, kad *Linux* branduolį platins *GNU GPL* licencijos sąlygomis. *Linux* integruojamas į *GNU* projektą. *Linux* pritraukė daugybės programuotojų iš viso pasaulio dėmesį ir buvo toliau tobulinamas.

1994 m. internete pradėta platinti *Linux* branduolio 1.0 versija. Į jo tobulinimą įsitraukė pasaulinė programuotojų bendruomenė ir greitai *GNU/Linux* operacinė sistema tapo žinoma kaip viena stabiliausių ir galingiausių profesionalių operacinių sistemų. Ja naudotis tada reikėjo profesionalių žinių ir įgūdžių. *GNU/Linux* programuotojai kūrė programuotojams.

Šiandien *Linux* vadinamas ne branduolys, o distributyvas, kurių sudaro *Linux* branduolys, laisvoji programinė įranga iš *GNU* projekto bei kitos laisvosios ar atvirojo kodo programos ir pagalbiniai įrankiai (14.2 pav.). Operacinę sistemą tiksliau būtų vadinti *GNU/Linux*.

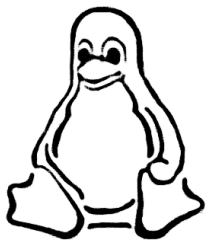
Lygiagrečiai su atvirais *Linux* distributyvais kuriami ir komerciniai *Linux* distributyvai, kurių kūrėjai uždirba iš pardavimo ir aptarnavimo. *GPL* licencija neleidžia uždaryti programos, tai yra uždrausti kopijuoti patobulintą jos versiją, tačiau atsirado nemažai įmonių, kurios rado būdų, kaip nepažeidžiant licencijos uždirbti iš *GPL* produktų. Dažniausiai tai pajamos, gautos iš produktų aptarnavimo. Tokiu atveju įmonė teikia profesionalią paramą ne tokiems profesionaliems kliento darbuotojams. Tai yra vienas iš ekonominių stimulų *GNU/Linux* produktus pritaikyti ir kompiuterių naudotojams: neprofesionaliems programuotojams ar visai ne programuotojams.

14.2 pav. GNU/Linux distributyvų medžio fragmentas³⁶

Vis daugiau GNU/Linux sistemų kūrimo darbų finansuojama filantropų arba kompiuterių įrangos gamintojų, kurie siekia kuo geriau parduoti savo įrangą. Tokių įmonių tikslas yra sukurti sistemą ir programų rinkinį (distributyvą), skirtą įvairiems varto-

³⁶ http://commons.wikimedia.org/wiki/Category:Linux_distro_timeline

tojams. Kuriama vis daugiau patogių *GNU/Linux* administravimo sistemų, jų naudojimas tampa vis paprastesnis.



14.3 pav. *Linux* logotipas – pingvinas vardu *Tux*

Apie 2002 m. nuosavybinių operacinių sistemų distributyvų ir atvirųjų operacinių sistemų distributyvų naudojimo sudėtingumas tapo palyginamas.

Apie 2004 m. neprofesionaliai naudotojo darbo vietai skirti *Linux* distributyvai (14.3 pav.) ėmė konkuruoti su rinkoje vis dar dominuojančia *MS Windows* operacine sistema.

14.2. Informacinės sistemos programinė įranga

Žiniatinklio serveriai dažnai veikia atvirosios programinės įrangos pagrindu: *GNU/Linux* operacinės sistemos branduolys, *GNU* sisteminių programų rinkinys, *Apache* serveris, duomenų bazių valdymo sistema (DBVS) – dažniausiai *MySQL*, ir specializuota programinė įranga, kuri dažniausiai yra kuriamos *PHP* kalba.

Tipinis serverio programinės įrangos komplektas vadinamas *LAMP* – tai laisvosios programinės įrangos *Linux*, *Apache*, *MySQL* ir *PHP* derinys.

Akronimą *LAMP* savo straipsnyje vokiškame žurnale „c’t“ 1998 m. (12 nr., 230 p.) pavartojo *Michael Kunze*. Straipsnyje rašė apie tai, kaip laisvų programų rinkinys gali būti gyvybinga ir gera alternatyva komerciniams paketams. Dėl kompiuterių vartotojų bendruomenių pomėgio akronimams *LAMP* greitai tapo terminu, populiariančiu atvirųjų programų naudojimą.

