



Microsoft Technology Centers

Microsoft Solutions Framework

PAT

Vladimirs NIKUĻŠINS

3.kursa doktorants

RTU DITF LDI

Lietišķo datorzinātņu katedra

Lekcijas plāns

- Kas ir MSF
- MSF principi
- MSF modeļi
 - Procesu modelis
 - Komandas modelis
- MSF disciplīnas
 - Sagatavošanas vadības disciplīnas
 - Projektu vadības disciplīna
 - Risku vadības disciplīna

Kas ir MSF

- Microsoft Solutions Framework ir augsti pielāgojama un mērogojama, pilnīgi integrēta programmatūras izstrādes procesu, principu un pierādīto prakšu kopa, kas ir paredzēta lietotāja pieprasītā programmatūras izstrādes procesa izveidošanai, automatizācijai un vadīšanai visa programmatūras izstrādes dzīves cikla laikā.

Vai MSF ir metodoloģija?

- Tā ir metodoloģija, kuru Microsoft dēvē par karkasi (framework). Kā uzsver Microsoft, klasiskā nozīmē MSF nav stingra metodoloģija, bet universāls, elastīgs un mērogojams karkass. Tas var būt adaptēts jebkura projektā biznesa-vadāmo risinājumu implementēšanai.

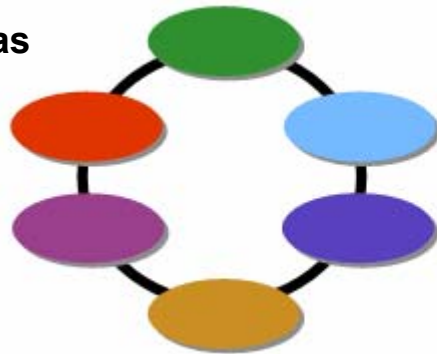
MSF vēsture

- Programmatūras izstrādes metodoloģija, kura attīstās kopš 1991. gada
- Pirmo reizi tika piedāvāta 1994. gadā kā kompānijas Microsoft labāko prakšu kopa
- 2003. gadā tika izlaista versija 3.0
- Pašlaik aktuālākā versija ir 4.0, kura tika izlaista 2005. gada novembrī
- MSF 4.0 pilnīgi integrēta ar programmatūras izstrādes rīku Microsoft Visual Studio 2005 Team System

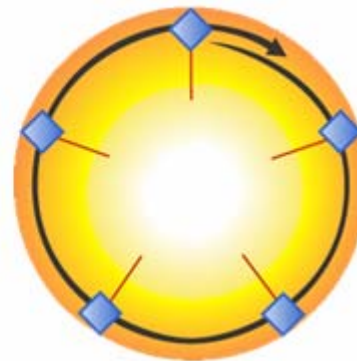
MSF struktūra

Divi modeļi

Komandas
modelis



Procesu
modelis



Trīs disciplīnas

Sagatavošanas
vadības
disciplīna



Risku vadības
disciplīna

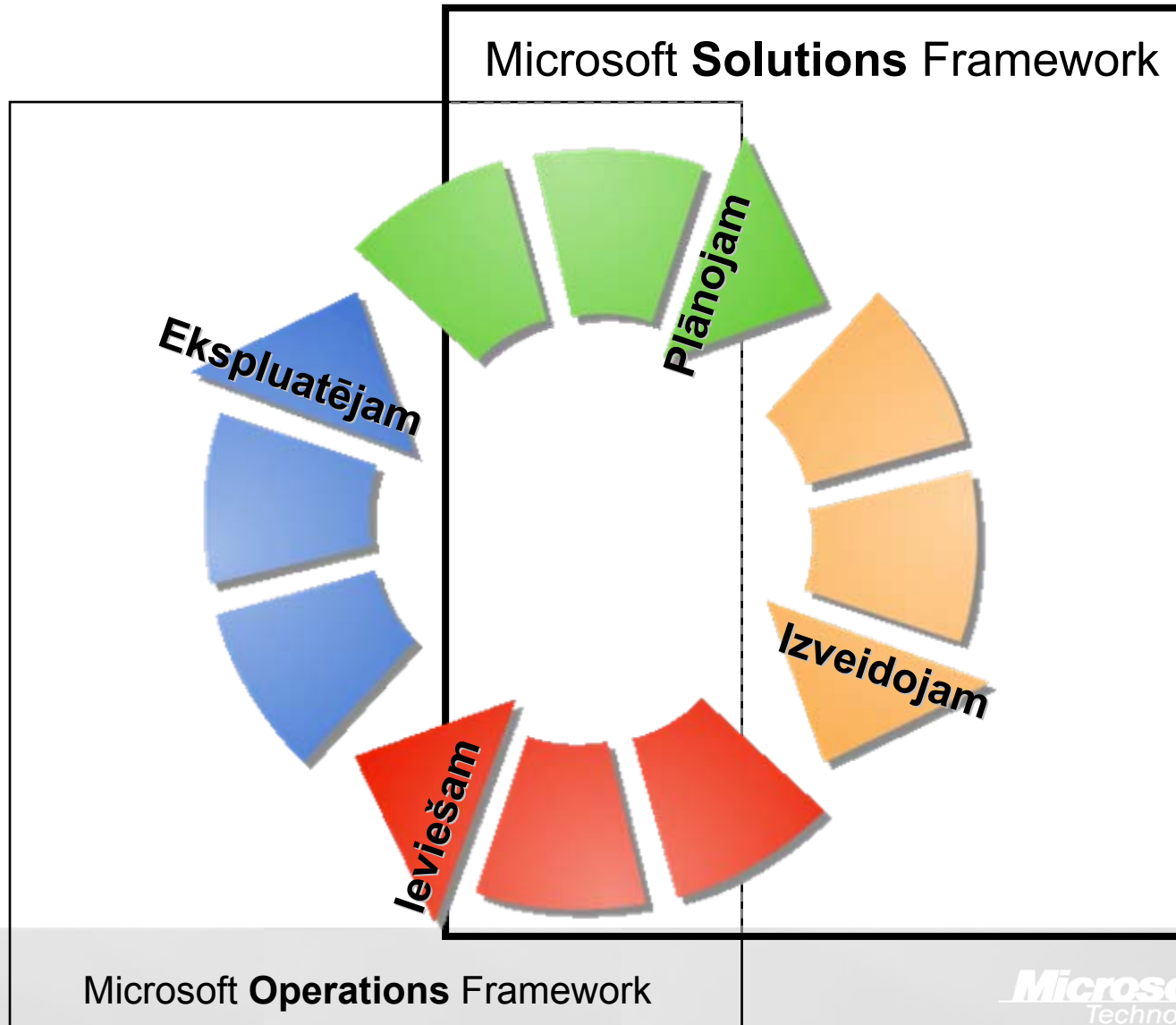
Projekta
vadības
disciplīna



MSF un MOF (1)

- MSF = Microsoft Solutions Framework
 - Microsoft pieeja IT projektu vadīšanai
 - Programmatūras izstrādes projekti
 - Infrastruktūras ieviešanas projekti
- MOF = Microsoft Operations Framework
 - Microsoft pieeja IT procesu (operāciju) vadīšanai uzņēmumā

MSF un MOF (2)



MSF principi (1)

- Veicināt atvērtas komunikācijas
 - *maksimizēt komandas locekļu individuālo efektivitāti un optimizēt darba produktivitāti; informācijai ir jābūt pieejamai un aktīvi koplietojamai*
- Kopīga vīzija
 - *vīzija palīdz noskaidrot mērķus, identificēt konfliktus un kļūdainus pieņēmumus tā, lai tie varētu būt atrisināti*
- Pilnvarot komandas locekļus
 - *MSF propagandē no lejas uz augšu saraksta sastādīšanu, sarakstu kuru komanda var atbalstīt jo uzticas tai*
- Uzskaitamība un sadalīta atbildība
 - *labi saprotamas uzskaitamības un atbildības samazina nenoteiktību par “kas, ko, kad un kā”*
- Fokuss un biznesa vērtībām
 - *kombinējot fokusu uz biznesa vērtībām ar kopīgo vīziju, projekta komanda un organizācija var precīzi definēt projekta eksistences izpratni, kā arī kā organizācijas panākumi tiks izmērīti biznesa vērtībās*

MSF principi (2)

- Paredzēt izmaiņas
 - *MSF ieteic paredzēt izmaiņas no ieinteresētām pusēm un pat no pašas izstrādātāju komandas kā tādas*
- Investīcijas kvalitātē
 - *investīcijas cilvēkos, kā arī procesos un rīkos*
- Mācīties no visām pieredzēm
 - *neveiksme mācīties no visām pieredzēm ir garantija kļūdu atkārtošanai*
- Partneri ar pasūtītājiem
 - *Izpratne par ierosināma risinājuma vērtību un efektīva paziņošana ir veiksmes vadošais faktors*
- Projektēt risinājumus gatavus lietošanai
 - *implementējot izmaiņas ir jābūt gatavam nodot risinājumu lietošanai jebkurā brīdī*

Requirements	Solution Definition	<ul style="list-style-type: none"> o Satisfy stakeholders o Define solution within project constraints 	<ul style="list-style-type: none"> o Stakeholders 	<ul style="list-style-type: none"> o Marketing/Corporate Communications o Business Analyst o Product Planning o Release Management
Program	Solution	<ul style="list-style-type: none"> o Coordinate identification and o Deliver solution within project constraints 	<ul style="list-style-type: none"> o Project Sponsor(s) 	<ul style="list-style-type: none"> o Project management o Program Management o Resource Management o Process Assurance o Project Quality Management o Project Operations
Design	Solution Design	<ul style="list-style-type: none"> o design solution within project constraints 		<ul style="list-style-type: none"> o Release Management o Technical Architecture
Development	Solution Construction	<ul style="list-style-type: none"> o Build solution to specifications 		
Test	Solution Quality	<ul style="list-style-type: none"> o Apply solution for release only after making sure all aspects of the solution meet or exceed their respective, defined quality levels 		<ul style="list-style-type: none"> o Functional Testing o System Testing
Release / Operations	Solution Usability	<ul style="list-style-type: none"> o Optimize/Enhance user performance/ effectiveness 	<ul style="list-style-type: none"> o Users o Operations o Support 	<ul style="list-style-type: none"> o Accessibility o Internationalization o Localization o Support o Training o Usability o Graphic Design
Release / Operations	Solution Deployment	<ul style="list-style-type: none"> o Smooth deployment and transition to operations 	<ul style="list-style-type: none"> o Operations 	<ul style="list-style-type: none"> o Delivery Infrastructure o Operations o Commercial Release Management o Build Master o Tool Administrator

