

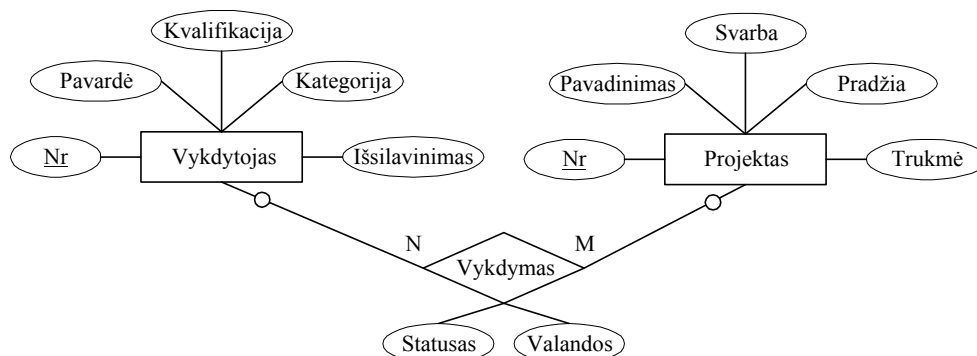
4. Semantinis modeliavimas

4.5". DB „Darbai“ ER modelis

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

Kaip ir anksčiau, nagrinėkime paprasčiausią įstaigos veiklos atvejį, kai įstaigos darbuotojai vykdo projektus, kiekvienas turi konkretų vaidmenį ir skiria projektui 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*.

Į vykdytojų dalyvavimą projektuose galima žiūrėti kaip į N:M tipo ryšį *Vykdymas* tarp esybių *Vykdytojas* ir *Projektas*. Abu ryšio dalyviai yra sąlygiški. Šis ryšys turi du atributus *Statusas* ir *Valandos*. Taip sudaryto ER modelio *Darbai* schema pateikta 4.4 pav.



4.4 pav. E-R modelio *Darbai* schema. Vykdytojų dalyvavimas projektuose modeliuojamas ryšiu.

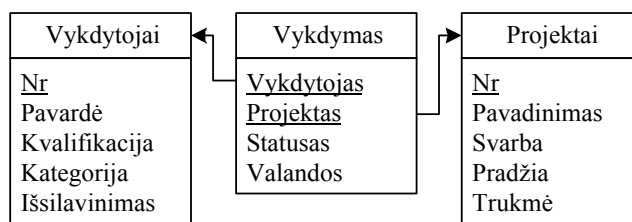
Pavaizduokime šią ER diagramą reliacine DB schema. Iš esybių *Vykdytojas* ir *Projektas* gauname dvi lenteles:

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

Sudarydami esybėms lenteles, jas pervardijome tik tam, kad lentelių vardai atitiktų anksčiau vartotus vardus. N:M tipo ryšiui *Vykdymas* sudarome atskirą lentelę *Vykdymas*, į kurią įtraukiame ryšio atributus ir jo dalyvių raktus, kurie tampa išoriniais raktais:

Vykdymas (Projektas, Vykdytojas, Statusas, Valandos),
 išoriniai raktai: *Projektas* nukreipia į *Projektai*,
Vykdytojas nukreipia į *Vykdytojai*.

Sudarytos duomenų bazės *Darbai* reliacinę schemą galima pavaizduoti grafiškai:



4.5 pav. Duomenų bazės *Darbai* reliacinė schema, atitinkanti ER modelį, pavaizduotą 4.4 pav.

4.5a. Ryšių realizavimo ypatumai

ER modelyje, pavaizduotame 4.4 pav., vykdytojų dalyvavimas projektuose yra išreikštas N:M ryšiu *Vykdymas* su dviem atributais. Taip apibrėžiama kiekvieno vykdytojo galimybė dalyvauti keliuose projektuose ir tai, kad vieną projektą gali vykdyti keli vykdytojai. Konkretaus vykdytojo indėlis projekte apibūdinamas ryšio atributais. Šiam ryšiui reliacinėje schemoje, pavaizduotoje 4.5 pav., atitinka lentelė *Vykdymas*, kurios raktas yra iš dviejų stulpelių, atitinkančių esybių raktus.

