

TALLINNA ÜLIKOOL

Informaatika Instituut

Kristo Vaher

**LAHINGUSIMULAATORI  
STSENAARIUMITARKVARA SERVEALFONS ANALÜÜS**

Bakalaureusetöö

Juhendaja: Inga Petuhhov

Autor: ..... "....." ..... 2008.a  
Juhendaja: ..... "....." ..... 2008.a  
Instituudi direktor: ..... "....." ..... 2008.a

Tallinn 2008

# SISUKORD

<b>SISSEJUHATUS</b>	<b>4</b>
<b>1. ALFONS 2M TULEJUHI JA LAHINGUSIMULAATOR</b>	<b>6</b>
1.1 LAHINGUSIMULAATORI AJALUGU	7
1.2 ARENDAJA	8
1.3 TELLIJAJA	9
1.4 RIISTVARA	9
1.5 SERVEALFONS	10
1.5.1 STSENAARIUMISÜSTEEMI ÜLDKIRJELDUS	10
1.5.1.1 PÕHISÜNDMUSED MILLELE VASTASE ÜKSUS REAGEERIB	11
1.5.1.2 VASTASE ÜKSUSTE PÕHITEGEVUSED	12
1.5.2 STSENAARIUMIEDITORI ÜLDKIRJELDUS	13
1.6 MUU TARKVARA	14
1.7 PLANEERITUD EDASIARENDUSED 2008 AASTAL	15
1.7.1 VIGADEPARANDUSED	15
1.7.2 UUS JA LAIENEV FUNKTSIONAALSUS	15
1.7.3 ALFONS 2M BAASIL ARENDATAVAD KÕRVALPROJEKTID	15
<b>2. SERVEALFONS'I BACK-END</b>	<b>16</b>
2.1 SÜSTEEMIDE ÜHENDUS JA VÕRGULIIKLUS	16
2.1.1 VÕRGUPAKETID	16
2.1.2 <i>Edasiarendus:</i> A2M KONSOOLI JA MIINIKONSOOLI ÜHENDAMINE	18
2.1.3 <i>Edasiarendus:</i> MIINIKONSOOLI ARVUTUSED SERVEALFONSI ALLA	18
2.2 HEARTBEAT	19
2.2.1 <i>Edasiarendus:</i> HEARTBEAT'I OPTIMISEERIMINE	20
2.3 KORDUSESITUSED	21
2.3.1 <i>Edasiarendus:</i> MUSTA KASTI PÕHINE KORDUSESITUS	21
2.3.2 <i>Edasiarendus:</i> KERIMISVÕIMALUSED	22
2.4 ANIMATSIOONIMOOTOR	23
2.4.1 <i>Edasiarendus:</i> RAG DOLL FÜÜSIKA	23
<b>3. SERVEALFONS'I FRONT-END</b>	<b>25</b>
3.1 KASUTAJALIIDES	26
3.1.1. <i>Edasiarendus:</i> HIIREPÕHINE NAVIGATSIOON JA MENÜÜD	26
3.1.2. <i>Edasiarendus:</i> VISUAALSED TÄIENDUSED	27
3.1.3 <i>Edasiarendus:</i> SERVEALFONS'I SEOS GRAAFIKAMOOTORIGA	28

<b>4. SERVEALFONS JA STSENAARIUMID</b>	<b>30</b>
4.1 STSENAARIUMITE ARENDAMISE LIIDES	30
4.1.1 <i>Edasiarendus</i> : SÕLTUMATUS NOTEPAD TEKSTIREDAKTORIST	30
4.2 STSENAARIUMITE ARENDUSE TURVALISUS	31
4.2.1 <i>Edasiarendus</i> : FUNKTSIOONIPÕHISED SKRIPTID	31
4.3 ILMASTIK	33
4.3.1 <i>Edasiarendus</i> : STSENAARIUMI ILMASTIKUTABEL	33
4.3.2 <i>Edasiarendus</i> : ILMAST SÕLTUV STSENAARIUMIMUDEL	34
<b>KOKKUVÕTE</b>	<b>35</b>
<b>SUMMARY</b>	<b>36</b>
<b>KASUTATUD KIRJANDUS</b>	<b>38</b>

## SISSEJUHATUS

Alfons 2M on Eesti Kaitseväes kasutuses olev lahingusimulaator, mille arendaja on ELI OÜ. Tegu on ühe kõrgtehnoloogilisema väljaõppe treeningu simulaatoriga, mis aktiivkasutuses Eesti lahingüksuste ja tulejuhtide väljaõppes Eesti Kaitseväes.

ELI OÜ aasta 2008 arenduskavas on simulaator Alfons 2M määratud paranduste ja edasiarenduste alla. Selle üks osa hõlmab süsteemi vigade ja puudujääkide parandamist, aga aeganõudvam osa simulaatori edasiarendusplaanist on simulaatori süsteemi laiendamine ja uuendamine. Sellelaadse arendamise tarbeks sai koostatud käesolev töö, mis analüüsib erinevaid Alfons 2M simulaatori põhitarvara ServeAlfonsi osasid ja toob välja võimalikud arengusuunad stsenaariumimootori täiendamiseks, parandamiseks või optimeerimiseks. See on autori poolt püstitatud ka töö peamiseks eesmärgiks. Töös kirjeldatu ja analüüsi põhineb täieulatuslikult arendatud tarkvarasüsteemil ja selle komponentidel.

Olemasoleva lahenduse analüüsimiseks lähenes autor tarkvara ja uute lahenduste leimiseks kolme külje alt. Esiteks uurides kirjutatud tarkvara programmikoode ja nendega seotud tagasisidet, mis saadud süsteemiga tegelenud tarkvara-arendajatelt, eesmärgiga paremini mõista erinevate tarkvaraosade tööd ja sõltuvust teineteisest. Teiseks kasutas autor süsteemioperaatorilt saadetud arvamusi ja tagasisidet. Kolmandaks uuris autor tarkvara enda tööd läbi erinevate katsetuste. Järeldusi ja ettepanekuid teeb autor nende kõigi põhjal, et välja tuua erinevaid võimalusi süsteemi töö parandamiseks.

Käesoleva töö autor osales kaheaastases arendusprotsessis stsenaariumisüsteemi tarkvaraarendajana ning digitaalse meedia spetsialistina. Lisaks oli autor peamine arendusemeeskonna poolne kontaktisik arendajate ja tellijapoolsete konsultantide ning koostöögrupi vahel. Autor on eelnevalt kirjutanud töös käsitletud projekti raames nii lahingusimulaatori tarkvaraosade dokumentatsiooni, kui ka koostanud presentatsiooni-materjale.