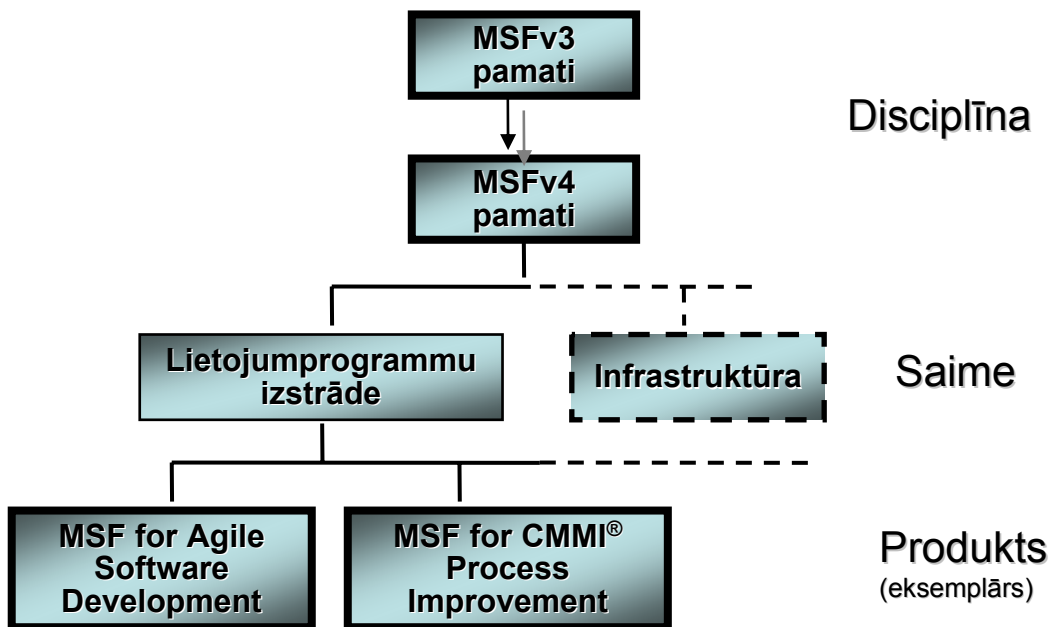
MSF domāšanas veidi

- Vienlīdzīgo komanda (team of peers)
 - *visam lomam ir jābūt īpašumtiesībām par produkta kvalitāti, jāatbalsta pasūtītāju, jāsaprot risināmo biznesa problēmu*
- Uz pasūtītāju orientēts domāšanas veids
 - *apmierināts pasūtītājās ir prioritāte numur viens*
 - *pasūtītāja fokuss uz izstrādes ietver sevī uzticēšanu no komandas pasūtītāja biznesa problēmas izprašanā un risināšanā*
- Risinājuma domāšanas veids
 - *uzskatīt sava darba rezultātus par risinājumu*
- Uzticēšanas domāšanas veids
 - *ir jāuzticas saviem komandas līdziniekiem*
- Kvalitātes domāšanas veids
 - *uzticēšana par kvalitāti*
 - *komandas pienākums ir izpildīt savu darbu visaugstākā kvalitātē; pat ja dienas laikā ir jānodod strādājošo programmatūru, ir jānodod reāli strādājošo programmas daļu*
- Gatavība apmācīties
 - *uzticēšana par pašattīstību apmācoties un koplietot zināšanas*

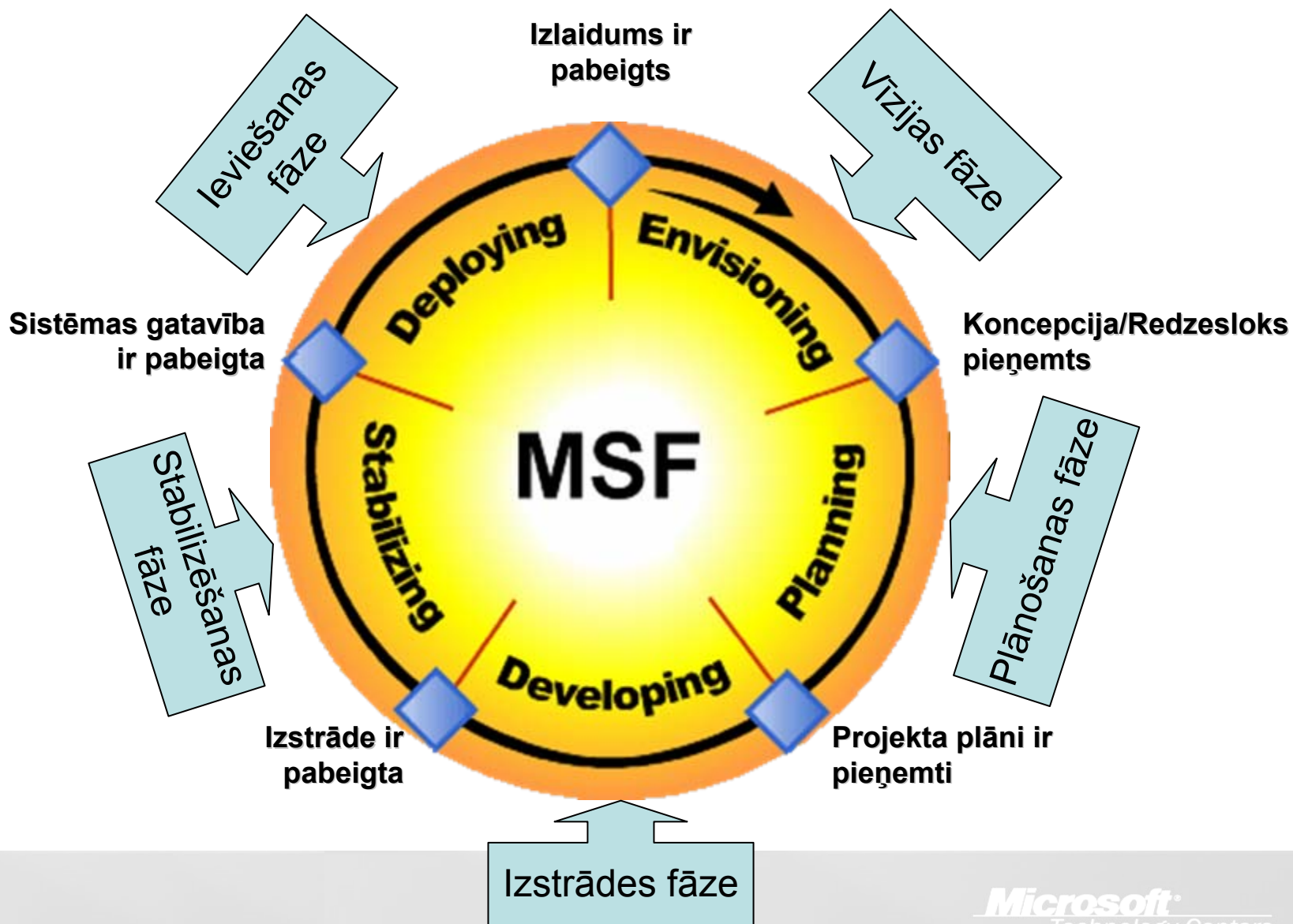
Versiju atšķirības (1)

- MSF v3 ir formāls programmatūras izstrādes process
- MSF v4 ir formāls + spējas programmatūras izstrādes process:
 - MSF for Agile Software Development
 - MSF for CMMI Process Improvement

Versiju atšķirības (2)



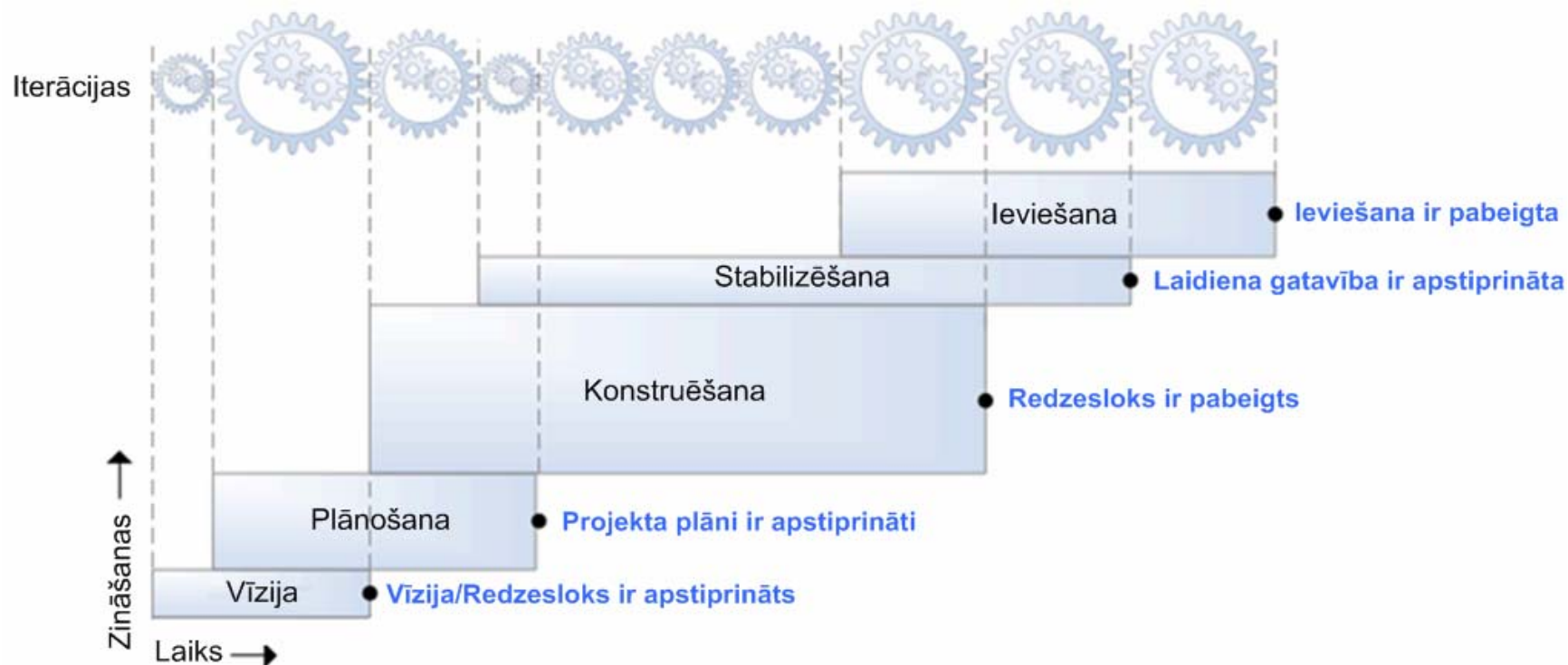
MSF Procesu Modelis



Katrai fāzei MSF definē:

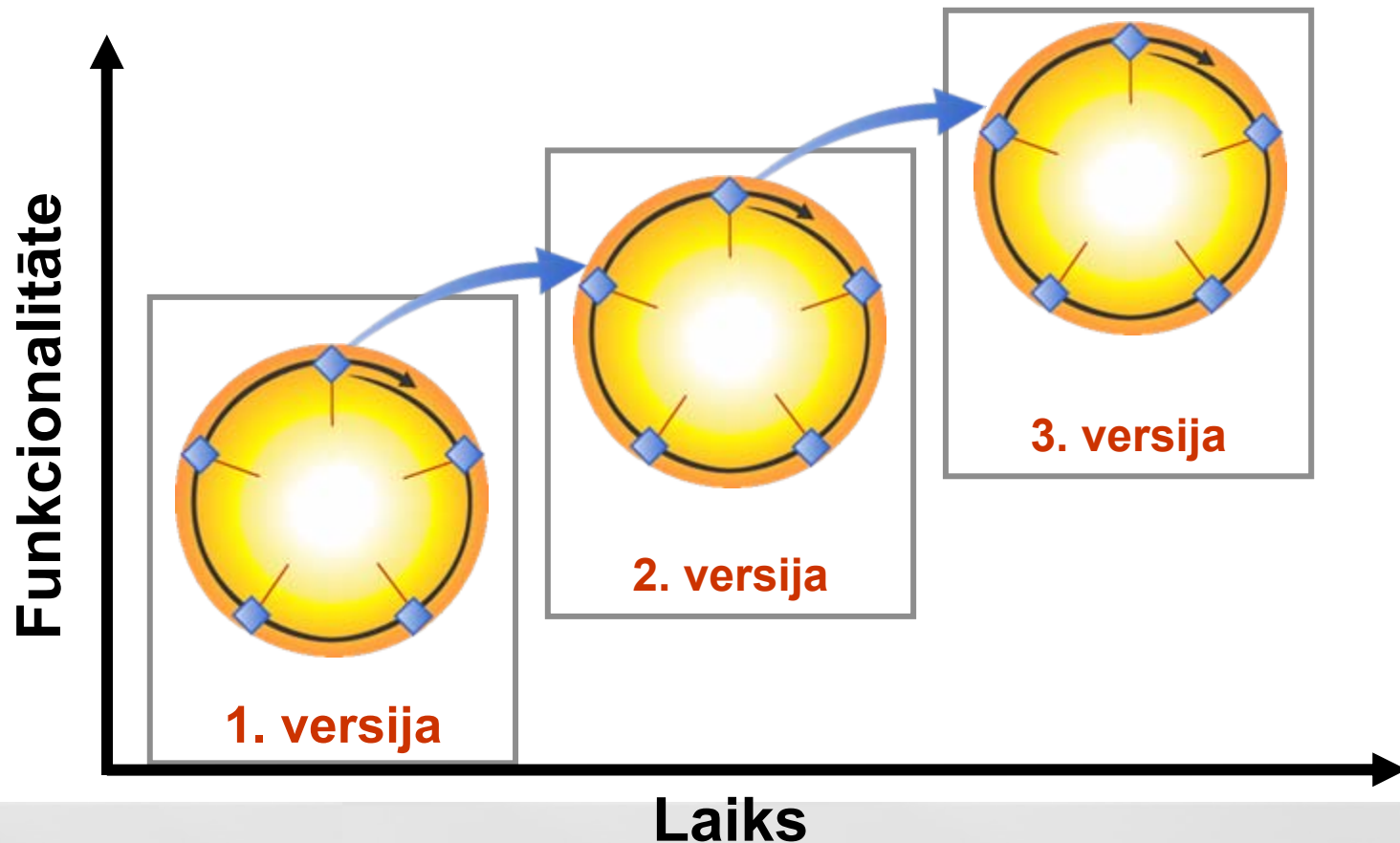
- Kādi ir katras konkrētas fāzes rezultāti (un kādi ir artefakti)
- Ar ko fāzes ietvaros strādā lomu klasteri