Lentelėje *Vykdymas* konkretaus vykdytojo dalyvavimas konkrečiame projekte gali būti pateiktas tik vienoje eilutėje. Tai reiškia, kad konkretus vykdytojas konkrečiame projekte gali dalyvauti tik viename statuse ir skirti tik vieną konkretų skaičių valandų. Taip sudarytoje lentelėje *Vykdymas* negalima kaupti vykdytojų dalyvavimo projektuose istorijos. Jeigu konkrečiame vykdytame vykdytojas projektui papildomai skiria konkretų skaičių valandų, tai jas reikia pridėti prie anksčiau skirtųjų. Jei vykdytojo statusas projekte keičiasi, tai mes turime „pamiršti“ ankstesnį statusą. Tačiau kartais būna svarbu išsaugoti ankstesnę informaciją.

N:M tipo ryšį atitinkančiai lentelei galima sudaryti **dirbtinį** raktą reiškiantį vykdyto numerį. Pavadinkime šį stulpelį *Nr*, o naująją lentelę – *Vykdymai*. Visi kiti stulpeliai – tokie pat kaip anksčiau sudarytoje lentelėje *Vykdymas*. Naujosios lentelės *Vykdymai* reliacinė schema yra

Vykdymai (*Nr*, *Projektas*, *Vykdytojas*, *Statusas*, *Valandos*),
išoriniai raktai: *Projektas* nukreipia į *Projektai*,
Vykdytojas nukreipia į *Vykdytojai*.

Taip sudarytoje lentelėje, kaip ir anksčiau sudarytoje lentelėje *Vykdymas*, galima saugoti duomenis apie vykdytojų dalyvavimą keliuose projektuose, bei apie visus kiekvieno projekto vykdytojus. Papildomai, kiekvienas vykdytojas konkrečiame projekte gali dalyvauti keliuose statusuose. Be to, kiekvieno vykdytojo konkrečiam projektui papildomai skiriamas valandas galima įsiminti naujoje eilutėje. Pavaizduokime naujosios lentelės fragmentą:

Vykdymai

<i>Nr</i>	<i>Projektas</i>	<i>Vykdytojas</i>	<i>Statusas</i>	<i>Valandos</i>
1	1	1	Programuotojas	30
2	1	1	Programuotojas	10
3	1	1	Testuotojas	40
4	1	2	Dokumentuotojas	100
5	1	2	Dokumentuotojas	200

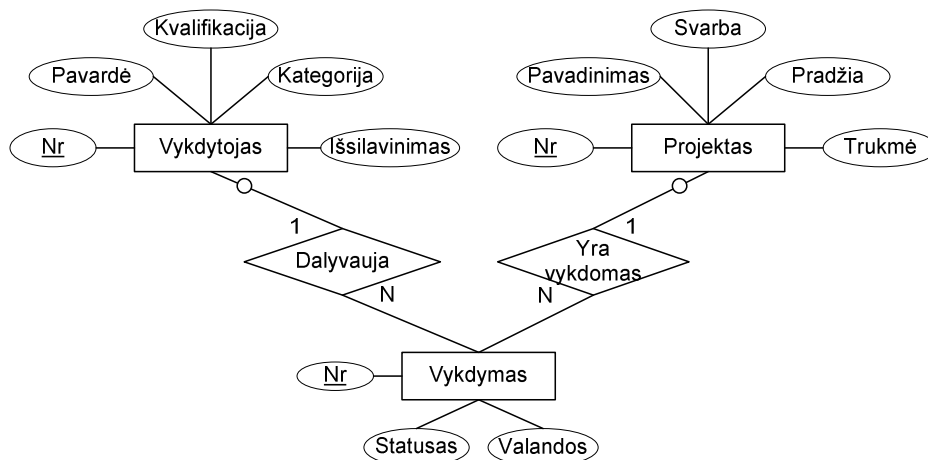
Taip sudaryta lentelė *Vykdymai* yra lankstesnė už anksčiau sudarytą lentelę *Vykdymai*. Tačiau, tai nereiškia, kad ji yra pranašesnė. Tai priklauso nuo konkrečių aplinkybių.

Tarkime, aplinkybės yra nusakomos FP {*Projektas*, *Vykdytojas*} → {*Statusas*, *Valandos*}. Tuomet, lentelė *Vykdymas* visiškai užtikrina šią FP. Lentelėje *Vykdymai* ši FP taip pat gali būti užtikrinama apibrėžiant papildomą raktą {*Projektas*, *Vykdytojas*}. Tačiau tuomet, stulpelis *Nr* lentelėje *Vykdymai* tampa beprasmiu. Šiuo atveju, anksčiau sudaryta lentelė *Vykdymas* yra pranašesnė už *Vykdymai*, nes ji yra paprastesnė.