Nii bakalaureuse töö tõttu tekkiva laiema lugejaskonna, kui ka Kaitseväge Alfons 2M koostöögrupi huviliste tõttu, käsitleb töö esimene osa süsteemi üldisemat tehnilist kirjeldust, et anda lugejale selgem pilt süsteemi sisemisest struktuurist ja erinevate tarkvara ja riistvara osade koostööst.

Töö teises osas kirjeldab ning analüüsib autor ServeAlfonsi sisemise süsteemi tähtsamate osade tööd ning toob välja nendega seotud võimalikud arengusuunad. Peamine rõhk on Alfons 2M stabiilsust ja simulaatori kiirust puudutavatel teemadel.

Töö kolmandas osas käsitleb autor peamiselt ServeAlfonsi kasutajaliidest nii operaatore, kui ka stsenaariumite arendaja vaatepunktist ja toob välja mitmeid suundi ServeAlfonsi kasutajaliidese kasutamise mugavuse tõstmiseks.

Töö neljandas osas keskendub autor ServeAlfonsi stsenaariumite loomisprotsessiga seotud komponentide uutele lahendustele, nende seas ka stsenaariumisüsteemi laienemist mis suurendaks üksiku stsenaariumi võimalusi märgatavalt.

Töö lõpuosas võtab autor kokku eelnevalt käsitletu ja hindab tööle püstitatud eesmärkide täitmist ning toob välja töö edasiarendamise ja laienemise võimalused.

Käesolev töö läheb firmas kasutusse edasiarenduste tarbeks tehtud uurimustööna ja töö võetakse aluseks lahingusimulaatorisse erinevate uute lahenduste lisamiseks ja vanade optimeerimiseks.

# 1. ALFONS 2M TULEJUHI JA LAHINGUSIMULAATOR

Alfons 2M on samanimeliste ELI OÜ simulaatorite põlvkonna uusima tarkvara ja riistvaralahendusega lahingutegevuse, miinipilduja ning tulejuhi simulaator.

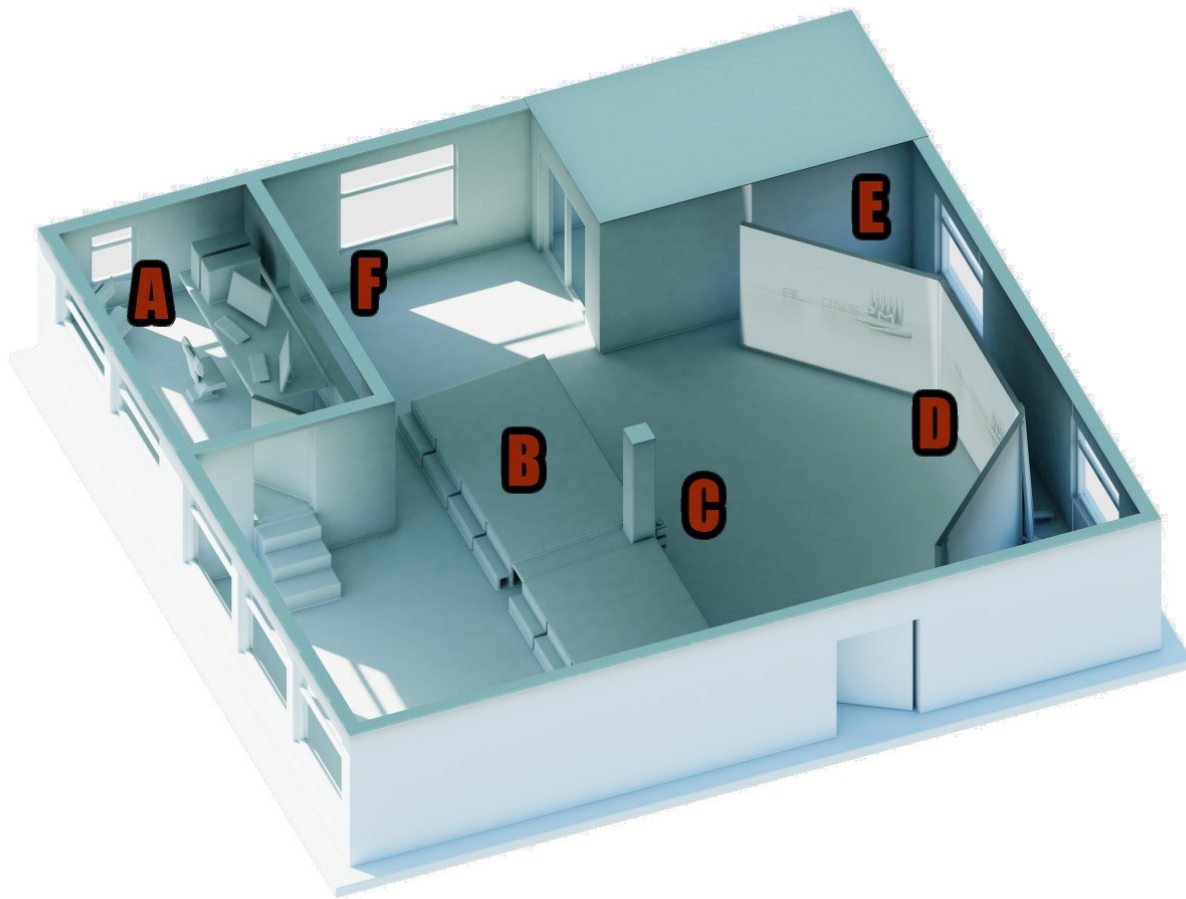
Simulaatori keskmeks on virtuaalne keskkond, mille teostamisel on aluseks võetud reaalne Eestis asuv maastik teede, ristmike, metsatukkade ning lagendikega. Kaitseväge poolt eraldi simulaatori stsenaariumite toimumispaigaks väljavalitud piirkond Eestis oli Meeri küla ja selle ümbrus Elva lähedal, 2x2 km suurune ala. Meeris ja selle lähedal reaalselt olemasolevad hooned ja objektid ehitati fotode baasil 3D mudeliteks ümber ning püstitati aerofotode ja keskkonna pildimaterjali järgi võimalikult elulähedaselt virtuaalsele kaardile.

Simulaator käsitleb tulejuhtimist ja erinevat kaitselahingutegevust virtuaalsete soomukite ja jalaväe üksuste vastu, mille tegevust juhitakse tarkvaramootori ServeAlfons kaudu. Võrreldes eelmise generatsiooni simulaatoriga sisaldab Alfons 2M modernset miinipilduja simulaatorit ning täielikku 3D keskkonda.

