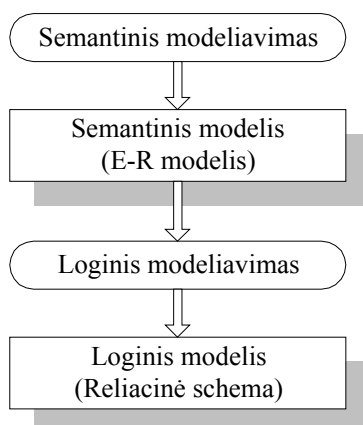


4. Semantinis modeliavimas

4.1. Įvadas

Duomenų bazės projektavimas yra gana sudėtingas ir ilgas procesas, kuriame būtinai turėtų dalyvauti tvarkomo dalyko specialistai. DB loginė struktūra (schema) yra visos informacinės sistemos pagrindas. Todėl projektuojant dideles DB svarbu sudaryti tikslų dalykinės srities aprašą. Formalizuotas dalykinės srities aprašas, kuris gali būti suprantamas tiek DB specialistams, tiek ir dalyko specialistams, vadinamas **semantiniu** (arba **sampratos, konceptualiu**) **modeliu**. Reliacinis modelis yra lakoniškas ir jame nelabai atspindi dalykinės srities semantika. Pastebėsime, kad modelyje vaizduojami ne visi realaus pasaulio objektai, reiškiniai, sąryšiai ir pan., o tik atskiros detalės. Šiame skyriuje aptarsime vieną populiariausių semantinių modelių – E–R (esybė – sąryšis, angl. *Entity-Relationship*) modelį, kurį pasiūlė 1976 m. Čenas (P. P.-S. Chen). Po to sužinosime kaip E-R modelį atvaizduoti DB reliacine schema.

E-R modelis plačiai taikomas sudėtingoms kompiuterinėms sistemoms projektuoti pagal objektų technologiją. Semantinio modeliavimo idėjos taip pat vertingos projektuojant DB. DB schemas projektavimo technologija, pagrįsta semantiniu modeliu, vadinama “iš viršaus į apačią” (angl. *top-down*), kadangi projektavimas pradedamas nuo realaus pasaulio aukščiausio abstrakcijos lygio ir baigiamas palyginti žemo abstrakcijos lygio konkrečia logine DB schema. Semantinio modelio sudarymas yra pirmasis DB projektavimo etapas. Turint semantinį modelį galima sudaryti DB reliacinę schemą.



4.1 pav. DB projektavimo etapai

Semantinio modeliavimo metodą išsiaiškinsime modeliuodami supaprastintą biblioteką. Tarkime, bibliotekoje laikomos tik knygos. Žurnalų, pavyzdžiui, nenagrinėsime. Biblioteka gali turėti keletą kiekvienos knygos egzempliorių. Kiekviena knyga turi unikalų ISBN numerį, o kiekvienas bibliotekos turimas knygos egzempliorius turi dar ir unikalų tapatumo (registracijos) numerį. Bibliotekos knygomis naudojasi skaitytojai, turintys skaitytojo pažymėjimą su numeriu. Kiekvienam skaitytojui suteikiama galimybė pasiimti kiek nori knygų. Imant knygą nustatoma jos grąžinimo data. Bibliotekoje yra sisteminis katalogas, kurio požymis – žinių sritis. Visos bibliotekos knygos suskirstytos atsižvelgiant į žinių sritį.

4.2. Pagrindinės E-R modelio sąvokos

Aptarkime pagrindines E-R modelio sąvokas.

Esybė. Esybe (angl. *entity*) grindžiama vieno tipo objektų klasė. Esybės - tai fiziškai egzistuojantys ar mintyse suvokiami ir lengvai skiriami modeliuojamo pasaulio bendrieji vaizdai (sąvokos, konceptai). Kiekvienai esybei modeliuojamoje sistemoje suteikiamas unikalus tapatumo vardas. Pavyzdžiui, modeliuojant biblioteką, galima išskirti tokias esybes: *Knyga* – abstraktus, užregistruotas bibliotekoje leidinys; *Katalogas* – sisteminis knygų katalogas, kuriame žinių sričiai priskiriama dalis bibliotekos knygų; *Skaitytojas* – bibliotekos skaitytojas; *Egzempliorius* – bibliotekos knygos egzemplioriaus abstrakcija. Konkrečios knygos – tai esybės *Knyga* objektai, konkretūs knygų egzemplioriai – esybės *Egzempliorius* objektai ir pan.

Atributai. Visi objektai turi tam tikrus požymius – **atributus** (angl. *attributes*). Visi vienos esybės objektai turi tuos pačius atributus. Kiekvienam atributui priskiriama galimų reikšmių aibė. Paprastai objekto atributai yra **vienareikšmiai** (kiekvienas objektas gali turėti tik vieną atributo reikšmę), pavyzdžiui, kiekvienas skaitytojas turi vieną asmens kodą. Tačiau kai kurie atributai gali būti **daugiareikšmiai** (angl. *multivalued*), pavyzdžiui, knyga gali būti parašyta keleto autorių. Taigi esybės *Knyga* atributas *Autorius* yra daugiareikšmis. Atributas gali būti **paprastas** (angl. *simple*) arba **sudėtinis** (angl. *composite*). Sudėtinį atributą sudaro keletas paprastų atributų. Pavyzdžiui, esybės *Skaitytojas* objektai gali turėti sudėtinį atributą *Adresas*, kurį, tarus, kad visi skaitytojai gyvena tame pačiame mieste, sudaro trys paprasti atributai: *Gatvė*, *Namas* ir *Butas*. Atributai, vienareikšmiškai atitinkantys esybės objektą, vadinami **raktu** (raktiniais atributais). Pvz., esybės *Knyga* objektų (knygų) atributas *ISBN* yra raktas, kadangi kiekviena knyga turi unikalų ISBN numerį. Kai kurie atributai gali būti **išvestiniai**, apskaičiuojami pagal kitų atributų reikšmes, pvz., skaitytojo amžių galima apskaičiuoti pagal jo gimimo ir einamąją datas.