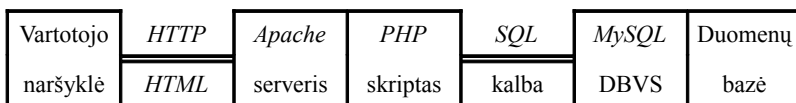
14.3. Programų rinkinio *LAMP* komponentai

GNU/Linux yra *Unix* tipo operacinės sistema su *GNU* sisteminių programų paketu. Kartas *Linux* pakeičiamas kita *Unix* tipo operacine sistema.

Apache yra žiniatinklio (*web*) serveris. Jis yra stabilus, lengvai įdiegiamas, turi standartinius nustatymus bei suteikia sudėtingo derinimo galimybių.

MySQL – tai paplitusi laisvoji *DBVS*. Anksčiau ji buvo naudojama tik smulkesniems projektams, o didesniems komerciniams projektams dažniau buvo naudojamos nuosavybinės *DBVS*. Tačiau *MySQL* pasirodė tinkama ir dideliems atsakingiems projektams, todėl pastaruoju metu ji vadinama universalia *DBVS*.

PHP yra programavimo kalba. *Apache PHP* modulis įvykdo *PHP* kalba parašytą skriptą ir vartotojui pateikia *HTML* dokumentą. Laisvai platinama daugybė programų *PHP* kalba, todėl informacinių sistemų kūrėjai paprastai nekuria programų nuo pirmos eilutės, bet ieško jau sukurtų panašios paskirties programų ir patobulina jas bei pritaiko savo reikmėms (14.4 pav.).



14.4 pav. Informacinės sistemos veikimo schema

Visi išvardinti *LAMP* komponentai susideda iš laisvai platinamų programų, todėl jie gali būti modifikuojami. Tam, kad geriausiai tiktų informacinei sistemai, sujungiami į vieną programų rinkinį ir kartu platinami. Į daugelį universalių *Linux* distributyvų arba įeina visas *LAMP* programų kompletas, arba nesunkiai instaliuojami trūkstami komponentai. Kuriamos ir tokios informacinės sistemos, kuriose vienas ar keli *LAMP* komponentai pakeičiami jų alternatyvomis.

14.4. Klausimai ir uždaviniai

1. Kas yra *Linux* ir kas yra *GNU/Linux*?
2. Internete raskite *Linux* distributyvų sąrašą. Aprašykite tris pasirinktus distributyvus. Kuo jie skiriasi?
3. Kokie pagrindiniai komponentai sudaro informacinę sistemą?
4. Kas yra *LAMP*?

15. Laisvas ir populiarus serveris *Apache*

Atvirasis žiniatinklio serveris *Apache* užėmė daugiau nei pusę rinkos, kai apie atvirąją programinę įrangą žinojo tik specialistai. Šiame skyriuje aptarsime, kas yra *Apache* ir kas nulėmė tai, kad *Apache* aplenkė daugelį kitų šiuo metu populiarių laisvųjų programų?

15.1. Žiniatinklio serveris *Apache*

Apache serveris yra reikšmingas žiniatinklio plėtrai. *Apache* buvo viena pirmųjų alternatyvų *Netscape Communications Corporation* tinklo serveriui, dabar žinomam *Sun Java System Web Server* vardu. Savo funkcionalumu ir našumu *Apache* konkuruoja su kitais *Unix* operacinės sistemos pagrindu kuriamais tinklo serveriais.

Projekto vardas pasirinktas iš pagarbos Amerikos indėnams apačiams³⁷, kurie yra gerai žinomi dėl savo ištvermingumo ir kovos įgūdžių. Projekto pavadinimą nulėmė ir tai, kad jis buvo kurtas iš *NCSA HTTPd 1.3* pataisų serverio, kurį dar vadino „*a patchy*“.

Į *Apache* tinklo serverį įtraukti moduliai, kurie išplečia serverio funkcionalumą – serverių programavimo kalbos (*mod_perl*, *mod_python*, *Tcl*, ir *PHP*), moduliai (*mod_access*, *mod_auth* ir *mod_digest*), *SSL* ir *TLS* (*mod_ssl*), tarpinio serverio modulio aptarnavimas, nuorodų (*URL*) redaktorius, duomenų filtravimo galimybės. *Apache* pranešimų failai gali būti analizuojami naudojant tinklo naršyklę ir kitas laisvai platinamas programas.

Virtual hosting funkcija leidžia vienai *Apache* programai aptarnauti kelias internetines svetaines. Pavyzdžiui, vienas kompiuteris su viena *Apache* programa gali vienu metu aptarnauti kelias svetaines.