Alfons 2M simulaator esitab virtuaalse keskkonna treenitavale üksusele läbi LCD projektorite ning arvutite, mis tekitavad omavahel sünkroonis oleva pildi suurele kolme osaga ekraanile. Lisaks on simulaatoriga seotud serveri arvutid, mis kogu tegevust juhivad.

Simulaatoril on võimalik kasutada nii käsitulirelvi, automaate, kuulipildujaid, kui ka tankitõrje relvastust ning simulaator võimaldab tulejuhil tellida ja suunata kaugtuld virtuaalsel maastikul, mis vastab tegelikele teede- ning Kaitseväge kaartidele. Simulaatoris kasutatakse ehtsaid lahingrelvi, mis Kaitseväge poolt deaktiveeritud ja milles on kasutusel tagasilööki imiteeriv süsteem. Simulaator võimaldab kuni üheksa laskepositsiooni kasutamist üheaegselt. Lisaks on simulaatoris kasutusel veel laserkaugusmõõdik kauguste hindamiseks virtuaalsel maastikul.

Eesti Kaitseväge kasutab simulaatorit Alfons 2M tulejuhtide ja nende allüksuste treenimiseks. Alfons 2M esimene täiskomplekt on üles seatud Kaitseväge Ühendatud Õppeasutustesse, mille tarbeks ehitati ümber eraldi ruum (vt Joonis 1).



*Joonis 1. ALFONS 2M RUUMI ILLUSTRATSIOON*

A – operaatoriruum, Miinikonsooli arvuti ning A2M Konsooli ja ServeAlfonsi arvuti. B – laskeplatvorm (üheksa positsiooni)  
 C – kolm graafikamootori arvutit, projektorid ja sensorkaamerad D – kolm ekraani E – kõlarid (ekraanide taga) F – relvakapid

## 1.1 LAHINGUSIMULAATORI AJALUGU

Jalaväe treeningsimulaatori esimene versioon valmis viieaastase arendustsükli käigus. Simulaatori arendaja oli ELI OÜ poolt palgatud üksikarendaja Toomas Tõnisson, kes lõpetas simulaatori esialgse arengutsükli aastal 2002. Esimese Alfonsi eesmärk oli võimaldada Eesti Kaitsevägele vastupidav ja rahasäästev treeningu-simulaator, mis võimaldaks laiendada treenitavate üksuste relvakäsitlemise oskust.

Simulaator koosnes ühest arvutist ja sellega ühendatud LCD projektorist, mis kuvas valgele seinale virtuaalse lahingukeskkonna: foto, mida mööda liikusid 2D sprite<sup>1</sup> tehnoloogial põhinevad

<sup>1</sup> - 2D Sprite on pilt või animatsioon, mis integreeritud suurema stseeni või pildi sisse.

kujutised. Relvad, mida simulaatoris kasutati, olid ehtsad lahingurelvad, mis läbinud Kaitsejõudude poolt deaktiveerimise ja millele ELI OÜ poolt paigaldati sisse spetsiaalsed väljaarendatud tagasilöögimehhanismid. Relvade külge oli kinnitatud laserid, mida kasutati laskmise hetkel relva sihitud punkti asukoha määramiseks. Relvad, arvuti ja kaamerad olid omavahel ühenduses läbi musta kasti, mis edastas peaarvutile informatsioonipakette laskude asukohtade ja ajahetkedega.

Esimene Alfons pandi üles mitmesse Eesti Kaitsejõudude pataljonidesse ja jalaväe lahingõppe treeningkeskustesse. Alfons simulaator sai tunnustust Kaitsejõududes, aga peale mitmeaastast kasutuses olemist tekkisid kasutusest mitmed probleemid ja simulaatorid nõudsid spetsiaalset hooldust või seadistamist. Aja möödudes ilmnisid ka otsesed märgid tarkvara vananemisest. Graafikamootor oli väga aegunud ja simulaator ei leidnud suuremat kasutust, kui kiire kuivtreeningu-platvorm.

2005 aasta lõpus esitas ELI OÜ Kaitseministeeriumile esialgse projektiplaani uue simulaatori Alfons 2M arendamiseks (täht 'M' tähistab Miinipildujasimulaatori tuge), eesmärgiga kasutada simulaatoris uuemaid ja töökindlamaid riistvaralisi lahendusi ning täies ulatuses ümber tehtud tarkvaratehnoloogiat. Projekt leidis finantseerimist ning Alfons 2M'i tarkvara arendusmeeskond kasvas ühelt arendajalt neljale. Edasi järgnes üheksateist kuud kestnud intensiivne arendusperiood, mille käigus valmis komplektne Alfons 2M.

Kuue kuu jooksul peale simulaatori installeerimist anti arendajatele aktiivset tagasisidet tarkvaraliste ja riistvaraliste probleemide kohta ja edasine arendustegevus selle aja jooksul toimus peamiselt erinevate vigade elimineerimiseks. Aastal 2008 on planeeritud simulaatori Alfons 2M edasiarendus.

## **1.2 ARENDAJA**

ELI OÜ on baltimaade juhtiv militaar-simulatsioonide ja mehitamata lennukite uurimus- ja arendustegevusega tegelev väikefirma kellel on 9 täiskohaga töötajat ja mitmed lepingupartnerid. Simulatsioonide ja simulaatorite arendamist alustas ELI 1995 aastal, võttes peamiseks suunaks simulaatorid tulirelvadele ning tankitõrje lahingvarustusele. ELI on läbi aastate saanud tunnustust Euroopas, tutvustades oma tooteid rahvusvahelistel näitustel (nende seas Eurosatory 2002, 2004, 2006). ELI OÜ nimetati ajakirja Military Training Technology poolt rahvusvaheliselt aastal 2004 viiekümne kõige enam militaar-treeningut mõjutanud firma nimekirja [1.].

Alfons 2M arendus hõlmas endas kogu firma osalust, serveri ja stsenaariumisüsteemi ServeAlfonsi ja teiste tarkvarapoolsete osadega tegeles neli tarkvara-arendajat, kes spetsialiseerusid erinevatele süsteemiosadele.

### **1.3 TELLIJAJA**

Simulaator arendati välja Eesti Kaitsejõududele, otsene Alfons 2M'i tellija ja finantseerija oli Eesti Kaitseministeerium, kes tellis simulaatori Kaitseväe Ühendatud Õppeasutustele Tartus. Eesti Kaitseministeerium määras arenduse tingimuste täitmiseks ja taktikaliste küsimuste tagasisideks spetsiaalse töögrupi ja hoidis silma peale ülejäänud arendusprotsessi majanduse poolel.