Jei aplinkybės lemia, kad FP {*Projektas*, *Vykdytojas*} → {*Statusas*, *Valandos*} bendruoju atveju negalioja, tai lentelė *Vykdymas* negali turėti apibrėžtojo rakto. Lentelėje *Vykdymai* galioja FP *Nr* → {*Projektas*, *Vykdytojas*, *Statusas*, *Valandos*}, kuris pakankamai gerai atspindi tokią situaciją. Tačiau ir šiuo atveju lentelėje *Vykdymai* stulpelį *Nr* galima laikyti pertekliniu, nors dažniausiai kiekviena lentelė turi raktą. Raktas yra būtinas kai, kitoje lentelėje norime apibrėžti išorinį raktą, nukreipiantį į šią lentelę. Lentelė *Vykdymai* yra pagrįsta, kai FP {*Projektas*, *Vykdytojas*} → {*Statusas*, *Valandos*} negalioja.

Nors abiejų lentelių (*Vykdymas* ir *Vykdymai*) struktūra yra labai panaši, jų savybės yra pakankamai skirtingos. Aplinkybės, įtakojančios ryšio realizavimą, ER modelyje *Darbai* atspindi nepakankamai. Tam mums pritrūko ER modelio raiškos priemonių. Panašias aplinkybes galima aprašyti ER schemas lydraštyje.

Kita vertus, kai lentelė *Vykdymai* labiau atitinka dalykinę sritį nei lentelė *Vykdymas*, t.y. kai negalioja FP $\{Projekto, Vykdytojas\} \rightarrow \{Statusas, Valandos\}$, vykdytojų dalyvavimą projektuose yra tikslingiau išreikšti ne ryšiu, bet esybe *Vykdymas*. Tokias aplinkybes vienareikšmiškai aprašanti ER schema yra pateikta 4.6 pav.



4.6 pav. E-R modelio *Darbai* schema. Vykdytojų dalyvavimas projektuose modeliuojamas esybe.

DB schema, atitinkanti ER schemą, pavaizduotą 4.6 pav., skirtą nuo pateiktosios 4.5 pav. tik vienos lentelės schema. Vietoje lentelės *Vykdymas* duomenų bazėje *Darbai* būtų lentelė *Vykdymai*. Realizuojant ER schemą nekyla neaiškumų, būdingų realizuojant N:M ryšius. Jei kiekvienas vykdytojas kiekviename projekte gali dalyvauti tik viename statuse ir vykdytojo indėlis projekte apibūdinamas bendru valandų skaičiumi, tai tokį uždavinį tikslinga vaizduoti ER schema, pavaizduota 4.4 pav., kitaip – 4.6 pav.

Atsižvelgdami į čia pateiktas pastabas, patikslinsime N:M ryšio realizavimo taisyklę. Ryšys N:M realizuojamas sudarant atskirą lentelę. Į lentelę įtraukiami abiejų lentelių, atitinkančių ryšį dalyvaujančios esybės, pirminiai raktai, kurie tampa išoriniais raktais. Į lentelę taip pat įtraukiami visi ryšio atributai. Lentelė gali neturėti nei vieno rakto. Tačiau dažniausiai lentelės pirminis raktas sudaromas vienu iš šių būdų:

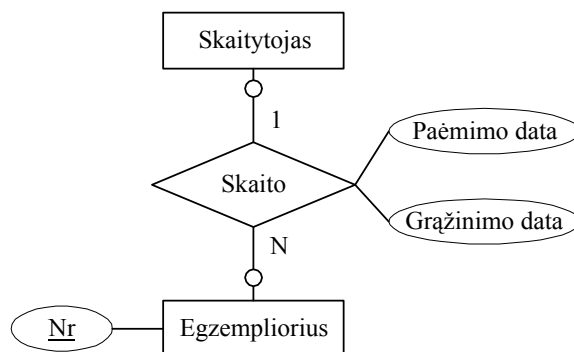
- lentelės pirminį raktą sudaro abiejų susietųjų lentelių pirminių raktų derinys;
- lentelė papildoma nauju stulpeliu, kuris tampa jos pirminiu (dirbtiniu) raktu.