Apache tinklo serveris naudojamas statinio ir dinaminio turinio svetainėms žiniatinklyje publikuoti. Dauguma internetinių

³⁷ <http://lt.wikipedia.org/wiki/Apačiai>

programų yra sukurtos *Apache* serverio pagrindu. *Apache* naudojamas užduotims atlikti, kai turinys turi būti pasiekiamas saugiu ir patikimu būdu, pavyzdžiui, dalintis failais per internetą.

Apache tinklo serveris yra įtrauktas į populiarių *LAMP* tinklo serverio programų rinkinį.

15.2. Kas remia *Apache* projektus

Apache HTTP serveris gali būti naudojamas pagal *Apache* licenciją ir tai yra laisvoji atvirojo kodo programinė įranga. *Apache* licencija leidžia naudoti programinę įrangą tiek atvirojo, tiek uždarojo kodo programoms.

Apache serverį kuria atvira kūrėjų bendrija, kurią palaiko *Apache Software Foundation* fondas³⁸ (15.1 pav.). Didžiausiems šio fondo rėmėjams suteikiamas platininio rėmėjo statusas³⁹. Suprantama, kad platininėmis rėmėjomis tapo didžiosios bendrovės, uždirbančios iš interneto turinio: *Google* ir *Yahoo*. Jau kuris laikas *Apache* serveriai užima apie pusę žiniatinklio serverių rinkos, o pagrindiniai jų konkurentai – analogiškai *Microsoft* produktai – užima apie trečdalį rinkos⁴⁰. Todėl daug kas nustebė, kai 2008 m. ši korporacija ne tik tapo platinine rėmėja, pinigais parėmusi fondą, bet ir palaiko susijusios programinės įrangos vystymą. Bendra tendencija: buvusios dvigubo licencijavimo programos tampa atvirosiomis, nuosavybines programas kuriančios bendrovės įsitraukia į atvirųjų programų kūrimą. Tai džiugina daugelį vartotojų.

Bruce Perens, atvirojo kodo apibrėžimo autorius, perspėja, kad *Microsoft* gali turėti ir kitų ketinimų, kurie nebūtų priimtini bendruomenei: „Praėjo tik keleri metai nuo to laiko, kai vienas iš *Microsoft* vadovų *Jim Allchin* pavadino atvirąjį kodą intelektualios nuosavybės naikintoju, *Craig Mundie* pavadino ekonomiškai liguistu. Bet tai buvo senasis *Microsoft*⁴¹.“ *Bruce Perens* toliau dėsto, kad *Microsoft*

³⁸ <http://www.apache.org/>

³⁹ <http://www.apache.org/foundation/thanks.html>

⁴⁰ <http://www.elektronika.lt/news/computers/12533/>

⁴¹ <http://itmanagement.earthweb.com/osrc/print.php/3762786>

galbūt ketina pasinaudoti *Apache* licencijos trūkumais. Jie galėtų imti leisti uždarojo kodo *Apache* produktų versijas. Tokie produktai galėtų nukrypti nuo standartų arba būti suvaržyti patentinių apribojimų.

Šie numanomi tikslai nepaaiškina, kodėl *Microsoft* turėtų paremti *Apache*. Licencija leidžia taip naudotis *Apache* produktais ir neremiant fondo. Jei reikalingi konkretūs darbai – tada galima asmeniškai samdytis *Apache* projektų programuotojus. *Bruce Perens* nuomone, pinigai skirti reklamos tikslais, norint parodyti naują bendrovės santykį su atvirojo kodu. Straipsnis užbaigiamas perspėjimu, kad pasaulio teisinės sistemos turi pabusti, kad pasaulis taptų saugus ir atvirojo kodo, ir nuosavybinėms programoms kurti. Atvirieji standartai turi būti visuotinai pripažįstamas IT politikos pagrindas. Tik tada bus realizuotas atvirojo kodo potencialas.

15.3. Kodėl *Apache* yra laisvas

Kuriant *Apache* programinę įrangą siekiama sukurti tvirtą pagrindą įvairių tipų programinei įrangai taikyti. Kuriamą platforma patikimoms sistemoms įvairiems uždaviniams spręsti.

Apache kūrėjai mano, kad internetinio turinio viešinimo įrankiai turi būti visiems laisvai pasiekiami ir kad programinės įrangos bendrovės turi uždirbti kurdamos pridėtinės vertės paslaugas.