Silpnoji esybė. Kai kurių esybių objektų prasmingumas priklauso nuo kitų esybių objektų. Pavyzdžiui, negali būti neegzistuojančios knygos egzemplioriaus. Todėl esybė *Egzempliorius* priklauso nuo esybės *Knyga*. Tačiau galima laikyti, kad knygos bibliotekoje nėra, jei nėra nei vieno jos egzemplioriaus. Nors šios esybės priklauso viena nuo kitos, tačiau jos yra gana savarankiškos. Kai kurios esybės gali būti tiek priklausomos nuo kitos (pagrindinės) esybės, kad priklausomos esybės objektai vienareikšmiškai nustatomi tik panaudojant pagrindinės esybės raktą. Tokio lygio priklausoma esybė vadinama **silpnąja** (angl. *weak*). Jei esybė nėra tokia priklausoma, tai ją galima vadinti **stipriąja** (angl. *strong*), jei ji nėra silpnoji. Silpnosios esybės pavyzdys - knygos skyrius. Knygos skyrius vienareikšmiškai identifikuojamas skyriaus numeriu kartu su knygos raktu (ISBN numeriu). Tačiau knygų skyriai bibliotekoje paprastai nėra tvarkomi (nagrinėjami). Todėl tokios esybės į bibliotekos modelį neįtrauksime.

Sąryšis. ER modelio sąvokų autorius sąryšį (angl. *relationship*) apibrėžė kaip dvinarę asociaciją, nusakančią esybių tarpusavio santykį arba sąveiką. Sąryšio pavadinimas turi atspindėti jo esmę. Pavyzdžiui, tarp esybių *Knyga* ir *Katalogas* objektų egzistuoja sąryšis, nusakantis knygos priklausomybę sisteminio katalogo skirsniams, t.y. žinių sritims. Pavadiname šį sąryšį *Priklauso*. Esybės, dalyvaujančios sąryšyje vadinamos **dalyviais**, o dalyvių kiekis – **sąryšio laipsniu**. Vienanariai sąryšiai – tai sąryšiai vienoje esybėje. Tokie sąryšiai taip pat vadinami **rekursyviais** (angl. *recursive*). Dažniausiai sutinkami **dvinariai** sąryšiai, kuriuose dalyvauja dvi esybės. Matematinė prasme, sąryšis R tarp esybių E_1, E_2, \dots, E_n yra Dekarto sandaugos poaibis:

$$R \subseteq E_1 \times E_2 \times \dots \times E_n,$$

čia n – sąryšio laipsnis. Sąryšio (Dekarto sandaugos) elementas $(e_1, e_2, \dots, e_n) \in R$ vadinamas sąryšio egzemplioriumi (angl. *instance*), kur $\forall i : 1 \leq i \leq n : e_i \in E_i$.

Yra trys pagrindinės sąryšių rūšys: **vienas su vienu** (sutrumpintai žymima **1:1**), **vienas su daug (1:N)**, **daug su daug (N:M)**. Tarp dviejų esybių egzistuoja sąryšis 1:1, jei vienas vienos esybės objektas yra susijęs su vienu kitos esybės objektu. Kitaip tariant, sąryšis $R \subseteq E_1 \times E_2$

yra 1:1 sąryšis, jei $\forall e_1 \in E_1$ priskiriamas daugiausiai vienas objektas $e_2 \in E_2$ ir atvirkščiai: $\forall e_2 \in E_2$ priskiriamas daugiausiai vienas objektas $e_1 \in E_1$. Sąryšio 1:M atveju vieną vienos esybės objektą gali atitikti daugiau nei vienas kitos esybės objektas. Kai tarp dviejų esybių egzistuoja N:M sąryšis, tai vieną pirmos esybės objektą gali atitikti keletas antros esybės objektų ir vieną antros esybės objektą gali atitikti keletas pirmos esybės objektų. Pavyzdžiui, tiek sąryšis *Yra egzempliorius* tarp esybių *Knyga* ir *Egzempliorius*, tiek ir sąryšis *Skaito* tarp esybių *Skaitytojas* ir *Egzempliorius* yra 1:N sąryšiai, nes bibliotekoje gali būti keletas konkrečios knygos egzempliorių, o vienas skaitytojas tuo pačiu metu gali būti pasiėmęs keletą knygų (egzempliorių). Be to, konkretus egzempliorius negali būti kelių knygų egzemplioriumi, kaip ir vieno egzemplioriaus vienu metu negali skaityti (pasiimti) keli skaitytojai. Tarp esybių *Knyga* ir *Katalogas* egzistuoja N:M tipo sąryšis *Priklauso*, nes vienai žinių sričiai gali priklausyti keletas bibliotekos knygų ir ta pati knyga gali priklausyti keletui sričių.

