

U.Y. Saidov, B.I. Navruzov, H.U. Sultonov

DASTURIY TA'MINOT ARXITEKTURASI

O'quv qo'llanma





O'ZBEKISTON RESPUBLIKASI
OLIY TA'LIM, FAN VA INNOVATSIYALAR VAZIRLIGI
BUXORO DAVLAT UNIVERSITETI

U.Y. Saidov, B.I. Navruzov, H.U. Sultonov

DASTURIY TA'MINOT
ARXITEKTURASI
O'QUV QO'LLANMA

“KAMOLOT” nashriyoti
Buxoro-2024

UO‘K: 004.273

KBK: 85.11

S16

Saidov Umed Yusuf o‘g‘li, Navruzov Baxtiyor Ixtiyorovich, Sultonov Humoyun Ulug‘murodovich. / [Matn]: o‘quv qo‘llanma / U.Y. Saidov, B.I. Navruzov, H.U. Sultonov - Buxoro : “BUXORO DETERMINANTI” MCHJning Kamolot nashriyoti, 2024. - 192 b.

ISBN: 978-9910-729-55-3

Taqrizchilar:

BuxMTI p.f.d. (DSc), professor – F.R.Muradova
BuxDU dotsenti – G.I.Atayeva

Ushbu o‘quv qo‘llanma Oliy ta’lim, fan va innovatsiyalar vazirligi Buxoro davlat universitetining 2024-yil 2-iyuldagi 452-sonli buyrug‘iga asosan nashrga ruxsat berildi.

© “KAMOLOT” nashriyoti

© Saidov Umed Yusuf o‘g‘li

© Navruzov Baxtiyor Ixtiyorovich

© Sultonov Humoyun Ulug‘murodovich

MUNDARIJA

KIRISH	4
1-MAVZU. MURAKKAB TIZIMLARNI YARATISHDA TIZIMLI YONDASHUV	6
2-MAVZU. DASTURIY TA'MINOT LOYIHALARNI BIZNES-JARAYONI MODEL. MODELLASHTIRISH NOTATSIYASI	13
3-MAVZU. DASTURIY TA'MINOT LOYIHALARNI YARATISHGA QO'YILGAN TALABLAR	27
4-MAVZU. DASTURIY TA'MINOT LOYIHALARNI YARATISHNING HUQUQIY, NORMATIV VA METODIK TAMINOTI. DTLARNI YARATISHNING XALQARO STANDARTLARI.....	31
5-MAVZU. BIZNES JARAYONLARI VA JARAYONLI BOSHQARUV	36
6-MAVZU. DASTURIY TA'MINOT HAYOTIY SIKLI ASOSIY JARAYONLARI	47
7-MAVZU. VIZUAL MODELLASHTIRISH IDEF METODOLOGIYASI	64
8-MAVZU. EPS METODOLOGIYASI	69
9-MAVZU. BPMN 2.0 SPESIFIKATSIYASI	81
10-MAVZU. BPMN NOTATSIYASI SUBKLASSLARI	107
11-MAVZU. BIZNES JARAYONLAR DIAGRAMMALARNING KATEGORIYALARI.....	141
12-MAVZU. OPERATSIYALAR	147
13-MAVZU. OPERATSIYALAR MARKERLARI	159
14-MAVZU. JARAYON MODEL.....	165
15-MAVZU. MANTIQIY OPERATORLAR.....	171
GLOSSARIY	190
FOYDALANILGAN ADABIYOTLAR RO'YXATI	191

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
ПРЕДМЕТ 1. СИСТЕМАТИЧЕСКИЙ ПОДХОД ПРИ СОЗДАНИИ СЛОЖНЫХ СИСТЕМ	6
ПРЕДМЕТ 2. МОДЕЛЬ БИЗНЕС-ПРОЦЕССА ПРОГРАММНЫХ ПРОЕКТОВ. МОДЕЛИРОВАНИЕ ОБОЗНАЧЕНИЙ	13
ПРЕДМЕТ 3. ТРЕБОВАНИЯ К СОЗДАНИЮ ПРОГРАММНЫХ ПРОЕКТОВ	27
ПРЕДМЕТ 4. ПРАВОВОЕ, НОРМАТИВНОЕ И МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ СОЗДАНИЯ ПРОГРАММНЫХ ПРОЕКТОВ. МЕЖДУНАРОДНЫЕ СТАНДАРТЫ СОЗДАНИЯ ДЦ	31
ПРЕДМЕТ 5. БИЗНЕС-ПРОЦЕССЫ И УПРАВЛЕНИЕ ПРОЦЕССАМИ	36
ПРЕДМЕТ 6. ПРОЦЕССЫ ЖИЗНЕННОГО ЦИКЛА БАЗОВОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	47
ПРЕДМЕТ 7. МЕТОДОЛОГИЯ ВИЗУАЛЬНОГО МОДЕЛИРОВАНИЯ IDEF	64
ПРЕДМЕТ 8. МЕТОДОЛОГИЯ ЭПС	69
ПРЕДМЕТ 9. СПЕЦИФИКАЦИЯ VRMN 2.0	81
ПРЕДМЕТ 10. ПОДКЛАССЫ ОБОЗНАЧЕНИЯ VRMN	107
ТЕМА 11. КАТЕГОРИИ ДИАГРАММ БИЗНЕС-ПРОЦЕССОВ	141
ТЕМА 12. ОПЕРАЦИИ	147
ТЕМА 13. ЗНАКИ ОПЕРАЦИЙ	159
ТЕМА 14. МОДЕЛЬ ПРОЦЕССА	165
ТЕМА 15. ЛОГИЧЕСКИЕ ОПЕРАТОРЫ	171
ГЛОССАРИЙ	190
СПИСОК ЛИТЕРАТУРЫ	191

KIRISH

Dasturiy ta'minot arxitekturasini zamonaviy dasturiy ta'minot tizimlarini loyihalash, ishlab chiqish va texnik xizmat ko'rsatishda juda muhim ahamiyatga ega. U tizimning tuzilishi va xatti-harakatlarini aniqlovchi asosiy tamoyillar va qarorlarni o'z ichiga oladi. Ushbu qo'llanma dasturiy ta'minot arxitekturasining asosiy tushunchalari, tamoyillari va amaliyotlari bilan tanishtiradi, shuningdek, arxitektura dizaynini amalga oshirish bo'yicha qo'llanma sifatida xizmat qiladi.

Dasturiy ta'minot arxitekturasini - bu dasturiy ta'minot tizimining yuqori darajadagi tuzilishi. U tizim komponentlari, ularning o'zaro bog'liqligi va o'zaro ta'siri, hamda tizimning xulq-atvorini aniqlovchi qoidalar va tamoyillarni o'z ichiga oladi. Bu tushuncha tizimni modullarga ajratish, ularning o'zaro aloqasi va interfeyslarini belgilash orqali, tizimning murakkabligini boshqarishga yordam beradi.

Dasturiy ta'minot arxitekturasini tizimning sifat ko'rsatkichlariga, masalan, ishlash qobiliyati, xavfsizlik, kengaytirish osonligi va texnik xizmat ko'rsatish osonligiga katta ta'sir ko'rsatadi. To'g'ri loyihalangan arxitektura tizimni yanada barqaror, ishonchli va ko'proq moslashtiriladigan qiladi. Shuningdek, u loyiha jamoasi o'rtasida yaxshiroq muloqot va tushunishni ta'minlaydi, chunki barcha ishtirokchilar tizimning umumiy tuzilishi va maqsadlari haqida umumiy tasavvurga ega bo'ladilar.

Ushbu o'quv qo'llanma 60610100 – Dasturiy injiniring bakalavriat ta'lim yo'nalishida tahsil olayotgan talabalarning o'zlashtirishi lozim bo'lgan bilimlari asosida tuzilgan bo'lib, talabalarning egallashi kerak bo'lgan bilimlar va ko'nikmalar mazmunini o'z ichiga oladi.

“Dasturiy ta'minot arxitekturasini” o'quv qo'llanmasi kirish, 15 ta mavzu, mundarija hamda glossariy, foydanilgan adabiyotlar va internet resurlari ro'yxatidan iborat.

O'quv qo'llanmaning umumiy hajmi 192 betdan iborat bo'lib, unda dasturiy ta'minot arxitekturasini, modellar va ularning turlari, operatsiyalar, operatsiya markerlari kabi ma'lumotlar berilgan.

1-MAVZU. MURAKKAB TIZIMLARNI YARATISHDA TIZIMLI YONDASHUV

Reja:

1. Axborot tizimlarini yaratishda qo'llaniladigan maxsus dasturiy paketlar
2. Tizimli yondashuv tamoyillari
3. Murakkab tizim yondashuvlari uchun loyihalashning umumiy xususiyatlari
4. Murakkab tizimlarni modellashtirishda blok-ierarxik yondashuv

Ishlab chiqarish jarayonida zaxiralarning aniq darajasini saqlab qolishdan tortib ishlab chiqarish funktsiyalarini nazorat qilishgacha bo'lgan ko'plab vazifalar mavjud. Ishlab chiqaruvchilar ishlab chiqarishni qo'lda rejalashtirish va sotib olish faoliyatini boshqarish uchun ishlatilgan. Bu sanoatda kompyuterlar hukmronlik qilishdan oldin edi. Ammo bugungi kunda ushbu murakkab vazifalarni hal qilishga yordam beradigan dasturiy echimlar mavjud. MRP tizimlarining birinchi avlodi yoki MRP I xom ashyo, ehtiyot qismlar, yig'ilishlar va mahsulotlar o'rtasidagi munosabatlarni aniqlash uchun mo'ljallangan. Bular MRP asosi bo'lgan materiallar ro'yxatini tashkil qiladi. MRP ishlab chiqarish jarayonida zarur bo'lgan materiallarni aniqlashga qaratilgan.

MRP ning asosiy funksiyalari quyidagilardan iborat:

- Minimal zarur zaxiralarni saqlash uchun
- Ishlab chiqarish uchun xom ashyo mavjudligini ta'minlash
- Tayyor mahsulotlarni o'z vaqtida yetkazib berishga tayyor bo'lishini ta'minlash
- Ishlab chiqarish faoliyatini, etkazib berish jadvallarini va sotib olish faoliyatini rejalashtirish

MRP ning afzalliklari:

- **Optimallashtirilgan inventarizatsiya.** MRP ishlab chiqarish jarayoni uchun zarur bo'lganda materiallar mavjudligini ta'minlaydi. U xatolarga moyil bo'lgan qo'lda yozishni tugatish uchun jarayonlarni avtomatlashtiradi. Va bu ishlab chiqarish hududiga mahsulotni yetkazib berish vaqti kelganini bilish imkonini beradi.

- **Kamroq xarajatlar.** MRP bilan inventar optimal darajada saqlanadi. Bu qo'lda ortiqcha zaxiralarni saqlash zarurati bilan bir qatorda so'nggi daqiqada favqulodda ta'minot buyurtmalarini

joylashtirish zaruratini yo‘q qiladi. MRP xarajatlarni kamaytirishga yordam beradi va daromadni oshiradi.

• **Hosildorlikning oshishi.** Kerakli materiallar har doim mavjud bo‘lsa, ishlab chiqarish samarali bo‘ladi. Xodimlarning ishlamay qolishi kamroq bo‘ladi va ish qobiliyati yaxshilanadi. Ishlab chiqaruvchilar o‘zlarining ta‘minot zanjirini optimallashtirishga qodir bo‘lishlari kerak. Resurslarni samarali boshqarish kam inventar xarajatlarini va o‘z vaqtida yetkazib berishni ta‘minlaydi. Resurslarni nazorat qilmaslik biznesning obro‘sigacha putur etkazishi mumkin. **ERP nima?** MRP II ajoyib funksiyalarga ega. Ishlab chiqaruvchilar uchun juda foydali bo‘lgan ko‘plab muhim funktsiyalar hali ham etishmayapti. ERP MRP II ning barcha asosiy biznes funktsiyalari va jarayonlarini o‘z ichiga olgan kengaytirilgan versiyasidir. U ishlab chiqarish va inventarizatsiya, moliya va buxgalteriya hisobi, savdo, CRM, HR va boshqalarni boshqaradi. ERP - bu har bir bo‘limning real vaqt rejimida ko‘rinishini ta‘minlaydigan biznesni boshqarish vositalarining integratsiyalashgan to‘plami. Barcha harakatlar ERP tizimidagi yagona ma‘lumotlar bazasida ko‘rinadi. Bu hammaga bir xil ma‘lumotdan ishlash imkonini beradi.

ERP ning funktsionalligi nimadan iborat?

- Moliya va buxgalteriya
- Inson kapitalini boshqarish
- Xarid qilish
- Ishlab chiqarish
- Buyurtmani boshqarish
- Inventarizatsiyani boshqarish
- Omborni boshqarish
- Ta‘minot zanjiri boshqaruvi
- Mijozlar bilan munosabatlarni boshqarish
- Elektron tijorat

ERP xarajatlarni kamaytirishi, rentabellikni oshirishi, ichki hamkorlikni yaxshilashi va mijozlar qoniqishini oshirishi mumkin. Bu korxonalarga muhim funktsiyalarni boshqarishda yordam beradi. Quyidagi diagrammada MRP, MRP II va ERP o‘rtasidagi farqlar keltirilgan.

1C: Korxona platformasi

1C: Enterprise - bu kompaniyaning operatsion faoliyatini avtomatlashtirish uchun universal bulutli va mahalliy dasturlar tizimi. 1C: Korxona bugungi biznesning turli ehtiyojlarini qondirish imkoniyatlarining kengligiga ega. Bunga "konfiguratsiya" orqali erishiladi - tizimni kompaniyalarning o'ziga xos ehtiyojlari va ularning biznes jarayonlari asosida sozlash qobiliyati.

1C: Enterprise Platformasi quyidagi afzalliklarni taklif etadi:

- Korporativ dasturiy ta'minot tizimlarining texnologik murakkabligi va ishlash muammolarini keskin kamaytiradi.
- Biznes ilovalarini ishlab chiqish, sozlash va qo'llab-quvvatlashni tezlashtiradi va standartlashtiradi.
- UI va yechim kodining ochiqqligini ta'minlaydi, bu biznes jarayonlarini yaxshiroq tushunish va o'zgartirish imkonini beradi.
- Mavjud 1C ilovalari va uchinchi tomon tizimlari bilan integratsiyaga tayyor.
- Veb-xizmatlarni, ODBC, COM va boshqalarni qo'llab-quvvatlaydi.
- Siz tanlagan arxitekturani qo'llab-quvvatlaydi: Windows/Linux, MS SQL, PostgreSQL, IBM DB2 va Oracle DB.
- Mahalliy yoki boshqariladigan hosting veb-ga asoslangan yetkazib berishni, shuningdek, Internet, planshet, MacOS yoki Windows mijozini o'z ichiga oladi.

Tizimli yondashuv tamoyillari

Murakkab tizimlarni kompyuterda modellashtirishning asosiy g'oyalari va tamoyillari tizimli yondashuvda ifodalangan. Loyihalash obyektlarini ularning matematik modellari yordamida o'rganish tizimli yondashuvning asosiy mohiyati deb hisoblanadi.

Tizimli yondashuvning shunday tamoyillari mavjud.

- *Ierarxiya* - har bir quyi tizim yoki element tizim sifatida qaralishi mumkin.
- *Strukturaviylik* - uning elementlari orasidagi kommutatsiya aloqalari yordamida tizimni tavsiflash qobiliyati.
- *O'zaro bog'liqlik* - tizim xususiyatlarining faqat atrof-muhit bilan o'zaro ta'sirida namoyon bo'lishi.
- *Tavsifning ko'pligi* - tizimni turli xil matematik modellar asosida tavsiflash.

- *O'rganilayotgan tizimning yaxlitligi* bu butun tizimning qismlarini tahlil qilish va bilish asosida butun tizimning xususiyatlarini o'rganishdir.

Tizimli yondashuvning asosiy umumiy printsiplari - hodisa yoki murakkab tizim qismlarini ularning o'zaro ta'sirini hisobga olgan holda ko'rib chiqishdir. Tizimli yondashuv tizim tuzilishini aniqlash, munosabatlarni tiplashtirish, atributlarni aniqlash va tashqi muhit ta'sirini tahlil qilishni o'z ichiga oladi. Tizimli yondashuvning eng muhim vazifalaridan biri matematik modellarning turi, soni, murakkablik darajasi va tasvirlash shaklini tanlashdir.

Tizimli yondashuvni talqin qilish va konkretlashtirish boshqa nomlar bilan bir qator taniqli yondashuvlarda amalga oshiriladi. Bular strukturaviy, blok-ierarxik, obyektga yo'naltirilgan yondashuvlardir.

Murakkab tizim yondashuvlari uchun loyihalashning umumiy xususiyatlari

Murakkab tizimlarni loyihalashning barcha yondashuvlari quyidagi xususiyatlar bilan ham tavsiflanadi.

- Dizayn topshiriqlari va hujjatlarining parchalanishi, bosqichlar, bosqichlar, loyihalash tartib-qoidalarini taqsimlash bilan ifodalangan loyihalash jarayonini *strukturalash*. Ushbu struktura blok-ierarxik dizayn yondashuvining mohiyatidir.

- *Dizaynning takroriy* xususiyati.
- Dizayn yechimlari va dizayn vositalarini *tiplashtirish* va *unifikatsiya* qilish.

Murakkab tizimlarni modellashtirishda blok-ierarxik yondashuv

Tabiatda ham, texnikada ham murakkab tizimlarning aksariyati ierarxik ichki tuzilishga ega. Buning sababi shundaki, odatda murakkab tizimlar elementlarining ulanishlari ham turi, ham kuchi jihatidan farq qiladi, bu bizga ushbu tizimlarni o'zaro bog'liq bo'lgan ma'lum bir quyi tizimlar to'plami sifatida ko'rib chiqishga imkon beradi. Bunday quyi tizimlar elementlarining ichki aloqalari quyi tizimlar orasidagi bog'lanishlarga qaraganda kuchliroqdir. Butun tizimning xatti-harakati, odatda, alohida qismlarning xatti-harakatlariga qaraganda murakkabroq va kuchli ichki aloqalar tufayli tizimning xususiyatlari, asosan, uning qismlari emas, balki uning qismlari o'rtasidagi munosabatlarga bog'liq.

Tizimni loyihalash jarayonida tizim haqida ma'lum g'oyalar shakllanadi, bu uning muhim xususiyatlarini turli darajadagi tafsilotlar bilan aks ettiradi. Ushbu vakilliklarda tarkibiy qismlarni - dizayn darajalarini ajratish mumkin. Muayyan ierarxik darajadagi model tillari, muammo bayonlari, tavsiflash usullari to'plami odatda dizayn darajasi deb ataladi. Dizayn darajalari gorizontal (ierarxik) va vertikal dizayn darajalariga bo'linadi. Vertikal dizayn darajalari *aspektlar deb ham ataladi*. Dizayn jarayonida har bir obyekt, qoida tariqasida, bir necha tomondan ko'rib chiqilishi kerak. Dizayn obyektiga turli xil qarashlar dizayn aspektlari deb ataladi. Har bir aspekt doirasidagi murakkab obyektlar haqidagi tasvirlar ierarxik darajalarga (abstraksiya darajalariga) bo'linishi kerak. Dizaynning vertikal darajalari orasida quyidagilar ajralib turadi.

- *Funksional dizayn* : tuzilmaning asosiy xususiyatlarini, ishlash tamoyillarini, yaratilayotgan obyektlarning eng muhim parametrlari va xususiyatlarini aniqlash.

- *Algoritmik loyihalash* : EHM va hisoblash tizimlarining (XS) ishlashi uchun algoritmlarni ishlab chiqish, ularning umumiy tizimi va amaliy dasturlarini yaratish.

- *Strukturaviy loyihalash*: funksional loyihalash natijalarini loyihalashtirish (shakllar, materiallarni tanlash, qismlarni joylashtirish va boshqalar).

- *Texnologik loyihalash* : dizayn loyihasi natijalarini amalga oshirish (mahsulot ishlab chiqarish uchun texnologik jarayonlarni yaratish).

Shu bilan birga, har bir quyi tizimning tavsifi juda batafsil bo'lmasligi kerak, chunki bu haddan tashqari noqulay tavsiflarga va paydo bo'ladigan dizayn muammolarini hal qilishning imkonsizligiga olib keladi.

Gorizontal (ierarxik) darajalarni taqsimlash dizaynga blok-ierarxik yondashuv asosida yotadi. Yuqori darajada, loyihalashtirilayotgan tizimning faqat eng umumiy xususiyatlari va xususiyatlarini aks ettiruvchi eng kam batafsil tasvirdan foydalaniladi. Keyingi darajalarda tavsifning batafsil darajasi oshadi va shu bilan birga tizimning alohida bloklari ko'rib chiqiladi, lekin ularning har biriga qo'shnilarining ta'sirini hisobga olgan holda. Ushbu yondashuv har bir ierarxik darajadagi maqbul murakkablikdagi vazifalarni shakllantirishga imkon beradi, ularni mavjud dizayn vositalaridan

foydalangan holda hal qilish mumkin. Darajaga bo'linish shunday bo'lishi kerakki, har qanday darajadagi blok uchun hujjatlar ko'rinadigan va bir kishi tomonidan qabul qilinadi. Boshqacha qilib aytadigan bo'lsak, blok-ierarxik yondashuv - bu katta o'lchamdagi murakkab muammolarni ketma-ket va (yoki) parallel ravishda kichik o'lchamli muammolar guruhlariga bo'lishga asoslangan dekompozitsion yondashuv (uni diaoptik deb ham atash mumkin), bu esa sezilarli darajada kamaytiradi. foydalaniladigan hisoblash resurslariga yoki hal qilish vaqtiga qo'yiladigan talablar. Parchalanishda alohida qismlar orasidagi bog'lanishlar qismlar ichidagi elementlarning ulanishlariga qaraganda zaifroq bo'lishi kerakligi hisobga olinadi. Bundan tashqari, ishlab chiqilayotgan obyektni olingan qismlardan yig'ish imkoniyatiga ega bo'lish uchun parchalanish jarayonida qismlar orasidagi barcha turdagi bog'lanishlarni aniqlash kerak. Juda murakkab obyektlarni yaratishda parchalanish jarayoni takrorlanadi: har bir blok, o'z navbatida, siz nisbatan oson ishlab chiqiladigan bloklarni olmaguningizcha qismlarga bo'linadi. Rivojlanishning ushbu usuli incremental detailing deb ataladi.

Bundan tashqari, parchalanish jarayonida ular umumiy asosda ishlab chiqilishi mumkin bo'lgan o'xshash bloklarni aniqlashga harakat qilishlari ham muhimdir. Shunday qilib, yuqorida aytib o'tilganidek, ular kodlarning takrorlanuvchanligi darajasini oshirishni va shunga mos ravishda ishlab chiqish xarajatlarini kamaytirishni ta'minlaydi.

Parchalanish natijasi odatda ierarxiya diagrammasi sifatida taqdim etiladi, uning pastki darajasida nisbatan oddiy bloklar va yuqori qismida - ishlab chiqiladigan obyekt joylashgan. Har bir ierarxik darajada bloklarning tavsifi ahamiyatsiz tafsilotlardan mavhumlashtirib, ma'lum darajada tafsilotlar bilan amalga oshiriladi. Binobarin, har bir daraja uchun ular har bir blok tomonidan bajariladigan jarayonlarning mohiyatini aks ettiruvchi o'z hujjat shakllari va o'z modellaridan foydalanadilar. Shunday qilib, umuman obyekt uchun, qoida tariqasida, faqat eng umumiy talablarni shakllantirish mumkin va quyi darajadagi bloklar, ular haqiqatda ishlaydigan obyektga yig'ilishi uchun aniqlanishi kerak. Boshqacha qilib aytganda, blok qanchalik katta bo'lsa, uning tavsifi shunchalik mavhum bo'lishi kerak. Agar ushbu printsipga rioya qilinsa, ishlab chiquvchi loyihani tushunish qobiliyatini saqlab qoladi va shuning uchun har bir bosqichda eng to'g'ri qarorlarni qabul qilishi mumkin, bu

mahalliy optimallashtirish deb ataladi (obyekt xususiyatlarini global optimallashtirishdan farqli o'laroq, bu har doim ham mumkin emas. haqiqatan ham murakkab obyektlar). Demak, blok-ierarxik yondashuv parchalanish va ierarxik tartiblashtirishga asoslangan. Quyidagi tamoyillar ham muhim rol o'ynaydi.

- *Konsistensiya* - elementlarning o'zaro muvofiqligini nazorat qilish.
- *To'liqlik* - keraksiz elementlarning mavjudligini nazorat qilish.
- *Rasmiylashtirish* - uslubiy yondashuvning qat'iyligi.
- *Takroriylik* - xarajatlarni kamaytirish va rivojlanishni tezlashtirish uchun bir xil bloklarni ajratish zarurati.

- *Lokal optimallashtirish* - ierarxiya darajasida optimallashtirish.

Blok-ierarxik yondashuv murakkab tizimlarni amalga oshirishga qo'shimcha ravishda:

- butun tizim va sog'lig'ini tekshirishni soddalashtiradi va alohida bloklar;
- interfeyslarni saqlab qolgan holda ishonchsiz bloklarni almashtirish kabi tizimlarni yangilash imkoniyatini beradi.

Nazorat savollari

1. Tizimli yondashuvning asosiy tamoyillari nimalardan iborat?
2. Murakkab tizimlarni loyihalash jarayonida tizimli yondashuvning afzalliklari nimalardan iborat?
3. Murakkab tizimlarning komponentlarini qanday aniqlash mumkin?
4. Tizim arxitekturasini yaratishda qanday metodlardan foydalaniladi?
5. Tizimli yondashuvda model va simulyatsiyaning roli qanday?
6. Tizimlarning funksional va nosimmetrik xususiyatlarini qanday tahlil qilish mumkin?
7. Tizim integratsiyasi va sinergetik effekti nima va qanday amalga oshiriladi?
8. Murakkab tizimlarni boshqarishda qanday strategiyalar qo'llaniladi?
9. Tizimlarning tahlili va loyihalashida qanday vositalar va dasturiy ta'minotlardan foydalaniladi?
10. Murakkab tizimlarni yaratishda resurslarni boshqarish qanday amalga oshiriladi?

2-MAVZU. DASTURIY TA'MINOT LOYIHALARNI BIZNES-JARAYONI MODEL. MODELLASHTIRISH NOTATSIYASI

Reja:

1. Axborot tizimlari biznes-jarayonlarini modellashtirish
2. modellashtirish tillari
3. Modellashtirish notatsiyasi

Bugungi kunda eng mashhur UML, BPMN modellashtirish belgilari va shuningdek, IDEF0, IDEF1 va DFD tizimli usullari nima uchun hali ham dolzarbligini ko'rib chiqamiz.

Biznes modellashtirish notatsiyasi - bu mavzu sohasining asosiy tushunchalarini va ularning munosabatlarini tavsiflash imkonini beradigan jarayonlar yoki tizimlarni tavsiflash uchun foyalaniladigan grafik elementlar va belgilar tizimidir.

Biznes-jarayonlarni tavsiflash va axborot tizimlarini loyihalashda ko'plab belgilar qo'llaniladi, masalan, UML standarti (Unified Modeling Language) dasturiy ta'minotni ishlab chiqishda obyektning modellashtirish uchun 12 turdagi diagrammalarni o'z ichiga oladi.

Notatsiya - bu muayyan metodologiyada qabul qilingan belgilar (belgilar) va ulardan foydalanish qoidalarini tizimidir.

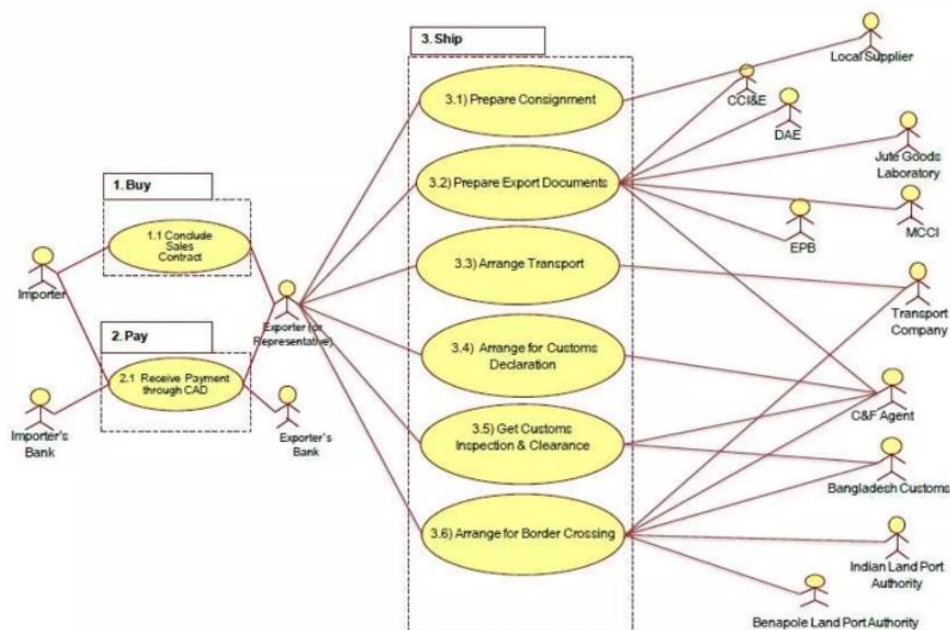
Notatsiya talablari:

- **oddiylik** - murakkabdan oddiy belgi afzalroq;
- **ko'rinish** - hech bo'lmaganda asl nusxaga uzoqroq bo'lsada o'xshashlik;
- **individuallik** - boshqa belgilardan etarli farq;
- **o'ziga xoslik** - turli xil obyektlarni bitta belgi bilan belgilash mumkin emas;
- **aniqlik** - modeldan foydalanishning aniq qoidalarini;

Obyektga yo'naltirilgan UML modellashtirish tili

UML (Unified Modeling Language - Birlashtirilgan modellashtirish tili) - bu axborot tizimlari (AT) modellarini yaratish uchun ularni obyektga yo'naltirilgan dasturlar ko'rinishida amalga oshirish uchun ishlab chiqilgan. Murakkab tizim modellaridagi barcha diagrammalar – maxsus grafik tuzilmalar (diagrammalar, grafiklar) ko'rinishida yaratilgan.

UseCase - Foydalanish diagrammasi - foydalanuvchining tizim bilan o'zaro ta'sirining grafik tasvirini izohlaydi.



2.1-rasm

Biznes modellashtirish - bu tashkilot faoliyatini rasmiylashtirish va optimallashtirish maqsadida uning turli biznes modellarini (strategiya, biznes jarayonlari, tashkiliy tuzilma, sifat va boshqalar) ishlab chiqish va amalga oshirish jarayoni. Biznes modelining ta'rifi darhol o'zini ko'rsatadi.

Biznes modeli - bu tashkilot faoliyatining ma'lum bir tomoni yoki sohasining rasmiylashtirilgan tavsifi (masalan, grafik).

Biznes modellarini rivojlantirishning 4 ta asosiy usuli mavjud. Biz ularni biznes modellarini qurish va ulardan foydalanish samaradorligi darajasining kamayish tartibida sanab o'tamiz.

• Biznesni modellashtirish bo'yicha ixtisoslashtirilgan dasturiy mahsulotning yozuvlarida (qoidalarida): grafikalar, jadvallar va matnlarning kombinatsiyasi. Batafsil ma'lumot uchun [2] 8-bobga qarang.

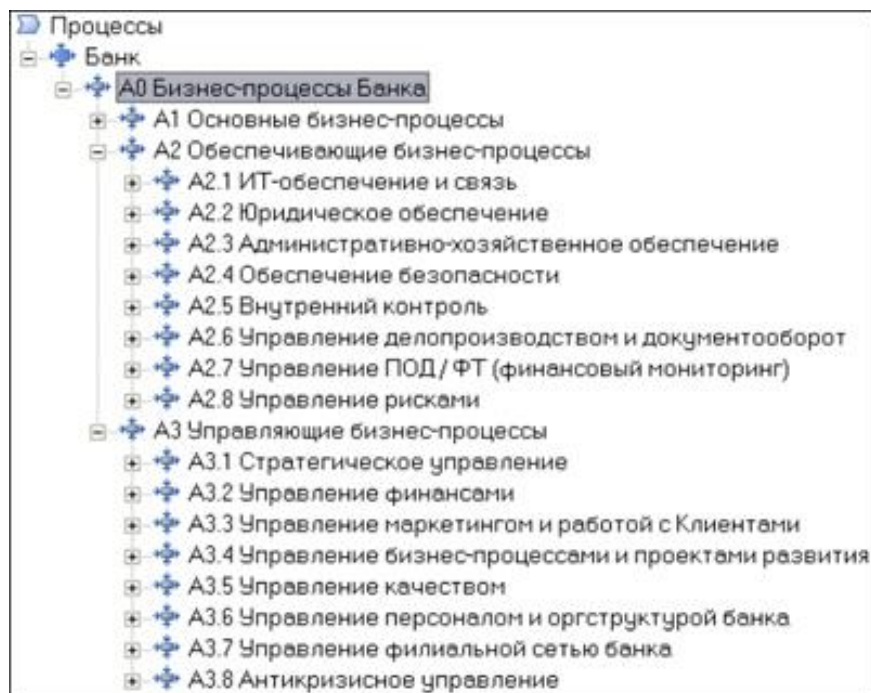
- Grafik: daraxt, blok-sxema, texnologik xarita va boshqalar.
- Jadval.
- Matn.

Ko'pgina tashkilotlar biznesni modellashtirish bilan shug'ullanadi, ammo ularning har biri bu sohada rivojlanishning turli bosqichlarida. Kimdir allaqachon keng qamrovli biznes modelini ishlab chiqqan va faol foydalanmoqda (tashkilotning barcha faoliyatini tavsiflovchi modellar, hujjatlar va tizimlar to'plami). Ba'zi odamlar faqat bir nechta biznes jarayonlari uchun grafik modellar va qoidalarga ega.

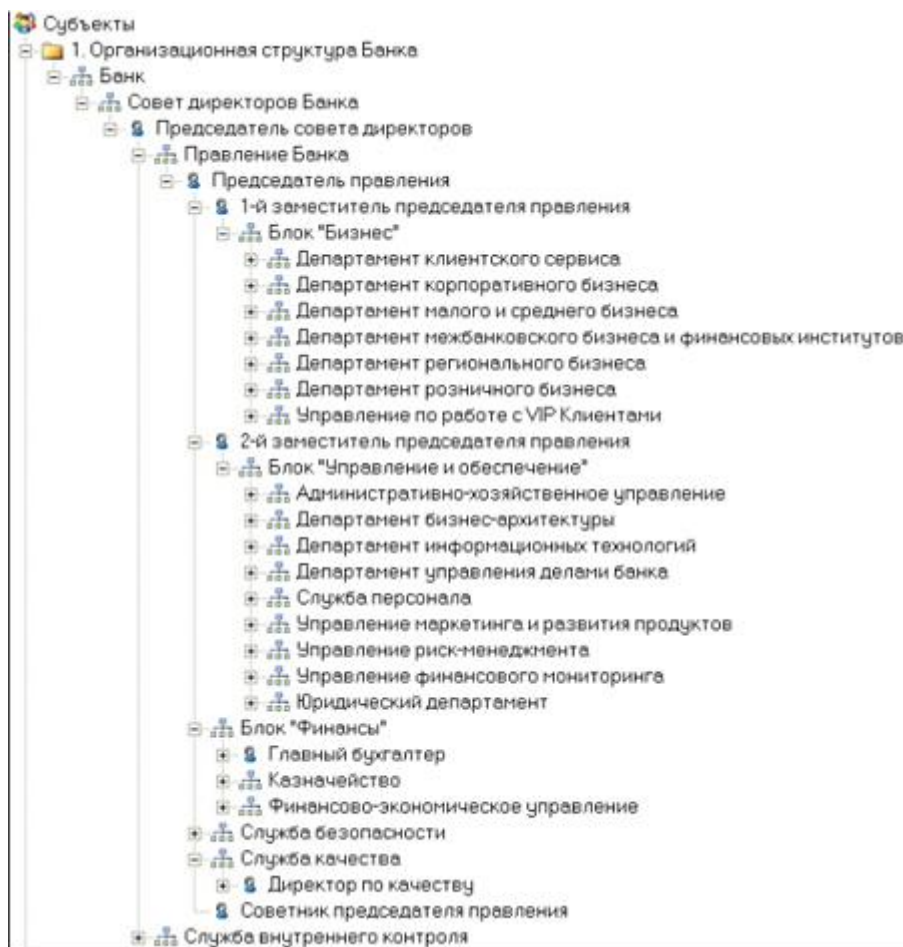
Tashkilotlarda ishlab chiqilgan biznes modellarining asosiy turlari:

- biznes jarayonlarining daraxti (ierarxik ro'yxati) - rasmga qarang. 1;
- biznes jarayonlarining grafik modellari;
- tashkiliy tuzilma modeli - rasmga qarang. 2;
- maqsadlar va ko'rsatkichlar modellari (strategik xaritalar BSC / KPI);
- hujjatlar kutubxonasi modellari (hujjatlar daraxti), axborot tizimlari modellari (tizim arxitekturasini) - rasmga qarang. 3;
- mahsulot va xizmatlar modellari - rasmga qarang. 4;
- sifat menejmenti modellari va boshqalar.

Ushbu modellarning barchasi biznesni modellashtirish uchun professional dasturiy mahsulotlarni (BPBM) ishlab chiqish imkonini beradi. 10 yildan ortiq vaqt davomida muallif PPBM bozorida ma'lum bo'lgan ko'pgina echimlardan loyihalarda va o'z ishlanmalarida foydalanmoqda: Business Studio, ARIS, AllFusion Process Modeler (BPWIN), Business Engineer, Microsoft Visio. Ularning har biri o'zining funktsional xususiyatlari, cheklovlari va afzalliklariga ega. Muallif tomonidan ishlab chiqilgan dasturiy mahsulotlarni solishtirish metodologiyasi bilan batafsil kitobda tanishishingiz mumkin. Business Studio dasturiy mahsulotida muallif "Tijorat bankining keng qamrovli standart biznes modeli"ni ishlab chiqmoqda, moliyaviy tashkilotlarni qiziqtiradigan.



2.2-rasm. Bank biznes jarayonlari daraxti (yuqori daraja)



2.3-rasm. Bankning tashkiliy tuzilmasi modeli (yuqori daraja)



2.4-*rasm. Bank hujjatlari kutubxonasi modeli (fragment)*



2.5-*рasm. Bank mahsulotlari va xizmatlari modeli (yuqori daraja)*

Biznes tahlilchisi uchun bilim va vositalarning "janoblar to'plami"

Keling, muallifning fikriga ko'ra, zamonaviy biznes tahlilchisi va biznesni modellashtirish bo'yicha mutaxassis ega bo'lishi kerak bo'lgan asosiy bilimlar va vositalar to'plamini sanab o'tamiz. Ushbu ro'yxat yosh mutaxassislarga o'zlarining kuchli tomonlari va rivojlanish imkoniyatlarini tahlil qilish uchun ham foydali bo'lishi mumkin.

1. Biznesni modellashtirish uchun dasturiy mahsulotlar: Business Studio, ARIS, AllFusion Process Modeler (BPWIN), Business Engineer, Microsoft Visio.

2. Biznesni modellashtirish va biznes jarayonlar tavsifi uchun notatsiyalar: IDEF0, IDEF3, Ma'lumotlar oqimi diagrammasi (DFD), kengaytirilgan hodisalarga asoslangan jarayonlar zanjiri (eEPC), qo'shilgan qiymat zanjiri diagrammasi (VAD), o'zaro funktsional oqim sxemasi va boshqalar.

Har bir dasturiy mahsulotda, biznesda modellash tirish o'ziga xos belgilar to'plamiga ega va ular dasturiy mahsulot uchun foydalanuvchi qo'llanmasida batafsil tavsiflangan.

3. Biznes muhandisligi/boshqaruv texnikasi va texnikasi.

- BSC/KPI balanslangan ko'rsatkichlar kartasi tizimini ishlab chiqish va joriy etish;
- Biznes jarayonlarining tavsifi;
- Biznes jarayonlarini tahlil qilish, optimallashtirish, sifatini oshirish;
- Uzoq muddatli asosda biznes jarayonlarini boshqarish;
- Funktsional xarajatlar tahlili (FCA) va simulyatsiya modellash tirish;
- Tashkiliy tuzilmani tavsiflash va optimallashtirish, xodimlar soni;
- Xodimlarni rag'batlantirish tizimini qurish;
- Sifat menejmenti tizimini qurish va uning faoliyatini tashkil etish (ISO 9000);
- Loyiha boshqaruvi (shu jumladan PMBOK - Loyiha boshqaruvi bilimlari organi);
- Tashkilot uchun keng qamrovli biznes modelini yaratish;
- Benchmarking;
- Lean, 6 Sigma;
- TQM (jami sifat menejmenti);
- Turli soha metodologiyalari va standartlari, konsalting kompaniyalarining ishlanmalari.

Bank sohasiga tegishli barcha usullarning to'liq ro'yxati va batafsil tavsifi [2] da keltirilgan.

4. Standart echimlar, misollar, ishlanmalar va materiallar. Ko'pgina materiallarni noldan ishlab chiqmaslik va boshqa mutaxassislar allaqachon qilgan xatolarga yo'l qo'ymaslik uchun sizga standart echimlar, modellar, hujjatlar va boshqalar kerak bo'ladi. Masalan, "Tijorat bankining kompleks standart biznes modeli" elektron ma'lumotlar bazasi (ma'lumotnomasi) [1].

Shunday qilib, quyidagi sxema tuzilishi mumkin:

Metodologiya + Standart echimlar + Dasturiy mahsulot = Natija.

Bu yerda Texnikalar va Texnikalar loyiha va vazifalarni QANDAY bajarishni ko'rsatadi.

PPBM yordamida barcha vazifalar va loyihalarning bajarilishi avtomatlashtirilgan. Bu vaqtni bir necha marta qisqartiradi va ish

samaradorligini oshiradi. Misol uchun, Business Studio tizimi bir tugmani bosish orqali moliyaviy va mehnat resurslarini sezilarli darajada tejashni ta'minlaydigan ishlab chiqilgan biznes-jarayon modellari asosida me'yoriy hujjatlarni avtomatik ravishda yaratish imkonini beradi.

Biznesni modellashtirish: amaliy qo'llash xususiyatlari

Biznesni modellashtirishning asosiy xususiyati shundaki, u biznes jarayonlariga asoslanishi kerak. Aynan biznes jarayonlarini boshqarish tizimi (BPS) ko'plab boshqa boshqaruv tizimlari va texnologiyalari qurilgan poydevordir.

Ko'pgina tashkilotlar boshqaruv, takomillashtirish va optimallashtirish uchun turli yondashuvlar, usullar va texnologiyalarni faol ravishda joriy qildilar va joriy etishda davom etmoqdalar.

Amaliyot shuni ko'rsatadiki, ba'zi hollarda bu usullar amalga oshirishning dastlabki bosqichida muvaffaqiyatli bo'ladi, lekin keyin asta-sekin samaradorligini yo'qotadi va unutiladi.

Ushbu yondashuvlar/usullardan foydalangan holda tashkilot faoliyatini yaxshilashga urinishlarning muvaffaqiyatsizligi ko'pincha chuqur tahlil va tashkilot faoliyatidagi tub o'zgarishlarni o'z ichiga olmaydigan tizimsiz va parchalangan harakatlar bilan bog'liq.

Ushbu muammoni bartaraf etishning asosiy usuli - tashkilotga boshqaruvga texnologik yondashuvni joriy etish (ya'ni, biznes-jarayonlarni boshqarish tizimini yaratish) boshqa texnikalarni, boshqaruv/takomillashtirish va optimallashtirish texnologiyalarini joriy qilish uchun asos sifatida.

Oddiy va tushunarli harakatlarga qisqartirib bo'lmaydigan biznesni modellashtirishning murakkab usullari odatda tashkilotlarda ishlamaydi. Oxir oqibat, ushbu usullarni amalga oshirish va ularni qo'llash natijalari har doim ham zamonaviy boshqaruv texnikasi va biznes injiniringi sohasida ixtisoslashgan vakolatlarga ega bo'lmagan, ba'zan esa ular bilan uchrashadigan tashkilot xodimlari va yo'nalish rahbarlari zimmasiga tushadi. dushmanlik.

Tashkilotda va umuman loyihada amalga oshirilgan metodologiya (texnologiya) muvaffaqiyatli bo'lishi va rejalashtirilgan natijalarni berishi uchun ular quyidagilar bo'lishi maqsadga muvofiqdir:

1. Arzon. Bu, ayniqsa, qimmat echimlarni amalga oshirishga qodir bo'lmagan o'rta va kichik tashkilotlar uchun to'g'ri keladi.

2. Oddiy va tashkilotning oddiy xodimlariga tushunarli.

3. Amalda juda "tez" va ayni paytda uzoq muddatli natijalarga erishishga qaratilgan.

4. Biz Rossiya kompaniyalarini boshqarishning o'ziga xos xususiyatlarini hisobga oldik.

5. Misollar va standart echimlarni o'z ichiga oladi.

Bu yerda biznesni modellashtirishning barcha vazifalariga taalluqli bo'lgan va ularning amalga oshirilishini ta'minlashga imkon beradigan sifat menejmentining 8 ta asosiy tamoyilini [4] keltirish o'rinlidir.

1. Mijozlarga e'tibor qaratish;

2. Ijro etuvchi rahbarlik;

3. Xodimlarning ishtiroki;

4. Jarayonga yondashuv;

5. Boshqaruvga tizimli yondashish;

6. Doimiy takomillashtirish;

7. Faktga asoslangan qaror qabul qilish;

8. Yetkazib beruvchilar bilan o'zaro manfaatli munosabatlar.

Darhaqiqat, hatto 1-2 tamoyilga rioya qilmaslik tashkilotning rivojlanishiga salbiy ta'sir ko'rsatishi mumkin.

Biznesni modellashtirishning ahamiyati

Biznes modellarini ishlab chiqishni boshlaganda, tashkilotlar loyihani amalga oshirish uchun ma'lum insoniy va moddiy resurslarni ajratadilar. Bundan tashqari, bajarilgan ishlar natijasida yaxshilanishlar ushbu xarajatlardan oshib ketishi kerak. Biznes modeli oxir-oqibat tashkilotning ishlashiga qanday yordam beradi? Biz biznes jarayonlarining malakali va tizimli tavsifi bilan o'zini namoyon qiladigan eng sezilarli va keng tarqalgan ijobiy ta'sirlardan bir nechtasini ajratib ko'rsatishimiz mumkin.

1. Barcha darajadagi tashkilot faoliyatining shaffofligi, boshqarilishi va nazorat qilinishini oshirish.

2. Bajarish vaqti va xarajatlarini qisqartirish, biznes jarayonlarining sifati va samaradorligini oshirish.

3. Tashkilotning biznesini takrorlash qobiliyati (qo'shimcha mijozlar filiallari, ofislari, vakolatxonalarini yaratish).

4. Tashkilotning kompleks va barqaror rivojlanishi, qaror qabul qilishda tizimli yondashuv.

5. Xodimlarga qaramlikni kamaytirish, xodimlarni to'g'ri tanlash, xodimlar va menejrlarning samaradorligini oshirish.

6. Mijozlarning sadoqati va qoniqishini oshirish va natijada tashkilotning obro'sini oshirish.

7. Moliyaviy natijalar.

Biroq, keng doiradagi menejerlar va biznes egalariga unchalik yaxshi ma'lum bo'lmagan boshqa jihatlar ham mavjud.

Biznesni modellashtirish va tegishli texnologiyalar/yechimlar tashkilotning reyting agentliklari, shu jumladan xalqaro (Fitch, Moody's, S&P va boshqalar) tomonidan beriladigan reytinglariga sezilarli ta'sir ko'rsatadi.

Turli xalqaro va Rossiya agentliklarining (shu jumladan [3]) reytinglarini berish usullarini tahlil qilish natijasida, shuningdek agentlik vakillari bilan suhbatlar natijalariga ko'ra, muallif ko'plab agentliklar qachon ekanligini aniqladi. tashkilotlarning reytinglarini hisoblashda "Korporativ boshqaruv / menejment" kod nomi ostida bir guruh omillarni hisobga olish (moliyaviy bo'lmagan baholashlar). Ushbu parametr quyidagi omillarni o'z ichiga oladi:

- Adekvat va batafsil tashkiliy strategiya.
- Rivojlangan risklarni boshqarish tizimi (shu jumladan, operatsion risklarni boshqarish tizimi).
- Biznes jarayonlarini tartibga solish (rasmiylashtirish) darajasi.
- Biznes jarayonlarining sifati (KPI ko'rsatkichlari tarixi).
- Biznes jarayonlarini avtomatlashtirish darajasi, axborot tizimlari va texnologiyalari (IT) holati.
- Tashkiliy tuzilma (rasmiylashtirish, samaradorlik, shaffoflik, mas'uliyat va vakolatlarni taqsimlash).
- Tashkilotdagi turli xil boshqaruv tizimlarining rivojlanishi va faoliyati (sifat menejmenti tizimi, ish tizimi va mijozlar bilan munosabatlar, xodimlarni boshqarish tizimi va boshqalar).

Batafsil shartlar va reytinglar agentlikka qarab farqlanadi.

Baho berish algoritmi juda oddiy va tushunarli. Reyting agentligining auditorlari tashkilot faoliyatini reytingni belgilash metodologiyasida belgilangan qoidalar va mezonlarga muvofiq o'rganadi va baholaydi. Kirish ma'lumotlari:

- tashkilotning me'yoriy va hisobot hujjatlari;
- tashkilot faoliyatini kuzatish va suhbatlar.

Shuning uchun nafaqat ko'p sonli to'g'ri va dolzarb hujjatlarni ishlab chiqish, balki ularni xodimlarga etkazish, biznes jarayonlarining

samarali bajarilishini va boshqaruv tizimlarining amalda ishlashini ta'minlash muhimdir.

Barcha mezonlar bo'yicha ballar ma'lum qoidalarga muvofiq yig'iladi va umumiy ball asosida tashkilot reytingi aniqlanadi. Har bir mezon guruhi har xil vaznga ega bo'lishi mumkin, shuning uchun kichik vaznga ega bo'lgan mezonlar guruhi uchun katta miqdordagi ballar yakuniy bahoga juda katta hissa qo'shmaydi.

Belgilangan reytinglarning belgilari (reyting shkalasi) reyting agentligiga va reyting turiga (kredit reytingi, ishonchlilik reytingi, boshqaruv sifati reytingi, moliyaviy barqarorlik reytingi va boshqalar) qarab farq qilishi mumkin. Masalan: ishonchlilikning eng yuqori darajasi, ishonchlilikning qoniqarli darajasi, ishonchlilikning past darajasi va boshqalar.

Yuqori reyting nafaqat tashkilot va uni boshqarishning ishonchliligi va barqarorligini ko'rsatadi, balki quyidagi muhim muammolarni hal qilish imkonini beradi.

1. Tenderlarda ishtirok etish va akkreditatsiya.
2. Bozorda, hamkorlar va pudratchilar orasida tashkilotning imidjini (avtoritetini) yaxshilash.
3. Tashkilotning davlat hokimiyati organlari o'rtasida nufuzini (avtoritetini) oshirish.
4. Mijozlar bazasini kengaytirish.
5. Investorlarni jalb qilish.
6. Yuqoridagi barcha fikrlar natijasida - yaxshilangan moliyaviy ko'rsatkichlar.

Shunday qilib, xalqaro yoki milliy reytinglarni yaxshilashdan manfaatdor davlat kompaniyalari uchun integratsiyalashgan biznes modelini yaratish loyihasi samaradorligini baholashda reyting pozitsiyalarini yaxshilash uchun qo'shimcha imkoniyatlarni hisobga olish tavsiya etiladi. Ta'kidlash joizki, tashkilot reytingiga ta'sir qiluvchi yuqorida sanab o'tilgan barcha omillarni etarli darajada ishlab chiqish, albatta, biznesni modellashtirish uchun professional dasturiy mahsulotlardan (BPBM) foydalanishni talab qiladi. Ushbu yo'nalishda qo'shimcha imkoniyatlar standart muvaffaqiyatli sanoat echimlaridan foydalanish orqali taqdim etiladi. Hozirgi misol - Business Studio dasturiy mahsulotida ishlab chiqilgan "Tijorat bankining kompleks standart biznes modeli" [1]. Kredit tashkilotlarida jarayonlarni boshqarishning ilg'or tajribalarini umumlashtirgan holda, ushbu model

yuqoridagi barcha parametrlar bo'yicha moliya sektoridagi kompaniyalar korporativ boshqaruvni yaxshilashi mumkin bo'lgan model bo'lib xizmat qiladi.

Moliya-kredit tashkilotlarida biznesni modellashtirish amaliyoti

Tashkilot uchun biznes modelini yaratish to'g'risidagi qaror muayyan kompaniyalarning boshqaruv xususiyatlariga qarab turlicha qabul qilinishi mumkin. Ba'zan bu top-menejerning yagona qarori bo'lib, kompaniya egalari biznesni modellashtirish zarurligini tushunishlari mumkin. Bank tashkilotlari bilan ishlash amaliyotida muallif bunday misollar bilan shug'ullanishi kerak edi.

"Barcha bank faoliyati kompyuterda bitta tugmani bosish orqali"

"A" bank boshqaruvi raisi yig'ilishlardan birida buyurdi: "Bankning barcha faoliyatini rasmiylashtirish kerak, toki kompyuterda tugmani bosish orqali men istalgan xodimning ishini ko'ra olaman va bankning har qanday biznes jarayoni: uning maqsadlari, ko'rsatkichlari, jarayonlari, texnologiyalari, natijalari va boshqalar.

Ushbu muammoni hal qilish uchun bankning elektron biznes modeli ishlab chiqildi. Boshqaruv raisi kompyuterining ish stoliga veb-brauzer oynasi joylashtirildi. Unda joylashgan havolalar sizga barcha faoliyatni kuzatish imkonini beradi: Menejer har qanday hujjatni, biznes-jarayon diagrammasini ochishi, biznes jarayonlari va protseduralari uchun mas'ul shaxslarni, biznes-jarayon ko'rsatkichlari va joriy qiymatlar bo'yicha statistik ma'lumotlarni, hozirda amalga oshirilayotgan loyihalar ro'yxatini topishi mumkin. bankdagi va ularning holati, har qanday bo'limning tashkiliy tuzilishi va boshqalar.