Simulaatori arenduse käigus konsulteerisid tellija poolt määratud Kaitseväe koostöögrupi liikmed ELI OÜ arendusmeeskonda, kes koostasid esialgsed stsenaariumi ideed ning testisid tarkvara ja stsenaariumite nõuetele vastavust. Üks konsultantidest, Andres Pall, muutus arendusprotsessi käigus simulaatori igapäevaseks operaatoriks, kellega on simulaatori ülessepaneku hetkest jätkunud põhiline arendaja-tellijaja vaheline arendustegevusega seotud suhtlus.

Alfons 2M'i füüsiliseks simulaatori ruumiks ehitati tellija poolt ümber Kaitseväe Ühendatud Õppeasutuste üks loengusaalidest, mis vastas arendaja poolt määratud simulaatori töökeskkonna vajadustele.

### **1.4 RIISTVARAJA**

Simulaatori täieulatuslikuks tööks kasutatakse viite arvutit, millest kolm tegelevad graafikamootori virtuaalse maailma pildi renderdamisega<sup>2</sup>, kuvades selle ekraanidele läbi LCD projektorite. Simulaatorit on võimalik ka üles panna kasutades ainult ühte graafika renderdamiseks mõeldud arvutit ja ekraani, aga see piirab treenitava üksuse lahingukeskkonna vaatevälja esialgsete stsenaariumite läbiharjutamisel.

Neljas arvuti on mõeldud miinipilduja konsooliks ja viies arvuti on ServeAlfonsi süsteemi ja seetõttu ka kogu Alfons2M simulaatori keskuseks. Riistvaraliselt on simulaatori juures veel ehtsad deaktiveeritud tulirelvad<sup>3</sup>, mis ELI OÜ poolt riistvaraliselt veel omakorda modifitseeritud. Relvade sisse on installeeritud CO<sub>2</sub> gaasi põhine tagasilöögimehhanism, mida simulaator kasutab laskude

---

2 Renderdamine on protsess kus arvuti genereerib pildi või pildiseeria 3D mudelitest, keskkonnast, tekstuuridest sprite'dest või muudest komponentidest.

3 Deaktiveeritud lahingrelv on tulirelv, mis on tehtud lahingvõimetuks kas relvaluku eemaldamise või poltide läbilöömise tulemusel relva vintrauast.

hetkel reaalse tagasilöögi tekitamiseks laskjale. Relvadele on kinnitatud ka punase diodiga laserid, mis aktiveeritakse lasu toimimise hetkel, tekitades murdsekundiks ereda punase täpi simulaatori ekraanil.

Ekraanidelt loevad täpi sisse kõrge pildisalvestamise sagedusega sensor-kaamerad, mille poolt salvestatud pilti reaalajas analüüsib süsteemi 'Must Kast'. Must Kast on eraldiseisev komponent, mille külge on kinnitatud simulaatori relvad ning läbi võrgukaardi ka ServeAlfons keskus. Lasu hetkel analüüsib Must Kast pildil salvestatud lasertäppi ja saadab lasu ekraanikoordinaadi edasi simulaatori ServeAlfonsile.

## **1.5 SERVEALFONS**

Alfons 2M'i tarkvara tähtsaim osa on ServeAlfons, mis on ühenduses otseselt või läbi teise süsteemi iga riistvara ja tarkvara komponendiga. ServeAlfons suhtleb graafikamootoriga saates neile käske muutuste kohta, mida teha virtuaalses maailmas, saab käske miinikonsoolilt ja analüüsib infot, mis edastatud Musta Kasti poolt.

ServeAlfons on kirjutatud programmeerimiskeeles Python. Pythoni eelis primaarse keele valikul oli selle süntaksi puhtus ning arendusmeeskonna positiivsed kogemused Pythoniga enne projekti algust. Lisaks aitasid Pythoni kergesti ja kiiresti õpitavad omadused kaasa stsenaariumite skriptide<sup>4</sup> kirjutamisel.

### **1.5.1 STSENAARIUMISÜSTEEMI ÜLDKIRJELDUS**

Kõige pealmine kiht Alfons 2M ServeAlfons'il on stsenaariumi skriptid. Stsenaarium on eraldiseisev Python-skript, milles on defineeritud stsenaariumi maailmatingimused, lahinguüksused ja teekonnad kaardil ning lahinguüksuste tegevus vastavalt stsenaariumis toimuvatele sündmustele ja treeningüksuse tegevusele interaktiivses virtuaalses keskkonnas.

Stsenaariumisüsteemi tähtsaim osa on ettemääratud teed, mida võib ette kujutada maastikukaardile asetatud graafidena, koosnedes kaartega ühendatud tippudest. Iga teekonnapunkt talletab endas ühte või mitut baaskäsku ja iga teekonnapunkt on servadega ühendatud maksimaalselt kahe teise punktiga. Baaskäsud edastatakse stsenaariumis liikuvatele üksustele ning nende baaskäskude kaudu juhitakse stsenaariumis ettemääratud vastase tegevust. Stsenaariumite peamine erinevus teineteisest on teekonnapunktide asukohad ja punktides skriptidena salvestatud baaskäsud.

---

<sup>4</sup> Skriptide all peetakse käesoleva töö puhul silmas Python programmeerimiskeeles kirjutatud programmi osa, funktsiooni või algoritmi.

Baaskäsud on lühikesed Python skriptid, mida käivitatakse siis, kui tipus defineeritud sündmus stsenaariumi käigus aset leiab. Kui virtuaalne üksus jõuab teekonnapunkti, siis kopeeritakse üksusele käsud, mis talletatud järgmises teekonnapunktis. Neid käske kannab üksus endaga kaasas hetkeni, mil saadakse sama klassi käsk, mis juba olemasoleva üle kirjutab.

Stsenaariumi üksustele on määratud ka universaalsed käsud, mida täidetakse sõltumata punktides saadud käskudest. Sedasorti universaalseteks käskudeks on näiteks sõduri langemine kuuli pihtasaamisel või pikaliviskamine miinipilduja plahvatuste korral.

Stsenaariumi teedel liiguvad peamiselt kahte tüüpi üksused. Esiteks on jalaväeüksused, mis koosnevad jalaväe jagudest ja liiguvad ühes kindlas formatsioonis. Jalaväeüksus koosneb eraldi virtuaalsetest sõduritest, kes töötavad koos ühe üksusena. Teekonnapunktides antud baaskäsud määratakse tervele üksusele ja selle tulemusel käitub iga üksuse sõdur vastavalt baaskäskudest tekkinud muudatustele. Näiteks üksusele baaskäsu kaudu antud käsk formatsiooni muutmiseks, mille tulemusel üksuse klassi jaotab formatsiooni käsu igale sõdurile eraldi alamkäskudeks, mille tulemusel hakkavad üksuse virtuaalsed sõdurid liikudes hoidma uut individuaalset kohta formatsioonis.