Kartais patogu atributus priskirti ne esybėms, bet sąryšiui. **Sąryšio atributai** apibūdina sąryšį. Pavyzdžiui, sąryšį *Skaito* galima papildomai nusakyti knygos egzemplioriaus paėmimo ir grąžinimo data. Taip atributai *Paėmimo data* ir *Grąžinimo data* gali būti sąryšio *Skaito* atributai. Tačiau sąryšiai su atributais nėra būtini E-R modelyje. Visuomet galima sudaryti naują esybę su atributais, nusakančiais sąryšį, ir įtraukti naująją esybę į sąryšį. Knygos egzemplioriaus paėmimo ir grąžinimo datas priskirsime esybei *Egzempliorius*, kuri dalyvauja sąryšyje *Skaito*.

Kiekvienas iš trijų tipinių sąryšių gali būti **besąlygišku** (*stipriu*, angl. *regular, non-weak*) arba **sąlygišku** (*silpnu*, angl. *weak*). Sąryšis yra besąlygiškas, jei sąryšyje dalyvauja visi esybės objektai. Sąlygiškame sąryšyje gali dalyvauti ne visi esybės objektai. Sąryšis gali būti sąlygiškas iš vienos pusės ir besąlygiškas iš kitos pusės. Pavyzdžiui, sąryšis *Yra egzempliorius* tarp esybių *Knyga* ir *Egzempliorius* yra besąlygiškas iš esybės *Egzempliorius* pusės, nes kiekvienas egzempliorius yra kažkurios knygos egzempliorius. Sąryšis *Yra egzempliorius* yra besąlygiškas ir iš kitos pusės, nes bibliotekoje nėra knygos, jei nėra nė vieno jos egzemplioriaus. Sąryšis *Priklauso* tarp esybių *Katalogas* ir *Knyga* yra besąlygiškas iš esybės *Knyga* pusės, nes kiekvieną knygą galima priskirti bent vienai žinių sričiai. Sąryšis *Priklauso* besąlygiškas iš esybės *Katalogas* pusės, nes kataloge nėra atitinkamos skilties (žinių srities), jei bibliotekoje nėra nei vienos knygos, priklausančios tai sričiai. Sąryšis *Skaito* yra sąlygiškas iš abiejų pusių, nes knygos egzempliorius nebūtinai turi būti skaitomas ir skaitytojas nebūtinai turi būti paėmęs bent vieną knygą.

Tarp stipriosios ir silpnosios esybių dažniausiai egzistuoja sąryšis 1:N, rečiau 1:1. Silpnąją ir stipriąją esybes negali sieti N:M sąryšis.

Ryšio priklausomumas kuriai nors rūšiai yra integralumo sąlyga, kuri modeliuojamame pasaulyje turi būti patenkinta visada, ne tik kurioje nors konkrečioje situacijoje.

Kategorizavimas. E-R modelyje galima taikyti kategorizacijos (tipizacijos) principą, pagal kurį įvedama esybės potipio ir virštipio sąvokos. Kiekviena esybė gali turėti keletą potipių. Daugelyje uždavinių tarp kelių esybių galima išskirti bendrą atributų aibę. Tuomet bendrus tokių esybių atributus galima išskirti į atskirą esybę – **virštipį**. Eybės iš kurių buvo išskirtos bendrosios dalys, tampa virštipio **potipiais** (virštipio kategorijomis). Procesas, kai išskiriamas virštipis vadinamas **apibendrinimu**. Galimas ir atvirkščias, **specializavimo** procesas, kurio metu esybėje išskiriami keli atributų poaibiai ir esybė suskaidoma į bendrąją dalį – virštipį bei kelias specializuotas dalis – potipius. Esminė sąryšio virštipis-potipiai savybė – paveldimumas. Potipiai paveldi visus savo virštipio atributus ir sąryšius. Paveldimumas išreiškiamas specialia sąryšio rūšimi “yra” (angl. *is-a*). Pavyzdžiui, esybę *Skaitytojas* galima specializuoti išskiriant du potipius: *Studentas* ir *Dėstytojas*. Pirmosios esybės-potipio atributais gali būti studijų knygelės numeris *SKNR* ir *Specialybė*. Eybė *Dėstytojas* gali būti charakterizuojama atributais *Katedra* ir *Pareigos*. Savo ruožtu, esybę *Dėstytojas* galima apibendrinti esybe *Darbuotojas* ar specializuoti į esybes *Asistentas*, *Profesorius* ir pan. Taip gaunama **tipų hierarchija**. Priklausomai nuo skaitytojų rato, esybę *Skaitytojas* galima specializuoti ir kitaip.

Pastebėsime, kad pagrindines E-R modelio sąvokas aptarėme tik bendriausia jų prasme. Kai kurias pateiktas sąvokas galima detalizuoti. Pavyzdžiui, 1:N ir N:M tipų sąryšius galima

charakterizuoti tiksliau, sąvoką “daug” (“N”, “M”) patikslinant minimalia ir maksimalia ribomis, nurodant, kiek kartų mažiausiai ir kiek kartų daugiausiai esybė gali dalyvauti sąryšyje. Tai sąryšio **kardinalumas** (angl. *cardinality*). Tačiau detalesnes E-R modelio charakteristikas paliksime savarankiškam darbui.

4.3. E-R diagramos

E-R modelį vaizdžiai atspindi **E-R diagrama**. Pateiksime pagrindinių E-R modelio sąvokų vaizdavimo E-R diagrama principus.

