
Programų vystymas

Tikslai

- Paaiškinti kodėl pakeitimai neišvengiami, kad sistema išliktų naudinga
- Aptarti programų palaikymą ir palaikymo kainos faktorius
- Aprašyti programų vystymo procesus
- Aptarti liktinių sistemų vystymo strategijos vertinimo metodus

Temos

- Programų vystymo dinamika
- Programų palaikymas
- Vystymo procesai
- Paveldėtų sistemų vystymas

Programinės įrangos keitimas

- Programinę įrangą neišvengiamai tenka keisti, jei:
 - Naudojant programinę įrangą išskyla naujų reikalavimų;
 - Pasikeičia verslo aplinka;
 - Tenka taisyti klaidas;
 - Sistemoje pradedami naudoti nauji kompiuteriai bei įranga;
 - Reikia gerinti veikimo spartą arba patikimumą.
- Pagrindinė organizacijų problema yra esamų PĮ sistemų pakeitimų realizavimas bei valdymas.

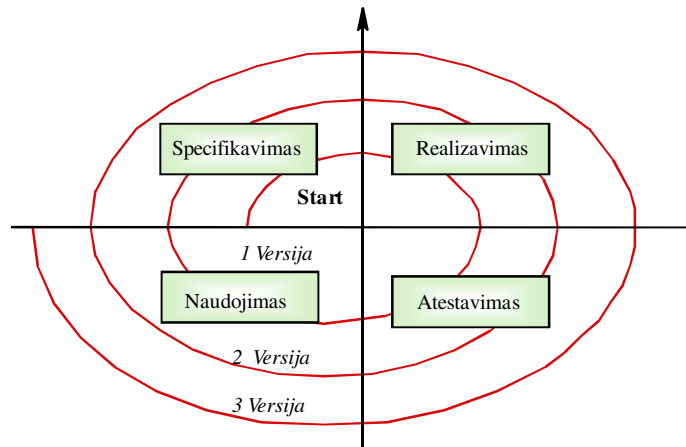
Vystymo svarba

- Organizacijos daug investavo į programų sistemas, kurios yra esminis verslo turtas.
- Kad išlaikyti turimų programų vertę jos turi būti keičiamos ir atnaujinamos.
- Didelėse kompanijose pagrindinis programinės įrangos biudžetas daugiau skirtas turimų programų plėtojimui negu naujų kūrimui.

Programinės įrangos keitimo strategijos

- Programinės įrangos palaikymas:
 - Dėl pasikeitusių reikalavimų sistemą tenka atnaujinti. Bendra programos struktūra nekeičiama.
- Architektūros keitimas:
 - Pavyzdžiui architektūra keičiama iš centralizuotos į paskirstytą..
- Programinės įrangos perdarymas:
 - Sistemai nepridedamos naujos funkcijos, tačiau jos struktūra keičiama siekiant palengvinti ateityje planuojamus atnaujinimus.
- Šios strategijos gali būti taikomos atskirai arba kartu.

Spiralinis vystymo modelis



Temos

- Programų vystymo dinamika
- Programų priežiūra
- Vystymo procesai
- Paveldėtų sistemų vystymas

Programų evoliucijos dinamika

- Programų evoliucijos dinamika nagrinėja sistemų keitimo procesus.
- Po reikšmingų bandymais paremtų tyrinėjimų, Lehmanas ir Belady nustatė, kad egzistuoja keletas evoliucijos “dėsnių”, bendrų daugeliui sistemų.
- Greičiau tai yra protingi pastebėjimai, o ne dėsniai. Jie tinka didelėms sistemoms, kurias sukūrė didelės organizacijos. Kitais atvejais jie nelabai tinka.

