

**Ахборот тизимларининг
фойдаланиш даври (яшаш
ёки хаёт цикли) модели.**

Ахборот тизимларининг архитектураси

Ахборот тизимларининг архитектураси

Ахборот тизимлари асосан икки хил архитектурада
қурилади. Улар қуйидагилардан иборат:

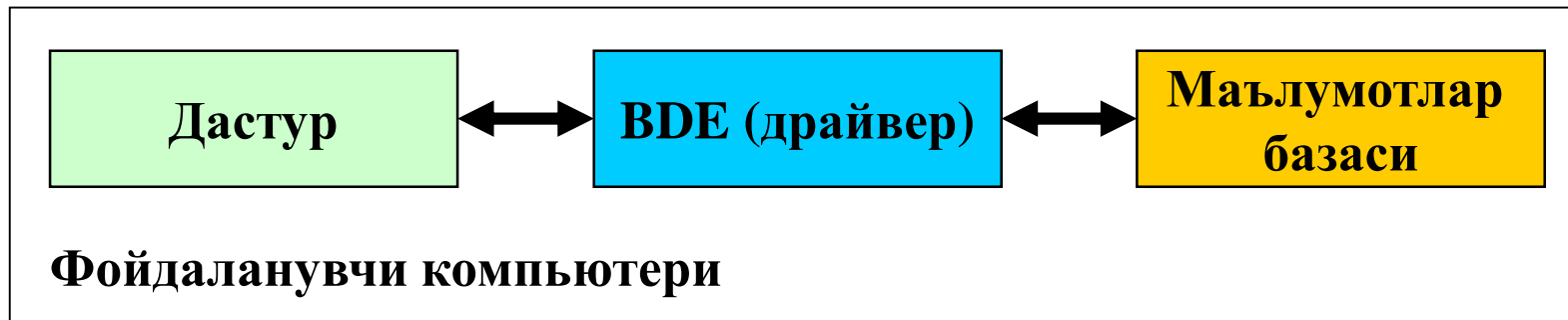
- локал маълумотлар базали
ахборот тизимлари;
- глобал маълумотлар базали
ахборот тизимлари.

Локаль архитектура

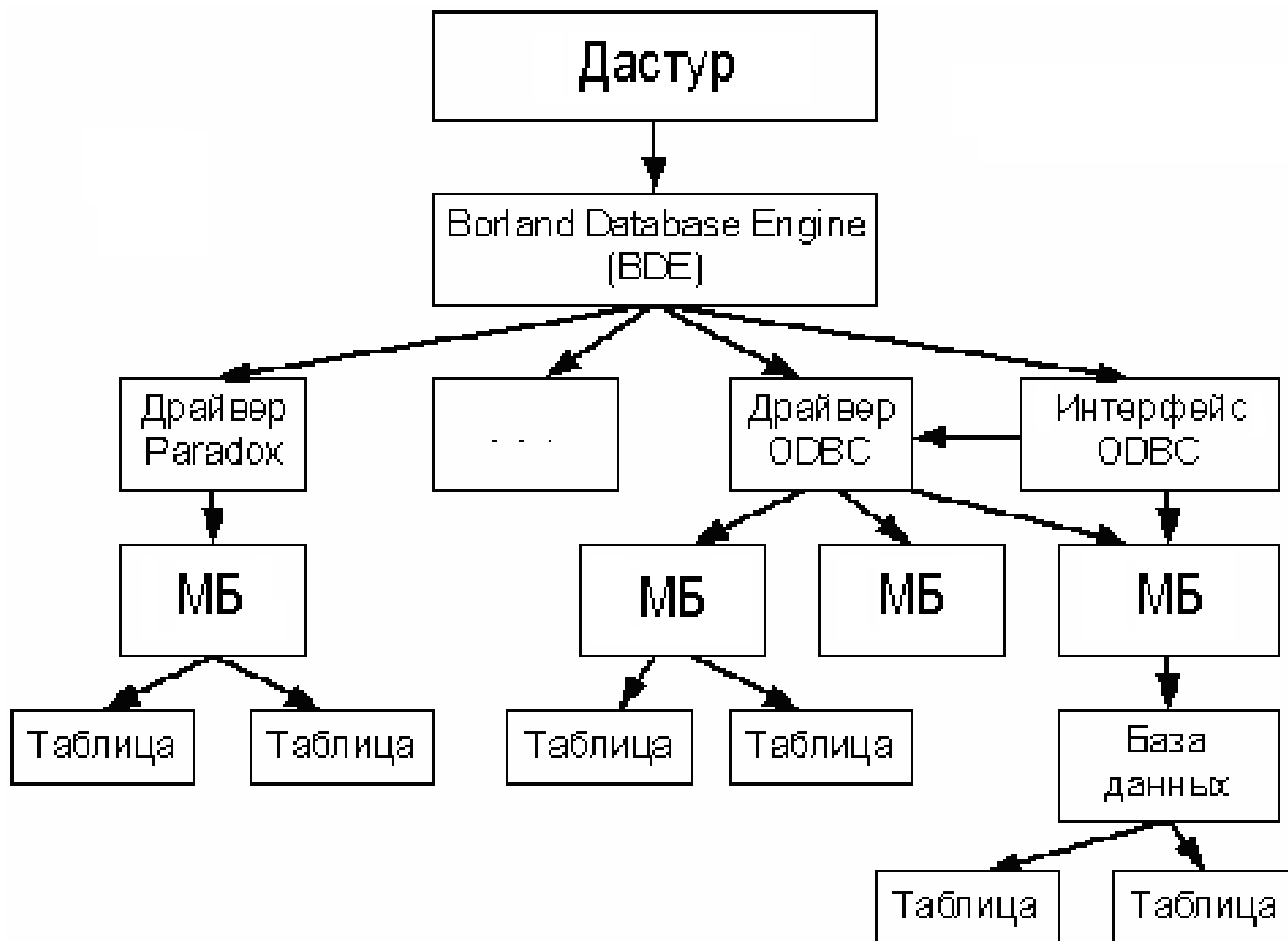
Локал маълумотлар базаси (МБ) ундан фойдаланаётган дастур жойлашган компьютернинг узида жойлашган булади.