Boshqaruv raisi amalga oshirilgan ishlardan juda mamnun bo'ldi. Shuni ta'kidlash kerakki, ish qisqa vaqt ichida yakunlandi: vazifa qo'yilgan paytdan boshlab yakuniy natijalarni olishgacha 1,5 yil o'tdi. Loyihani amalga oshirishning yuqori tezligiga uslubiy asos sifatida standart yechim – "Tijorat bankining keng qamrovli standart biznes modeli" [1] dan foydalanish tufayli erishildi, bu o'zaro bog'langan modellar, hujjatlar va ma'lumotnomalar tizimidan iborat. universal tijorat banki faoliyatining aksariyat sohalari va boshqaruv tizimlari.

Aytgancha, loyiha tugallanganda, boshqaruv raisi allaqachon bank aksiyadoriga aylangan edi. Natijada yaratilgan Integratsiyalashgan biznes modeli bank boshqaruviga tizimli yondashuvni ta'minladi, bu sizga tezkor qarorlar qabul qilish va bank ishida har qanday o'zgarishlarni amalga oshirish imkonini beradi,

alohida biznes jarayonlari va bo‘linmalari samaradorligi va sifatini oshiradi, shuningdek, bank sifatida. butun.

“Bank rivojlanishiga tizimli yondashuv”

“B” banki aksiyadorlari oldiga zamonaviy boshqaruv texnologiyalari asosida bankni kompleks va uzoq muddatli rivojlantirish strategiyasini ishlab chiqish vazifasini qo‘ydi. Tadqiqotlar olib borgan va bir nechta biznes treninglarda qatnashgandan so‘ng, bankning tashkiliy va korporativ rivojlanish bo‘yicha mutaxassislari aksiyadorlarga quyidagi yechimni taklif qilishdi.

Bankning korporativ strategiyasi allaqachon aniqlanganligi sababli, siz bankning biznes-jarayonlarini boshqarish tizimini ishlab chiqishdan boshlashingiz mumkin, chunki biznes-jarayonlar bankning barcha ishining mohiyatini tashkil etadi va mijozlarning qoniqishi va bankning foydasi bankning biznes-jarayonlariga bog‘liq. biznes jarayonlarining natijalari.

1. Biz barcha asosiy biznes jarayonlarini tavsiflaymiz, jarayon guruhlarini yaratamiz va ularni o‘qitamiz, biznes jarayonlarining barcha ishtirokchilarining samarali o‘zaro ta’sirini ta’minlaymiz, shunda biznes jarayonlar tezroq yakunlanadi.

2. Biz kerak bo‘lganda jarayonlarni yaxshilaymiz (optimallaymiz), so‘ngra doimiy ravishda biznes jarayonlarini boshqarishni tashkil qilamiz. Har bir biznes-jarayon doirasida biz strategik rejalashtirishni shunday tashkil qilamizki, har bir biznes-jarayon bozor tendentsiyalari, mijozlar talablari va bank strategiyasi, shuningdek, maqsad va ko‘rsatkichlarga asoslangan strategiyaga ega bo‘ladi.

3. Biznes jarayonlari va ularni boshqarish shaffof va soddalashtirilganda biz keyingi vazifaga – jarayonlarni boshqarish tizimiga asoslangan bank sifat menejmenti tizimini (ISO 9000 standartlari bo‘yicha) qurishga o‘tamiz. Bular. SMS jarayonlarni boshqarish tizimi uchun yuqori tuzilma bo‘ladi. Bu bankka ISO 9001 standartiga muvofiqlik sertifikatini olish va mijozlar va hamkorlar orasida o‘z nufuzini oshirish imkonini beradi. Shuningdek, QMS va ISO 9000 standartlari tufayli biz mijozlarning bankka bo‘lgan da’volari sonini, sifatsiz mahsulot va xizmatlarga bo‘lgan xarajatlarni sezilarli darajada qisqartiramiz, operatsion risklarni minimallashtiramiz va bank faoliyatini yangi talablar va boshqaruv usullari bilan to‘ldiramiz.

4. Bunga parallel ravishda biz biznes jarayonlarini avtomatlashtirishni boshlaymiz. Biz elektron hujjat aylanishi va operativ boshqaruv (DocFlow/WorkFlow), mijozlar bilan ishlash (CRM) va boshqalar tizimlarini yangilaymiz va sifat jihatidan yangi bosqichga olib chiqamiz. bankning korporativ rivojlanishi va bu faoliyat tizimli bo'lishi uchun bank xodimlarini boshqarishni sifat jihatidan takomillashtiradi.

Natijada biz bank boshqaruvining integratsiyalashgan tizimi – bank aksiyadorlari va top-menejerlari uchun zamonaviy, samarali tashkiliy boshqaruv vositasiga ega bo'lamiz.

Nazorat savollari

1. BPMN nima uchun ishlatiladi va uning asosiy maqsadi nima?
2. BPMN notatsiyasining eng asosiy elementlari nima? Ularning mazmuni va vazifalari nimalar?
3. BPMN notatsiyasida biznes-jarayonlarini modellovchi elementlar qanday?
4. BPMN notatsiyasida qanday turli jarayonlarni modellovchi elementlar mavjud? Ularning turli turdagi vazifalari va qo'llanish sohalarini ko'rsating.
5. BPMN notatsiyasida jarayonlar orasidagi aloqani ko'rsatish uchun qanday elementlar foydalaniladi?
6. BPMN notatsiyasi yordamida qanday jarayonlarni avtomatlashtirish mumkin? Ularning avtomatlashtirishda qanday metodologiyalar va vositalar foydalaniladi?
7. BPMN notatsiyasini yaratish jarayonida qanday bosqichlar amalga oshiriladi?
8. BPMN notatsiyasidagi shlyuzlar (gateways) qanday turdagi aloqalarni bildirish uchun ishlatiladi?
9. BPMN notatsiyasidagi voqealar (events) qanday turlarga bo'linadi? Ularning vazifasi va qo'llanish sohalarini ta'riflang.

3-MAVZU. DASTURIY TA'MINOT LOYIHALARNI YARATISHGA QO'YILGAN TALABLAR

Reja:

1. Dasturiy ta'minot talablari
2. Talablarni aniqlash usullari
3. Talablar tahlili

Dasturiy ta'minot talablari - bu amalga oshirilishi kerak bo'lgan dasturiy ta'minot tizimining atributlari, xususiyatlari yoki sifatleri bo'yicha so'rovlar bayonotlar to'plami. Ular dasturiy ta'minotni ishlab chiqish/modernizatsiya qilish bo'yicha vazifalarni ishlab chiqish (tahlil va sintez) jarayonida yaratiladi.

Talablar matnli bayonotlar va grafik modellar shaklida ifodalanishi mumkin.

Klassik texnik yondashuvda dasturiy ta'minotni loyihalash bosqichida talablar to'plami qo'llaniladi. Talablar dasturiy ta'minotni tekshirish jarayonida ham qo'llaniladi, chunki testlar talablarga asoslanadi.

Talablarni ishlab chiqish bosqichidan oldin texnik-iqtisodiy asoslash yoki kontseptual dizayn tahlili bosqichi bo'lishi mumkin. Talablarni ishlab chiqish bosqichi talablarni aniqlash (manfaatdor tomonlarning ehtiyojlarini yig'ish, tushunish, ko'rib chiqish va aniqlashtirish), tahlil qilish (yaxlitlik va to'liqlikni tekshirish), spetsifikatsiya (talablarni hujjatlashtirish - matn va grafik modellarni sintez qilish) va tasdiqlashga bo'linishi mumkin.

Daraja bo'yicha talablar turlari

- Biznes talablari - dasturiy ta'minotning maqsadini aniqlash, ko'rish (ko'rish) va loyiha chegaralari (ko'lami) haqidagi hujjatda tasvirlangan.
- Foydalanuvchi talablari - dastur hal qilishi kerak bo'lgan foydalanuvchi vazifalari to'plamini, shuningdek ularni tizimda hal qilish usullarini (stsenariylarini) aniqlash.
- Funktsional talablar - tizimning funktsiyalariga qo'yiladigan talablarni aniqlash.

Tabiatan talablar turlari

- Funktsional tabiat - bu tizim xatti-harakatlariga qo'yiladigan talablar
 - Biznes talablari
 - Foydalanuvchi talablari

- Funktsional talablar
 - Funktsional bo‘lmagan tabiat (arxitektura/amalga oshirish/ishlash cheklovlari) - tizim xatti-harakatlarining tabiatiga qo‘yiladigan talablar
 - Biznes qoidolari - avtomatlashtirilgan obyektning (korxonaning) predmeti va xususiyatlaridan kelib chiqadigan cheklovlarni belgilaydi.
 - Tizim talablari va cheklovlari - bu tizimda bo‘lishi kerak bo‘lgan elementar operatsiyalarning ta’riflari, shuningdek, u qondirishi mumkin bo‘lgan turli shartlar. Tizim talablari va cheklovlari quyidagilarni o‘z ichiga oladi:
 - Dasturiy ta’minot interfeyslari, shu jumladan tashqi tizimlar uchun cheklovlar
 - Sifat atributlariga qo‘yiladigan talablar
 - Amaldagi uskunalar va dasturiy ta’minotga qo‘yiladigan talablar
 - Hujjatlarga qo‘yiladigan talablar
 - Dizayn va foydalanishga qo‘yiladigan talablar
 - Xavfsizlik va ishonchlilik talablari
 - Belgilangan ko‘rsatkichlarga qo‘yiladigan talablar (ishlash, nosozliklarga chidamlilik va boshqalar).
 - Operatsion va xodimlarga qo‘yiladigan talablar
 - Boshqa talablar va cheklovlar (tashqi ta’sirlar, harakatchanlik, avtonomiya va boshqalar).

Talablar manbalari

Talablarning sifat ko‘rsatkichlari turli manbalar tomonidan turlicha aniqlanadi. Biroq, quyidagi xususiyatlar odatda qabul qilinadi.

Xarakterli	Tushuntirish
Yakkalik	Talab bitta narsani va faqat bitta narsani tasvirlaydi.
To‘liqlik	Talab to‘liq bir joyda aniqlangan va barcha kerakli ma'lumotlar mavjud.
Keyingi ketma-ketlik	Talab boshqa talablarga zid emas va tashqi hujjatlarga to‘liq mos keladi.
Atomlik	Talab "atom". Ya'ni, to‘liqligini yo‘qotmasdan, bir qator batafsilroq talablarga bo‘linib bo‘lmaydi.
Kuzatish imkoniyati	Talab manfaatdor tomonlar tomonidan bildirilgan va hujjatlashtirilgan holda biznes ehtiyojlarining to‘liq yoki bir qismini qondiradi.
Muvofiqlik	Vaqt o‘tishi bilan talab eskirgani yo‘q.
Fizibilite	Talab loyiha doirasida amalga oshirilishi mumkin.

Aniqlik	Talab texnik jargon, qisqartmalar yoki boshqa yashirin tillarga murojaat qilmasdan, qisqacha tavsiflanadi. U sub'ektiv fikrlarni emas, balki obyektiv faktlarni ifodalaydi. Bitta va faqat bitta talqin qilish mumkin. Ta'rifda noaniq iboralar mavjud emas. Salbiy bayonotlar va qo'shma bayonotlardan foydalanish taqiqlanadi.
Majburiy	Talab manfaatdor tomon tomonidan belgilangan xususiyatni ifodalaydi, uning yo'qligi e'tibordan chetda qoldirib bo'lmaydigan past yechimga olib keladi. Ixtiyoriy talab - bu talab tushunchasiga ziddir.
Tekshirish imkoniyati	Talabning maqsadga muvofiqligi to'rtta mumkin bo'lgan usullardan biri orqali aniqlanishi mumkin: tekshirish, namoyish qilish, sinov yoki tahlil.

Talablarni aniqlash usullari

- Suhbatlar, so'rovlar, anketalar
- Aqliy hujum, seminar
- Ishlab chiqarish faoliyatini nazorat qilish, ish kunini "fotosuratga olish"
- Normativ hujjatlarni tahlil qilish
- Faoliyat shakllarini tahlil qilish
- Raqobatbardosh mahsulot tahlili
- Tizimning oldingi versiyalaridan foydalanish bo'yicha statistik ma'lumotlarni tahlil qilish

Talablarni tekshirish

Barcha talablar tekshirilishi kerak. Tasdiqlashning eng keng tarqalgan usuli bu testlardir. Agar testlar orqali tekshirish mumkin bo'lmasa, boshqa tekshirish usuli (tahlil, namoyish, tekshirish yoki loyihani ko'rib chiqish) qo'llanilishi kerak.

Ba'zi talablar tabiatan tasdiqlanmaydi. Ular tizim *hech qachon* yoki *har doim* ma'lum bir xususiyatni namoyish etmasligi kerakligi haqidagi talablarni o'z ichiga oladi. Ushbu talablarni to'g'ri sinovdan o'tkazish cheksiz sinov siklini talab qiladi. Bunday talablar tekshirilishi uchun qayta belgilanishi kerak. Yuqorida aytib o'tilganidek, barcha talablar tekshirilishi kerak.

Dasturiy ta'minot darajasida tekshirib bo'lmaydigan funktsional bo'lmagan talablar mijozning niyatini tasdiqlovchi hujjat sifatida saqlanishi kerak. Bunday mahsulot talablari jarayon talablariga tarjima

qilinishi mumkin. Masalan, dasturiy ta'minotda orqa eshiklar bo'lmashligi kerak bo'lgan funktsional bo'lmagan talab, uni juft dasturlashdan foydalanish talabi bilan almashtirish orqali qondirish mumkin. DO-178B ishlab chiqish jarayoniga rioya qilish orqali aviatsiya dasturiy ta'minotining murakkab xavfsizligi talablarini qondirish mumkin.

Talablar tahlili

Talablarni ishlab chiqishda ko'pincha individual talablarning noaniqligi, to'liq emasligi va nomuvofiqligi muammolari paydo bo'ladi. Talablarni ishlab chiqish bosqichida ushbu muammolarni hal qilish, keyinchalik ishlab chiqishda bir xil muammolarni hal qilishdan ko'ra bir necha baravar kam xarajat qiladi. Ushbu muammolarni hal qilish va hal qilish uchun talablar muhandislik jarayoni mavjud.

Talablar muhandisligida juda noaniq talablar va shu qadar batafsil talablar o'rtasida texnik qarama-qarshilik mavjudki, ular:

- rivojlanish uchun uzoq vaqt talab etiladi, ba'zan hatto rivojlanishning oxirigacha eskirib qolish xavfi mavjud;
- amalga oshirishning mumkin bo'lgan usullarini cheklash;
- juda qimmat.

Nazorat savollari

1. Dasturiy ta'minot talablarini aniqlashning asosiy bosqichlari qanday?
2. Funktsional va nofunktsional talablar o'rtasidagi farq nima?
3. Talablarni yig'ish va tahlil qilishda qaysi metodlardan foydalaniladi?
4. Talablarni prioritizatsiya qilish qanday amalga oshiriladi va nima uchun bu muhim?
5. Talablarni hujjatlashtirishda qanday standartlar va vositalardan foydalaniladi?
6. Talablarning izchil va to'liq ekanligini qanday tekshirish mumkin?
7. Talablarni boshqarish jarayonida qanday vositalar va metodlar qo'llaniladi?
8. Manfaatdor tomonlar (stakeholders) bilan qanday qilib samarali muloqot qilish mumkin?
9. Talablarning o'zgarishini boshqarish (Change Management) jarayoni qanday amalga oshiriladi?
10. Yuqori sifatli dasturiy ta'minotni ta'minlash uchun talablar qanday sifat mezonlariga javob berishi kerak?

4-MAVZU. DASTURIY TA'MINOT LOYIHALARNI YARATISHNING HUQUQIY, NORMATIV VA METODIK TAMINOTI. DTLARNI YARATISHNING XALQARO STANDARTLARI

Reja:

1. ISO (International Organization for Standardization)
2. ISO standartini ishlab chiqish jarayonining bosqichlari
3. AT larini yaratishning xalqaro standartlari va loyihalari
4. AT da sifat menejmenti

ISO (International Organization for Standardization)

ISO (International Organization for Standardization - Xalqaro standartlashtirish tashkiloti) a'zo mamlakatlarning milliy standartlar tashkilotlari vakillaridan tashkil topgan xalqaro standartlarni ishlab chiqish tashkilotidir. 1947-yil 23-fevralda tashkil etilgan va o'shandan beri texnologiya va biznesning barcha jabhalarini qamrab oluvchi 19500 dan ortiq xalqaro standartlarni nashr etgan. Tashkilot elektrotexnika va elektron muhandislikdan tashqari barcha texnik va texnik bo'lmagan sohalarda standartlashtirishni ishlab chiqadi va nashr etadi. Bosh qarorgohi Shveytsariyaning Jeneva shahrida joylashgan va 2022-yil holatiga ko'ra 167 mamlakatda ishlaydi. ISO ning uchta rasmiy tili ingliz, frantsuz va rus tillaridir.

Pragadagi ISO oldingi, ISA

Pragadagi ISO oldingi, ISA asos solingan binoga faxtiy nishon belgisi qo'yilgan. Bugungi kunda ISO nomi bilan tanilgan tashkilot 1926-yilda Milliy standartlashtirish assotsiatsiyalari xalqaro federatsiyasi (ISA) sifatida boshlangan bo'lib, u asosan mashinasozlikga mo'ljallangan. ISA 1942-yilda Ikkinchi Jahon urushi paytida to'xtatilgan; ammo, urushdan so'ng, ISA ga yaqinda tuzilgan Birlashgan Millatlar Tashkilotining Standartlar bo'yicha muvofiqlashtiruvchi qo'mitasi (UNSCC) yangi global standartlar organini shakllantirish taklifi bilan murojaat qildi. 1946-yil oktabr oyida 25 ta mamlakatdan kelgan ISA va UNSCC delegatlari Londonda uchrashib, Xalqaro standartlashtirish tashkilotini yaratish uchun kuchlarni birlashtirishga kelishib oldilar. Tashkilot 1947-yil 23-fevralda rasman o'z faoliyatini boshladi

ISO standartini ishlab chiqish jarayonining bosqichlari

Standartlashtirish jarayoni ISO/IEC tomonidan e'lon qilingan standart odatda qo'mita ichida yangi ish taklifi bilan boshlanadigan uzoq jarayonning oxirgi bosqichidir. Standartni uning holati bilan belgilash uchun ishlatiladigan ba'zi qisqartmalar:

- PWI - Dastlabki ish elementi
 - NP yoki NWIP - Yangi taklif / yangi ish elementi taklifi (masalan, ISO/IEC NP 23007)
 - AWI - tasdiqlangan yangi ish elementi (masalan, ISO/IEC AWI 15444-14)
 - WD - Ishchi qoralama (masalan, ISO/IEC WD 27032) • CD - Qo'mita loyihasi (masalan, ISO/IEC CD 23000-5)
 - FCD - Yakuniy qo'mita loyihasi (masalan, ISO/IEC FCD 23000-12)
 - DIS - Xalqaro standart loyihasi (masalan, ISO/IEC DIS 14297)
 - FDIS - Xalqaro standartning yakuniy loyihasi (masalan, ISO/IEC FDIS 27003)
 - PRF - Yangi xalqaro standartning isboti (masalan, ISO/IEC PRF 18018) • IS – Xalqaro standart (masalan, ISO/IEC 13818-1:2007)
- O'zgartirishlar uchun quyidagi qisqartmalar qo'llaniladi:
- NP Amd – Yangi taklifga tuzatish (masalan, ISO/IEC 15444-2:2004/NPAmd)
 - AWI Amd – Tasdiqlangan yangi ish elementi tuzatish (masalan, ISO/IEC 14492:2001/AWI Amd 4)
 - WD Amd – Ishchi tuzatish loyihasi (masalan, ISO 11092:1993/WD Amd 1)
 - CD Amd / PDAmD - Qo'mitaning o'zgartirish loyihasi / Taklif etilayotgan tuzatish loyihasi (masalan, ISO/IEC 13818-1: 2007/CD Amd 6)
 - FPDAmD / DAM (DAmd) - Yakuniy taklif qilingan tuzatish loyihasi / tuzatish loyihasi (masalan, ISO/IEC 14496- 14: 2003/FPDAmd 1)
 - FDAM (FDAmD) – Yakuniy tuzatish loyihasi (masalan, ISO/IEC 13818-1:2007/FDAmd 4)
 - PRF Amd – (masalan, ISO 12639:2004/PRF Amd 1)
 - Amd – Tuzatish (masalan, ISO/IEC 13818-1:2007/Amd 1:2007)

Standartlashtirish jarayoni

Xalqaro standartlar ISO texnik qo'mitalari (TC) va quyi qo'mitalari (SC) tomonidan olti bosqichdan iborat jarayon orqali ishlab chiqiladi:

- 1-bosqich: taklif bosqichi
- 2-bosqich: tayyorgarlik bosqichi
- 3-bosqich: Komissiya bosqichi
- 4-bosqich: surishtiruv bosqichi
- 5-bosqich: tasdiqlash bosqichi
- 6-bosqich: nashr etish bosqichi

AT larini yaratishning xalqaro standartlari va loyihalari

Standartlarni tasniflashning boshqa usullari: Standartlashtirish obyekti bo'yicha:

- mahsulot va xizmatlar uchun standartlar;
- jarayonlar va texnologiyalar uchun standartlar;
- jamoaviy faoliyat shakllari standartlari yoki boshqaruv standartlari;

AT da sifat menejmenti

AT boshqaruvidan tashqari, sifat menejmenti ham xavfsizlikning asosiy rolini o'ynaydi. Yuqori raqobatbardosh axborot xavfsizligi sohasida o'z faoliyatini doimiy ravishda yaxshilashga intilayotgan korxonalar raqobatchilardan muhim ustunlikka ega bo'lishadi. ISO 27001 yoki ISO IEC 20000-1 sertifikatiga ega bo'lish orqali siz samaradorlik va natijangizni oshirish uchun barcha asosga ega bo'lasiz. Sifat menejmenti sertifikati tasdiqlangan va o'lchanadigan jarayonlarni amalga oshirishni o'z ichiga oladi - bu doimiy xarajatlarni tejash, mijozlar ehtiyojini qondirish va mas'uliyatli o'sishni anglatadi. Iste'molchilar bu imtiyozlardan tobora ko'proq xabardor bo'lib bormoqda. Ko'pchilik sifat kafolati va uzluksiz xizmat ko'rsatishga sodiqligini namoyish eta oladigan xamkorlar bilan biznes yuritishni talab qiladi. Sertifikatga ega bo'lish orqali IT tashkilotlari allaqachon bozorda ajralib turishga yordam beradigan muhim savdo nuqtasiga ega bo'lishadi.

Axborot texnologiyalarida AXBT boshqaruvining qanday afzalliklari bor?

Barcha turdagi va o'lchamdagi tashkilotlarning IT bo'limlari axborot texnologiyalari sifatini ta'minlashdan foydalanishlari mumkin.

Axborot xavfsizligini boshqarish tizimining asosiy afzalliklari quyidagilardan iborat:

- **Muvofiqlik talablariga javob berish.** Hozirgi vaqtda axborot xavfsizligini boshqarish tizimlariga tegishli bir qator qonunchilik, shartnomaviy va me'yoriy-huquqiy talablar mavjud. ISO sertifikatiga ega tashkilotga aylanish muvofiqlikka erishish va uni saqlab qolishning nisbatan oson va tejamkor vositasidir.

- **Qimmatbaho ma'lumotlar buzilishi ehtimolini kamaytirish.** Axborot aktivlaringizni etarli darajada himoya qilmaslik ma'lumotlarning buzilishiga olib kelishi mumkin, natijada rioya qilmaslik jarimaga tortiladi. Sizning brendingiz va obro'ingizga zarar yetkazish ham halokatli moliyaviy oqibatlarga olib kelishi mumkin. Ushbu axborot xavfsizligi standartlarini amalga oshirish buzilish ehtimolini kamaytirishi mumkin.

- **Xavfsizlik auditi talablariga javob berish.** ISO standartlari bo'yicha sertifikatlangan kompaniyalar sifatni ta'minlash bo'yicha umume'tirof etilgan amaliyotlarga rioya qilishlarini namoyish etadilar. Bu ko'pincha mijozlar va manfaatdor tomonlarning xavfsizlik tekshiruvlari bo'yicha so'rovlarini kamaytiradi, vaqt va hujjatlarni tejaydi.

- **Globallikni kengaytirish.** Ushbu axborot xavfsizligi standartlari butun dunyo bo'ylab kompaniyalar tomonidan tan olingan va joriy etilgan. Ushbu standartlarga rioya qilish sizga yangi global mijozlar va xalqaro biznes hamkorlarni orttirishga yordam beradi.

- **Mavjud mijozlar va manfaatdor tomonlar bilan ishonchni mustahkamlash.** ISO axborot xavfsizligi standartlariga rioya qilish sizning joriy biznes hamkorlaringiz bilan ishonchni mustahkamlashga yordam beradi. Ular siz axborot xavfsizligini boshqarishni ustuvor vazifa qilib qo'yayotganingizni bilib, ishonch va xotirjamlikka ega bo'ladilar.

- **Axborot xavfsizligini boshqarish bo'yicha yanada oqilona va ma'lumotli qarorlar qabul qilish.** Xatarni samarali boshqarish axborot texnologiyalarida sifat menejmentining muhim tarkibiy qismidir. ISO standartlariga rioya qilish ishonchli texnik, ma'muriy va operatsion nazoratni amalga oshirishni ta'minlaydi, bu AT xavfsizligining turli xavflarini baholash va ustuvorliklarni aniqlashni osonlashtiradi va yanada oqilona qarorlar qabul qiladi.

Nazorat savollari

1. Dasturiy ta'minot loyihalarini yaratishda qanday huquqiy me'yorlar mavjud?
2. ISO/IEC 27001 standarti nima va u dasturiy ta'minot loyihalariga qanday tatbiq etiladi?
3. Dasturiy ta'minot ishlab chiqishda qanday xalqaro standartlar mavjud?
4. Dasturiy ta'minot loyihalarini yaratishda qanday normativ hujjatlar va yo'riqnomalar mavjud?
5. ISO/IEC 90003 standarti nimani anglatadi va u qanday maqsadlar uchun qo'llaniladi?
6. Dasturiy ta'minot loyihalarida metodik ta'minotning o'rni va ahamiyati qanday?
7. ISO/IEC 25010 standarti bo'yicha dasturiy ta'minot sifatini qanday baholash mumkin?
8. Dasturiy ta'minot loyihalarida huquqiy risklar qanday boshqariladi?
9. Dasturiy ta'minot loyihalarida mualliflik huquqlarini himoya qilish usullari qanday?
10. Dasturiy ta'minot yaratishda xalqaro standartlarning qo'llanishi qanday afzalliklarni beradi?

5-MAVZU. BIZNES JARAYONLARI VA JARAYONLI BOSHQARUV

Reja:

1. Biznes jarayonlarni boshqarishda axborot tizimlari
2. Biznes jarayonlarni boshqarish tizimlari tarkibi
3. Biznes jarayonlarni boshqarish tizimlari (MRP, MRP II, ERP)

Biznes-jarayon – bu nazorat harakati va resurslarni ishlab chiqarishning kirishlari iste'molchilar uchun qimmatli bo'lgan resurslarga aylantirilishi, maqsadli va tartibga solinadigan korxonalar tizimi. Biznes jarayonlari: *o'zaro bog'liq, ishlab chiqarish, ta'minlovchi, boshqaruv jarayonlari va jarayonlariga* bo'linadi.

Jarayonli boshqaruvning mohiyati

Zamonaviy korxonalar ko'p profilli faoliyati, bo'linmalarining hududlar bo'yicha joylashganligi, sheriklar bilan ko'p sonli kooperativ aloqalariga ega bo'lgan murakkab tarkibga egadir. Shuning bilan birga bozor talablarining doimiy o'zgarishi, tovar va xizmatlarni ishlab chiqarishini iste'molchilar va mijozlarning individual ehtiyojlariga qaratilganligi, texnik imkoniyatlarni uzluksiz takomillashtirish va kuchli raqobat korxonalarning moliyaviy-xo'jalik faoliyati dinamikligini oshiradi va korxonadan bozor talablariga moslasha oladigan menejment tizimini yaratishga undaydi.

Korxona faoliyat turi quyidagi farqli belgilarga ega:

- Aniq shakllantirilgan maqsadlar doirasida mahsulotlar (xizmatlar) sotish yo'li bilan bozor vazifalarini mustaqil hal etadi;
- qat'iy belgilangan tashqi raqobatchilarga ega;
- asosiy xo'jalik funktsiyalarini amalga oshirishda nisbiy xo'jalik mustaqilligiga ega (ishlab chiqarish, sotish).

Korxonalarning faoliyat turi tarkibi va mohiyatini aniqlashda asos bo'lib, “qo'shimcha qiymat yaratish zanjiri” kontseptsiyasi hisoblanadi. Qo'shilgan qiymatni yaratish zanjiri korxona tarkibiy bo'linmalarining funktsiyalari ketma-ketligini xuddi aniq iste'mol qiymatini yaratishda bajariladigandek tartibda o'z ichiga oladi. Istalgan asosiy faoliyat turi uchun qo'shilgan qiymat yaratilishi zanjirining o'ziga xos funktsiyalari bo'lib xarid qilish, ishlab chiqarish, etkazib berish, mahsulot sotish va xizmat ko'rsatish funktsiyalari hisoblanadi. Qo'shilgan qiymat zanjirini

– foyda markazi xuquqiga ega bo‘lgan korxona amalga oshiradi. Qo‘shilgan qiymat yaratish zanjirining ayrim funktsiyalarini xarajatlar markazi deb nomlangan tarkibiy bo‘limlar bajaradilar.

Korxonalarda jarayonli boshqaruvni qo‘llash zarurligi shundaki, uning yordamida biznes-jarayonlarining masalalarini shunday hal qilish kerakki, natijada korxonaning strategik maqsadlariga erishish mumkin bo‘lsin. Biznes-jarayonlarini ajratib olish, ularning chegaralarini, funktsiyalarni bajargan bo‘linmalarning tashkiliy javobgarligini, o‘zaro bog‘liqligini shakllantiruvchi funktsiyalar tarkibini aniqlaydi. Biznes-jarayonlarini ajratib olish quyidagi belgilarda aks etadi.

- Biznes-jarayoni egasi (menejer) – jarayon natijalari va tashkil etishga hamda uning tarkibini o‘zgartirishga ma’sul shaxs hisoblanadi;
- biznes-jarayoni oqimi – jarayonning kirish (dastlabki obyektlar) va chiqishi (natijalar)ni aniqlaydi;
- jarayonning tashqi muhiti. Istalgan biznes-jarayonning etkazib beruvchilari, ya’ni ular jarayon kirishiga dastlabki material etkazib beradilar va chiqish natijalarini iste’mol qiluvchi ma’lum mijozlar mavjud. Etkazib beruvchilar va mijozlar ham ichki, ham tashqi bo‘lishi mumkin.
- Biznes-jarayoni interfeysi – biznes-jarayonning boshqa jarayonlar bilan birgalikda faoliyat olib borishga ko‘maklashuvchi obyektlar to‘plami.

Xalqaro ISO standartining 9000:2000 seriyasida “jarayonlar yondashuv” (process approach) termini kiritilgan bo‘lib, unda “jarayonli yondashuvga” quyidagi ta’rif berilgan: "Agar faoliyat va u bilan bog‘liq resurslar jarayonlar sifatida boshqarilsa, kutilgan natijaga yanada samaraliroq erishiladi”.

ISO 9000:2000 standartida "jarayon" atamasiga quyidagicha ta’rif berilgan: “Kirishlarni chiqishlarga o‘zgartirishda resurslardan foydalanadigan istalgan faoliyat turi yoki faoliyat turlari to‘plami jarayon sifatida qaralishi mumkin”.

Korxonalarda jarayonli boshqaruvni qo‘llashning asosiy maqsadi¹ “... “ortiqcha” operatsiyalarni bartaraf qilish, biznesni olib borish jarayonlarini kamaytirish va mijoz uchun iste’mol qiymatini yaratuvchi birinchi kategoriyali biznes-jarayonlarini bajarilish samaradorligini oshirishdir”.

Jarayonli boshqarish asosiga qo'yilgan imkoniyatlarni amalga oshirish uchun quyidagi tadbirlar samarali hisoblanadi:

- soddalashtiruvchi biznes-jarayonlar (hujjatlarni uzatish vaqtini qisqartirish, oddiy bog'liqliklar ustuvorligi);

- birlashtiruvchi funktsiyalar (muvofiglashtirish funktsiyalarini ma'lum darajada qisqartirish, axborot uzatish vaqtini qisqartirish);

- nazoratning kamayuvchi funktsiyasi (bajariladigan ishga mas'uliyatni aniq o'rnatish, holatlarni kamaytirish).

Xalqaro standartlar turli xil korxonalarda jarayonli boshqarish uslubiylatini qo'llashni taklif etadilar. Bular:

A) *“Korxonada jarayonlarni identifikatsiyalash (bixillashtirish)”*

- 1) Korxona maqsadini aniqlash.

- 2) Korxona vazifalari va siyosatini aniqlash.

- 3) Korxonadagi jarayonlarni aniqlash.

- 4) Jarayonlar ketma-ketligini aniqlash.

- 5) Jarayonlar egalarini aniqlash.

- 6) Jarayonlar bo'yicha hujjatlarni aniqlash.

B) *“Jarayonni rejalashtirish”*.

- 1) Jarayon doirasida ishlarni aniqlash.

- 2) O'lchovlar va monitoringga bo'lgan talablarni aniqlash.

- 3) Zarur resurslarni aniqlash.

- 4) Qo'yilgan maqsadlarga qarab jarayon va uning ishlarini tekshirish.

Jarayonli boshqaruv turlari

Korxonalarda biznes-jarayonlarini modellashtirish, tahlil qilish va bashoratlash asosida uning qanchalik samarali faoliyat yuritayotganligini aniqlash zarur.

Jarayonli boshqaruvning korxonalarda tadbiq etilishi, har bir jarayonning chegarasi, reglamenti, iste'molchisi va egasini aniq belgilab beradi.

Jarayonli boshqaruvning quyidagi turlari mavjud¹: loyiha ustida ishlash bilan bog'liq jarayonlar, ishlab chiqarish jarayonlari, taqsimot jarayonlari va mijozlarga xizmat ko'rsatish jarayonlari.

Loyiha ustida ishlash bilan bog'liq jarayonlar. Qoidaga ko'ra, bunday jarayonlar bir kishi yoki odamlar guruhi tomonidan bajariladi.

Bunga misol bo'lib, yangi mahsulotni ishlab chiqarish va ma'muriy jarayonlar xizmat qiladi. Odatda, bunday jarayonlar tahlili loyihalarni boshqarish usullaridan foydalanib olib boriladi. Shunga qaramasdan, tahlil natijasida olinadigan resurslarga bo'lgan talablar va jarayon to'liq siklini baholash ko'plab o'zaro bog'liq jarayonlarning paydo bo'lishiga olib keladi.

Loyihaning aniq modelini yaratishda birinchi navbatda quyidagi elementlarni modellashtirish qarab chiqiladi: ustuvor yo'nalishlar, darhol qilinishi lozim bo'lgan ishlarni bajarish, ishlarni smenalarga bo'lish, ishlarni nazorat qilish.

Ishlab chiqarish jarayonlari. Ishlab chiqarish jarayonlarining natijasi bo'lib, guruhlariga bo'lingan, katta hajmdagi turli xil mahsulotlar hisoblanadi. Bularga buyurtmalarni bajarish, to'lov schyotlari bo'limi va buyurtmalarni qayta ishlash bo'limining faoliyati misol bo'la oladi.

Guruhlariga bo'lish, guruhlarni birlashtirish, detallarni yig'ish, montaj qilish, sifatni nazorat qilish va brakni bartaraf etish ishlab chiqarish jarayoni tomonidan amalga oshiriladigan odatdagi funktsiyalar hisoblanadi. Ushbu funktsiyalarni aniq modellashtirish uchun oqimlarning alohida obyektlari va ularning belgilari to'g'risidagi axborotni doimo kuzatib borish mumkin.

Ishlab chiqarish jarayonlarini modellashtirishdan maqsad, qoidaga ko'ra tizimni barqarorlashtirishdan iborat, chunki ishlab chiqarilayotgan mahsulot ketma-ketligi takrorlanib turadi. Samaradorlikni tahlil etish jarayonida muhim vazifa bo'lib, ishlarning beqarorlik davrini aniqlash va buzilishlar, chetga chiqishlarni bartaraf etish hisoblanadi.

Taqsimlash jarayonlari. Taqsimlash jarayonlari yuklarni transportda tashish va etkazib berishni o'z ichiga oladi. Ushbu holda mahsulotlarning taqsimlash tarmog'ida turli nuqtalar orasida ko'chib yurishi yuz beradi. Transportirovkaning etkazib berishdan asosiy farqi shundaki, transportirovkada oqim obyektlari – bu tovarlar emas, balki odamlardir.

Transportirovkaning an'anaviy jarayonlarini jamoat transporti tizimidan topish mumkin. Etkazib berishning an'anaviy jarayonlariga ishlab chiqarilgan mahsulotni sotish, pochta va tovarlarni iste'molchilarga etkazib berish kiradi.

Mijozlarga xizmat ko'rsatish jarayonlari. Mijozlarga xizmat ko'rsatish jarayonlari biznes-jarayonlarini modellashtirishda eng muhim soha hisoblanadi. Chunki, xizmat ko'rsatish jarayonida kutish vaqti ishlov berish umumiy vaqtining 95% ga etishi mumkin.

Jarayon - bu o'zaro bog'liq faoliyat turlarining tartiblashtirilgan, maqsadga muvofiq to'plami bo'lib, ma'lum texnologiya bo'yicha kirishlar va resurslarni chiqishlar (mahsulotlar) ga aylantiradi hamda iste'molchi uchun afzallik va qadriyat kasb etadi.

Asosiy va yordamchi jarayonlar. Har bir korxona faoliyati marketing, rejalashtirish, sotish va sotishdan keyingi xizmat ko'rsatish jarayonlari ketma-ketligidan iborat. Barcha jarayonlar quyidagi ajralib turuvchi xususiyatlarga ega:

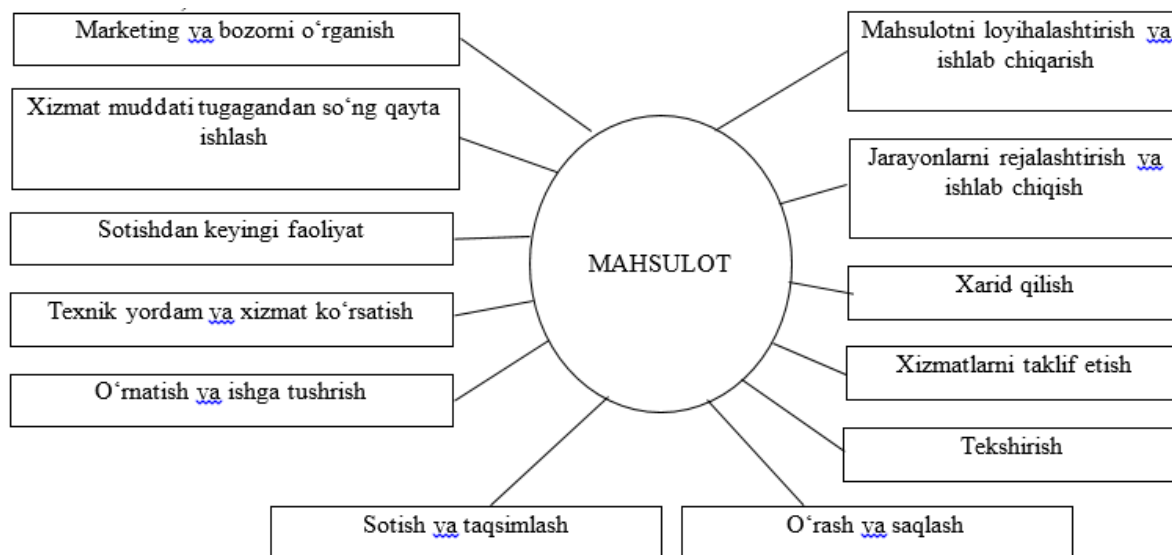
- Jarayon egasi – jarayonni borishi va natijalariga mas'ul mansabdor shaxs;
- Resurslar – jarayonni o'tkazish uchun uning egasi ixtiyoriga ajratilgan resurslar; u o'z ichiga – jihozlar (ishlab chiqarish, nazorat-o'lchov, ofis va boshqalar), personal, alohida bino, materiallar, moliyaviy mablag'lar, hujjatlarni olishi mumkin;
- Jarayon parametrlari – jarayon qanchalik samarali bajarilayotgani to'g'risidagi xarakteristika (axborot);
- Iste'molchi – jarayon samaradorligini baholash uchun mo'ljallangan, jarayon natijalari iste'molchisi (ichki va tashqi);
- Jarayon kirishlari - resurslar yoki axborot ko'rinishida aks etadi.
- Jarayon chiqishlari – mahsulot yoki axborot ko'rinishida aks etadi. Ko'p hollarda bir jarayonning kirishi boshqa jarayonning chiqishi hisoblanadi
- Korxona jarayonlari tarmog'i – korxona o'zaro bog'liq va o'zaro kelishgan jarayonlarini bir tizimga keltirilgan to'plami hisoblanadi.

Korxonadagi har bir jarayon bir necha kichik jarayonlarga bo'linishi mumkin. Bunday holda jarayonlarni tahlil qilishda asosiy jarayonning mohiyati to'liq aniqlab olinadi. Korxonalardagi barcha jarayonlarni ikki guruhga bo'lishi mumkin:

1. Asosiy jarayonlar.
2. Yordamchi jarayonlar

Asosiy jarayonlar. Korxonadagi asosiy jarayonlar mahsulotni yaratish va ishlab chiqarish davrida yuzaga keladi. Jarayonlar dastavval loyiha, marketing axboroti, keyinchalik esa moddiy obyekt (detallar, tovarlar, dasturiy mahsulotlar va hokazo) ko'rinishida bo'ladi. Asosiy

jarayonlarni yordamchi jarayonlardan ajratilgan sxemasi sifatida mahsulotning "hayotiy sikli" sxemasidan foydalanish mumkin(1-chizma).

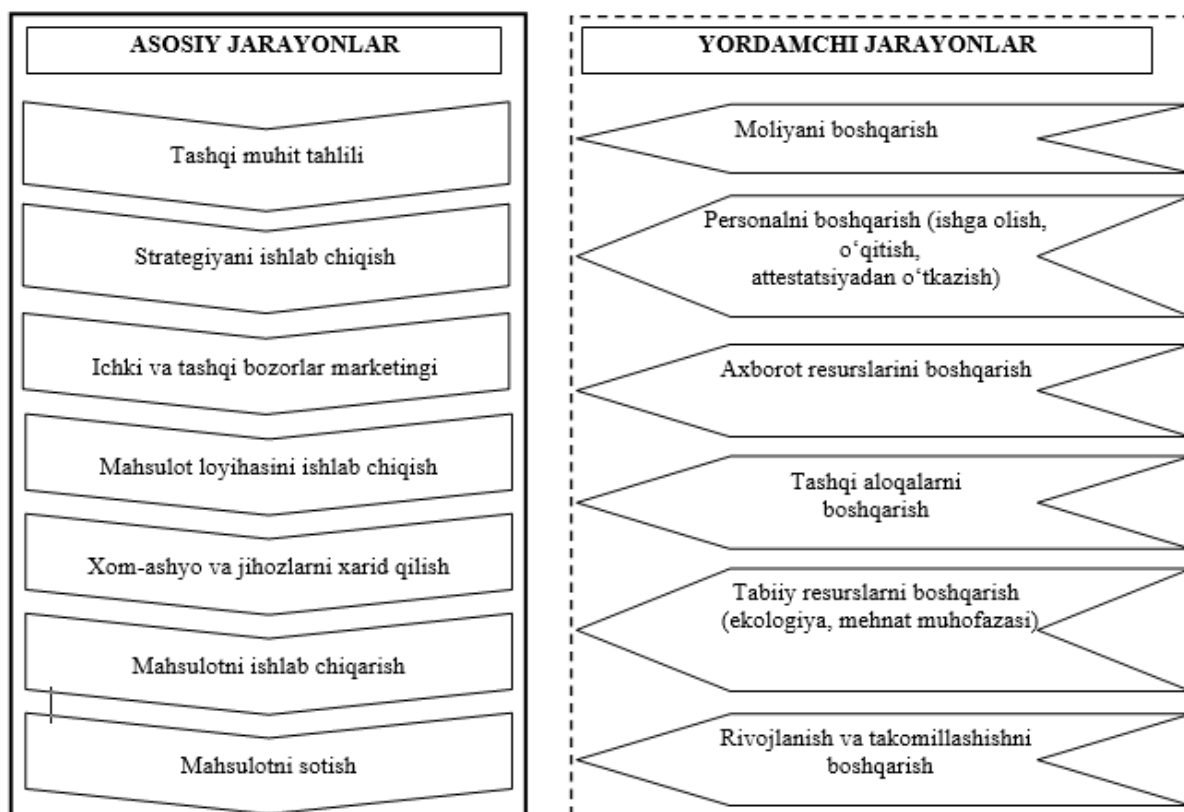


1-chizma. Mahsulotning hayotiy sikli

Yordamchi jarayonlar. Ushbu jarayonlar mahsulotni yaratishga bevosita bevosita ta'sir ko'rsatmaydilar, lekin asosiy jarayonlarning bir maromda faoliyat olib borishini ta'minlash uchun mo'ljallangan. Bunday jarayonlarga quyidagilar kiradi:

- personalni tayyorlash, o'qitish va attestatsiyadan o'tkazish;
- hujjatlarni boshqarish jarayoni. Ushbu jarayon korxonadagi barcha jarayonlarning ishini samarali ta'minlashda qatnashuvchi ayrim tarkibiy bo'linmalar hamda butun korxona faoliyatini muvofiqlashtiruvchi hujjatlarni ishlab chiqish, tasdiqlash va yuritishning tartib va qoidalarini o'rnatadi;
- ta'minlash jarayonlari. Yordamchi jarayonlar asosiy jarayonlarning ishini ta'minlaydi (jihozlarga servis xizmat ko'rsatish, energoresurslar va ishlab chiqarish muhiti bilan ta'minlash, ofis ishini ta'minlash, axborot bilan ta'minlash, moliyaviy ta'minlash va boshqalar).

Asosiy va yordamchi jarayonlarning bo'linishi quyidagi 2-chizmada keltirilgan.



2-chizma. Korxonada jarayonlar tarmog'ini ajratish

2-chizmada keltirilgan jarayonlarni ajratish tahlil masalalari uchun qulay hisoblanadi, ammo amaliyotda ushbu jarayonlar bir-birlari bilan o'zaro bog'lanib ketgan va doimo bir-birini taqozo etib turadi.

Ma'lumki, har bir jarayon boshqa bir jarayon bilan to'g'ridan-to'g'ri aloqada bo'ladi. Biror jarayonning kirishi boshqa jarayonga chiqish bo'lib xizmat qiladi. Ammo dastlabki jarayondan olingan natija keyingi jarayonga ta'sirini qanday aniqlash mumkin? – degan savol paydo bo'ladi. Buni aniqlash uchun esa teskari aloqa tamoyilidan foydalaniladi.

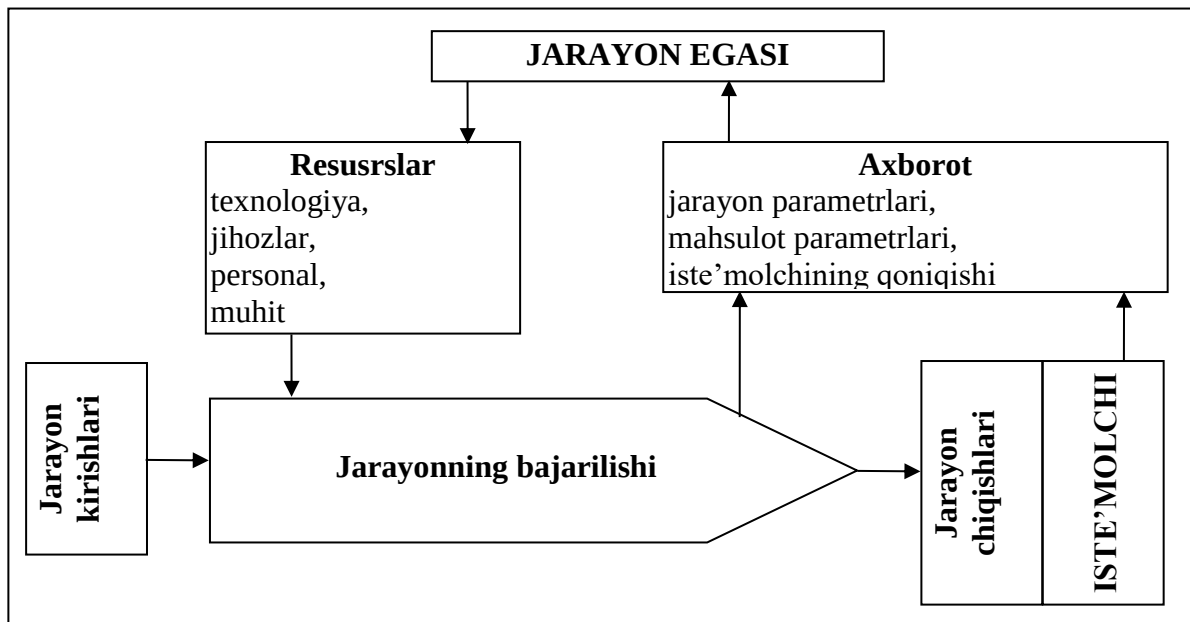
Jarayonlarda teskari aloqalarning soddalashtirilgan sxemasi quyidagi 3-chizmada keltirilgan.

2-chizmada keltirilgan jarayonlarni ajratish tahlil masalalari uchun qulay hisoblanadi, ammo amaliyotda ushbu jarayonlar bir-birlari bilan o'zaro bog'lanib ketgan va doimo bir-birini taqozo etib turadi.

Ma'lumki, har bir jarayon boshqa bir jarayon bilan to'g'ridan-to'g'ri aloqada bo'ladi. Biror jarayonning kirishi boshqa jarayonga chiqish bo'lib xizmat qiladi. Ammo dastlabki jarayondan olingan natija keyingi jarayonga ta'sirini qanday aniqlash mumkin? – degan savol

paydo bo‘ladi. Buni aniqlash uchun esa teskari aloqa tamoyilidan foydalaniladi.

Jarayonlarda teskari aloqalarning soddalashtirilgan sxemasi quyidagi 3-chizmada keltirilgan.



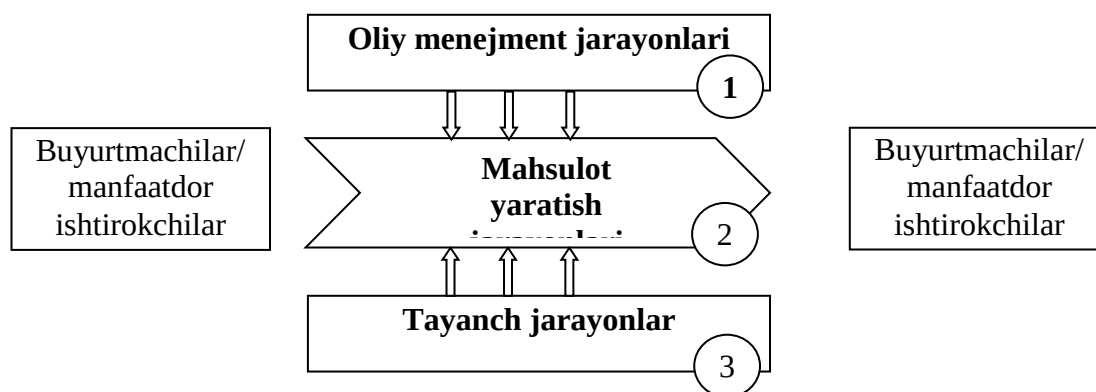
3-chizma. Jarayon va teskari aloqaning soddalashtirilgan sxemasi

3-chizmada keltirilgan sxema deyarli barcha ishlab chiqarish va xizmat ko‘rsatish korxonalarida doimiy ravishda amalga oshiriladi. Bu yerda teskari aloqa sifatida jarayonning chiqishlaridan va iste‘molchidan kelayotgan axborot hisoblanadi. Ushbu axborotlar asosida jarayon egasi jarayonning qanday borayotganligini va qanday natija bilan tugaganligini aniqlashi mumkin hamda jarayon to‘g‘risida boshqaruv qarorini qabul qiladi.

Korxona menejmenti tizimida jarayonlarning quyidagi uchta turi mavjud¹:

1. Oliy menejment jarayonlari.
2. Mahsulot yaratish jarayonlari.
3. Tayanch jarayonlar.

Ushbu jarayon turlari quyidagicha bir-biri bilan bog‘langan (4-chizma).

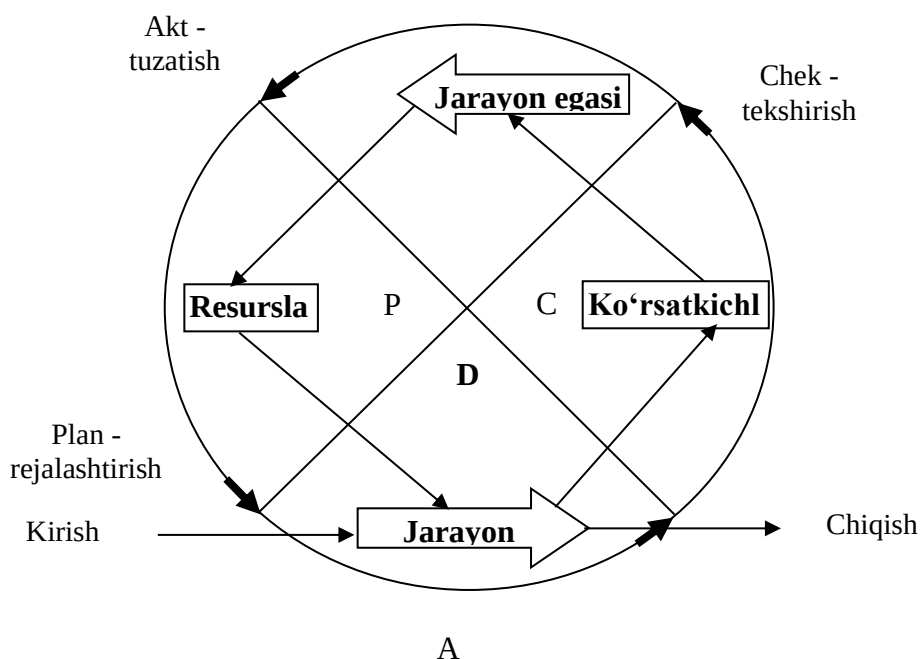


- ① – Strategik rejalashtirish, moliyaviy menejment...
- ② – Marketing, shartnomalar tuzish, xarid qilish, ishlab chiqarish...
- ③ – resurslarni boshqarish, jihozlarga xizmat ko'rsatish, axborot ta'minoti...