Lehmano dėsniai

Dėsnis	Aprašymas
Nuolatinis keitimasis	Realioje aplinkoje naudojama programa turi būti atnaujinama, nes priešingu atveju jos naudingumas toje aplinkoje vis mažės.
Didėjantis sudėtingumas	Programą pastoviai atnaujinant, jos struktūra tampa vis sudėtingesnė. Norint, kad struktūra išliktų paprasta arba ją supaprastinti, reikia papildomų investicijų.
Didelės programos evoliucija	Programos evoliucija yra savireguliuojantis procesas. Tokie sistemos atributai kaip dydis, pastebėtų klaidų skaičius ir laikas tarp naujų versijų išleidimo, beveik nekinta.
Organizacinis stabilumas	Per programos gyvavimo laiką, kūrimo sparta yra daugmaž pastovi, ir nepriklauso nuo sistemos kūrimui skiriamų papildomų resursų.
Konservatyvumo supratimas	Per sistemos gyvavimo laiką visi atnaujinimai būna panašios apimties.

Lehmano dėsnų taikomumas

- Taikomumas dar galutinai neapibrėžtas.
- Jie labiausiai pritaikomi didelėms sistemoms, kurias kuria didelės organizacijos.
- Neaišku kaip šiuos dėsnius modifikuoti:
 - Gaubiančiam (Shrink-wrapped) programinei įrangai;
 - Sistemoms, kurios integruoja daugelį COTS <“pirktas pilnai pagamintas, nuo lentynos”> komponentų;
 - Mažoms organizacijoms;
 - Vidutinio dydžio sistemoms.

Temos

- Programų vystymo dinamika
- Programų priežiūra
- Vystymo procesai
- Likitinių sistemų vystymas

Programinės įrangos palaikymas

- Programinės įrangos modifikavimas po to, kai ji buvo pateikta naudojimui.
- Palaikymo procedūrų metu dideli sistemos architektūros pakeitimai dažniausiai nedaromi.
- Atnaujinimas realizuojamas arba modifikuojant jau egzistuojančius sistemos komponentus, arba pridedant naujus.

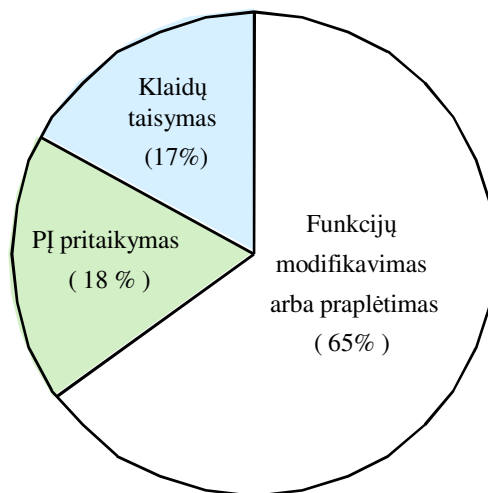
Palaikymas yra neišvengiamas

- Tikėtina, kad reikalavimai sistemai keisis jau sistemos kūrimo metu, nes keičiasi aplinka. Todėl net ir naujai pateikta sistema neatitiks reikalavimų!
- Sistema yra tvirtai susieta su savo aplinka. Į aplinką įdiegiama sistema pakeičia tą aplinką, todėl keičiasi ir reikalavimai sistemai.
- Jei norima, kad sistemos naudingumas duotoje aplinkoje nesumažėtų, ji TURI būti palaikoma.

Palaikymo tipai

- Programinės įrangos klaidų taisymas
 - Sistemos trūkumų taisymas, kad ji atitiktų reikalavimus.
- Programinės įrangos pritaikymas skirtingai darbo aplinkai
 - Toks sistemos atnaujinimas, kad ji veiktų ir kitoje aplinkoje (kitame kompiuteryje, OS ir t.t.), nei buvo įdiegta iš pradžių.
- Sistemos funkcijų modifikavimas arba praplėtimas
 - Sistemos atnaujinimas siekiant patenkinti naujus reikalavimus.