Бундай холларда ахборот тизимлари **локал архитектурада** курилган деб юритилади.

МБ билан ишлаш **бир фойдаланувчили** режимда амалга оширилади.



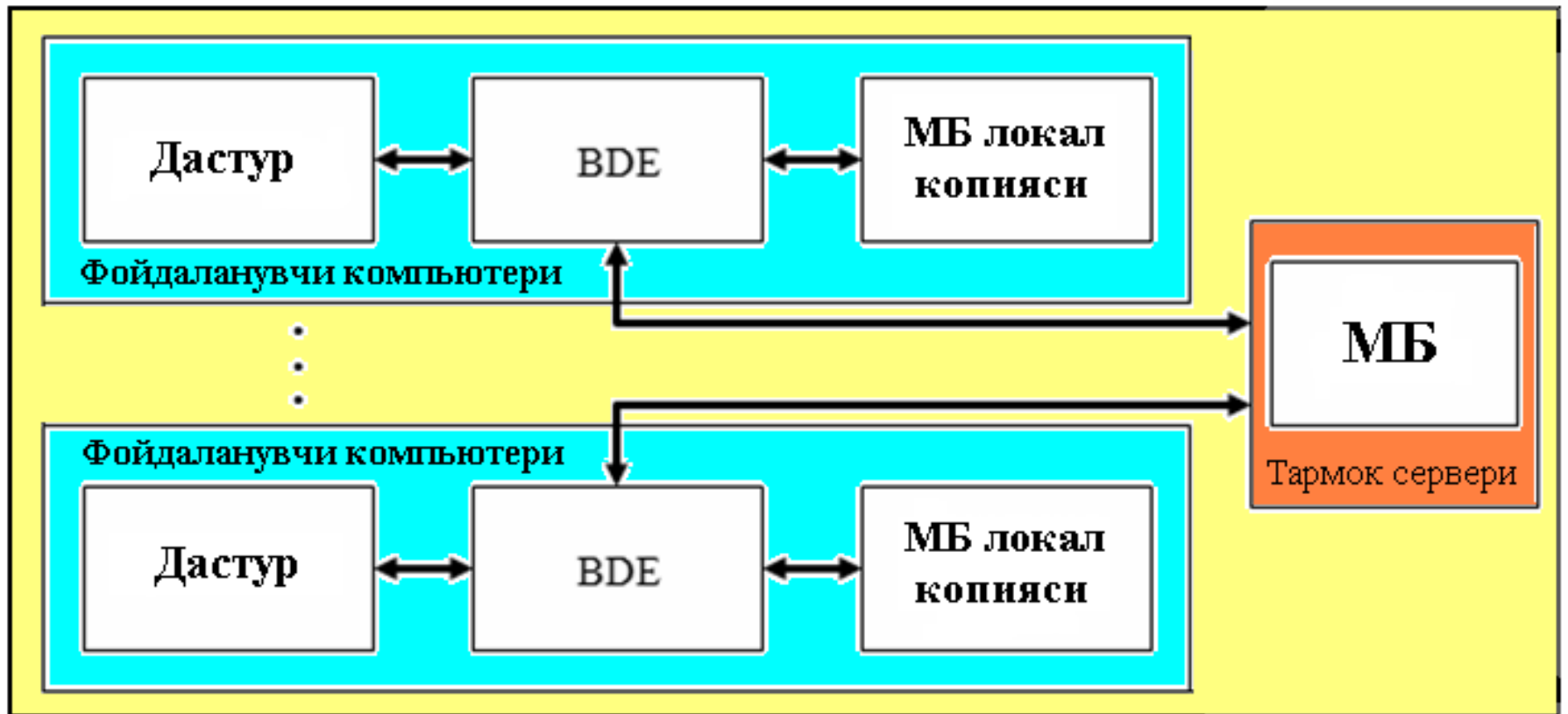
Локал архитектурали ахборот тизимларининг структураси