4-chizma. Korxonalarda jarayonlarning o'zaro bog'liqligi sxemasi

Albatta, korxonalar uchun asosiy maqsad mahsulotlar ishlab chiqarib, iste'molchilarning ehtiyojlarini qondirishdir. Shuning uchun ham oliy menejment jarayonlari va tayanch jarayonlar mahsulot yaratish jarayonlariga xizmat qiladilar.

Yuqorida ko'rib o'tilganidek, jarayonlarni boshqarish murakkab vazifa hisoblanadi. Chunki ko'pchilik jarayonlar siklli xarakterga ega, ya'ni ular doimiy ravishda takrorlanib turadi. Quyidagi 5-chizmada jarayonni boshqarish sikli keltirilgan.



5-chizma. Jarayonni boshqarish sikli.

5-chizmadan ko‘rish mumkinki, rejalashtirish bosqichida (Plan) jarayon egasi korxona maqsadlari va mijozlar talabiga mos ravishda jarayonning reja ko‘rsatkichlarini bajarilishini ta‘minlash uchun aniq maqsadlar qo‘yadi va buning uchun resurslarni taqsimlaydi. D - bu jarayonning bajarilishi. Bajarilish vaqtida mahsulot va jarayon monitoringi o‘tkaziladi, mahsulotga bo‘lgan talablar va maqsadlarga bo‘lgan munosabatlar bo‘yicha natijalar o‘lchanadi. Me‘yordan chetga chiqish yuz berganda, jarayon egasi va uning xodimlari ularning paydo bo‘lish sabablarini tahlil qiladilar (Chek). Jarayon egasi o‘tkazilgan tahlil asosida me‘yordan chetga chiqish sabablarini aniqlaydi va ularni tuzatish tadbirlarini (Act) ishlab chiqadi. Ushbu tadbirlar me‘yordan chetga chiqishlarni bartaraf etadi, bu esa jarayonning yaxshilanishiga olib keladi.

Barcha turdagi korxonalar, firmalar va tashkilotlardagi asosiy va yordamchi jarayonlar boshqariluvchi jarayonlar hisoblanadi. Jarayonlarning boshqaruvchanligi negizida quyidagilar yotadi:

- Jarayon egasini tayinlash;
- jarayon egasi tomonidan jarayonni muvaffaqiyatli bajarilishi uchun barcha resurslarni olishi;
- jarayon egasi tomonidan jarayon borishi to‘g‘risida, mahsulot parametrlari va iste‘molchining qoniqishi to‘g‘risida obyektiv axborot to‘plash tizimini yaratish;
- jarayonning pirovard natijalari bo‘yicha maksimal samaradorlikka erishish uchun jarayon egasi tomonidan boshqarish ta’sirlarini (resurslarni boshqarish) ishlab chiqishi va qabul qilishi.

3-chizma istalgan darajadagi jarayonni tavsiflash uchun qo‘llanilishi mumkin.

Nazorat savollari

1. Biznes jarayonlari nima va ularning asosiy xususiyatlari qanday?
2. Jarayonli boshqaruv nima va u qanday afzalliklarni taqdim etadi?
3. Biznes jarayonlarni modellashtirish va tahlil qilish usullari qanday?
4. Biznes jarayonlarni optimallashtirishning qanday usullari mavjud?
5. Biznes jarayonlarini boshqarishda qanday zamonaviy

texnologiyalar qo'llaniladi?

6. Biznes jarayonlarini baholash va samaradorligini o'lchash qanday amalga oshiriladi?

7. Biznes jarayonlarini boshqarish jarayonida qanday qiyinchiliklar yuzaga kelishi mumkin?

8. Lean va Six Sigma kabi jarayonli boshqaruv metodologiyalari qanday ishlaydi?

9. Biznes jarayonlarni boshqarishning muvaffaqiyatli amalga oshirilishi uchun qanday muhim omillar mavjud?

10. Biznes jarayonlarini avtomatlashtirish qanday afzalliklar va kamchiliklarga ega?

6-MAVZU. DASTURIY TA'MINOT HAYOTIY SIKLI ASOSIY JARAYONLARI

Reja:

1. Dasturiy ta'minot jarayoni
2. Professional dasturiy ta'minotni ishlab chiqish
3. Integratsiya va konfiguratsiya
4. Dasturiy ta'minotni loyihalash

Dasturiy ta'minot jarayoni - bu dasturiy ta'minot tizimini ishlab chiqarishga qaratilgan harakatlar to'plami. Avvalgi mavzuda muhokama qilganidek, dasturiy ta'minot tizimlarining ko'plab turlari mavjud va ularning barchasiga tegishli bo'lgan universal dasturiy injiniring usuli mavjud emas. Bundan kelib chiqadiki, universal qo'llaniladigan dasturiy ta'minot jarayoni ham mavjud emas. Turli kompaniyalarda qo'llaniladigan jarayon ishlab chiqilayotgan dasturiy ta'minot turiga, dasturiy ta'minot mijozining talablariga va dasturiy ta'minotni dastur tuzuvchilarning ko'nikmalariga bog'liq.

Har xil dasturiy ta'minot jarayonlari mavjud bo'lsa-da, ularning barchasi qaysidir shaklda to'rtta asosiy dasturiy injiniring faoliyatini o'z ichiga olishi kerak:

1. Dasturiy ta'minot spetsifikatsiyasi. Dasturiy ta'minotning funktsionalligi vauning ishlashidagi cheklovlar aniqlanishi kerak.
2. Dasturiy ta'minotni ishlab chiqish. Spetsifikatsiyaga javob beradigan dasturiy ta'minot ishlab chiqarilishi kerak.
3. Dasturiy ta'minotni tekshirish. Dasturiy ta'minot mijoz talablarini bajarishini ta'minlash uchun tekshirilishi kerak.
4. Dasturiy ta'minot evolyutsiyasi. Dasturiy ta'minot mijozlarning o'zgaruvchan ehtiyojlarini qondirish uchun rivojlanishi kerak.

Ushbu jarayonlar murakkab jarayonlar bo'lib, ular talablarni tekshirish, arxitekturani loyihalashtirish va modullarni sinovdan o'tkazish kabi kichik faoliyatlarni o'z ichiga oladi. Jarayonlar, shuningdek, ishlab chiqarish faoliyatini qo'llab-quvvatlaydigan dasturiy ta'minot konfiguratsiyasini boshqarish va loyihani rejalashtirish kabi boshqa jarayonlarni ham o'z ichiga oladi.

Jarayonlarni tavsiflash va muhokama qilishda odatda ma'lumotlar modelini ko'rsatish va foydalanuvchi interfeysini loyihalash va bu harakatlarni tartibga solish kabi harakatlar haqida gapiriladi. Ammo jarayonlarni tavsiflashda kim ishtirok etayotganini, nima ishlab

chiqarilishini va harakatlar ketma-ketligiga ta'sir qiluvchi shartlarni tavsiflash ham muhimdir:

1. Mahsulotlar yoki amaliy natijalar jarayon faoliyatining natijasidir. Masalan, arxitekturaning loyihalash faoliyatining natijasi dasturiy ta'minot arxitekturasining modeli bo'lishi mumkin.

2. Rollar jarayonda ishtirok etayotgan odamlarning mas'uliyatini aks ettiradi. Rollarga misol sifatida loyiha menejeri, konfiguratsiya menejeri va dasturchikiradi.

3. Oldindan belgilangan va keyin shartlar - bu jarayon yoki mahsulot ishlab chiqarilishidan oldin va keyin bajarilishi kerak bo'lgan shartlar. Misol uchun, arxitektura loyihasi boshlanishidan oldin, iste'molchi barcha talablarni tasdiqlagan shartlari bo'lishi mumkin; Ushbu faoliyat tugagandan so'ng, arxitekturaning tavsiflovchi UML modellari ko'rib chiqilish shartlari bo'lishi mumkin.

Dasturiy ta'minot jarayonlari murakkab bo'lib, barcha intellektual va ijodiy jarayonlar kabi, qarorlar va mulohazalar qabul qiladigan insonlarga tayanadi. Barcha turdagi dasturiy ta'minot uchun to'g'ri keladigan universal jarayon yo'qligi sababli, ko'pgina dasturiy ta'minot kompaniyalari o'zlarining ishlab chiqish jarayonlarini ishlab chiqqanlar. Jarayonlar tashkilotdagi dasturiy ta'minotni ishlab chiquvchilarning imkoniyatlari va ishlab chiqilayotgan tizimlarning xususiyatlaridan foydalanish uchun modifikatsiya qilib boriladi. Xavfsizlik nuqtai nazaridan tizimlar uchun muhim bo'lgan batafsil yozuvlar saqlanadigan juda tuzilmaviy ishlab chiqish jarayoni talab qilinadi. Tez o'zgaruvchan talablarga ega biznes tizimlari uchun yanada moslashuvchan, tezkor jarayon yaxshiroq bo'lishi mumkin.

Professional dasturiy ta'minotni ishlab chiqish bu boshqariladigan faoliyatdir, shuning uchun rejalashtirish barcha jarayonlarning ajralmas qismidir. Rejaga asoslangan jarayonlar - bu jarayonning barcha harakatlari oldindan rejalashtirilgan va taraqqiyot ushbu rejaga muvofiq o'lchanadigan jarayonlar. Men mavzuda muhokama qiladigan tezkor jarayonlarda, dasturiy ta'minot ishlab chiqilayotganda rejalashtirish bosqichma-bosqich va doimiydir. Shuning uchun mijoz yoki mahsulotning o'zgaruvchan talablarini aks ettirish uchun jarayonni o'zgartirish osonroq. Boehm va Turner (Boehm and Turner 2004) tushuntirganidek, har bir yondashuv har xil turdagi dasturlarga mos keladi. Umuman olganda, katta tizimlar uchun siz rejaga asoslangan va tezkor jarayonlar o'rtasidagi muvozanatni topishingiz kerak.

Universal dasturiy jarayoni mavjud bo'lmasa-da, ko'plab tashkilotlarda jarayonni yaxshilash uchun imkoniyatlar mavjud. Jarayonlarni tashkil etishda eskirgan texnikasidan foydalanishi yoki zamonaviy dasturiy injiniringining eng yaxshi amaliyotidan foydalanmasligi mumkin. Darhaqiqat, ko'pgina tashkilotlar hali ham dasturiy ta'minotni ishlab chiqishda dasturiy injiniring usullaridan foydalanmaydi.

Ular UML modellashtirish va testlash asosida ishlab chiqish kabi usullarni joriy etishorqali o'z jarayonlarini yaxshilashlari mumkin.

Dasturiy ta'minot jarayonlari modellari

1-mavzuda ta'kidlanganidek, dasturiy ta'minot jarayoni modeli (ba'zan dasturiy ta'minotni ishlab chiqishning hayot sikli modeli deb ataladi) dasturiy ta'minot jarayonining soddalashtirilgan tasviridir. Har bir jarayon modeli ma'lum bir nuqtai nazardan jarayonni ifodalaydi va shuning uchun bu jarayon haqida faqat qisman ma'lumot beradi. Masalan, jarayon faoliyati modeli faoliyat va ularning ketma- ketligini ko'rsatadi, lekin bu faoliyatda ishtirok etuvchi insonlarning rolini ko'rsatmasligi mumkin. Ushbu mavzuda bir qancha umumiy jarayon modellarini (ba'zan jarayon paradigmalari deb ataladi) tanishtiriladi va ular arxitektura nuqtai nazaridan taqdim etiladi. Ya'ni, jarayonning asosi ko'riladi, lekin jarayonning tafsilotlari to'liq ko'rib chiqilmaydi.

Ushbu umumiy modellar dasturiy ta'minotni ishlab chiqishga turli yondashuvlarni tushuntirish uchun ishlatilishi mumkin bo'lgan dasturiy jarayonlarning yuqori darajadagi, abstrakt tavsifidir. Bu mavzuda quyidagi umumiy jarayon modellari ko'riladi:

1. Sharshara modeli. Bu model spetsifikatsiya, ishlab chiqish, tasdiqlash va evolyutsiyaning asosiy jarayonlarini o'z ichiga oladi va ular talablarni spetsifikatsiya qilish, dasturiy ta'minotni loyihalash, amalga oshirish va testdan o'tkazish kabi alohida jarayon bosqichlari sifatida taqdim etiladi.

2. Incremental (bosqichma-bosqich) rivojlanish. Ushbu yondashuv spetsifikatsiya, ishlab chiqish va tasdiqlash faoliyatini birlashtiradi. Tizim bir qancha versiyalar (qo'shimchalar) bilan ishlab chiqilgan bo'lib, har bir versiya oldingi versiyadan ko'ra takomillashgan va qo'shimcha funktsionallikka ega bo'ladi.

3. Integratsiya va konfiguratsiya. Ushbu yondashuv qayta foydalanish mumkin bo'lgan komponentlar yoki tizimlarning

mavjudligiga tayanadi. Tizimni ishlab chiqish jarayoni ushbu komponentlarni yangi sozlamalarda foydalanish uchun sozlash va ularni tizimga integratsiyalashga qaratilgan.

Aytilganidek, dasturiy ta'minotni ishlab chiqishning barcha turlariga mos keladigan universal jarayon modeli yo'q. To'g'ri jarayon mijozning talablariga va normativ talablarga, dasturiy ta'minot foydalaniladigan muhitga va ishlab chiqilayotgan dasturiy ta'minot turiga bog'liq. Masalan, xavfsizlik nuqtai nazaridan dasturiy ta'minot odatda sharshara modeli yordamida ishlab chiqiladi, chunki amalga oshirish bosqichidan oldin ko'pgina tahlil va hujjatlar talab qilinadi. Xozirda dasturiy ta'minot mahsulotlari ishlab chiqishda incremental modelidan ko'p foydalanilmoqda. Mavjud tizimlarni sozlash va ularni zarur bo'lgan funksionallikka ega yangi tizim yaratish uchun integratsiyalash orqali biznes tizimlari tobora rivojlanib kelmoqda.

Amaliy dasturiy ta'minot jarayonlarining aksariyati umumiy modelga asoslanadi, lekin ko'pincha boshqa modellarning xususiyatlarini ham o'z ichiga olishi mumkin. Bu, ayniqsa, katta tizim uchun to'g'ri keladi. Katta tizimlar uchun barcha umumiy jarayonlarning eng yaxshi xususiyatlarini birlashtirish mantiqan to'g'ri keladi. Ushbu talablarni bajarish uchun dasturiy ta'minot arxitekturasini loyihalashda asosiy tizim talablari haqida ma'lumotga ega bo'lish kerak. Buni bosqichma-bosqich rivojlantirib bo'lmaydi. Kattaroq tizimdagi quyi tizimlar turli yondashuvlar yordamida ishlab chiqilishi mumkin. Tizimning yaxshi tushunilgan qismlari sharsharaga asoslangan jarayon yordamida aniqlanishi va ishlab chiqilishi yoki konfiguratsiya uchun tayyor tizimlar sifatida sotib olinishi mumkin. Tizimning oldindan belgilash qiyin bo'lgan boshqa qismlari har doim qo'shimcha yondashuv yordamida ishlab chiqilishi kerak. Ikkala holatda ham dasturiy ta'minot komponentlari qayta ishlatilishi mumkin.

Ushbu umumiy modellarning barchasini birlashtirib "universal" jarayon modellarini ishlab chiqish uchun turli xil urinishlar bo'lgan. Ushbu universal modellarning eng mashhurlaridan biri AQShning dasturiy ta'minot injiniring kompaniyasi Rational tomonidan ishlab chiqilgan Rational Unified Process (RUP) (Krutchen 2003) hisoblanadi. RUP moslashuvchan model bo'lib, ushbu fan doirasida o'tiladigan har qanday umumiy jarayon modellariga o'xshash jarayonlarni yaratish uchun turli yo'llar bilan yaratilishi mumkin. RUP ba'zi yirik dasturiy ta'minot kompaniyalari (xususan, IBM) tomonidan

qabul qilingan, ammo u ommalashmagan.

Sharshara modeli(kaskad modeli)

Dasturiy ta'minotni ishlab chiqish jarayonining birinchi nashr etilgan modeli yirik harbiy tizimda qo'llaniladigan muhandislik jarayonlari modellaridan olingan (Royce 1970). U 6.1-rasmda ko'rsatilganidek, dasturiy ta'minotni ishlab chiqish jarayonini bir necha bosqichlardan iborat. Bir bosqichdan ikkinchisiga o'tgan kaskad tufayli ushbu model sharshara modeli yoki dasturiy ta'minotning hayotiy sikli sifatida tanilgan. Sharshara modeli rejaga asoslangan jarayonga misoldir. Printsipial jihatdan, dasturiy ta'minotni ishlab chiqishni boshlashdan oldin barcha jarayonlar rejalashtirib chiqiladi.

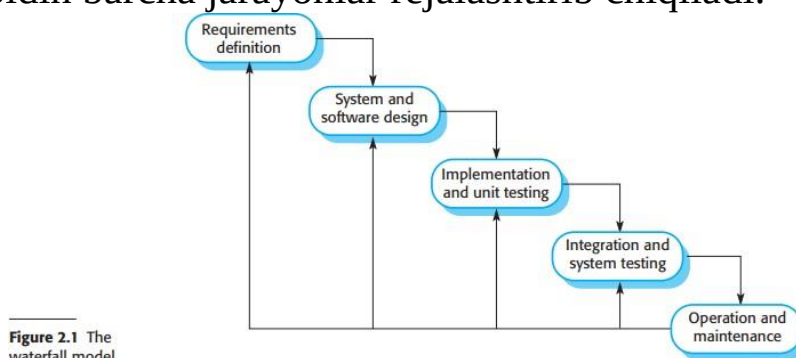


Figure 2.1 The waterfall model

6.1-rasm

Sharshara modelining bosqichlari dasturiy ta'minotni ishlab chiqishning asosiy faoliyatini bevosita aks ettiradi:

1. Talablarni tahlil qilish va aniqlash. Tizim xizmatlari, cheklovlari va maqsadi tizim foydalanuvchilari bilan maslahatlashgan holda belgilanadi. Keyin ular batafsil tavsiflanadi va tizim spetsifikatsiyasi sifatida xizmat qiladi.

2. Tizim va dasturiy ta'minotni loyihalashtirish. Tizimlarni loyihalash jarayoni apparat yoki dasturiy ta'minot tizimlariga talablarni taqsimlab beradi. Umumiy tizim arxitekturasini aniqlaydi. Dasturiy ta'minot loyihalashtirish jarayoni asosiy dasturiy ta'minot tizimining abstraktsiyalarini va ularning munosabatlarini aniqlash va tavsiflash vazifalarini o'z ichiga oladi.

3. Amalga oshirish va modullarni testdan o'tkazish. Ushbu bosqichda dasturiy ta'minot loyihasi dasturlar to'plami yoki dasturiy modullar sifatida ishlab chiqiladi. Modullarni testlash bosqichida har bir modul o'z spetsifikatsiyasigamos kelishini tekshirib chiqiladi.

4. Integratsiya va tizimni testlash. Alohida dastur modullari yoki

dasturlari dasturiy ta'minot talablari bajarilishini ta'minlash uchun to'liq tizim sifatida birlashtiriladi va testdan o'tkaziladi. Sinovdan so'ng dasturiy ta'minot tizimi mijozga etkazib beriladi.

5. Foydalanish va texnik xizmat ko'rsatish. Odatda, bu hayot siklining eng uzoq bosqichidir. Tizim o'rnatiladi va amaliy foydalanishga topshiriladi. Texnik xizmat ko'rsatish hayot siklining oldingi bosqichlarida aniqlanmagan xatolarni tuzatishni, tizim modullarini ishlab chiqishni yaxshilashni va yangi talablar aniqlanganda tizim xizmatlarini kengaytirishni o'z ichiga oladi.

Aslida, sharshara modelidagi har bir bosqichning natijasi tasdiqlangan ("imzolangan") bir yoki bir nechta hujjatdan iborat bo'ladi. Keyingi bosqich oldingibosqich tugamaguncha boshlanmasligi kerak. Yuqori ishlab chiqarish xarajatlarini talab qiladigan apparat ta'minotini ishlab chiqishda bu mantiqan to'g'ri narsa. Biroq, dasturiy ta'minotni ishlab chiqish uchun bu bosqichlar bir-biriga mos kelishi va ma'lumotlarni bir-biriga etkazib berib turishi lozim. Loyihalash jarayonida talablar bilan bog'liq muammolar aniqlanishi; kodlash paytida loyiha muammolari topilishi mumkin va boshqalar. Dasturiy ta'minot jarayoni, amalda, hech qachon oddiychiziqli model bo'lmaydi, balki bir bosqichdan ikkinchisiga qayta aloqa qilish xususiyatiga ega bo'lishi kerak.

Jarayon davomida yangi ma'lumotlar paydo bo'lganligi sababli, oldingi bosqichlarda ishlab chiqarilgan hujjatlar kerakli tarzda o'zgartirilib borishi kerak. Misol uchun, agar talabni amalga oshirish juda qimmat ekanligi aniqlansa, ushbu talabni olib tashlash uchun talablar hujjatini o'zgartirish kerak. Biroq, bu mijozning roziligini talab qiladi va umumiy ishlab chiqish jarayonini kechiktiradi.

Natijada, mijozlar ham, ishlab chiquvchilar ham dasturiy ta'minot spetsifikatsiyasini muddatidan oldin muzlatib qo'yishi mumkin, shunda unga boshqa o'zgartirishlar kiritilmaydi. Afsuski, bu muammolar keyinchalik hal qilish uchun qoldirilishi, e'tiborga olinmaganligi anglatadi. Talablarni muddatidan oldin muzlatish tizim foydalanuvchi xohlagan narsani qilmasligini anglatadi. Bu, shuningdek, noto'g'ri tuzilgan tizimlarga olib kelishi mumkin, chunki loyiha muammolari amalga oshirish uslublari orqali chetlab o'tish mumkin.

Hayot siklining yakuniy bosqichida (ishlash va texnik xizmat ko'rsatish) dasturiy ta'minot foydalanishga topshiriladi. Dasturiy

ta'minotning dastlabki talablarida xato va kamchiliklar aniqlanadi. Dastur va loyihalash xatolari paydo bo'ladi va yangi funksiyalarga bo'lgan ehtiyoj aniqlanadi. Shuning uchun tizim keyinchalik foydali bo'lib qolishi uchun rivojlantirib borilishi kerak. Ushbu o'zgarishlarni amalga oshirish uchun (dasturiy ta'minotga texnik xizmat ko'rsatish) avvalgi jarayon bosqichlari takroran bosib o'tiladi.

Aslida, dasturiy ta'minot moslashuvchan bo'lishi va ishlab chiqilayotgan vaqtda o'zgarishlarga moslashuvchan bo'lishi lozim. Tizimni qayta ishlash zaruriyati sharshara modeli tizimning ayrim turlariga mos kelishini anglatadi:

1. Dasturiy ta'minot apparat tizimlari bilan o'zaro ta'sir qilishi kerak bo'lgan o'rnatilgan tizimlar. Uskunaning moslashuvchan emasligi sababli, odatdasturiy ta'minotning funktsionalligi bo'yicha qarorlarni u amalga oshirilgungaqadar orqaga qoldirilmasligi lozim.

2. Dasturiy ta'minot spetsifikatsiyasi va loyihasini xavfsizligi va himoyalanganligining keng qamrovli tahlili talab etiladigan jiddiy tizimlar. Ushbu tizimlarni tahlil qilish uchun uning spetsifikatsiya va loyiha hujjatlari to'liq bo'lishi kerak. Spetsifikatsiya va loyihaning xavfsizligi bilan bog'liq muammolarni, odatda, amalga oshirish bosqichida tuzatish juda qimmatga tushadi.

3. Bir nechta hamkor kompaniyalar tomonidan ishlab chiqilgan kengroq injiniring tizimlarining bir qismi bo'lgan yirik dasturiy ta'minot tizimlari. Tizimlardagi apparat shunga o'xshash model yordamida ishlab chiqilishi mumkin va kompaniyalar apparat va dasturiy ta'minot uchun umumiy modeldan foydalanishni osonlashadi. Bundan tashqari, bir nechta kompaniyalar ishtirok etganda, turli quyi tizimlarning mustaqil rivojlanishi uchun to'liq spetsifikatsiyalar talab qilinishi mumkin.

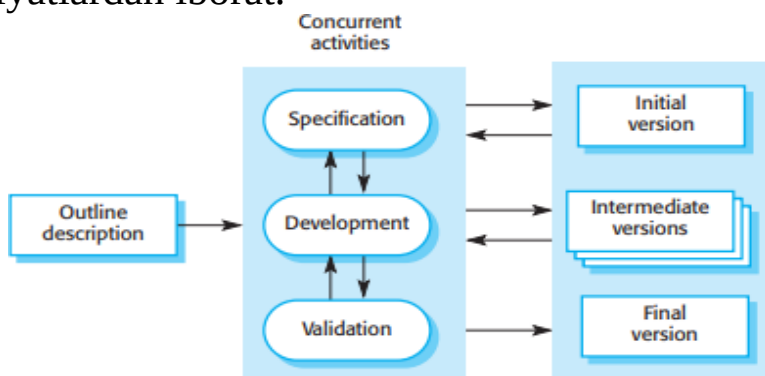
Sharshara modeli norasmiy jamoaviy muloqotda va dasturiy ta'minot talablari tez o'zgarib turadigan vaziyatlarda to'g'ri kelmaydi. Ushbu tizimlar uchun iterativ rivojlanish va tezkor usullar yaxshiroq hisoblanadi.

Sharshara modelining muhim varianti tizimni rasmiy ishlab chiqish bo'lib, unda tizim spetsifikatsiyasining matematik modeli yaratiladi. Keyinchalik, ushbu model izchillikni saqlaydigan matematik ifodalar yordamida bajariladigan kodga aylantiriladi. Rasmiy ishlab chiqish jarayonlari, masalan, B usuli (Abrial 2005, 2010) asosan kuchli himoyalanganlik, ishonchlilik yoki xavfsizlik talablariga ega

bo'lgan dasturiy ta'minot tizimlarini ishlab chiqishda qo'llaniladi. Rasmiy yondashuv himoyalanganlik yoki xavfsizlik ishini ishlab chiqarishni soddalashtiradi. Bu mijozlar yoki tartibga soluvchilarga tizim haqiqatan ham xavfsizlik yoki himoyalanganlik talablariga javob berishini ko'rsatadi. Biroq, rasmiy spetsifikatsiyani ishlab chiqishning yuqori xarajatlari tufayli, ushbu ishlab chiqish modeli juda muhim tizim injiniringidan tashqari boshqa hollarda kamdan-kam qo'llaniladi.

Incremental ishlab chiqish

Incremental ishlab chiqish - boshlang'ich versiyasini ishlab chiqish, foydalanuvchilar va boshqalarning fikr-mulohazalarini inobatga olish va kerakli tizim ishlab chiqilgungacha bo'lgan davrda dasturiy ta'minotni bir necha versiyalar orqali rivojlantirish g'oyasiga asoslanadi. Spetsifikatsiya, ishlab chiqish va tasdiqlash faoliyati alohida emas, balki bir-biriga bog'langan va tezkor teskari aloqqa ega bo'lgan faoliyatlardan iborat.



6.2-rasm

Amaliy tizimlar va dasturiy mahsulotlarni ishlab chiqishda eng keng tarqalgan yondashuvlardan biri bu bosqichma-bosqich rivojlanish (Incremental ishlab chiqish) hisoblanadi. Ushbu yondashuv rejaga asoslangan, tezkor yoki ushbu yondashuvlarning aralashmasi bo'lishi mumkin. Rejaga asoslangan yondashuvda tizim rivojlanish darajalari oldindan aniqlanadi; agar tezkor yondashuv qo'llanilsa, dastlabki rivojlanishlar aniqlanadi, ammo keyingi bosqichlarning rivojlanishi biznes-tizimlari va mijozlar ustuvorligiga bog'liq.

Dasturiy ta'minotning Incremental ishlab chiqish usullari moslashuvchan usullarining asosiylaridan bo'lib, u sharshara usulidan ko'ra ishlab chiqish jarayonida talablari o'zgaruvchan bo'lgan tizimlarni yaratishda qulay hisoblanadi. Aksariyat biznes tizimlari bu usul yordamida ishlab chiqiladi. Masalani hal qilish usuliga qarab Incremental ishlab chiqish amalga oshiriladi. Kamdan-kam hollarda

muammoning to'liq yechimini oldindan topiladi, xatoga yo'l qo'yilganini topilganida, orqaga qaytib, ketma-ket qadamlar bilan yechimga o'tiladi. Dasturiy ta'minotni bosqichma-bosqich ishlab chiqish orqali unga o'zgartirishlar kiritish arzonroq va osonroq bo'ladi.

Tizimning har bir qo'shimchasi yoki versiyasi mijozga kerak bo'lgan ba'zi funksiyalarni o'z ichiga oladi. Umuman olganda, tizimning dastlabki bosqichlarida eng muhim yoki eng tez talab qilinadigan funksiyalarni o'z ichiga oladi. Bu shuni anglatadiki, mijoz yoki foydalanuvchi tizimni ishlab chiqishning nisbatan ertabosqichida uning talab qilinadigan narsalarni yetkazib bera olishini baholashi mumkin. Agar yo'q bo'lsa, unda faqat rivojlanish strategiyasini o'zgartirishi kerak va keyingi modifikasiyalar uchun yangi funksiyalar aniqlanadi.

Incremental rivojlanish sharhara modeliga nisbatan uchta asosiy afzalliklarga ega:

1. Talablardagi o'zgarishlarni amalga oshirishga ketadigan xarajatlari kamayadi. Qayta ko'rib chiqilishi kerak bo'lgan tahlil va hujjatlar miqdori sharhara modelida talab qilinadiganidan ko'ra kamroq bo'ladi.

2. Ishlab chiqilgan ishlari bo'yicha mijozlarning fikr-mulohazalarini olish osonroq bo'ladi. Mijozlar dasturiy ta'minot haqida fikr bildirishlari va oldingidan ko'ra amalga oshirilgan ishlarni ko'rishlari mumkin. Mijozlarga dasturiy ta'minotni loyihalash hujjatlaridan uning rivojlanish bosqichlarini baholashi qiyinroq bo'ladi.

3. Foydali dasturiy ta'minotni mijozga erta yetkazib berish va joylashtirish, hattobarcha funksiyalar kiritilmagan bo'lsa ham mumkin. Mijozlar ertaroq dasturiy ta'minotdan foydalanishlari hamda foyda olishlari mumkin.

Boshqaruv nuqtai nazaridan, incremental yondashuv ikkita muammoga ega:

1. Jarayon ko'rinmaydi. Menejerlar rivojlanish darajasini o'lchash uchun muntazam natijalarni bilib turishlari lozim. Agar tizimlar tezda ishlab chiqilsa, tizimning har bir versiyasini aks ettiruvchi hujjatlarni ishlab chiqarish iqtisodiy jihatdan samarali emas.

2. Yangi qo'shimchalar qo'shilishi bilan tizim strukturasi yomonlashadi. Muntazam o'zgartirish tartibsiz kodga olib keladi, chunki har qanday usulda yangi funksiyalar qo'shiladi. Tizimga yangi xususiyatlarni qo'shish tobora qiyin va qimmatga tushadi. Tuzilmaning

buzilishini va kodning umumiy chalkashligini kamaytirish uchun moslashuvchan usullar dasturiy ta'minotni muntazam ravishda qayta ishlash (yaxshilash va qayta qurish) kerakligini taklif qiladi.

Incremental rivojlanish muammolari, ayniqsa, turli jamoalar tizimning turli qismlarini ishlab chiqadigan katta, murakkab, uzoq davom etadigan tizimlar uchun og'irlik qiladi. Katta tizimlar qat'iy tuzilma yoki arxitekturaga muhtoj va tizimning mustaqil qismlari bilan ishlaydigan turli guruhlarining mas'uliyati ushbu arxitekturaga nisbatan aniq belgilanishi kerak. Bu bosqichma-bosqich emas, balki oldindan rejalashtirilishi kerak.

Incremental ishlab chiqish tizim mijoziga har bir rivojlanish bosqichi yetkazib berish kerak degani emas. Tizimni bosqichma-bosqich ishlab chiqish va uni mijozlarga hamda boshqa manfaatdor tomonlarga o'z fikrlarini bildirishga taqdim etish mumkin, uni etkazib berish va mijozning muhitida joylashtirish shart emas. Qo'shimcha yetkazib berish dasturiy ta'minot real operatsion jarayonlarda qo'llanilishini anglatadi, shuning uchun foydalanuvchining fikr-mulohazalari haqiqatga yaqin bo'lishi kerak. Biroq, fikr-mulohazalarni taqdim etish har doim ham mumkin emas, chunki yangi dasturiy ta'minot bilan tajriba o'tkazish oddiy biznes jarayonlarini buzishi mumkin.

Integratsiya va konfiguratsiya

Ko'pgina dasturiy ta'minot loyihalarida dasturiy ta'minotdan qayta foydalanish mavjud. Bu ko'pincha loyihada ishlaydigan insonlar o'zlarini ishlariga o'xshash kodlarni bilishadi yoki ularni qidirib topishga xarakat qiladi, bu ish rasmiy ravishda bo'lmaydi. Kod qismlarini qidirishadi, kerak bo'lganda ularni o'zgartirishadi hamda o'zlari ishlab chiqqan yangi kod bilan birlashtirishadi.

Ushbu norasmiy qayta foydalanish turli ishlab chiqish jarayonlarida amalga oshiriladi. 2000 yildan boshlab, mavjud dasturiy ta'minotni qayta foydalanishga qaratilgan dasturiy ta'minotni ishlab chiqish jarayonlari keng qo'llanila boshlandi. Qayta foydalanishga yo'naltirilgan yondashuvlar qayta foydalanish mumkin bo'lgan dasturiy ta'minot komponentlari bazasiga va ushbu komponentlarning tarkibi integratsiyalashgan asosga tayanadi.

Uch turdagi dasturiy ta'minot komponentlari tez-tez qayta foydalaniladi:

1. Muayyan muhitda foydalanish uchun tuzilgan mustaqil dastur tizimlari.

Ushbu tizimlar ko‘plab xususiyatlarga ega bo‘lgan umumiy maqsadga ega bo‘lgan tizimlardir, ammo ular ma’lum bir dasturda foydalanish uchun moslashtirilishi kerak.

2. Java Spring (Wheeler and White 2013) kabi komponentlar fraymvorki bilan integratsiya qilinadigan komponent yoki paket sifatida ishlab chiqilgan obyektlar to‘plami.

3. Xizmat ko‘rsatish standartlariga muvofiq ishlab chiqilgan va Internet orqali masofadan turib murojat qilishga mo‘ljallangan veb-xizmatlar.

6.3-rasmda komponentlardan qayta foydalanishga yo‘naltirilgan integratsiya va konfiguratsiyaga asoslangan ishlab chiqishning umumiy jarayon modeliko‘rsatilgan.

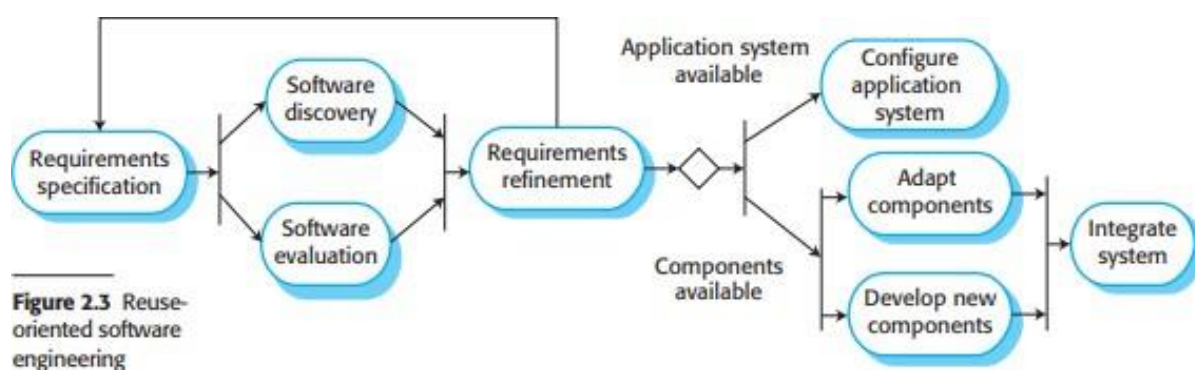


Figure 2.3 Reuse-oriented software engineering

6.3-rasm

Ushbu jarayonning bosqichlari:

1. Texnik talablar (Requirements specification). Tizim uchun dastlabki talablar taklif etiladi. Ular batafsil ishlab chiqilishi shart emas, lekin asosiy talablar va kerakli tizim xususiyatlarining qisqacha tavsiflarini o‘z ichiga olishi kerak.

2. Dasturiy ta’minotni aniqlash va baholash (Software discovery and evaluation). Dasturiy ta’minotga bo‘lgan talablarning konturini hisobga olgan holda, talab qilinadigan funkcionallikni ta’minlaydigan komponentlar va tizimlar uchun qidiruv amalga oshiriladi. Nomzod komponentlar va tizimlar asosiy talablarga javob beradimi yoki umuman tizimda foydalanish uchun mos yoki yo‘qligini aniqlash uchun baholanadi.

3. Talablarni takomillashtirish (Requirements refinement). Ushbu bosqichda talablar qayta foydalanish mumkin bo‘lgan komponentlar va ilovalar haqidagi ma’lumotlardan foydalangan holda aniqlanadi.

Talablar mavjud komponentlarni aks ettirish uchun o'zgartiriladi va tizim spetsifikatsiyasi qayta aniqlanadi. O'zgartirishlar mumkin bo'lmagan hollarda, muqobil echimlarni izlashda komponentlarni tahlil qilish faoliyati qayta kiritilishi mumkin.

4. Ilova tizimining konfiguratsiyasi (Application system configuration). Agar talablarga javob beradigan tayyor dastur tizimi mavjud bo'lsa, u yangi tizimni yaratishda foydalanish uchun sozlanishi mumkin.

5. Komponentlarning moslashuvchanligi va integratsiyasi (Component adaptation and integration). Agar tayyor tizim bo'lmasa, alohida qayta foydalanish mumkin bo'lgan komponentlar o'zgartirilishi va yangi komponentlar ishlab chiqilishi mumkin. Keyin ular tizimni yaratish uchun birlashtiriladi.

Konfiguratsiya va integratsiyaga asoslangan qayta foydalanishga yo'naltirilgan dasturiy injniring ishlab chiqilishi kerak bo'lgan dasturiy ta'minot qiymatini kamaytirish, shuningdek xarajatlar va xavflarni kamaytirish afzalligiga ega. Bu, odatda, dasturiy ta'minotni tezroq yetkazib berishga olib keladi. Biroq, talablarning murosaga kelishi muqarrar va bu foydalanuvchilarning haqiqiy ehtiyojlarini qondirmaydigan tizimga olib kelishi mumkin. Bundan tashqari, tizim evolyutsiyasi ustidan ba'zi nazorat yo'qoladi, chunki qayta foydalanish mumkin bo'lgan komponentlarning yangi versiyalari ulardan foydalanadigan tashkilotning nazorati ostida bo'lmaydi.

Jarayon faoliyati

Amaliy dasturiy ta'minot jarayonlari - bu dasturiy ta'minot tizimini aniqlash, loyihalash, amalga oshirish va sinovdan o'tkazishdan iborat bo'lgan umumiy maqsadli texnik, hamkorlik va boshqaruv faoliyatining ketma-ketligidan iborat. Umuman olganda, jarayonlar asboblardan tomonidan qo'llab-quvvatlanadi. Bu shuni anglatadiki, dasturiy ta'minot ishlab chiquvchilar ularga yordam berishi uchun talablarni boshqarish tizimlari, loyihalash modeli muharrirlari, dastur muharrirlari, avtomatlashtirilgan sinov vositalari va tuzatuvchilar kabi bir qator dasturiy vositalardan foydalanishlari mumkin.

Spetsifikatsiya, ishlab chiqish, testlash va evolyutsiyaning to'rtta asosiy jarayon faoliyati turli rivojlanish jarayonlarida turlicha tashkil etilgan. Sharshara modelida ular ketma-ket tashkil etilgan bo'lsa, bosqichma-bosqich rivojlanishda ular aralashtiriladi. Ushbu

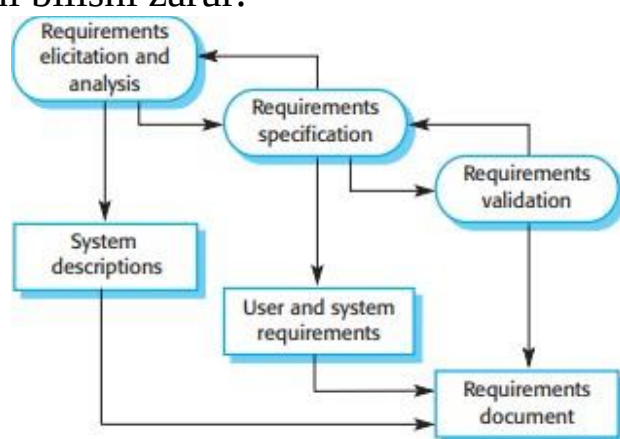
tadbirlarning qanday amalga oshirilishi ishlab chiqilayotgan dasturiy ta'minot turiga, ishlab chiquvchilarning tajribasi va malakasiga, dasturiy ta'minotni ishlab chiquvchi tashkilot turiga bog'liq.

Dasturiy ta'minot spetsifikatsiyasi

Dasturiy ta'minot spetsifikatsiyasi yoki talablar ishlab chiqarish - bu tizimdan qanday xizmatlar talab qilinishini tushunish va aniqlash hamda tizimning ishlashi va rivojlanishidagi cheklovlarni aniqlash jarayonidir. Talablarni ishlab chiqish dasturiy ta'minot jarayonining ayniqsa muhim bosqichidir, chunki bu bosqichda yo'l qo'yilgan xatolar muqarrar ravishda keyinchalik tizimni loyihalash va joriy etishda muammolarga olib keladi.

Talablarni ishlab chiqish bosqichi boshlanishidan oldin texnik-iqtisodiy yoki marketing tadqiqotlarini o'tkaziladi, kompaniya dasturiy ta'minotga ehtiyoji mavjud yoki yo'qligini tekshiradi va zaruriy dasturiy ta'minotni ishlab chiqishning texnik vamoliyaviy jihatdan real yoki real emasligini baholaydi. Texnik-iqtisodiy asoslar - bu qisqa muddatli, nisbatan arzon tadqiqotlar bo'lib, unda batafsilroq tahlilni davom ettirish yoki ettirmaslik to'g'risida qaror qabul qilinadi.

Talablarni ishlab chiqish jarayoni manfaatdor tomonlarning talablarini qoniqtiradigan hamda tizimda belgilangan tegishli talablar hujjatini ishlab chiqarishga qaratilgan. Talablar odatda detallashtirishning ikki darajasida taqdim etiladi. Yakuniy foydalanuvchilar va mijozlar talablarning yuqori darajadagi bayonotini ko'rishi kerak; tizim ishlab chiquvchilari batafsilroq tizim spetsifikatsiyasini bilishi zarur.



6.4-rasm

Talablarni ishlab chiqish jarayonida uchta asosiy faoliyat mavjud:

1. Talablarni aniqlash va tahlil qilish (Requirements elicitation and analysis.). Bu mavjud tizimlarni kuzatish, potentsial foydalanuvchilar va

xaridorlar bilan muhokamalar, vazifalarni tahlil qilish va boshqalar orqali tizim talablarini olish jarayoni. Bu bir yoki bir nechta tizim modellari va prototiplarini ishlab chiqishni o'z ichiga olishi mumkin. Bular aniqlanishi kerak bo'lgan tizimni tushunishga yordam beradi.

2. Talablar spetsifikatsiyasi (Requirements specification). Talablar spetsifikatsiyasi - bu talablarni tahlil qilish jarayonida to'plangan ma'lumotlarni talablar to'plamini belgilaydigan hujjatga o'tkazish faoliyati. Ushbu hujjatga ikki turdagi talablar kiritilishi mumkin. Foydalanuvchi talablari - bu tizimning mijozi va oxirgi foydalanuvchisi uchun tizim talablarining abstrakt bayonotlari; tizim talablari – taqdim etiladigan funktsionallikning batafsil tavsifi.

3. Talablarni tasdiqlash (Requirements validation). Ushbu faoliyat amaliylik, mustahkamlik va to'liqlik talablarini tekshiradi. Ushbu jarayon davomida talablar hujjatidagi xatolar albatta ravishda aniqlanadi. Keyin bu muammolarni tuzatish uchun uni o'zgartirish kerak.

Talablarni tahlil qilish bosqichi aniqlash va spetsifikatsiya qilish jarayonida davometadi va butun jarayon davomida yangi talablar paydo bo'ladi. Shuning uchun tahlil qilish, ta'riflash va spetsifikatsiya qilish faoliyati bir-biriga bog'langan.

Agile usullarda talablarni spetsifikatsiya qilish alohida faoliyat emas, balki tizimni ishlab chiqishning bir qismi sifatida qaraladi. Talablar tizimning har bir o'sishi uchun ushbu rivojlanish ishlab chiqilishidan oldin norasmiy ravishda belgilanadi. Talablar foydalanuvchi ustuvorliklariga muvofiq belgilanadi. Talablar ishlab chiqish guruhining bir qismi bo'lgan yoki ular bilan yaqindan ishlaydigan foydalanuvchilardan kelib chiqadi.

Dasturiy ta'minotni loyihalash va amalga oshirish

Dasturiy ta'minotni ishlab chiqishni amalga oshirish bosqichi - mijozga etkazib berish uchun bajariladigan tizimni ishlab chiqish jarayonidir. Ba'zan bu dasturiy ta'minotni loyihalash va dasturlashning alohida faoliyatini o'z ichiga oladi. Biroq, agar rivojlanishga tezkor yondashuv qo'llanilsa, loyihalash va amalga oshirish bir- biri bilan chambarchas bog'liq bo'lib, jarayon davomida rasmiy loyiha hujjatlari ishlab chiqilmaydi. Ammo ba'zi holatlarda dasturiy ta'minot hali ham ishlab chiqishda loyihalash norasmiy ravishda doskada va dasturchining daftarida qayd etilmoqda.

Dasturiy ta'minotni loyihalash - bu yaratiladigan dasturiy ta'minotning tuzilishi, tizim tomonidan ishlatiladigan ma'lumotlar modellari va tuzilmalari, tizim komponentlari orasidagi interfeyslar va ba'zan foydalaniladigan algoritmlarning tavsifi hisoblanadi. Loyihachilar tayyor loyihaga kelmaydilar, loyihani bosqichma- bosqich ishlab chiqadilar. Ular loyihani ishlab chiqishda tafsilotlarni qo'shadilar va oldingi loyihalarni o'zgartirish uchun doimiy orqaga qaytishadi.

6.4-rasmda loyihalash jarayonining abstrakt modeli bo'lib, loyihalash jarayoniga kirishlar, jarayon faoliyati va jarayon natijalarini ko'rsatadi. Loyihalash jarayonidagi xarakteristikalar ketma-ket keladi va bir-biriga bog'liqdir. Loyiha haqidagi yangi ma'lumotlar doimiy ravishda yaratiladi va bu avvalgi loyiha qarorlariga ta'sir qiladi. Shuning uchun albatta loyihani qayta ishlanadi.

Ko'pgina dasturlar boshqa dasturiy ta'minot tizimlari bilan o'zaro aloqa bo'ladi. Bu boshqa tizimlarga operatsion tizim, ma'lumotlar bazasi, vositachi dasturlar va boshqa amaliy tizimlar kiradi. Ular "dasturiy ta'minot platformasi" ni, ya'ni dasturiy ta'minot amalga oshiriladigan muhitni tashkil qiladi. Ushbu platforma haqidagi ma'lumotlar loyihalash jarayonining muhim qismidir, chunki loyihachilar uni muhit bilan qanday qilib yaxshi integratsiya qilish yo'lini topishlari kerak. Agar tizim mavjud ma'lumotlarni qayta ishlasa, u holda ushbu ma'lumotlarning tavsifi platforma spetsifikatsiyasiga kiritilishi mumkin. Aks holda, tizim ma'lumotlarini tashkil etishni aniqlash uchun ma'lumotlar tavsifi loyiha jarayoniga kirish bo'lishi kerak.

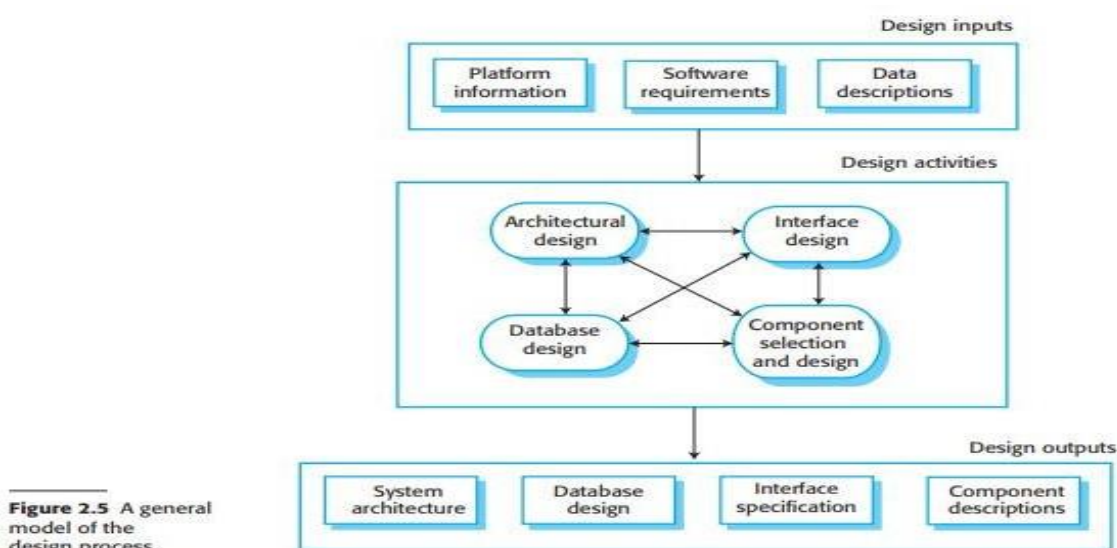


Figure 2.5 A general model of the design process

6.5-rasm

Loyihalash jarayonidagi faoliyatlar ishlab chiqilayotgan tizim turiga qarab farqlanadi. Misol uchun, real vaqt tizimlari vaqtni loyihalashning qo'shimcha bosqichini talab qiladi, lekin ma'lumotlar bazasini o'z ichiga olmaydi, shuning uchun ma'lumotlar bazasi talab etilmaydi. 2.5-rasmda axborot tizimlarini loyihalash jarayonining bir qismi bo'lishi mumkin bo'lgan to'rtta faoliyat ko'rsatilgan:

1. Arxitekturani loyihalash, bu erda tizimning umumiy tuzilishini, asosiy komponentlarini (ba'zan quyi tizimlar yoki modullar deb ataladi), ularning aloqalarini va ularning taqsimlanishi aniqlanadi.

2. Ma'lumotlar bazasi loyihalash, bu erda tizim ma'lumotlar tuzilmalarini loyihalashtiriladi va ular ma'lumotlar bazasida ifodalanishi ko'rsatib beriladi. Shunga qaramay, bu erda ish mavjud ma'lumotlar bazasidan qayta foydalanish yoki yangi ma'lumotlar bazasi yaratishga bog'liq.

3. Tizim komponentlari orasidagi interfeyslarni aniqlaydigan interfeysni loyihalash. Ushbu interfeys spetsifikatsiyasi bir ma'noli bo'lishi kerak. Aniq interfeysga ega bo'lgan holda, komponent boshqa komponentlar tomonidan qanday amalga oshirilishini bilmasdan foydalanishi mumkin. Interfeys spetsifikatsiyalari kelishilganidan so'ng, komponentlar alohida loyihalanishi va ishlab chiqilishi mumkin.

4. Komponentlarni tanlash va loyihalash, bu erda siz qayta foydalanish mumkin bo'lgan komponentlarni qidirasiz va agar mos komponentlar mavjud bo'lmasa, yangi dasturiy ta'minot komponentlarini loyihalanadi. Ushbu bosqichdagi loyiha amalga oshirish bosqichi uchun dasturchiga qoldirilgan oddiy komponent tavsifi bo'lishi mumkin. Shuningdek, bu qayta foydalanish mumkin bo'lgan komponentga yoki UMLda ifodalangan batafsil loyihalash modeliga kiritiladigan o'zgarishlar ro'yxati bo'lishi mumkin. Keyinchalik, loyihalash modeli avtomatik ravishda amalga oshirish bosqichida foydalanishi mumkin.

Nazorat savollari

1. Dasturiy ta'minot hayotiy sikli (SDLC) nima va uning asosiy bosqichlari qanday?

2. Talablarni yig'ish va tahlil qilish jarayoni qanday amalga oshiriladi?

3. Dizayn bosqichida nimalar amalga oshiriladi va uning asosiy maqsadi nima?

4. Dasturiy ta'minotni ishlab chiqish (kodlash) jarayoni qanday amalga oshiriladi?
5. Testlash jarayonida qanday usullar va texnikalar qo'llaniladi?
6. Dasturiy ta'minotni joriy etish va joylashtirish jarayoni qanday amalga oshiriladi?
7. Texnik xizmat va qo'llab-quvvatlash bosqichida nimalar amalga oshiriladi?
8. SDLC ning Agile va Waterfall modellarini solishtiring.
9. Dasturiy ta'minot hayotiy sikli davomida sifatni ta'minlash (QA) qanday amalga oshiriladi?
10. SDLC bosqichlarida qanday hujjatlar tayyorlanadi va ularning mazmuni qanday bo'lishi kerak?