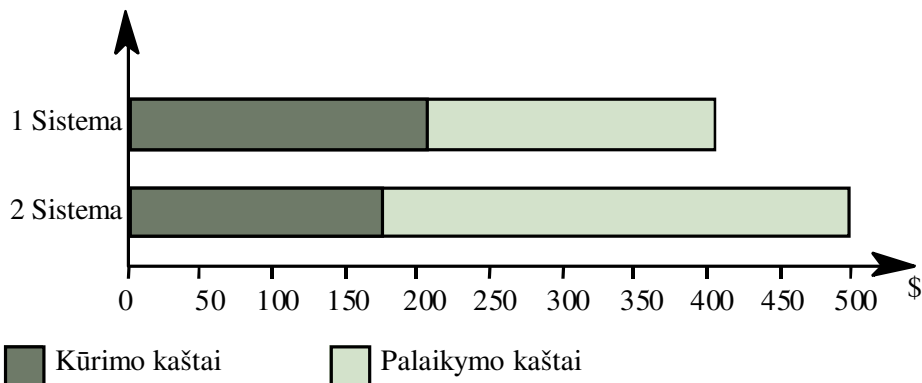
Palaikymo pastangų pasiskirstymas



Palaikymo kaštai

- Paprastai būna didesni nei kūrimo kaštai (nuo 2 iki 100 kartų, priklausomai nuo taikymo srities)
- Įtakojami tiek techninių, tiek ir netechninių veiksnių.
- Palaikymo procedūros gadina programinės įrangos struktūrą, taigi tolimesnis palaikymas sudėtingėja, o kaštai didėja.
- Senstant programinei įrangai, jos palaikymo kaštai didėja (nes naudojamos senos kalbos, kompiliatoriai ir t.t.).

Programų kūrimo ir palaikymo kaštų santykis



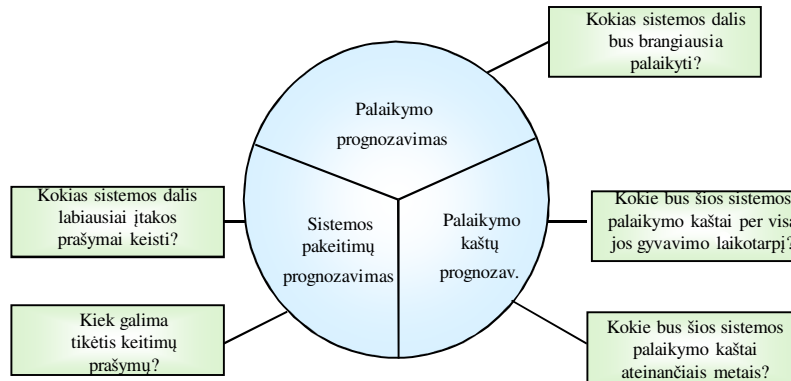
Palaikymo kaštų veiksniai

- Komandos stabilumas
 - Palaikymo kaštai mažesni, jei ją atnaujina tas pats personalas. Tačiau komanda paprastai yra išformuojama užbaigus projektą.
- Kontrakto sąlygos
 - Kontrakte gali būti nenumatyti sistemos kūrėjų įsipareigojimai palaikymui, taigi jie nesuinteresuoti projektuoti programą taip, kad ją būtų lengva atnaujinti.
- Personalo įgūdžiai
 - Palaikymo komanda dažnai neturi patirties ir jų sistemos taikymo žinios yra ribotos.
- Programos amžius ir struktūra
 - Kai programos sensta, jų struktūra blogėja ir darosi sunkiau jas suprasti bei atnaujinti.

Palaikymo prognozavimas

- Palaikymo prognozavimui reikia įvertinti sistemos dalis, kurios gali kelti problemas ir kurių palaikymas gali būti brangus. Visuomet reikia atsiminti:
 - Keitimo išlaidos priklauso nuo su tuo atnaujinimu susijusių komponentų palaikomumo;
 - Pakeitimų realizavimas sumažina sistemos palaikomumą;
 - Palaikymo kaštai priklauso nuo keitimų skaičiaus, o keitimo kaštai priklauso nuo palaikomumo.