Kai lentelė papildoma dirbtiniu pirminiu raktu, tenka spręsti, kaip parinkti šio stulpelio reikšmes. Dažniausiai tam naudojama natūraliųjų skaičių aibė. Kiekvieną kartą įvedant naują eilutę, parenkamas vis kitas natūrinis skaičius – eilutės numeris. Numerio parinkimas nesudaro sunkumų, nes SQL kalboje tam yra numatytos specialios priemonės, kurias aptarsime vėliau.

Papildomai panagrinėkime ER modelyje *Biblioteka* esantį ryšį *Skaityto*. Šis ryšys sieja dvi esybes: *Egzempliorius* ir *Skaitytojas*. Bibliotekos modelyje, kuris pavaizduotas 4.2 pav., ir realizuotas DB schema, pavaizduota 4.3 pav., yra stebimi tik šiuo metu paimtų egzempliorių skaitytojai. Gali būti prasminga prisiminti visus ar bent kelis paskutiniuosius egzempliorių skaitytojus. Pavyzdžiui, paaiškėjus, kad konkretus egzempliorius yra sugadintas, turint duomenis apie visus skaičiusius jį skaitytojus, galima ieškoti kaltininko. Žinoti ankstesnį skaitytoją naudinga ir tuomet, kai naujasis skaitytojas randa knygoje ankstesnio skaitytojo paliktą daiktą, pvz. svarbų dokumentą. Visų skaitytojų išiminimas taip pat yra naudingas statistiniuose tyrimuose ir kt.

ER modelyje *Biblioteka* egzemplioriaus paėmimo ir grąžinimo datos yra esybės *Egzempliorius* atributai. Vadinasi, kiekvienam egzemplioriui būdinga tik viena paėmimo ir viena grąžinimo datos bei vienas skaitytojas. Kadangi tarp lentelės *Egzemplioriaus* stulpelių galioja FP $Nr \rightarrow \{Paėmimo_data, Grąžinimo_data, ISBN, Skaitytojo_Nr\}$, tai jau sudaryta DB schema visiškai atitinka modelį.

Sudarydami bibliotekos modelį, paėmimo ir grąžinimo datas mes galėjome priskirti ne egzemplioriui, bet ryšiui *Skaito*. Taip pakeisto modelio ER diagramos fragmentas yra pavaizduotas 4.7 pav.



4.7 pav. Ryšys *Skaito* su egzemploriaus paėmimo ir grąžinimo datomis

4.3 pav. pavaizduota DB schema visiškai atitinka ir taip atnaujintą ER modelį *Biblioteka*. Taip sudarytą ryšį *Skaito*, kaip ir bet kuri kitą ryšį, galime realizuoti atskira lentele *Skaitymas*, įtraukiant į ją ryšio atributus atitinkančius stulpelius (*Paėmimo_data*, *Grąžinimo_data*) ir stulpelius *Egzemploriaus_Nr*, *Skaitytojo_Nr*, kurie atitinka susijusių su ryšiu lentelių *Egzempliorius* ir *Skaitytojas* pirminius raktus. Kadangi kiekvieną egzempliorių gali skaityti tik vienas skaitytojas, tai tokios lentelės pirminis raktas yra išorinis raktas nekreipiantis į lentelę *Egzempliorius*:

Skaitymas (*Egzemploriaus_Nr*, *Skaitytojo_Nr*, *Paėmimo_data*, *Grąžinimo_data*),
išoriniai raktai: *Egzemploriaus_Nr* nukreipia į *Egzempliorius*,
Skaitytojo_Nr nukreipia į *Skaitytojas*.

Lentelė *Egzempliorius* žymiai supaprastėja:

Egzempliorius(*Nr*, *ISBN*, *Skaitymas*),
išoriniai raktai: *ISBN* nukreipia į *Knyga*,
Skaitymas nukreipia į *Skaitymas*.