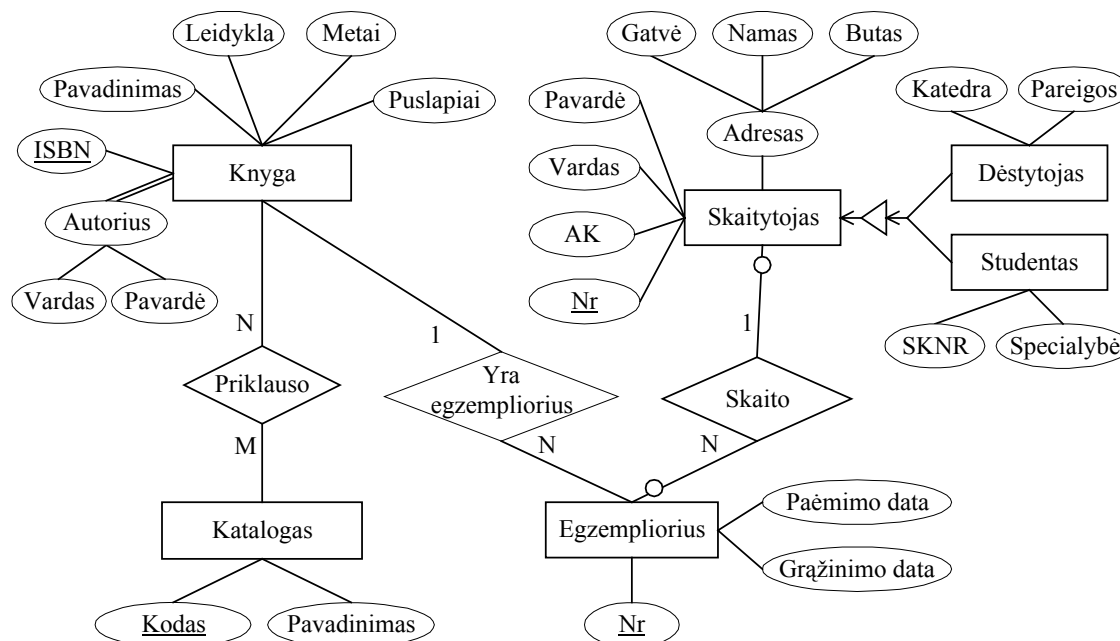
Esybės. Kiekviena esybė vaizduojama stačiakampiu, kurio viduje rašomas esybės vardas. Silpnosios esybės vaizduojamos stačiakampiu su dvigubu rėmeliu.

Atributai. Atributai E-R diagramoje vaizduojami ovalais, susiejant juos su atitinkamos esybės stačiakampiu ištisine linija. Ovalo viduje rašomas atributo vardas. Išvestinių (apskaičiuojamų) atributų ovalai brėžiami punktyrine linija. Sudėtinio atributo sudedamosios dalys vaizduojamos ovalais, sujungtais ištisine linija su sudėtinio atributo ovalu. Raktinių atributų vardai pabraukiami. Daugiareikšmiai atributai jungiami su esybe dviguba linija.

Sąryšiai. Kiekvienas sąryšis vaizduojamas rombu su sąryšio vardu viduje. Rombas apvedamas dviguba linija, jei sąryšis yra tarp silpnosios esybės ir pagrindinės esybės. Sąryšį vaizduojantis rombas sujungiamas ištisinėmis linijomis su visais sąryšio dalyviais. Kiekviena tokia linija pažymima “1”, “N” ar “M” sąryšio tipui pažymėti. Linija, jungianti sąryšio rombą su silpnąja esybe, yra dviguba. Jei sąryšis turi atributus, tai šie vaizduojami atskirais ovalais ir sujungiami linija su sąryšį žyminčiu rombu. Jei sąryšis yra sąlyginis, tai linija, jungianti jį su esybe, iš kurios pusės sąryšis sąlyginis, yra pažymima tuščiu skrituliuku.

Potipiai ir virštipiai. Jeigu esybės X_1, X_2, \dots, X_n yra esybės Y potipiai, tai iš kiekvienos esybės X_i ($1 \leq i \leq n$) stačiakampio brėžiama linija į grafinį elementą, vadinamą **deskriptoriumi** (jį vaizduosime trikampi), o iš jo į esybės Y stačiakampį su rodykle linijos gale.

Pavaizduokime, bibliotekos E-R modelį *Biblioteka* E-R diagrama:



4.2 pav. Semantinio modelio *Biblioteka* E-R diagrama

4.4. E-R modelio vaizdavimas reliaciniu duomenų modeliu

Semantinis modelis vartojamas ankstyvosiose projektavimo stadijose. Juo gali naudoti kiti sistemos kūrėjai vėlesniuose etapuose. DB projektavimas yra gana subjektyvus procesas, nes priklauso nuo projektuotojo patirties ir dalykinės srities supratimo. Tačiau E-R modeliui egzistuoja algoritmas, vienareikšmiškai siejantis semantinį modelį su reliaciniu.

Apžvelkime ER modelio suvedimo į reliacinį duomenų modelį **bendrasias taisykles**.