Palaikymo prognozavimas



Keitimų prognozavimas

- Norint prognozuoti keitimų skaičių, reikia suprasti sistemos ir jos aplinkos tarpusavio santykius.
- Tvirtai susijusioms sistemoms keitimai reikalingi kiekvieną kartą keičiantis aplinkai.
- Šią sąveiką įtakojantys veiksniai:
 - Sistemos sąsajų skaičius ir sudėtingumas;
 - Dažnai besikeičiančių sistemos reikalavimų skaičius;
 - Verslo procesai, kuriuose ši sistema naudojama.

Sudėtingumo matai

- Palaikomumą galima prognozuoti pagal sistemos komponentų sudėtingumą.
- Tyrimai parodė, kad didžiausia palaikymo pastangų dalis skiriama sąlyginai nedideliame sistemos sudėtingų komponentų skaičiui.
- Komponento sudėtingumas priklauso nuo:
 - Valdymo struktūrų sudėtingumo;
 - Duomenų struktūrų sudėtingumo;
 - Procedūrų ir modulių dydžio.

Temos

- Programų vystymo dinamika
- Programų priežiūra
- Vystymo procesai
- Likitinių sistemų vystymas

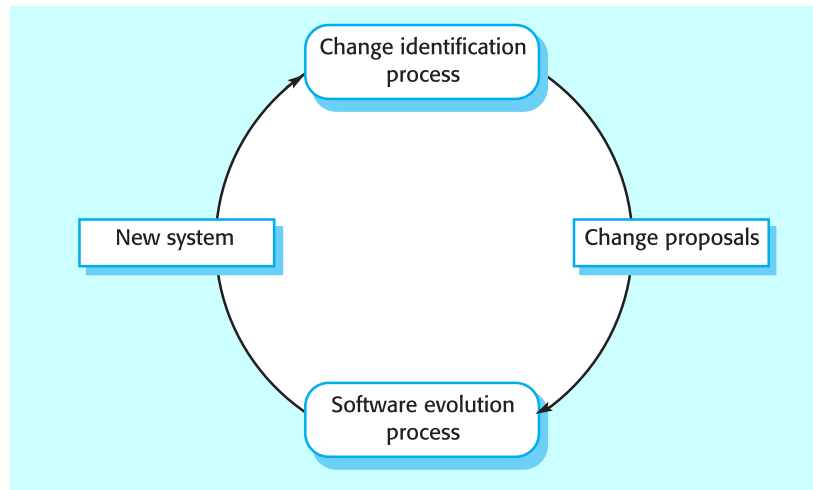
Procesų matai

- Procesų matai gali būti naudojami palaikomumui įvertinti:
 - Prašymų taisyti klaidas skaičius;
 - Vidutinis laikas, reikalingas atlikti poveikio analizę;
 - Vidutinė pakeitimo realizavimo trukmė;
 - Laukiančių prašymų atnaujinti skaičius.
- Jeigu vienas arba visi šie rodikliai auga, tai gali rodyti palaikomumo sumažėjimą.

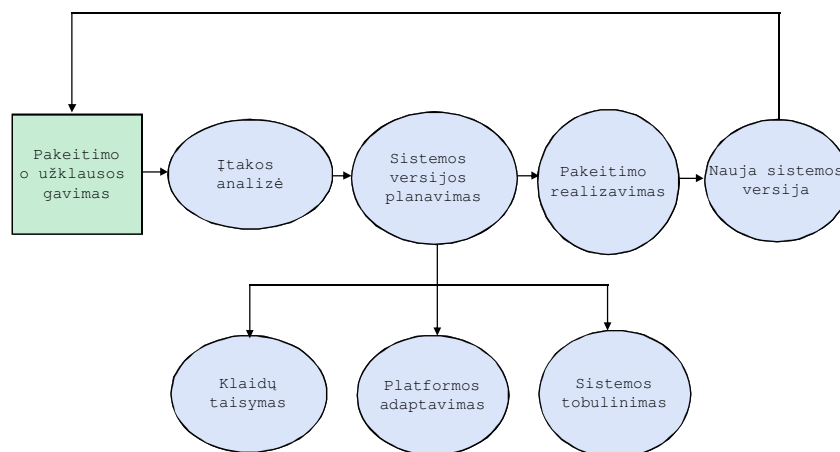
Vystymo procesai

- Vystymo procesai priklauso nuo:
 - Palaikomų programų tipo;
 - Naudoto kūrimo proceso;
 - Dalyvaujančių žmonių sugebėjimų ir patirties.
- Pasiūlymai keitimui stimuliuoja sistemos vystymą. Keitimai ir vystymas trunka visą sistemos gyvavimo ciklą.