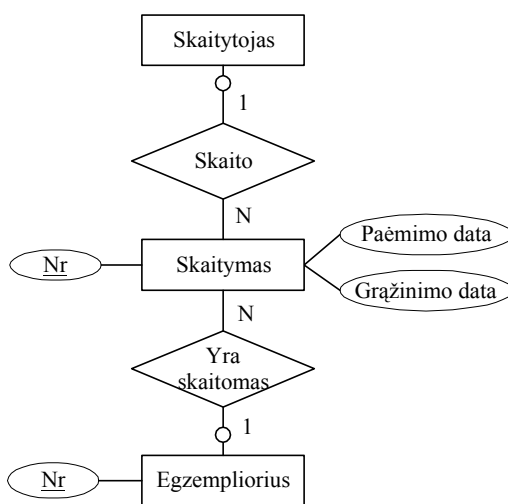
Tačiau, tokia lentelių *Skaitymas* ir *Egzempliorius* sandara nėra prasminga. Duomenų savybių prasme tokia sandara visiškai sutampa su ankstesniąja, kuri pateikta 4.3 pav., tačiau dabar mes turime vieną lentelę daugiau. Taip sudaryta DB tikriausiai užims daugiau vietos kompiuterio atmintyje. Tačiau yra ir teigiamų pastarosios sandaros bruožų. Jei didžioji dauguma egzempliorių bibliotekoje, dažniausiai, būna nepaimti, tai anksčiau sudarytoje lentelėje *Egzempliorius* (žr. 4.3 pav.) būtų daug NULL reikšmių – neefektyviai naudojami trys jos stulpeliai: *Skaitytojo_Nr*, *Paėmimo_data*, *Grąžinimo_data*. Naudojant naujesniąją DB sandarą, lentelėje *Skaitymas* NULL reikšmių visiškai nebus. Tiesiog lentelėje *Skaitymas* bus labai nedaug eilučių. Atsižvelgdamas į egzempliorių skaičių ir jų paėmimų dažnį, duomenų bazės administratorius turėtų įvertinti ir duomenų atnaujinimo pobūdį. Kai lentelės *Skaitymas* nėra, egzemploriaus paėmimas ir grąžinimas yra lentelės *Egzempliorius* reikšmių atnaujinimas. Kai naudojama lentelė *Skaitymas*, tai egzemploriaus paėmimas yra naujos eilutės įterpimas į lentelę *Skaitymas*, o grąžinimas – eilutės trynimasis. Tai pakankamai subtilūs duomenų bazės administravimo klausimai. Kiekvienu atveju yra tiek teigiamų, tiek ir neigiamų aspektų. Dažniausiai, kuo mažiau nereikalingų lentelių, tuo geriau.

Jeigu realizuodami ryšį *Skaito* atskira lentele, lentelėje *Skaitymas* apibrėžtąjį pirminį raktą *Egzemploriaus_Nr* pakeistume dirbtiniu raktu *Nr* – egzemploriaus paėmimo registracijos numeriu, tai pažeistume ryšio pobūdį. Tuomet vienam knygos egzemplioriui lentelėje *Skaitymas* galėtų būti keli skaitytojai, ryšys 1:N taptų ryšiu N:M.

Jei norime duomenų bazėje išsaugoti duomenis apie visus visų egzempliorių paėmimus, tai mums reikia labiau pakeisti modelį *Biblioteka*. Vienas iš būdų tai padaryti yra ryšio *Skaito* pavertimas N:M ryšiu. Tokį ryšį galima pavaizduoti ER schema, labai panašia į pateiktąją 4.7

pav. Pakanka linijos žymenį „1“ reikia pakeisti žymeniu „M“. Realizuodami tokį ryšį galime elgtis panašiai kaip jau aptarto N:M ryšio *Vykdytas* atveju. Jeigu N:M ryšiui *Skaito* sukurtume atskirą lentelę su raktu iš dviejų atributų: *Egzemploriaus_Nr*, *Skaitytojo_Nr*, tai suteiktume kitokią interpretaciją išsivaizduojamam paėmimų istorijos saugojimui. Tuomet, pavyzdžiui, joks skaitytojas negalėtų paimti tą patį egzempliorių du kartus. Tik dirbtinio rakto įvedimas atitiktų visos paėmimų istorijos išsiminimą.

ER modelis pakankamai tiksliai išreikštų siekį saugoti „istoriją“, jeigu ryšį *Skaito* pakeistume esybe *Skaitymas* suteikdami jai raktinį atributą *Nr*. Toks atributas yra prasmingas, jis atitinka paėmimo registracijos eilės numerį galimame egzempliorių paėmimo žurnale ar kartotekoje. Atnaujintos ER schemas fragmentas pavaizduotas 4.8 pav.



4.8 pav. Esysbė *Skaitymas* ir jos ryšiai

Atnaujinto ER modelio *Biblioteka* esybę *Skaitymas* (4.8 pav.) bei ryšius *Skaito* ir *Yra skaitomas* galima realizuoti atskira lentele su dviem išoriniais raktais:

Skaitymas(*Nr*, *Egzemploriaus_Nr*, *Skaitytojo_Nr*, *Paėmimo_data*, *Gražinimo_data*),

išoriniai raktai: *Egzemploriaus_Nr* nukreipia į *Egzemploriaus*,

Skaitytojo_Nr nukreipia į *Skaitytojas*.