7-MAVZU. VIZUAL MODELLASHTIRISH IDEF METODOLOGIYASI

Reja:

1. Vizual modellashtirishning umumiy tushunchasi
2. IDEF metodologiyasining turlari
3. IDEF metodologiyasining afzalliklari va kamchiliklari

Loyihani amalga oshirish vaqtini kamaytirish uchta tabiiy cheklovga ega: texnik imkoniyatlar, texnologik talablar va ish sifati. Ushbu funksiyalarni bajarishda ushbu va boshqa parametrlarni hisobga olish kerak. Vaqtni boshqarish loyihalarni boshqarishning asosiy elementlaridan biridir va quyidagilarni o‘z ichiga oladi:

- vaqt parametrlari uchun loyihani boshqarish konsepsiyasi;
- loyihani rejalashtirish;
- loyihani amalga oshirishni monitoring qilish;
- ishlarni bajarish jarayonini tahlil qilish va tartibga solish;
- loyiha boshqaruvini yopish.

Vaqt parametrlari odatda vaqt yoki ishlarning bir qismi amalga oshiriladigan davrlarni. shuningdek. loyihani amalga oshirish jarayonini nazorat qilish vaqtlarini tushunadi. Ko‘pincha loyihani vaqtida bajarish juda qiyin. Muammolarining barchasi biz boshqarayotgan narsalar haqida tushunchaning yetishmasligidan kelib chiqadi. Ko‘pchilik muammolar rejalashtirish bosqichida boshlanadi. Jadval bilan mos kelmasligi ob-havo sharoitlari, materiallarni yetkazib berishdagi kechikishlar va hokazolarga bog‘liq bo‘lishi mumkin, shuningdek, nuqsonni bartaraf etish uchun yetarli vaqtli zaxiralar mavjud emas. Agar loyiha doirasi va mavzusi dastlab noto‘g‘i aniqlangan bo‘lsa, ish vaqtida ish va jadvaldagi o‘zgarishlar talab etiladi.

Qayta takrorlanuvchi, odatdagi loyihalarda, vaqt va ishlarning ketma-ketligini aniq aniqlash uchun o‘tgan tajribalar natijasidan foydalanishingiz mumkin, lekin loyihalarning ko‘pi takrorlanmaydi.

Loyihada vaqtni o‘qotishining asosiy sabablari-rejalashtirishdagi xatolardir; sifatsiz sifatni boshqarish va byudjetlashtirish, tashqi muhitning noaniqligi sharoitida kutilmagan xarajatlar rejasining mavjud emasligi, loyiha ishtirokchilari orasida xavflarning noto‘g‘ri taqsimlanishi. strukturaviy bo‘lmagan aloqa tizimi va loyihaning haqiqiy bo‘lmagan hisobot tizimi.

Biznes jarayonlarini modellashtirish sohasida eng mashhur va keng qo'llaniladigan metodologiyalardan biri IDEF oilasining metodologiyalaridir. IDEF oilasi XX asrning 60-yillari oxirida paydo bo'lgan SADT (Structured Analysis and Design Technique – tuzilmaviy tahlil va dizayn texnikasi) deb nomlanadi, u hozirda quyidagi standartlarni o'z ichiga oladi.

1) IDEF0 – bu funktsional modellashtirish metodologiyasi. U tizimning tuzilishi va funksiyalarini, shuningdek, ushbu funksiyalarni bog'laydigan axborot va moddiy obyektlar oqimini aks ettiruvchi funktsional modelni yaratish uchun ishlatiladi.

2) IDEF1 – bu tizimlar ichidagi axborot oqimlarini modellashtirish metodologiyasi bo'lib, ularning tuzilishi va munosabatlarini ko'rsatish imkonini beradi. Metodologiya tizim funksiyalarini ta'minlash uchun zarur bo'lgan axborot oqimlarining tuzilishi va mazmunini aks ettiruvchi axborot modelini yaratish uchun ishlatiladi.

3) IDEF1X (IDEF1X Extended) – aloqador axborot tuzilmalarini qurish metodologiyasi. IDEF1X obyektlar o'rtasidagi munosabatlar metodologiyasi turiga kiradi va odatda ko'rib chiqilayotgan tizimga tegishli relyatsion ma'lumotlar bazalarini modellashtirish uchun ishlatiladi.

4) IDEF2 – tizimni rivojlantirishni dinamik modellashtirish metodologiyasi bo'lib, tizim funksiyalari, ma'lumotlar va resurslarning vaqt bo'yicha o'zgaruvchan xatti-harakatlarining dinamik modelini yaratishga imkon beradi. Hozirgi vaqtda IDEF0 statik diagrammalari to'plamini „Rangli Petri tarmoqlari“ (CPN - Color Petri Nets) asosida qurilgan dinamik modellarga aylantirish imkonini beruvchi algoritmlar va ularning kompyuter amaliyotlari ma'lum.

5) IDEF3 – bu tizimda sodir bo'ladigan jarayonlarni hujjatlashtirish metodologiyasi. IDEF3 har bir jarayon uchun stsenariy va operatsiyalar ketma-ketligini tavsiflaydi. IDEF3 diagrammasidagi funksiya IDEF3 vositalaridan foydalangan holda alohida jarayon sifatida taqdim etilishi mumkin.

6) IDEF4 – bu obyektga yo'naltirilgan tizimlarni qurish metodologiyasi. IDEF4 vositalari sizga obyektlarning tuzilishini va ularning o'zaro ta'sirining asosiy tamoyillarini vizual tarzda ko'rsatishga imkon beradi va shu bilan obyektga yo'naltirilgan murakkab tizimlarni tahlil qilish va optimallashtirish imkonini beradi.

7) IDEF5 – murakkab tizimlarni ontologik tadqiq qilish metodologiyasi. Ushbu metodologiyadan foydalanib, tizimning ontologiyasi atamalar va qoidalarining ma'lum bir lug'atidan foydalangan holda tavsiflanadi, buning asosida ko'rib chiqilayotgan tizimning muayyan vaqtdagi holati to'g'risida ishonchli bayonotlar shakllanishi mumkin. Ushbu bayonotlar asosida tizimni yanada rivojlantirish bo'yicha xulosalar chiqariladi va uni optimallashtirish amalga oshiriladi.

8) IDEF6 (Design Rational Capture) – bu loyihalashtirilgan modellarga bo'lgan ehtiyojni asoslash, sabab-oqibat munosabatlarini aniqlash va buni tizimning yakuniy hujjatlarida aks ettirish imkonini beruvchi axborot tizimlarini loyihalash jarayonini oqilona tasvirlash usuli.

9) IDEF8 (User Interface Modeling – Foydalanuvchi interfeysini modellashtirish) – Human-System Interaction Design Method – Inson va tizimning o'zaro ta'sirini loyihalash usuli, ya'ni foydalanuvchining turli xil tabiatdagi tizimlar bilan o'zaro ta'sirini loyihalash usuli (ma'lumotni hisoblash shart emas).

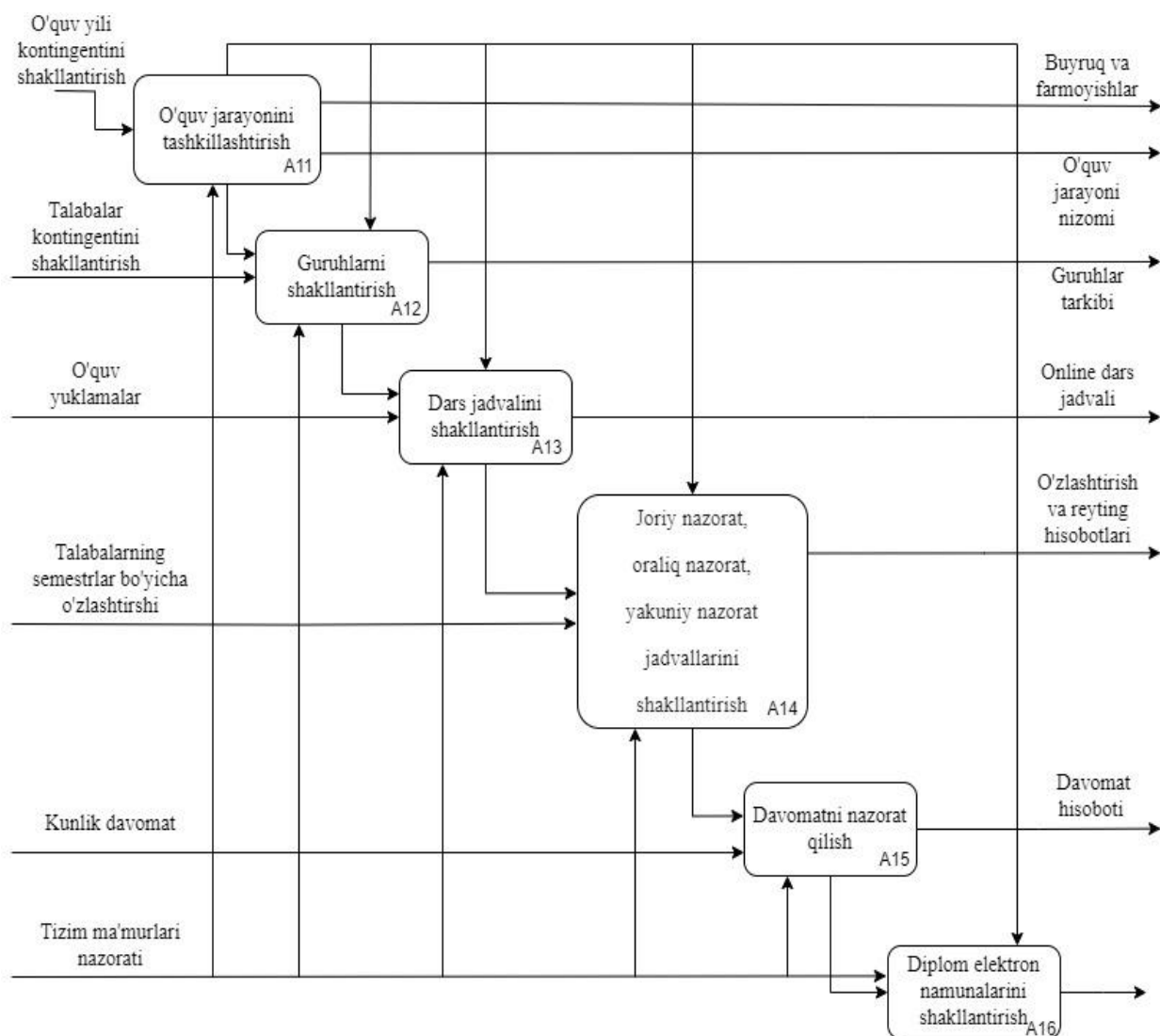
10) IDEF9 (Business Constraint Discovery Method) – bu „cheklovlar“ nuqtai nazaridan biznes tizimlarini o'rganish va tahlil qilish usuli. Cheklovlar natijani boshlaydi, obyektlar va agentlarning xatti-harakatlarini yo'naltiradi va cheklaydi (avtonom dasturiy modullar) tizimning maqsadlari yoki niyatlarini amalga oshirish uchun.

11) IDEF14 (Network Design Method) – bu kompyuter tarmog'ini loyihalash usuli bo'lib, u talablarni belgilash, tarmoq komponentlarini aniqlash, mavjud tarmoq konfiguratsiyalarini tahlil qilish va kerakli tarmoq xususiyatlarini shakllantirish imkonini beradi.

Amaliy qism

1. Ixtiyoriy dasturiy ta'minot loyihasida kechuvchi biznes jarayonlarning IDEF0 grafik notatsiyasi va dekompozitsiya modelini yarating.

Misol: O'quv jarayonlarini shakllantirish axborot tizimi uchun



Nazorat savollari

1. IDEF metodologiyasi nima va uning asosiy maqsadi nima?
2. IDEF metodologiyasining qaysi turlari mavjud va ularning har biri nima bilan farqlanadi?
3. IDEF0 metodologiyasining asosiy tamoyillari va komponentlari qanday?
4. IDEF1 metodologiyasi nimani ifodalaydi va uning asosiy qoidalar qanday?
5. IDEF1X metodologiyasi qanday maqsadga xizmat qiladi va uning diagrammasi qanday tuziladi?
6. IDEF3 metodologiyasi qanday ishlaydi va u qaysi jarayonlarni modellashtirishda qo'llaniladi?
7. IDEF0 diagrammasini yaratish bosqichlari qanday amalga oshiriladi?

8. IDEF metodologiyasining kuchli va zaif tomonlari qanday?
9. IDEF metodologiyasining turli sanoat sohalarida qo'llanilishi bo'yicha misollar keltiring.
10. IDEF metodologiyasining boshqa modellashtirish yondashuvlari bilan solishtirganda qanday farqlari mavjud?

8-MAVZU. EPS METODOLOGIYASI

Reja:

1. Grafik modellashtirish standartlarining paydo bo'lish tarixi
2. EPC belgilarining xususiyatlari
3. EPC diagrammasini qurish algoritmi
4. EPC notatsiyasining afzalliklari va kamchiliklari

"Ma'lumot qanchalik aniq uzatilsa, u murojaat qilingan odam tomonidan tezroq va aniqroq idrok qilinadi." Bu haqiqat uzoq vaqtdan beri marketologlar, o'qituvchilar va tadqiqotchilar tomonidan faol foydalanilgan. Rahbarlar ham chetda qolmadi. Ushbu maqolada EPC yozuvi eng mashhur zamonaviy usullardan biri sifatida muhokama qilinadi, bu sizga murakkab ish tavsifini nafaqat qulay, balki loyiha yoki jarayonning barcha ishtirokchilari uchun ancha aniqroq va tushunarli qilish imkonini beradi.

Grafik modellashtirish standartlarining paydo bo'lish tarixi

Hozir, jamiyatdagi barcha jarayonlarning rivojlanish sur'ati oshib, tizimlar murakkablashib borayotgan bir paytda, odamlarga ta'sir qilish san'ati sifatida boshqaruv ham muhandislik tizimlarini boshqarish kabi tizimlarni boshqarish qobiliyatiga ega bo'lishga majbur. O'tgan asrning 90-yillari boshlarida "reinjiniring" so'zi biznes leksikoniga kirdi. Va undan keyin biznesning "muhandisligi" tushunchasi paydo bo'ldi. Agar birinchisi biznes jarayonlarini qayta loyihalash bo'lsa, ikkinchisi noldan boshlab samarali tashkiliy tizimni loyihalashdir.

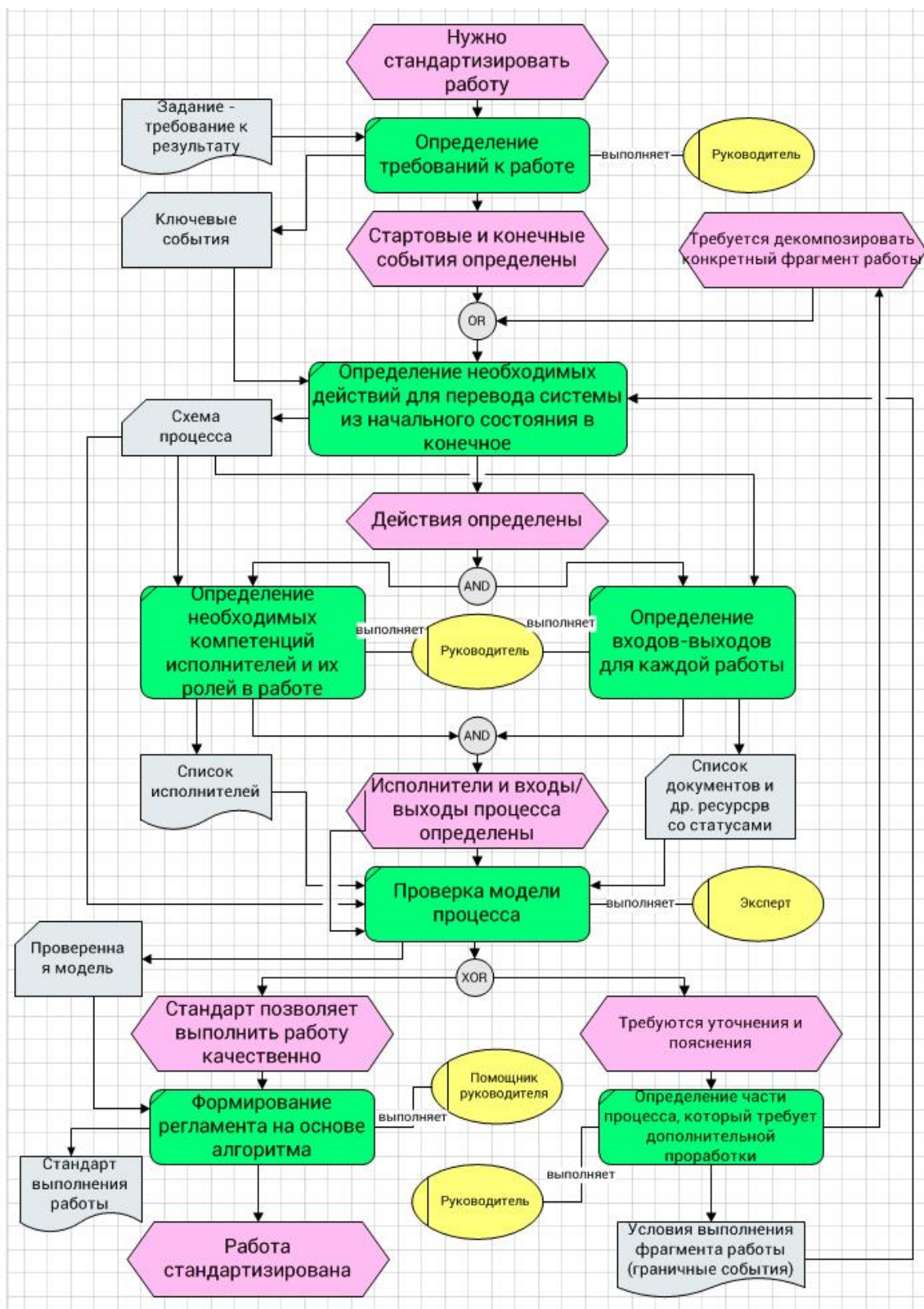
Ushbu tendentsiya shuni ko'rsatadiki, tashkilotni tizim sifatida tavsiflash va hatto qurish usullarini izlash uzoq vaqtdan beri va juda muvaffaqiyatli davom etmoqda. Agar biz menejmentni nafaqat san'at, balki fan sifatida ham ko'rib chiqsak, u boshqa har qanday fan kabi formulalar va qonunlarni belgilash uchun o'ziga xos belgilar tizimiga muhtoj. Bunday tizimli echimlar ko'plab faoliyat sohalarida muhandislar, kimyogarlar, fiziklar, matematiklar va boshqalar tomonidan muvaffaqiyatli qo'llaniladi.

Tashkilot bo'lgan ijtimoiy-iqtisodiy tizim tabiiy fanlarga qaraganda ancha xilma-xil bo'lib, "aksiomalar" va boshqaruv formulalarini qayd etishning yagona shakli hali topilmagan. Ammo yo'l asfaltlangan. Va u bizga ma'lum bir natijaga erishish tartibini

tavsiflash imkonini beruvchi taniqli oqim sxemalaridan boshlanadi. Ammo, afsuski, oqim sxemasining vizual imkoniyatlari juda cheklangan va u biznesni boshqarish jarayonining barcha xilma-xil elementlarini ko'rsatishga imkon bermaydi.

Bugungi kunda ijodiy faoliyat jarayonida odamlar va tizimlarning o'zaro ta'sirini tasvirlash mumkin bo'lgan sxemalar uchun juda ko'p variantlar ishlab chiqilgan. Ular korxona faoliyatining barcha jabhalarini to'liqroq tavsiflash uchun asboblarni majmuasiga birlashtirilgan. Bunday vositalar modellashtirish metodologiyalari deb ataladi. Ulardan eng to'liq va taniqli uchasi:

- ARIS (Integratsiyalashgan axborot tizimlari arxitekturasini);
- SADT (Tuzilgan tahlil va dizayn texnikasi);
- UML (Unified Modeling Language).



8.1-rasm

EPC belgisi yordamida ishni standartlashtirish jarayoniga misol (kattalashtirish uchun bosing)

Korxona faoliyatini to'liq va har tomonlama tavsiflash uchun metodologiyaga kiritilgan va notatsiyalar deb ataladigan turli xil grafik standartlardan foydalanish kerak. Ammo tor muammolarni hal qilish uchun tegishli tavsif uchun ixtiro qilingan belgini tanlash kifoya. Yuqorida ushbu maqolada muhokama qilinadigan yozuvning grafik turlaridan biri.

EPC belgilarining xususiyatlari

EPC (Event-driven Process Chain) modellashtirish yozuvi ma'lum bir ishni bajarish jarayonida o'zaro ta'sir qilish algoritmlarini yaratishga qaratilgan. Uning asosiy elementlari quyidagilardir:

- ishni boshlaydigan yoki tugatadigan voqealar;
- tizimni bir holatdan ikkinchi holatga o'tkazuvchi harakatlar (ish);
- ishni bajaruvchilar;
- resurslar va ish natijalari (kirish va chiqishlar).

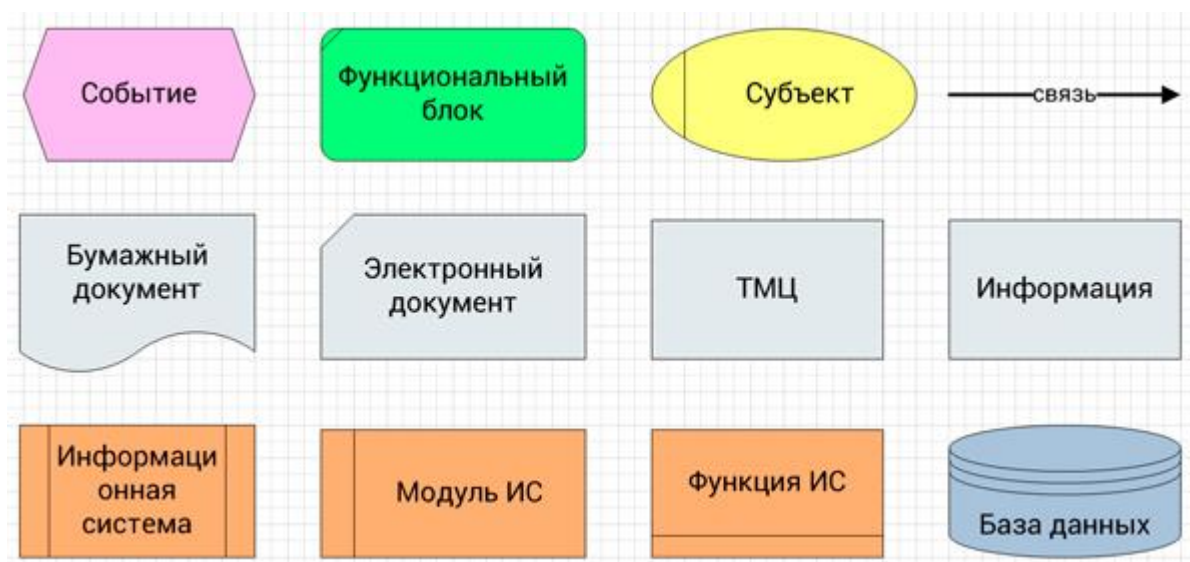
Ushbu belgi 1990-yillarning boshida ishlab chiqilgan Vilgelm-August Scheer muallifligidagi ARIS metodologiyasining ajralmas qismi hisoblanadi. Oldingi bo'limning oxirida rasmda EPC belgisi yordamida ishlarni standartlashtirish jarayonining umumiy ko'rinishi ko'rsatilgan. Keling, ushbu belgidan foydalanib, tashkilotning biznes jarayonlarini tavsiflash xususiyatlarini ko'rib chiqaylik. Diagrammaning mohiyatini o'rganmasdan ham, qizil va yashil elementlarning almashinishi darhol e'tiborni tortadi - bu yozuv nomiga xos bo'lgan voqealar va jarayonlar zanjiri. Model elementlarining tarkibi to'rtta asosiy pozitsiya bilan belgilanadi.

1. **Pushti shakllar** hodisalardir. Hodisa - bu tizim bir yoki bir nechta biznes-jarayonlarning keyingi rivojlanishini qabul qiladigan va belgilaydigan holat. Hodisalar funksiyalarni faollashtirishi yoki funksiyalar tomonidan yaratilishi mumkin.

2. **Yashil elementlar** funksiyalar yoki funktsional bloklardir. Funktsional blok - bu berilgan natijani olish va tizimni bir holatdan ikkinchi holatga o'tkazish uchun bajariladigan harakat yoki kichik jarayon. Funksiyalar tartibi diagrammada yuqoridan pastgacha ko'rsatilgan.

3. Har bir ishning ijrochilari haqidagi ma'lumotlar ham yaxshi tasvirlangan, ular yordamida ko'rsatiladi **sariq ovallar**. Ushbu element "Mavzu" deb nomlanadi, u ijrochilar, egalar yoki ishtirokchilarni bildiradi. Ijrochi lavozimlari, bo'linmalari va rollari sub'ektlar sifatida belgilanishi mumkin.

4. Nomlangan raqamlardan tashqari, diagrammada **kul rang to'rtburchaklar** yoki ularga o'xshash shakllar qo'llaniladi, ular ishlatilgan va yaratilgan resurslarning xilma-xilligini ko'rsatadi va jarayonda natijalar beradi. Agar biz pul o'tkazmasini ko'rsatishni istasak, kulrang to'rtburchakdan ham foydalanishimiz kerak bo'ladi.



8.2-rasm

EPC yozuvining asosiy elementlari

Jarayon modelini tavsiflash jarayonida biz axborot tizimidan foydalanamiz. Keyin biz uni maxsus uch darajali elementlar to'plamidan foydalanib ko'rsatishimiz mumkin.

1. IS - axborot tizimi.
2. IC moduli.
3. IS funktsiyasi.

Ma'lumotlar bazalari an'anaviy ravishda o'z tasviriga ega - silindr shaklida. Garchi ulardan axborot tizimisiz va ma'lumotlar bazasisiz tizimdan foydalanish bugun menga imkonsizdek tuyuladi. Funktsiyalar orasidagi o'tishlar mantiqini ko'rsatish uchun mantiqiy operatorlar qo'llaniladi, ular parallel ishlarni bajarish yoki hodisalarning yuzaga kelishi shartlarini belgilashga yordam beradi. Ular ikkala funktsiya va hodisalarni birlashtirish yoki tarmoqlash variantlarini ko'rsatadi. Faqat uchta mantiqiy operator mavjud: "AND", "OR" va "Exclusive OR". Turli xil tizimlar turli xil grafik belgilardan foydalanishi mumkin.

Оператор AND ("И") Λ AND	Используется если: • завершение выполнения функции должно инициировать одновременно несколько событий; • событие происходит только после обязательного завершения выполнения нескольких функций; • функция может начать выполняться только после того, как произойдут несколько событий; • одно событие инициирует одновременное выполнение нескольких функций.
Оператор OR ("ИЛИ") V OR	Используется если: • завершение выполнения функции может инициировать одно или несколько событий; • событие происходит после завершения выполнения одной или нескольких функций; • функция может начать выполняться после того, как произойдет одно или несколько событий.
Оператор XOR ("Исключающее ИЛИ") XOR	Используется если: • завершение выполнения функции может инициировать только одно из событий в зависимости от условия; • событие происходит сразу после завершения выполнения либо одной, либо другой функции; • функция может начать выполняться сразу после того, как произойдет либо одно событие, либо другое.

8.3-rasm

EPC yozuvida mantiqiy operatorlarni belgilash va ulardan foydalanish

EPC diagrammasini tuzish algoritmi

Ko‘rib turganimizdek, ushbu belgining shubhasiz afzalligi - bu diagrammalarni qurish uchun intuitiv elementlar va qoidalar to‘plami. Jarayon diagrammalarini yaratish uchun jarayonni modellashtirish uchun maxsus dasturlardan foydalanish tavsiya etiladi. EPC uchun bu birinchi navbatda ARIS tizimi. Ammo bu qimmat va juda murakkab va bo‘limlar faoliyatini tartibga solish uchun kichik davriy loyihalar uchun foydalanilmaydi.

Belgilanishning soddaligi va ommabopligi biznes jarayonlarini chizish uchun boshqa vositalarni, shu jumladan EPC notatsiyasini yaratishni rag'batlantirdi. Ulardan eng oddiyi Visio - undagi shablonlardan biri "EPC diagrammasi" deb ataladi. Men uchun eng foydali vosita bu Business Studio tizimidir. Unda jarayonni chizish qobiliyatiga qo'shimcha ravishda siz avtomatik ravishda hujjat (Jarayon reglamenti) va uning ishtirokchilari uchun ish ko'rsatmalarini yaratishingiz mumkin, bu esa ishlash standartlarini ishlab chiqish jarayonining muntazam qismini sezilarli darajada soddalashtiradi.

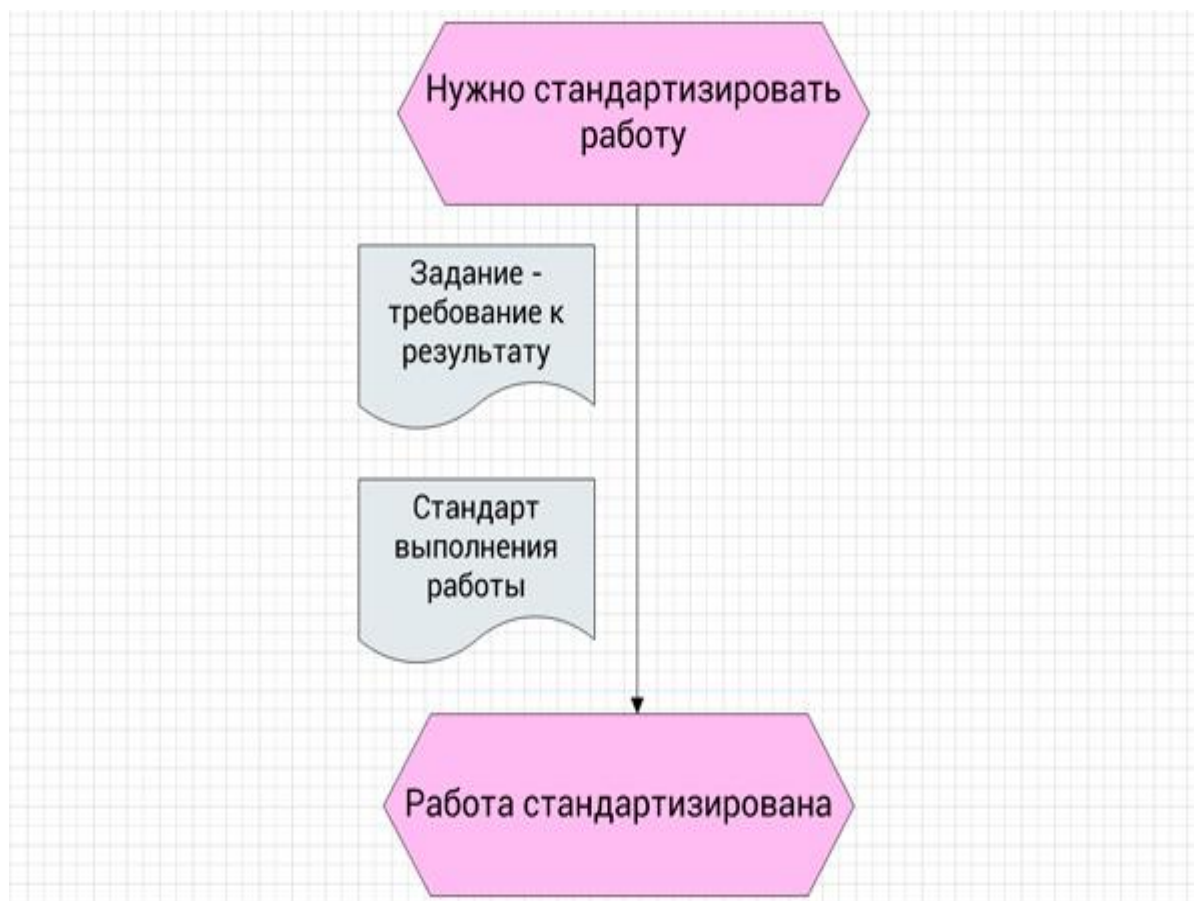
Turli dasturlarda ikkilamchi elementlarning ranglari va belgilari biroz farq qilishi mumkin, ammo umumiy qoidalar har doim kuzatiladi. Maqolaning birinchi qismida keltirilgan EPC notatsiyasining misoli ushbu belgi bilan ishlashning soddalashtirilgan algoritmini aks ettiradi. Keling, buni bosqichma-bosqich ko'rib chiqaylik.

1. Birinchi qadam. Avval bizda nima bor va nimani xohlayotganimizni aniqlaymiz - chegara hodisalari.

2. Ikkinchi bosqichda biz chegara hodisalarini harakatlar va tegishli oraliq hodisalar bilan "suyultiramiz".

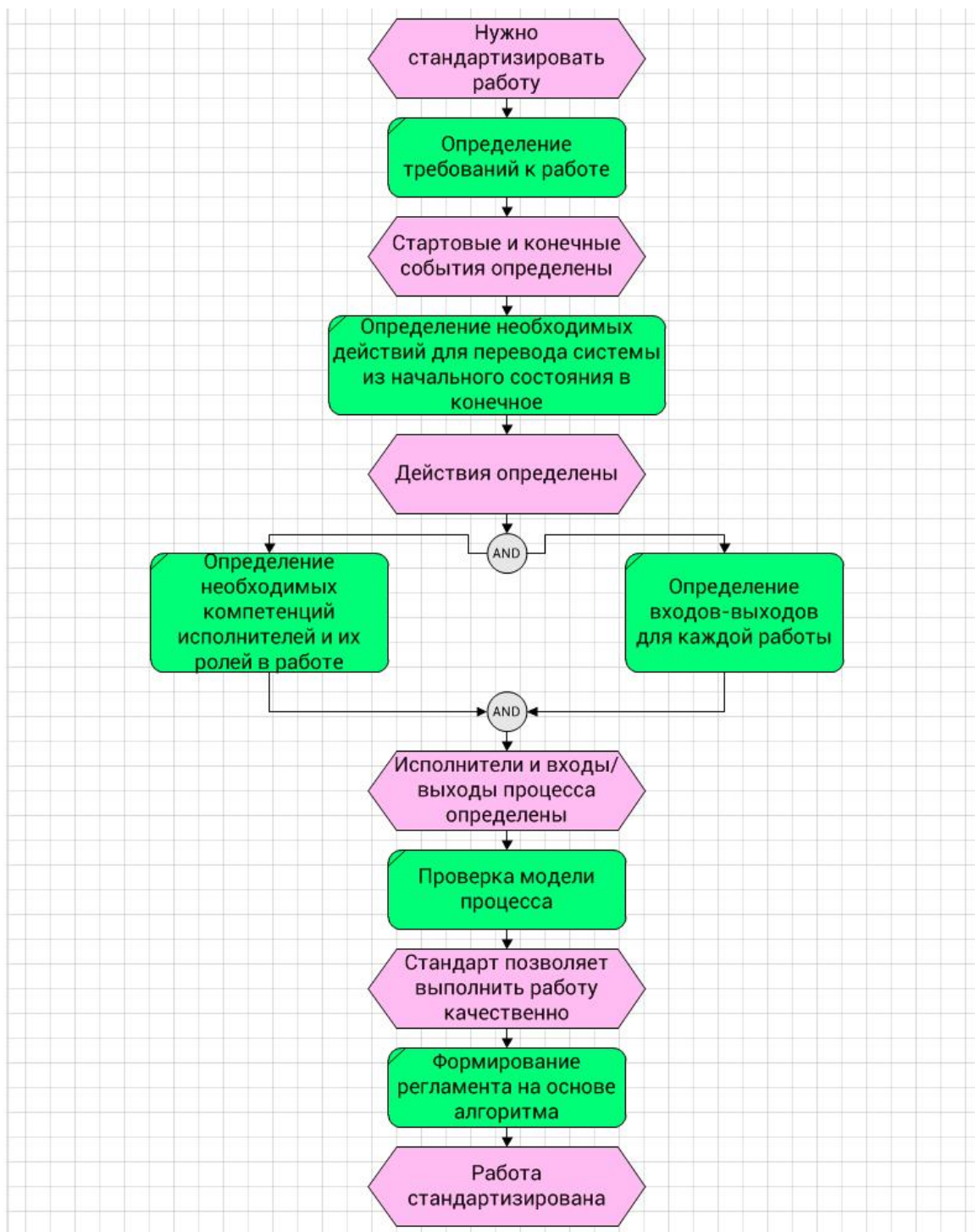
3. Uchinchidan, biz har bir bosqichni (kirishlarni) bajarish uchun zarur bo'lgan hujjatlar va (yoki) ma'lumotlarni va har bir bosqichda (chiqishlar) ish natijalari bo'lgan hujjatlarni ilova qilamiz. Bundan tashqari, asarga rollarni belgilash bilan ijrochilar bilan aloqalar qo'shiladi. Eng keng tarqalgan rol "bajaradi", ammo boshqalar ham bo'lishi mumkin: "natijani tasdiqlash", "texnik qism uchun mas'ul", "nostandart bajarilganligi haqida xabar berish kerak" va boshqalar.

4.4-5 bosqichlarda biz diagrammaning to'liqligi va sifatini baholaymiz, diagrammada jarayonni bajarishning barcha variantlari hisobga olinganligini tahlil qilamiz. Agar diagramma kerakli harakatlar to'plamini to'liq aks ettirsa, biz jarayonni bajarish uchun talablar bilan hujjat tuzamiz. Agar dekodlashni talab qiladigan mutlaqo aniq funktsiyalar bo'lmasa, biz ulardan kichik jarayonlarni yaratamiz va butun jarayonni takrorlaymiz, lekin bu alohida ish bilan bog'liq. Ta'riflangan bosqichlarni vizual aks ettirish misollari quyida keltirilgan.



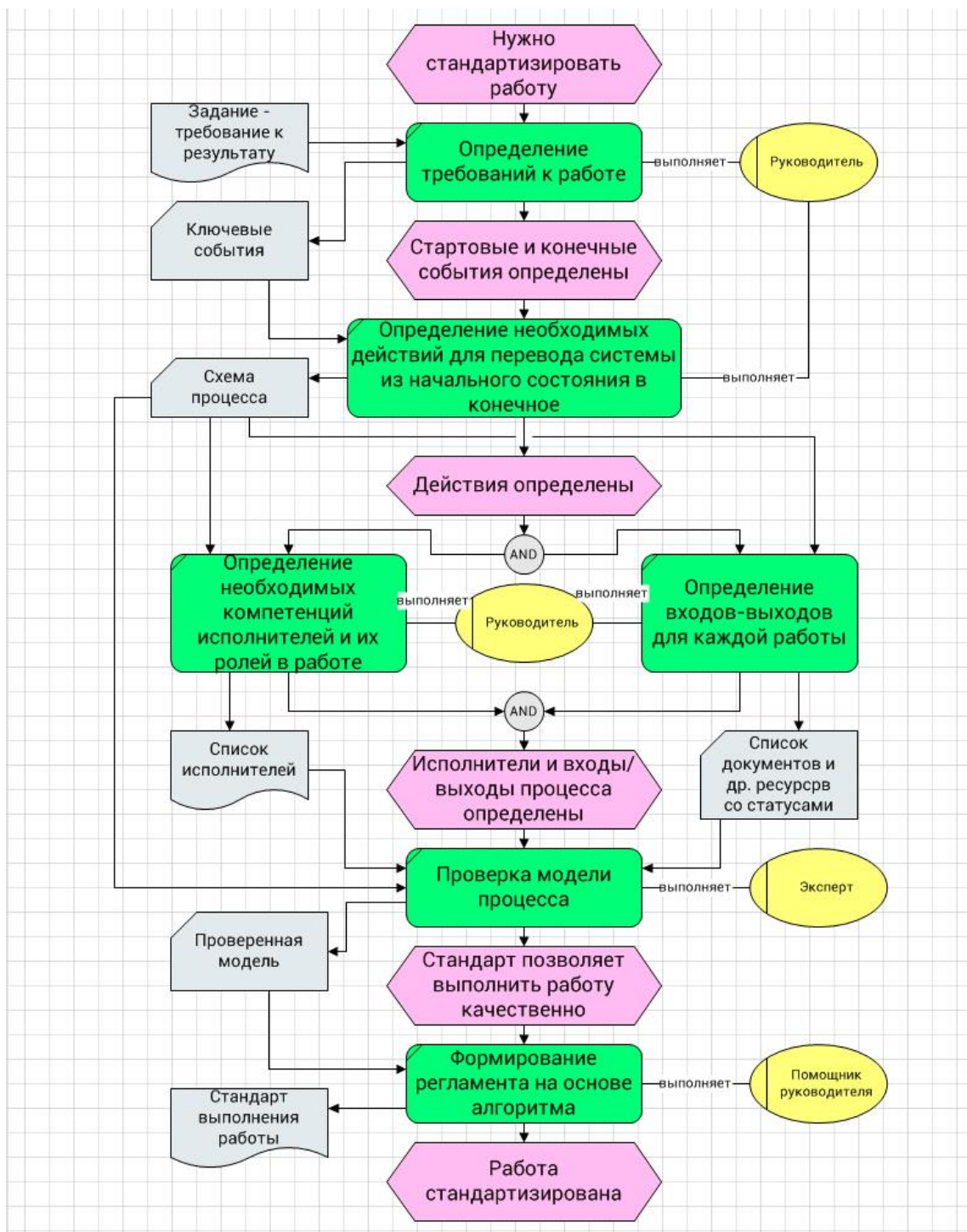
8.5-rasm

EPC notatsiyasidan foydalangan holda ishni standartlashtirish jarayonini yaratishning birinchi bosqichiga misol (kattalashtirish uchun bosing)



8.6-rasm

Jarayon ustida ishlashning ikkinchi bosqichiga misol (kattalashtirish uchun bosing)



8.7-rasm

EPC notatsiyasining uchinchi bosqichi uchun jarayon diagrammasi (kattalashtirish uchun bosing)

Barcha bosqichlarni tugatgandan so‘ng, biz jarayonni amalga oshirishning batafsil rejasini olamiz, uni ijrochilar tushunadi. Va aniq belgilangan voqealar va natijalar jarayonning barcha muhim bosqichlari uchun nazorat nuqtalarini o‘rnatishga imkon beradi. Va

yana sizni maqolaning boshida e'lon qilingan vizual model misoliga havola qilaman.

EPC notatsiyasining afzalliklari va kamchiliklari

Oddiylilik va qulaylikdan tashqari, EPC dan foydalanish quyidagi afzalliklarga ega.

1. Barcha muhim tashkiliy elementlarni bitta diagrammada ko'rsatishga imkon beradi (oddiy oqim sxemasidan farqli o'laroq).

2. U modelning turli darajalarida ishlatilishi mumkin - ikkala global jarayonlarni tavsiflash va har bir funktsional blok kichik jarayonga aylanishi mumkinligi sababli batafsil ko'rsatmalar berish uchun.

3. Jarayonni murakkab parallellashtirishni amalga oshirish oson, chunki siz bir qatorga istalgan miqdordagi hodisalarni kiritishingiz mumkin.

Shu bilan birga, ushbu belgi quyidagi kamchiliklar tufayli yagona va yagona bo'lib qolmadi.

1. Har bir kichik harakat uchun voqealarni o'ylab topish zarurati sxemani juda murakkablashtiradi.

2. Tashkiliy buzilishlar, ehtimol, topshiriqlarni kuzatishning noqulayligi bilan bog'liq.

3. Kirish va chiqishlarni yuqori sifatli ro'yxatga olish kontaktlarning zanglashiga olib keladigan to'rtburchaklar va strelkalar bilan ortiqcha yuklanishiga olib keladi va shu bilan kontaktlarning zanglashiga olib kelishini yanada murakkablashtiradi.

4. Ishni parallellashtirishda ijrochilarni aks ettirish juda qiyin. Agar bir kishi bir guruh funktsiyalarni bajarsa, rasm o'qlar bilan murakkablashadi. Agar bir nechta ijrochilar bo'lsa yoki biz uzun o'qlarni chizishni istamasak, biz "ovals" ni ijrochilar bilan takrorlashimiz kerak. Bularning barchasi juda tez orada diagrammada to'liq chalkashlikka olib kelishi mumkin.

O'z tajribamdan shuni aytishim mumkinki, ushbu belgida chizilgan mahalliy operatsion protseduralar ham ishlab chiquvchi, ham ko'rsatmalar foydalanuvchisi uchun juda qulaydir. Belgilar loyiha menejerlari uchun ham mos keladi, chunki bu ularga loyihadagi ishlarni taqsimlashni turli loyiha ishtirokchilari uchun intuitiv tilda vizual ravishda rejalashtirish imkonini beradi. Korxona faoliyatining yanada murakkab ko'p darajali modelini ishlab chiqish uchun boshqa

modellashtirish belgilari ko‘proq mos keladi, biz ularni keyingi maqolalarda ko‘rib chiqamiz.

Nazorat savollari

1. EPS metodologiyasi nima va uning asosiy maqsadi nima?
2. EPS diagrammasining asosiy komponentlari qanday?
3. EPS metodologiyasining asosiy tamoyillari nimalardan iborat?
4. Voqealarga asoslangan jarayon zanjiri (EPS) diagrammasini yaratish bosqichlari qanday?
5. EPS diagrammasida bog‘lovchilar qanday ishlatiladi va ular qanday turlarga bo‘linadi?
6. EPS diagrammasida resurslar va tashkilot elementlari qanday ifodalanadi?
7. EPS metodologiyasining kuchli va zaif tomonlari qanday?
8. EPS metodologiyasi qanday biznes jarayonlarni modellashtirishda qo‘llaniladi?
9. EPS diagrammasining boshqa modellashtirish yondashuvlari (masalan, BPMN yoki UML) bilan solishtirganda qanday farqlari bor?
10. EPS metodologiyasini qo‘llashda qanday amaliy qiyinchiliklar va muammolar yuzaga kelishi mumkin?

9-MAVZU. BPMN 2.0 SPESIFIKATSIYASI

Reja:

1. BPMN 2.0 ning Asosiy Tushunchalari
2. Biznes jarayonlar va notatsiyalar
3. BPMN 2.0 Diagrammalarini Yaratish
4. BPMN 2.0 Standartlari va Talablari

So'nggi bir necha yil ichida biznes jarayonlarini boshqarish tizimlari uchun XML ijro tillarini ishlab chiqish bo'yicha faol ish olib borildi. WSBPEL kabi tillar biznes jarayonlarini aniqlashning rasmiy mexanizmini ta'minlaydi. Bunday tillarning asosiy elementi shundaki, ular biznes jarayonlarini boshqarish tizimlarining ishlashi va o'zaro ta'siri uchun optimallashtirilgan. Bunday tillarni dasturiy ta'minot operatsiyalari uchun optimallashtirish ularni biznes jarayonlarini bevosita loyihalash, boshqarish va monitoring qilish uchun kamroq moslashtiradi. WSBPEL ham blok, ham grafik tuzilishga ega va pi-hisob (1) kabi rasmiy matematik modellar tamoyillaridan foydalanadi.

Ushbu texnik magistral ichki va B2B o'zaro ta'sirining murakkabligini hal qila oladigan biznes jarayonlari uchun asos bo'lib xizmat qiladi, shu bilan birga veb-xizmatlarning afzalliklaridan foydalanadi. WSBPEL ning tabiatini hisobga olgan holda, murakkab biznes jarayoni dasturiy ta'minot tizimi (yoki dasturchi) tomonidan juda yaxshi boshqariladigan potentsial murakkab, ajratilmagan va intuitiv bo'lmagan formatda tashkil etilishi mumkin. lekin biznes tahlilchisi uchun tushunish qiyin va menejerlarga jarayonni ishlab chiqish, boshqarish va monitoring qilish vazifasi yuklangan. Shunday qilib, bunday veb-xizmatga asoslangan XML ijro tillari tomonidan ko'rib chiqilmaydigan "muvofiqlik" yoki "ko'chirish" ning "inson darajasi" mavjud.

Biznes bilan shug'ullanadigan odamlar biznes jarayonlarini vizualizatsiya qilish uchun ko'pincha oqim sxemalariga murojaat qilishadi. Minglab biznes-tahlilchilar korxonalar faoliyatini o'rganadilar va oddiy sxemalar yordamida biznes jarayonlarini aniqlaydilar. Bu erda biznes-jarayonlarni dastlabki loyihalash formati va ushbu biznes-jarayonlarni amalga oshiradigan WSBPEL kabi tillar formati o'rtasida texnik bo'shliq paydo bo'ladi. Ushbu bo'shliqni rasmiy mexanizm yordamida yopish kerak, shunda biznes-jarayonning

vizualizatsiyasi (notlash) bunday biznes-jarayonlarni bajarish uchun tegishli formatga (biznes-jarayonlarni bajarish tili) mos keladi.

Biznes jarayonlarining dasturiy ta'minot darajasida emas, balki inson darajasida o'zaro ishlashini vizual biznes jarayonlarini modellashtirish (BPMN) standartlashtirish orqali hal qilish mumkin. Vizual biznes jarayonlarini modellashtirish biznes jarayonlarini boshqaruvchi odamlar tomonidan qo'llanilishi uchun mo'ljallangan bir nechta diagrammalardan iborat. Shunday qilib, vizual biznes-jarayonlarni modellashtirish biznes jarayonlar tizimlarini bajarish tili (WSBPEL) uchun xaritalashni ta'minlaydi. Shu sababli, vizual biznes-jarayonlarni modellashtirish, bajarish uchun optimallashtirilgan biznes-jarayon tilida aniqlangan biznes jarayonlarini vizualizatsiya qilishning standart mexanizmini taqdim etadi.

Vizual biznes jarayonlarini modellashtirish grafik shaklda tasvirlangan ichki biznes jarayonlarini tushunish qobiliyatini ta'minlaydi va tashkilotlarga ushbu protseduralarni standartlashtirilgan tarzda boshqarishga imkon beradi. Hozirgi vaqtda jarayonlarni modellashtirishning ko'plab vositalari va metodologiyalari mavjud. Odamlar bir kompaniyadan boshqasiga o'tishini va kompaniyalarning birlashishi va ajralib chiqishini hisobga olsak, biznes-tahlilchilar biznes-jarayonlarning turlicha qarashlarini tushunishlari kerak: bir xil jarayonning uning hayotiy siklidagi turli ko'rinishlari (ishlab chiqish, amalga oshirish, bajarish, monitoring va tahlil qilish). Shu sababli, standart grafik belgilar tashkilotlar ichida va o'rtasidagi hamkorlik va tijorat operatsiyalarini tushunishni osonlashtirish uchun mo'ljallangan. Bu biznes va uning ishtirokchilarini yaxshiroq tushunishni ta'minlaydi va tashkilotlarga yangi ichki yoki B2B biznes sharoitlariga tezda moslashishga yordam beradi.

O'qilishi va moslashuvchanligi uchun vizual biznes jarayonini modellashtirish an'anaviy oqim diagrammasidan foydalanadi. Bundan tashqari, biznes jarayonlarini vizual modellashtirishni amalga oshirish semantikasi to'liq rasmiylashtirilgan. Object Management jamoasi o'qilishi, moslashuvchanligi va kengaytirilishini o'zida mujassamlashtirgan yangi avlod yozuvini yaratish uchun vizual biznes-jarayonlarni modellashtirish yaratilishidan oldin orttirilgan biznes-jarayon yozuvlari tajribasidan foydalanadi.

Shunday qilib, vizual biznes-jarayonlarni modellashtirish B2B biznes-jarayonlari tushunchalarini, masalan, davlat va xususiy

jarayonlar va xoreografiya, shuningdek, istisnolarni ko‘rib chiqish, tranzaktsiyalar va kompensatsiya kabi ilg‘or modellashtirish tushunchalarini boshqarish orqali an'anaviy biznes-jarayon belgilarining imkoniyatlarini kengaytirishga yordam beradi.

BPMN doirasi

Ushbu xalqaro standart turli xil vositalar o‘rtasida BPMN jarayoni ta'riflarini (ham domen modeli va diagramma diagrammasi) almashish uchun ishlatilishi mumkin bo‘lgan belgi, biznes-jarayon modeli va almashish formatini taqdim etadi. Ushbu xalqaro standartning maqsadi foydalanuvchilar bir xizmat ko‘rsatuvchi provayder muhitida yaratilgan jarayon ta'riflaridan boshqa provayder muhitida foydalanishlari uchun jarayon ta'riflarining ko‘chmaligini ta'minlashdan iborat. BPMN 2.0.1 xalqaro standarti BPMN 1.2 ko‘lami va imkoniyatlarini bir necha sohalarda kengaytiradi:

- Barcha BPMN elementlarining bajarilishi semantikasini rasmiylashtiradi.
- Jarayon modeli kengaytmalari va grafik kengaytmalarni kengaytirish mexanizmini belgilaydi.
- Hodisalarning tarkibi va o‘zaro bog‘liqligini batafsil bayon qiladi.
- Odamlarning o‘zaro ta'sirining ta'rifini kengaytiradi.
- Xoreografiya modelini belgilaydi.

Ushbu xalqaro standart BPMN 1.2 ning ba'zi nomuvofiqliklari va noaniqliklarini ham ko‘rib chiqadi. BPMN faqat biznes jarayonlariga qo‘llanilishi mumkin bo‘lgan modellashtirish tushunchalarini qo‘llab-quvvatlash bilan cheklangan. Bu shuni anglatadiki, tashkilotlar biznes yuritish uchun foydalanadigan boshqa modellashtirish turlari BPMN doirasidan tashqarida. Quyidagi jihatlar ushbu xalqaro standart doirasidan tashqaridadir:

- Tashkiliy modellar va resurslarni aniqlash.
- Funktsional shovqinlarni modellashtirish.
- Ma'lumotlar modellari va axborot modellari.
- Strategiyani modellashtirish.
- Biznes qoidalari modellari.

Yuqori darajadagi modellashtirishning bunday turlari biznes jarayonlariga bevosita yoki bilvosita ta'sir qilganligi sababli, BPMN va boshqa yuqori darajadagi biznes modellashtirish o‘rtasidagi

munosabatlar BPMN va boshqa spetsifikatsiyalar aniqlanganligi sababli rasmiy ravishda aniqlanishi mumkin.

BPMN ma'lumotlar oqimini (xabarlarni) va artefaktlarning faoliyat bilan bog'lanishini ko'rsatsa ham, bu ma'lumotlar oqimi tili emas. Bundan tashqari, ish sharoitlarini modellashtirish, biznes-jarayonlarni kuzatish va joylashtirish ushbu xalqaro standartning amal qilishiga kirmaydi.