АТ «Файл-сервер» архитектураси

- «Файл-сервер» архитектураси купинча фойдаланувчилари сони куп булмаган компьютер тармокларида кулланилади. Бундай тизимни жорий килиш учун Paradox ва dBase туридаги локал маълумотлар базасидан фойдаланилади.
- Бундай архитектурани афзаллиги шундан иборатки, уни жорий килиш жуда оддий булиб, фойдаланиладиган дастурлар компьютер тармогига боғлиқ булмай фақат битта фойдаланувчига мулжаллаб яратилади.

«Файл-сервер» архитектура



«Файл-сервер» архитектура камчилликлари

- «Файл-сервер» архитектура куйидаги камчилликларга эга:
 - Фойдаланувчи хар доим узининг МБ локал копияси билан ишлайди. Агар бирон бир жадвалдаги маълумотларни узгартириш учун суров жунатса, у холда сервер унга талаб килинган МБ янги локал копиясини узатади. Бундай холларда тармоқда фойдаланувчилар ва сервер уртасида катта хажмдаги маълумотларни айланиши хисобидан **тармоқ юкламаси** ошиб кетади ва ахборот тизимини ишлаш **тезлиги** пасаяди.
 - Хар бир компьютерда узини маълумотлар базасини локал копияси булганлиги учун шу копияда бажарилган **узгартиришлар** биров вақт бошка фойдаланувчиларга маълум булмайди.

Масофадаги МБ

Масофадаги МБ тармок сервери компютерида жойлаштирилади, МБ билан ишловчи дастурлар эса фойдаланувчи компютерларида жойлашган булади.

Бундай холларда биз «**клиент-сервер**» архитектураси билан ишлаймиз. Бунда ахборот тизими **сервер** ва **клиент** қисмларга ажратилади. Сервер компютери клиентлардан алоҳида жойлаштирилади. Клиентлардан алоҳида жойлаштирилган серверлар масофавий **серверлар** ҳам деб юритилади.

Клиент - эса бу фойдаланувчи дастурдир.

Клиент бирор маълумотни олиш учун маълумотлар базаси жойлашган масофавий серверга суров жунатади. **Суровлар** – реляцион маълумотлардан фойдаланилганда серверларга мурожат қилишни стандарт воситаси булган **SQL (Structured Query Language)** тилида яратилади. Суров қабул қилингандан кейин сервер уни SQL серверга йуналтиради.

Маълумотлар базаси сервери – бу масофавий МБ бошқарувчи, суровларни бажарилишини ва суров натижаларни клиентга узатилишини таъминловчи махсус дастурдир.