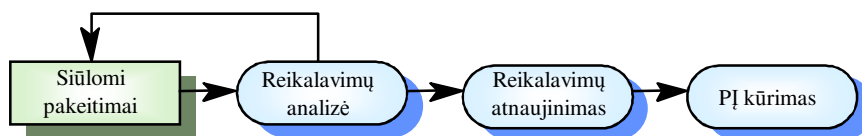
Keitimų identifikavimas ir vystymas



Sistemos vystymosi procesas



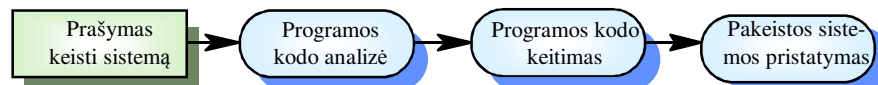
Pakeitimų realizavimo etapai



Prašymai keisti

- Prašymai keisti gaunami iš sistemos vartotojų, klientų arba vadovų.
- Teoriškai visi prašymai keisti turėtų būti atidžiai išanalizuoti kaip palaikymo proceso dalis ir tada realizuoti.
- Praktiškai kai kurie reikalavimai atnaujinti turi būti patenkinti nedelsiant:
 - Defekto taisymas;
 - Atnaujinimai dėl sistemos aplinkos pasikeitimo;
 - Skubiai reikalingi verslo logikos atnaujinimai.

Skubaus taisymo etapai



Sistemos perdarymo inžinerija

- Tik dalies arba visos paveldėtos sistemos restruktūrizavimas arba perrašymas nekeičiant jos funkcionalumo
- Pritaikoma ten, kur kelios, bet ne visos didelės sistemos posistemės reikalauja dažno palaikymo
- Perdarymo inžinerija reikalauja pastangų padaryti sistemą lengviau palaikoma. Sistemai gali reikėti restruktūrizavimo arba pakeitimų dokumentacijoje.

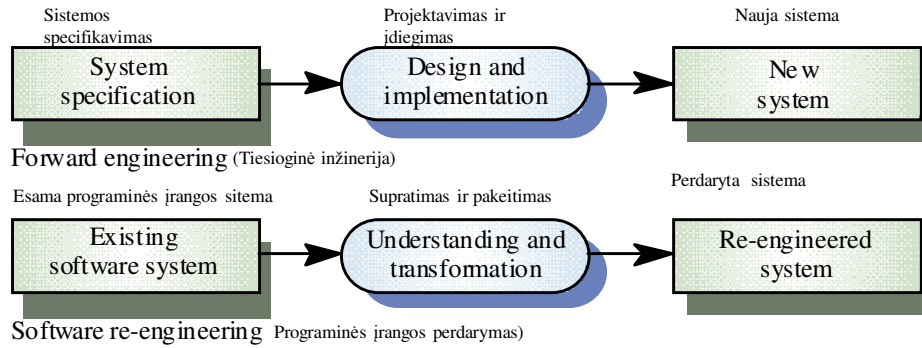
Kada perdaryti?