BPMN 2.0.1 WS-BPEL 2.0 kabi platformaga bog'liq bo'lgan ba'zi modellashtirish tillariga moslashtirilishi mumkin. Ushbu xalqaro standart BPMN kichik to'plamini WS-BPEL 2.0 bilan taqqoslashni o'z ichiga oladi. Boshqa yangi standartlar bilan taqqoslash alohida harakatlar hisoblanadi. Ushbu xalqaro standart ma'lumotlar turlarini, ifodalarini va xizmat ko'rsatishni aniqlash uchun boshqa standartlardan foydalanadi. Bunday standartlar mos ravishda XML Schema, XPath va WSDL hisoblanadi.

BPMN dan foydalanish

Biznes jarayonlarini modellashtirish turli xil ma'lumotlarni odamlarning keng doirasiga etkazish uchun ishlatiladi. BPMN ko'plab modellashtirish turlarini qamrab olish uchun mo'ljallangan va sizga biznes jarayonlarini oxirigacha yaratish imkonini beradi. BPMN qurilish bloklari BPMN diagrammasining bo'limlarini ajratishni osonlashtiradi. BPMN modelida uchta asosiy turdagi submodellar mavjud:

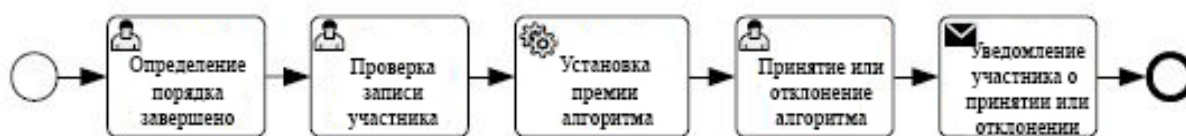
1. Jarayonlar (nazorat mexanizmlari)
 - Xususiy bajarilmaydigan (ichki) biznes jarayonlari
 - Xususiy bajariladigan (ichki) biznes jarayonlari
 - Ochiq jarayonlar
2. Xoreografiya
3. Jarayonlar va/yoki xoreografiyalarni o'z ichiga olishi mumkin bo'lgan hamkorlik
 - Suhbatlarni ko'rish

Xususiy (ichki) biznes jarayonlari

Xususiy biznes jarayonlari bitta tashkilotning ichki jarayonlaridir. Bunday jarayonlar odatda ish oqimi yoki BPM jarayonlari deb ataladi (10.4-rasmga qarang). Veb-xizmatlar sohasida keng qo'llaniladigan yana bir nom - bu servis orkestratsiyasi. Shaxsiy jarayonlarning ikki turi mavjud: bajariladigan va bajarilmaydigan. Bajariladigan jarayonlar 14-

bandda belgilangan semantikaga muvofiq bajarish maqsadida modellashtirilgan jarayonlardir. Albatta, jarayonning evolyutsiyasi jarayonida detallarning etishmasligi tufayli jarayon bajariladigan bo‘lib qolmaydigan bosqichlar bo‘ladi. Amalga oshirilmaydigan jarayon - bu jarayonning harakatini modellashtirish vositasi tomonidan aniqlangan tafsilotlar darajasida hujjatlashtirish maqsadida modellashtirilgan jarayon. Shunday qilib, rasmiy shartlarni ifodalash kabi bajarish uchun zarur bo‘lgan ma'lumotlar odatda bajarilmaydigan jarayonga kiritilmaydi.

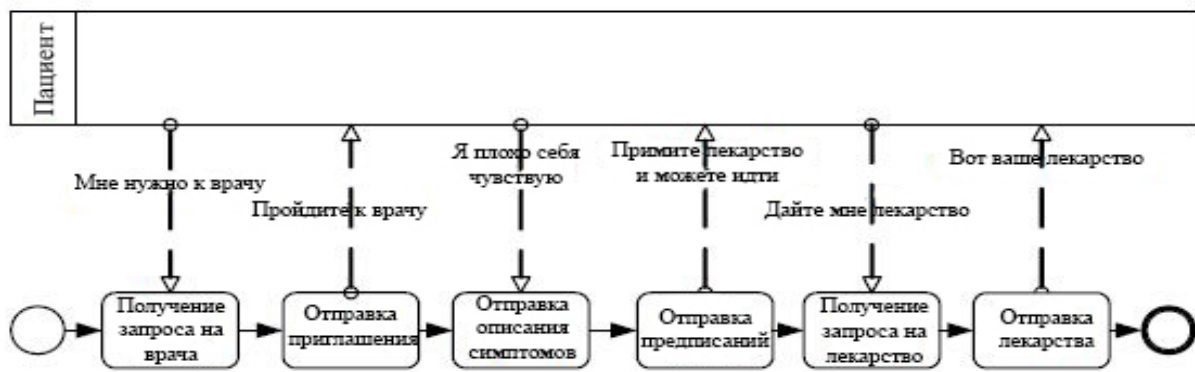
Agar trek belgisi ishlatilsa (masalan, hamkorlik, quyida ko‘ring), u holda shaxsiy biznes jarayoni bitta hovuzda joylashgan bo‘ladi. Shuning uchun jarayonlar ketma-ketligi hovuzda mavjud va uning chegaralarini kesib o‘ta olmaydi. Xabar oqimi individual xususiy biznes jarayonlari o‘rtasida mavjud bo‘lgan o‘zaro ta'sirlarni ifodalash uchun hovuz chegaralarini kesib o‘tishi mumkin.



9.1-rasm - Xususiy biznes jarayoniga misol

Ochiq jarayonlar

Ochiq jarayon - bu xususiy biznes jarayoni va boshqa jarayon yoki ishtirokchi o‘rtasidagi o‘zaro ta'sir (9.1-rasmga qarang). Ochiq jarayonga faqat boshqa ishtirokchilar bilan muloqot qilish uchun foydalaniladigan harakatlar kiritiladi. Boshqa "ichki" xususiy biznes jarayonlari ommaviy jarayonlarda ko‘rinmaydi. Ochiq jarayon xabarlar oqimini va jarayon bilan o‘zaro aloqada bo‘lish uchun zarur bo‘lgan xabarlar oqimining tartibini ko‘rsatadi. Ochiq jarayonlar alohida yoki hamkorlikning bir qismi sifatida ochiq jarayon faoliyati va boshqa ishtirokchilar o‘rtasidagi xabarlar oqimini ifodalash uchun modellashtirilishi mumkin. E'tibor bering, ommaviy jarayon BPMN 1.2 da "mavhum" deb ataladi.



9.2-rasm - Ochiq jarayonga misol

Hamkorlik

Hamkorlik ikki yoki undan ortiq xo‘jalik yurituvchi sub'ektlar o‘rtasidagi hamkorlikni ifodalaydi. Hamkorlik odatda hamkorlarni ifodalovchi ikki yoki undan ortiq hovuzlarni o‘z ichiga oladi. Ishtirokchilar o‘rtasida xabar almashinuvi ikkita hovuzni (yoki hovuz ichidagi obyektlarni) bog‘laydigan xabarlar oqimi bilan ifodalanadi. Xabar to‘plamlari bilan bog‘liq xabarlar ham ko‘rsatilishi mumkin. Hamkorlik bir-biri bilan o‘zaro ta'sir qiluvchi ikki yoki undan ortiq ochiq jarayonlar sifatida ifodalanishi mumkin (9.3-rasmga qarang). Ochiq jarayonda hamkorlik ishtirokchilarining harakatlari ishtirokchilar o‘rtasidagi "aloqa nuqtalari" sifatida qaralishi mumkin. Tegishli ichki (bajariladigan) jarayonlar ochiq jarayonlarga qaraganda ko‘proq harakatlar va tafsilotlarni o‘z ichiga oladi. Yoki hovuz bo‘sh bo‘lishi mumkin ("qora quti"). Xoreografiyalari Hovuzlar o‘rtasida "o‘rtada" paydo bo‘lishi mumkin, chunki ular havzalar o‘rtasida xabarlar to‘plamini almashadilar. Hamkorlikda barcha turdagi hovuzlar, jarayonlar va xoreografiyalarga ruxsat beriladi.



9.3-rasm - Hamkorlik jarayoniga misol

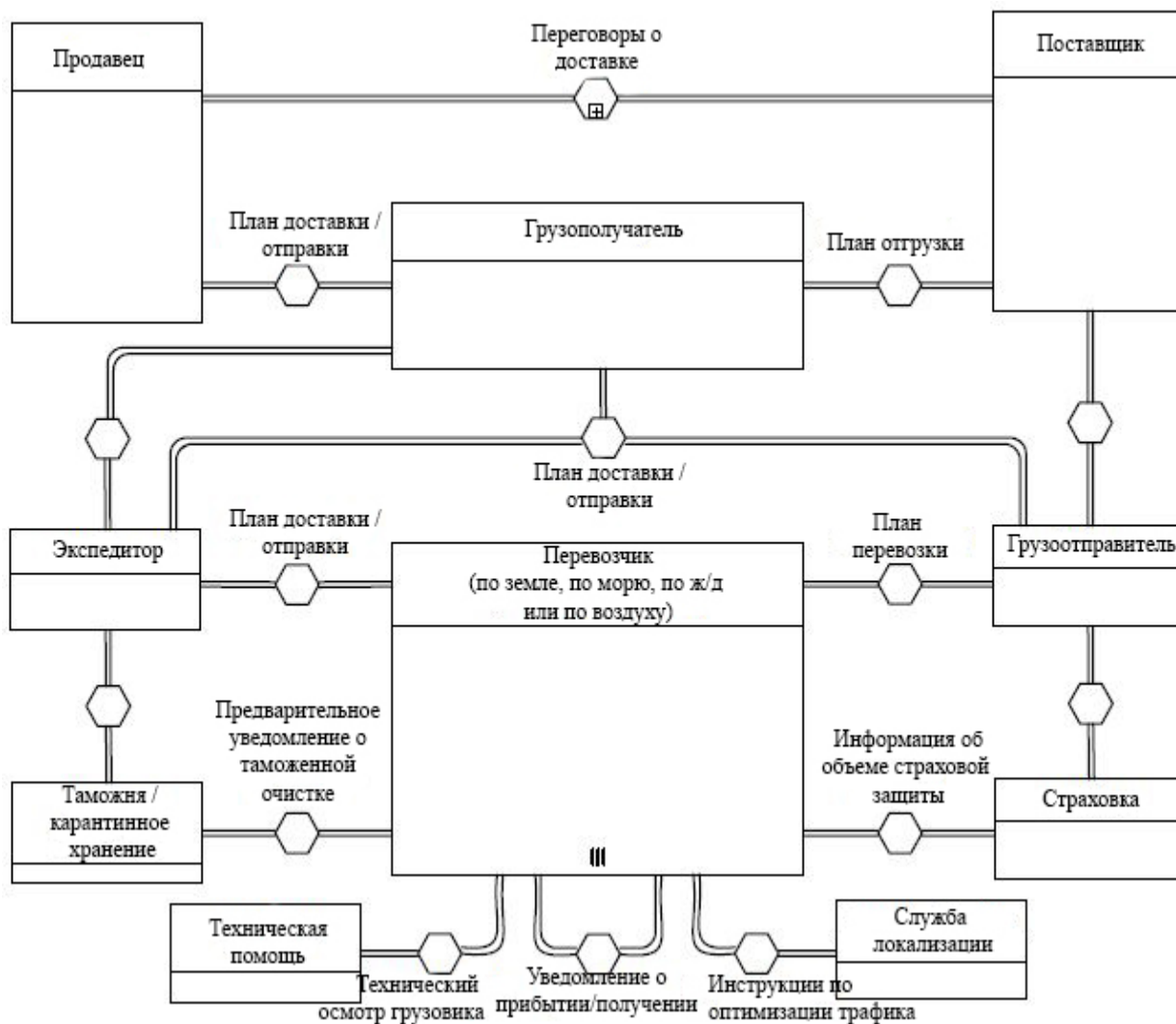
Xoreografiya

Mustaqil xoreografiya (pul yoki orkestr yo‘q) - kutilgan xatti-harakatni, o‘zaro ta’sir qiluvchi ishtirokchilar o‘rtasidagi protsessual shartnomani belgilash. Hovuz ichida normal jarayon mavjud va hovuzlar (ishtirokchilar) o‘rtasida xoreografiya paydo bo‘ladi. Xoreografiya xususiy biznes jarayoniga o‘xshaydi, chunki u faoliyat, hodisalar va shlyuzlar tarmog‘idan iborat (9.4-rasmga qarang). Shu bilan birga, xoreografiya ikki yoki undan ortiq ishtirokchilar o‘rtasida (1 yoki undan ortiq) xabar almashinuvi to‘plami bo‘lgan o‘zaro ta’sirlar bo‘lishi bilan farq qiladi. Bundan tashqari, oddiy jarayondan farqli o‘laroq, xoreografiyada jarayonni kuzatuvchi markaziy boshqaruvchi, nuqtali shaxs yoki kuzatuvchi yo‘q.



9.4-rasm - Xoreografiyaga misol

Suhbat diagrammasi muayyan foydalanishni ko‘rsatadi va hamkorlik diagrammasining mazmunli tavsifini beradi. Biroq, suhbat hovuzlari odatda jarayonni o‘z ichiga olmaydi va xoreografiya odatda suhbat diagrammasi hovuzlari o‘rtasida mos kelmaydi. Suhbat - bu mantiqiy xabar almashish munosabati. Amalda, mantiqiy munosabat ko‘pincha Buyurtma, Yetkazib berish va Yetkazib berish yoki Hisob-faktura kabi manfaatdor biznes obyekt(lar)ni nazarda tutadi. Xabar almashinuvi bir-biri bilan bog‘liq va turli biznes stsenariylarini aks ettiradi. Masalan, logistikada to‘ldirish quyidagi stsenariy turlarini o‘z ichiga oladi: savdo buyurtmasini yaratish, bir nechta savdo buyurtmalarini birlashtirgan holda yuklarga tashuvchilarni tayinlash, bojxona/karantin saqlashini kesib o‘tish, to‘lovlarni qayta ishlash va istisnolarni o‘rganish. Shunday qilib, suhbat diagrammasi (9.5-rasmga qarang) ishtirokchilar (hovuzlar) o‘rtasidagi suhbatlarni (olti burchakli shaklda) ko‘rsatadi. Ushbu diagramma sizga domen bilan bog‘liq bo‘lgan turli suhbatlarga qushning nazari bilan qarashga yordam beradi.



9.5-rasm - Suhbat diagrammasining namunasi

Diagrammadagi nuqtai nazar

BPMN diagrammasi turli ishtirokchilarning jarayonlarini ifodalashi mumkinligi sababli, diagramma har bir ishtirokchi uchun har xil bo'lishi mumkin. Ya'ni, ishtirokchilar jarayonning ularga nisbatan qanday qo'llanilishi haqida turli nuqtai nazarga ega. Ba'zi harakatlar muayyan ishtirokchilar uchun ichki bo'ladi (ya'ni, ishtirokchi tomonidan bajariladi yoki boshqariladi), ba'zi harakatlar esa tashqi bo'ladi. har bir ishtirokchi o'ziga ichki va tashqi bo'lgan narsaga nisbatan o'z nuqtai nazariga ega bo'ladi. Ishtirokchi faoliyat holatini ko'rishi yoki muammolarni bartaraf etishi uchun tashqi va ichki qismlarni ajratish muhimdir. Biroq, diagrammaning o'zi bir xil bo'lib qoladi. 7.3-rasmda biznes jarayonlarining ikki nuqtai nazari ko'rsatilgan. Bir nuqtai nazar bemorga, ikkinchisi shifokorga tegishli. Diagrammada jarayonning ikkala ishtirokchisining harakatlari

ko'rsatilgan, ammo jarayon tugagandan so'ng, har bir ishtirokchi faqat o'z harakatlarini nazorat qiladi. Diagrammaning nuqtai nazari jarayonning muayyan ishtirokchiga nisbatan xatti-harakatlarini tushunish uchun muhim bo'lsa-da, BPMN nuqtai nazarni ta'kidlash uchun grafik mexanizmlardan foydalanmaydi. Dizayner yoki modellashtirish vositasi diagrammaning ushbu xususiyatini ta'kidlash uchun vizual belgilardan foydalanishi mumkin.

Diagramma harakatining talqini

Ushbu xalqaro standartda biz jarayonda ketma-ketlik oqimlaridan qanday foydalanishni tasvirlaymiz. Bunday tavsifni osonlashtirish uchun biz ketma-ketlik oqimlari bo'ylab harakatlanadigan va jarayon elementlaridan o'tuvchi token tushunchasidan foydalanamiz. Token - bu ishlaydigan jarayonning xatti-harakatlarini aniqlash uchun ishlatiladigan nazariy tushuncha. Jarayon elementlarining xatti-harakati jarayon davomida harakatlanayotganda ularning token bilan o'zaro ta'sirini tavsiflash orqali aniqlanadi. Biroq, BPMN uchun zarur bo'lgan modellashtirish va bajarish vositalari tokenning har qanday shaklini amalga oshirish uchun TALAB EMAS. Boshlanish hodisasi yakuniy hodisa bilan tugashi KERAK token hosil qiladi (agar yakuniy hodisa grafikda ko'rsatilmagan bo'lsa, yashirin bo'lishi mumkin). Tokenning harakati ketma-ketlik oqimlari, shlyuzlar va jarayon harakatlari tarmog'i bo'ylab osongina kuzatilishi kerak. QAYD: Token xabarlar oqimini kesib o'tmaydi, chunki xabarning o'zi (nomidan ko'rinib turibdiki) xabarlar oqimini kesib o'tadi.

BPMN elementlari

Shuni ta'kidlash kerakki, BPMNni yaratish sabablaridan biri biznes-jarayon modellarini yaratishning sodda va tushunarli mexanizmini yaratish zarurati va shu bilan birga, biznes jarayonlariga xos bo'lgan murakkablikni engish zarurati edi. Ikki xil qarama-qarshi muammolarni hal qilish uchun yozuvning grafik elementlarini toifalarga tizimlashtirish yondashuvidan foydalanildi. Natijada BPMN diagrammalari bilan ishlaydigan odamlarga asosiy element turlarini osongina tanib olish va diagrammalarni tushunish imkonini beruvchi notatsiya toifalarining kichik ro'yxati paydo bo'ldi. Elementlarning asosiy toifalari ichida qo'shimcha ichki o'zgarishlar, shuningdek, tushunish qulayligi uchun diagrammaning umumiy tuzilishiga sezilarli

o'zgarishlar kiritmasdan murakkablik talablariga javob beradigan qo'shimcha ma'lumotlar qo'shilishi mumkin.

Elementlarning beshta asosiy toifasi mavjud:

1. Oqim elementlari;
2. Ma'lumotlar;
3. Birlashtiruvchi elementlar;
4. Mas'uliyat sohalari;
5. Artefaktlar.

Oqim elementlari biznes jarayonining harakatini belgilaydigan eng muhim grafik elementlardir. Oqim elementlari ham quyidagilarga bo'linadi:

1. Voqealar;
2. Harakatlar;
3. Shlyuzlar.

Grafik ma'lumotlarini quyidagi to'rtta element yordamida ko'rsatish mumkin:

1. Ma'lumotlar obyekti
2. Ma'lumotlarni kiritish
3. Imprint
4. Ma'lumotlarni saqlash

Oqim elementlarini bir-biriga yoki boshqa ma'lumotlarga ulashning to'rtta usuli (to'rtta ulash elementi) mavjud:

1. Operatsiyalar oqimi;
2. Xabarlar oqimi;
3. Uyushma;
4. Ma'lumotlar assotsiatsiyasi.

Mas'uliyat sohalaridan foydalangan holda asosiy modellash elementlarini guruhlashning ikki yo'li mavjud:

1. Hovuz yordamida guruhlash;
2. Trek yordamida guruhlash.

Artefaktlar jarayon haqida qo'shimcha ma'lumot qo'shish uchun ishlatiladi. Ikkita tipik artefakt mavjud, ammo biznes-jarayon modelarlari yoki modellashtirish dasturlari kerakli miqdordagi artefaktlarni qo'shishi mumkin. Qo'shimcha BPMN imkoniyatlaridan umumiy foydalanish yoki vertikal bozorlar uchun kattaroq artefaktlar to'plamini standartlashtirish uchun foydalanish mumkin. Hozirgi vaqtda artefaktlar ro'yxati quyidagi elementlarni o'z ichiga oladi:

1. Guruh

2. Matn izohi

BPMN modellashtirishning asosiy elementlari

Voqea elementi - bu jarayon yoki uning xoreografiyasi davomida sodir bo'ladigan narsa. Voqea biznes jarayonining borishiga ta'sir qiladi va ko'pincha sabab (tetik) yoki ta'sir (natija) bo'ladi. Ichki markerlar yordamida turli triggerlarni yoki ularning natijalarini farqlash uchun mo'ljallangan bo'sh markazli doira shaklida tasvirlangan. Hodisalarning biznes jarayoniga ta'siriga ko'ra, hodisalarning uch turi mavjud: boshlang'ich voqea, oraliq voqea va yakuniy voqea.



Harakat

Harakat - bu jarayon davomida kompaniya tomonidan bajariladigan ish uchun umumiy atama. Harakatlar elementar yoki elementar bo'lmagan (kompozitsiya) bo'lishi mumkin. Jarayon modelining bir qismi bo'lgan quyidagi harakatlar turlari ajratiladi: subprocess va Vazifa. Vazifa ham, kichik jarayon ham burchaklari yumaloq to'rtburchaklar shaklida tasvirlangan. Harakatlar standart jarayonlarda ham, xoreografiyalarda ham qo'llaniladi.



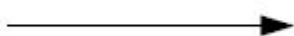
Gateway

Shlyuzlar jarayonda va xoreografiyada ketma-ketlik oqimining divergensiyasi va yaqinlashuvini boshqarish uchun ishlatiladi. Shunday qilib, atama shoxlanish, bifurkatsiya, birlashma va bog'lovchi yo'llarni nazarda tutadi. Ichki belgilar biznes jarayonining xatti-harakatlarini nazorat qilish turini ko'rsatadi.



Ketma-ket oqim

Ketma-ketlik oqimi jarayon harakatlari yoki xoreografiya shartlari tashkil etilgan tartibni ko'rsatish uchun ishlatiladi.



Xabar oqimi

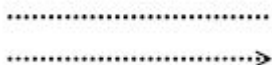
Bunday xabarlarini jo'natish va qabul qilishga tayyor bo'lgan ikki ishtirokchi o'rtasida xabar almashishni ko'rsatish uchun xabarlar oqimi

ishlatiladi. BPMN hamkorlik diagrammasida ikkita alohida hovuz ikkita jarayon ishtirokchisini ifodalaydi (masalan, PartnerEntities va/yoki PartnerRoles).



Uyushma

Assotsiatsiya BPMN grafik elementlari bilan ma'lumot va artefaktlar o'rtasida aloqa o'rnatishga xizmat qiladi. Matn izohlari va boshqa artefaktlar grafik elementlar bilan bog'lanishi mumkin. Assotsiatsiyaga ishora qiluvchi strelka oqim yo'nalishini ko'rsatadi (masalan, ma'lumotlar oqimi).



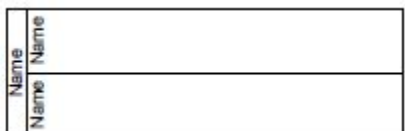
Hovuz

Hovuz hamkorlik ishtirokchisining grafik tasviridir. Hovuz, shuningdek, mas'uliyat sohasi yoki odatda B2B tipidagi vaziyatlarda mavjud bo'lgan boshqa hovuzlardan muayyan harakatlar to'plamini ajratish uchun mas'ul bo'lgan grafik konteyner sifatida ham harakat qilishi mumkin. Hovuzda bajariladigan jarayon ko'rinishidagi ichki qismlar bo'lishi mumkin. Agar hovuzda bunday ma'lumotlar bo'lmasa, u "qora quti" bo'lishi mumkin.



Trek

Bo'lak - bu jarayonning uzunligini vertikal yoki gorizontal ravishda uzaytiruvchi jarayon yoki hovuz ichidagi kichik bo'lim. Harakatlarni tashkil qilish va tasniflash uchun xizmat qiladi.



Ma'lumotlar obyekti

Ma'lumotlar obyekti qanday harakatlar bajarilishi kerakligi va/yoki bunday harakatlarning natijasi qanday bo'lishi haqida ma'lumot beradi. Ma'lumotlar obyektlari bitta obyekt yoki obyektlar to'plamini ifodalashi mumkin. Ma'lumotlar obyektining kirish va chiqish ma'lumotlari jarayon haqida bir xil ma'lumotlarni beradi.



Xabar

Xabar ikki ishtirokchi o'rtasidagi suhbat mazmunini ko'rsatish uchun ishlatiladi (ishtirokchilar biznes PartnerRole yoki biznes PartnerEntity yordamida aniqlanadi).



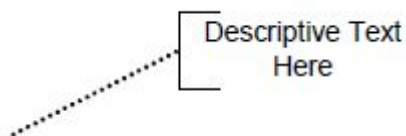
Guruh (bir xil toifadagi obyektlar guruhini o'z ichiga olgan blok)

Guruh - bir xil toifadagi grafik elementlarning birlashmasi (68-betga qarang). Ushbu turdagi guruhlash guruh ichidagi ketma-ketliklar oqimiga ta'sir qilmaydi. Kategoriya nomi diagrammada guruh nomi sifatida ko'rinadi. Kategoriyalar hujjatlashtirish yoki tahlil qilish uchun ishlatilishi mumkin. Guruhlar - obyektlar toifalarini diagrammada vizual tarzda ko'rsatishning bir usuli.



Matn izohi (Assotsiatsiya bilan bog'langan)

Matn izohlari - bu modelerga BPMN diagrammalari bilan ishlaydiganlar uchun matn ma'lumotlarini qo'shish imkonini beruvchi mexanizm.



BPMN modellashtirish elementlarining to'liq ro'yxati

Tadbir

Voqea - bu jarayon (237-bet) yoki uning xoreografiyasi (339-bet) davomida sodir bo'ladigan narsa. Voqea biznes jarayonining borishiga ta'sir qiladi va ko'pincha sabab (tetik) yoki ta'sir (natija) bo'ladi. Ichki markerlar yordamida turli triggerlarni yoki ularning natijalarini farqlash uchun mo'ljallangan bo'sh markazli doira shaklida tasvirlangan.

Hodisalarning biznes jarayoniga ta'siriga ko'ra, hodisalarning uch turi mavjud: boshlang'ich voqea, oraliq voqea va yakuniy voqea.



Oqim holati (masalan, boshlanish hodisasi, oraliq hodisa, yakunlanish hodisasi)

1. Boshlanish hodisasi
2. Oraliq hodisa
3. Tadbirni yakunlash

Nomidan ko'rinib turibdiki, boshlanish hodisasi ma'lum bir jarayon yoki xoreografiya qaysi nuqtada boshlanishini ko'rsatadi.

Boshlanish va tugatish hodisalari o'rtasida oraliq hodisa sodir bo'ladi. Oraliq hodisa jarayonning borishiga yoki xoreografiyaga ta'sir qiladi, lekin jarayonning boshlanishi yoki (darhol) oxiri bo'la olmaydi. Nomidan ko'rinib turibdiki, yakuniy hodisa ma'lum bir jarayon yoki xoreografiya qaysi nuqtada tugashini ko'rsatadi.

Начальное
событие



Промежуточное
событие



Конечное
событие



Boshlang'ich va ba'zi oraliq hodisalar ushbu turdagi hodisalarning paydo bo'lish sabablarini aniqlaydigan triggerlarga ega ("Boshlanish hodisasi" va "Oraliq hodisa" bo'limlariga qarang). Voqea sodir bo'lishi uchun juda ko'p sabablar bor. Yakuniy hodisalar ketma-ketliklar oqimining tugashi natijasida kelib chiqadigan natijani aniqlay oladi. Boshlanish hodisasi faqat triggerga javob berishi mumkin. Yakuniy hodisa faqat natijani yaratadi. Oraliq hodisalar tetikni ham qayta ishlashi, ham ishga tushirishi mumkin. Triggerlarni qayta ishlaydigan

hodisalar uchun markerlar to'ldirilmasdan ko'rsatiladi, triggerlarni boshlaydigan hodisalar uchun markerlar esa rang bilan ajratiladi.

Bundan tashqari, BPMN 1.1 da harakatning borishini to'xtatish uchun ishlatiladigan ba'zi hodisalar endi boshqacha ishlatilishi mumkin. Bunday hodisa nuqta chegaralari bo'lgan doira shaklida tasvirlangan (o'ngdagi rasmga qarang).

	"Catching"	"Throwing"	Non-interrupting
Message			
Timer			
Error			
Escalation			
Cancel			
Compensation			
Conditional			
Link			
Signal			
Terminate			
Multiple			
Parallel Multiple			

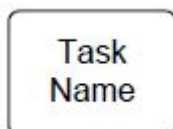
Harakat

Harakat - bu jarayon davomida kompaniya tomonidan bajariladigan ish uchun umumiy atama. Harakatlar elementar yoki elementar bo'lmagan (kompozitsiya) bo'lishi mumkin. Jarayon modelining bir qismi bo'lgan quyidagi harakatlar turlari ajratiladi: subprocess va Vazifa. Vazifa ham, kichik jarayon ham burchaklari yumaloq to'rtburchaklar shaklida tasvirlangan. Harakatlar standart jarayonlarda ham, xoreografiyalarda ham qo'llaniladi.



Vazifa (asosiy harakat)

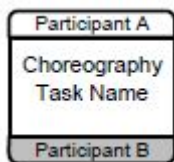
Vazifa - bu jarayonga kiritilgan elementar harakat (154-betga qarang). Jarayon kichikroq darajadagi tafsilotlarga bo'linmasa ishlatiladi.



Xoreografiya muammosi

Xoreografiya vazifasi xoreografiyaning elementar harakatini ifodalaydi. Bir yoki bir nechta xabar almashinuvini ko'rsatadi. Har bir

xoreografiya vazifasi ikkita ishtirokchini o‘z ichiga oladi. Xoreografiya topshirig‘ining nomi va ishtirokchilarning ismlari grafik yozuvni tashkil etuvchi uch xil qismda ko‘rsatiladi. Xoreografiya muammosi ishtirokchilarning ikki yoki undan ortiq qatorini va topshiriq nomi bilan bir qatorni ko‘rsatadi.

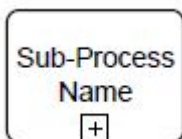


Jarayon/sub-jarayon (elementar bo‘lmagan harakat)

Pastki jarayon - bu jarayon yoki xoreografiya tarkibiga kiritilgan murakkab harakatdir. Sub-jarayonni kichik harakatlar to‘plamidan foydalangan holda tarkibiy qismlarga (jarayon yoki xoreografiya) bo‘lish mumkin.

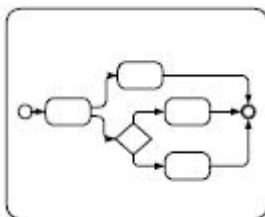
Yiqilgan quyi jarayon

Diagramma kichik jarayon tafsilotlarini ko‘rsatmaydi. Rasmning pastki markazida joylashgan ortiqcha belgisi faoliyat subprocess ekanligini va tafsilotlar pastroq darajada ekanligini ko‘rsatadi.



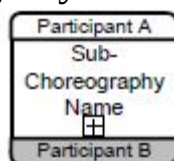
Kengaytirilgan pastki jarayon

Pastki jarayon chegaralari kengaytiriladi va (jarayon) tafsilotlarini chegaralar bo‘ylab ko‘rish mumkin. Shuni ta’kidlash kerakki, ketma-ketlik oqimi quyi jarayonlar chegaralarini kesib o‘tolmaydi.



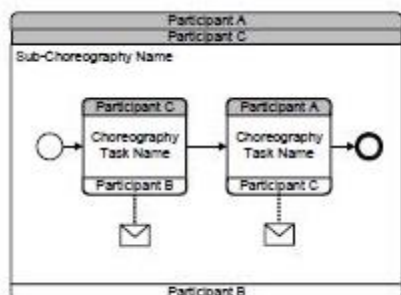
Yiqilgan subxoreografiya

Subxoreografiya tafsilotlari diagrammada ko‘rsatilmagan. Vazifa nomi satrining pastki markazidagi ortiqcha belgisi faoliyatning kichik jarayon ekanligini va tafsilotlar pastroq darajada ekanligini ko‘rsatadi.



Kengaytirilgan subxoreografiya

Subxoreografiya chegaralari kengaytiriladi va tafsilotlarni (xoreografiya) chegaralar bo'ylab ko'rish mumkin. Shuni ta'kidlash kerakki, ketma-ketliklar oqimi subxoreografiya chegaralarini kesib o'ta olmaydi.



Gateway

Shlyuzlar jarayonda va xoreografiyada ketma-ketliklar oqimining divergensiyasi va yaqinlashishini nazorat qilish uchun ishlatiladi. Shunday qilib, atama shoxlanish, bifurkatsiya, birlashma va bog'lovchi yo'llarni nazarda tutadi. Ichki belgilar biznes jarayonining xatti-harakatlarini nazorat qilish turini ko'rsatadi.



Shlyuzlarni boshqarish turlari

Shlyuzlar olmos shaklidagi shakllar bo'lib, oqim harakatini boshqarish turini ko'rsatadi. Quyidagi nazorat turlari ajratiladi:

- Istisno echimlar va birlashishlar. Eksklyuziv shlyuzlar va hodisalarga asoslangan shlyuzlar eksklyuziv qarorlar va birlashmalarni amalga oshirishi mumkin. Istisno shlyuzlar "X" belgisi bilan paydo bo'lishi yoki ko'rinmasligi mumkin.
- Voqealarga asoslangan shlyuzlar va parallel hodisalarga asoslangan shlyuzlar yangi jarayon misolining paydo bo'lishiga ta'sir qiladi.
- Shu jumladan shlyuz va birlashma echimlari.
- Murakkab shlyuzlar murakkab shartlar va vaziyatlarni ifodalaydi (masalan, 5 tadan 3 tasi).
- Parallel shlyuzlar bifurkatsiya va birlashishni ifodalaydi.

Har bir nazorat turi kiruvchi va chiquvchi oqimlarga ta'sir qiladi.



Ketma-ket oqim

Ketma-ketlik oqimi jarayon harakatlari (95-betga qarang) yoki xoreografiya shartlari (320-betga qarang) tashkil etilgan tartibni ko'rsatish uchun ishlatiladi.

Oddiy oqim

Oddiy oqim harakat chegarasiga birlashtirilgan oraliq hodisadan boshlanadigan ketma-ket oqim yo'llarini anglatadi.



Nazoratsiz oqim

Nazorat qilinmagan oqim deganda hech qanday sharoit yoki shlyuzlardan o'tmaydigan oqimlar ta'sir qilmaydigan oqimlar tushuniladi. Bunday oqimning eng oddiy misollari ikkita harakatni bog'laydigan bitta ketma-ketlik oqimi bo'ladi. Yoki harakatda yaqinlashuvchi yoki undan keyin ajralib chiqadigan ketma-ketlikning kompozit oqimi. Har bir nazoratsiz ketma-ketlik oqimi uchun resurs obyektidan ketma-ketlik oqimi orqali maqsadli obyektga o'tadigan token paydo bo'ladi.



Shartli oqim

Ketma-ketlik oqimi ketma-ketlik oqimidan foydalanish yoki yo'qligini aniqlash uchun ish vaqtida baholanadigan muayyan shartlarga bog'liq bo'lishi mumkin. Agar shartli oqim harakatdan kelib chiqsa, u holda o'qning boshida kichik olmos chiziladi (o'ngdagi rasmga qarang). Agar shartli oqim shlyuzdan kelsa, u holda olmos o'qning boshida joylashtirilmaydi (yuqoridagi rasmga qarang).



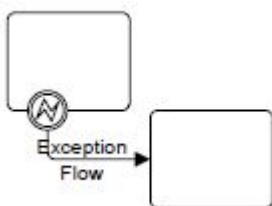
Standart oqim

Eksklyuziv va eksklyuziv bo'lmagan shlyuzlar uchun faqat bitta turdagi oqim ishlatiladi - standart shartli oqim. Bu oqim amallarni bajarishda boshqa barcha chiquvchi shartli oqimlar noto'g'ri bo'lsa ishlatiladi. Bunday ketma-ketlik oqimlarini tasvirlash uchun o'qning boshida chiziq qo'yiladi (o'ngdagi rasmga qarang).



Istisno oqimi

Istisno oqimi jarayonning normal oqimidan tashqarida sodir bo'ladi va jarayon davomida yuzaga keladigan oraliq hodisalarga asoslanadi.



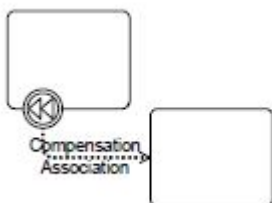
Xabar oqimi

Bunday xabarlarini jo'natish va qabul qilishga tayyor bo'lgan ikki ishtirokchi o'rtasida xabar almashishni ko'rsatish uchun xabarlar oqimi ishlatiladi. BPMN hamkorlik diagrammasida ikkita alohida hovuz ikkita jarayon ishtirokchisini ifodalaydi (masalan, PartnerEntities va/yoki PartnerRoles).



Kompensatsiya uyushmasi

Kompensatsiya qiluvchi assotsiatsiya odatdagi oqimdan tashqarida sodir bo'ladi va tranzaksiya xatosi yoki kompensatsiya tetiklash hodisasi bilan qo'zg'atilgan kompensatsiya qiluvchi oraliq hodisaga asoslanadi. Kompensatsiya uyushmasining maqsadi kompensatsiya harakati bo'lishi kerak.



Ma'lumotlar obyekti

Ma'lumotlar obyekti qanday harakatlar bajarilishi kerakligi va/yoki bunday harakatlarning natijasi qanday bo'lishi haqida ma'lumot beradi. Ma'lumotlar obyektlari bitta obyektga yoki obyektlar to'plamini

ifodalashi mumkin. Ma'lumotlar obyektining kirish va chiqish ma'lumotlari jarayon haqida bir xil ma'lumotlarni beradi.

Объект данных



Объект данных
(собрание)



Вход
данных



Выход
данных

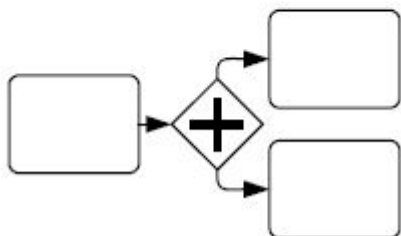
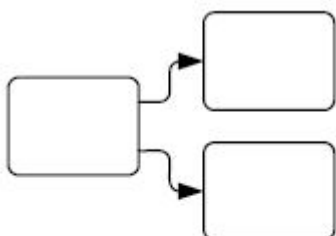


Bifurkatsiya

BPMNda "vilka" atamasi ikki yoki undan ortiq parallel yoʻllarga boʻlinish uchun ishlatiladi (bu boʻlinish "AND-Split" deb ataladi). Bifurkatsiya - bu jarayondagi ketma-ket harakatlardan koʻra parallel harakatlarni bajarish afzalroq boʻlgan nuqta.

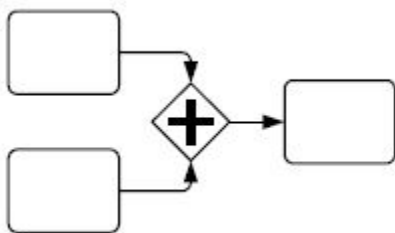
Bifurkatsiyaning ikki turi mavjud:

- Bir nechta chiquvchi ketma-ketlik oqimi (yuqoridagi oʻngdagi rasmga qarang) Bu koʻp holatlarda afzal koʻrilgan nazoratsiz oqimdir.
- Parallel Gateway (pastdagi oʻngdagi rasmga qarang). Kamroq ishlatiladi, odatda boshqa shlyuzlar bilan birgalikda.



Murakkab

BPMN da "ulanish" atamasi ikki yoki undan ortiq parallel marshrutlarni bittaga ulash uchun ishlatiladi (bu ulanish "AND-Join" yoki sinxronizatsiya deb ataladi). Parallel shlyuz bir nechta ketma-ket oqimlarning ulanishini xaritalash uchun ishlatiladi.

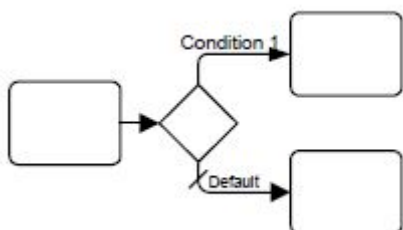


Yechim, ajralish nuqtasi

Qarorlar jarayon yoki xoreografiya ichida boshqaruv oqimi bir yoki bir nechta muqobil marshrutlar bo‘ylab harakatlanadigan shlyuzlardir.

Istisno

Qaror muqobil marshrutlar chiquvchi ketma-ketlik oqimidagi shartli iboralarga asoslangan filial nuqtasidir. Muqobil yo‘llardan faqat bittasini tanlash mumkin.

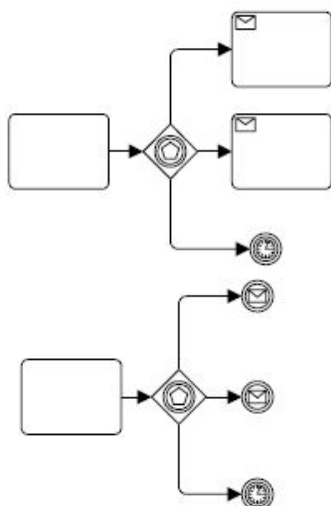


Tadbirga asoslangan

Ushbu qaror, muqobil marshrutlar jarayonning o‘sha nuqtasida yoki xoreografiyaga sodir bo‘lgan voqeaga asoslanadigan tarmoqlanish nuqtasini ifodalaydi. Muayyan hodisa, odatda xabarni qabul qilish, qaysi yo‘nalishni tanlashni belgilaydi. Taymer hodisasi kabi boshqa hodisa turlaridan ham foydalanish mumkin. Muqobil yo‘llardan faqat bittasini tanlash mumkin.

Xabarni qabul qilishning ikkita varianti mavjud:

- "Qabul qilish" tipidagi vazifalardan foydalanish (yuqori o‘ngdagi rasmga qarang).
- Xabar turidagi oraliq hodisalardan foydalanish (quyidagi o‘ngdagi rasmga qarang).

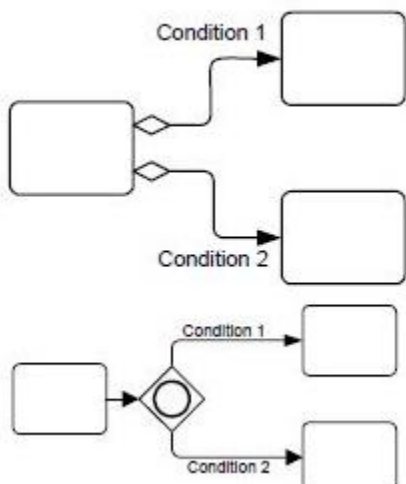


Istisno emas

Ushbu turdagi qaror shoxlanish nuqtasini ifodalaydi, bunda muqobil marshrutlar chiquvchi ketma-ketlik oqimidagi shartli ifodalarga asoslanadi. Qaysidir ma'noda qarorning bu turi o'zaro bog'langan mustaqil ikkilik qarorlar guruhidir (Ha/Yo'q). Har qanday marshrut mustaqil bo'lganligi sababli, marshrutlarning har qanday kombinatsiyasidan foydalanish mumkin (noldan maksimal kombinatsiyalar sonigacha). Biroq, diagrammalarni qurishda kamida bitta marshrut tanlanishi kerakligini hisobga olish kerak. Kamida bitta marshrut tanlanishini tekshirish uchun standart shartdan foydalanish mumkin.

Bunday yechimlarning ikki turi mavjud:

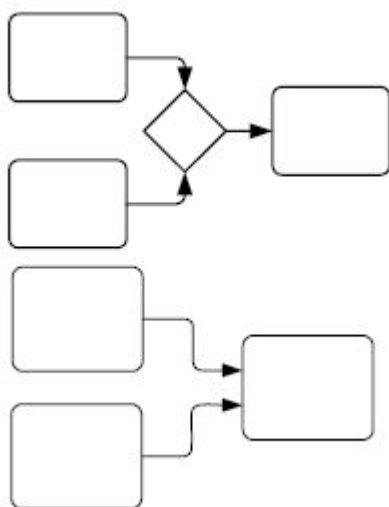
- Birinchi tur shartli ketma-ketlik oqimlari to'plamidan foydalanadi. Ular diagrammada olmos bilan ko'rsatilgan.
- Ikkinchi tur eksklyuziv bo'lmagan shlyuzlardan foydalanadi.



Birlashish

"Birlashtirish" atamasi BPMN-da ikki yoki undan ortiq marshrutlarning bittaga eksklyuziv birikmasiga ishora qilish uchun ishlatiladi (shuningdek, "OR-Join" deb ham ataladi). Eksklyuziv birlashma eshigi bir nechta ketma-ketlik oqimlarining birlashishini ko'rsatish uchun ishlatiladi.

Agar barcha kiruvchi ketma-ketlik oqimlari muqobil bo'lsa, u holda shlyuzdan foydalanilmasligi mumkin. Bu nazoratsiz oqimning bir xil ta'sirga ega ekanligini anglatadi.

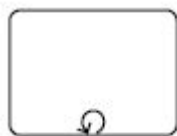


Siklik

BPMN-da jarayon davomida velosipedda harakatlanishning ikkita mexanizmi mavjud.

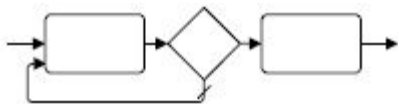
Siklik harakat

Vazifalar va kichik jarayonlarning atributlari ularning faqat bir marta takrorlanishi yoki bajarilishini belgilaydi (189-betga qarang). Ikki turdagi loop mavjud: standart va ko'p nusxali. Grafik jihatdan, sikl harakatning pastki markazida kichik marker sifatida ko'rsatilgan.



Oqimning ketma-ketligi

Looplar ketma-ketlik oqimini "qarama-qarshi" obyektga biriktirish orqali yaratilishi mumkin. Obyekt, agar u bir qator boshqa ketma-ketlik oqimlariga olib keladigan chiquvchi ketma-ketlik oqimiga ega bo'lsa, qarama-qarshidir, ularning oxirgisi asl obyekt uchun kiruvchi faoliyat oqimidir.



Ko'p nusxali

Vazifalar va kichik jarayonlarning atributlari ularning faqat bir marta takrorlanishi yoki bajarilishini belgilaydi. Harakatning pastki qismining markazidagi uchta gorizontal chiziq ketma-ket ko'p misolni ko'rsatadi (yuqori o'ngdagi rasmga qarang). Harakatning pastki qismining markazidagi uchta vertikal chiziq parallel ko'p misolni bildiradi (o'ngdagi quyidagi rasmga qarang).

Sequential



Parallel



Jarayonni tugatish (jarayonni to'xtatib qo'ygan nazoratsiz hodisa)

Jarayonni to'xtatish - bu jarayonning qaysi qismida kutilgan jarayon kechikishi sodir bo'lishini ko'rsatadigan bo'limi. Jarayonning haqiqiy harakatini ko'rsatish uchun oraliq harakat ishlatiladi (yuqoridagi o'ngdagi rasmga qarang). Bundan tashqari, to'xtab qolgan artefakt modellashiruvchi yoki simulyatsiya dasturining ixtiyoriga ko'ra voqealarga ip ichidagi kechikish joyini ta'kidlash uchun tayinlanishi mumkin.



Tranzaksiya

Tranzaksiya - bu muayyan protokol bilan qo'llab-quvvatlanadigan kichik jarayon bo'lib, barcha ishtirokchilar harakatni bajarish yoki rad etish kerakligiga rozi bo'lishini ta'minlaydi. Harakat atributlari harakatning tranzaksiya ekanligini aniqlaydi. Ikki tomonlama chegara ushbu kichik jarayon tranzaksiya ekanligini ko'rsatadi.



Kiritilgan/ichiga qo'yilgan kichik jarayon (chiziqli blok)

Kiritilgan (yoki ichki o'rnatilgan) kichik jarayon - bu asosiy jarayon bilan bir xil ma'lumotlar to'plamiga ega bo'lgan faoliyat. Bunday kichik jarayon qayta ishlatilishi mumkin bo'lgan va asosiy jarayondan foydalanadigan mustaqil kichik jarayonga qarama-qarshidir. Ketma-ketlik oqimidan foydalanilganda, ma'lumotlar kiritish quyi jarayoniga emas, balki asosiy jarayonga uzatilishi kerak.

Sahifa ulagichi

Chop etish uchun foydalanilgan ushbu element ketma-ketlik oqimining oldingi sahifada tugashini va keyingi sahifada qaerdan boshlanishini ko'rsatadi. Oraliq bog'lovchi hodisa sahifa bog'lovchisi sifatida ishlatilishi mumkin.



Diagrammadagi matn, ranglar, o'lcham va chiziqlar

Matnni izohlash obyektlari BPMN diagrammasida obyektlar jarayoni yoki atributlari haqida qo'shimcha ma'lumotlarni ko'rsatish uchun modeler tomonidan ishlatilishi mumkin.

- BPMN elementlari (masalan, oqim obyektlari) yorliqlarga ega bo'lishi mumkin (masalan, oqim nomi va/yoki boshqa atributlar) shakl ichida, tepasida yoki ostida joylashgan. Yorliqlar modeler yoki modellash vositasining maqsadlariga qarab har qanday joyda va har qanday yo'nalishda bo'lishi mumkin.

- Grafik elementni to'ldirish oq yoki shaffof bo'lishi mumkin.

- Belgilash modeler yoki modellash vositasining maqsadiga muvofiq boshqa to'ldirish ranglarini ishlatish uchun kengaytirilishi mumkin (masalan, obyekt atributining qiymatini ta'kidlash uchun). Shunga qaramasdan,

- Oqimning keyingi rivojlanishini belgilaydigan hodisa belgilari to'q rangda bo'lishi SHART.

- Harakat tashabbuskori bo‘lmagan xoreografiya va qo‘shimcha xoreografiya ishlaridagi ishtirokchi treklari engil to‘ldirishga ega bo‘lishi KERAK.

- Oqimli obyektlar va markerlar modellashtiruvchi yoki modellash vositasi maqsadlariga mos keladigan har qanday o‘lchamda bo‘lishi mumkin.

- Grafik elementlarni chizish uchun ishlatiladigan chiziqlar qora bo‘lishi mumkin.

- Belgilash modeler yoki modellash vositasining maqsadiga mos keladigan boshqa chiziq ranglarini ishlatish uchun kengaytirilishi mumkin (masalan, obyekt atributining qiymatini ta’kidlash uchun).

Nazorat savollari

1. BPMN 2.0 nima va uning asosiy maqsadi nima?
2. BPMN 2.0 ning asosiy diagramma turlari qanday?
3. BPMN 2.0 diagrammasining asosiy elementlari nimalardan iborat?
4. Voqealar (Events) nima va ular qanday turlarga bo‘linadi?
5. Shlyuzlar (Gateways) qanday turlarga bo‘linadi va ularning har biri qanday ishlaydi?
6. Aktivliklar (Activities) qanday turlarga bo‘linadi va ularning vazifasi nima?
7. BPMN 2.0 diagrammasida qo‘shimcha elementlar qanday qo‘llaniladi?
8. BPMN 2.0 diagrammasini yaratish jarayonida qanday bosqichlar amalga oshiriladi?
9. BPMN 2.0 spetsifikatsiyasi qanday standartlarga mos keladi?
10. BPMN 2.0 metodologiyasining afzalliklari va kamchiliklari qanday?

10-MAVZU. BPMN NOTATSIYASI SUBKLASSLARI

Reja:

1. BPMN Notatsiyasining Asosiy Elementlari
2. BPMN Notatsiyasining Asosiy Elementlari
3. Shlyuzlar (Gateways) Subklasslari

Shuni ta'kidlash kerakki, BPMNni yaratish sabablaridan biri ham oddiy, ham murakkab biznes jarayonlari modellarini loyihalash uchun oddiy mexanizmni yaratish zarurati edi. Ushbu ikki qarama-qarshi talabni qondirish uchun yozuvning grafik elementlarini toifalarga tizimlashtirish yondashuvi qo'llanildi. Natijada odamlar bilan ishlashga ruxsat beruvchi notatsiya toifalarining kichik ro'yxati paydo bo'ldi BPMN diagrammasi, elementlarning asosiy turlarini osongina tanib olish va diagrammalarni to'g'ri o'qish. Elementlarning asosiy toifalari ichki o'zgarishlarga, shuningdek, tushunish qulayligi uchun diagrammaning umumiy tuzilishiga sezilarli o'zgarishlar kiritmasdan murakkablik talablariga javob beradigan ma'lumotni qo'shish imkonini beradi.

Elementlarning beshta asosiy toifasi mavjud:

1. Oqim obyektlari;
2. Ma'lumotlar
3. Obyektlarni ulash;
4. Mas'uliyat sohalari (suzish yo'llari);
5. Artefaktlar.

Oqim elementlari biznes jarayonining oqimini belgilaydigan eng muhim grafik elementlardir. *Oqim elementlari*, o'z navbatida, quyidagilarga bo'linadi:

1. Tadbirlar;
2. Faoliyat;
3. Shlyuzlar.

Grafikdagi *ma'lumotlar* quyidagi to'rtta elementdan biri bilan ifodalanishi mumkin:

1. Ma'lumotlar obyektlari
2. Ma'lumotlarni kiritish
3. Ma'lumotlar chiqishlari
4. Ma'lumotlar do'konlari

Bir-biri bilan va boshqa elementlar bilan aloqa o'rnatadigan oqim elementlarining to'rtta turi mavjud:

1. Ketma-ket oqim;
2. Xabarlar oqimi;
3. Uyushma;
4. Ma'lumotlar assotsiatsiyasi.

Mas'uliyat sohalaridan foydalangan holda asosiy modellash elementlarini guruhlashning ikki yo'li mavjud:

1. Hovuz yordamida guruhlash;
2. Lane yordamida guruhlash.

Artefaktlar jarayon haqida qo'shimcha ma'lumot qo'shish uchun ishlatiladi.

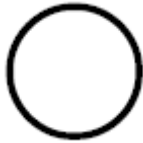
Ikkita standart artefakt mavjud, ammo ular **biznes jarayonlari modellari yoki modellashtirish dasturlarini ishlab chiquvchilarga kerakli miqdordagi artefaktlarni** qo'shishni taqiqlamaydi.