Procesa fāzes un vadīšana

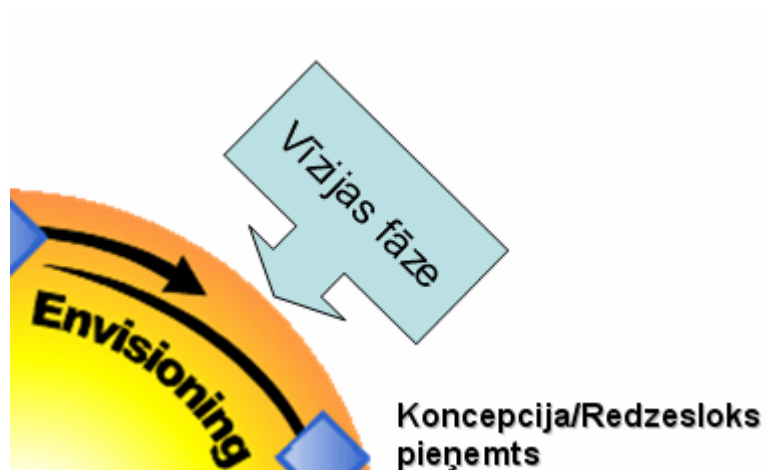


Iteratīva pieeja

Risku minimizēšana, sadalot lielus projektus vairākos mazākos projektos



Vīzijas fāze



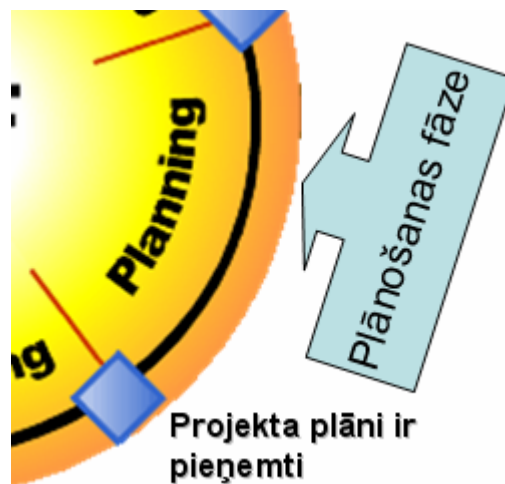
Vīzijas fāze: procesi

- Sastādīt komandu;
- Definēt projekta struktūru;
- Definēt biznesa mērķus;
- Novērtēt tekošo situāciju;
- Veidot vīzijas nostādni un definēt projekta redzesloku;
- Definēt prasības un lietotāju profilus;
- Izstrādāt risinājuma koncepciju;
- Novērtēt riskus;
- Noslēgt vīzijas fāzi;

Vīzijas fāzes robežpunkti un nodevumi

- Vīzijas robežpunkti:
 - Organizēts komandas kodols;
 - Izstrādāts redzesloks;
- Vīzijas nodevumi:
 - Redzes loka dokumenti;
 - Problēmas nostādne un mērķi;
 - Esoša procesa apskats;
 - Lietotāja prasību robežas definīcija;
 - Risinājuma koncepcija un projektēšanas stratēģijas;
 - Projekta struktūra;
 - Komandas dalībnieku lomas un atbildības;
 - Procesu standarti, pēc kā vadīsies šajā projektā;
 - Risku pārskats;
 - Iekšēja dokumentācija.

Plānošanas fāze



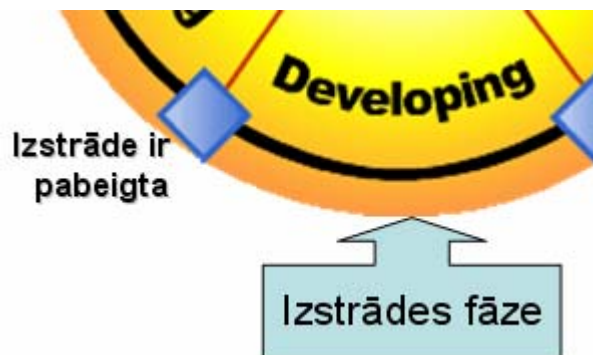
Plānošanas fāzes procesi

- Izstrādāt risinājuma projektējumu un arhitektūru;
- Izveidot funkcionālo specifikāciju;
- Izstrādāt projekta plānu;
- Izstrādāt projekta grafiku;
- Izveidot izstrādāšanas, testēšanas un versionēšanas vides;
- Noslēgt plānošanas fāzi;

Plānošanas fāzes robežpunkti un nodevumi

- Plānošanas robežpunkti:
 - Pabeigta lietojamo tehnoloģiju novērtēšana;
 - Izstrādāta funkcionālā specifikācija;
 - Izstrādāts projekta plāns;
 - Izstrādāts projekta grafiks;
 - Uzstādītas (set up) izstrādes un testēšanas vides;
- Plānošanas nodevumi:
 - Funkcionālā specifikācija;
 - Riska vadības plāns;
 - Projekta plāns un grafiks.

Izstrādes fāze



Izstrādes fāzes procesi

- Uzsākt izstrādes ciklu;
- Izstrādāt lietojuma prototipu;
- Izstrādāt risinājuma komponentus;
- “Uzbūvēt” risinājumu (savienot komponentus);
- Noslēgt izstrāde fāzi;

Izstrādes fāzes robežpunkti un nodevumi

- Izstrādes robežpunkti:
 - Izstrādāta koncepcijas realizācija lietojumā;
 - Sagatavotas iekšējas produkta versijas priekš testēšanas;
- Izstrādes nodevumi:
 - Pirmkods un izpildāmie faili;
 - Instalācijas scenāriji un konfigurācijas direktīvas (settings) ieviešanai;
 - Fināla funkcionālā specifikācija;
 - Produkta darbību uzturošie elementi;
 - Testa specifikācijas un testa gadījumi.

Stabilizēšanas fāze



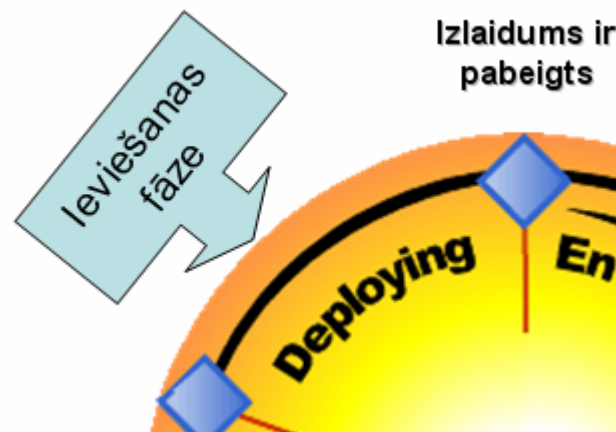
Stabilizēšanas fāzes procesi

- Stabilizēšanas fāzes procesi:
- Risinājuma iekšēja testēšana (t.s. α -testēšana);
- Risinājuma testēšana klienta pusē (t.s. β -testēšana);
- Noslēgt stabilizēšanas fāzi;

Stabilizēšanas fāzes robežpunkti un nodevumi

- Stabilizēšanas robežpunkti:
 - Bug-convergence - atrisināto kļūdu skaits vienāds ar atklāto kļūdu skaitu;
 - Zero-bug – atklāto kļūdu skaits ir vienāds ar 0;
 - Gala produkta kandidāts;
 - Gala produkts;
- Stabilizēšanas nodevumi:
 - Gala produkts un tā apraksts;
 - Produkta darbību uzturošie elementi;
 - Testu rezultāti;
 - Pirmkods un izpildāmie faili;
 - Projekta dokumentācija.

leviešanas fāze



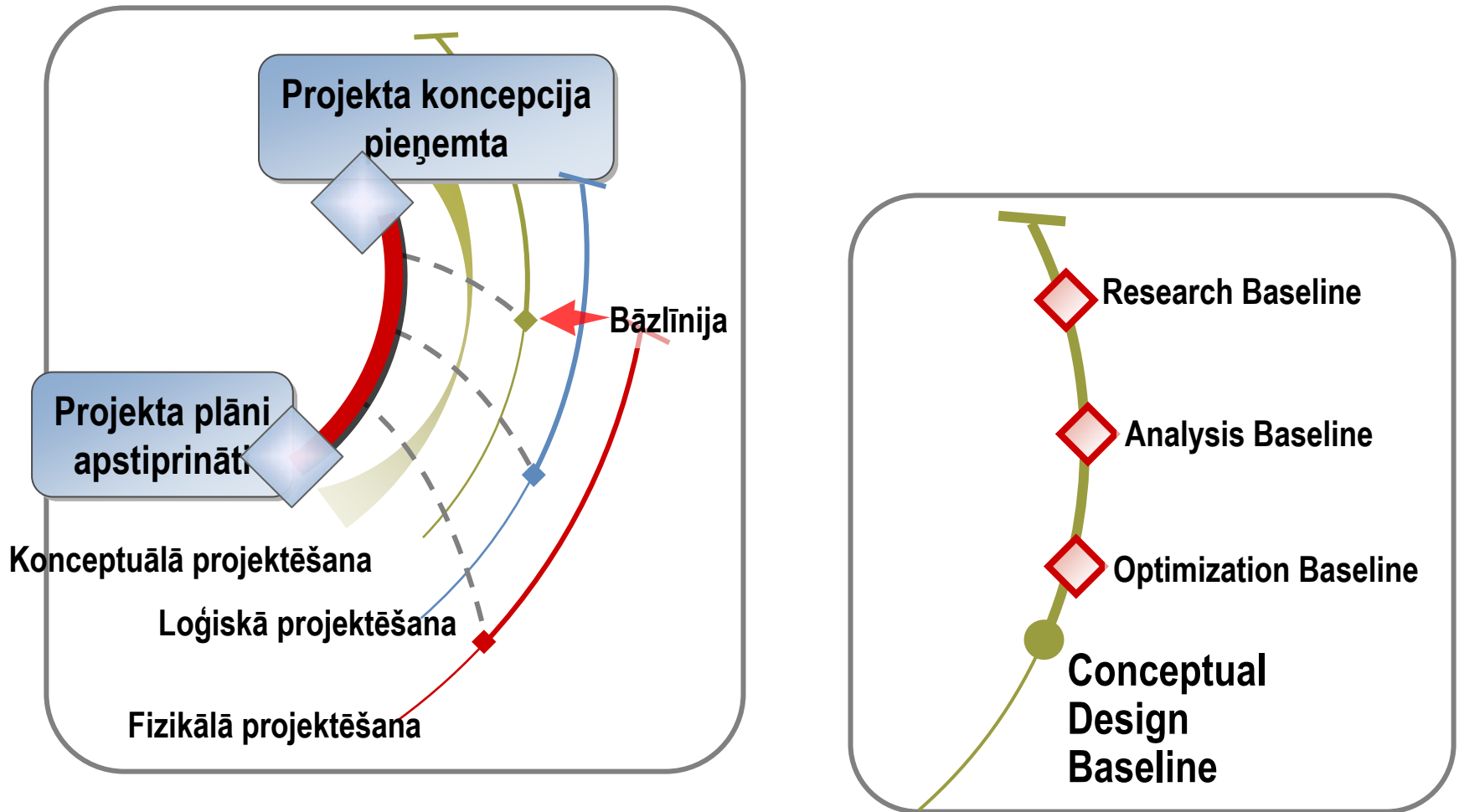
ieviešanas fāzes procesi

- Tiek ieviestas visas komponentes un tehnoloģijas;
- Tiek nodrošināta atbilstoša pieeja noteiktiem lietotājiem;
- Tiek stabilizēta ieviesta produkta darbība (piedalās gan izstrādātājs, gan visi lietotāji);