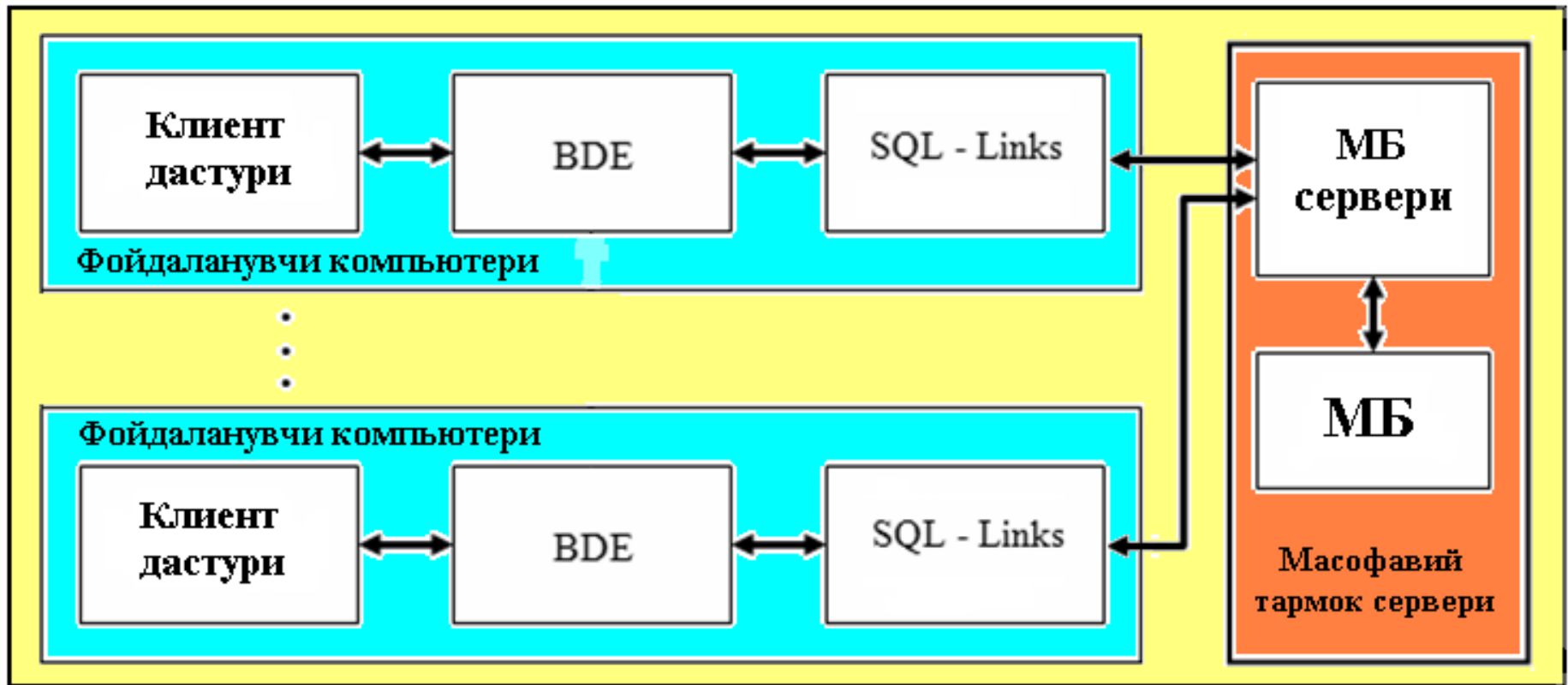
«Клиент-сервер» архитектура

«Клиент-сервер» архитектурасида клиент бирор маълумотни олиш учун суров жунатади ва факат суровда талаб килиган маълумотларни олади. Барча суровларни кайта ишлаш масофавий серверда амалга оширилади.

«Клиент-сервер» архитектураси куйидаги афзалликларга эга:

- **Тармок юкламасини пасайиши**, энди факат керакли маълумотлар алмашинади.
- **Ахборот хавфсизлигини ошади**, чунки барча клиентларни суровларини кайта ишлаш серверда жойлашган ягона дастур оркали амалга оширилади. Сервер барча фойдаланувчилар учун МБ дан фойдаланишни умумий коидасини урнатади, клиентларни маълумотларга булган мурожат режимларини бошкаради, хар хил фойдаланувчилар томонидан битта маълумотни баравар узгартирилишини чеклайди.
- **Клиент дастурларини мураккаблиги камаяди**, чунки уларда МБ ни бошкарувчи ва назорат килувчи дастур кодлари мавжуд булмайди.

Икки сатхли «клиент-сервер» архитектурси



Катта (толстый) клиент

- Delphi дастурларни маълумотлар базаларини бошқариш тизимларига (МББТ) боғланиши **SQL-Links** драйверлари орқали амалга оширилади.
- Келтирилган архитектура **икки сатхли** хисобланади, яъни клиент дастури ва маълумотлар базаси серверидан иборат
- Клиент дастури «кучли» ёки «калин» клиент деб аталади.

Уч сатхли «клиент-сервер» архитектурси



Кичик (тонкий) клиент

- Уч сатхли архитектурада маълумотларга мурожат килиш ва уларни кайта ишлаш воситаларини ва дастур кодларини бир кисми клиент дастурларидан ажратилиб дастурлар серверига юклатилади.
- Клиент дастурлари бунда «кучсиз» ёки «кичик» клиент деб юритилади.
- Дастурлар серверида барча клиентлар дастурлари учун умумий булган воситалар ва дастур кодларини жойлаштириш кулайдир.

Уч сатхли архитектуранин афзалликлари

Уч сатхли «клиент-сервер» архитектурасинин асосий афзалликларини куйидагилардан иборат:

- МБ серверинин юкламасини бир кимс операцияларнин бажарилишини дастурлар серверига утказилганли билан камайтирилган;
- Клиент дастурларини улчамини камайдн ва соддалашадин (ортикча кодларнин олиб ташланишини хисобидан);
- Барча клиентлар учун ягона муносабат;
- Клиентлар созлашларинин соддалашувини (дастурлар серверидаги умумий коднин узгаришини билан автоматик равишда барча клиент дастурларинин муносабати узгарадин).