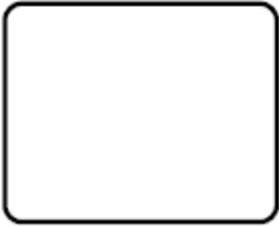
Foydalanuvchilarning keng doirasi uchun, shuningdek, vertikal bozorlar uchun *Artefaktlarning* to'liq ro'yxatini standartlashtirish mumkin. Ayni paytda *artefaktlarning* joriy ro'yxati quyidagi elementlarni o'z ichiga oladi:

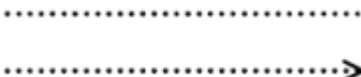
1. Guruh (guruh);
2. Matnga izoh.

Modellashtirishning asosiy grafik elementlari

10.1-jadvalda grafik belgilar yordamida tasvirlangan asosiy grafik modellash elementlari ro'yxati keltirilgan.

Element	Tavsif	Belgilash
Tadbir	Voqea - bu biznes jarayoni yoki uning xoreografiyasi davomida sodir bo'ladigan narsa. Voqea biznes jarayonining borishiga ta'sir qiladi va ko'pincha sabab (tetik) yoki ta'sir (natija) bo'ladi. Turli triggerlarning ichki belgilari yoki ularning natijalari bilan farqlash uchun mo'ljallangan, erkin markazga ega bo'lgan	

	doira shaklida tasvirlangan. Voqealarning biznes-jarayonning borishiga ta'siriga ko'ra, uch tur mavjud: Boshlanish hodisasi (Boshlanish), Oraliq voqea (Oraliq) va Yakuniy hodisa (Yakunlash).	
Faoliyat	Harakat - bu ish jarayonida ijrochi tomonidan bajariladigan ishni bildiruvchi umumiy atama. Harakatlar elementar yoki elementar bo'lmagan (kompozitsiya) bo'lishi mumkin. Jarayon modelining bir qismi bo'lgan quyidagi turdagi harakatlar mavjud: Sub-jarayon va Vazifa. Vazifa ham, pastki jarayon ham burchaklari yumaloq to'rtburchaklar shaklida tasvirlangan. Barcha harakatlar standart jarayonlarning ham, xoreografiyaning ham elementlari bo'lishi mumkin.	
Gateway (shlyuz)	Shlyuzlar jarayonda ham, xoreografiyada ham ish oqimining divergensiyasi va konvergensiyasini boshqarish uchun	

	ishlatiladi. Shunday qilib, bu atama tarmoqlanish, vilkalar, birlashtirish va ulash marshrutlarini nazarda tutadi. Ichki belgilar biznes jarayonining rivojlanishi ustidan nazorat turini ko'rsatadi.	
Oqim ketma-ketligi	Ish jarayoni jarayon harakatlari yoki xoreografiya shartlarini tashkil qilish tartibini ko'rsatish uchun ishlatiladi.	
Xabarlar oqimi	Xabarlar oqimi ushbu xabarlarni jo'natish va qabul qilishga tayyor bo'lgan ikki ishtirokchi o'rtasidagi xabar almashinuvini ko'rsatish uchun ishlatiladi. BPMN o'zaro ta'sir diagrammasida ikkita alohida hovuz jarayonning ikkita ishtirokchisini (biznes sub'ektlari yoki biznes rollarini) ifodalaydi.	
Uyushma	Assotsiatsiya ma'lumotlar yoki artefaktlar (oqim elementlari bilan bog'liq bo'lmagan obyektlar) va oqim elementlari o'rtasida aloqa o'rnatish uchun xizmat qiladi. Matn	

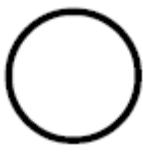
	obyektlari, shuningdek, oqim elementlari bilan bogʻliq boʻlmagan grafik obyektlarni oqim elementlari bilan bogʻlash mumkin. Agar kerak boʻlsa, Assotsiatsiya oqim yoʻnalishini koʻrsatishi mumkin (masalan, ma'lumotlar oqimi).	
Hovuz	Hovuz oʻzaro ta'sir ishtirokchisini ifodalaydi. Hovuz, shuningdek, Mas'uliyat sohasi yoki odatda biznesdan biznesga (B2B) vaziyatlarda topiladigan boshqa Hovuzlar bilan bogʻliq muayyan harakatlar toʻplamini ajratish uchun mas'ul boʻlgan grafik konteyner sifatida ham harakat qilishi mumkin. Hovuzda ishlayotgan jarayon haqida qoʻshimcha ma'lumotlar boʻlishi mumkin. Agar Hovuzda bunday ma'lumotlar boʻlmasa, u "qora quti" boʻlishi mumkin.	Name
Lane	Trek rollarning taqsimlanishini koʻrsatish uchun ishlatiladi va vertikal yoki gorizontal boʻlishi mumkin	Name Name

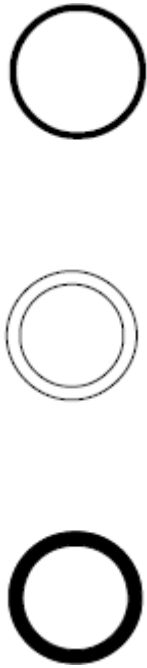
	(shuningdek, Hovuzning ichki maydonini ajratish uchun ham foydalanish mumkin). Harakatlarni tashkil qilish va tasniflash uchun xizmat qiladi.	
Ma'lumotlar obyekti	Ma'lumotlar obyekti qanday harakatlar bajarilishi kerakligi va/yoki bu harakatlarning natijasi nima ekanligi haqida ma'lumot beradi. U bitta nusxada ham, bir nechta nusxada ham tasvirlanishi mumkin. Ma'lumotlar obyektining kirish va chiqish ma'lumotlari jarayon haqidagi bir xil ma'lumotlarni ifodalaydi.	
Xabar	Xabar biznes jarayonidagi ikki Ishtirokchi o'rtasidagi o'zaro munosabatlarning mohiyatini ko'rsatish uchun ishlatiladi (Ishtirokchilar biznes PartnerRole yoki biznes PartnerEntity buyruqlari bilan belgilanadi).	
Guruh (bir xil toifadagi obyektlar guruhini o'z ichiga olgan blok)	Guruh bir xil toifaga kiruvchi grafik elementlarni guruhlash uchun mo'ljallangan. Ushbu guruhlash ish jarayoniga ta'sir qilmaydi. Biznes jarayonlari	

(Guruh)	diagrammasida guruhlangan elementlar tegishli bo'lgan toifa nomi guruh nomi sifatida ko'rinadi. Ushbu turdagi guruhlash hujjatlashtirish yoki tahlil qilish uchun ishlatilishi mumkin. Grafik jihatdan, Guruhlar obyekt toifalari kabi ko'rsatiladi.	
Matn izohi (Assotsiatsiya bilan bog'liq) (Matnga izoh)	Matn izohlari biznes-jarayon modeleriga BPMN diagrammalari bilan ishlaydiganlar uchun qo'shimcha ma'lumotlarni taqdim etish imkonini beruvchi mexanizmdir.	

Biznes jarayonlari diagrammalarining grafik elementlarining to'liq ro'yxati

10.2-jadvalda grafik belgilar yordamida tasvirlangan biznes jarayonlarini modellashtirishning asosiy grafik elementlarining to'liqroq ro'yxati keltirilgan.

Element	Tavsif	Belgilash
Tadbir	Voqea - bu biznes jarayoni yoki uning xoreografiyasi davomida sodir bo'ladigan narsa. Voqea biznes jarayonining borishiga ta'sir qiladi va ko'pincha sabab (tetik) yoki ta'sir (natija)	

	bo‘ladi. Turli triggerlarning ichki belgilari yoki ularning natijalari bilan farqlash uchun mo‘ljallangan, erkin markazga ega bo‘lgan doira shaklida tasvirlangan. Voqealarning biznes-jarayonning borishiga ta'siriga ko‘ra, uch tur mavjud: Boshlanish hodisasi (Boshlanish), Oraliq voqea (Oraliq) va Yakuniy hodisa (Yakunlash).	
Oqim o‘lchami (masalan, Boshlanish hodisasi, Oraliq hodisa, yakuniy hodisa) Boshlanish hodisasi Oraliq hodisa Tadbirni yakunlash	Nomidan ko‘rinib turibdiki, Boshlanish hodisasi ma'lum bir jarayon yoki xoreografiya qaysi nuqtada boshlanishini ko‘rsatadi. Boshlanish va tugatish hodisalari bilan chegaralangan segmentda oraliq	

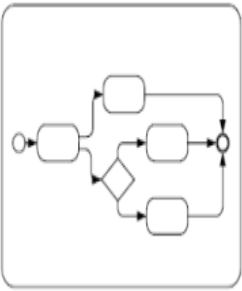
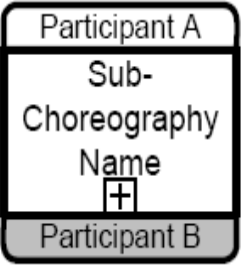
	hodisa sodir bo'ladi. Oraliq hodisa jarayon yoki xoreografiya jarayoniga ta'sir qiladi, ammo bu jarayonning boshlanishi yoki darhol yakunlanishi bo'lishi mumkin emas. Nomidan ko'rinib turibdiki, Yakuniy voqea ma'lum bir jarayon yoki xoreografiya qaysi nuqtada tugashini ko'rsatadi.	
O'lcham turi (masalan, Aniqlanmagan, Xabar, Taymer, Xato, Bekor qilish, Kompensatsiya, Shart, Aloqa, Signal, Ko'p, Tugatish)	Boshlanish va ba'zi Oraliq hodisalarda ushbu turdagi hodisalarning paydo bo'lish sabablarini aniqlaydigan triggerlar mavjud (quyida Boshlang'ich voqea va Oraliq hodisa bo'limlariga qarang). Voqea sodir bo'lishiga olib keladigan ko'plab sabablar mavjud. Yakuniy hodisalar ish	The legend displays various BPMN event symbols categorized into three groups: 'Catching' (solid shapes), 'Throwing' (dashed shapes), and 'Non-Interrupting' (circles with a thick border). The categories listed on the left are: Message, Timer, Error, Escalation, Cancel, Compensation, Conditional, Link, Signal, Terminate, Multiple, Parallel, and Multiple. Each category has corresponding symbols for catching, throwing, and non-interrupting types.

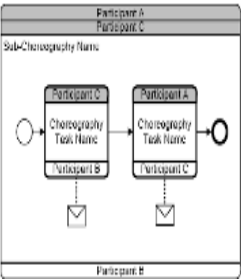
	oqimining tugashi natijasida kelib chiqadigan natijani aniqlashi mumkin. Faqat triggerlarni qayta ishlovchi Start hodisasidan farqli o'laroq, Intermediate hodisasi ham triggerlarni boshqarishi, ham ularni ko'tarishi mumkin. Yakuniy hodisa faqat natijani aniqlaydi (tetikni boshlaydi). Triggerlarni qayta ishlaydigan hodisalar uchun markerlar to'ldirilmasdan ko'rsatiladi, triggerlarni boshlaydigan hodisalar uchun markerlar to'ldiriladi. Bundan tashqari, BPMN 1.1 da Harakatning borishini to'xtatish uchun foydalaniladigan Voqealarning ayrim turlari ushbu nashrda boshqa	
--	---	--

	maqsadlarda ham qo'llanilishi mumkin. Bunday hodisa nuqta chegaralari bo'lgan doira shaklida tasvirlangan (o'ngdagi Hodisalar qatoriga qarang).	
Faoliyat	Harakat - bu ish jarayonida ijrochi tomonidan bajariladigan ishni bildiruvchi umumiy atama. Harakatlar elementar yoki elementar bo'lmagan (kompozitsiya) bo'lishi mumkin. Jarayon modelining bir qismi bo'lgan quyidagi turdagi harakatlar mavjud: Sub-jarayon va Vazifa. Vazifa ham, pastki jarayon ham burchaklari yumaloq to'rtburchaklar shaklida tasvirlangan. Barcha harakatlar standart	

	jarayonlarning ham, xoreografiyaning ham elementlari bo‘lishi mumkin.	
Vazifa (elementar harakat) (topshiriq)	Vazifa - bu jarayonga kiritilgan elementar harakat. Jarayon ushbu Modelda batafsil ko‘rsatilmagan bo‘lsa ishlatiladi.	Task Name
Xoreografiya vazifasi	Xoreografiya vazifasi xoreografiya ichidagi elementar harakatdir. Xabarning bir yoki bir nechta nusxalarini ko‘rsatadi va kamida ikkita Ishtirokchi mavjudligini bildiradi. Xoreografiya topshirig‘ining nomi va Ishtirokchilarning ismlari ushbu grafik elementning uch xil qismida ko‘rsatilgan. Shunday qilib, grafik jihatdan	Participant A Choreography Task Name Participant B

		Xoreografiya topshirig'i ishtirokchilarning ismlari (ikki yoki undan ko'p) bilan treklarga bo'linishi kerak, shuningdek, ushbu vazifa nomi uchun mo'ljallangan trekni o'z ichiga olishi kerak.	
Jarayon/pastki jarayon (elementar bo'lmagan harakat) (jarayon/pastki jarayon)		Quyi jarayon - bu jarayonga kiritilgan murakkab harakat. Ushbu turdagi harakat qo'shma hisoblanadi, chunki sub-faoliyatlardan foydalanish orqali tarkibiy qismlarga (Jarayon, Xoreografiya) bo'linishi mumkin.	Keyingi 4 ga qarang raqamlar
Yiqilgan jarayon (Yiqilgan jarayon)	pastki quyi	Diagrammada quyi jarayonning tafsilotlari ko'rsatilmagan. Plyus belgisi rasmning pastki qismining markazida joylashgan bo'lib,	

		pastki jarayonni bildiradi va bu faoliyat sub-jarayon ekanligini ko'rsatadi. Bunday holda, jarayon tafsilotlari quyi darajada bo'ladi.	
Kengaytirilgan Subprocess (Kengaytirilgan jarayon)	quyi	Subprocess chegaralari kengaytirildi. Tafsilotlar chegaralar ichida ko'rinadi. Shuni ta'kidlash kerakki, Ish jarayoni quyi jarayon chegaralarini kesib o'ta olmaydi.	
Yiqilgan xoreografiya	pastki	Diagrammada Subxoreografiya tafsilotlari ko'rsatilmagan. Plyus belgisi Vazifalar deb nomlangan trekning pastki markazida joylashgan va bu harakat pastki jarayon ekanligini bildiradi. Bunday holda, Xoreografiya tafsilotlari pastki darajada.	

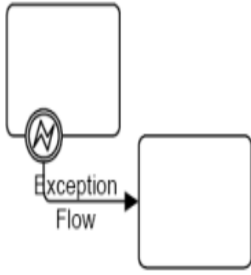
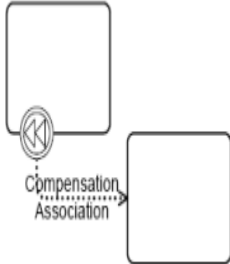
Kengaytirilgan subxoreografiya (Kengaytirilgan pastki xoreografiya) -	Subxoreografiya chegaralari kengaytirildi. Tafsilotlar chegaralar ichida ko‘rinadi. Shuni ta’kidlash kerakki, Ish oqimi sub-xoreografiya chegaralarini kesib o‘tolmaydi.	
Gateway (shlyuz)	Shlyuzlar bir nechta jarayon va xoreografiya ish oqimlarining divergensiya va yaqinlashuvini boshqarish uchun ishlatiladi. Shunday qilib, bu atama tarmoqlanish, vilkalar, birlashtirish va ulash marshrutlarini nazarda tutadi. Oqim yo‘nalishini farqlash uchun mo‘ljallangan ichki belgilarni o‘z ichiga olishi mumkin.	

Gateway turlari (Gateway boshqaruv turlari)	Shlyuzlar - olmoshaklidagi raqamlar - oqimlarga ta'sir qiladi. Quyidagi turdagi shlyuzlar ajralib turadi: Eksklyuziv shartlar va uyushmalar. Eksklyuziv va voqeaga asoslangan bo'lishi mumkin. Ushbu turdagi Gateway "X" belgisi bilan yoki bo'lmasdan ko'rsatilishi mumkin. Voqealarga asoslangan shlyuzlar va Parallel voqealarga asoslangan shlyuzlar jarayonning yangi nusxasini boshlaydi. Inklyuziv shartlar va uyushmalar. Murakkab sharoit va vaziyatlarni ifodalovchi murakkab shlyuzlar (masalan, 5 tadan 3 tasi).	Exclusive  or  Event-Based   Parallel Event-Based  Inclusive  Complex  Parallel 
--	---	--

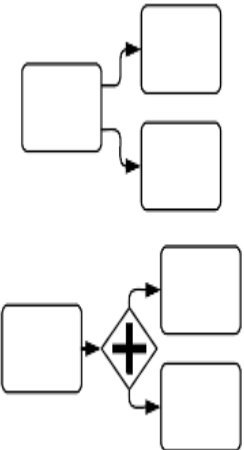
	Bifurkatsiya va birlashishni ifodalovchi parallel shlyuzlar. Har bir shlyuz turi kiruvchi va chiquvchi oqimlarga ta'sir qiladi.	
Ish jarayoni (Oqim ketma-ketligi)	Ish jarayoni jarayon yoki xoreografiya faoliyatining bajarilishi tartibini ko'rsatish uchun ishlatiladi.	Keyingi 7 ga qarang raqamlar
Standart ish jarayoni (Oddiy oqim)	Standart ish jarayoni Boshlanish hodisasidan boshlanadigan va Harakatlar oqimini kuzatib boruvchi oqimlarga ishora qiladi.	
Nazoratsiz ish jarayoni (Nazoratsiz oqim)	Nazorat qilinmagan oqim deganda hech qanday sharoit ta'sir qilmaydigan oqimlar yoki shlyuzlardan o'tmaydigan oqimlar tushuniladi. Eng oddiy misollar	

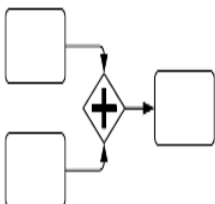
	Nazorat qilinmagan ish jarayoni ikkita Faoliyatni birlashtirgan yagona Ish oqimi yoki Faoliyatga yaqinlashadigan yoki undan ajralib turadigan kompozit ish oqimi bo'lishi mumkin. Har bir boshqarib bo'lmaydigan oqim uchun resurs obyektidan maqsadli obyektga o'tadigan "token" hosil bo'ladi.	
Shartli ish jarayoni (Shartli oqim)	Ish oqimi ipdan foydalaniladimi yoki yo'qligini aniqlash uchun bajarilish vaqti bo'yicha baholanadigan shartli iboralarga bog'liq bo'lishi mumkin (masalan, token Ish oqimi bo'ylab harakatlanadimi). Agar Shartli operatsiyalar oqimi Harakatdan kelib chiqsa, u holda chiziqning tagida	

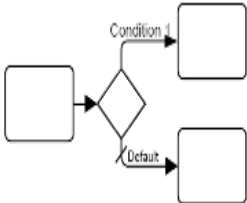
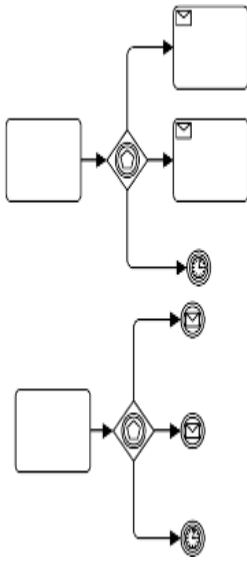
	kichik olmos chiziladi (o'ngdagi rasmga qarang). Agar shartli operatsiyalar oqimi Gatewaydan chiqsa, u holda chiziqning pastki qismida olmos bo'lmaydi (yuqoridagi qator rasmiga qarang).	
Standart ish jarayoni (Standart oqim)	Ma'lumotlarga asoslangan eksklyuziv va nomaxsus shartlar uchun faqat bitta oqim turi mavjud, ya'ni standart shartli oqim. Ushbu turdagi ish jarayoni boshqa barcha chiquvchi Shartli ish oqimlari amalni bajarish vaqtida haqiqiy bo'lmasa ishlatiladi. Bunday operatsiya oqimlarini tasvirlash uchun chiziqning tagida joylashgan diagonal chiziqdan foydalaniladi (o'ngdagi rasmga qarang).	

Istisno oqimi (Istisno oqimi)	Istisno oqimi Standart oqimdan tashqarida sodir bo‘ladi. Jarayon davomida sodir bo‘ladigan oraliq hodisalarga asoslangan.	 The diagram shows two rectangular process boxes. The left box has a circle with a lightning bolt symbol on its bottom-left corner. An arrow points from this symbol to the right box. The text "Exception Flow" is written below the arrow.
Xabar oqimi (Xabar oqimi)	Xabarlar oqimi xabarlarni qabul qilish va jo‘natish uchun tayyor bo‘lgan ikkita Jarayon ishtirokchilari o‘rtasidagi xabarlar oqimini ko‘rsatish uchun ishlatiladi. O‘zaro ta'sir diagrammasida ikkita alohida Hovuz ikkitani ifodalaydi Jarayon ishtirokchilari (masalan, <i>PartnerEntities</i> va/yoki <i>PartnerRoles</i>).	 The diagram shows a horizontal dashed line with a key symbol at the left end and a right-pointing arrowhead at the right end.
Kompensatsiya uyushmasi (Kompensatsiya Uyushma)	Kompensatsion assotsiatsiya operatsiyalarning standart oqimidan tashqarida sodir bo‘ladi. Ushbu turdagi	 The diagram shows two rectangular process boxes. The left box has a circle with a double arrow symbol on its bottom-left corner. A dashed arrow points from this symbol to the right box. The text "Compensation Association" is written below the arrow.

	uyushmaning asosi oraliq hisoblanadi Tranzaktsiya paytida qilingan xato yoki tetikni ishga tushirgan kompensatsiya hodisasi tufayli yuzaga kelgan "kompensatsiya" hodisasi. Kompensatsiya uyushmasining maqsadi kompensatsiya harakati bo'lishi kerak.	
Ma'lumotlar obyekti (Ma'lumotlar obyekti)	Biroq, Ma'lumotlar obyekti qanday harakatlar bajarilishi kerakligi va/yoki bu harakatlarning natijasi qanday bo'lishi haqida ma'lumot beradi. U bitta nusxada ham, bir nechta nusxada ham tasvirlanishi mumkin. Ma'lumotlar obyekti kirish va chiqish ma'lumotlari jarayon haqidagi bir xil	Data Object  Data Objec (Collection)  Data Input Data Output  

	ma'lumotlarni ifodalaydi.	
Xabar	Xabar biznes jarayonidagi ikki Ishtirokchi o'rtasidagi o'zaro munosabatlarning mohiyatini ko'rsatish uchun ishlatiladi (Ishtirokchilar biznes PartnerRole yoki biznes PartnerEntity buyruqlari bilan belgilanadi).	
Vilka	"Forking" atamasi BPMN-da ikki yoki undan ortiq parallel yo'llarga bo'linish uchun ishlatiladi (shuningdek, "AND-Split" deb ataladi). Bifurkatsiya ketma-ket emas, balki harakatlarning parallel bajarilishiga ustunlik berilganda sodir bo'ladi. Bifurkatsiyaning ikki turi mavjud:	

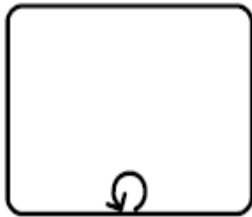
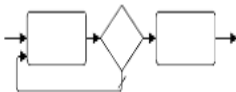
	Bir nechta chiquvchi ish oqimi (yuqoridagi o'ngdagi rasmga qarang) Ko'p holatlarda afzal ko'rilgan nazoratsiz ish oqimini ifodalaydi. Parallel Gateway (quyidagi o'ngdagi rasmga qarang). Kamroq ishlatiladi, odatda boshqa turdagi shlyuzlar bilan birgalikda.	
Qo'shilish	"Qo'shilish" atamasi BPMN-da ikki yoki undan ortiq parallel marshrutlarni bittaga (shuningdek, AND-iyun yoki sinxronizatsiya deb ataladi) birlashtirishga ishora qilish uchun ishlatiladi. Parallel Gateway bir nechta iplarni birlashtirish uchun mo'ljallangan.	
Holati, filial nuqtasi (qaror, filial nuqtasi)	Shartlar jarayon yoki xoreografiya ichidagi shlyuzlar	Keyingi 5 ga qarang hujayralar

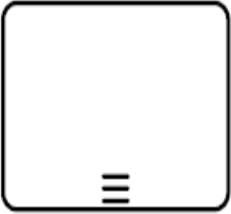
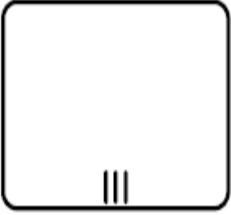
	bo‘lib, unda boshqaruv oqimi bir yoki bir nechta muqobil marshrutlar bo‘ylab harakatlanadi.	
Eksklyuziv shlyuz (ekslyuziv)	Eksklyuziv shlyuz - bu filial punkti bo‘lib, unda marshrut tanlash chiquvchi ish oqimida saqlangan shartli ifodalarga asoslanadi. Bunday holda, taklif qilingan marshrutlardan faqat bittasini tanlash mumkin.	 <pre> graph LR Start([]) --> Gateway{ } Gateway -- "Condition 1" --> End1([]) Gateway -- "Default" --> End2([]) </pre>
Voqealarga asoslangan shlyuz (Tadbirga asoslangan)	Ushbu turdagi shlyuz filial nuqtasini ifodalaydi, unda marshrutni tanlash jarayon yoki xoreografiya paytida sodir bo‘lgan voqeaga asoslanadi. Bitta hodisa, odatda xabarni qabul qilish, kerakli marshrutni tanlashni belgilaydi.	 <pre> graph LR Start([]) --> Gateway{ } Gateway --> End1(()) Gateway --> End2(()) </pre>

	Voqealarning boshqa turlaridan ham foydalanish mumkin, masalan, Taymer hodisasi. Bunday holda, taklif qilingan marshrutlardan faqat bittasini tanlash mumkin. Xabarni qabul qilishning ikki yo‘li mavjud: Vazifalarni qabul qilish (yuqori o‘ngdagi rasmga qarang) va Oraliq xabar hodisalari (quyidagi o‘ngdagi rasmga qarang).	
Eksklyuziv bo‘lmagan shlyuz (shu jumladan)	Gatewayning bu turi Chiqish ish oqimida saqlangan shartli iboralarga asoslangan marshrutni tanlash bo‘lim nuqtasi. Qaysidir ma'noda, bu turdagi shlyuzlar o‘zaro bog‘langan mustaqil Ikkilik shlyuzlar guruhidir (Ha/Yo‘q). Chunki Agar marshrutlardan	<pre> graph LR subgraph OR_Gateway A[] --> B{ } B --> C[Condition 1] B --> D[Condition 2] end subgraph XOR_Gateway E[] --> F{ } F --> G[Condition 1] F --> H[Condition 2] end </pre>

	birortasi mustaqil bo'lsa, u holda marshrutlarning har qanday kombinatsiyasidan foydalanish mumkin (noldan maksimal marshrut kombinatsiyasi sonigacha). Biroq, diagrammalarni qurishda kamida bitta marshrut tanlanishi kerakligini hisobga olish kerak. Kamida bitta marshrut tanlanganligini tekshirish uchun standart shartdan foydalanish mumkin. Ushbu turdagi Gatewayning ikki turi mavjud. Birinchi tur shartli ish oqimlari to'plamidan foydalanadi. U diagrammada kichik olmoslar yordamida ta'kidlangan (yuqori o'ngdagi rasmga qarang).	
--	--	--

	Ikkinchi turdagi eksklyuziv bo‘lmagan shlyuzlardan foydalanadi (quyidagi o‘ngdagi rasmga qarang).	
Birlashish	"Birlashtirish" atamasi BPMN-da ikki yoki undan ortiq marshrutlarni bittaga (shuningdek, OR-Join deb ham ataladi) eksklyuziv birlashtirishga ishora qilish uchun ishlatiladi. Eksklyuziv Merge eshigi bir nechta oqimlarning birlashishini ko‘rsatish uchun mo‘ljallangan. Agar barcha kiruvchi tranzaksiya oqimlari muqobil bo‘lsa, u holda Gatewayga ehtiyoj qolmaydi. Bu nazoratsiz ish oqimi (quyidagi o‘ngdagi rasmga qarang) jarayonning borishiga bir xil	

	ta'sir ko'rsatishini anglatadi.	
Loop	BPMN-da jarayonda velosipedni ta'minlaydigan ikkita mexanizm mavjud.	Keyingi ikkitasini ko'ring raqamlar
Siklik harakat (Faoliyat sikli)	Vazifalar va quyi jarayonlarning atributlari ularning takrorlanishi yoki bir marta bajarilishini bildiradi. Looplarning ikki turi mavjud: standart va ko'p nusxali. Grafik jihatdan sikliklik rasmning pastki qismining markazida kichik marker sifatida ko'rsatiladi.	
Ish jarayonining siklligi (Oqimning ketma-ket aylanishi)	Looplar Ish oqimini "qarama-qarshi" obyektga biriktirish orqali paydo bo'lishi mumkin. Obyekt bir qator boshqa ish oqimlariga olib keladigan chiquvchi ish oqimiga ega bo'lsa,	

	obyekt qarama-qarshi hisoblanadi, ularning oxirgisi asl obyekt uchun kiruvchi ish oqimidir.	
Ko‘p misolli (Bir nechta misollar)	Vazifalar va quyi jarayonlarning atributlari ularning takrorlanishi yoki bir marta bajarilishini bildiradi. Rasmning pastki markazidagi uchta gorizontaal chiziq ketma-ket ko‘p misolni ko‘rsatadi (yuqoridagi o‘ngdagi rasmga qarang). Rasmning pastki markazidagi uchta vertikal chiziq parallel ko‘p misolni bildiradi (quyidagi o‘ngdagi rasmga qarang).	Sequential  Parallel 
Jarayondagi tanaffus (jarayonni to‘xtata oladigan va nazorat qilib bo‘lmaydigan narsa) (Jarayonning uzilishi)	Jarayonning uzilishi - bu jarayonning qaysi nuqtasida kutilgan kechikish sodir bo‘lishini ko‘rsatadigan bo‘lim. Jarayonning	

	haqiqiy borishini ko'rsatish uchun oraliq harakat qo'llaniladi (yuqori o'ngdagi rasmga qarang). Shuni ta'kidlash kerakki, jarayondagi tanaffus artefakti, model yoki simulyatsiya dasturini ishlab chiquvchining iltimosiga binoan, kechikishning oqim ichidagi joylashishini ta'kidlaydigan Voqea sifatida tasniflanishi mumkin.	
Tranzaksiya (Tranzaksiya)	Tranzaksiya - bu ma'lum bir protokol tomonidan qo'llab-quvvatlanadigan kichik jarayon bo'lib, barcha ishtirokchilar harakatni bajarish yoki rad etish kerakligiga rozi bo'lishini ta'minlaydi. Harakat grafiklari harakatning kelishuv yoki	

	yoʻqligini koʻrsatadi. Ikki chiziqli chegara ushbu quyi jarayonning tranzaksiya ekanligini koʻrsatadi.	
Ichki/Inline quyi jarayon (ichiga oʻrnatilgan/oʻrnatilgan pastki Jarayon (Inline blok)	Ichki (yoki ichki) quyi jarayon - bu asosiy jarayon bilan bir xil ma'lumotlar toʻplamiga ega boʻlgan faoliyat. Ushbu turdagi Sub-jarayon mustaqil quyi jarayonga qarama-qarshi boʻlib, undan qayta foydalanish va asosiy jarayon tomonidan havola qilinishi mumkin. Ish oqimidan foydalanganda ma'lumotlar ichki quyi jarayonga emas, balki asosiy quyi jarayonga uzatilishi kerak.	Diagrammada ushbu turdagi Subprocess hech qanday maxsus belgilarga ega emas
Guruh (bir xil toifadagi obyektlar guruhini oʻz ichiga olgan blok) (Guruh)	Guruh bir xil toifaga kiruvchi grafik elementlarni guruhlash uchun moʻljallangan.	

	Ushbu guruhlash ish jarayoniga ta'sir qilmaydi. Biznes jarayonlari diagrammasida guruhlangan elementlar tegishli bo'lgan toifa nomi guruh nomi sifatida ko'rinadi. Ushbu turdagi guruhlash hujjatlashtirish yoki tahlil qilish uchun ishlatilishi mumkin. Grafik jihatdan, Guruhlar obyekt toifalari kabi ko'rsatiladi.	
Sahifa ulagichi (Sahifadan tashqari ulagich)	Diagrammada ushbu grafik element Ish jarayoni oldingi sahifada tugaydigan joyda, keyin esa keyingi sahifada davom etadigan joyda paydo bo'ladi. Oraliq hodisa sahifa ulagichi sifatida ishlatilishi mumkin "Ulanish". Asosan chop etish uchun mo'ljallangan.	 

Uyushma	Assotsiatsiya ma'lumotlar yoki artefaktlar (oqim elementlari bilan bog'liq bo'lmagan obyektlar) va oqim elementlari o'rtasida aloqa o'rnatish uchun xizmat qiladi. Matn obyektlari, shuningdek, oqim elementlari bilan bog'liq bo'lmagan grafik obyektlarni oqim elementlari bilan bog'lash mumkin. Agar kerak bo'lsa, Assotsiatsiya oqim yo'nalishini ko'rsatishi mumkin (masalan, ma'lumotlar oqimi).	>
Matn izohi (Assotsiatsiya bilan bog'liq) (Matnga izoh)	Matn izohlari biznes-jarayon modeleriga BPMN diagrammalari bilan ishlaydiganlar uchun qo'shimcha ma'lumotlarni taqdim etish imkonini beruvchi mexanizmdir.	> Descriptive Text Here

Hovuz	Hovuz o‘zaro ta'sir ishtirokchisini ifodalaydi. Hovuz, shuningdek, Mas'uliyat sohasi yoki odatda biznesdan biznesga (B2B) vaziyatlarda topiladigan boshqa Hovuzlar bilan bog‘liq muayyan harakatlar to‘plamini ajratish uchun mas'ul bo‘lgan grafik konteyner sifatida ham harakat qilishi mumkin. Hovuzda ishlayotgan jarayon haqida qo‘shimcha ma'lumotlar bo‘lishi mumkin. Agar Hovuzda bunday ma'lumotlar bo‘lmasa, u "qora quti" bo‘lishi mumkin.	Name
Lane	Trek rollarning taqsimlanishini ko‘rsatish uchun ishlatiladi va vertikal yoki gorizontal bo‘lishi mumkin (shuningdek,	Name Name

	Hovuzning ichki maydonini ajratish uchun ham foydalanish mumkin). Harakatlarni tashkil qilish va tasniflash uchun xizmat qiladi.	
--	---	--

Nazorat savollari

1. BPMN notatsiyasi nima va uning asosiy maqsadi nima?
2. BPMN notatsiyasining asosiy elementlari qanday?
3. Oddiy aktivliklar (Tasks) subklasslari qanday turlarga bo‘linadi?
4. Kompleks aktivliklar (Sub-processes) qanday turlarga bo‘linadi?
5. Voqealar (Events) subklasslari qanday turlarga bo‘linadi?
6. Oraliq voqealar (Intermediate Events) subklasslarining qanday turlari mavjud?
7. Shlyuzlar (Gateways) qanday subklasslarga bo‘linadi va ularning vazifasi nima?
8. Ketma-ketlik oqimlari (Sequence Flows) qanday turlarga bo‘linadi?
9. Havolalar (Message Flows) va uyushmalar (Associations) qanday vazifalarni bajaradi?
10. Ma'lumotlar obyektlari (Data Objects) qanday subklasslarga bo‘linadi va ularning vazifasi nima?
11. Ishchi kuchi va rollar (Pools and Lanes) qanday ko‘rsatiladi va qanday vazifalarni bajaradi?
12. BPMN diagrammasida anotatsiyalar (Annotations) qanday qo‘llaniladi va ularning vazifasi nima?
13. BPMN diagrammasini yaratishda subklasslardan qanday foydalangan holda diagrammalarni yaratish qoidalari qanday?
14. BPMN subklasslarining afzalliklari va kamchiliklari qanday?

11-MAVZU. BIZNES JARAYONLAR DIAGRAMMALARNING KATEGORIYALARI

Reja:

1. Diagrammani modellashtirishda matn, rang va chiziqlardan foydalanish
2. Oqim elementlarini ulash qoidalari
3. Ish oqimlarini ulash qoidalari

BPMN 2.0 uchta asosiy jarayon modellarining tavsifini o‘z ichiga oladi: shaxsiy jarayon (*bajariladigan* va *bajarilmaydigan*) ommaviy **jarayon** va **xoreografiya**.

Yuqoridagi asosiy modellardan foydalanib, **biznes jarayonlari** diagrammalarining ko‘plab o‘zgarishlarini yaratish mumkin. Quyida **BPMN 2.0** yordamida ishlab chiqilgan biznes-jarayon submodellari keltirilgan:

- Yuqori darajadagi *bajarilmaydiganlar*
- Harakatlar (funktsional bo‘lmagan tahlil).
- Batafsil *bajariladigan* **biznes jarayoni**.
- Biznes-jarayon "Xuddi shunday" (eskirgan).
- "Bo‘lajak" biznes jarayoni (yangi).
- **Xoreografiya** (**Xoreografiya**). Jarayonning ikki yoki undan ortiq ishtirokchilaridan kutilgan xatti-harakatlarning tavsifi.
- Bir yoki bir nechta tashqi ishtirokchilar (qora quti **jarayoni**) o‘rtasidagi munosabatlarni o‘z ichiga olgan batafsil, *shaxsiy biznes jarayoni* (*bajariladigan* va *bajarilmaydigan*).
- Ikki yoki undan ko‘p batafsil *ishlaydigan* o‘zaro ta'sir qiluvchi **jarayonlar**.
- **Xoreografiya** bilan o‘zaro aloqada bo‘lgan batafsil, *bajariladigan* biznes jarayoni.
- Ikki yoki undan ortiq *ommaviy jarayonlar*.
- **Xoreografiya** bilan o‘zaro aloqada bo‘lgan ommaviy jarayon.
- **Xoreografiya** orqali o‘zaro ta'sir qiluvchi ikki yoki undan ortiq batafsil *bajariladigan* **biznes jarayonlari**.

Ushbu belgi yuqoridagi biznes-jarayonlarning misollarini tavsiflash uchun yaratilgan. Ammo shuni ta'kidlash kerakki, submodellarni birlashtirish uchun turli xil variantlarni yaratish biznes jarayonlarini modellashtirish vositalarini ishlab chiqaruvchilarga topshiriladi. **BPMN 2.0** dan foydalanilganda, biznes-jarayon modelerini tanlangan modellashtirish obyektiga, masalan, shaxsiy biznes jarayoni yoki xoreografiyaga yo‘naltirish tavsiya etiladi, garchi belgining o‘zi hech narsa yuklamaydi.

Diagrammani modellashtirishda matn, rang va chiziqlardan foydalanish

Obyektlarning matnli izohlari modeler tomonidan **jarayon** yoki diagrammada joylashgan obyektlarning atributlari haqida qo'shimcha ma'lumotlarni ko'rsatish uchun ishlatiladi.

Oqim elementlari va boshqa diagramma elementlari matn belgilariga ega bo'lishi mumkin (masalan, oqim nomi va/yoki uning boshqa atributlari nomlari). Matn yorliqlari shaklning ichiga, uning ustiga yoki ostiga joylashtirilishi mumkin. Matn belgilarining joylashuvi, shuningdek ularning yo'nalishi model ishlab chiqaruvchisi yoki modellashtirish dasturining g'oyalariga qarab har qanday bo'lishi mumkin. Grafik elementni to'ldirish oq yoki shaffof bo'lishi mumkin.

Oqim elementlarini ulash qoidalari

Kiruvchi ish oqimi Oqim elementining istalgan nuqtasiga (chap, o'ng, yuqori, pastki) biriktirilishi mumkin. Shunga o'xshab, chiquvchi ish oqimi oqim elementining istalgan nuqtasidan (chap, o'ng, yuqori, pastki) boshlanishi mumkin. Xabarlar oqimi ish oqimi bilan bir xil xususiyatlarga ega. **BPMN** tili sizning talablaringizni qondirish uchun moslashtirilishi mumkin, ammo model ishlab chiquvchilari Oqim elementlarini ulash uchun mavjud tajribalaridan foydalanishi KERAK, bu yaratilgan diagrammalarni tushunishni osonlashtiradi va tasvirlangan biznes jarayoni oqimini shaffof va tushunarli qiladi. Bu diagrammada **ish oqimi** yoki **xabar oqimi** kabi grafik elementlar mavjud bo'lsa, bu ayniqsa muhimdir. Bunday holda, eng yaxshi variant - chapdan o'ngga yoki yuqoridan pastga joylashgan Operatsion oqimining yo'nalishini tanlash va keyin 90 ° burchak ostida joylashgan bo'lishi kerak bo'lgan xabar oqimining yo'nalishini tanlash. allaqachon tanlangan **Operatsiya oqimiga** nisbatan. Yuqoridagi barcha talablar bajarilganda, foydalanish uchun qulay diagrammalar tuziladi.

Ish oqimlarini ulash qoidalari

11.1-jadvalda BPMN tili tomonidan qo'llaniladigan Oqim elementlarining tasvirlari mavjud va shuningdek, ushbu grafik elementlarning Work Flow orqali bir-biri bilan qanday bog'langanligi ko'rsatilgan. Ushbu qoidalar **jarayon** diagrammasi va **xoreografiya** diagrammasi uchun ham amal qiladi.  belgisi jadval satrlaridan birida ko'rsatilgan grafik elementni tegishli ustunda ko'rsatilgan grafik

elementga ulash mumkinligini bildiradi. Jadvalda turli xil konfiguratsiyalarga bog‘liq bo‘lgan grafik elementning kiruvchi va chiquvchi ulanishlari soni ko‘rsatilmagan. Keyingi bobda har bir alohida grafik elementni ulash qoidalari haqida batafsil ma'lumotlar mavjud. *Esda tutingki, agar quyi jarayon diagrammaning butun uzunligini egallasa, ushbu kichik jarayon ichida joylashgan grafik elementlarni uning chegaralaridan tashqarida joylashgan grafik elementlarga ulash mumkin emas. Xuddi shunday, Ish oqimi Hovuz chegaralarini kesib o‘ta olmaydi.*

11.1-jadval – Ish oqimini ulash qoidalari

Ot/K						
		↗	↗	↗	↗	↗
		↗	↗	↗	↗	↗
		↗	↗	↗	↗	↗
		↗	↗	↗	↗	↗
		↗	↗	↗	↗	↗
						

Eslatma: Jadvalda faqat kirish yoki chiquvchi ish oqimlari bo‘lgan grafik elementlar ko‘rsatilgan. Pool, Track, Data Object, Group va Text Annotation kabi obyektlar jadvalda mavjud emas. Harakat bu yerda **Jarayon** kontekstidagi Harakat va **Sub-jarayonni**, shuningdek **Xoreografiya** kontekstidagi **Xoreografiya** va **Sub-Xoreografiya** Harakatini anglatadi.

Xabar oqimlarini ulash qoidalari

11.2-jadvalda BPMN modellash tirish obyektlarining tasvirlari mavjud va shuningdek, ushbu obyektlar bir-biriga Xabar oqimi orqali qanday bog‘langanligi ko‘rsatilgan. Ushbu qoidalar O‘zaro ta'sir diagrammasi elementlariga ham tegishli. Belgisi ↗ jadval satrlaridan birida ko‘rsatilgan grafik elementni tegishli ustunda ko‘rsatilgan grafik elementga ulash mumkinligini bildiradi. Jadvalda turli xil konfiguratsiyalarga bog‘liq bo‘lgan grafik elementning kiruvchi va chiquvchi ulanishlari soni ko‘rsatilmagan. Keyingi bobda har bir

alohida grafik elementni ulash qoidalari haqida batafsil ma'lumotlar mavjud. Esda tutingki, xabarlar oqimi bir xil hovuz ichida joylashgan obyektlarga ulanmaydi.

11.2-jadval – Xabarlar oqimini ulash qoidalari

Ut/R		Pool		  
				
Pool				 
				 
  		  	  	     

Eslatma: Jadvalda faqat kiruvchi yoki chiquvchi xabar oqimlari bo'lgan grafik elementlar ko'rsatilgan. Track, Gateway, Data Object, Group va Text Annotation kabi obyektlar jadvalda mavjud emas.

BPMN kengaytirilishi

BPMN 2.0 standart grafik diagramma elementlari uchun atributlar ro'yxatini kengaytirish imkonini beruvchi mexanizmni tavsiflaydi. Agar kerak bo'lsa, standart bo'lmagan grafik element atributlari yoki artefaktlar, masalan, vertikal mintaqa uchun noyob talablar modeler yoki modellashtirish dasturiga kiritilishi mumkin. **BPMN** da tasvirlangan mantiqni buzmaslik uchun bunday atributlar **BPMN** grafik elementlaridan foydalanish semantikasiga zid bo'lmazligi kerak. Shuni ta'kidlash kerakki, yangi atributlarni qo'shish imkoniyatiga qaramay, har qanday darajadagi ta'lim foydalanuvchisi tomonidan yaxshiroq idrok etilishi uchun diagrammaning barcha asosiy tamoyillari va ravshanligi saqlanishi kerak. Esda tutingki, asosiy oqim elementlarining shakllari (hodisalar, harakatlar va shlyuzlar) o'zgartirilmasligi kerak.

Ushbu spetsifikatsiya zarur va ixtiyoriy elementlar o'rtasidagi farqni ko'rsatadi (kengaytmalarni yaratish sintaksisi tavsifi uchun

8.3.2-bo'limga qarang). Majburiy elementlardan foydalanilsa, diagrammalarni amalga oshirishda ularni hisobga olish kerak. Agar qo'shimcha elementlar ishlatilsa, diagrammalarni amalga oshirishda ular o'tkazib yuborilishi mumkin.

Nazorat savollari

1. Biznes jarayonlar diagrammalari nima uchun foydalaniladi?
2. Biznes jarayonlar diagrammalari qanday maqsadlar uchun ishlatiladi?
3. Biznes jarayonlar diagrammalari qaysi asosiy kategoriyalarga bo'linadi?
4. Tashkilotlar arasida ishlab chiqarish jarayonlarini modellovchi diagrammalarning afzalliklari qanday?
5. Biznes jarayonlarining harakatlarini va jarayonlarni ifodalovchi diagrammalarning muhim elementlari nimalardir?
6. Biznes jarayonlar diagrammalari qanday turli asosiy elementlardan foydalanadi?
7. BPMN (Business Process Model and Notation) va UML (Unified Modeling Language) diagrammalari o'rtasidagi asosiy farqlar nima?
8. Biznes jarayonlar diagrammalari tarkibidagi voqealar va shlyuzlar qanday vazifalarni bajaradi?
9. Biznes jarayonlar diagrammalari qanday turdagi ma'lumotlar bilan to'ldiriladi?
10. Biznes jarayonlar diagrammalari yaratish jarayonida qaysi dasturiy vositalar va platformalardan foydalaniladi?

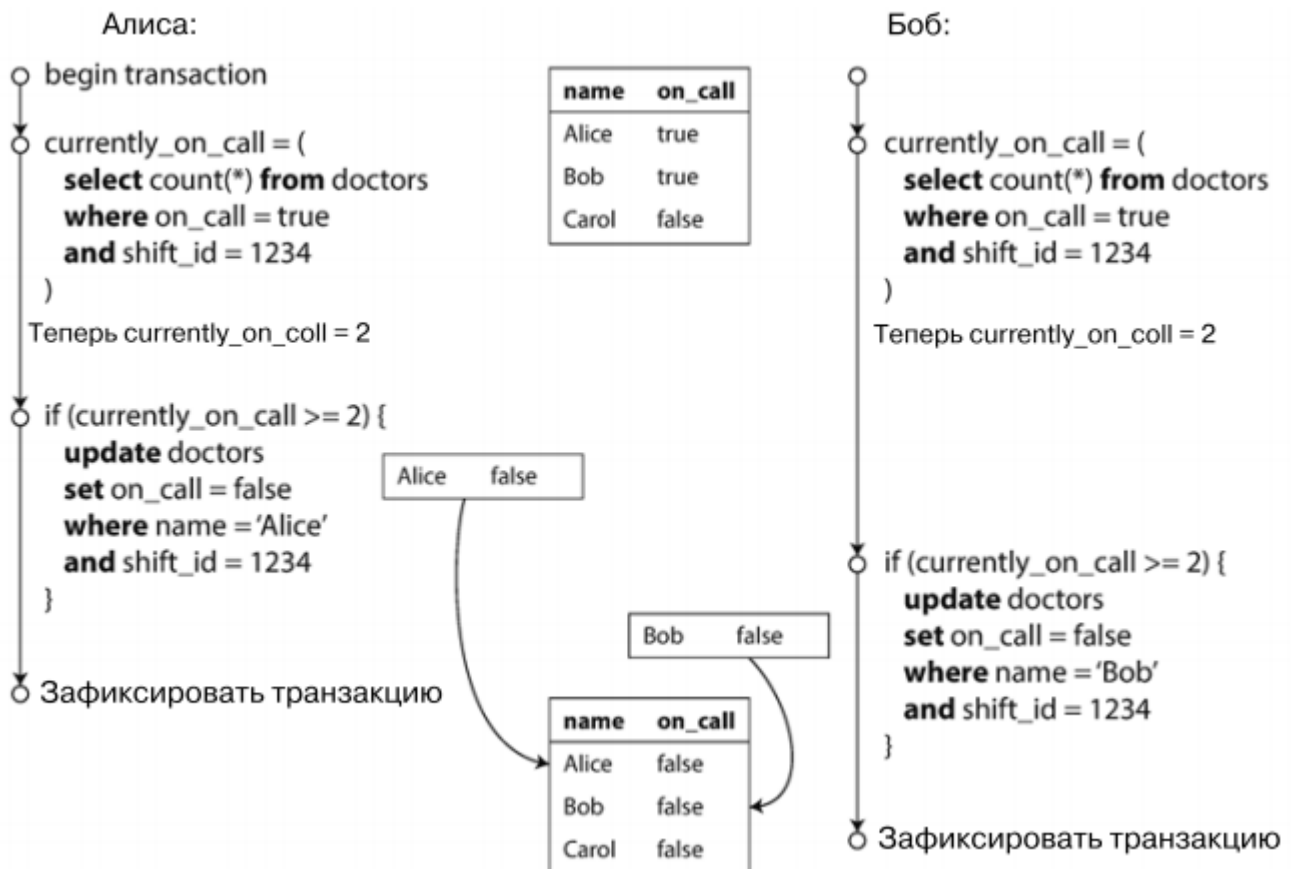
12-MAVZU. OPERATSIYALAR

Reja:

1. Seriyalash imkoniyati
2. Ikki fazali blokirovkalash (2PL)
3. Serializatsiya qilinadigan oniy tasvir izolyatsiyasi (SSI)

Tasavvur qilaylik, biz shifokorlar uchun shifoxonada navbatchilikni tashkil qilish uchun mo'ljallangan dastur yaratmoqdamiz. Kasalxonalar odatda har qanday vaqtda bir nechta shifokorlarni navbatchilik qilishga harakat qilishadi, lekin, albatta, bittadan kam emas. Shifokorlar, agar ularning hamkasblaridan kamida bittasi xizmatda qolsa, xizmat vazifasini bajarishdan bosh tortishi mumkin.

Endi tasavvur qiling-a, Elis va Bob ma'lum bir smenada navbatchi ikki shifokor. Ikkalasi ham o'zlarini yomon his qilishdi va ikkalasi ham dam olishni so'rashga qaror qilishdi. Afsuski, ular bir vaqtning o'zida dam olish so'rovini bosishdi. Shundan keyin nima sodir bo'lishi rasmda ko'rsatilgan. 1.



12.1-rasm - Ilova kodidagi xatolik tufayli qayd etilgan assimetriyaga misol

Har bir tranzaksiyada ariza birinchi navbatda hozirda ikki yoki undan ortiq shifokorlar bor-yo'qligini tekshiradi. Natija ijobiy bo'lsa, shifokorlardan biriga dam olish vaqtini berish xavfsiz deb hisoblaydi. Ma'lumotlar bazasi oniy tasvir izolyatsiyasidan foydalanganligi sababli, ikkala tekshiruv ham 2 ni qaytaradi va ikkala tranzaksiya ham bajarishda davom etadi. Elis va Bob o'zlarining jadvallarini muvaffaqiyatli yangilaydilar va dam olishadi. Ikkala operatsiya ham qayd etilgan, shundan so'ng bitta navbatchi shifokor yo'qligi ma'lum bo'ldi. Kamida bitta shifokorning navbatchilik qilishi haqidagi talab buzilgan.

Ushbu anomaliya yozish egriligi deb ataladi. Bu iflos yozish operatsiyasi emas va yo'qolgan yangilanish ham emas, chunki bizning ikkita tranzaksiyalarimiz ikki xil obyektни yangilaydi. Bu erda mojaroning mavjudligi unchalik sezilmaydi, lekin bu, albatta, poyga sharti: agar ikkita tranzaksiya birin-ketin amalga oshirilgan bo'lsa, ikkinchi shifokor dam olish vaqtini olmaydi.

Yozish assimetriyasi yangilanishni yo'qotish muammosini umumlashtirish sifatida ko'rib chiqilishi mumkin. Ushbu assimetriya ikkita tranzaksiyalar bir xil obyektlarni o'qib, keyin ularning ba'zilarini yangilaganda paydo bo'lishi mumkin (turli tranzaksiyalar turli obyektlarni yangilashi mumkin). Turli xil tranzaksiyalar bir xil obyektни yangilaganida, "iflos" yozish operatsiyasi sodir bo'ladi yoki yangilanish yo'qoladi (vaqtga qarab).

Agar serializatsiya qatlamidan foydalanishning iloji bo'lmasa, ikkinchi eng yaxshi yechim tranzaksiya uchun zarur bo'lgan qatorlarni aniq blokirovka qilishdir. Shifokorlar bilan misolda siz taxminan quyidagi so'rovlarni yozishingiz mumkin:

```
BEGIN TRANSACTION;  
SELECT * FROM doctors  
WHERE on_call = true  
AND shift_id = 1234 FOR UPDATE;  
UPDATE doctors  
SET on_call = false  
WHERE name = 'Alice'  
AND shift_id = 1234;  
COMMIT;
```

FOR UPDATE bandi ma'lumotlar bazasiga ushbu so'rov orqali qaytarilgan barcha qatorlarga blokirovka qo'yish kerakligini aytadi.