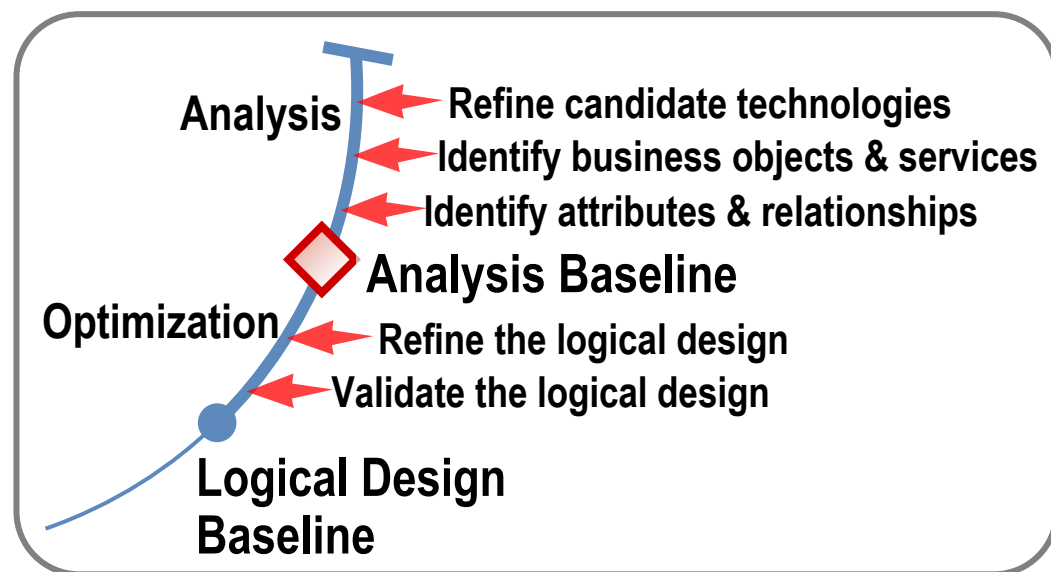
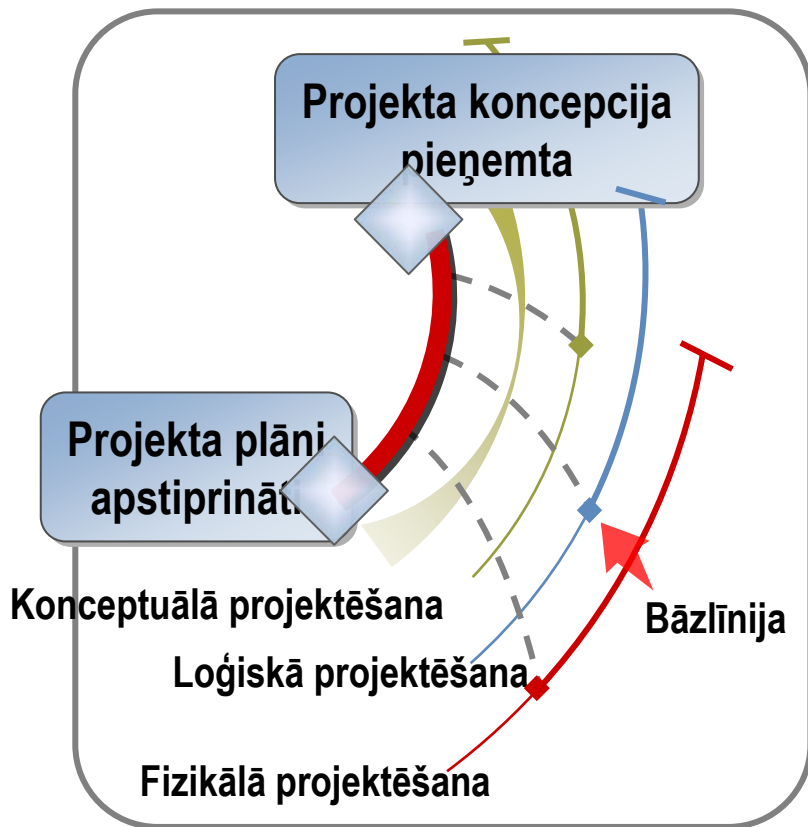
ieviešanas fāzes robežpunkti un nodevumi

- ieviešanas robežpunkti:
 - Pabeigta risinājuma komponentu ieviešana;
 - Nodrošināta pieeja risinājumam no lietotāja puses;
 - Ieviests risinājums strādā stabili;
- ieviešanas nodevumi:
 - Darbības un atbalsta IS;
 - Dokumentācijas glabātuve visām dokumentu versijām un kodam, kas tika izstrādāti projekta gaitā;
 - Apmācības plāns;
 - Projekta noslēguma atskaite (finālā dokumentācija, dati par klienta apmierinātību);

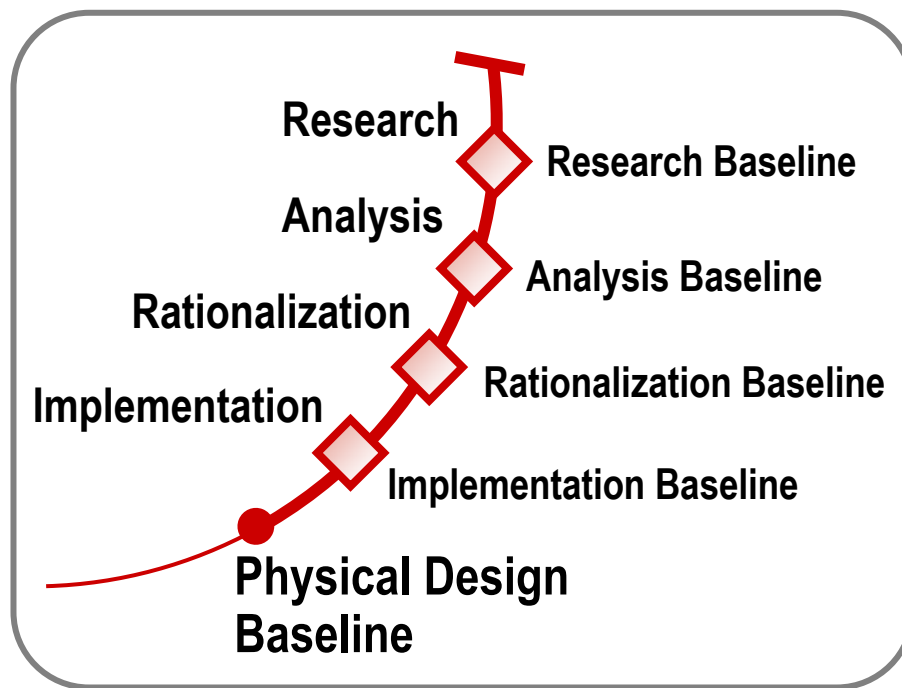
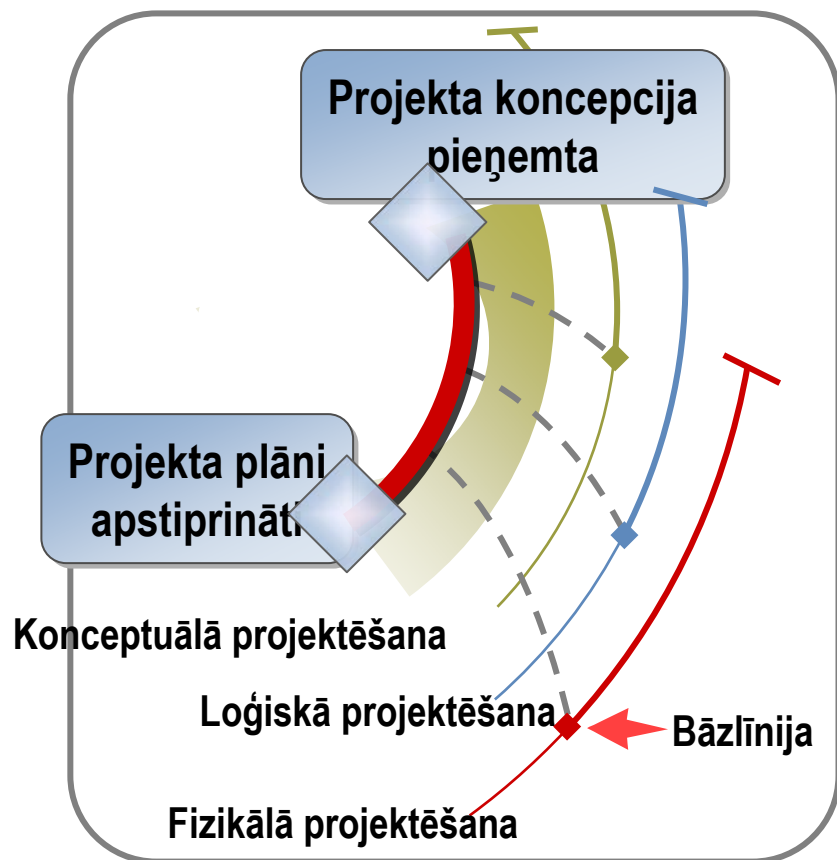
Konceptuālā projektēšana



Loģiskā projektēšana



Fizikālā projektēšana



Konceptuālā projektēšana

- Konceptuālā projektēšana ir savākšanas, analīzes un biznesa un lietotāju prioritāšu izvietojšanas process, un vēlāk risinājuma veidošana augstākā reprezentācijas līmenī.
- Konceptuālo projektēšanu raksturo:
 - Biznesa problēmas izpratne
 - Biznesa, pasūtītāja un galalietotāja prasību izpratne
 - Biznesa nākotnes stāvokļa apraksts

Loģiskā projektēšana (1)

- Loģiskā projektēšana ir definēta kā risinājuma apraksta process organizācijas, struktūras un daļu savstarpējas mijiedarbības ietvaros no projektu komandas perspektīvas.
- Loģisko projektēšanu raksturo:
 - Definē risinājuma saskaņotas daļas
 - Piedāvā infrastruktūru, kas satur visas sistēmas daļas
 - Ilustrē risinājuma salikšanu un mijiedarbību ar lietotāju un citiem risinājumiem