Sedasorti skripti-põhine stsenaariumisüsteem on hea ettemääratud lahingutegevuse esitlemiseks, sobides kõige paremini treeningsimulaatorisse, milles täpne ettemääratud tegevus on vajalik ja tellija poolt esitatud nõue. Samas on ServeAlfonsi stsenaariumimootor piisavalt automatiseeritud, et ei peaks kõike toimuvat stsenaariumiskripti detailselt sisse kirjutama, võimaldades kergemate stsenaariumite loomist arendaja poolt väga lühikese aja jooksul.

#### ***1.5.1.1 PÕHISÜNDMUSED MILLELE VASTASE ÜKSUS REAGEERIB***

Iga teekonnapunktis antakse üksusele teatud grupp käske, mille toimumist üksus ootab kuni see käsk üle kirjutatakse, nagu mainitud punktis 1.5.1. Käsud, mida üks või teine vastase üksus teekonnapunktis saada võib, on täiesti sõltumatud grupist endast. Käsk võib endas sätestada nii kõikide üksuste tegevuse jälgimist, kui ka spetsiifilise üksuse jälgimist. Iga sündmuse toimimise hetkel kuulutab sündmust kogunud grupp, et üksus seda koges. Neid kuulutusi kontrollivad põhisündmuse skriptid ja skripti tingimuse täitmisel määravad üksuse käituma vastavalt uuele käsule.

Süsteem on programmeerimise seisukohalt täiesti lahtine, üks vastase üksus võib kuulata teise üksusega toimunud sündmust ja selle tulemusel hoopis kolmanda üksuse panna käituma vastavalt punktis 1.5.1.2 mainitud tegevustele.

Üksustele määratud käsud, mille toimumist üksus kuulama hakkab on järgmised:

- Sõdur X kuulutab, et ta sai pihta  
*sõdur võib olla nii tema ise kui keegi teine*
- Esimene meiepoolne lask  
*toimub iga stsenaariumi käivitusel ainult üks kord ja on mõeldud vastase üksuste esialgse reaktsiooni kontrollimiseks*
- Üksus X kuulutab, et nad on tule all / Soomuk kuulutab, et ta on tankitõrje tule all  
*kui üksuse ühest või mitmest sõdurist mööduvad lähi-raadiuses kuulid, kuulutab üksus, et nemad on tule all, soomuki puhul tankitõrje mürsk*
- Üksus X kuulutab, et nad on miinipilduja tule all  
*kui üksuse ühele või mitmele sõdurile langeb lähedusse miin siis kuulutab üksus, et nad on miinipilduja tule all*
- Üksus kuulutab, et ta märkas õist valgusraketti  
*öösetes lahingustsenaariumites on valgusraketi kasutamisel vastane kiire reageerima ja treenitava üksuse poolt tellitud valgusraketi tulemusel võib selle käsu läbi vastase üksus tegevust muuta*
- Alguskäsk  
*skript, mis käivitatakse kohe kui üksusele on määratud teekonnapunkt millele alguskäsk selle grupiga seotult juurde lisatud on*

### **1.5.1.2 VASTASE ÜKSUSTE PÕHITEGEVUSED**

Vastase üksused võivad teha ühte või mitut nimetatud tegevust sõltuvalt punktis 1.5.1.1 määratud sündmustele ja nende baasil kirjutatud skriptile.

- Üksus sihhib kükakil või lamades positsiooni X suunas
- Üksus roomab / Soomuk sõidab jalaväe kiirusega
- Üksus roomab kiiresti
- Üksus vahetab formatsiooni
- Üksus vahetab teekonnapunkti, mille poole liigub
- Üksus kõnnib / Soomuk sõidab keskmise kiirusega
- Üksus jookseb / Soomuk sõidab kiiresti
- Üksus viskab enda positsiooni ette suitsugranaadi
- Üksus avab tule

- Soomuk pöörab kahuri punkti X suunas
- Soomuk vahetab edaspidi-tagurpidi käiku

Lisaks on süsteemis määratud maailmasündmused, mida stsenaariumis võib kasutada nii üksustepõhiste sündmuste, kui ka universaalsete sündmuste juures. Peamiselt on tegu erinevate otseselt üksusest sõltumatute stsenaariumivahenditega, mida kasutatakse kaardiväliste üksuste tegevuse simuleerimiseks, nagu vastase miinipilduja tuli:

- Miiniplahvatus positsioonis X
- Miinipilduja tuli positsioonis X
- Suitsumiin positsioonis X
- Valgusrakett positsioonis X

### 1.5.2 STSENAARIUMIEDITORI<sup>5</sup> ÜLDKIRJELDUS

Iga stsenaarium põhineb esialgsel Pythonis programmi baaskoodil, mille sees on erinevad osad, mida võimalik muuta läbi ServeAlfonsi visuaalse kasutajaliidese. Seda võib ette kujutada kui lünkteksti täitmist, millel lünkadesse kirjutatakse erinevad teekonnapunktide käsud, defineeritakse vastase üksused ja seadistatakse maailmaparameetrid. ServeAlfonsi kasutajaliides muudab parajasti käimasolevat stsenaariumi, see omakorda tähendab, et stsenaariumis tehtud muutused on nähtavad koheselt reaajas ja ei nõua eraldi kompileerimist või väliselt käivitamist. Kui tehti muudatus teekonnapunktides kuhu enam üksused ei jõua, võib muudetud stsenaariumi salvestada ja uuesti programmist väljumata stsenaariumi ümberlaadida.

Stsenaariumite muutmise ja uute loomise toimub läbi teekonnapunktide ja nendest koostatud teede asetamise kaardile. Lisaks asetatakse kaardile virtuaalsed üksused ja seotakse need teekonnapunktiga, igal lahingüksusel võib korraga eesmärgiks olla vaid üks teekonnapunkt. Üksused võivad teid ja teekonnapunkte jagada ning teekonnapunktides salvestatud käsud võivad olla nii universaalsed ja määratud kõikidele üksustele, kui ka individuaalsed, eristatud üksteisest gruppide identifikaator-numbrite kaudu. Üksusele ja valitud teepunktile skripti lisamine algab baastingimuse valikust mis skriptis kirjeldatu käivitab. Skript ise kirjutatakse valmis Notepad tekstiredaktoris, mille ServeAlfons automaatselt käivitab, kirjutatud programmikood salvestatakse ning kopeeritakse ServeAlfonsist automaatselt stsenaariumi põhikoodi sisse.