Suprantama, kad vienai kompanijai kontroliuoti rinką yra ekonominis pranašumas, ypač programinės įrangos pramonėje, tai lyg kontroliuoti tam tikrą vamzdyną taip, kad visi kiti būtų priversti mokėti už jo naudojimą. Taip daroma, valdant protokolus, verslas plečiamas kitų kompanijų lėšomis. Žiniatinklio protokolai, būdami atviri ir nepriklausomi, sukuria vienodas sąlygas dideliems ir mažiems rinkos dalyviams.



15.1 pav. *Apache* logotipas

Kita nurodyta priežastis – tai noras įtraukti į programinės įrangos kūrimą jos vartotojus. Tas, kas užmokėjo už programinę įrangą, paprastai nenorės taisyti jos klaidų veltui.

15.4. Skirtumas tarp vairuotojo ir keleivio

Apache turi dvi vartotojų grupes. Jei *Apache* būtų troleibusas, tuomet vieni iš jų būtų keleiviai, kiti – vairuotojai. Čia aptarsime, kaip toks vartotojų susisluoksniavimas nulėmė *Apache* sėkmę dar iki laisvųjų programų bumo pradžios.

Pirmiausia apie „keleivius“. Milijardai dokumentų, atverčiamų vartotojų naršyklėse, yra pateikiami *Apache* programinės įrangos. Tokie vartotojai nedaro įtakos serveryje veikiančios programinės įrangos pasirinkimui. Dauguma iš jų nesidomi, kokia programinė įranga jiems tiekia dokumentus. Kaip ir troleibuso keleivis paprastai nesidomi, kokios firmos troleibusu jis važiuoja ar kaip valdomas troleibuso variklis. Tokie vartotojai sukuria didžiulę rinką. Didžiuliai resursai skiriami kuriant jiems skirtus produktus.

Kita vartotojų dalis – „vairuotojai“. Kiekvieną įdiegtą serverį tvarko bent vienas specialistas – jo administratorius. Kaip ir profesionalus transporto priemonės vairuotojas, jis išmano savo valdomą techniką, gilinasi į jos veikimo subtilybes. *Apache* augimui didelę įtaką turi dvi tokios situacijos pasekmės. Specialistas yra pajėgus pasirinkti optimalų techninį sprendimą. Jis remiasi ne reklama, o techniniais kriterijais. Tai mažina nuosavybinių konkurentų investicijų į rinkodarą efektyvumą, daug sunkiau išstumti gerą produktą iš informuotų specialistų rinkos. Antroji pasekmė – sistemą eksploatuoja specialistai, siekiantys ir sugebantys ją tobulinti, kvalifikuotai padedantys kūrėjams.

15.5. *Apache* licencijos tekstas

Apache License. Version 2.0, January 2004. [http://www.apache.org/licenses/TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION](http://www.apache.org/licenses/TERMS_AND_CONDITIONS_FOR_USE,_REPRODUCTION,_AND_DISTRIBUTION)

I. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is

conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. *Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.*

3. Grant of Patent License. *Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.*

4. Redistribution. *You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:*

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative

Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. *Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.*

6. Trademarks. *This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.*

7. Disclaimer of Warranty. *Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.*

8. Limitation of Liability. *In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.*

9. Accepting Warranty or Additional Liability. *While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, accep-*

tance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor; and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

15.6. Literatūra

1. *Apache HTTP Server Project.*
<http://httpd.apache.org/>
2. *Apache HTTP serveris.* Straipsnis Wikipedijoje.
http://lt.wikipedia.org/wiki/Apache_HTTP_serveris
3. *Apache License, Version 2.0.*
<http://www.apache.org/licenses/LICENSE-2.0>
4. *Frequently Asked Questions [Apache].*
<http://www.apache.org/foundation/faq.html>

15.7. Klausimai ir uždaviniai

1. Kas yra *Apache*?
2. Apibūdinkite aptartas dvi *Apache* vartotojų grupes ir nurodykite skirtumą tarp jų.
3. Nubraižykite priežastinių ryšių diagramą, iliustruojančią *Apache* sėkmės priežastis.

16. Duomenų bazių valdymo sistema *MySQL*

Kiekviena informacinė sistema turi išsaugoti kažkokius duomenis. Kiekvienas patyręs programuotojas nesunkiai sukurs bet kokių duomenų įrašymo failuose sistemą. Tačiau tokią sistemą optimizuojant didesniems duomenų kiekiams ir didesniems srautams tektų sugaišti nemažai laiko.