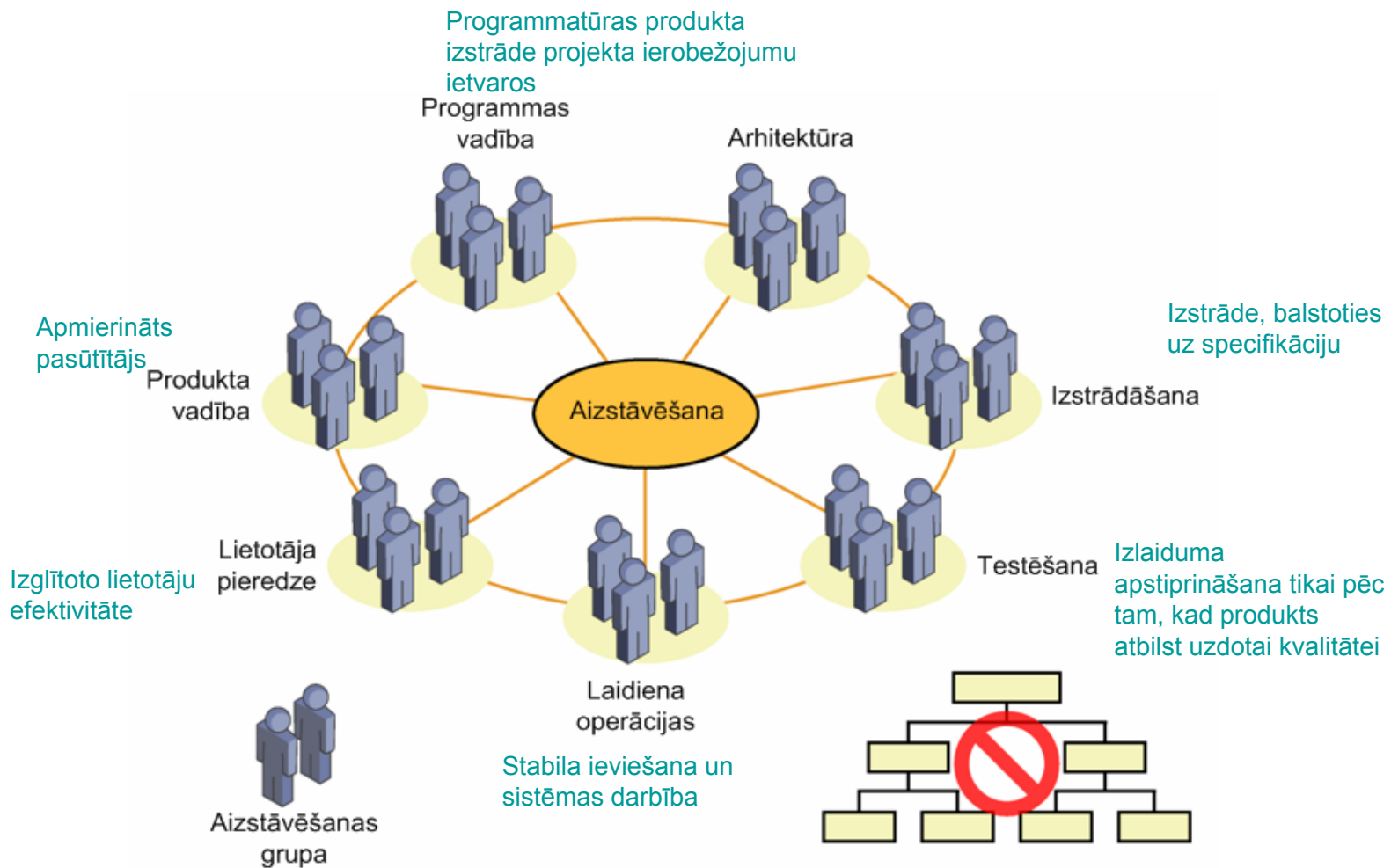
Loģiskā projektēšana (2)

- Loģiskas projektēšanas soļi:
 - Definēt potenciālas tehnoloģijas un instrumentus
 - Definēt biznesa objektus un servisu
 - Identificēt svarīgus atribūtus un pamata saites
- Rezultāti:
 - Loģiskais objektu modelis
 - Loģiskais datu modelis
 - Lietotāja saskarnes skice

Fizikālā projektēšana

- Fizikālā projektēšana – risinājuma komponentu, servisu un tehnoloģiju aprakstīšanas process izstrādātāju prasību perspektīvā. Fizikāla projektēšana nosaka izstrādāta risinājuma daļas, kā viņi tiks izvērsti un kā komunicēs viena ar otru.
- Fizikālās projektēšanas soļi:
 - Nepieciešamo tehnoloģiju identificēšana
 - Loģiska modeļa pārtaisīšana fizikālā
 - Projekta plāna izstrāde

MSF komandas modelis



Projekta grupas mērogošana

- Vienā lomu klasterī var būt vairāki cilvēki
- Viens cilvēks var izpildīt vairākas lomas
- Lielajās komandās:
 - Izveidot virzienu grupas
 - Izveidot funkcionālas grupas
- Mazajās komandās:
 - Izmantot lomu apvienošanas tabulu

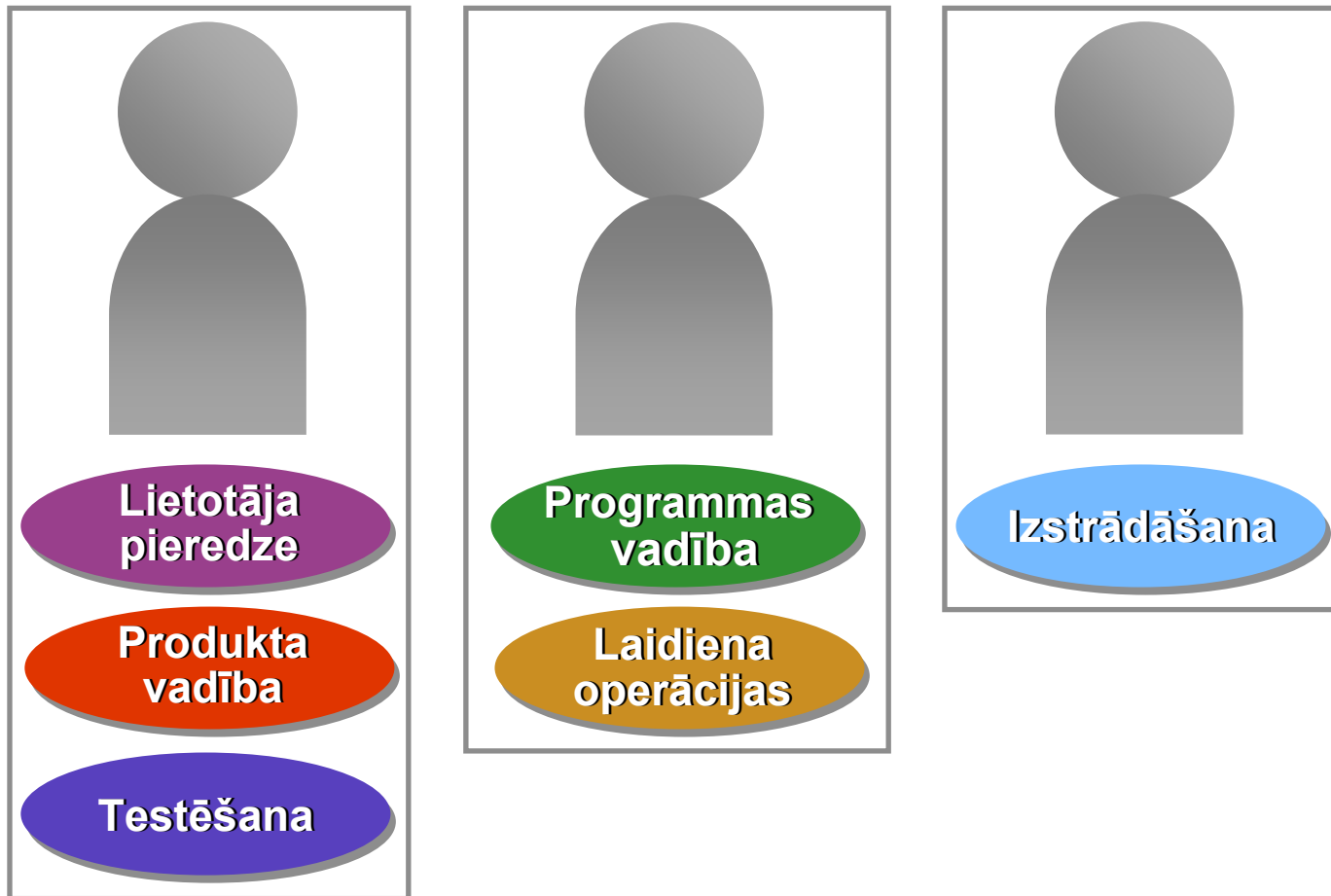
Lomu apvienošana

	Produkta vadība	Programmas vadība	Izstrāde	Testēšana	Izlaiduma vadība	Lietotāju pieredzes nodrošinājums
Produkta vadība		-	-	+	+	±
Programmas vadība	-		-	±	±	+
Izstrāde	-	-		-	-	-
Testēšana	+	±	-		+	+
Izlaiduma vadība	+	±	-	+		±
Lietotāju pieredzes nodrošinājums	±	+	-	+	±	

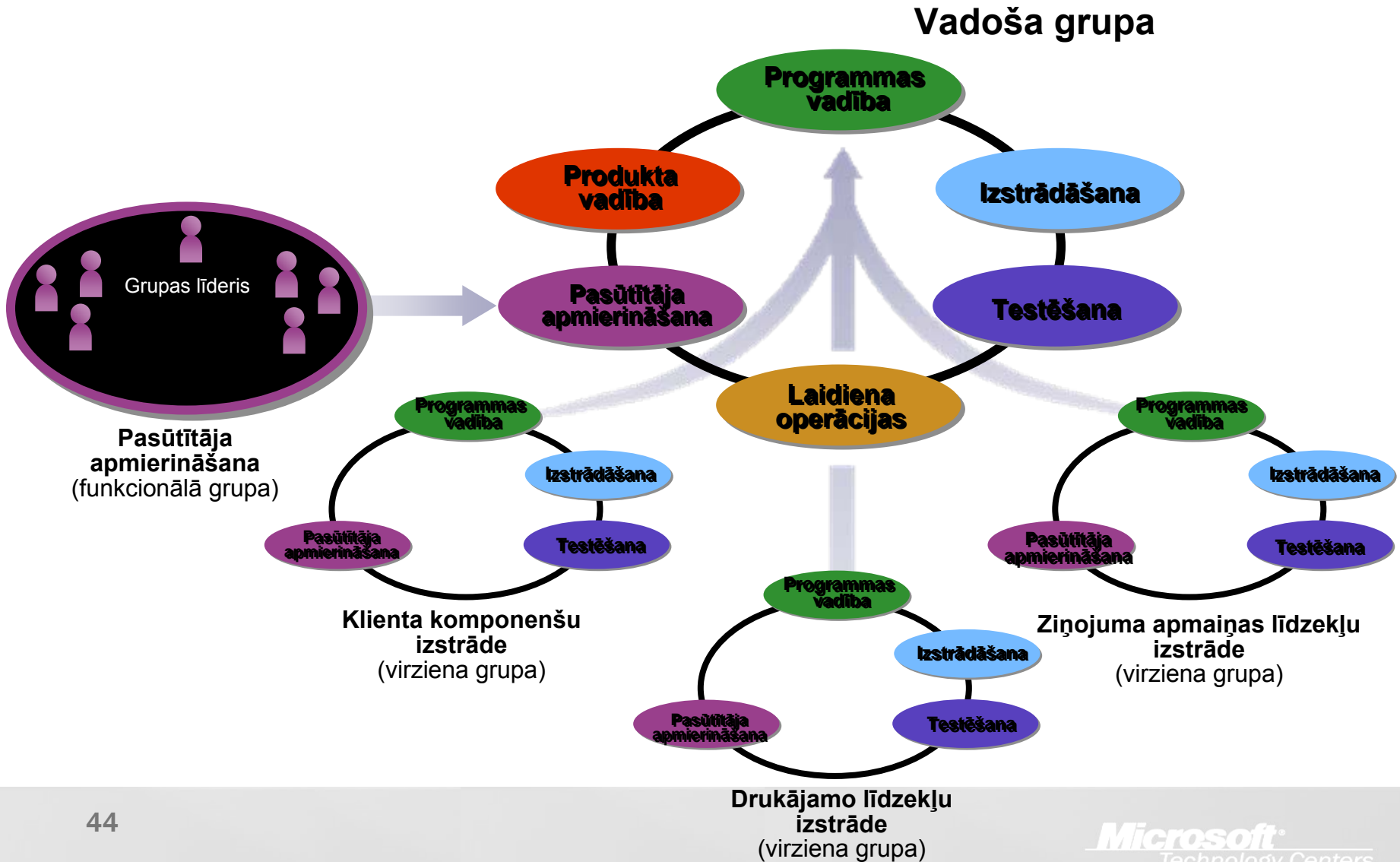
Apzīmējumi		
+	Pieļaujams	± Nevēlams
-	Nedrīkst	