---

<sup>5</sup> Stsenaariumieditor on stsenaariumi muutmise või loomise tarkvara

Stsenaariumi editor on samas arendusprogramm ja uute stsenaariumite loomine või olemasolevate stsenaariumite muutmine on esialgselt projektilepingu kohaselt puhtalt arendajapoolne tegevus, mis tähendab, et ServeAlfonsi funktsionaalsus tellijale seisneb puhtalt stsenaariumite käivitamise ja majandamisega. Süsteemist on puudu lihtsasti kasutatav liides tellijale ServeAlfonsi stsenaariumite loomiseks.

## 1.6 MUU TARKVARA

ServeAlfonsi juurde käivad erinevad lisatarkvara programmid, milleta üks või teine simulaatori süsteem töötada ei saa. Miinikonsooli tarkvara on interneti brauseri põhine konsool, kuhu süsteemioperaator sisestab miinipilduja edastatud käsu, mida siis Miinikonsool edastab virtuaalsele miinipilduja meeskonnale. Virtuaalse meeskonna miinide laadimist, sihtimist ja miinide lendu simuleeritakse reaalse parameetrite alusel. Miinikonsool ja selle simuleeritud miinide lennuparameetrid ja sisestatavad tulekäsud on välja arendatud koos eraldiseisva Kaitseväge koostöörühmaga.

ServeAlfons suhtleb veel A2M Konsooliga, mis on teine interneti brauseril töötav keskkond ja on kasutusel ainult süsteemi pealiskaudsete käskude sisselugemiseks. Peamised A2M Konsooli tööülesanded on kasutajale visuaalne stsenaariumite valiku võimalus ja võimalus neid käivitada, käimasolevat stsenaariumi salvestada või salvestatut kordusesitada. A2M Konsoolis on ka valik relvapositsioonidele relvatüübi valik, mille järgi ServeAlfons teab kuidas sünteesida Mustast Kastist edastatud laskeinformatsiooni.

Viimane tarkvara element Alfons 2M simulaatorisüsteemis on graafikat renderdavatele arvutitele paigaldatud graafikamootor. Alfons 2M graafikamootorina kasutatakse vaba lähtekoodiga Ogre 3D graafikamootorit<sup>6</sup>, mis osutus parimaks lahenduseks, arvestades arenduseks määratud aega ja arendusmeeskonna suurust. Ogre 3D mootorit arenduse käigus ei muudetud, arendajameeskonna poolt lisati sellele liides ServeAlfonsi poolt esitatud käskude tõlkimiseks ja esitlemiseks 3D keskkonnas.

---

6 Ogre 3D on lahingusimulaatoris kasutatud GNU Lesser Public License all

## **1.7 PLANEERITUD EDASIARENDUSED 2008 AASTAL**

### **1.7.1 VIGADEPARANDUSED**

Alfons 2M kasutamisel Kaitseväes tekib erinevaid tarkvara ja riistvara poolseid vigu, mis arendustsükli pikkuse tõttu ei saanud parandatud või süsteemist tuvastatud enne simulaatori ülespanekut Tartus, 2007 aasta augustis. Simulaatori kasutamise käigus saadetakse Tartust vigaderaporteid ja süsteemi logifaile arendajatele, mida kasutatakse vigade parandamiseks.

Vigadeparandused ja arendajapoolne kasutajatugi ei maksa tellijale eraldi, olles kaetud aastase garantiilepingu poolt ja on arendaja-tellija vaheline loomulik protsess.

### **1.7.2 UUS JA LAIENEV FUNKTSIONAALSUS**

Alfons 2M on planeeritud laienema 2008 aastal, täiendades või ümber tehes erinevat simulaatori funktsionaalsust eesmärgiga tõsta süsteemi stabiilsust ning tõstes simulaatori efektiivsust Eesti Kaitseväe jalaväeüksuste väljatreenimisel.

Suurem osa edasiarendusest on planeeritud stsenaariumite valiku laienemise osas, mida esialgses simulaatoris on 12. Stsenaariumite valiku suurenemine oleks kliendi tagasiside põhjal üks tähtsamaid osasid kinnistamiseks Alfons 2M'i kui elutähtsat tulejuhtimise väljaõppe osa Eesti Kaitseväes.

Stsenaariumite laienemise kõrval on planeeritud ka tõsta süsteemi stabiilsust ning optimeerida erinevaid süsteemisestest osade omavahelist tööd.

### **1.7.3 ALFONS 2M BAASIL ARENDATAVAD KÕRVALPROJEKTID**

Alfons 2M on väga lahtine simulaatoripakett, mis tähendab seda, et simulaatorile uute relvade lisamine või simulaatori ühe või teise osa kasutamist väiksemate projektide juures laiendab simulaatori võimalusi suuremal turul. Alfons 2M'i põhjal on aasta 2008 edasiarenduses planeeritud ühe ekraaniga tankitõrje harjutus-simulaator, mille riistvaralised ja ruumi nõudmised Alfons 2M'ist tunduvalt väiksemad, aga mis samal ajal saab kasu simulaatori detailsest virtuaalsest keskkonnast ja stsenaariumisüsteemist.

Lisaks on Alfons 2M'i esitletud naaberriikides kui potentsiaalset modernset treening-simulaatorit ja simulaator on saanud positiivset tagasisidet, nii esitletud riistvara, kui ka tarkvara osas. Täpsemaid simulaatori rahvusvahelisi laienemisi kirjutamise hetkel aasta 2008 arengukavasse pole planeeritud.

## 2. SERVEALFONS'I BACK-END<sup>7</sup>

### 2.1 SÜSTEEMIDE ÜHENDUS JA VÕRGULIIKLUS

Alfons 2M simulaatori arvutid on omavahel ühendatud baaskomplektis läbi TRENDnet® ruuteri kohalikus ja piiratud ligipääsuga võrgus. Erinevate riistvara ja tarkvara elementide vahel toimub aktiivne side, mis kõik omavahel suhtlevad ServeAlfons stsenaariumimootoriga.

#### 2.1.1 VÕRGUPAKETID

Võrgupakettide all kirjeldatakse lühidalt lahingusimulaatori komponentide vahelist üldist informatsiooni liiklemise tööpõhimõtet praegusel kujul. Defineeritud on esialgne komponent ja komponent kuhu pakett saadetakse ning lühidalt on kirjeldatud informatsiooni, mida pakett sisaldab.

- ***Must Kast -> ServeAlfons***

Milline sensor-kaamera (ja seetõttu millisel ekraanil) tuvastas lasertäpi, millise identifikaatoriga relv lasu tekitas ning mis oli lasertäpi koordinaat 1024x768 pildil. Informatsiooni täpsus sõltub tugevalt laseri seadistamisest ja teravusest.