Chaqiruv misolida, o'zgartirilgan qator birinchi so'rovda qaytarilgan qatorlardan biri edi, shuning uchun tranzaktsiyani himoya qilish va satrlarni blokirovka qilish orqali yozuvning burilishiga yo'l qo'ymaslik mumkin edi (SELECT FOR UPDATE). Biroq, agar ma'lum bir qidiruv shartini qondiradigan qatorlar yo'qligiga ishonch hosil qilish uchun tekshirish amalga oshirilsa va yozish operatsiyasi bir xil shartga mos keladigan qatorni qo'shsa, ma'lumotlarni olish uchun birinchi so'rov hech narsani qaytarmaydi va YANGILANISH UCHUN SELECT ni tanlang. qulf o'rnatish uchun hech narsa bo'lmaydi.

Bitta tranzaksiyadagi yozish operatsiyasi boshqasida qidiruv so'rovi natijasini o'zgartiradigan bu effekt fantom deb ataladi.

Bunday holatlarning oldini olish uchun ko'p hollarda ketma-ketlik darajasi izolyatsiyasidan foydalanish afzalroqdir.

Seriyalash imkoniyati

Serializatsiya odatda eng kuchli izolyatsiya darajasi hisoblanadi. U tranzaktsiyalar bir vaqtning o'zida amalga oshirilgan taqdirda ham, natija ketma-ket bajarilgandek, hech qanday parallelliksiz qolishini kafolatlaydi. Shuning uchun, baza barcha mumkin bo'lgan poyga sharoitlarini oldini oladi.

Ammo agar seriyali bo'lish zaifroq izolyatsiya darajalariga qaraganda ancha yaxshi bo'lsa, nega uni hamma joyda ishlatmaslik kerak? Bu savolga javob berish uchun keling, serializatsiyaning mumkin bo'lgan ilovalarini va ularning ishlashini ko'rib chiqaylik. Serializatsiyani ta'minlaydigan ko'pgina zamonaviy ma'lumotlar bazalari uchta usuldan birini qo'llaydi:

- tranzaktsiyalarning chinakam ketma-ket bajarilishi;
- ikki fazali blokirovka qilish;
- Serializatsiya qilinadigan oniy tasvir izolyatsiyasi (SSI) kabi optimistik parallellikni boshqarish usullari.

Biz ushbu usullarni bir tugunli ma'lumotlar bazalari bilan bog'liq holda ko'rib chiqamiz. Biroq, kitobda ularning taqsimlangan tizimlardagi imkoniyatlari alohida bobda muhokama qilinadi.

Ketma-ket bajarish. Paradoksallik bilan bog'liq muammolardan qochishning eng oddiy usuli - bu parallellikdan butunlay qochishdir: bir vaqtning o'zida faqat bitta tranzaktsiyani ketma-ket bajaring.

Bu g'oya juda ravshan bo'lsa-da, ma'lumotlar bazasi dizaynerlari bir torli tranzaksiya sikli foydali deb qaror qilishdi. Qanday qilib bitta ipli bajarish maqbul bo'ldi?

Kontsepsiyani bunday qayta ko'rib chiqishga ikki omil sabab bo'ldi.

- RAM yetarlicha arzonlashdiki, ko'p hollarda operativ xotirada barcha faol ma'lumotlar to'plamini saqlash mumkin. Agar barcha kerakli ma'lumotlar RAMda bo'lsa va qattiq diskdan yuklanishini kutishning hojati bo'lmasa, tranzaktsiyalar ancha tezroq yakunlanadi.

- Ma'lumotlar bazasi dizaynerlari OLTP tranzaktsiyalari qisqa bo'lishini va faqat oz sonli yozish va o'qishni amalga oshirishini tushunishdi. Aksincha, uzoq davom etadigan analitik so'rovlar odatda faqat ma'lumotni o'qiydi, shuning uchun ular ketma-ket bajarilish siklidan tashqarida izchil suratda (snapshot izolyatsiyasi darajasida) bajarilishi mumkin.

Tranzaktsiyalarni ketma-ket bajarish ba'zi cheklovlar bilan oniy tasvirni izolyatsiyasiga erishishning istiqbolli usuli sifatida paydo bo'ldi.

- Barcha operatsiyalar kichik va tez bo'lishi kerak, chunki bitta sekin tranzaksiya butun operatsiyani to'xtatib qo'yishi mumkin.

- Barcha ma'lumotlar to'plami xotiraga sig'ishi kerak. Siz kamdan-kam qo'llaniladigan ma'lumotlarni diskka ko'chirishingiz mumkin, ammo agar siz unga bitta tarmoqli tranzaksiyada kirishingiz kerak bo'lsa, tizim juda sekinlashishi mumkin.

- Bir torli tizimlar ba'zan bir vaqtning o'zida ishlaydigan tizimlarga qaraganda tezroq ishlaydi, chunki ular yuqori blokirovka qilishni talab qilmaydi. Biroq, ularning o'tkazish qobiliyati bitta protsessor yadrosining o'tkazuvchanligi bilan cheklangan.

Ikki fazali blokirovkalash (2PL). Uzoq vaqt davomida ma'lumotlar bazalarida faqat bitta serializatsiya algoritmi keng qo'llanilgan: ikki fazali qulflash (2PL).

E'tibor bering, ikki fazali qulflash (2PL) nomi ikki fazali tranzaksiya qilish (2PC) nomiga juda o'xshash bo'lsa-da, ular ikkita butunlay boshqa narsadir.

Ikki fazali blokirovkalash odatiy holga o'xshaydi, lekin talablarni sezilarli darajada oshiradi. Bitta obyektни bir vaqtning o'zida bir nechta bitimlar bo'yicha o'qishga, agar unga hech kim yozmasa, ruxsat beriladi. Ammo ushbu obyektда yozish operatsiyasini bajarish uchun eksklyuziv kirish talab qilinadi.

Ikki fazali blokirovka bilan yozish operatsiyalari nafaqat boshqa yozish operatsiyalarini, balki o'qish operatsiyalarini ham bloklaydi va aksincha. Suratni izolyatsiya qilish qoidasi - o'qish tranzaktsiyalari

hech qachon yozuvchilarni bloklamaydi va yozuvchilar hech qachon o'quvchilarni bloklamaydi - oniy tasvirni izolyatsiya qilish va ikki fazali blokirovka o'rtasidagi asosiy farqni ko'rsatadi. Boshqa tomondan, ikki fazali qulflash ketma-ketlikni ta'minlash bilan birga, har qanday poyga sharoitidan himoya qiladi.

Qulflar juda ko'p bo'lganligi sababli, A tranzaksiyasi B tranzaksiyasi o'z qulfini ochishini kutishi odatiy holdir va aksincha. Bu holat o'lik deb ataladi. Ma'lumotlar bazasi tranzaksiyalar o'rtasidagi to'siqlarni avtomatik ravishda aniqlaydi va ulardan birini bekor qiladi, shunda qolganlari davom etishi mumkin.

Ikki fazali blokirovkaning asosiy kamchiligi va uning keng qo'llanilmasligining sababi uning yomon ishlashidir: tranzaksiya o'tkazuvchanligi va so'rovlarga javob berish muddati zaif izolyatsiyaga qaraganda ikki fazali blokirovka bilan sezilarli darajada yomonroq.

An'anaviy relyatsion ma'lumotlar bazalari tranzaksiya vaqtlarini cheklamaydi, chunki ular foydalanuvchi kiritishini kutadigan interaktiv ilovalar uchun mo'ljallangan. Shuning uchun, bir tranzaksiya boshqasini cheksiz kutishi mumkin. Agar siz barcha tranzaksiyalarinigizni iloji boricha qisqaroq qilishga harakat qilsangiz ham, bir nechta tranzaksiyalar bir xil obyektga kirishi kerak bo'lganda, navbat paydo bo'lishi mumkin, chunki bitta tranzaksiya biror narsa qilishdan oldin bir necha boshqasi bajarilishini kutishi kerak.

Tizimning qolgan qismi to'liq to'xtab qolishi uchun bitta sekin tranzaksiya yoki ko'p ma'lumotlarga kirish va ko'plab qulflarni olish uchun bitta tranzaksiya kifoya qiladi. Ishonchli ishlash talab qilinadigan ilovalarda bunday beqarorlik istalmagan.

Serializatsiya qilinadigan oniy tasvir izolyatsiyasi (SSI). Ushbu xulosada biz ma'lumotlar bazasiga bir vaqtda kirishni boshqarishning juda g'amgin rasmini ko'rdik. Bir tomondan, mavjud ketma-ketlashtirish ilovalari yomon ishlashni ko'rsatadi (ikki fazali qulflash) yoki yaxshi miqyosda emas (ketma-ket bajarish). Boshqa tomondan, zaif izolyatsiya darajalari yuqori ish faoliyatini ko'rsatadi, lekin turli xil poyga sharoitlariga (yangilanishlarni yo'qotish, assimetriya yozish, fantomlar va boshqalar) sezgir. Shunday qilib, seriyali izolyatsiya va yaxshi ishlash bir-birini mutlaqo istisno qiladimi?

Serializable snapshot isolation (SSI) deb nomlangan algoritm juda istiqbolli ko‘rinadi. Bu an'anaviy suratni izolyatsiya qilish bilan solishtirganda, unchalik katta bo‘lmagan ish haqi evaziga to‘liq serializatsiyani ta'minlaydi. SSI nisbatan yangi usul: u birinchi marta 2008 yilda tasvirlangan.

Ikki fazali blokirovka - bu pessimistik parallellikni boshqarish mexanizmi deb ataladigan narsa: u quyidagi printsipga asoslanadi: agar biror narsa noto‘g‘ri bo‘lishi mumkin bo‘lsa, biron bir chora ko‘rishdan oldin vaziyat normallashtirishini kutish yaxshiroqdir. Ketma-ket bajarish qaysidir ma'noda o‘ta pessimistikdir: u tranzaksiyaning butun muddati davomida butun ma'lumotlar bazasida (yoki butun ma'lumotlar bazasi bo‘limida) eksklyuziv qulfga ega bo‘lgan har bir tranzaksiyaga mos keladi.

Bundan farqli o‘laroq, ketma-ketlashtiriladigan oniy tasvirni izolyatsiya qilish moslashuvni boshqarishning optimistik usuli hisoblanadi. Bu kontekstda "optimistik" potentsial xavfli faoliyat sodir bo‘lganda blokirovka qilish o‘rniga, tranzaksiyalar hamma narsa yaxshi bo‘ladi degan umidda amalga oshirishda davom etishini anglatadi. Tranzaksiya amalga oshirilganda, ma'lumotlar bazasi yomon narsa sodir bo‘lganligini (masalan, izolyatsiya buzilganligini) tekshiradi. Agar shunday bo‘lsa, tranzaksiya to‘xtatiladi va yana takrorlanishi kerak.

Agar tizimning quvvat zaxirasi etarlicha katta bo‘lsa va tranzaksiyalar o‘rtasidagi qarama-qarshilik unchalik katta bo‘lmasa, u holda optimistik parallellik nazorati usullari pessimistiklarga qaraganda yuqori samaradorlikni namoyish etadi.

SSI snapshot izolyatsiyasiga asoslangan, ya'ni tranzaksiya ichidagi barcha o‘qishlar ma'lumotlar bazasining izchil surati asosida amalga oshiriladi. Bu optimistik parallellikni boshqarishning oldingi usullari o‘rtasidagi asosiy farq. Yozuvlar orasidagi ketma-ketlashtirish ziddiyatlarini aniqlash va qaysi tranzaksiyalarni bekor qilishni hal qilish uchun SSI oniy tasvir izolyatsiyasi ustiga algoritm qo‘shadi.

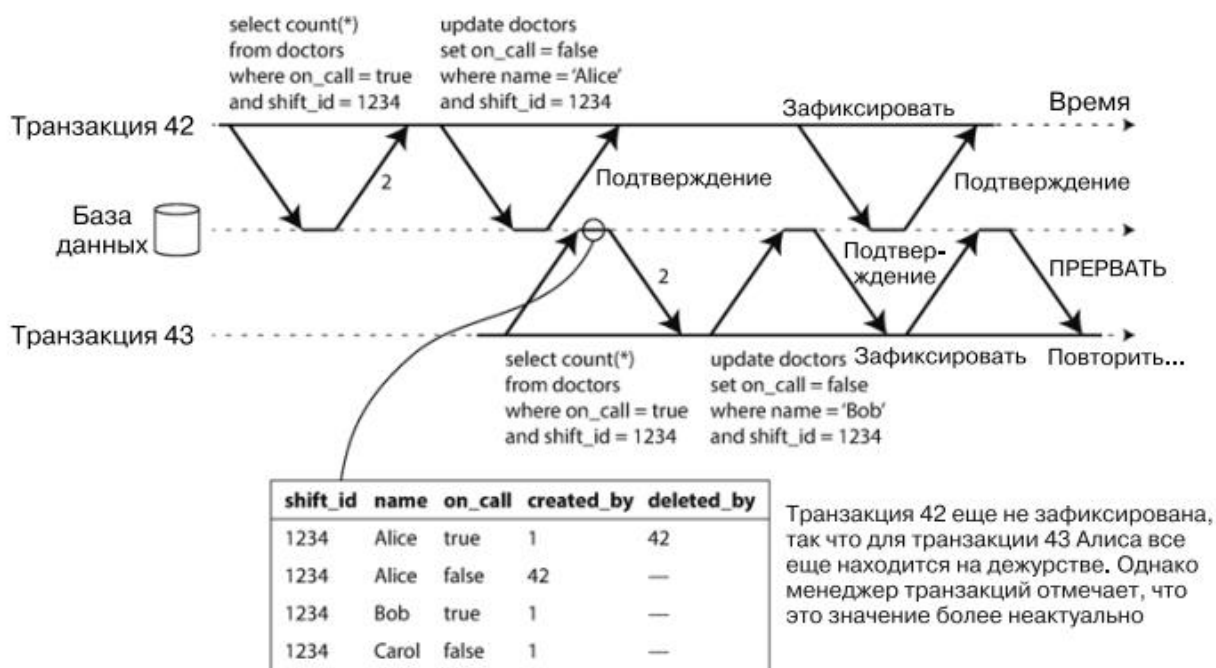
Snapshot izolyatsiyasida yozish assimetriyasi haqida yuqoridagi muhokamamizda biz muntazam ravishda bir xil naqshni kuzatdik: tranzaksiya ma'lumotlar bazasidan ma'lumotlarni o'qiydi, so'rov natijalarini tekshiradi va o'qigan ma'lumotlarga asoslanib harakat qilishga qaror qiladi. Biroq, oniy suratlarni ajratishda, dastlabki so'rov natijasida olingan ma'lumotlar tranzaksiya amalga oshirilgan paytda joriy bo'lmashligi mumkin, chunki u oraliqda o'zgartirilgan.

Ma'lumotlar bazasi ilova unga qarshi qilingan so'rovlar natijalaridan qanday foydalanishini bilmaydi. Xavfsizlik nuqtai nazaridan, so'rov natijalariga kiritilgan har qanday o'zgarishlar joriy tranzaksiyaning yozish operatsiyalari noto'g'ri bo'lishi mumkinligini anglatadi. Boshqacha qilib aytganda, so'rovlar va tranzaksiya orqali amalga oshiriladigan yozish operatsiyalari o'rtasida sabab-natija aloqasi bo'lishi mumkin. Seriyalash imkoniyatini ta'minlash uchun ma'lumotlar bazasi tranzaksiya eskirgan dastlabki shartlar asosida harakatlarni amalga oshirgan bo'lishi mumkin bo'lgan vaziyatlarni aniqlashi va bunday operatsiyalarni to'xtatishi kerak. Keling, ikkita holatni ko'rib chiqaylik:

- MVCC obyektlarining eskirgan versiyalarini o'qish operatsiyalarini aniqlash (o'qishdan oldin yozilmagan yozish operatsiyasi sodir bo'lgan);
- oldingi o'qish operatsiyalariga ta'sir qiluvchi yozish operatsiyalarini aniqlash (yozish operatsiyasi o'qishdan keyin sodir bo'lgan).

MVCC obyektlarining eskirgan versiyalaridan o'qishlarni aniqlash. Snapshot izolyatsiyasi odatda MVCC yordamida amalga oshiriladi. MVCC ma'lumotlar bazasidagi izchil suratdan o'qiladigan tranzaksiya suratga olinganda hali amalga oshirilmagan tranzaksiyalar tomonidan bajarilgan har qanday yozuvlarni e'tiborsiz qoldiradi. Shaklda. 2 tranzaksiya 43 Elisning navbatchi ekanligini ko'radi (on_call = rost), chunki tranzaksiya 42 amalga oshirilmagan. Biroq, 43 tranzaksiya amalga oshirilganda, 42 tranzaksiya allaqachon amalga oshirilgan. Bu shuni anglatadiki, izchil suratdan o'qishda e'tiborga

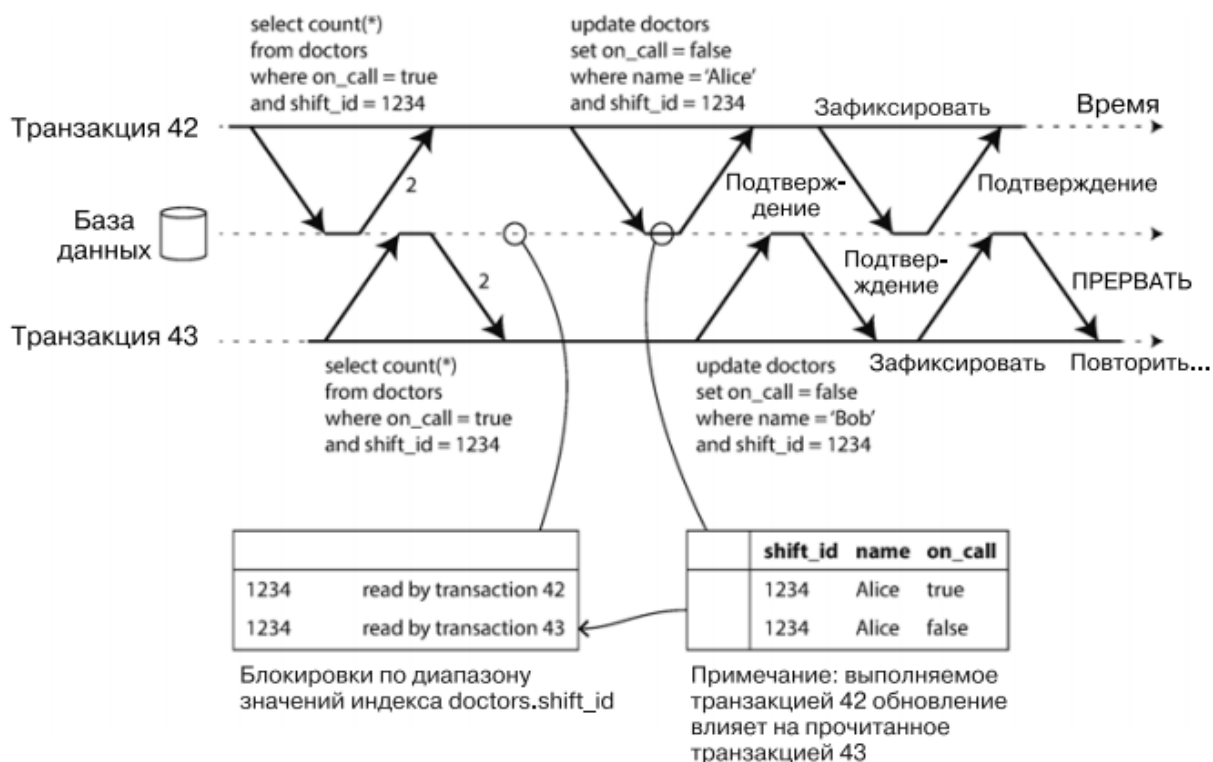
olinmagan yozish operatsiyasi endi kuchga kirdi va tranzaksiya 43ning asl shartlari endi haqiqiy emas.



12.2-rasm - MVCC snapshotidan eski qiymatlarni o'qiydigan tranzaksiyani aniqlash

Ushbu anomaliyaning oldini olish uchun ma'lumotlar bazasi MVCC ko'rinish qoidalariga ko'ra tranzaksiya boshqa tranzaksiyalarning yozish operatsiyalarini e'tiborsiz qoldiradigan holatlarni kuzatishi kerak. Tranzaksiyani amalga oshirishga urinayotganda, ma'lumotlar bazasi e'tiborsiz qoldirilgan yozishmalarning bajarilganligini tekshiradi. Ha bo'lsa, tranzaksiyani bekor qilish kerak.

Oldingi o'qish operatsiyalariga ta'sir qiladigan yozish operatsiyalarini aniqlang. Ikkinchi holat, boshqa tranzaksiya ma'lumotlarni o'qib bo'lgandan keyin o'zgartirganda.



12.3-rasm - Snapshotlarni ketma-ket izolyatsiyalashda, boshqa tranzaksiya tomonidan o'qilgan ma'lumotlarning tranzaksiyasi bilan o'zgartirish holatlarini aniqlash

12.3-rasmda, har ikkala tranzaksiya 42 va 43 smenada navbatchi 1234 shifokorni qidiradi. Shift_id-da indeks mavjud bo'lsa, ma'lumotlar bazasi o'zining 1234 yozuvidan 42 va 43 tranzaktsiyalari ushbu ma'lumotlarni o'qiganligini belgilash uchun foydalanishi mumkin (agar indeks bo'lmasa, ma'lumotni jadval darajasida kuzatish mumkin). Ma'lumotlarning o'zi faqat vaqtinchalik talab qilinadi: tranzaksiya tugagandan so'ng (tugallangan yoki bekor qilingan) va barcha bir vaqtning o'zida amalga oshirilgan tranzaktsiyalar tugagandan so'ng, ma'lumotlar bazasi ushbu tranzaksiya tomonidan qanday ma'lumotlar o'qilganligini osongina "unutishi" mumkin.

Ma'lumotlar bazasiga ma'lumotlarni yozadigan tranzaksiya tegishli ma'lumotlarni yaqinda o'qigan boshqa barcha operatsiyalar indekslarini qidirishi kerak. Bu jarayon mos keladigan kalit diapazoniga blokirovka qo'yishga o'xshaydi, lekin ma'lumotlarni o'qiydigan tranzaktsiyalar amalga oshirilgunga qadar blokirovka qilish o'rniga, qulf "tuzoq" bo'lib xizmat qiladi - tranzaktsiyalarga ular o'qigan ma'lumotlar tegishli bo'lmasligi mumkinligi haqida xabar beradi.

12.3-rasmda , tranzaksiya 42 oldingi o‘qish operatsiyasida o‘qigan ma'lumotlar eskirganligi va aksincha, 42- tranzaksiyaga xabar beradi. 42-tranzaksiya birinchi bo‘lib amalga oshiriladi va u muvaffaqiyatli bo‘ladi: 43-tranzaksiyaning yozish operatsiyasi 42-ga ta'sir qilsa-da, 43-tranzaksiya hali amalga oshirilmagan, shuning uchun yozish operatsiyasi hali kuchga kirmagan. Biroq, tranzaksiya 43 amalga oshirilganda, 42 tranzaksiyaning ziddiyatli yozish operatsiyasi allaqachon amalga oshirilgan, shuning uchun 43-tranzaksiyani bekor qilish kerak bo‘ladi.

Xulosa. Tranzaksiyalar abstraksiya qatlamini ta'minlaydi, bu ilovaga ba'zi parallellik muammolari va ba'zi turdagi apparat va dasturiy ta'minot nosozliklarini e'tiborsiz qoldirish imkonini beradi. Xatolarning keng toifasi oddiy tranzaksiyani to‘xtatishga olib keladi va dastur shunchaki tranzaksiyani qayta urinib ko‘rishi kerak.

Tranzaksiyalarsiz turli xil xato stsenariylari (jarayonlarning nosozliklari, tarmoqdagi uzilishlar, elektr uzilishlari, disk to‘laligi, kutilmagan parallellik va boshqalar) ma'lumotlarning nomuvofiqligiga olib kelishi mumkin.

Ushbu eslatmada biz raqobatbardosh kirish masalasini ko‘rib chiqdik. Bir nechta tez-tez ishlatiladigan izolyatsiya darajalari, jumladan o‘qilgan ma'lumotlar, oniy tasvirni izolyatsiya qilish va ketma-ketlashtirish mumkinligi muhokama qilindi. Biz poyga sharoitlarining turli misollaridan foydalanib, ushbu izolyatsiya darajalarining xususiyatlarini o‘rgandik.

- "Nopok" o‘qish operatsiyalari. Mijoz boshqa mijoz tomonidan yozilgan ma'lumotlarni qabul qilishdan oldin o‘qiydi. Qabul qilingan ma'lumotlarni o‘qish izolyatsiya darajalari va kuchliroqlari iflos o‘qishni oldini oladi.

- "Nopok" yozish operatsiyalari. Mijoz boshqa mijoz yozgan, lekin hali bajarmagan ma'lumotlarni qayta yozmoqda. Deyarli barcha tranzaksiya dasturlari iflos yozishni oldini oladi.

- O‘qish assimetriyasi (qayta ishlab chiqarilmaydigan o‘qish). Mijoz ma'lumotlar bazasining turli qismlarini vaqtning turli nuqtalarida ko‘radi. Ushbu muammo ko‘pincha oniy tasvirni izolyatsiya qilish orqali oldini oladi, bunda tranzaksiya doimiy nuqtadagi oniy suratdan ma'lumotlarni o‘qiydi. Bunga odatda Multi-Version Contention Control (MVCC) orqali erishiladi.

- Yo‘qolgan yangilanishlar. Ikki mijoz bir vaqtning o‘zida o‘qish-o‘zgartirish-yozish siklini amalga oshiradi. Ulardan biri o‘zi kiritgan o‘zgarishlarni hisobga olmagan holda ikkinchisi tomonidan yozilgan ma'lumotlarni qayta yozadi, shuning uchun ma'lumotlar yo‘qoladi. Ba'zi suratlarni izolyatsiyalash ilovalari bu anomaliyani avtomatik ravishda oldini oladi, boshqalari esa sizdan qulfni qo‘lda o‘rnatishingizni talab qiladi (YANGILASH UCHUN SELECT).

- Asimmetriyani qayd etish. Tranzaktsiya ba'zi ma'lumotlarni o‘qiydi, o‘qish qiymatiga qarab keyingi harakatlar to‘g‘risida qaror qabul qiladi va ma'lumotlar bazasiga yozish operatsiyasini bajaradi. Biroq, uni amalga oshirish vaqtida qaror qabul qilingan dastlabki shartlar endi haqiqatga mos kelmaydi. Faqat serializatsiya bu anomaliyaning oldini oladi.

- Fantom o‘qishlari. Tranzaktsiya muayyan qidiruv shartiga mos keladigan obyektlarni o‘qiydi. Boshqa mijoz ushbu qidiruv natijalariga qandaydir tarzda ta'sir qiladigan yozish operatsiyasini bajaradi. Snapshot izolyatsiyasi fantomlarni to‘g‘ridan-to‘g‘ri o‘qishni oldini oladi, lekin yozish assimetriyasi kontekstidagi fantomlar indeks qiymati oraliq‘ini blokirovka qilish kabi alohida ishlov berishni talab qiladi.

Zaif izolyatsiya darajalari ushbu anomaliyalarning ba'zilaridan himoya qiladi, ammo boshqalari qo‘lda ishlov berishga to‘g‘ri keladi (masalan, aniq blokirovkadan foydalanish). Faqat seriyali tranzaksiya darajasidagi izolyatsiya bunday muammolarni oldini oladi. Biz ketma-ketlashtirilgan tranzaktsiyalarni amalga oshirishning uch xil yondashuvini ko‘rib chiqdik.

- Haqiqatan ham izchil tranzaksiya bajarilishi. Agar siz individual tranzaktsiyalarni juda tez amalga oshirishingiz mumkin bo‘lsa va bir protsessor yadrosida vaqt birligida qayta ishlangan tranzaktsiyalar soni etarlicha kichik bo‘lsa, unda bu variant ishlov berish uchun oddiy va samarali bo‘ladi.

- Ikki fazali blokirovka. Bu o‘nlab yillar davomida ketma-ketlashtirishga erishishning standart usuli bo‘lib kelgan, ammo ko‘pgina ilovalar yomon ishlashi tufayli uni ishlatishdan qochishadi.

- Serializatsiya qilinadigan oniy tasvir izolyatsiyasi (SSI). Oldingi yondashuvlarning deyarli barcha kamchiliklaridan mahrum bo‘lgan juda yangi algoritm. Tranzaktsiyalar blokklanmasdan yakunlanishi uchun optimistik yondashuvdan foydalanadi. Tranzaktsiyani amalga

oshirishdan oldin tekshiruv o'tkaziladi va agar bajarish ketma-ketlikda bo'lmasa, bitim tuzilmasdan to'xtatiladi.

Nazorat savollari

1. Operatsiyalar nima? Ularning amaliyati va maqsadi nima?
2. Operatsiyalar modellashtirishda qanday vositalar va metodologiyalar foydalaniladi?
3. Operatsiyalarning turli turlari va ularning asosiy farqlari nima?
4. Operatsiyalar modellashtirishda UML (Unified Modeling Language) ning ro'li qanday?
5. Operatsiyalar modellashtirishning boshqa metodologiyalari qanday?
6. Operatsiyalar modellashtirish jarayonida qanday amaliyotlar bajariladi?
7. Operatsiyalar modellashtirishda qaysi muammolar uchun diqqatni qaratish lozim?
8. Operatsiyalar modellashtirish jarayonida dasturiy vositalar qanday foydalaniladi?

13-MAVZU. OPERATSIYALAR MARKERLARI

Reja:

1. Modellar va ularning turlari
2. Sikl markerlari
3. Jarayonosti markeri

Texnologik operatsiyalarni bajarish tartibi va marshrutlarni modellashtirish uchun boshqaruv oqimlari qo'llaniladi. Ular diagrammada oldingi operatsiyani keyingisi bilan yo'nalishli qalin, uzluksiz o'q orqali birlashtiradi. Boshqaruv oqimi har doim hovuz ichida joylashadi. Yo'nalishli o'qda o'tish shartini ko'rsatadigan yozuv bo'lishi mumkin. Jarayonni amalga oshirishning afzal yo'nalishini modellashtirish uchun odatda shartsiz ketma-ket oqim ishlatiladi, unda hech qanday shartlar mavjud emas. Agar keyingi operatsiya muvaffaqiyatli yakunlangan bo'lsa va uning maqsadiga erishilgan bo'lsa, jarayon operatsiyalari qanday bajarilishini ko'rsatadi. Jarayondagi tarmoqlanishlarni modellashtirish uchun shartli uzatish oqimi qo'llaniladi, bu oqimning bajarilishini aniqlaydigan ba'zi mezonlarni o'z ichiga olishi mumkin. Shartli o'tishda shart tekshiriladi, agar shart rost bo'lganda bajariladi. Agar tekshirilayotgan shart boshqa barcha oqimlar uchun to'g'ri bo'lmasa, standart boshqaruv oqimi ishlatiladi.

Boshqarishning shartsiz ketma-ket oqimi, jarayon operatsiyalari bajarilish tartibini ko'rsatadi. Shartli oqim boshqaruvni o'tkazish kerak bo'lgan yo'nalishni tanlashga yordam beradi. Agar shartli oqim operatsiyadan chiqsa, u holda rom belgisi talab qilinadi. Agar shartli oqim mantiqiy operatoridan chiqsa, u holda rom belgisi o'rnatilmaydi. Standart bo'yicha boshqarish oqimi boshqa barcha oqimlar uchun shart yolg'on bo'lsa, bajariladigan marshrutni belgilaydi.

BPMN yozuvida mantiqiy operatorlar boshqaruv oqimlarini turli yo'nalishlarga yo'naltirish va bir nechta oqimlarni sinxronlashtirish uchun ishlatiladi. Ular ikki xil bo'ladi: tarmoqlanish va birlashtirish, birinchisi boshqaruv oqimini bir necha yo'nalishga ajratadi va yo'naltiradi, ikkinchisi bir nechta oqimlarni birlashtiradi va sinxronlashtiradi. Tarmoqlangan mantiqiy operator bitta kiruvchi va kamida ikkita chiquvchi oqimga ega, birlashtiruvchi mantiqiy operator esa bir nechta kiruvchi va bitta chiquvchi oqimga ega. Kiruvchi va

chiquvchi boshqaruv oqimlari faqat burchaklarga emas, balki mantiqiy operator chegarasining istalgan nuqtasiga biriktirilishi mumkin.

Kontseptual modellar - simulyatsiya qilingan jarayonning tuzilishini, uni amalga oshirishning asosiy bosqichlarini, ularning sabab-oqibat munosabatlarini aniqlaydi. Bu ko'rib chiqilayotgan jarayonning dastlabki, taxminiy g'oyasi, buning uchun cheklangan belgilar to'plamidan foydalangan holda yuqori darajadagi abstraktsiyalarni o'z ichiga oladi

Analitik modellar o'rganilayotgan jarayonning biznes muammolarini aniqlash vositasi bo'lib xizmat qiladi. Ular keyinchalik bajarish uchun mo'ljallanmagan, shuning uchun ular ba'zi tafsilotlarni o'tkazib yuborishlari mumkin, masalan, faqat eng mumkin bo'lgan ijro senariylarini ko'rsatishi, kamdan-kam qo'llaniladiganlarini tashlab qo'yishi va past darajadagi harakatlar tafsilotlarini o'z ichiga olmaydi. Bu modellar biznes-jarayonning vizual tasviriga qaratilgan. Ular biznes-tahlilchilar tomonidan eng ko'p qo'llaniladigan kengaytirilgan elementlar to'plamidan foydalanadi, lekin ijro semantikasi dasturlashtirilgan elementlardan foydalanadi.

Bajariladigan modellar BPMS(Business Process Management System) muhitida ishlash uchun mo'ljallangan. Ular barcha qabul qilinadigan bajarish senariylarni, hatto kamdan-kam qo'llaniladigan senariylarni qamrab oladi, ularsiz tizim ishlamaydi, elementar harakatlar tafsilotlarini o'z ichiga oladi, ishtirokchilarni, ijrochilarni va ularning tizim obyektlariga kirish huquqlarini belgilaydi. Bajariladigan modellar notatsiya elementlarining to'liq to'plamini, shu jumladan ijro semantikasi aniq ko'rsatilganlarni o'z ichiga oladi.

Jarayonosti - bu tugallangan harakatni bajaradigan boshlanishi va oxiri bo'lgan mantiqiy bog'liq harakatlar ketma-ketligi.

Jarayon ostilar ikki xil bo'lishi mumkin:

Ichki (embedded) va qayta foydalanish mumkin, **global** miqiyosda ma'lum (global).

Boshlanish hodisasi sifatida belgilangan voqea sodir bo'lgan bo'lsa, jarayonostilar boshlanadi. Mavjud bo'lgan barcha nazorat zanjirlari tugagandan so'ng u tugaydi. Ichki jarayonostilar asosiy jarayonning bir qismidir, u faqat mahalliy sifatida aniqlanadi va ma'lum va boshqa loyihalarda foydalanilmaydi. Ichki jarayonostilar yuqori darajadagi asosiy jarayon bilan bir xil model varag'ida ko'rsatilgan. U turli ko'rinishda ko'rsatilishi mumkin. Birinchi holda, u quyi jarayon

belgisi qo'shilgan operatsiyaga o'xshaydi, ikkinchi holatda u operatsiyalar, mantiqiy operatorlar, hodisalar va boshqaruv o'tish belgilaridan tashkil topgan alohida jarayonni ifodalaydi. Ichki jarayonostilar faqat bitta oddiy yozilmagan hodisani bajarishni boshlaydigan boshlang'ich hodisa sifatida ishlatadi. Jarayon merosxo'rlik sxemasida tasvirlanganligi sababli, u pool(hovuz)lar va chiziqlarni ifodalay olmaydi.

Operatsiya notatsiyaning asosiy grafik elementi bo'lib, u bir nechta bajarish usullarini beradi. Belgilar operatsiyalarni qanday bajarish kerakligini ko'rsatish uchun ishlatiladi. Ular grafik elementning ustiga joylashtiriladi, shu bilan ma'lum bir semantikani anglatadi. BPMN 2.0 notatsiyasi 6 turdagi markerlarga buysunadi, jumladan har xil turdagi sikllarni tasvirlash uchun 3 xil marker: oddiy sikl; parallel bajarish, parallel ijro etish; ketma-ket ijro, kompensatsiya, "tasodifiy" operatsiya (Ad-Hoc).

Sikl markerlari. Sikl markerlari operatsiyalarni bajarilishi ketma-ketligini, siklligini bildiradi. Sikl uchun takrorlanish shartlarini o'rnatish mumkin: "while...do", "repeat...until". Takrorlashlar soni va bajarish usuli operatsiyaning o'rnatilgan atributlari bilan belgilanadi va ularni moslashuvchan tarzda qayta sozlash mumkin.

Jarayonosti marker. Jarayonosti belgisi tahlilchiga u kompozitsion operatsiya bilan shug'ullanayotgani haqida signal beradi - ya'ni alohida operatsiyalarga bo'linishi mumkin bo'lgan jarayon ostilar. Tarkibiy operatsiyalarning tarkibiy tuzilishi cheklanmagan. Tarkibiy operatsiyani qisqartirish yoki kengaytirish mumkin. Kengaytirilgan jarayonosti alohida diagramma sahifasida batafsil tavsiflanadi.

Parallel bajarish markeri operatsiyaning kerakli nusxasining sonini bir vaqtning o'zida parallel bajarilishini anglatadi. Tahlilchi modellashtirish bosqichida bajarilish vaqtida nusxalarning sonini aniq bo'lmaganda ishlatadi. Nusxalar soni hisoblash faqat bir marta amalga oshiriladi, operatsiyaning kerakli soni ishga tushiriladi. Takrorlashlar soni va bajarilish semantikasi operatsiyaning o'rnatilgan atributlari bilan belgilanadi. Tahlilchi nusxalar sonini aniq ko'rsatishi yoki jarayonning boshqa nusxasi yaratilishi uchun aniq shartlarni belgilashi mumkin. Tegishli atributlar bir nechta jarayon nusxalarini ishga tushirishni muvofiqlashtirish va tugatish shartlarini belgilaydi.

Ketma-ket bajarilish markeri bir nechta nusxalarning siklik bajarilishini ifodalaydi. Ish odatdagi sikldagi kabi tashkil etilgan. Sikl uchun "while...do", "takrorlash... qadar" kabi shartlarni ham oʻrnatish mumkin. Takrorlashlar soni va bajarish usuli xuddi parallel bajarish kabi operatsiyaning oʻrnatilgan atributlari bilan belgilanadi. Ketma-ket bajarilish markerida, agar isSequential atributining mos qiymati oʻrnatilgan boʻlsa, parallel bajarilishini bildiradi, faoliyat markeri mos ravishda oʻzgaradi.

Ad-hoc operatsiya muayyan muammo yoki vaziyat uchun moʻljallangan yechimni anglatadi. "Ad-hoc" markeri har biri oʻz harakatni hal qilish uchun moʻljallangan ichki operatsiyalar toʻplamidan iborat murakkab harakatni koʻrsatish uchun ishlatiladi va bu operatsiyalar bir-biri bilan bogʻliq boʻlishi shart emas. Foydalanuvchi bajarilish bosqichida tegishli operatsiyani tanlashi mumkin. Amaliyotlarni bajarish ketma-ketligi va ularni takrorlash soni tartibga solinmaydi va ish jarayonida foydalanuvchi tomonidan belgilanadi. Ushbu marker tuzilmalashmagan jarayonlarni modellashtirishda foydalidir. "sayohatga tayyorgarlik" "ad-hoc" operatsiyasini koʻrsatadi.

Kompensatsiya operatsiyasi oldindan tugallangan baʼzi operatsiyalarda bajarilgan harakatlarni bekor qilish mantiqini tavsiflash uchun ishlatiladi. Faraz qilaylik, bir nechta operatsiyalardan tranzaksiya tashkil etilsin. Tranzaksiyani tashkil etuvchi operatsiyalarning baʼzilari allaqachon muvaffaqiyatli yakunlangan, ammo qolganlari muvaffaqiyatsiz yakunlangan. Bu shuni anglatadiki, bitim tugallanmagan, va ilgari tugallangan operatsiyalar bekor qilinishi kerak, buning uchun ulardagi oʻzgartirilgan maʼlumotlar tranzaksiya boshlanishidan oldingi holatga qaytarilishi kerak. Misol uchun, biz chiptalarni bron qilishni boshladik, mijozning hisobidagi mablagʻlarni blokladik, ammo bron qilish bekor qilindi, mablagʻlarni blokdan chiqarish kerak edi. A operatsiyasi kompensatsiya hodisasiga biriktirilgan boʻlib, u yoʻnaltirilgan assotsiatsiyadan foydalanib, kompensatsiyani amalga oshiradigan operatsiyaga ishora qiladi. Agar tranzaksiya tugallanmagan boʻlsa, u holda signal hosil boʻladi, kompensatsiya operatsiyasini boshlaydi, bu esa maʼlumotlarni asl holatiga qaytaradi.

Jarayon operatsiyalari elementar yoki tarkibiy boʻladi. Elementar operatsiyalar bitta ijrochiga tayinlangan boʻlishi mumkin. Tarkibiy

operatsiyalar quyi tartibdagi tarkibiy harakatlariga dekompozitsiyalanish nazarda turiladi. Qoidaga ko'ra, tarkibiy operatsiyaning tafsilotlari(detalizatsiya) alohida jarayon diagrammasida tasvirlanadi. Ushbu turdagi elementlar jarayon ostilar ataladi.

BPMN notatsiyada operatsiyalarni tasvirlash uchun turli grafik elementlardan foydalaniladi. Bir necha turlari mavjud: operatsiya, jarayonosti, global (chaqiruv) operatsiya, tranzaksiya.

Operatsiya deganda jarayon diagrammasida tasvirlangan ish birligi tushuniladi. Operatsiyalar har qanday jarayonning asosiy tarkibiy qismidir.

Jarayonosti - bu mantiqiy jihatdan bog'liq bo'lgan operatsiyalar guruhi. Jarayonosti bir yoki bir nechta oqim bilan bog'liq operatsiyalarni o'z ichiga olishi mumkin.

Global (chaqiruv) operatsiya qayta foydalanish mumkin bo'lgan, global ma'lum bo'lgan jarayonostilarni chaqirishni bildiradi.

Tranzaksiya - bu to'liq yoki umuman bajarilmasligi mumkin bo'lgan jarayonostidir. Agar tranzaksiya tugallanmagan bo'lsa, barcha o'zgartirilgan ma'lumotlarni tranzaksiya boshlanishidan oldingi holatga qaytarish kerak bo'ladi.

Foydalanuvchining operatsiyasi - bu dasturiy ta'minotdan foydalangan holda ijrochi-inson tomonidan vazifani bajarish. Quyidagi misol buyurtmani qayta ishlash jarayonini ko'rsatilgan. Foydalanuvchi operatsiyasi xaridni tasdiqlash masalasida tasvirlashda foydalanadi. Bunda xarid xaridor(ya'ni, ijrochi-inson tomonidan) tomonidan xarid qilish tizim(ya'ni, dasturiy ta'minot ilovasi) o'zaro harakat vositasi orqali bajariladi.

AT ning hayotiylik sikli doimiy jarayon bo'lib, u AT yaratish zarurligi to'g'risida qaror qabul qilingan paytdan boshlab boshlanadi va u foydalanishdan butunlay olib tashlangan paytda tugaydi.

Tizimni yaratish to'g'risida qaror qabul qilingan paytdan boshlab tizim faoliyatini to'xtatgunga qadar AT rivojlanishida o'tadigan bosqichlar va bosqichlar to'plami ATning hayotiylik sikli deb ataladi.

Loyihaning hayotiylik sikli bir nechta bosqichlardan iborat bo'ladi. Bosqichlar sonini va ularning mazmunini aniqlashda ba'zi farqlar mavjud, chunki bu xususiyatlar ko'p jihatdan ma'lum bir loyihani amalga oshirish shartlariga va asosiy ishtirokchilarning

tajribasiga bog‘liq. Shunga qaramay, deyarli barcha holatlarda ATni ishlab chiqish jarayonining mantiqiy va asosiy mazmuni umumiydir.

Nazorat savollari

1. Operatsiyalar markerlari nima? Ularning asosiy maqsadi va foydalanish sohalarini ta'riflang.

2. Operatsiyalar markerlarini qanday turlarga bo‘lishi mumkin? Ularning turlari va ularning har birining vazifasi qanday?

3. Operatsiyalar markerlarini qanday tuzish lozim? Ularning tuzilishi va o‘ziga xosliklari nimalar?

4. Operatsiyalar markerlari nima uchun ishlatiladi modellashtirishda? Ularning modellashtirish jarayonida roli va ahamiyati nimalar?

5. Operatsiyalar markerlarining foydalanish sohalarini qanday aniqlash mumkin? Ularning biznes jarayonlarni boshqarishda qo‘llanish sohalarini ko‘rsatishning eng yaxshi usullari qanday?

6. Operatsiyalar markerlarini qanday shakllantirish lozim? Ularning yaratilishi va qo‘llanilishida qanday qoidalar va standartlar mavjud?

7. Operatsiyalar markerlarini qanday tuzilishi mumkin? Ularning modellashtirishda o‘zlashtirilgan usullari nima?

8. Operatsiyalar markerlarining foydalanishining afzalliklari va kamchiliklari qanday? Ularning foydalanishining o‘rtacha xilma-xilligi va modellashtirishning eng muhim qiyinchiliklari nimalar?

9. Operatsiyalar markerlari vaqt, resurs, va hisob-kitoblarni boshqarishda qanday yordam beradi? Ularning operatsion jarayonlarni optimallashtirishda va ish rejalarini rivojlantirishda roli nima?

14-MAVZU. JARAYON MODELI

Reja:

1. Modellashirish bosqichlari
2. Asboblari
3. Tolerantliklar

Kontseptual modellashirish - bu real dunyo yoki muammo sohasining mavhum tasvirlarini yaratish jarayoni. Kontseptual model - bu mohiyat va tuzilmani tavsiflash va aniqlashtirishga yordam beradigan tushunchalar, munosabatlar va cheklovlar to'plami. U axborot tizimlarini keyingi loyihalash va joriy etish uchun asos bo'lib xizmat qiladi.

Bu nima uchun kerak?

Kontseptual modellashirishning asosiy maqsadi muammo sohasining aniq va rasmiylashtirilgan tasvirini yaratishdir. Bu loyiha ishtirokchilariga (mijoz, tahlilchilar, ishlab chiquvchilar va boshqa manfaatdor tomonlar) ma'lumotlar va operatsiyalarni tashkil etish va turlari haqida umumiy tushunchaga ega bo'lish imkonini beradi.

Modellashirish bosqichlari

1. Identifikatsiya: nima qilish kerakligini aniqlang. Bu biznes jarayoni, tashkilot, tizim yoki boshqa obyekt bo'lishi mumkin.

2. Ma'lumot to'plash: tayyorgarlik ko'rish - suhbatlar o'tkazish, hujjatlarni o'rganish, ish jarayonlarini kuzatish va boshqa tadqiqot usullari. Maqsad - xususiyatlar va talablarni to'liq tushunishdir.

3. Ta'rif: Taqdim etiladigan asosiy tushunchalarni aniqlang. Loyihani boshqarish tizimi uchun - loyihalar, vazifalar, resurslar.

4. Aloqalarni aniqlang: loyihada bir nechta vazifalar bo'lishi mumkin va vazifa muayyan resurslar bilan bog'lanishi mumkin. Bu turli elementlar o'rtasidagi o'zaro ta'sirlarning to'liq tasvirini yaratishga yordam beradi.

5. Rasmiylashtirish va hujjatlashtirish: Endi kontseptsiyani maxsus belgilar va tillar yordamida rasmiylashtiring (ER diagrammasi yoki UML sinf diagrammasi). Loyihada ishtirok etayotgan boshqalar uning mazmuni va maqsadlarini bilishlari uchun hujjatlashtirish ham muhimdir.

Asboblari

Mana bir nechta asosiy vositalar:

ER: obyektlar (tushunchalar) va ular o'rtasidagi munosabatlarning grafik tasviri. Obyektlar to'rtburchaklar bilan, munosabatlar esa olmos yoki chiziqlar bilan ifodalanadi. Ushbu vosita, ayniqsa, ma'lumotlar bazalari uchun foydalidir.

UML: Har xil turdagi diagrammalarni, jumladan sinf diagrammalarini, foydalanish holatlari diagrammalarini, faoliyat diagrammalarini va boshqalarni yaratishni qo'llab-quvvatlaydigan til. UML sinf diagrammasi Sinflar, ularning atributlari va munosabatlarini kengaytiradi.

Ontologiyalar: ma'lum bir domenning rasmiylashtirilgan tavsiflari. Ular ma'lumotlarning semantikasini tushunish va tavsiflashda yordam beradigan rasmiy va tuzilgan modellarni yaratish uchun ishlatiladi.

Aql xaritalari: g'oyalar, tushunchalar va ularning munosabatlarining grafik tasvirlari. Ular dastlab asosiy qiymatlarni ko'rsatish uchun ishlatilishi mumkin.

CASE: Modellashtirish va dasturiy ta'minotni ishlab chiqish jarayonini qo'llab-quvvatlash uchun maxsus ishlab chiqilgan dasturiy vositalar. Ular yaratish, tahrirlash, vizualizatsiya qilish va hujjatlashtirish uchun vositalar to'plamini taqdim etadi.

Ilova

Kontseptual modellashtirish turli sohalarda keng qo'llaniladi, bu erda ma'lumotlarning tuzilishi va munosabatlarini aniq tushunish muhimdir. Ba'zi ilovalar misollari:

1. Axborot tizimlari: ma'lumotlar bazasini loyihalash, axborot tizimlarini ishlab chiqish va dasturiy ta'minot talablarini aniqlashda yordam beradi.

2. Biznes jarayonlari: ish jarayonlarini optimallashtirish, qiyinchiliklarni aniqlash va tashkilot samaradorligini oshirish imkonini beradi.

3. Ilmiy tadqiqotlar: murakkab tadqiqotlarni tavsiflash va turli o'zgaruvchilar o'rtasidagi munosabatlarni aniqlashga yordam beradi.

4. Muhandislik dizayni: aviatsiya yoki energiya kabi murakkab loyihalarni loyihalash uchun ishlatiladi.

Kontseptual modellashirish murakkab tizimlar va muammoli sohalarni tushunish va loyihalash uchun muhim vositadir. Bu sizga ma'lumotlarni tuzish, ma'lumotlarni tahlil qilish va asosli qarorlar qabul qilishda yordam beradigan mavhum tasvirlarni yaratishga imkon beradi.

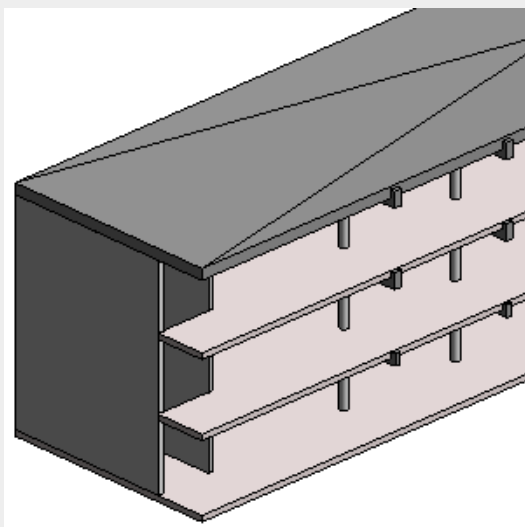
Analitik model yuk ko'taruvchi konstruksiyalarning fizik modelining to'liq muhandislik dizayni tavsifining soddalashtirilgan 3D ko'rinishidir. Analitik model yuk ko'taruvchi elementlarni, geometriyani, materialning xususiyatlarini va birgalikda muhandislik tizimini tashkil etuvchi yuklarni o'z ichiga oladi.

Strukturaviy analitik model har bir konstruktiv element uchun bittadan yuk ko'taruvchi elementlarning analitik modellaridan iborat. Quyidagi konstruktiv elementlar analitik konstruktiv elementlar modellariga ega: konstruktiv ustunlar, konstruktiv ramka elementlari (nurlar yoki qavslar kabi), konstruktiv pollar, strukturaviy devorlar va strukturaviy poydevor elementlari.

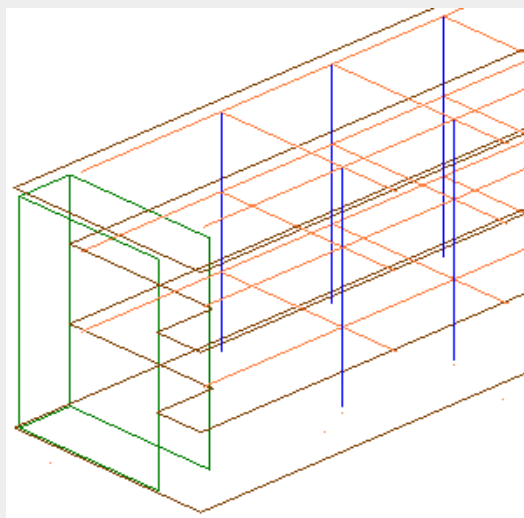
Har qanday yuk ko'taruvchi elementning analitik modeli quyidagilarni o'z ichiga oladi:

- Namuna parametrlari
- Materialning fizik xususiyatlari
- Strukturaviy elementning o'ziga nisbatan standart pozitsiya
- Proyeksiya tekisligiga nisbatan joylashuv, joylashtirish yoki joylashuvni sozlash orqali aniqlanadi

Jismoniy model



Analitik model



Jismoniy model yaratganingizda analitik model avtomatik ravishda yaratiladi va tahlil va dizayn ilovalariga eksport qilinishi mumkin.

Revitdagi jismoniy model jismoniy tuzilmani ifodalovchi ko'rinishlar ichida yaratilgan. Jismoniy modelni ishchi chizmalar to'plami sifatida ko'rish mumkin. Jismoniy model shakllanar ekan, analitik modelni yaratish dinamik ravishda sodir bo'ladi. Revit foydalanuvchiga analitik model bilan ishlash imkoniyatini beradi.

Analitik model jismoniy model bilan ishlashda ko'rish mumkin bo'lgan strukturaviy elementlarning soddalashtirilgan tasviri emas. Dastlab, analitik model fizik modelga geometrik jihatdan bog'liq, lekin uni mustaqil obyektlar to'plami sifatida ko'rish qobiliyatiga ega.