Nelielas komandas

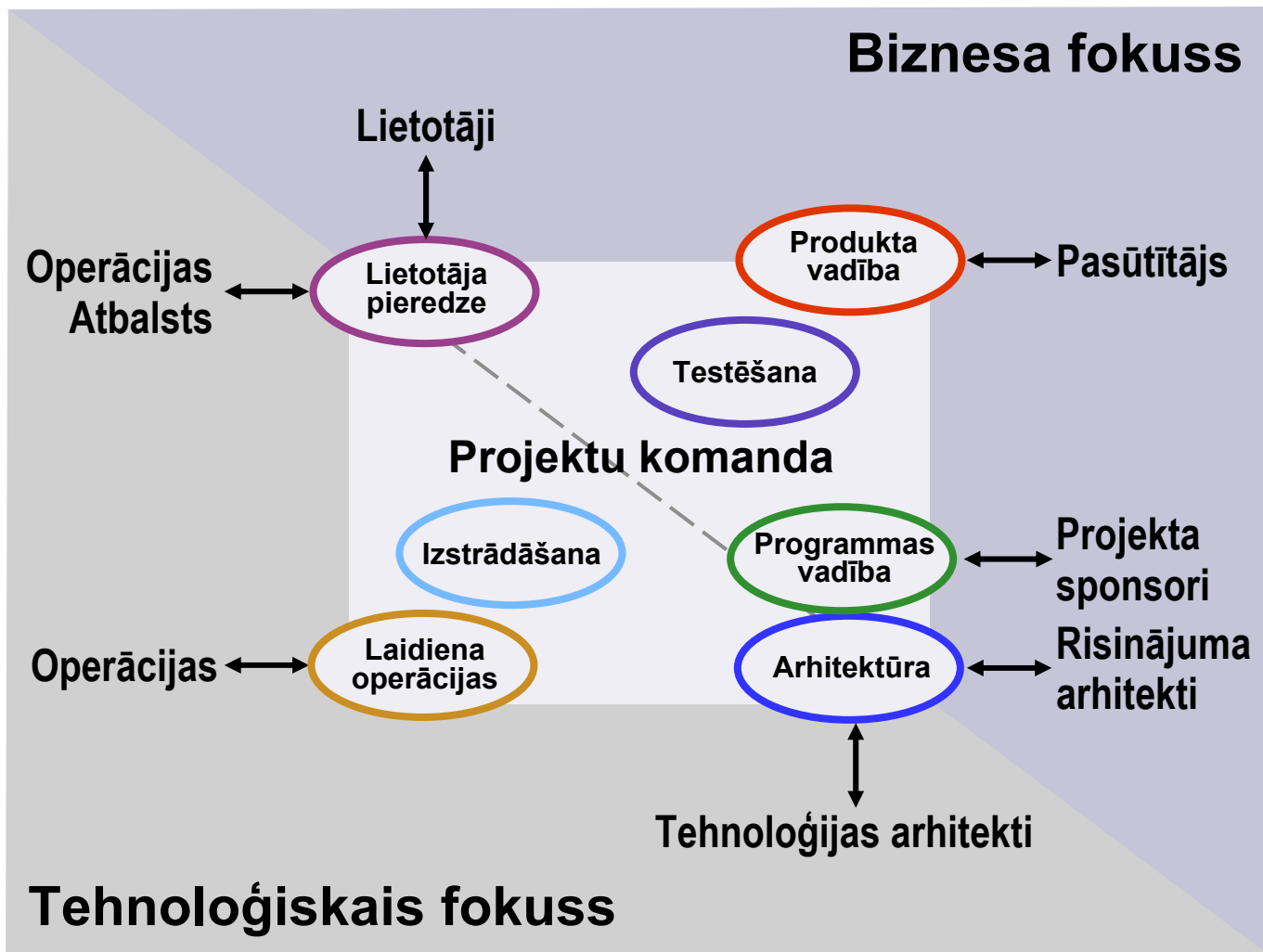
Mazas komandas, kombinētas lomas



Lielas komandas



Lomu interešu loki



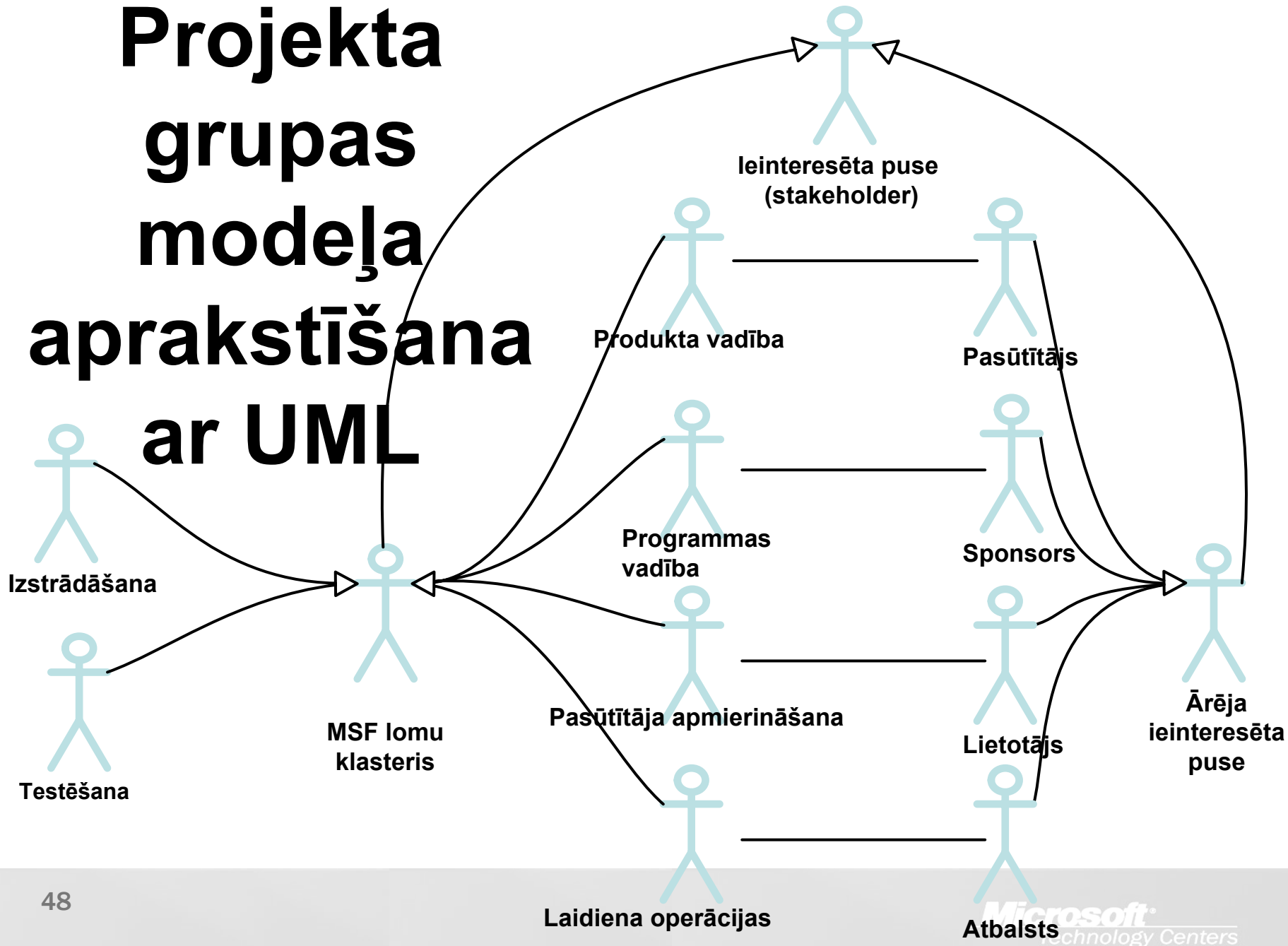
MSF komandas locekļu atbildība

Advocacy Group	Envisioning	Planning	Development	Stabilization	Deployment
Product Management	<ul style="list-style-type: none"> Overall goals Identify customer requirements Vision / scope document 	<ul style="list-style-type: none"> Business requirements analysis Communications plan 	<ul style="list-style-type: none"> Customer expectations 	<ul style="list-style-type: none"> Communications plan execution Launch planning 	<ul style="list-style-type: none"> Customer feedback, assessment, signoff
Program Management	<ul style="list-style-type: none"> Project structure Constraint identification 	<ul style="list-style-type: none"> Master project plan Master project schedule Budget 	<ul style="list-style-type: none"> Functional specification management Project tracking Plan updating 	<ul style="list-style-type: none"> Project tracking Constraint and scope trade-offs 	<ul style="list-style-type: none"> Stabilization management
Architecture	<ul style="list-style-type: none"> Design goals Solution concept 	<ul style="list-style-type: none"> Conceptual and logical design Technology evaluation Functional specification Initial development estimates 	<ul style="list-style-type: none"> Architectural validation 	<ul style="list-style-type: none"> Issue triage 	<ul style="list-style-type: none"> Solution / scope comparison
Development	<ul style="list-style-type: none"> Prototypes Development and technology options Feasibility analysis 	<ul style="list-style-type: none"> Logical and physical design Development plan / schedule 	<ul style="list-style-type: none"> Building the solution Configuration documentation 	<ul style="list-style-type: none"> Issue resolution Solution optimization 	<ul style="list-style-type: none"> Issue resolution Escalation support
User Experience	<ul style="list-style-type: none"> User Performance needs and implications 	<ul style="list-style-type: none"> Usage scenarios / use cases User requirements Localization / accessibility requirements User documentation, training plans and schedules 	<ul style="list-style-type: none"> Training Training plan updates Usability testing Graphic design 	<ul style="list-style-type: none"> User documentation stabilization Training materials User acceptance 	<ul style="list-style-type: none"> Training Training schedule management
Test	<ul style="list-style-type: none"> Testing approach Test acceptance criteria 	<ul style="list-style-type: none"> Design evaluation Testing requirements Test plan and schedule 	<ul style="list-style-type: none"> Technology and limited functional testing Issues identification Documentation testing Updated test plan 	<ul style="list-style-type: none"> Functional, system and integration testing Issue reporting and status Configuration testing 	<ul style="list-style-type: none"> Performance testing Problem resolution
Release Management	<ul style="list-style-type: none"> Deployment implications Operations management and supportability Operations acceptance criteria 	<ul style="list-style-type: none"> Design evaluation Operations requirements Pilot and deployment plan and schedule Define and build various project environments 	<ul style="list-style-type: none"> Rollout checklists Rollout and pilot plan updates Site preparation checklists 	<ul style="list-style-type: none"> Pilot setup and support Deployment planning Operations and support training 	<ul style="list-style-type: none"> Site deployment management Change approval