- Kai pakeitimai dažniausiai vykdomi tam tikroje sistemos dalyje, būtina perdaryti tą dalį
- Kuomet techninės arba programinės įrangos palaikymas pasensta (tampa nebevertotinu)
- Kuomet turime įrankius, reikalingus perprojektavimui

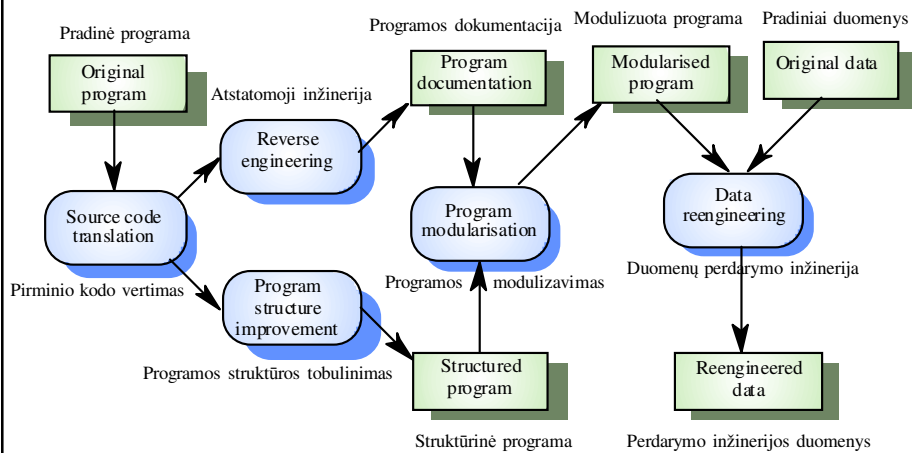
Perdarymo privalumai

- Sumažėjusi rizika:
 - Naujos programinės įrangos gamyboje yra didelė rizika. Gali iškilti gamybos, personalo ir specifikacijų problemų
- Sumažėję kaštai:
 - Perdarymo kaina dažniausiai yra žymiai mažesnė nei naujos programinės įrangos kūrimo

Tiesioginė ir perdarymo inžinerija



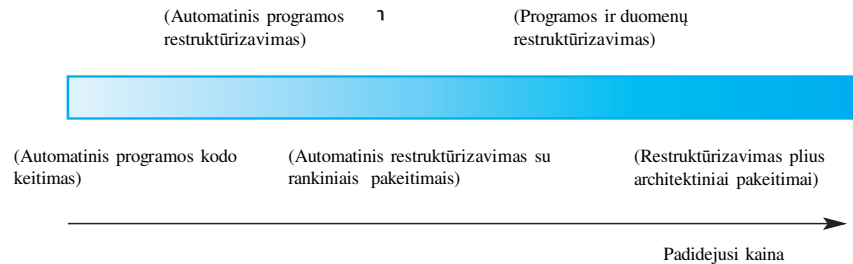
Perdarymo procesas



Perdarymo proceso veiklos

- Išėities kodo “vertimas” į naują kalbą.
- Atbulinė (reverse) inžinerija, jos metu analizuojame programą ir andome iš jos ištraukti visą įmanomą informaciją. Kartais šis procesas dar vadinamas PĮ archeologija
- Programos išėities kodo struktūros gerinimas siekiant padidinti suprantamumą.
- Programos struktūros reorganizavimas.
- Duomenų perdarymas ir restruktūrizavimas

Perdarymo metodai



Perdarymo kaštų faktoriai

- Programinės įrangos, kurią reikia perprojektuoti, kokybė
- Esamų įrankių parama perdarymui
- Reikiamas duomenų pakeitimų dydis
- Personalo patirtis