- ***ServeAlfons -> A2M Konsool***

ServeAlfons kogub kokku konfiguratsioonifailidest informatsiooni stsenaariumite kirjelduste ja klassifikatsioonide ja salvestatud korduste kohta, mida A2M Konsool esitab kasutajaliideses vastavate nimekirjadena.

- ***A2M Konsool -> ServeAlfons***

A2M konsool saadab ServeAlfonsisse ainult informatsiooni valitud stsenaariumi kohta ning relvade tüüpide ja nende identifikaatorite seose kohta, et süsteem teaks millise ID'ga on määratud milline relv. Seda informatsiooni kasutab ServeAlfons stsenaariumi laadimiseks ning Mustast Kastist saadud informatsiooni korrektseks töötlemiseks.

- ***Miinikonsool -> ServeAlfons***

Kuna Miinikonsool töötab suhteliselt iseseisvalt (miinipilduja käsud, miinipilduja

---

<sup>7</sup> Back-end on tarkvara sisemine osa, mis töötleb kasutaja poolt sisestatud või teisiti genereeritud andmeid ja suhtleb süsteemisiseselt teiste komponentidega.

meeskonna simuleerimine, lennuajad ja tabamuskohad on kõik lokaalselt Miinikonsoolis arvutatud) siis saadab Miinikonsool ServeAlfonsile ainult asukohta 2x2km ruudul hetkel millal miin tabab asukohta kaardil, täpsusega 1m.

- ***ServeAlfons -> Miinikonsool***

ServeAlfons saadab miinikonsooli ainult informatsiooni stsenaariumi nime, algus ning lõppaja kohta.

- ***ServeAlfons -> Graafikamootor(id)***

ServeAlfons saadab arvutitesse informatsiooni üksuste positsioonide ja animatsiooni-kaadrite kohta 30 korda sekundis. Plahvatuste ja suitsuefektide loomise käsud saadetakse ainult kaadritesse hetkel, millal neid graafikamootorid visuaalselt esitama peavad ja stsenaariumi kaardi ja ilmastikutingimuste pakett saadetakse ainult stsenaariumi käivitamisel. Mitme graafikamasina sünkroonset tööd kontrollib 'sync' pakett, mis iga kaader identifikaatoriga saadetakse.

Saadest informatsiooni graafikamootoritelt tabamuste kohta virtuaalses ruumis, kohandab ServeAlfons graafikamootoritele saadetavaid käsked vastavalt, näiteks sõduri tabamuse korral saadetakse sõdurile pihtasaamisega seotud animatsioon.

ServeAlfons saadab graafikamootoritesse ka informatsiooni lasu tekitamise kohta, saates graafikamootoritele käsu punkti kohta kuhu virtuaalne lask tekitada. Lasu tabamust hindab graafikamootor, mis tegeliku ballistikakurvi kaudu simuleerib virtuaalse kuuli lendu ning tabamusi.

- ***Graafikamootor(id) -> ServeAlfons***

Kuna graafikamootorites on salvestatud erinevate relvatüüpide ja kuulide ballistika informatsioon lennu kohta, siis toimuvad kuulide arvutused graafikamootorites. Punkti asukohas, mille Must Kast saatis ServeAlfonsisse ja sealt edasi ServeAlfonsi poolt graafikamootorisse, tekitab graafikamootor virtuaalse lasu. Tabamuse korral saadab graafikamootor tagasi ServeAlfonsisse informatsiooni, mis objekti virtuaalses ruumis tabati.

### **2.1.2 Edasiarendus: A2M KONSOOLI JA MIINIKONSOOLI ÜHENDAMINE**

Kuna Alfons 2M simulaatorit kontrollib üks operaator, siis otsus eraldada A2M süsteemikonsool ja Miinikonsool kahe erineva arvuti peale, on toonud esimeste kasutuskuude jooksul tagasilööke. Kasutades tarkvara ise ning saades tagasisidet operaatorilt selgus, et kasutajal on ebamugav kasutada üheaegselt kahte arvutit ja konsooli korraga ilma, et sellel oleks nähtavat vajadust. Lisaks oli kahe süsteemi stabiilsus koostöös nõrgem ning mitmeid probleeme oli Miinikonsooli arvuti võrguühendusega.

Uurides süsteemi komponentide omavahelist tööd selgus, et A2M Konsool ja Miinikonsool on üles ehitatud sarnasel baassüsteemil ja mõlemad on kontrollitavad läbi Javascript'i toega interneti brauseri. Selle tulemusel oleks kahe süsteemi ühendamine samm lähemal süsteemi ühtlustamiseks ning ei nõuaks pikka arendustegevust, samal ajal lihtsustades süsteemioperaatori tööd märgatavalt.

***Peamised eelised:*** Kahe konsooli ühendamine võimaldab operaatoril keskenduda tööle ühe konsooliarvutiga ja seetõttu lihtsustada operaatori tööd simulaatorit juhtides.

***Võimalikud tagasilöögid:*** Vähemärgatav ServeAlfonsi protsessorinõudluse suurenemine lisakohustuste tõttu.

### **2.1.3 Edasiarendus: MIINIKONSOOLI ARVUTUSED SERVEALFONS'I ALLA**

Miinikonsooli üks tööülesannetest on simuleerida virtuaalse miinipilduja-meeskonna tööd ja miinide lennuaega ning tabavust vastavalt miinikonsoolile sisestatud andmete järgi. Tehtud testimise käigus esinesid vähestel juhtudel võrguühenduse probleemid ning olukorrad, kus simuleeritud miinid ei jõua reaalselt stsenaariumi ka siis, kui käsk on antud või jõuavad sinna erinevatel hetkedel. Leidsin, et selle peamine põhjus on see, et Miinikonsooli ei hoita sünkroonis ei ServeAlfonsis ega graafikamootorites toimuvaga.

Kui kasutada eelnevat A2M Konsooli ja Miinikonsooli ühendamise lahendust, mis toodud välja punktis 2.1.2, siis Miinikonsool hakkaks tegelema puhtalt miinipilduja tõlgendatud käskude sisestamise ja edastamisega ServeAlfonsisse. Stsenaariumimootor ise arvutaks sellisel juhul välja miinide lennu aja ning tabamuskohad, mille tulemusel toimuvad miinisimulaatori arvutused samade tingimuste all nagu ülejäänud ServeAlfonsi poolt juhitud tegevus.

**Peamised eelised:** Peamine eelis on süsteemi stabiilsuse suurenemine ning erinevate stsenaariumisündmuste ühendamine ühe stsenaariumimootori alla.