1. Kiekvieną E-R modelio **esybę** reliaciniame modelyje atitinka lentelė. Lentelės vardas turi atitikti konkrečios RDBVS reikalavimus, todėl jis gali skirtis nuo esybės vardo.
2. Kiekvienam **esybės vienareikšmiam atributui** atitinkamoje reliacinėje lentelėje turi būti stulpelis. Stulpelio vardas gali sutapti su atributo vardu arba skirtis nuo jo. Stulpelio vardas formuojamas pagal RDBVS reikalavimus. Išvestiniai (apskaičiuojami) atributai į lentelę neįtraukiami. **Sudėtiniams** atributams į lentelę įtraukiami tik juos sudarantys paprastieji atributai.
3. Kadangi reliaciniame modelyje aibė negali būti reikšmė, tai **daugiareikšmio** atributo negalima atvaizduoti lentelės stulpeliu. Galimi du daugiareikšmių atributų realizavimo būdai:
 - Jei žinoma daugiareikšmio atributo A maksimalus paprastųjų reikšmių kiekis n , tai į esybę atitinkančią lentelę įtraukiama n stulpelių A_1, A_2, \dots, A_n – po vieną kiekvienai galimai atributo reikšmės dedamajai. Šiuo atveju jei n yra gana didelis, tai esybės lentelė turės daug stulpelių. Be to, dauguma lentelės stulpelių A_1, A_2, \dots, A_n reikšmių gali būti NULL.
 - Daugiareikšmiui atributui A sukurama atskira lentelė, į kurią įtraukiamas atitinkamas stulpelis A . Jei A yra sudėtinis, tai įtraukiami visi jį sudarantys paprastieji atributai. Į lentelę taip pat įtraukiamas esybės, kuriai priklauso A , raktas K , kuris tampa lentelės išoriniu raktu. Lentelės pirminiu raktu tampa A ir K kartu paėmus. Šio būdo trūkumas – atributui kuriama atskira lentelė, tačiau šioje lentelėje nebus NULL reikšmių.
4. **Esybės raktas** tampa pirminiu lentelės raktu. Semantiniame modelyje raktiniai atributai esybei gali būti ir neapibėžti. Tokiu atveju iš esybės atributų išskiriama vienas ar keli atributai, sudarysiantys pirminį raktą. Arba galima parinkti visiškai naują papildomą atributą, kuris taptų pirminiu raktu. Jei esybė turi keletą raktų, tai vienas iš jų parenkamas pirminiu.
5. Į kiekvieną lentelę, atitinkančią **silpnąją esybę**, įtraukiami pagrindinę (stipriąją) esybę atitinkančios lentelės pirminio rakto atributai. Šie atributai silpnąją esybę atitinkančioje lentelėje tampa išoriniu raktu į pagrindinę lentelę. Silpnąją esybę atitinkančioje lentelėje pirminis raktas sudaromas iš atributų (stulpelių), vienareikšmiškai identifikuojančių silpnąją esybę pagrindinės esybės atžvilgiu, ir visų pagrindinės lentelės pirminio rakto stulpelių.
6. Bendriausiu atveju kiekvieną **sąryšį** galima realizuoti atskira lentele. Į lentelę įtraukiami abiejų sąryšyje dalyvaujančių esybių raktai, kurie tampa lentelės išoriniais raktais. Jei esybė yra privaloma sąryšyje, tai atitinkamo išorinio rakto stulpeliai negalės įgyti NULL reikšmės. Realizuojančios sąryšį lentelės pirminiu raktu gali būti visi išorinių raktų stulpeliai. Dvinarius sąryšius 1:1 ir 1:N galima realizuoti ir neįvedant specialios lentelės. Šių tipų dvinarius sąryšius, o taip pat ir N:M tipo sąryšius, galima realizuoti pagal specializuotas žemiau pateiktas taisykles.
7. Kiekvienam **1:N sąryšiui** į lentelę L_N , atitinkančią esybę, pažymėtą "N", įtraukiamas lentelės, atitinkančios esybę, pažymėtą "1", pirminis raktas ir jis tampa lentelės L_N išoriniu raktu. Jei esybė, atitinkanti "1", yra privaloma sąryšyje, tai išorinio rakto stulpeliai negalės įgyti NULL reikšmės. Į lentelę L_N taip pat įtraukiami visi sąryšio atributai, jei tokie yra.
8. Kadangi **sąryšis 1:1** yra simetrinis, tai laisvai pasirenkama, kuri esybė bus "pagrindinė", o kuri "priklausoma". Tokie pat vaidmenys suteikiami ir atitinkamoms lentelėms. Jei vienos esybės dalyvavimas sąryšyje yra būtinas, o kitos ne, tai rekomenduojama "priklausoma"

lentele pasirinkti būtinąją esybę atitinkančią lentelę. Į “priklausomą” lentelę įtraukiami “pagrindinės” lentelės pirminio rakto stulpeliai ir jie paskelbiami išoriniu raktu. Jei “priklausoma” esybė yra privaloma sąryšyje, tai išorinio rakto stulpeliai negalės įgyti NULL reikšmės. Jei sąryšis turi atributus, tai jie įtraukiami į “pagrindinę” lentelę.

9. Kiekvienas **N:M sąryšis** realizuojamas sudarant atskirą lentelę. Į lentelę įtraukiami abiejų lentelių, atitinkančių sąryšyje dalyvaujančias esybes, pirminiai raktai, kurie tampa išoriniais raktais. Į lentelę taip pat įtraukiami visi sąryšio atributai. Lentelės pirminiu raktu tampa abiejų susietų lentelių pirminių raktų kombinacija.

10. Ryšį **virštipis-potipis** galima atvaizduoti keliais būdais:

- Visoms susietoms šio tipo sąryšiu esybėms yra sukurama viena lentelė. Į lentelę įtraukiami visi virštipio ir visų potipių atributai. Tokiu atveju, kiekvienam esybės egzemplioriui (objektui) dalis atributų bus neprasmingi. Visi potipių atributus atitinkantys stulpeliai galės įgyti NULL reikšmę. Šiuo atveju galima situacija, kai dauguma lentelės reikšmių bus NULL. Taip prarandama apibendrinimo sąvokos prasmė. Be to, teks įvesti papildomas taisykles potipiams atskirti vieną nuo kito. Šio būdo privalumas – sukurama tik viena lentelė.
- Tiek virštipiui, tiek ir kiekvienam potipiui sukuriamos atskiros lentelės. Į visas potipius atitinkančias lenteles įtraukiamas virštipio pirminis raktas, kuris tampa potipio lentelės tiek pirminiu raktu, tiek ir išoriniu raktu. Šio būdo trūkumas – sukurama daugiau lentelių.