Эътиборларингиз учун
рахмат

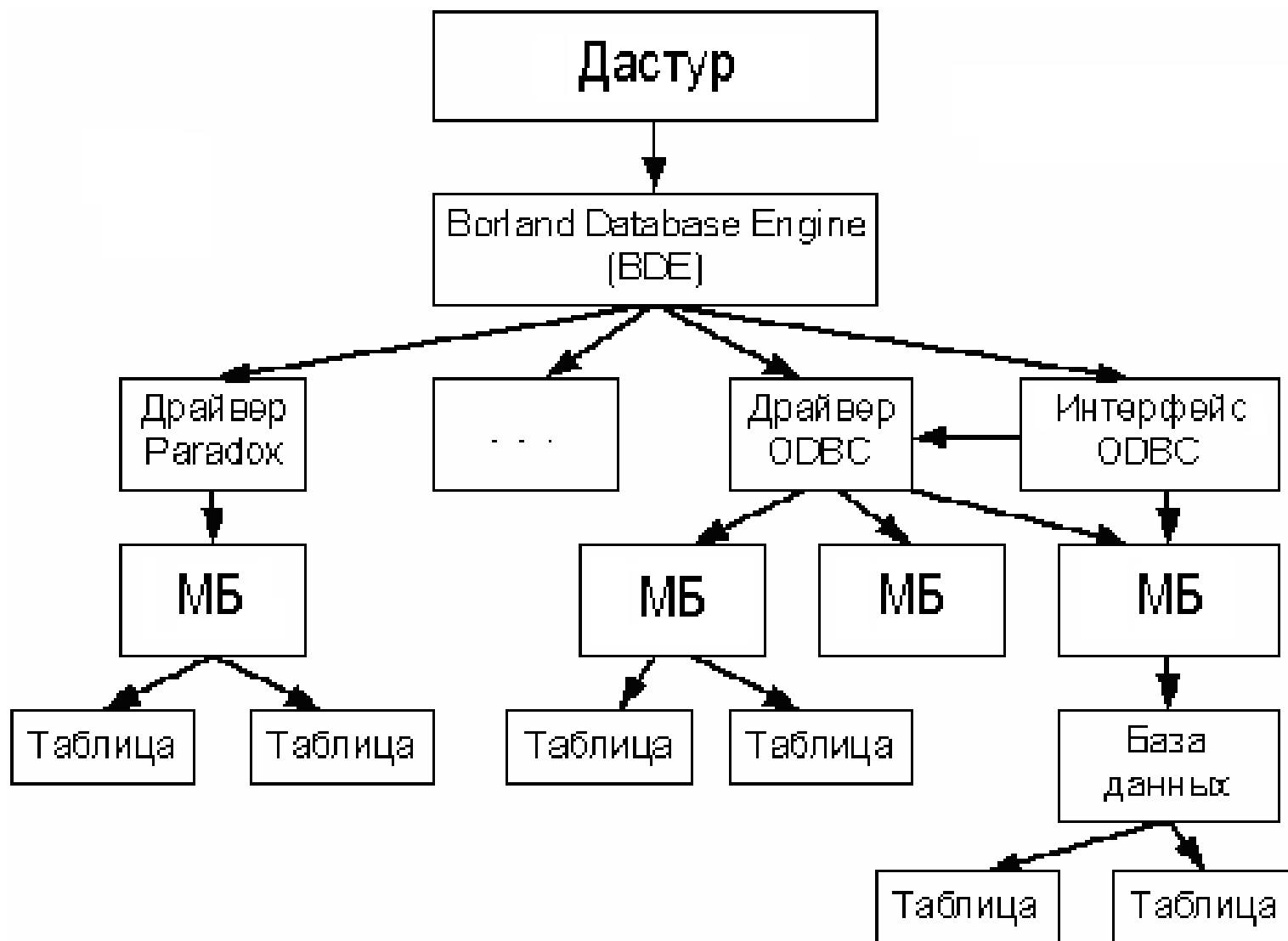
РЕЖА

1. Ахборот тизимини ишлаб чиқиш.
2. Ахборот тизимини ишга тушириш
3. Ахборот тизимини кузатиб бориш
4. Фойдаланиш даври моделлари

Берилган давлат стандартига мувофиқ ахборот тизимининг
яшаш(фаолият кўрсатиш) даври таркиби уч гуруҳ
жараёнларига асосланади:

- ✓ Яшаш даврининг асосий жараёни (ўзлаштириш, етказиб бериш, ишлаб чиқиш, фойдаланиш, кузатиб бориш);
- ✓ асосий жараёнларни бажарилишини таъминлаб берувчи ёрдамчи жараёнлар (хужжатлаштириш, конфигурацияни **(тузилишини)** бошқариш, сифатини таъминлаш, верификация **(таққослаш)**, аттестация, баҳолаш, аудит, муаммоларни ҳал қилиш);
- ✓ Ташкиллаштириш жараёнлари (лойихани бошқариш, лойихани инфраструктурасини яратиш, фойдаланиш даври аниқлаш, баҳолаш ва яхшилаш, ўқитиш).