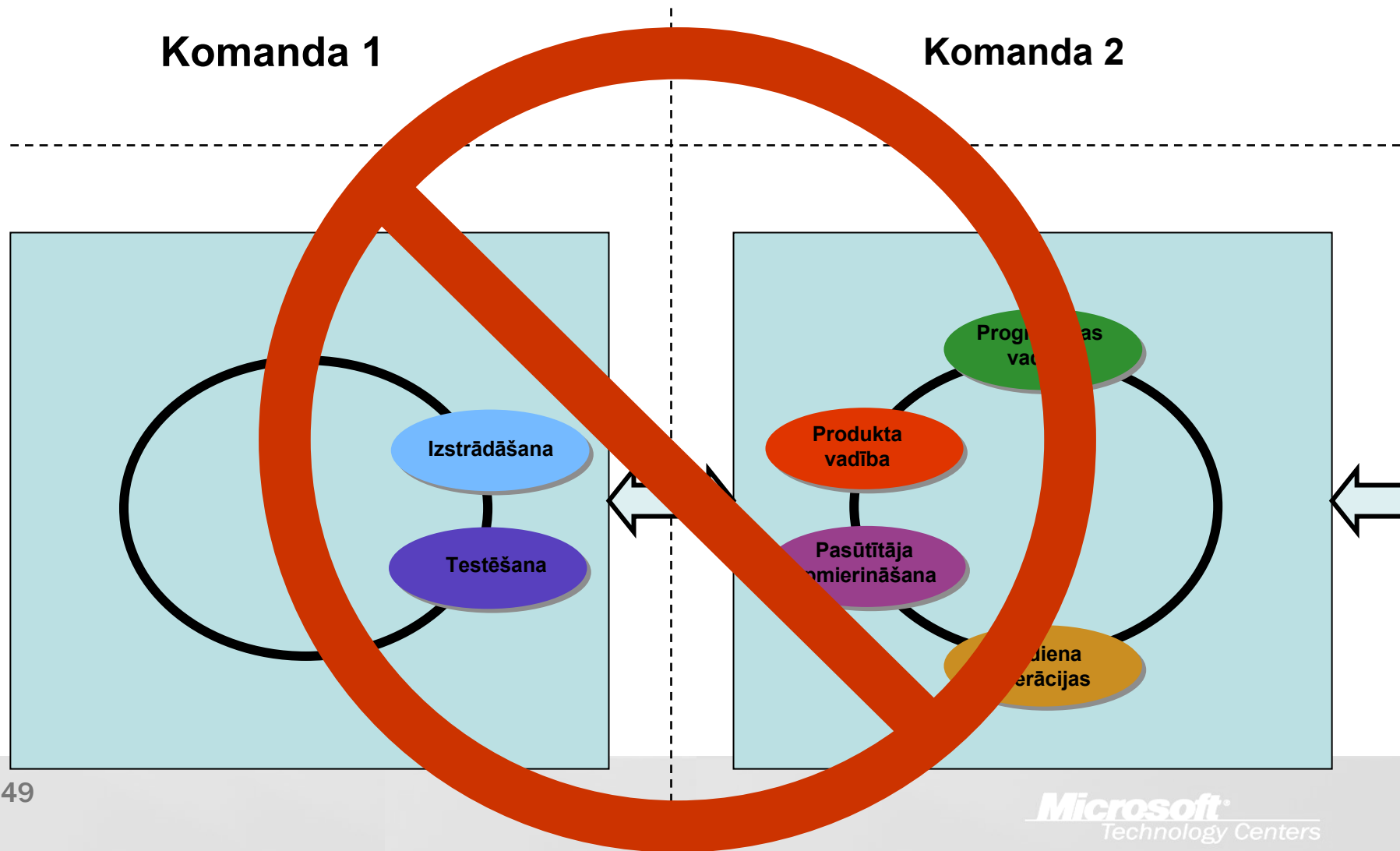
MSF komandas locekļu uzdevumi un mērķi

Advocacy Group	Advocate s	Quality Goals	Constituents	Functional Areas
Product Management	Solution Definition	<ul style="list-style-type: none"> ○ Satisfy stakeholders ○ Define solution within project constraints 	<ul style="list-style-type: none"> ○ Stakeholders 	<ul style="list-style-type: none"> ○ Marketing/Corporate Communications ○ Business Analyst ○ Product Planning ○ Release Management
Program Management	Solution Delivery	<ul style="list-style-type: none"> ○ Coordinate identification and optimization of project constraints ○ Deliver solution within project constraints 	<ul style="list-style-type: none"> ○ Project Spōnsor(s) 	<ul style="list-style-type: none"> ○ Project management ○ Program Management ○ Resource Management ○ Process Assurance ○ Project Quality Management ○ Project Operations
Architecture	Solution Design	Design a solution within project constraints		<ul style="list-style-type: none"> ○ Solution Architecture ○ Technical Architecture
Development	Solution Construction	Build solution to specifications		
Test	Solution Quality	Approve solution for release only after making sure all aspects of the solution meet or exceed their respective, defined quality levels		<ul style="list-style-type: none"> ○ Functional Testing ○ System Testing
User Experience	Solution Usability	Maximize/ Enhance user performance/ effectiveness	<ul style="list-style-type: none"> ○ Users ○ Operations Support 	<ul style="list-style-type: none"> ○ Accessibility ○ Internationalization ○ Technical Support Communications ○ Training ○ Usability ○ Graphic Design
Release / Operations	Solution Deployment	Smooth deployment and transition to operations	<ul style="list-style-type: none"> ○ Operations 	<ul style="list-style-type: none"> ○ Delivery Infrastructure ○ Operations ○ Commercial Release Management ○ Build Master ○ Tool Administrator

Projekta grupas modeļa aprakstīšana ar UML



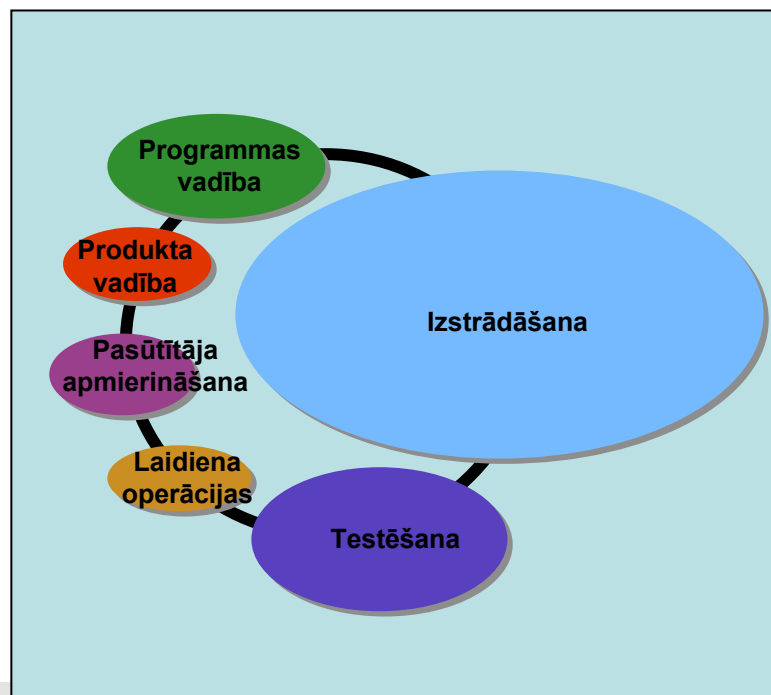
MSF modelis darba decentralizētos projektos



Sadalītās komandās projekta pamata lomam ir jābūt reprezentētām visās apakšgrupās

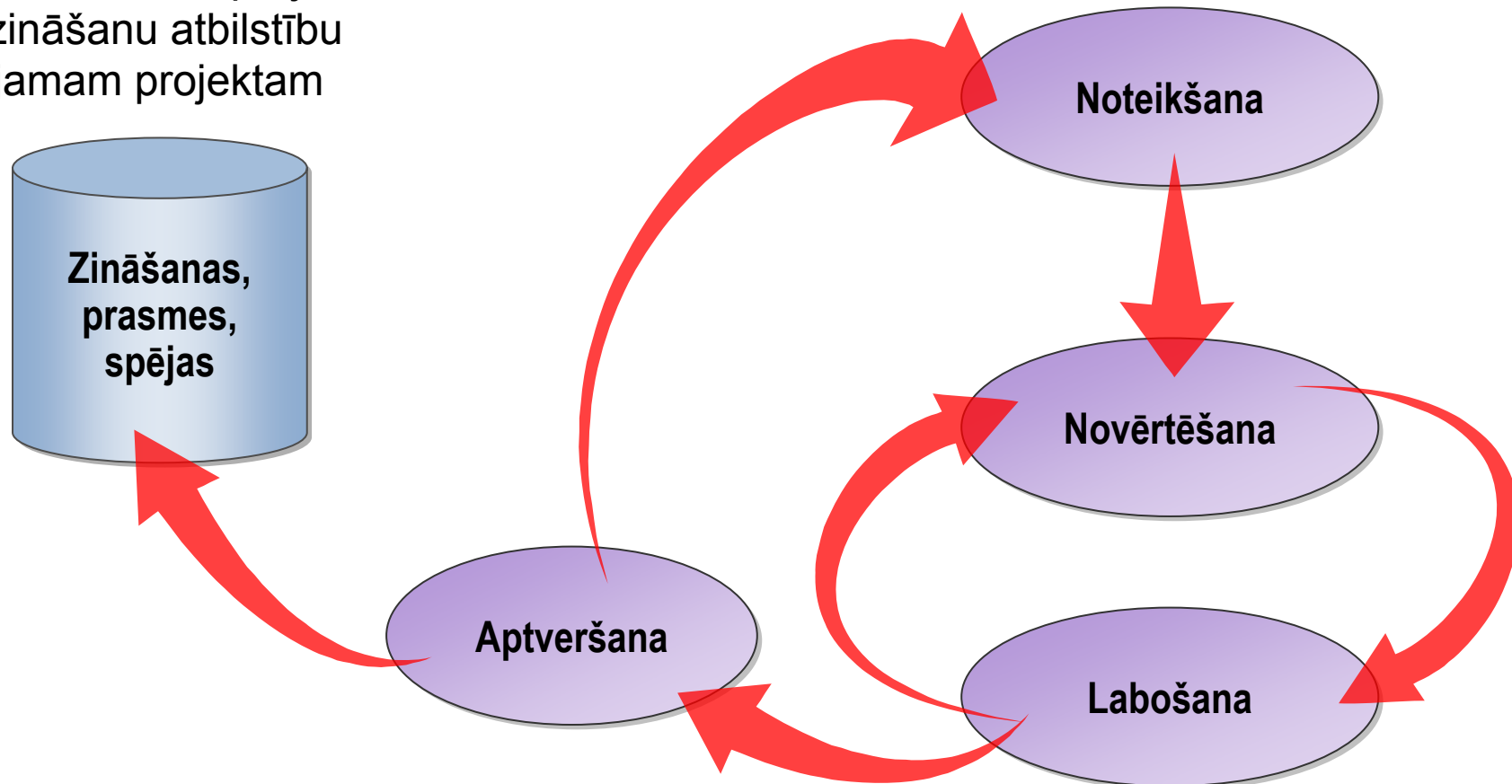
Komanda 1

Komanda 2



Sagatavošanas vadības disciplīna

Sagatavošanas vadības disciplīnas mērķis ir nodrošināt projektu grupas zināšanu atbilstību izstrādājamam projektam



Sagatavošanas vadības disciplīnas soļi

- 1. Noteikšana
 - Projektu scenāriji (scenarios)
 - Kompetence (competencies)
 - Profesionālas prasmes (proficiencies)
- 2. Novērtēšana
 - Zināšanu un prasmju novērtēšana (measure knowledge, skills, abilities)
 - Nesaskaņas analīze (analyze gaps)
 - Mācību plāna izveidošana (create learning plans)
- 3. Labošana
 - Apmācība (train)
 - Progresu novērošana (track progress)
- 4. Aptveršana
 - Rezultātu analīze (review results)
 - Zināšanu vadīšana (manage knowledge)

Projekta vadības disciplīna

- Projektu vadības disciplīna atspoguļo MSF pieeju pie projektu vadīšanas. Darba organizatorisko lomu un vadību lielākā mērā realizē programmas vadīšanas klasteris.
- MSF projekta vadības procesu secība:
 - Projektēšanas process (ko izveidot?)
 - Plānošanas process (kā izveidot?)
 - Kalendāra grafika izstrāde (kad izveidot?)