Atvaizduokime E-R diagramą *Biblioteka* duomenų bazės *Biblioteka* reliacine schema.

Pagal taisykles 1, 2 ir 4 kiekvienai pagrindinei esybei gauname po lentelę:

Knyga(ISBN, Pavadinimas, Leidykla, Metai, Puslapiai);

Egzempliorius(Nr, Paėmimo_data, Grąžinimo_data);

Katalogas(Kodas, Pavadinimas);

Skaitytojas(Nr, AK, Vardas, Pavardė, Gatvė, Namas, Butas).

Pagal taisyklę 3 esybės *Knyga* daugiareikšmiui atributui *Autorius* gauname naują lentelę:

Autorius(ISBN, Vardas, Pavardė),

išorinis raktas: *ISBN* nurodo į *Knyga*.

Silpnųjų esybių E-R diagramoje nėra, todėl taisyklės 5 netaikome. Kadangi mūsų modelyje yra tik 1:N, N:M bei virštipis-potipis tipo sąryšiai, tai taisyklių 6 ir 8 netaikome. Pagal taisyklę 7, lentelę *Egzempliorius* papildome stulpeliais – lentelių *Knyga* ir *Skaitytojas* pirminiais raktais, kurie lentelėje *Egzempliorius* tampa išoriniais raktais. Kadangi stulpelis *Nr* jau yra lentelėje, lentelės *Skaitytojas* pirminį raktą įtraukdami jį į lentelę *Egzempliorius* pervardijame. Papildyta lentelės *Egzempliorius* schema:

Egzempliorius(Nr, Paėmimo_data, Grąžinimo_data, *ISBN*, *Skaitytojo_Nr*),

išoriniai raktai: *ISBN* nurodo į *Knyga*,

Skaitytojo_Nr nurodo į *Skaitytojas*.

Panaudoję taisyklę 9 suformuojame naują lentelę

Priklauso(ISBN, Kodas),

išoriniai raktai: *ISBN* nurodo į *Knyga*,

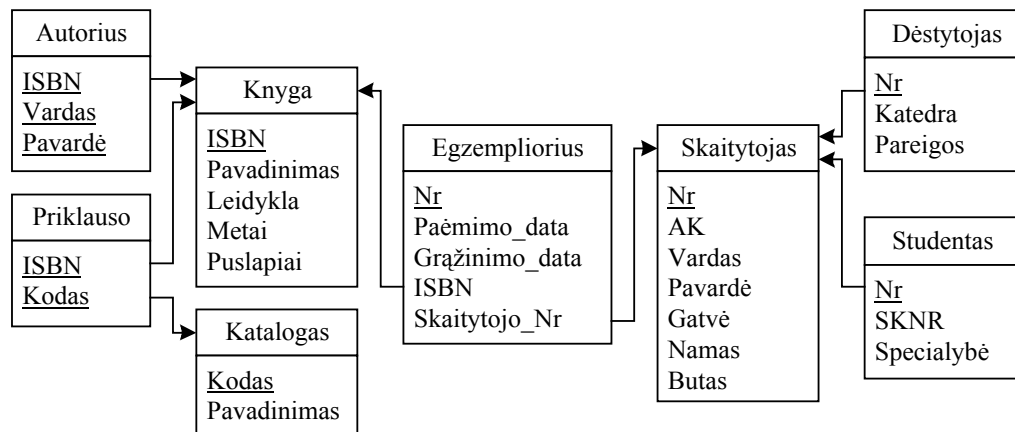
Kodas nurodo į *Katalogas*.

Jau minėjome, kad atributus *Paėmimo_data* ir *Grąžinimo_data* E-R modelyje galima buvo priskirti ne esybei *Egzempliorius*, bet sąryšiui *Skaito*. Tačiau ir tokiu atveju, realizuojant sąryšį *Skaito* pagal taisyklę 7, atitinkami stulpeliai *Paėmimo_data* ir *Grąžinimo_data* taptų lentelės *Egzempliorius* dalimi. Todėl, ir tokiu atveju gautume tokią pačią lentelės *Egzempliorius* schemą.

Pagal taisyklę 10 kiekvienam esybės-virštipio *Skaitytojas* potipiui *Dėstytojas* ir *Studentas* suformuojame po atskirą lentelę:

Dėstytojas(Nr, *Katedra*, *Pareigos*),
išorinis raktas: *Nr* nurodo į *Skaitytojas*;
Studentas(Nr, *SKNR*, *Specialybė*),
išorinis raktas: *Nr* nurodo į *Skaitytojas*.

Pavaizduokime DB *Biblioteka* reliacinę schemą grafiškai:



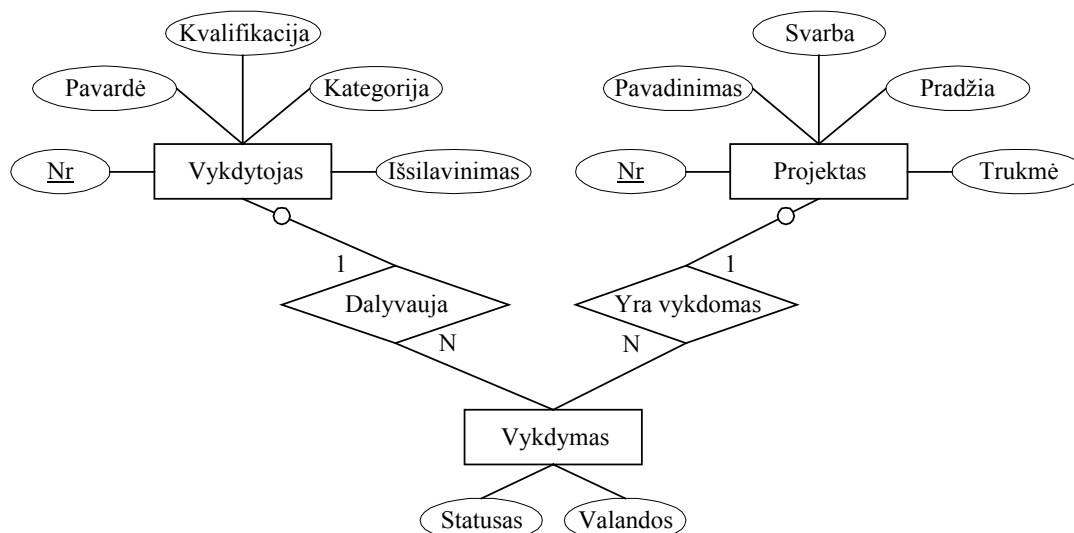
4.3 pav. Duomenų bazės *Biblioteka* reliacinė schema.

4.5. DB “Darbai” ER modelis

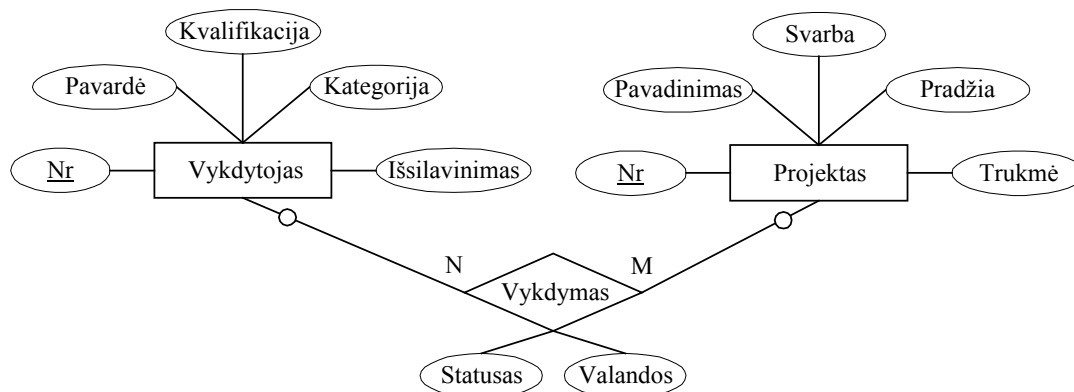
Pavaizduokime E-R diagrama anksčiau nagrinėtą dalykinę sritį, atitinkančią duomenų bazę *Darbai*. Tiksliau, pakartokime DB *Darbai* projektavimą – sudarykime šios srities E-R modelį ir po to atvaizduokime jį reliacine schema.

Kaip ir anksčiau, nagrinėkime paprasčiausią įstaigos veiklos atvejį, kai įstaigos darbuotojai vykdo projektus, atlikdami juose tam tikrą vaidmenį ir skirdami jo vykdymui tam tikrą laiką. Praktiškai nedvejojant galima išskirti dvi esybes: *Vykdytojas* ir *Projektas*. Kiekvienos iš šių esybių atributų rinkinys priklauso nuo įstaigos veiklos parametrų. Apsiribokime ankstesniame skyriuje nagrinėtais vykdytojo atributais: Nr, *Pavardė*, *Kvalifikacija*, *Kategorija*, *Išsilavinimas*, ir projekto atributais: Nr, *Pavadinimas*, *Svarba*, *Pradžia*, *Trukmė*. Sumodeliuokime darbuotojų (vykdytojų) dalyvavimą projektuose. Šį uždavinį galima spręsti keliais būdais.

- Į vykdytojų dalyvavimą projektuose galima žiūrėti kaip į atskirą esybę *Vykdymas* su atributais *Statusas* ir *Valandos*, susiejant šią esybę tiek su esybe *Vykdytojas*, tiek ir su *Projektas*. Kadangi vienas vykdytojas gali dalyvauti vykdant kelis projektus, tai tarp esybės *Vykdytojas* ir *Vykdymas* egzistuoja 1:N tipo sąryšis *Dalyvauja*. Tars, kad vykdytojas gali nedalyvauti nė viename projekte, *Vykdytojas* yra nebūtinai šio sąryšio dalyvis. Panašiai tarp esybės *Projektas* ir *Vykdymas* taip pat yra 1:N sąryšis ir jis yra neprivalomas iš esybės *Projektas* pusės. Taip sudaryto E-R modelio *Darbai* schema pateikta 4.4 pav.
- Į vykdytojų dalyvavimą projektuose galima žiūrėti ir kaip į N:M tipo sąryšį *Vykdymas* tarp esybės *Vykdytojas* ir *Projektas*. Abu sąryšio dalyviai yra neprivalomi. Šis sąryšis turi du atributus *Statusas* ir *Valandos*. Taip sudaryto E-R modelio *Darbai* schema pateikta 4.5 pav.



4.4 pav. E-R modelio *Darbai* schema, modeliuojant vykdytojų dalyvavimą projektuose esybė.



4.5 pav. E-R modelio *Darbai* schema, modeliuojant vykdytojų dalyvavimą projektuose sąryšiu.