16.1 pav. *MySQL* logotipas

Nėra prasmės išradinėti dviratį. Geriau duomenų valdymą patikėti profesionalų sukurtai ir milijonų vartotojų išbandytai duomenų bazių valdymo sistemai. *LAMP* komplekte paprastai siūloma naudotis *MySQL* (16.1 pav.).

MySQL – tai viena iš reliacinių duomenų bazių valdymo sistemų (liet. RDBVS, angl. *RDBMS*), palaikanti daugelį naudotojų, valdoma *SQL* kalba.

16.1. Akcinė bendrovė *MySQL*

MySQL yra atvirojo kodo programinė įranga (*GPL* ir kitos licencijos), vystoma ir palaikoma švedų kompanijos „*MySQL AB*“, kurios įkūrėjai – švedai *David Axmark*, *Allan Larsson* ir suomis *Michael „Monty“ Widenius*.

2008-02-26 akcinę bendrovę *MySQL* įsigijo korporacija *Sun Microsystems*. Žiniasklaidos pranešimuose buvo juokaujama, kad *Sun* įsigijo žodžio *LAMP* raidę „M“.

Sun Microsystems anksčiau jau buvo įsigijusi *StarOffice*, pagal kurį sukūrė ir tobulina laisvąjį *OpenOffice.org*. *Sun Microsystems* remia ir kitus laisvuosius atvirojo kodo produktus, todėl *MySQL* įsigijimas bendruomenės nebuvo priimtas priešišškai, nors kai kam kelia nerimą didelė laisvųjų atvirojo kodo sprendimų koncentracija vienoje rankose.

16.2. *MySQL* galimybės

Kaip ir kiekvienos RDBVS, *MySQL* duomenys prieinami per abstrakčias lenteles ir ryšius tarp skirtingų lentelių ar jų dalių. Duomenims įvesti, keisti, ieškoti bei lentelėms ir duomenų bazei valdyti yra naudojama *SQL* (angl. *Structured Query Language*) kalba.

MySQL RDBVS veikia daugelyje platformų, ji dažnai pasirenkama programuojant internetines svetaines. Šiame sektoriuje su *MySQL* bando konkuruoti *PostgreSQL*.

Pastaruoju metu *MySQL* vis dažniau pritaikoma labai didelėse informacinėse sistemose. Pavyzdžiui, Vikipedija, kurios apkrova kartais viršija 10 tūkstančių užklausų per sekundę, arba vienas iš didžiausių JAV kabelinės televizijos tinklų *Cox Communications*, kurio duomenų bazėje – daugiau kaip 3 600 lentelių. Šiame sektoriuje pagrindinis *MySQL* konkurentas yra *Oracle*.

Nors prieigai prie *MySQL* duomenų bazių dažniausiai pasirenkama *PHP* kalba, ją taip pat galima pasiekti įvairiomis kitomis programinėmis priemonėmis: *C*, *C++*, *C#*, *Java*, *Perl*, *Python* ir kitomis. Kiekvienai šių kalbų sukurtos specialios bibliotekos. Taip pat *MySQL* duomenų bazėms yra sukurta *ODBC* sąsaja *MyODBC*, leidžianti duomenis pasiekti bet kuria kalba, neturinčia specialios bibliotekos, tačiau palaikančia *ODBC* komunikavimo mechanizmą.

16.3. *MySQL* valdymo įrankis *phpMyAdmin*

PHP kalba parašytas *MySQL* valdymo įrankis *phpMyAdmin*^{42,43}, skirtas administruoti *MySQL* duomenų bazes naudojantis naršykle nežinant *SQL* komandų bei tiesiogiai nesijungiant prie serverio.

1998 m. *MySQL* sąsają *PHP* kalba pradėjo kurti *Tobias Ratschiller*, IT konsultantas ir kompanijos *Maguma* įkūrėjas. Nuo 2000 m. jis nebedirbo prie projekto, bet *phpMyAdmin* jau buvo

⁴² <http://www.phpmyadmin.net/>

⁴³ <http://lt.wikipedia.org/wiki/PhpMyAdmin>

tapęs populiariu *MySQL* administravimo įrankiu su didele vartotojų ir tobulintojų bendruomene.