MSF nav lomas “Projekta vadītājs”

Projekta vadības aktivitāte tiek sadalīta starp grupu līderiem un lomu klasterim “Programmas vadība”

	Plānošana, monitoring	Projekta ietvaru vadība	Kalendāra plāna vadība	Izmaksas vadība	Personāla vadība	Komunikāciju vadība	Risku vadība	Apgādes vadība	Kvalitātes vadība
Grupu līderi									
Programmas vadība	●	●	●	●	●	●	●	●	●
Produkta vadība	○	○	○		○	●	○		○
Izstrādāšana	○	○	○		○	○	○		○
Testēšana	○	○	○		○	○	○		○
Pasūtītāja apmierināšana	○	○	○		○	○	○		○
Laidiena operācijas	○	○	○		○	○	○	●	○

Risku vadības disciplīna (1)

- Risku vadības disciplīnas etapi:
 - Identifikācija
 - Analīzes un prioritātes definēšana
 - Plānošana
 - Novērošana
 - Kontrole
 - Pieredzes izmantošana

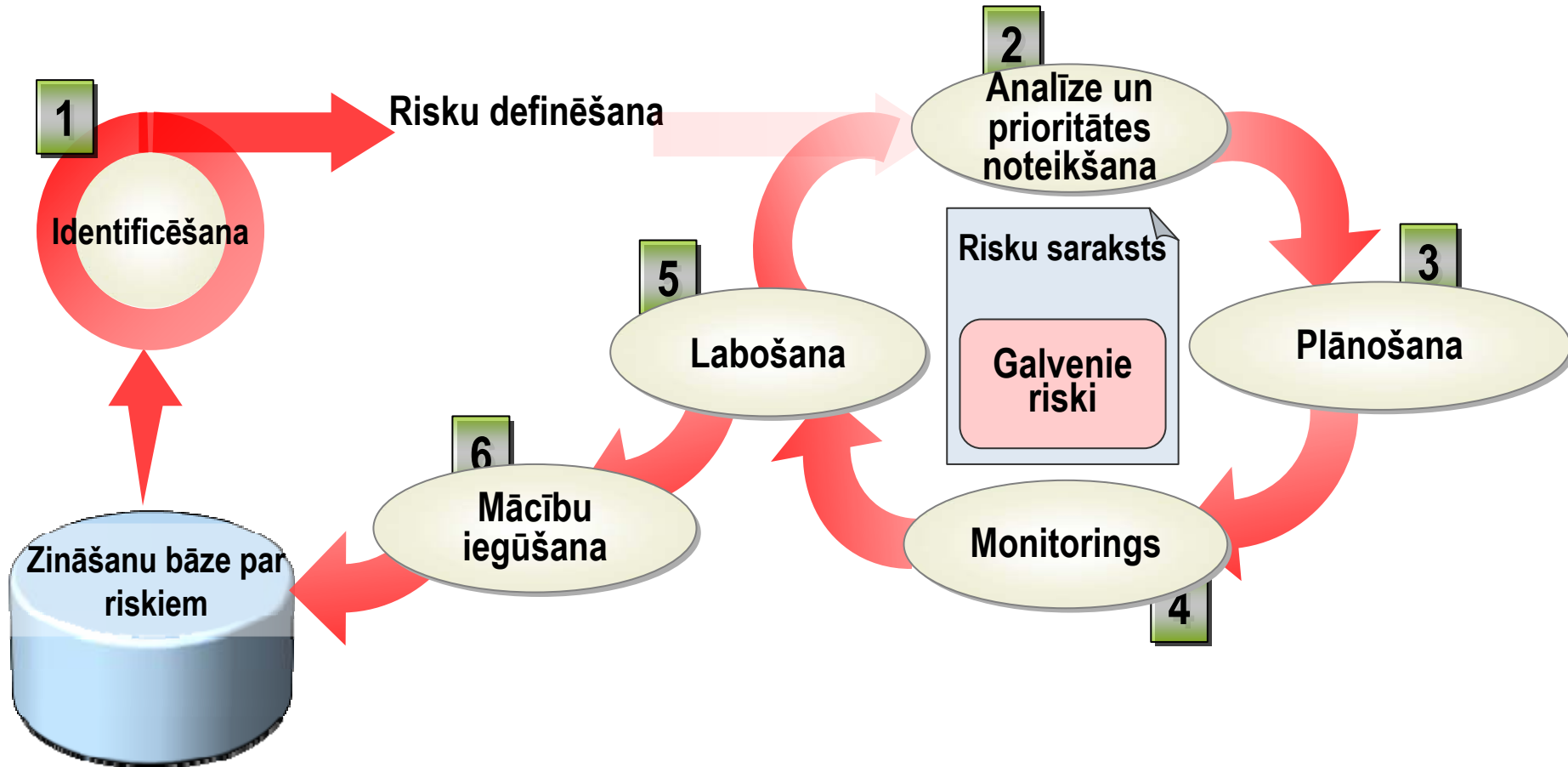
Risku vadības disciplīna (2)

- Iteratīvs process
- Ir izpildīts visa projekta garumā
- Jebkurā projektā eksistē riski
- Orientēts uz risku novēršanu

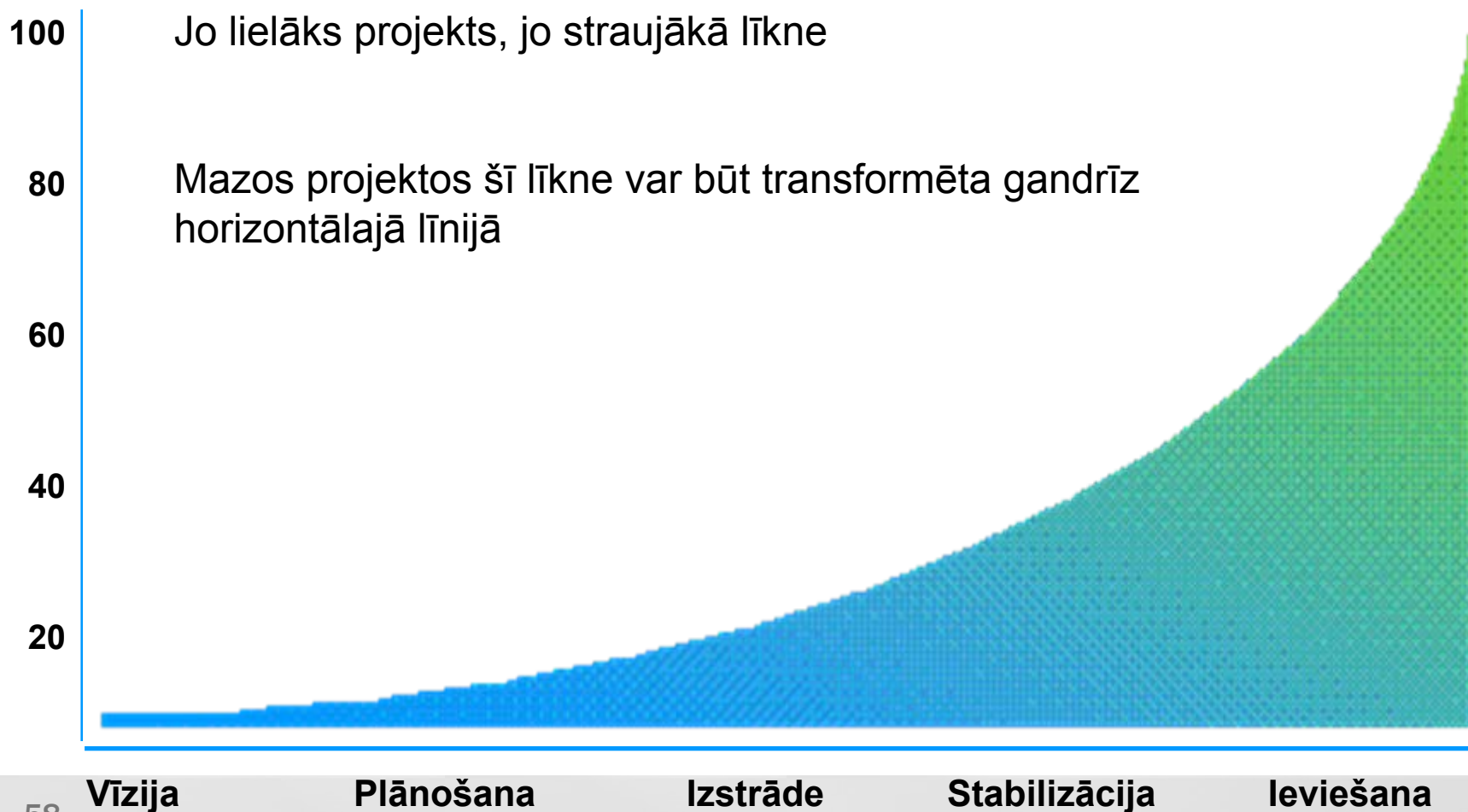


Nav jācīnās ar riskiem – riskus ir jāvada

Risku vadības process

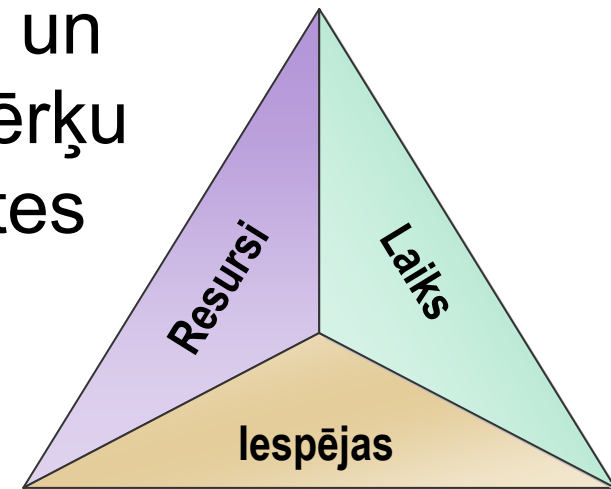


Kļūdu labojumu izmaksas



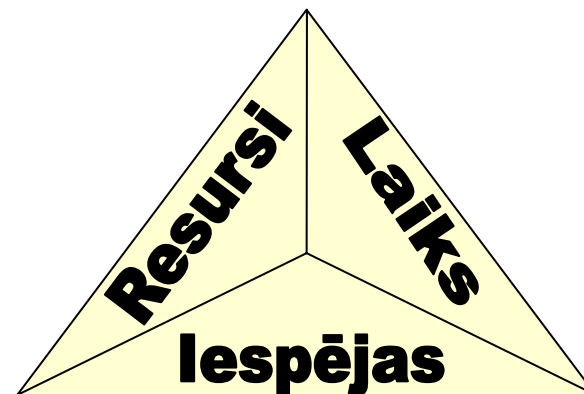
Projektu vadības disciplīna

- Projekts (project) – ir ar laiku ierobežota darbība, ar mērķi izveidot unikālo produktu vai pakalpojumu.
- Projektu vadība (project management) – zināšanu, iemaņu, instrumentu un paņēmienu apgabals projekta mērķu sasniegšanai saskaņotas kvalitātes parametru, budžeta, laika un citu ierobežojumu ietvaros.



Izmaiņu vadība

- Nav iespējams izvairīties no izmaiņām
- Ir iespējams definēt prioritātes, kurus ir jāņem vērā reaģējot uz izmaiņām
- Tiek lietota kompromisu matrica



	Fiksēt	Vienoties	Pieņēm
Resursi	✓		
Laiks		✓	
Iespējas			✓

Nobeigums

- Stingrs un vienlaicīgi elastīgs IT projektu izstrādes process; samazina izstrādes riskus
- Jaunais MSF v4.0 ir atbalstīts ar Visual Studio 2005 Team System
- MSF nenosaka citu Microsoft produktu pielietošanu. Piemēram, programmatūras izstrādes procesa organizēšanai var izmantot MSF un vienlaicīgi lietot Borland rīkus
- MSF dokumentāciju var atrast šeit:
<http://www.microsoft.com/msf>