Atvaizduokime abi E-R diagramas reliacinėmis schemomis. Visų pirma, abiejose schemose (abiejuose modeliuose) esybės *Vykdytojas* ir *Projektas* yra modeliuojamos vienodai. Abiem atvejais gauname dvi tokias pačias lenteles:

Vykdytojai (Nr, Pavardė, Kvalifikacija, Kategorija, Išsilavinimas);
Projektai (Nr, Pavadinimas, Svarba, Pradžia, Trukmė).

Atvaizduojant esybes lentelėmis, jas pervardijome tik tam, kad lentelių vardai atitiktų anksčiau vartotus vardus. Esysbė *Vykdytmas* atvaizduojama lentele su dviem atributais:

Vykdytmas (Statusas, Valandos).

Modeliuojant sąryšį *Dalyvauja*, į pastarąją lentelę įtraukiamas lentelės *Vykdytojas* pirminis raktas, kuris tampa išoriniu raktu:

Vykdytmas (*Vykdytojas*, Statusas, Valandos).
 išorinis raktas: *Vykdytojas* nurodo į *Vykdytojai*.

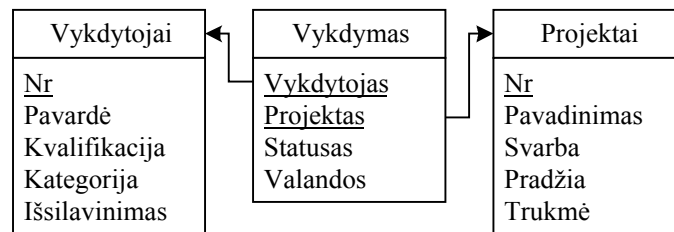
Sąryšis *Vykdomas* modeliuojamas į lentelę *Vykdytmas* įtraukiant lentelės *Projektas* pirminį raktą atitinkantį stulpelį, kuris tampa jos išoriniu raktu. Mūsų modelyje neatspindėjo faktas, kad vykdytojas, vykdant konkretų projektą, gali dalyvauti ne daugiau kaip viename vaidmenyje. Šis faktas reliacinėje schemoje gali būti atvaizduotas lentelės *Vykdytmas* pirminiu raktu. Galutinė lentelės *Vykdytmas* reliacinė schema:

Vykdytmas (*Projektas*, *Vykdytojas*, Statusas, Valandos),

išoriniai raktai: *Projektas* nurodo į *Projektai*,
Vykdytojas nurodo į *Vykdytojai*.

Antruoju atveju, kai E-R modelyje *Darbai* vykdytojų dalyvavimas projektuose vaizduojamas sąryšiu, lentelės *Vykdymas* reliacinę schemą pagal taisyklę 9 gauname vienu žingsniu. Be to, lentelės pirminis raktas nustatomas pagal tą pačią taisyklę.

Abu E-R modelius *Darbai* atvaizdavome tokia pačia reliacine schema. Tačiau antruoju atveju prireikė mažiau žingsnių tikslui pasiekti. Sugeneruotos duomenų bazės *Darbai* reliacinę schemą galima pavaizduoti grafiškai:



4.6 pav. Duomenų bazės *Darbai* reliacinė schema, gauta iš E-R modelio

4.6. ER modelis ir duomenų normalizavimas

Sudarę E-R modelio reliacinę schemą, šiai galime dar pritaikyti duomenų normalizavimo teoriją, kurią aptarėme ankstesniame skyrelyje. Tačiau duomenų normalizavimą galima taikyti ir semantinio modelio lygyje. Pateiksime E-R modelio normalizavimo algoritmą, užtikrinantį 4NF.

1. Išanalizuoti E-R modelį siekiant nustatyti esybes, kurios savo turiniu modeliuoja kelias skirtingas realaus pasaulio objektų klases, tarpusavyje susijusias tam tikrais sąryšiais. Jei yra tokių esybių, tai visas jas reikia išskaidyti į naujas esybes, apibrėžiant tarp jų reikiamus sąryšius. Gautas modelis bus 1NF.
2. Išanalizuoti visas esybes su sudėtiniais raktais. Jei yra esybių, kuriose nepirminiai atributai funkciškai priklauso nuo rakto dalies, tai kiekvieną tokią esybę suskaidyti į dvi ir apibrėžti tarp jų reikiamus sąryšius. Gautas modelis bus 2NF.
3. Išanalizuoti visus neraktinius visų esybių atributus, ieškant jų tranzityvių priklausomybių nuo esybės rakto. Kiekvienai tranzityviai priklausomybei suskaidyti esybę į dvi, nepaliekant tranzityvios priklausomybės, ir apibrėžti tarp naujųjų esybių reikiamus sąryšius. Gautas modelis bus 3NF.
4. Išanalizuoti visus visose esybėse egzistuojančius f-sąryšius, ieškant tokių, kurių determinantas nėra galimas raktas. Radus tokį f-sąryšį, esybę suskaidyti į dvi ir apibrėžti tarp naujųjų esybių reikiamus sąryšius. Gautas modelis bus BKNF.
5. Išanalizuoti visus visose esybėse egzistuojančius netrivialius daugiareikšmius sąryšius, ieškant tokių, kurie nėra f-sąryšiai. Radus tokį MV-sąryšį, esybę suskaidyti į dvi ir apibrėžti tarp naujųjų esybių atitinkamus sąryšius. Gautas modelis bus 4NF.
6. Visi nurodyti žingsniai yra vykdomi nuosekliai. Kadangi kiekvienos aukštesnės normalinės formos reikalavimai apima žemesnės formos reikalavimus, kai kuriuos žingsnius galima praleisti. Taip apdorojus E-R modelį ir vėliau jį atvaizdavus reliacine schema pagal anksčiau pateiktas taisykles, gausime duomenų bazės reliacinę schemą, esančią 4NF.