Pastaruoju metu programa išversta į daugiau nei 50 kalbų, tarp jų ir į lietuvių. *phpMyAdmin* yra įprastas *LAMP* komplekto elementas. Kalbant apie *LAMP* sudėtį *phpMyAdmin* dažnai nepaminimas, nes jis turimas omenyje kaip *MySQL* komponentas, nors ir yra atskiras kūrinys.

16.4. *AB MySQL* licencijos ir verslo modelis

Prieš keletą metų *MySQL* buvo platinama pagal dvi licencijas. Vienos sąlygos – asmeniniam naudojimui, kitos – įmonėms.

Vėliau *AB MySQL* paskelbė, kad visiems vartotojams jų pagrindinė licencija yra *GNU GPL* ir kad jie norėtų paskatinti kiekvieną skelbti savo programinę įrangą su šia licencija⁴⁴:

Mūsų programinė įranga yra 100 % *GPL*. Jei ir jūsiškė yra 100 % suderinama su *GPL*, tada neturite mokėti už *MySQL* naudojimą savo programose. Tai yra puiki galimybė atvirojo kodo bendruomenei ir tiems iš jūsų, kas kuria atvirojo kodo programas.

Taip pat *AB MySQL* siūlo naudotojui pasirinkti priimtinas sąlygas iš kitų licencijų sąrašo⁴⁵.

Norintys panaudoti *MySQL* programinę įrangą savo nuosavybiniuose produktuose gali pasirinkti tinkamą komercinę licenciją iš „*OEM Embedded Database Products*“ variantų sąrašo. Tai modernus verslo modelis, kai mokama už licencijos suteikiamas teises, bet ne už techninius sprendimus. Perkantiems ir nemokamai naudojantiems prieinami tie patys programos komponentai.

Ne visiems atvirojo kodo šalininkams tinkama *GNU GPL* licencija. Atvirojo kodo iniciatyvos (*OSI*) licencija yra artima *GNU GPL*, tačiau yra skirtumų, neleidžiančių *GNU GPL* išvestinio kūri-

⁴⁴ <http://www.mysql.com/about/legal/licensing/opensource-license.html>

⁴⁵ <http://www.mysql.com/about/legal/licensing/>

nio platinti su *OSI* licencijomis. *MySQL* nėra platinama su *OSI* licencija, tačiau yra parengta speciali *GNU GPL* su išimtimis, kurios leidžia išvestinius kūrinius platinti su *OSI* licencija.

Su brangiais nuosavybiniais konkurentais galėtų būti lyginami *MySQL Enterprise* prenumeratos variantai. Priklausomai nuo sutarties specialistai gali konsultuoti dėl konkrečios duomenų bazės projektavimo ir netgi atvykti pas klientą optimaliai suderinti jo duomenų bazę.

Taip *MySQL* programinė įranga platinama pagal visą licencijų įvairovę: nuo *GNU GPL* iki nuosavybinių, nesudarydama kliūčių kiekvienam naudotojui pasirinkti optimalų variantą.

16.5. Klausimai ir uždaviniai

1. Kas yra *MySQL*?
2. Kokios vartotojų grupės gali nemokamai atsisiųsti naujausią *MySQL* versiją?
3. Ar *MySQL* gali būti panaudota didelės įmonės serveryje?
4. Ar *MySQL* gali būti panaudota asmeniniame serveryje?

17. Laisvasis įrankis laisvam programuotojui – PHP

PHP yra programavimo kalba. *Apache* modulis įvykdo *PHP* kalba parašytą skriptą ir vartotojui pateikia *HTML* dokumentą. Laisvai platinama daugybė programų *PHP* kalba, todėl informacinių sistemų kūrėjai paprastai nekuria programų nuo pirmos eilutės, bet ieško jau sukurtų panašios paskirties programų ir patobulina jas bei pritaiko jas savo reikmėms.



17.1 pav. *PHP* logotipas

Nedidelė programuotojų dalis teigia, kad „P“ raidė akronime *LAMP* turėtų reikšti *Perl* arba *Python*. Jų nuomone *Python* yra modernesnė programavimo kalba ir taip teigiančių programuotojų gretos nuolat didėja. *Perl* kalba šiame kontekste vis mažiau naudojama.

Anksčiau aptartą *HTML* dokumentą puikiai papildo skriptai *PHP* kalba (17.1 pav.). Šis įrankis puikiai tinka ir yra dažnai naudojamas įvairios apimties projektuose. *PHP* kūrėjų bendruomenė yra puiki terpė įvairių tipų atviriesiems projektams⁴⁶.