Локал архитектурали ахборот тизимларининг структураси



Ахборотлашган дастурий таъминотни ишлаб чиқариш жараёнига куйидагилар киради:

- ✓ «лойихалаш ва эксплуатация;
- ✓ хужжатларни ишлаб чиқиш (оформление)
- ✓ дастурий махсулотларни тестдан ўтказиш учун зарур материалларни тайёрлаш;
- ✓ персоналларни ўқитишни ташкиллаштириш учун зарур материалларни ишлаб чиқиш.

Эксплуатация жараёнини тайёрлов ва асосий қисмга бўлиш мумкин.

Тайёрлов ишларига куйидагилар киради:

- ✓ маълумотлар базасини ва фойдаланувчилар иш жойларини конфигурациялаш (шакллантириш);
- ✓ хизматчиларни (персонал) ўқитиш;

Асосий қисм ишлари ўз ичига куйидагиларни олади:

- ✓ ишлатиш;
- ✓ муаммоларни бўлаклаш(локализациялаш) ва уларнинг келиб-чиқиш сабабларини аниқлаш ва бартараф этиш чораларини кўриш;
- ✓ дастурий таъминотни ўзгартириш; (модификация)
- ✓ тизимни мукамаллаштириш бўйича таклифлар тайёрлаш;
- ✓ тизимни ривожлантириш ва модернизациялаш.

Лойиханинг техник ва ташкилий таъминоти қуйидагилардан ташкил топган:

- ✓ лойихани амалга оширишда усул ва услубларни танлаш;
- ✓ оралиқ ҳолатларни ишлаб чиқиш усуллари аниқлаш;
- ✓ яратилган дастурий таъминотнинг синовдан ўтказиш усул ва услубларини ишлаб чиқиш;
- ✓ **персонални ўқитиш.**

Фойдаланиш даври модели

Фойдаланиш даври модели ахборот тизимининг хусусияти (**спецификаси**) ва имкониятларига боғлиқ. Ҳозирги вақтда куйидаги асосий фойдаланиш даври моделлари кенг тарқалган:

Масалалар (**ёки масалали моделлар**) модели;

Каскад модели (**ёки тизимли**) (70-85й.й.);

Спиралли модел (ҳозирги замон).

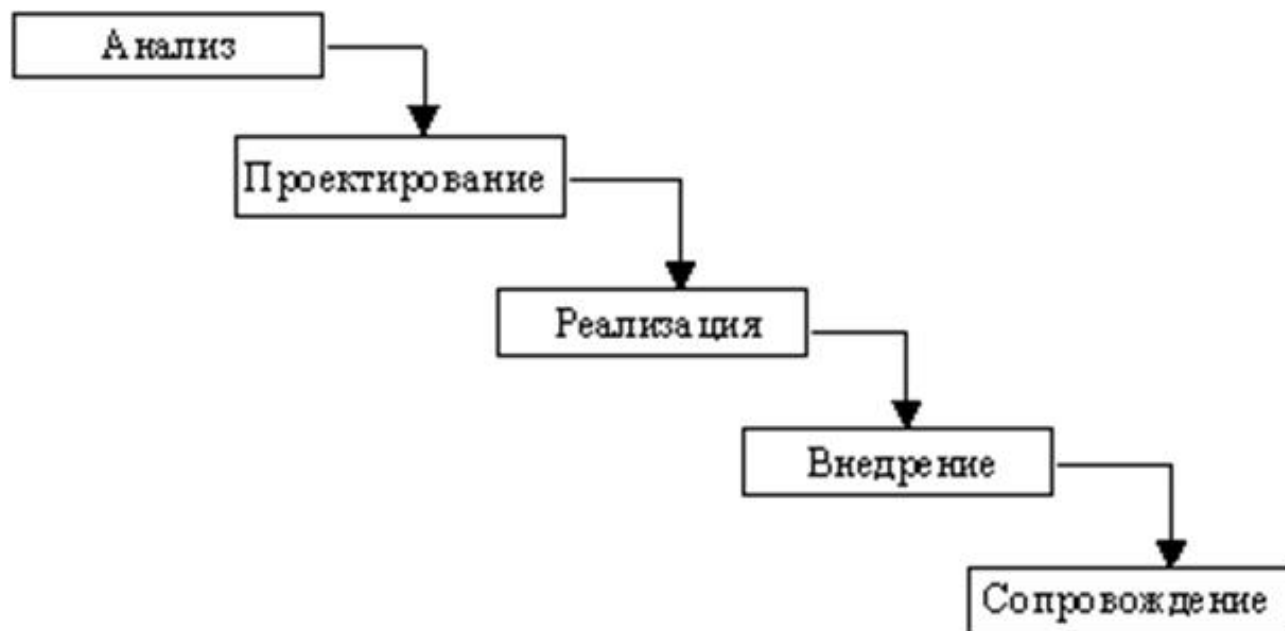
Масалали модел:

“Пастдан-тепага” масала модели ишлаб чиқилганда, алохида масалалардан умумий тизимга, ишланмага ягона ёндашув бўлмайди, алохида элементларни ахборотга боғланишида муаммолар пайдо бўлади. Коида бўйича (одатда) масалалар сони ўсиши билан қийинчиликлар ошиб боради, мавжуд дастур ва маълумотлар структурасига ҳар доим ўзгартиришлар киритиш керак бўлади

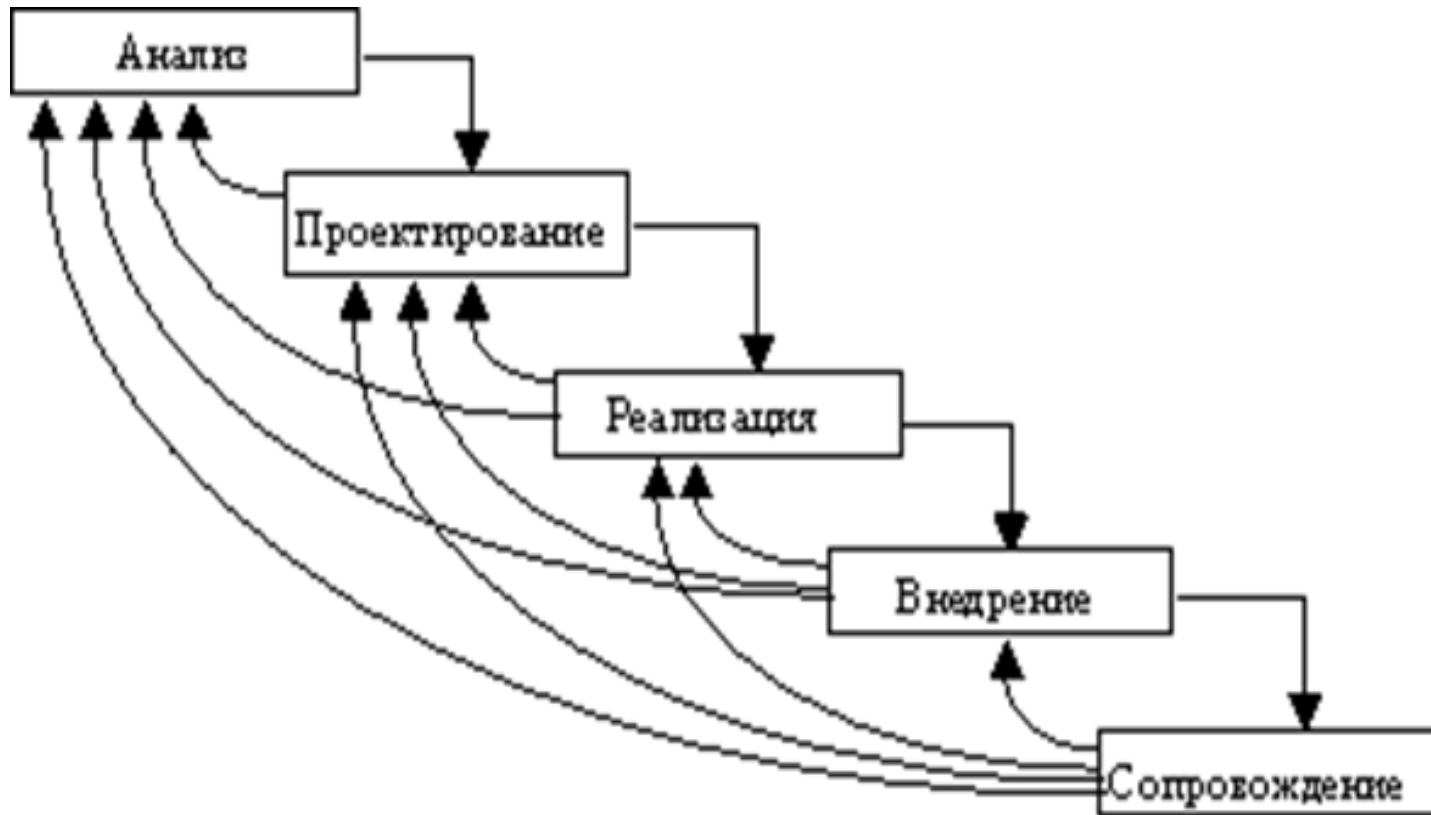
Каскад модели.