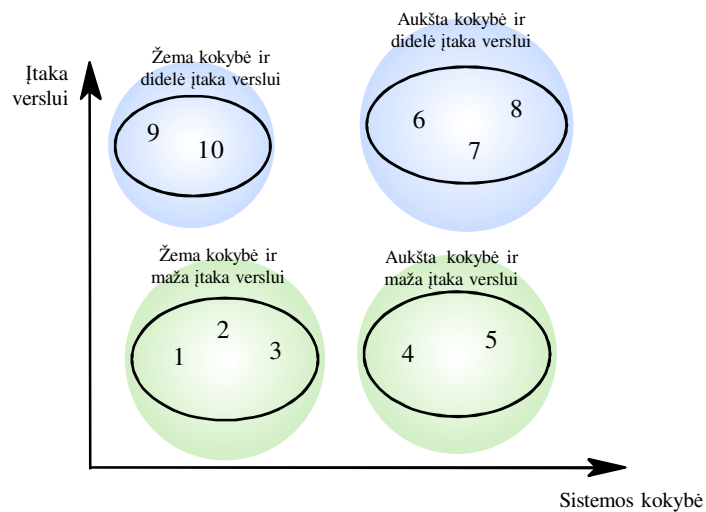
Temos

- Programų vystymo dinamika
- Programų priežiūra
- Vystymo procesai
- Paveldėtų sistemų vystymasis

Paveldėtų sistemų įvertinimas

- Organizacijos, kurios pasitiki paveldėtosiomis sistemomis, privalo pasirinkti jų vystymo strategiją
 - visiškai išmesti sistemą ir pakeisti verslo procesus, taip kad sistemos neberekėtų
 - toliau palaikyti paveldėtą sistemą
 - sistemą perprojektuoti, kad pagerėtų jos palaikomumas
 - keisti sistemą nauja
- Strategija pasirenkama atsižvelgiant į sistemos kokybę ir naudingumą verslui

Sistemos kokybė ir jos įtaka verslui



Paveldėtų sistemų kategorijos

- Žema kokybė, maža įtaka verslui
 - Dėmesio tokioms sistemoms skiriame mažai
- Žema kokybė, didelė įtaka verslui
 - Sistema svarbi verslui, tačiau ją brangu išlaikyti. Turėtų būti perprojektuota arba pakeista nauja sistema, jei tokia egzistuoja.
- Aukšta kokybė, žema įtaka verslui
 - Pakeisti su COTS, palaikyti.
- Aukšta kokybė, aukštas verslo įvertinimas
 - Tęsti įprastines sistemos palaikymo procedūras

Reikšmės verslui įvertinimas

- Įvertinimas turi apimti įvairius požiūrius, skirtingas nuomones
 - Galutinių sistemos vartotojų
 - Verslo partnerių
 - Vadovų
 - IT vadovų
 - Aukščiausių vadovų
- Apklausti skirtingus suinteresuotus asmenis(akcininkus) ir lyginti rezultatus

Sistemos kokybės įvertinimas

- Verslo proceso įvertinimas
 - Kaip gerai verslo procesas atitinka einamuosius verslo tikslus.
- Aplinkos įvertinimas
 - Kiek efektyvi sistemos aplinka, ir kiek kainuoja sistemos palaikymas
- Taikomosios programos įvertinimas
 - Kokia yra taikomosios programinės įrangos kokybė

Verslo proceso įvertinimas

- Bandoma įvertinti skirtingus požiūrius į sistemą, apklausiant įvairius jos vartotojus (arba kitus suinteresuotus asmenis, akcininkus)
 - Ar procesų modelis apibrėžtas tiksliai ir ar jo laikomasi?
 - Ar skirtingi organizacijos padaliniai tai pačiai funkcijai atlikti naudoja skirtingus procesus?
 - Kokiomis aplinkybėmis procesas buvo priimtas?
 - Kokie yra sąryšiai su kitais verslo procesais ir ar jie būtini?
 - Kiek efektyviai paveldėta taikomoji programinė įranga palaiko procesą?