PHP programavimo kalba su atvirąja programine įranga yra susijusi dviem aspektais:

1. Visų pirma pats šios kalbos modulis yra atvirasis projektas. Pati kalba yra kuriama atviros bendruomenės. Laisvai prieinami išsamūs *PHP* kalbos aprašai.
2. Šios kalbos techninė ir organizacinė realizacija puikiai tinka atviriesiems projektams, nepatogi nuosavybiniams projektams.

Šiame skyriuje aptarsime antrąjį aspektą – kodėl dauguma *PHP* programų yra atvirojo kodo. Trumpai susipažinsime su pagrindinėmis kalbos konstrukcijomis. Išmoksime, kaip toliau mokytis kalbos: kur rasti kalbos aprašus ir profesionaliai parašytų atvirųjų *PHP* programų.

⁴⁶ <http://php.net/>

17.1. Programos tekstų redagavimo poreikis

Maža universalių programų vartotojų dalis yra peržiūrinęjusi savo naudojamų programų pradinį tekstus. Dar mažesnė dalis yra juos redagavusi. Tokie vartotojai paprastai pasinaudoja nedidele sudėtingos programos funkcijų dalimi. Sudėtingesnėms internetinėms informacinėms sistemoms paprastai naudojama arba specializuota programinė įranga, arba universali programinė įranga, pritaikyta konkrečioms poreikiams. Abiem atvejais sistemą kurti turi specialistas, mokantis programuoti. Žiniatinklio eros pradžioje daug programų tekdavo programuoti nuo pirmos eilutės. Tokių programų kūrėjas turi kelis motyvus viešai paskelbti savo programų tekstus.

- Naudotojai ne tik pastebi programos klaidas, bet ir pasiūlo, kaip jas būtų galima ištaisyti. Aptarėme tai, kad dauguma instaliuojančių *PHP* programas moka programuoti. Tai natūralus bendradarbiavimas, nereikalaujantis jokių specifinių licencinių sutarčių.
- Kiti vartotojai programuotojai greičiausiai patobulins ir papildys programą. Pirminės programos kūrėjas nebūtinai gaus patobulintą versiją. Tačiau patobulintos versijos autoriai turi analogiškus motyvus viešai atskleisti savo programas. Dauguma atvirųjų licencijų reikalauja, kad patobulinta programa būtinai turi išlikti atviroji. Tai šiame motyvui formaliai padeda atvirosios licencijos.
- Šalutinis programos paviešinimo efektas – savęs pateikimas pasaulio programuotojų bendruomenės kontekste. Daugelį *PHP* programų kuria pavieniai programuotojai. Jie turi unikalią galimybę parodyti savo darbą, rasti bendraminčių, kuriančių panašias programas visame pasaulyje.

Tokių programų užsakovai dažnai nėra IT įmonės ir jie neturi komercinio intereso padaryti programą nuosavybinę. Pradiniai kodo fragmentai dažnai platinami su licencijomis, neleidžiančiomis uždaryti išvestinio kūrinio.

Praėjusiam skyriuje aptarėme, kaip veikia žiniatinklio serveris, pateikiantis nekintančius iš anksto sukurtus dokumentus. Dažnai to neužtenka. Interaktyviose sistemose, pavyzdžiui, el. parduotuvėse, vartotojo matomas vaizdas priklauso nuo jo atliktų veiksmų, todėl turi būti kuriamas programos. Net ir statinėse sistemose, pavyzdžiui, el. laikraščiuose, yra patogiau sukurti kuo paprastesnę sąsają straipsniams į duomenų bazę pateikti. Tada programa kuria vaizdą ekrane prie straipsnio pridėdama kitus komponentus. Bet kokios programos išvedamus duomenis būtų galima perduoti *Apache* serveriui, kuris juos pateiktų vartotojui. Tuo tikslu buvo sukurta *CGI-BIN* sąsaja. Žiniatinklio užduotims spręsti paplito *Perl* kalba, turinti patogių eilučių apdorojimo įrankių, kurie dažnai praverčia formuojant *HTML*.

17.2. Kodėl nuo *Perl* pereita prie *PHP*

Vėliau buvo sukurtas *Apache PHP* modulis. Sukurtos *PHP* kalbos sintaksė mažai skyrėsi nuo *Perl*. *PHP* tapo integruotas *Apache* komponentas. Maždaug 2000 m. ne mažai el. sistemų buvo perrašyta iš *Perl* į *PHP*. Tai suteikė ne tik techninių privalumų, bet ir lėmė patogesnę programavimą, nes pasitaikius klaidai *PHP* modulis galėjo pateikti išsamią informaciją, o „pakibus“ *Perl* programai *Apache* serveris nežinodavo problemos priežasčių, todėl ir vartotojui negalėdavo daug pasakyti.