Олдинги, ҳажми унча катта бўлмаган бир туркумли ахборот тизимларнинг ҳар бир дастури яхлитликни ўзида акс этар эди. Бундай дастурларни ишлаб чиқишда каскад усули қўлланилар эди. Унинг асосий тавсифи – ишлаб чиқаришни босқичларга бўлиниши, бунда бир босқичдан кейингисига ўтиш, жорий босқичда иш якунлангандан сўнг амалга оширилади.

Каскад ёндашувининг афзалликлари:
Ҳар бир босқичда тугалланган лойиҳавий
хужжатлар йиғиндиси шаклланади.
Мантиқий кетма-кетликда бажариладиган иш барча
босқичларда ишларни якунлаш вақтигача
харажатларни мос равишда режалаштириш
имконини беради.

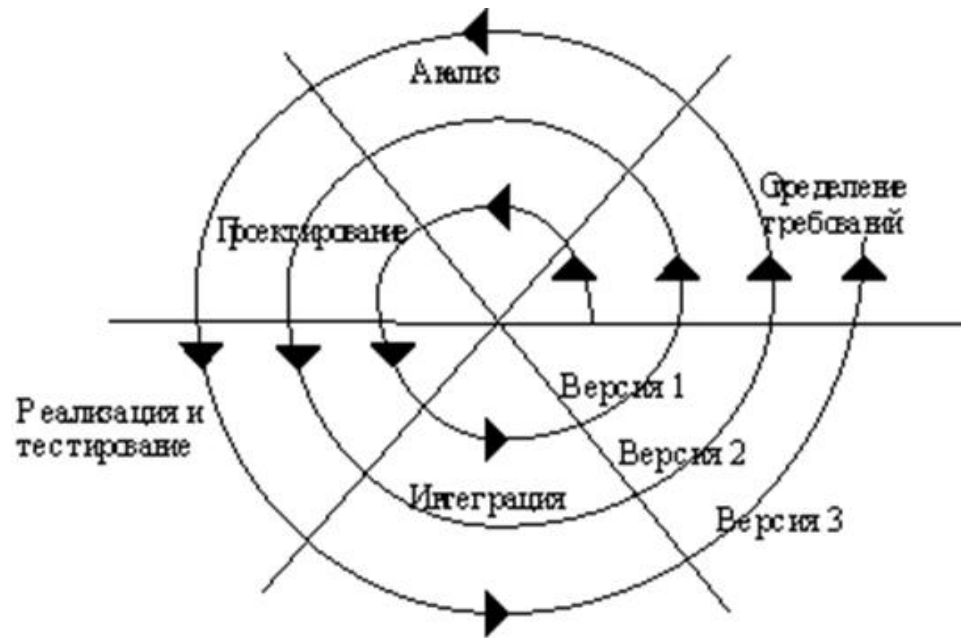


Уч сатхли «клиент-сервер» архитектурси



Спиралли модел

Кўриб чиқилган муаммоларни бартараф этиш учун фойдаланиш даврининг спиралли модели (рис.3) тавсия этилди.



Фойдаланиш даврининг спиралли модели доирасида дастурий таъминот ишланмасига мумкин бўлган ёндашувлардан бири бўлиб ҳозирги кунда кенг тарқалган дастурни тезкор ишлаб чиқиш методологияси RAD (Rapid Application Development Тезкор дастур ишлаб чиқиш) ҳисобланади

Бу атама остида одатда 3 та элементдан ташкил топган дастурий таъминотни ишлаб чиқиш жараёни тушунилади:

- ✓ кичик дастурчалар гуруҳи (2-10гача);
- ✓ қисқа ва аниқ тузилган ишлаб чиқариш графиги (2-6 ой);
- ✓ такрорланувчи цикл.

RAD методологияси бўйича дастурий таъминотнинг фойдаланиш даври 4 та бўлинмадан (фазадан) ташкил топган:

- 1) талаб ва таклифни аниқлайдиган бўлинма;
- 2) лойиҳалаштириш бўлинмаси;
- 3) тадбиқ этиш бўлинмаси.
- 4) Фойдаланиш бўлинмаси

Эътиборларингиз учун
рахмат