Avtomatik analitik model tekshiruvi bilan loyihaning analitik modelida mumkin bo'lgan xatolik mavjud bo'lganda ogohlantirish ko'rsatiladi. Agar strukturaning ko'p qismi allaqachon modellashtirilgan bo'lsa va siz modelga kiritilgan o'zgarishlar qo'llab-quvvatlanmaydigan elementlarga yoki analitik modelning yaxlitligini yo'qotishiga olib keladimi yoki yo'qligini kuzatmoqchi bo'lsangiz, ushbu variantlardan foydalaning. Loyihaning dastlabki bosqichlarida ushbu variantlardan foydalanish tavsiya etilmaydi. Modelni yaratishda qo'llab-quvvatlanmaydigan elementlarning soni sezilarli.

- **Element qo'llab-quvvatlaydi.** Model yaratilayotganda yoki o'zgartirilayotganda element qo'llab-quvvatlanmasa, ogohlantirish ko'rsatiladi. Doiraviy havolalar ushbu dialog oynasining Elementni qo'llab-quvvatlashni tekshirish bo'limida aniqlanishi mumkin.

- **Analitik/fizik modelning yaxlitligi haqida** ogohlantirishlar elementni yaratish yoki o'zgartirish uchun ko'rsatiladi:

- tayanchsiz barcha yuk ko'taruvchi elementlar;
- analitik modeldagi barcha qarama-qarshiliklar;
- analitik va jismoniy modellar o'rtasidagi barcha qarama-qarshiliklar;
- belgilangan jismoniy moddiy xususiyatlar to'plamiga ega bo'lmagan barcha analitik elementlar.

Ushbu tekshirish uchun parametrlar ushbu dialog oynasining Toleranslar va Analitik/Fizik model yaxlitligini tekshirish bo'limlarida aniqlanishi mumkin.

Tolerantliklar

Toleranslar parametrlari analitik/fizik modelning yaxlitligini tekshirish va analitik modelni avtomatik aniqlash uchun tolerantliklarni o'rnatish uchun ishlatiladi.

- **Qo'llab-quvvatlovchilar orasidagi masofa.** Elementning jismoniy modeli va qo'llab-quvvatlovchi elementning jismoniy modeli o'rtasidagi maksimal ruxsat etilgan masofa. Agar bu bardoshlik oshib ketgan bo'lsa, yaxlitlikni tekshirish paytida ogohlantirish ko'rsatiladi.

- **Analitik va jismoniy modellar orasidagi masofa.** Analitik va jismoniy modellar orasidagi maksimal ruxsat etilgan masofa. Agar bu tolerantlik oshib ketgan bo'lsa, yaxlitlikni tekshirish paytida ogohlantirish ko'rsatiladi.

- **Analitik model ulanishlarini avtomatik aniqlash - gorizontal.** Analitik va jismoniy modellar orasidagi maksimal gorizontal masofa.

- **Analitik model ulanishlarini avtomatik aniqlash - vertikal.** Analitik va jismoniy modellar orasidagi maksimal vertikal masofa.

- **Qattiq havolani avtomatik aniqlash.** Avtomatik qattiq bog'lanishni yaratish mumkin bo'lgan 3D kosmosdagi minimal masofa (gorizontal yoki vertikal). Qattiq bog'lanishlar fizik geometriyani qo'shmasdan analitik modelga qattiqlikni ta'minlaydi. Ushbu bardoshlik bilan bog'lanishlarni hisoblashda jismoniy model hisobga olinmaydi.

Tekshirish elementi qo'llab-quvvatlaydi

Elementni qo'llab-quvvatlash testi opsiyasi avtomatik yoki moslashtirilgan elementlarni qo'llab-quvvatlash sinovi paytida ishlatiladi.

- **Siklik qo'llab-quvvatlash elementlari.** Dairesel siklik yordamni tekshirishga imkon beradi.

Nazorat savollari

1. Jarayon modeli nima? Uning asosiy maqsadi va ahamiyati nima?
2. Jarayon modelini yaratishda qanday asosiy qadamlar mavjud?
3. Jarayon modellari qanday turli ko'rinishlarda bo'lishi mumkin? Ularning turli turdagi foydalanish sohalarini ko'rsating.
4. Jarayon modelini yaratish jarayonida qanday metodologiyalar va modellar foydalaniladi?
5. Jarayon modeli tuzishda muhim elementlar qanday? Ularning tuzilishi va vazifalari nimalar?

6. Jarayon modellari uchun foydalaniladigan muhim dasturiy vositalar va platformalar qanday? Ularning afzalliklari va qo'llanish sohalarini ko'rsating.

7. Jarayon modellari va biznes jarayonlar modellari o'rtasidagi farqlar nima?

8. Jarayon modellari yaratishda qanday monitoring va yangilanish usullari foydalaniladi?

15-MAVZU. MANTIQUIY OPERATORLAR

Reja:

1. Mantiqiy dastur tuzilmasi.
2. Shartli, shartsiz va tanlash operatorlari.
3. Takrorlash operatorlari. Ularning turli formalari (parametrli, shartni oldin va keyin tekshiruvchi operatorlar).
4. Takrorlanivchu strukturali dasturlar. Murakkab algoritmlarni ifodalaydigan soha masalalari.

Mantiqiy tur – mantiqiy ifodalarning qiymatlarini qabul qiluvchi o‘zgaruvchilarni e’lon qilishda ishlatiladi. Bu turga tegishli o‘zgaruvchilar *0 (yolg‘on)* yoki *1 (rost)* qiymatlardan birini qabul qiladi.

Dasturlash tillarida tuzilgan mantiqiy ifodalar to‘liq matematik mantiqqonuniyatlariga asoslanadi.

Matematik mantiq – fikrlashning shakli va qonuniyatlari haqidagi fan. Uning asosini mulohazalar hisobi tashkil qiladi. *Mulohaza* – bu ixtiyoriy jumla bo‘lib, unga nisbatan rost yoki yolg‘on fikrni bildirish mumkin. *Masalan* « $-5 > -25$ », «*11 – juft son*», «*Toshkent- Hindistonning poytaxti*», «*Samarqand – tarixiy shahar*» va hokazolar. Lekin «*78 987 256 katta son*» jumlasini to‘g‘ri mulohaza hisoblanmaydi, chunki «*katta son*» tushunchasi juda ham nisbiy, ya’ni katta son deganda qanday sonni nazarda tutish aniq emas. Shuning uchun ham yuqoridagi jumlaning rost yoki yolg‘onligi haqida fikr bildirish qiyin. Mulohazalarning *rost* yoki *yolg‘on* bo‘lishi holatlarga bog‘liq ravishda ham o‘zgarishi mumkin: *Masalan*, «*hozir soat 13.00*» jumlasining rost yoki yolg‘onligi qaralayotgan vaqtga bog‘liq. Xuddi shunday « $x > 0$ » ifodaning qiymati x o‘zgaruvchisining qiymatiga bevosita bog‘liq bo‘ladi.

C oilasidagi dasturlash tillarida mantiqiy ifodalar qurish uchun quyidagi taqqoslash amallari kiritilgan “ $>$ ” (*katta*), “ $<$ ” (*kichik*), “ $>=$ ” (*katta yoki teng*), “ $<=$ ” (*kichik yoki teng*), “ $==$ ” (*teng*), “ $!=$ ” (*teng emas*).

C++ tilida mantiqiy tur «Bul algebrasi» asoschisi, Angliyalik matematik Jorj Bul sharafiga *bool* kalit so‘zi bilan ifodalangan.

«*Bul algebrasi*» da mantiqiy mulohazalar ustida uchta amal aniqlangan:

Inkor – A mulohazaning inkori deganda A rost bo‘lganda yolg‘on, yolg‘on bo‘lganda rost qiymat qabul qiluvchi mulohazaga aytiladi. C++

tilida inkor – ‘!’ belgisi bilan ifodalanadi. *Masalan*, A mulohazaning inkori « $\neg A$ » ko‘rinishida yoziladi;

Konyunksiya – ikkita A va B mulohazalar konyuksiyasi yoki mantiqiy ko‘paytmasi « $A \& B$ » ko‘rinishida yoziladi. Bu mulohaza qiymati A va B mulohazalarning har ikkalasi rost bo‘lgandagina rost bo‘ladi, aks holda esa yolg‘on bo‘ladi (« $\&$ » amali «va» deb o‘qiladi). *Masalan*, «bugun oyning 8-kuni va bugun payshanba» mulohazasi oyning 8-kuni payshanba bo‘lgan kunlar uchun rost bo‘ladi;

Dizyunksiya – ikkita A va B mulohazalar dizyunksiyasi yoki mantiqiy yig‘indisi « $A \vee B$ » ko‘rinishida yoziladi. Bu mulohaza rost bo‘lishi uchun A yoki B mulohazalardan hech bo‘lmaganda bittasi rost bo‘lishi etarli. (« $\vee$ » amali «yoki» deb o‘qiladi). *Masalan*, bugun “dushanba yoki chorshanba” mulohazasi haftaning dushanda yoki chorshanba kunlari uchun rost hisoblanadi;

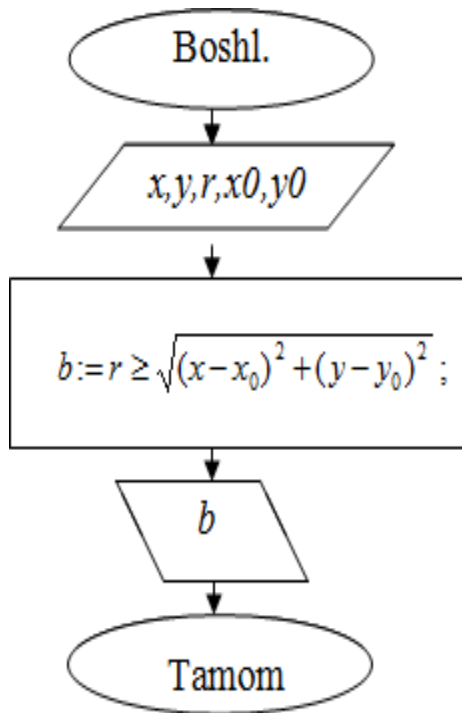
Yuqorida keltirilgan fikrlar asosida mantiqiy amallar uchun rostlik jadvali aniqlangan:

Rostlik jadvali.

<i>Mulohazalar</i>		<i>Mulohazalar ustida amallar</i>		
A	V	$\neg A$	$A \& V$	$A \vee V$
0	0	1	0	0
0	1	1	0	1
1	0	0	0	1
1	1	0	1	1

Mantiqiy tur qiymatlari ustida mantiqiy ko‘paytirish, qo‘shish va inkor amallarini qo‘llash orqali murakkab mantiqiy ifodalarni qurish mumkin. Misol uchun, « x –manfiy va y ning qiymati $[-1..3]$ sonlar oralig‘iga tegishli emas» mulohazasining mantiqiy ifodasi quyidagicha bo‘ladi: $(x < 0) \& (u < -1) \vee (y > 3)$.

Masala. Tekislikda markazining koordinatalari x, y , radiusi r bo'lgan aylana, hamda nuqta x_0, y_0 koordinatalari bilan berilgan. Berilgan nuqtaning aylanaga tegishli ekanligini aniqlovchi algoritmi va dastur tuzilsin.



```

#include <cstdlib> #include <iostream>
#include <math.h> using namespace std;
int main(int argc, char *argv[])
{ float x,y,r,x0,y0;
/*Aylana markazining koordinatalari va
radiusini kiritish*/
cin>>x>>y>>r;
/*Berilgan nuqta koordinatalarini
kiritish*/ cin>>x0>>y0;
/*Berilgan nuqtaning aylana ichida
yotishlik shartini tekshirish*/
bool b=r>=sqrt(pow(x-x0,2) +pow(y-
y0,2));
//Natijani chiqarish cout<<"b="<<b;
system("PAUSE");
return EXIT_SUCCESS;}
  
```

Funksiy algoritmi

Berilgan masalalar uchun mantiqiy mulohazalar bajarilganda 1 (rost), aks holda 0 (yolg'on) qiymat qaytaruvchi algoritmlar va dasturlar tuzilsin:

Ko'pgina masalalar yuzaga keladigan turli holatlarga bog'liq ravishda qaror qabul qilishni talab etadi. Bu borada C++ tili dasturning alohida bo'laklarining bajarilish tartibini boshqarishga imkon beruvchi qurilmalarning etarlicha katta majmuasiga ega. *Masalan*, dastur bajarilishining birorta qadamida qandaydir shartni tekshirish natijasiga ko'ra boshqaruvni dasturning u yoki bu bo'lagiga uzatish mumkin (tarmoqlanuvchi algoritmi). Bunday tarmoqlanishni amalga oshirish uchun shart operatorlaridan foydalaniladi.

if operatori: if operatori qandaydir shartni rostlikka tekshirishi natijasiga ko'ra dasturda tarmoqlanishni amalga oshiradi:

if (<shart>)<operator>;

Bu yerda <shart> mantiqiy turga tegishli o'zgaruvchi yoki mantiqiy ifoda bo'lishi mumkin.

Agar shart 0qiymatidan farqli, ya'ni *rost* bo'lsa, *<operator>* bajariladi, aks holda, ya'ni shartning qiymati 0(*yolg'on*) bo'lsa, hech qanday amal bajarilmaydi va boshqaruv *if* operatoridan keyingi operatorga o'tadi (*agar u mavjud bo'lsa*).

C++ tilining qurilmalari operatorlarni blok ko'rinishida tashkil qilishga imkon beradi. Blok – '{' va '}' belgilar oralig'iga olingan operatorlar ketma-ketligi bo'lib, u kompilyator tomonidan yaxlit bir operator deb qabul qilinadi. Blok ichida o'zgaruvchilar e'lonlari ham bo'lishi mumkin va blokda e'lon qilingan o'zgaruvchilar faqat shu blok ichida (ko'rinadi) amal qiladi, blokdan tashqarida (ko'rinmaydi) amal qilmaydi. Blokdan keyin ';' belgisi qo'yilmasligi mumkin, lekin blok ichidagi har bir ifoda ';' belgisi bilan yakunlanishi shart.

If – else operatori: Shart operatorining *if – else* ko'rinishi quyidagicha:
if (<shart>) <operator1>; else<operator2>;

Bu erda *<shart>* 0qiymatidan farqli, ya'ni *rost* bo'lsa *<operator1>*, aks holda

<operator2> bajariladi. *If-else* shart operator mazmuniga ko'ra algoritmnining tarmoqlanuvchi blokini ifodalaydi.

?: shart amali: Agar tekshirilayotgan shart nisbatan sodda bo'lsa, shart amalining '?' ko'rinishini ishlatish mumkin:

<shart> ? <ifoda1> : <ifoda2>;

Ushbu shart amali *if* shart operatoriga o'xshash ko'rinishda ishlaydi: agar *<shart>* 0qiymatidan farqli, ya'ni *rost* bo'lsa, *<ifoda1>*, aks holda *<ifoda2>* bajariladi. Odatda ifodalarning qiymatlari biror o'zgaruvchiga o'zlashtiriladi.

Goto operatori va nishonlar: Nishon – bu davomida ikkita nuqta (':') qo'yilgan identifikator. Nishon bilan qandaydir operator belgilanadi va keyinchalik dasturning boshqa bir qismidan unga shartsiz o'tish amalga oshiriladi. Nishon bilan har qanday operator belgilanishi mumkin, shu jumladan e'lon operatori va bo'sh operatorlar ham. Nishon faqat funksiyalar ichida amal qiladi.

Nishonga shartsiz o'tish goto operatori yordamida bajariladi. Goto operatori orqali faqat uning o'zi joylashgan funksiya ichidagi operatorlarga o'tish mumkin. Goto operatorining sintaksisi quyidagicha:

goto<nishon>;

Ayrim hollarda, goto operatorining «sakrab o'tishi» hisobiga xatoliklar yuzaga kelishi mumkin. *Masalan*,


```
int i=0;
i++; if(i) goto m; int j;
m: j+=i;
```

operatorlarining bajarilishi xatolikka olib keladi, chunki jo‘zgaruvchisining e’loni chetlab o‘tilgan.

Shartsiz o‘tish operatori dastur tuzishdagi kuchli va shu bilan birga xavfli vositalardan biri hisoblanadi. Kuchliligi shundaki, u yordamida algoritmnining «*boshi berk*» joylaridan chiqib ketish mumkin. Ikkinchi tomondan, bloklarning ichiga o‘tish, masalan, takrorlash operatorlarining ichiga «*sakrab*» kirish kutilmagan holatlarni yuzaga keltirishi mumkin. Shu sababli, imkon qadar *goto* operatoridan foydalanmaslik kerak, foydalanilgan taqdirda ham quyidagi qoidalarga amal qilish zarur – blok ichiga, *if*, *if-else* va *switch* operatorlari ichiga, hamda takrorlash operatorlari tanasiga tashqaridan “sakrab” kirish mumkin emas. Garchi, nishon yordamida dasturning ixtiyoriy joyiga o‘tish mumkin bo‘lsa ham, boshlang‘ich qiymat berish e’lonlaridan sakrab o‘tish man etiladi, lekin bloklardan sakrab o‘tish mumkin.

Masalan:

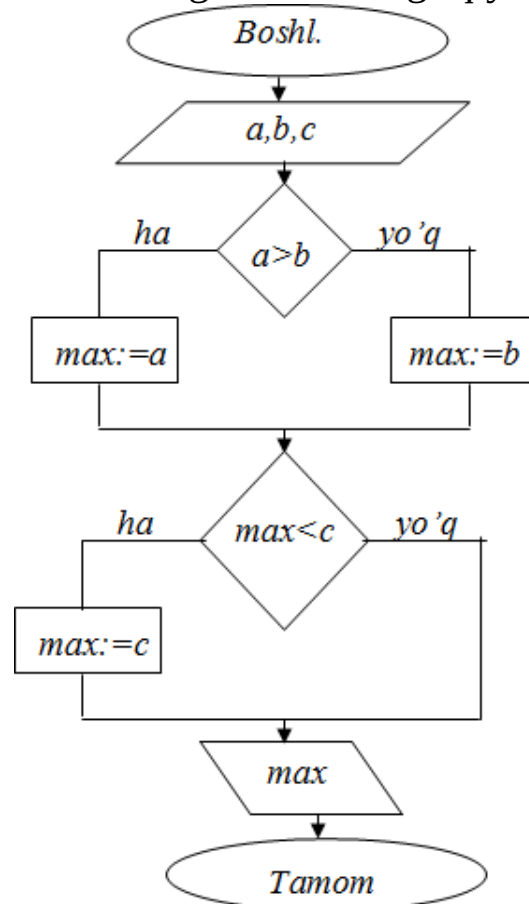
```
...
goto B3A; \xato float x=0.5;
goto B3A; \ to ‘g ‘ri
{int n=10;x=n*x+8;} B3A: cout<< “x=”<<x;
```

Masala. Berilgan 3 ta *a, b, co*‘zgaruvchilardagi sonli qiymatlarning eng kattasini aniqlovchi algoritm va dastur tuzilsin.

```

#include <cstdlib> #include
<iostream> using namespace std;
int main(int argc, char *argv[])
{ int a,b,c,max;
/*Berilgan 3 ta o'zgaruvchiga sonli
qiymatlar kiritish*/
cin>>a>>b>>c;
/*Dastlabki 2 ta sonning kattasini
topib, max o'zgaruvchisiga
qiymatlash*/
if (a>b) max=a; else max=b;
/*max o'zgaruvchisidagi qiymat bil

```



```

an 3-sonni taqqoslash*/
if (max<c) max=c;
//Natijani chiqarish;
cout<<"max="<<max;
system("PAUSE");
return EXIT_SUCCESS;}

```

Tarmoqlanuvchi jarayon algoritmi

Dastur bajarilishini boshqarishning yana bir kuchli mexanizmlaridan biri – takrorlash operatorlari hisoblanadi.

Takrorlash operatori «*takrorlash sharti*» deb nomlanuvchi ifodaning rost qiymatida dasturning ma'lum bir qismidagi operatorlar(takrorlash tanasi)ning ko'p marta takror ravishda bajarilishini amalga oshiradi.

Takrorlash o'zining kirish va chiqish nuqtalariga ega bo'lib, ba'zan chiqish nuqtasi bo'lmasligi ham mumkin. Agar chiqish nuqtasi bo'lmasa cheksiz takrorlanish sodir bo'ladi. Cheksiz takrorlanish uchun takrorlashni davom ettirish sharti doimo rost bo'ladi.

Takrorlash shartini tekshirish, takrorlanish tanasidagi operatorlarning bajarilishidan oldin tekshirilishi mumkin (*for*, *while* takrorlash operatorlarida) yoki takrorlanish tanasining operatorlari bir marta bajarilgandan keyin tekshirilishi mumkin (*do-while*).

Takrorlanishlarni tashkil etishda takrorlanishlar soni va qadami oldindan ma'lum bo'lgan holatlar uchun *for* operatoridan, boshqa hollarda esa *while* hamda *do-while* operatorlaridan foydalanish maqsadga muvofiq.

Takrorlash operatorlari ichma-ich joylashishi ham mumkin. Quyida takrorlash operatorlari va ularning qo'llanilishi bilan batafsil tanishamiz.

for takrorlash operatorining sintaksisi quyidagi ko'rinishga ega:

for (<*ifoda1*>; <*ifoda2*>;<*ifoda3*>) <*operator yoki blok*>;

Bu operator o'z ishini <*ifoda1*> ifodasining bajarilishidan boshlaydi. Keyin takrorlash qadamlari boshlanadi. Har bir qadamda <*ifoda2*> tekshiriladi, agar <*ifoda2*> ning qiymati 0 qiymatidan farqli, ya'ni rost bo'lsa, takrorlash tanasi – <*operator yoki blok*> bajariladi, so'ngra

<*ifoda3*> bajariladi. Agar <*ifoda2*> ning qiymati 0 (*yolg'on*) bo'lsa, takrorlanish jarayoni to'xtaydi va boshqaruv takrorlash operatoridan keyingi operatorga o'tadi. Shuni ta'kidlash kerakki, <*ifoda2*> ifodasi vergul bilan ajratilgan bir nechta ifodalar birlashmasidan iborat bo'lishi ham mumkin, bu holda vergul bilan ajratilgan ifodalarning oxirgisi takrorlanish sharti hisoblanadi. Takrorlanish tanasi sifatida bitta operator, jumladan bo'sh operator bo'lishi yoki operatorlar bloki ham kelishi mumkin.

Quyida *for* operatori parametrlarining vazifalari bilan tanishamiz:

<ifoda1> – takrorlash sanagichiga boshlang‘ich qiymat berishga xizmat qiladi va u takrorlash jarayoni boshida faqat bir marta hisoblanadi. Ifodada o‘zgaruvchi e‘loni hambo‘lishi mumkin va bu o‘zgaruvchi takrorlash operatori tanasida amal qiladi va takrorlash operatoridan tashqarida

«ko‘rinmaydi».

<ifoda2> – takrorlanishning bajarishi yoki bajarilmasligini aniqlab beruvchi mantiqiy ifoda (*shart*), agar shart *rost* bo‘lsa, takrorlanish davom etadi, aks holda to‘xtaydi. Agar bu ifodaning o‘rni bo‘sh bo‘lsa, shart doimo *rost* deb hisoblanadi.

<ifoda3> – odatda takrorlanish sanagichining qiymatini oshirish (kamaytirish) uchun xizmat qiladi yoki unda takrorlanish shartiga ta’sir etuvchi boshqa amallar ham bo‘lishi mumkin.

Takrorlash operatorida qavs ichidagi ifodalar bo‘lmasligi ham mumkin, lekin sintaksis ‘;’ bo‘lmasligiga ruxsat bermaydi. Shu sababli sodda ko‘rinishdagi takrorlash operatori quyidagicha bo‘ladi:

for(;;).

Agar takrorlanish jarayonida bir nechta o‘zgaruvchilarning qiymati sinxron ravishda o‘zgarishi kerak bo‘lsa, <ifoda1> va <ifoda3> ifodalarida zarur operatorlarni ‘,’ bilan yozish orqali bunga erishish mumkin:

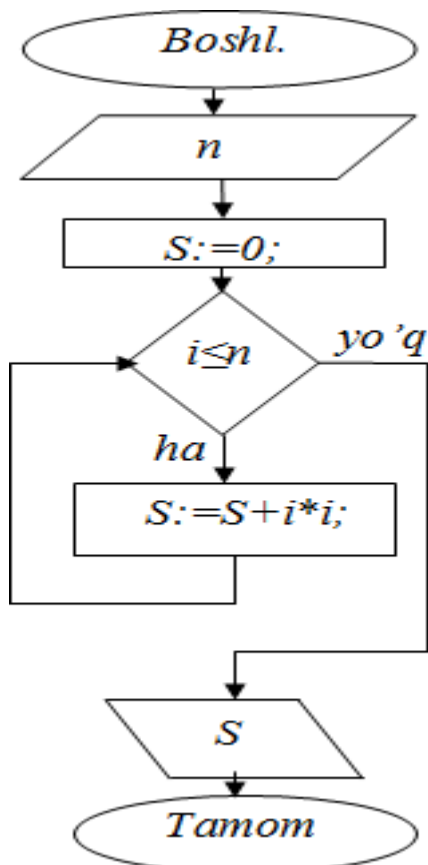
```
for(int i=1,j=2;i<=15;i++,j=i+1){...}
```

Takrorlash operatorining har bir qadamida *i* va *j* o‘zgaruvchilarning qiymatlari mos ravishda o‘zgarib boradi.

for operatorida takrorlanish tanasi bo‘lmasligi ham mumkin.

Shuni ham ta’kidlash joizki, C++ tilida *for* operatorining parametrlari haqiqiy turga tegishli bo‘lishi, takrorlanishlar qadami ham ixtiyoriy holatda belgilanishi mumkin. Bu imkoniyatlar operatorning keng qamrovli ekanligini bildiradi.

Masala. Dastlabki n ta ($n > 0$) natural sonlarning kvadratlari yig‘indisini hisoblovchi algoritm va dastur tuzilsin.



Takrorlanuvchi jarayon algoritmi.

While takrorlash operatori, operator yoki blokni takrorlanish sharti yolgʻon (0) boʻlguncha takrorlashni amalga oshiradi. Uning sintaksisi quyidagicha:

while (<ifoda>) <operator yoki blok>;

Agar <ifoda> qiymati doimo *rost* boʻlsa, takrorlanish jarayoni cheksiz davom etadi. Xuddi shunday, <ifoda> takrorlanish boshlanishida *rost* boʻlib, uning qiymatiga takrorlanish tanasidagi hisoblash taʼsir etmasa, yaʼni uning qiymati oʻzgarmasa, bu holda ham takrorlanish cheksiz davom etadi.

while takrorlanish shartini oldindan tekshiruvchi takrorlash operatori hisoblanadi. Agar takrorlanish boshida <ifoda> yolgʻon boʻlsa, *while* operatori tarkibidagi <operator yoki blok> bajarilmasdan cheklab oʻtiladi.

Ayrim hollarda <ifoda> qiymat berish operatori koʻrinishida kelishi mumkin. Bunda qiymat berish amali bajariladi va natija 0 bilan taqqoslanadi. Natija noldan farqli boʻlsa, takrorlanish davom ettiriladi.

```

#include <cstdlib> #include
<iostream> using namespace std;
int main(int argc, char *argv[])
{
    int i,n;
    //Berilgan qiymatni kiritish
    cin>>n;
    /*Yigʻuvchiga boshlangʻich
    qiymat berish*/ int s=0;
    /*1 dan n gacha boʻlgan sonlar
    kvadrlarini yigʻish*/
    for (i=1;i<=n;i++) s+=i*i;
    //Natijani chiqarish
    cout<<"S="<<s;
    system("PAUSE"); return
    EXIT_SUCCESS;}
  
```

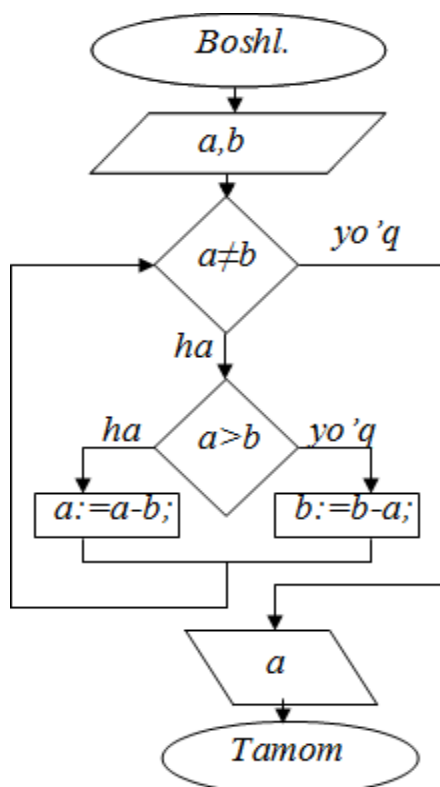
Agar rost ifodaning qiymati noldan farqli o'zgarmas bo'lsa, cheksiz takrorlanish ro'y beradi.

Masalan:

while(1); // cheksiz takrorlanish

Xuddi *for* operatoridagi kabi bu operatorida hamqavs ichida ',' yordamida ajratilgan bir nechta ifodalarni yozish mumkin.

Masala. Ikkita natural sonning eng katta umumiy bo'luvchisi (EKUB)ni *Evklid* algoritmi yordamida topish masalasi uchun algoritmi va dastur tuzilsin.



```

#include <cstdlib> #include
<iostream> using namespace std;
int main(int argc, char *argv[])
{   int a,b;
  /*Berilgan o'zgaruvchilarga qiymat
  kiritish*/ cin>>a>>b;
  /*a va b lar uchun Ekub ni topish*/
  while (a!=b)
  { if (a>b) a=a-b; else b=b-a; }
  //Natijani chiqarish cout<<a;
  system("PAUSE");
  return EXIT_SUCCESS;}
  
```

Tarmoqlanuvchi jarayon algoritmi.

Butun turdagi *a* va *b* qiymatlari oqimdan o'qilgandan keyin ular qiymatlari toki o'zaro teng bo'lmaguncha takrorlash jarayoni ro'y beradi. Takrorlashning har bir qadamida *a* va *b* sonlarining kattasidan, kichigi ayriladi va ularning tengligi tekshiriladi. Takrorlashdan keyingi ko'rsatma vositasida *a* o'zgaruvchisining qiymati natija sifatida chop etiladi.

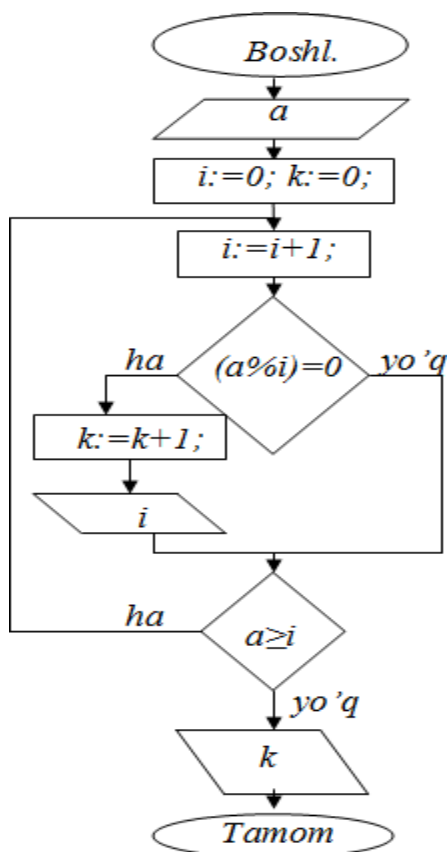
do-while takrorlash operatori *while* operatoridan farqli ravishda oldin operator yoki blokni bajaradi, keyin takrorlanish shartini tekshiradi. Bu qurilma takrorlanish tanasining kamida bir marta

bajarilishini ta'minlaydi. *do-while* takrorlash operatori quyidagi sintaksisga ega:

do <operator yoki blok> *while* (<ifoda>);

Bunday takrorlash operatorining qo'llaniladigan holatlari takrorlanishni boshlamasdan turib, takrorlanish shartini tekshirishning iloji bo'lmagan holatlar hisoblanadi. *Masalan*, birorta jarayonni davom ettirish yoki to'xtatish haqidagi so'rovga javob olish va uni tekshirish zarur bo'lsin. Ko'rinib turibdiki, jarayonni boshlamasdan oldin bu so'rovni berishning ma'nosi yo'q. Hech bo'lmaganda takrorlanish jarayonining bitta qadami amalga oshirilgan bo'lishi kerak.

Masala. Berilgan a musbat butun sonining barcha musbat butun bo'luvchilari va ularning miqdorini chop etadigan algoritmi va dastur tuzilsin.



Murakkab tarmoqlanuvchi jarayon algoritmi.

```

#include <cstdlib> #include <iostream>
#include<stdio.h> using namespace std;
int main(int argc, char *argv[])
{ unsigned int a;
  /*berilgan musbat sonni a o'zgaruvchisiga o'qish*/
  scanf("%d",&a);
  /*bo'luvchilar soni va bo'luvchini o'zida saqlovchi o'zgaruvchilar*/
  int k=0,i=0; do { i++;
  /*berilgan son bo'luvchilarini topish,ularni miqdorini hisoblash va chop etish*/
  if(!(a%i)){ k++; cout<<i<<"\n";
  cout<<"\n";}} while(a>=i);
  /*son bo'luvchilari miqdorini chop etish*/
  cout<<a<<" ning bo'luvchilari soni "<<k<<" ta"; system("PAUSE"); return
  EXIT_SUCCESS;}
  
```

Tarmoqlanuvchi jarayonlarni dasturlash.

Mantiqiy tur – mantiqiy ifodalarning qiymatlarini qabul qiluvchi o'zgaruvchilarni e'lon qilishda ishlatiladi. Bu turga tegishli o'zgaruvchilar 0 (yolg'on) yoki 1 (rost) qiymatlardan birini qabul qiladi.

Dasturlash tillarida tuzilgan mantiqiy ifodalar to'liq matematik mantiq qonuniyatlariga asoslanadi.

Matematik mantiq – fikrlashning shakli va qonuniyatlari haqidagi fan. Uning asosini mulohazalar hisobi tashkil qiladi. Mulohaza – bu ixtiyoriy jumla bo'lib, unga nisbatan rost yoki yolg'on fikrni bildirish mumkin. Masalan «-5>-25», «11–juft son», «Toshkent-Hindistonning poytaxti», «Samarqand–tarixiy shahar» va hokazolar. Lekin «78 987 256 katta son» jumlasini to'g'ri mulohaza hisoblanmaydi, chunki «katta son» tushunchasi juda ham nisbiy, ya'ni katta son deganda qanday sonni nazarda tutish aniq emas. Shuning uchun ham yuqoridagi jumlaning rost yoki yolg'onligi haqida fikr bildirish qiyin. Mulohazalarning rost yoki yolg'on bo'lishi holatlarga bog'liq ravishda ham o'zgarishi mumkin: Masalan, «hozir soat 13.00» jumlasining rost yoki yolg'onligi qaralayotgan vaqtga bog'liq. Xuddi shunday « $x>0$ » ifodaning qiymati x o'zgaruvchisining qiymatiga bevosita bog'liq bo'ladi.

C oilasidagi dasturlash tillarida mantiqiy ifodalar qurish uchun quyidagi taqqoslash amallari kiritilgan “>” (katta), “<” (kichik), “>=” (katta yoki teng), “<=” (kichik yoki teng), “==” (teng), “!=” (teng emas).

C++ tilida mantiqiy tur «Bul algebrasi» asoschisi, Angliyalik matematik Jorj Bul sharafiga bool kalit so'zi bilan ifodalanagan.

“Bul algebrasi” da mantiqiy mulohazalar ustida uchta amal aniqlangan:

Inkor – A mulohazaning inkori deganda A rost bo'lganda yolg'on, yolg'on bo'lganda rost qiymat qabul qiluvchi mulohazaga aytiladi. C++ tilida inkor – ‘!’ belgisi bilan ifodalanadi. Masalan, A mulohazaning inkori «!A» ko'rinishida yoziladi;

Konyunksiya – ikkita A va B mulohazalar konyuksiyasi yoki mantiqiy ko'paytmasi «A && B» ko'rinishida yoziladi. Bu mulohaza qiymati A va B mulohazalarning har ikkalasi rost bo'lgandagina rost bo'ladi, aks holda esa yolg'on bo'ladi («&&» amali «va» deb o'qiladi).

Masalan, «bugun oyning 8-kuni va bugun payshanba» mulohazasi oyning 8-kuni payshanba bo‘lgan kunlar uchungina rost bo‘ladi;

Dizyunksiya – ikkita A va B mulohazalar dizyunksiyasi yoki mantiqiy yig‘indisi «A || B» ko‘rinishida yoziladi. Bu mulohaza rost bo‘lishi uchun A yoki B mulohazalardan hech bo‘lmaganda bittasi rost bo‘lishi yetarli. («||» amali «yoki» deb o‘qiladi). Masalan, bugun “dushanba yoki chorshanba” mulohazasi haftaning dushanda yoki chorshanba kunlari uchun rost hisoblanadi;

Yuqorida keltirilgan fikrlar asosida mantiqiy amallar uchun rostlik jadvali aniqlangan:

Mulohazalar		Mulohazalar ustida amallar		
A	V	!A	A&&V	A V
0	0	1	0	0
0	1	1	0	1
1	0	0	0	1
1	1	0	1	1

Mantiqiy tur qiymatlari ustida mantiqiy ko‘paytirish, qo‘shish va inkor amallarini qo‘llash orqali murakkab mantiqiy ifodalarni qurish mumkin. Misol uchun, «x–manfiy va y ning qiymati [-1..3] sonlar oralig‘iga tegishli emas» mulohazasining mantiqiy ifodasi quyidagicha bo‘ladi: $(x < 0) \&\& (u < -1) || (y > 3)$.

Shart operatori

Programma tuzish mobaynida o‘zgaruvchilar qiymatiga qarab u yoki bu natijani qabul qilishga to‘g‘ri keladi. Bu o‘z navbatida programmani tarmoqlanishga olib keladi. Tarmoqlarning qaysi qismi bajarilishi ayrim shartlarga qarab aniqlanadi.

Biz shartga ko‘ra bir necha harakat yo‘lidan bittasini tanlaymiz. Misol uchun: agar bolaning yoshi 7 ga teng yoki katta bo‘lsa u maktabga borishi mumkin bo‘lsin. Buni C++ da if ni qo‘llab yozaylik. `if (yosh >= 7) maktab();`

Bu yerda shart bajarilishi yoki bajarilmasligi mumkin. Agar yosh o‘zgaruvchisi 7 ga teng yoki undan katta bo‘lsa shart bajariladi va `maktab()` funksiyasi chaqiriladi. Bu holat `true` (to‘g‘ri) deyiladi. Agar yosh 7 dan kichik bo‘lsa, `maktab()` tashlab o‘tiladi. Yani `false` (noto‘g‘ri) holat yuzaga keladi.

Shart operatori boshqarishni qaysi tarmoqqa uzatishni ta'minlaydi. Shart operatorining ikki xil ko'rinishi mavjud. Operatorning umumiy ko'rinishi va qisqa ko'rinishi.

Shart operatorining umumiy ko'rinishi:

```
if (<shart>)
```

```
<operator1>; else
```

```
<operator2>;
```

if agar, else aks holda ma'nolarini anglatadi. Shart operatorining qisqa ko'rinishi:

```
if (<shart>) <operator1>;
```

<shart> tekshirilishi lozim bo'lgan mantiqiy ifoda

<operator 1> Agar shart rost (true) qiymatga ega bo'lsa bajarilishi lozim bo'lgan operator.

<operator 2> Agar shart yolg'on (false) qiymatga ega bo'lsa bajarilishi lozim bo'lgan operator.

if ni qo'llaganimizda ifoda faqat shart haqiqat bo'lgandagina bajariladi, aks holda tashlanib o'tiladi. if/else yordamida esa shart bajarilmaganda (false natija chiqqanda) else orqali boshqa bir yo'ldan borishni belgilash mumkin. Misolimizni takomillashtirsak. Bola 7 yosh yoki undan katta bo'lsa maktabga, 7 dan kichkina bo'lsa bog'chaga borsin.

```
if (yosh >= 7)
```

```
maktab(); //nuqta-vergul majburiydir else
```

```
bog'cha();
```

Yuqorida if ga tegishli bo'lgan blok bitta ifodadan (maktab()) iborat. Shu sababli nuqta-vergul qo'yilishi shart. Buni aytib o'tishimizning sababi, masal Pascalda hech narsa qo'yilmasligi shart. C++ da bitta ifoda turgan joyga ifodalar guruhini {} qavslarda olingan holda qo'ysa bo'ladi. Masalan:

```
if (yosh >= 7){
```

```
cout << "Maktabga!\n"; maktab();
```

```
}
```

```
else{
```

```
cout << "Bog'chaga!\n"; bogcha();
```

```
}
```

Shart operatori tarkibida ixtiyoriy operatoridan foydalanish mumkin. Shu o'rinda Shart operatoridan ham.

C++ tili operatorlarni blok ko‘rinishida bo‘lishiga imkon beradi. Blok ‘{‘ va ‘}’ belgi oralig‘iga olingan operatorlar ketma-ketligi bo‘lib, u kompilyator tomonidan yaxlit bir operator deb qabul qilinadi. Blok ichida yangi o‘zgaruvchilarni ham e‘lon qilish mumkin. Bu o‘zgaruvchilar faqat blok ichida ko‘rinadi, undan tashqarida ko‘rinmaydi, ya‘ni blokdan tashqarida bu o‘zgaruvchilarni ishlatib bo‘lmaydi. Blokdan keyin nuqtali vergul qoyilmaydi, lekin blok ichida har bir operator nuqtali vergul bilan yakunlanishi shart.

Agar tekshirilayotgan shart nisbatan sodda bo‘lsa, shart amalini «?: » ko‘rinishini ishlatish mumkin. Bu operator quyidagi ko‘rinishga ega:

```
<shart ifoda> ? <ifoda1> : <ifoda2>;
```

if shart operatoriga o‘xshash holda bu shart amali quyidagicha ishlaydi: agar <shart ifoda> rost (true) bo‘lsa <ifoda1> bajariladi, aks holda <ifoda2>. Odatda ifodalar qiymatlari birorta o‘zgaruvchiga o‘zlashtiriladi.

Tanlash operatori

Boshqarishni uzatish operatorlaridan yana biri tanlash operatoridir. Tanlash operatori asosan bir nechta qiymatdan, o‘zgaruvchiga mos qiymatni tanlashda va qiymatlarga mos ravishda boshqarishni uzatishda ishlatiladi.

Tanlash operatorining umumiy ko‘rinishi:

```
switch (<o‘zgaruvchi>)
```

```
{
```

```
case <o‘zgarvas ifoda1> : <operator 1>; break; case <o‘zgarvas  
ifoda2> : <operator 2>; break;
```

```
...
```

```
case <o‘zgarvas ifodaN> : <operator N>; break; [default : operator N  
+ 1];
```

```
}
```

Tanlash operatorida boshqarilish o‘zgaruvchiga mos ravishda qiymatlarga uzatiladi va mos operator ishga tushadi. default operatori birorta ham qiymat o‘zgaruvchiga to‘g‘ri kelmasa ishlatiladi. default operatorini ishlatmasdan tashlab ketish ham mumkin.[8]

Eslatma: Dasturlashga doir kitoblarni o‘qiganingizda, biror operatorning umumiy ko‘rinishining to‘rtburchak qavs [] belgisi oralig‘ida yozilgan qismini ishlatmasdan tashlab ketish mumkin. Operatorning bu qismidan foydalanish ixtiyoriy bo‘ladi.

Takrorlanuvchi jarayonlarni dasturlash

Qator masalalarni yechish uchun ko‘pincha bitta amalni bir necha marotaba bajarish talab qilinadi. Amaliyotda bu rekursiyalar va iterativ algoritmlar yordamida amalga oshiriladi. Iterativ jarayonlar – bu operatsiyalar ketma-ketligini zaruriy sonda takrorlanishidir.

Dastur bajarilishini boshqarishning yana bir kuchli mexanizmlaridan biri – takrorlash operatorlari hisoblanadi.

Takrorlash operatori *«takrorlash sharti»* deb nomlanuvchi ifodaning rost qiymatida dasturning ma’lum bir qismidagi operatorlar (takrorlash tanasi)ning ko‘p marta takror ravishda bajarilishini amalga oshiradi.

Takrorlash o‘zining kirish va chiqish nuqtalariga ega bo‘lib, ba’zan chiqish nuqtasi bo‘lmasligi ham mumkin. Agar chiqish nuqtasi bo‘lmasa cheksiz takrorlanish sodir bo‘ladi. Cheksiz takrorlanish uchun takrorlashni davom ettirish sharti doimo rost bo‘ladi.

Takrorlash shartini tekshirish, takrorlanish tanasidagi operatorlarning bajarilishidan oldin tekshirilishi mumkin (*for*, *while* takrorlash operatorlarida) yoki takrorlanish tanasining operatorlari bir marta bajarilgandan keyin tekshirilishi mumkin (*do-while*).

Takrorlanishlarni tashkil etishda takrorlanishlar soni va qadami oldindan ma’lum bo‘lgan holatlar uchun *for* operatoridan, boshqa hollarda esa *while* hamda *do-while* operatorlaridan foydalanish maqsadga muvofiq.

C++ programmalashtirish tilida sikl operatorining bir necha xil turi mavjud.

- *for* sikl operatori
- *do.. while* sikl operatori
- *while* sikl operatori

Yechilayotgan masalaga qarab, programmist o‘zi uchun qulay bo‘lgan sikl operatoridan foydalanishi mumkin.

for takrorlash operatorining sintaksisi quyidagicha:

```
for (<ifoda1>; <ifoda2>; <ifoda3>)  
<operator yoki blok>;
```

Bu operator amal qilishni <ifoda1> bajarishdan boshlaydi. Keyin takrorlash qadamlari boshlanadi. Har bir qadamda <ifoda2> bajariladi, agar natija 0 dan farqli yoki rost (true) bo‘lsa, sikl tanasi - <operator yoki blok> bajariladi va oxirida <ifoda3> bajariladi, aks holda boshqaruv takrorlash operatoridan keyingi operatorga o‘tiladi. Sikl

tanasi – <operator yoki blok> sifatida bitta operator, shu jumladan bo'sh operator, yoki operatorlar bloki kelishi mumkin. [7]

Sikl takrorlanishi davomida bajarilishi lozim bo'lgan operatorlar majmuasi sikl tanasi deyiladi. Sikl tanasi sifatida bir yoki bir nechta operatorlardan foydalanish mumkin.

Agar sikl tanasida bir nechta operatorlardan foydalanmoqchi bo'lsak bu operatorlarni blok {
} orasiga olishimiz kerak.

for sikl operatorining boshqa imkoniyatlari

for sikl operatorida qavs ichidagi ifodalar bo'lmasligi mumkin, lekin ";" bo'lishi shart. Eng sodda doimiy takrorlanuvchi sikl operatori quyidagicha:

```
for ( ; ; )
```

```
cout << "doimiy takrorlanish";
```

Agar takrorlash jarayonida bir nechta o'zgaruvchi bir vaqtda sinxron o'zgarishi lozim bo'lsa, ularni <ifoda1> va <ifoda3> da zarur bo'lgan o'rinda vergul bilan ajratib yozish mumkin.

```
#include <iostream> using namespace std; int main()
```

```
{
```

```
int n;
```

```
cin >> n;
```

```
for (int i = 1, j = 1; i <= n; i++, j += i) cout << i << " " << j << endl;
```

```
return 0;
```

```
}
```

Shuni ham ta'kidlash joizki, C++ tilida *for* operatorining parametrlari haqiqiy turga tegishli bo'lishi, takrorlanishlar qadami ham ixtiyoriy holatda belgilanishi mumkin. Bu imkoniyatlar operatorning keng qamrovli ekanligini bildiradi.

do - while operatorining umumiy ko'rinishi: do {

operator;

} while (shart);

Bu yerda do va while xizmatchi soʻzlar. (shart) sikl tanasi bajarilgandan soʻng, sikldan chiqish uchun tekshiriladigan shart. (mantiqiy ifoda).

do - while operatorning ishlash tartibi:

do xizmatchi soʻzidan keyingi operatorlar bajariladi, keyin while xizmatchi soʻzidan keyingi shart tekshiriladi. Agar shart rost (true) natija bersa do xizmatchi soʻzidan keyingi operatorlar qayta bajariladi. Shart qayta tekshiriladi, bu jarayon shart yolgʻon (false) natija berguncha takrorlanadi. Qachon while xizmatchi soʻzidan keyingi shart yolgʻon (false) qiymatga ega boʻlsa, boshqarilish do - while operatoridan keyingi operatorga uzatiladi.

while operatorining umumiy koʻrinishi:

```
while ( shart ) { sikl_tanasi;  
}
```

while operatori sikl tanasida qanday operatorlar boʻlishi mumkin?

sikl_tanasi ixtiyoriy operator yoki operatorlar majmuidan iborat boʻlishi mumkin. while sikl operatorning ishlash tartibi:

Agar (shart) rost (true) qiymatga ega boʻlsa, sikl_tanasi bajariladi. Qachonki shart yolgʻon (false) qiymatga teng boʻlsa sikl tugatiladi.

Agar (shart) true qiymatga ega boʻlmasa sikl tanasi biror marta ham bajarilmaydi. while sikl operatoridan qanday chiqiladi?

while sikl operatoridan, qachon (shart) false (yolgʻon) qiymat qabul qilsa chiqiladi. Yaʼni boshqarilish while operatoridan keyingi operatorga uzatiladi. Agar (shart) false qiymat qabul qilmasa, while sikl operatoridan chiqib ketilmaydi va bu jarayon sikllanib qolish deyiladi.

Programmash sanʼati. do - while va while sikl operatorlarida sikl tanasi sifatida faqat bitta operator ishlatiladigan boʻlsa, bu operatorni blok orasiga { } olmasdan ham yozish mumkin. Lekin professional programmistlar har qanday holda sikl tanasini blokka { } olib yozishni tavsiya qilishadi. Bu esa sodir boʻlishi mumkin boʻlgan mantiqiy xatoliklarning oldini oladi.

Nazorat savollari:

1. Mantiqiy turga tegishli bo'lgan o'zgaruvchilar qanday qiymatlarni qabul qiladi?
2. C++ tilida mantiqiy tur kimni sharafiga *bool* kalit so'zi bilan ifodalangan?
3. "*Bul algebrasi*" da mantiqiy mulohazalar ustida nechta amal aniqlangan va ular qaysilar?
4. Shart operatorining nechta ko'rinishi mavjud?
5. Takrorlash operatorini qanday turlari bor?
6. Tanlash operatorini qanday ishlatiladi?
7. Ma'lumotlarni qanday turlari bor?

GLOSSARIY

Arxitektura Dizayni: Dasturiy ta'minot tizimining tuzilishi va xatti-harakatlarini aniqlash jarayoni, shu jumladan tizim komponentlari va ularning o'zaro bog'liqligini belgilash.

Abstraktsiya: Murakkablikni kamaytirish uchun tizimning yuqori darajadagi ko'rinishini yaratish usuli.

Biznes Mantiq Qatlami: Tizimning biznes qoidalari va jarayonlarini amalga oshiradigan qatlami.

Dasturiy Ta'minot Arxitekturas: Tizimning yuqori darajadagi tuzilishi, komponentlari va ularning o'zaro bog'liqligi hamda xatti-harakatlarini aniqlovchi tamoyillar va qoidalar.

Ierarxiya: Modullarni qatlamlarga bo'lish orqali tizimni boshqarish va kengaytirish osonligini ta'minlash.

Kapsulyatsiya: Har bir modul o'zining ma'lumotlari va funksiyalarini yashirish orqali boshqa modullardan mustaqil ishlashini ta'minlash.

Modullik: Tizimni mustaqil ishlaydigan modullarga bo'lish orqali murakkablikni boshqarish tamoyili.

Monolit Arxitektura: Tizimning barcha funksiyalari bir joyda jamlangan yagona kod bazasida amalga oshiriladigan arxitektura uslubi.

Mikroxizmatlar Arxitekturas: Tizim kichik, mustaqil xizmatlar to'plamidan tashkil topgan arxitektura uslubi bo'lib, har biri o'z ma'lum bir vazifani bajaradi.

Qatlamli Arxitektura: Tizim turli qatlamlarga bo'lingan bo'lib, har bir qatlam ma'lum bir funktsiyani bajaradi, masalan, taqdimot qatlami, biznes mantiq qatlami va ma'lumotlar qatlami.

Tizimli Arxitektura: Tizim turli qatlamlarga bo'lingan bo'lib, har bir qatlam ma'lum bir funktsiyani bajaradi.

Uzluksizlik: Tizimning vaqt o'tishi bilan ham barqaror ishlash va o'zgartirishlarga moslasha olish qobiliyati.

FOYDALANILGAN ADABIYOTLAR RO‘YXATI

1. Bass, L., Clements, P., & Kazman, R. (2003). *Software Architecture in Practice*. Addison-Wesley Professional.
2. Garlan, D., & Shaw, M. (1993). *An Introduction to Software Architecture*. World Scientific Publishing.
3. Richter, J. (2006). *CLR via C#*. Microsoft Press.
4. Fowler, M. (2002). *Patterns of Enterprise Application Architecture*. Addison-Wesley.
5. Evans, E. (2003). *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Addison-Wesley.
6. Rozanski, N., & Woods, E. (2011). *Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives*. Addison-Wesley.
7. Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley.
8. Brown, N., & Booch, G. (1998). *Inside the Unified Modeling Language*. Addison-Wesley.
9. Taylor, R. N., Medvidovic, N., & Dashofy, E. M. (2009). *Software Architecture: Foundations, Theory, and Practice*. Wiley.
10. Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., & Stal, M. (1996). *Pattern-Oriented Software Architecture Volume 1: A System of Patterns*. Wiley.

U.Y. Saidov, B.I. Navruzov, H.U. Sultonov

DASTURIY TA'MINOT

ARXITEKTURASI

O'QUV QO'LLANMA

Muharrir:

E.Eshov

Tex. muharrir:

D.Abduraxmonova

Musahhih:

M.Shodiyeva

Badiiy rahbar:

M.Sattorov

Nashriyot litsenziyasi № 022853. 04.03.2022.

Original maketdan bosishga ruxsat etildi: 11.07.2024. Bichimi 60x84. Kegli 16 shponli. "Times New Roman" garnitura 1/16.

Elektrografik usulda. Oddiy bosma qog'ozi.

Bosma tabog'i 12. Adadi 100. Buyurtma № 422



KAMOLOT

“BUXORO DETERMINANTI” MCHJ

bosmaxonasida chop etildi.

Buxoro shahar Namozgoh ko'chasi 24 uy

Tel.: + 998 91 310 27 22