17.3. Kaip veikia *PHP* modulis

Apache serverio aplanke gali būti įrašomi ne tik *HTML* failai, kurie pateikiami vartotojui tokie, kokie yra, bet ir *PHP* failai, kurie perduodami *PHP* moduliui. Tokiu atveju *PHP* modulis analizuoja tekstą ir apdoroja tai, kas yra tarp ženklų „<?“ ir „?>“. Likusi dalis perduodama vartotojui nepakeista. Tekstą nuo „<?“ iki „?>“ *PHP* modulis interpretuoja kaip programą *PHP* kalba. Jei failu yra keli tokie intarpai, jie interpretuojami kaip viena programa, išsaugomos kintamųjų reikšmės. Pateikiamo dokumento *PHP* programos vietoje išvedamas programos pateikiamas tekstas.

Sudėtingesniuose projektuose failas dažnai prasideda ženklų „<?“ ir baigiasi „?>“, visą *HTML* tekstą kuria *PHP* programa. Į statinius puslapius kartais įterpiamos atskiros *PHP* komandos.

17.4. Programavimo kalbų *Perl* ir *PHP* sintaksė

Matomiausias sintaksės skirtumas tarp čia aptariamų ir senesnių kalbų yra kintamųjų pavadinimai. *C* ir kitų klasikinių kalbų kintamųjų pavadinimai būdavo sudaromi iš raidžių ir skaičių. *Perl* ir *PHP* kalbose kintamojo pavadinimo pradžioje rašomas ženklas „\$“. Dėl šio skirtumo programos tekstas *C* ir *PHP* kalbomis atrodo labai skirtingas, tačiau *Perl* ir *PHP* perėmė daugelį *C* kalbos konstrukcijų.

Priskyrimo sakiny:

```
$a = $b = 500;
```

Čia sujungti du priskyrimai. Šis sakiny iliustruoja iš *C* kalbos perimtą konstrukciją, kai priskyrimo sakiny turi reikšmę, analogišką priskirtajai, todėl ši reikšmė gali būti dar kartą priskiriama.

Aritmetiniai veiksmai:

```
$n = 2 * $a + $b / 300;
```

Rezultato išvedimas:

```
echo $n;
```

Perl ir *PHP* turi sudėtingus eilučių apdorojimo mechanizmus. Taip pat šiomis kalbomis patogiu atlikti paprasčiausius veiksmus – kurti naują eilutę įtraukiant kintamųjų reikšmes.

```
echo "Kintamųjų reikšmės yra: $a, $b ir $n";
```

17.5. Elektroninių projektų specifika

Programinė įranga įkeliama į serverį. Serverį prižiūrinčios įmonės darbuotojai gali peržiūrėti serveryje esančių failų turinį. Mažesniuose projektuose programinė įranga ir serveris užsakomi iš skirtingų asmenų. Todėl programos kūrėjas sunkiai galėtų apsaugoti programos pradinį tekstą.



17.2 pav. Bendruomenės kuriamas *PHP* vadovas⁴⁷

Kai kurie programuotojai šią problemą sprendžia apsunkindami trečiųjų šalių galimybes pasinaudoti jų programa. Tačiau dažniau pasirenkamas kitas būdas. Dauguma užsakovų keliamų

⁴⁷ http://en.wikibooks.org/wiki/PHP_Programming

uždavinių nėra unikalūs. Ne vienas programuotojas jau yra kūręs ką nors panašaus į užsakomą programą, todėl prieš pradėdant programuoti šioje srityje būtina paieškoti, kokios programos jau yra sukurtos.

WWW sprendimams reikia specializuotų programų. Ypač daug programuotojų visame pasaulyje kuria *PHP* programas. Internete laisvai prieinami ne tik programų tekstai, bet ir knygos apie *PHP* (17.2 pav.).

17.6. Klausimai ir uždaviniai

1. Kas yra *PHP*?
2. Sukurkite dokumentą su *PHP* skriptu, kurio antraštėje ir dokumento tekste būtų nurodyta, kiek milisekundžių praėjo nuo vidurnakčio iki tol, kol dokumentas buvo užklaustas. Abu šie skaičiai turėtų sutapti.