Aplinkos įvertinimas

Veiksny	Klausimai
Tiekėjo stabilumas	Ar tiekėjas vis dar egzistuoja? Ar tiekėjas finansiškai stabilus ir neplanuoja dingti? Jei tiekėjas nebedalyvauja versle, tai kas atsakingas už sistemų palaikymą?
Nesėkmių dažnis	Ar dažnai sistema nepavyksta pasinaudoti dėl aparatūrinės įrangos trikžių? Ar dažnai tarnybinė programinė įranga veikia neapibrėžtai ir verčia perkrauti sistemą?
Amžius	Koks programinės ir aparatūrinės įrangos amžius? Kuo senesnė aparatūra ir tarnybinės programos, tuo sistema labiau susenusi. Ji gali funkcionuoti teisingai, bet gali būti žymiai didesnė ekonominė ir verslo nauda pereiti prie modernesnių sistemų.
Veikimo sparta	Ar sistemos darbo sparta pakankama? Ar sparta turi pastebimą poveikį sistemos vartotojams?
Palaikymo reikalavimai	Kokios darbuotojų priežiūros reikalauja aparatūrinė ir programinė įranga? Ar priežiūra nekainuoja per daug? Gal verta apsvarstyti galimybę keisti sistemą kita. Galbūt verta panagrinėti sistemos pakeitimą?
Palaikymo kaštai	Kokios yra aparatūrinės įrangos palaikymo ir programinės įrangos palaikymo licencijų kainos? Senesnės aparatūrinės įrangos palaikymo kaštai didesni nei modernių sistemų. Gali tekti pirkti brangias metines licencijas programinės įrangos priežiūrai.
Suderinamumas	Ar yra problemų jungiant sistemą su kita sistema? Ar kompiliatoriai gali būti naudojami su einamomis veikiančių sistemų versijomis? Ar reikia emuliuoti aparatūrinę įrangą?

Taikomosios programos įvertinimas

Veiksny	Klausimai
Suprantamumas	Kaip sunku suprasti naudojamos sistemos kodą? Kiek sudėtingos valdančiosios struktūros? Ar kintamieji turi reikšmingus vardus kurie atspindi jų paskirtį?
Dokumentacija	Kokia sistemos dokumentacija prieinama? Ar dokumentacija yra pilna, nuosekli ir atnaujinta?
Duomenys	Ar egzistuoja aiškiai apibrėžtas sistemos duomenų modelis? Kiek dažnas duomenų dubliavimas (pertekliškas) skirtinguose failuose? Ar sistemos naudojami duomenys atnaujinami nuosekliai?
Sparta	Ar taikomosios programos darbo sparta pakankama? Ar vartotojams kyla nepatogumų dėl sistemos darbo spartos?
Programavimo kalba	Ar egzistuoja modernūs kompiliatoriai programavimo kalbai, kuri naudota sistemai vystyti? Ar programavimo kalba vis dar naudojama naujų sistemų vystymui?
Versijų valdymas	Ar visos visų sistemos dalių versijos valdomos automatiškai? Ar yra tikslus naudojamos sistemos komponentų versijų aprašymas?
Testavimų duomenys	Ar egzistuoja duomenų rinkinys sistemai testuoti? Ar yra rezultatai regresinių testų, atliktų papildžius sistemą naujomis savybėmis?
Personalo įgūdžiai	Ar yra žmonių turinčių įgūdžių palaikyti taikomąją programą? Ar tik ribotas skaičius žmonių supranta sistemą?

Sistemos rodikliai

- Norint įvertinti taikomosios sistemos kokybę, reikia surinkti kiekybinius duomenis
 - Prašymų keisti sistemą skaičius
 - Sistemos naudojamų skirtingų vartotojo sąsajų skaičius
 - Sistemos naudojamų duomenų apimtis

Esminiai aspektai

- Programų kūrimas ir vystymas turėtų būti vieningas iteratyvinis procesas.
- Lemano dėsniai aprašo sistemų vystymo esminius aspektus
- Palaikymo tipai: klaidų taisymas; pritaikymas naujai aplinkai; sistemos papildymas naujomis funkcijomis, arba senų funkcijų modifikavimas, siekiant patenkinti naujus reikalavimus.
- Programinės įrangos keitimo kaštai dažniausiai viršija programinės įrangos kūrimo kaštus.

Esminiai aspektai

- Vystymo procesas yra stimuliuojamas suinteresuotų asmenų prašymų daryti pakeitimus.
- Programų perdarymas siejamas su restruktūrizavimu, dokumentacijos perdarymu siekiant padaryti lengviau keičiamą programą ,
- Paveldėtų sistemų reikšmė verslui ir jų kokybė sąlygoja naudojamą vystymo strategiją.