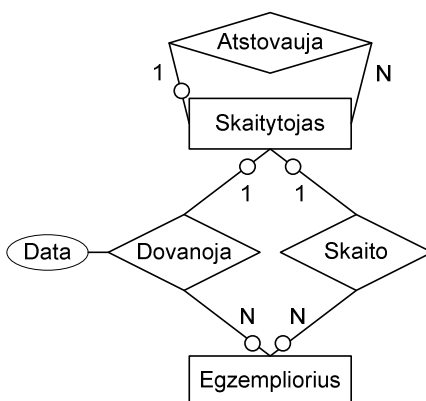
Lentelės *Egzempliorius*(*Nr*, *ISBN*, *Skaitymas*) sandara išlieka nepakitęs. Taip sudarę duomenų bazę, lentelėje *Skaitymas* galėsime išiminti duomenis apie visus visų egzempliorių paėmimus.

Jau minėjome, kad ryšys gali sieti esybę su ja pačia. Tarkime, kiekvienam skaitytojui yra priskiriamas kitas bibliotekos skaitytojas, į kurį bibliotekos darbuotojai gali kreiptis, jei, pavyzdžiui, skaitytojas vengia grąžinti knygą. Studentui gali būti priskirtas jo grupės seniūnas, o darbuotojui – padalinio vadovas. Tai - ryšys esybėje. 4.9 pav. pavaizduotas vienanaris ryšys *Atstovauja*, kurio dalyvis yra esybė *Skaitytojas*. Ryšys *Atstovauja* skaitytojui – atstovui priskiria jo atstovaujamus skaitytojus, arba atvirkščiai. Tarus, kad kiekvienas skaitytojas būtinai turi savo atstovą, bet ne kiekvienas skaitytojas atstovauja kurį nors skaitytoją, ryšys *Atstovauja* yra besąlygiškas iš vienos pusės ir sąlygiškas iš kitos.

Naująjį 1:N ryšį *Atstovauja* reliacinėje schemoje galima pavaizduoti lentelės *Skaitytojas* papildomu stulpeliu *Atstovas*. Šis stulpelis taip pat yra išorinis raktas į tą pačią lentelę *Skaitytojas*. Jau sudarytos lentelės *Skaitytojas* atnaujinta schema yra

Skaitytojas(*Nr*, *AK*, *Vardas*, *Pavardė*, *Gatvė*, *Namas*, *Butas*, *Atstovas*),

išorinis raktas: *Atstovas* nukreipia į *Skaitytojas*.



4.9 pav. Dvinaris ryšys *Atstovauja* ir vienanaris ryšys *Dovanoja*, papildantys ryšį *Skaito*.

Tarp tų pačių esybių gali būti ir keli ryšiai. Tarkime, skaitytojai gali dovanoti bibliotekai knygų egzempliorius. Taip tarp esybių *Skaitytojas* ir *Egzempliorius*, greta jau turimo ryšio *Skaito*, atsiranda ryšys *Dovanoja*, kuriuo egzempliorius siejamas su jį dovanojusi skaitytoju. Tarkime, šis ryšys papildomai charakterizuojamas dovanojimo data. Naujasis ryšys yra sąlygiškas iš abiejų pusių. Jis pavaizduotas 4.9 pav.

Reliacinėje DB schemoje ryšį *Dovanoja* atitinka du papildomi stulpeliai lentelėje *Egzempliorius*. Vienas iš jų - *Dovanotojas* yra egzempliorių dovanojusio skaitytojo numeris, o kitas - *Dovanojimo_data*. Papildžius ER schemą *Biblioteka* (4.2 pav.) ryšiu *Dovanoja* (4.9 pav.), atnaujiname lentelės *Egzempliorius* reliacinę schemą:

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

išoriniai raktai: *ISBN* nukreipia į *Knyga*,
Skaitytojo_Nr nukreipia į *Skaitytojas*,
Dovanotojas nukreipia į *Skaitytojas*.

Dabar lentelėje *Egzempliorius* yra trys išoriniai raktai, du iš kurių nukreipia į tą pačią lentelę *Skaitytojas*.