**Võimalikud tagasilöögid:** Vajadus ümberprogrammeerida suur osa Miinikonsoolist, ühendada selle üks osa A2M'i konsooliga ning miinilennu algoritmid ServeAlfonsiga. Peamine tagasilöök on ettearvamatu arendustsükli pikkus.

## 2.2 HEARTBEAT<sup>8</sup>

Uurides süsteemi koodi selgus, et ServeAlfons töötab 1/30 sekundi aja rütmis, eesmärgiga arvutada stsenaariumis toimuvat kolmkümmend korda sekundis. Süsteem on otseselt sellest kiirusest ka sõltuv, ehk siis kui süsteemi ressursidega probleemid tekivad ja ServeAlfons aeglasemalt töötab, liigub aeglasemalt ka samal ajal töötav stsenaarium. ServeAlfons võimaldab ka stsenaariumitegevust edasi kerida kiiremini, kui elimineerida 30-kaadrit-sekundis ajakontroll. Selle tulemusel arvutatakse stsenaariumis toimuvat vastavalt protsessorite võimekusele.

Võttes ajalise kontrolli maha ja testides erinevaid stsenaariumeid selgus, et keskmises stsenaariumis saab piiranguteta stsenaariumit tööle panna maksimaalselt ainult kaks korda kiiremini. Suuremate stsenaariumite puhul, kus simuleeritakse jalaväerühma ja soomukite liikumist, ei kiirenda piirangu mahavõtmise tihti midagi, sest süsteemi ressursid on juba täies mahus hõivatud.

Uurides mida ServeAlfons täpsemalt teeb kolmkümmend korda sekundis, avastas ka selge põhjuse miks heartbeat süsteemi ressursi nii palju enda alla haarab:

- Iga sõduri asukoha liikumise ja uue positsiooni arvutus
- Iga sõduri asukoha parandus vastavalt simuleeritud üksuse formatsioonile
- Iga sõduri kauguse-arvutus teistest sõduritest ja soomukitest ja sellest tulenevad asukoha ja suuna parandused
- Iga sõduri hetke animatsiooni-kaader

Sedasorti ülesehitus tähendab, et uute vastase üksuste lisamisel stsenaariumitega langeb süsteemi kiirus märgatavalt. Kaitseväge nõutud tingimus süsteemi võimekuse kohta – 30 jalaväeüksust ja neli soomukit korruga – on süsteemi poolt küll täidetud, aga sellest juba vähesel määral üle minnes, pole ServeAlfons võimeline stabiilselt üleval hoidma stsenaariumis toimuva tegevuse arvutust 30 korda sekundis.

---

<sup>8</sup> Heartbeat on töös käsitletud kui kindla määratud aja tagant tarkvaras toimuvad käsud ja kontrollid millega muudetakse tarkvara elementide seisu, parameetreid või väärtusi

Sedasorti olukord piirab võimalikku süsteemi laienemist või kasutamist nõrgema riistvaraga arvutitel või suuremate ja detailsemate stsenaariumite arendamist tulevikus.

### **2.2.1 Edasiarendus: HEARTBEAT'I OPTIMISEERIMINE**

ServeAlfonsi tegevust on võimalik kiirendada mitmete tegevuste ärajaotamisega väiksemale kui 30-kaadrit-sekundis rütmile. Sõdurite suunaparanduste arvutused vähendada 30'lt arvutuselt sekundis 15'ne peale, ei mõjutaks stsenaariumis toimuvat visuaalset pilti, samas kergendaks ServeAlfonsi tööd.

Kõige suurem süsteemi töö vähendamine tuleks siiski proximity-alert<sup>9</sup> arvutuste vähendamisest. Uurides ServeAlfonsi programmikoodi selgus, et stsenaariumitegevuses arvutab süsteem iga sõduri kaugust ja suunda kõikidest teistest sõduritest sama heartbeat alusel. See tähendab, et iga jalaväe jao lisamisel stseeni väheneb süsteemi kiirus märgatavalt. Piirates kauguse arvutused ainult sõduri enda jao siseseks, väheneks ServeAlfonsi koormus märgatavalt, sealjuures muutmata stsenaariumite taktikalist pilti.

Sarnase lähenemisega saaks optimiseerida ka edasikerimise-funktsiooni, kus stsenaariumis toimuvat kiirendatakse kuni sobiva ajahetkeni. Kuna stsenaariumitegevus ilma relvade laskmisteta sõltub puhtalt üksuste liikumisel mööda määratud teed, siis edasikiirendades poleks tähtis detailne visuaalne liikumine graafikamootorites ekraanidel. Seetõttu ei pea ServeAlfons aktiivselt välja arvutama sõdurite uusi positsioone ja animatsioonikaadreid graafikamootorile, kui nendel pole antud kontekstis tähtsust. Vähendades virtuaalsete üksuste arvutustihedust kolmekümnelt kaadrit sekundis viiele, võimaldaks ServeAlfons stsenaariumitegevuste edasikerimist palju efektiivsemalt.

**Peamised eelised:** Peamine eelis on suures osas ServeAlfonsi töökoormuse vähendamine ning selle kaudu avanevad võimalused arendada probleemideta töötavaid suuremaid kaitselahingu-stsenaariume.

**Võimalikud tagasilöögid:** Tegelik protsessori vähendatud töökoormus võib olla väiksem ning nõuab lisaks peale edasiarendust pikemat testimisperioodi, mille käigus kontrollitakse kas kõik eelnevalt heartbeat'ist sõltunud süsteemid ka uue süsteemi all probleemideta töötavad.

---

<sup>9</sup> Proximity Alert'i käsitletakse töös kui kahe ruumis erineval kohal asuva elemendi kaugust teineteisest, mille piirile lähenedes muudab ruumis olevad elemendid oma suunda, üritades vältida teineteist.

## 2.3 KORDUSESITUSED

Alfons 2M arenduse lõpufaasis sai lisatud esialgne kordusesituse funktsioon. Tegu oli simulaatoris tellija poolt nõutud osaga, aga stsenaariumisüsteemi viimaste hetkede paranduste tõttu jäi kordusesituste võimaluse arenduseks vähe aega.

Uurides kordusesituste salvestamise lahendust selgus, et arenduse käigus lahendati ServeAlfonsi stsenaariumi kordusesituse funktsionaalsus kõige lihtsamat võimalikku teed pidi: salvestades kergelt filtreeritud võrguliiklust - täpsemalt käske - mida ServeAlfons saatis graafikamasinatele.

Sellel süsteemil on mitmeid tagasilööke. Visuaalne ServeAlfonsi editor ei võimalda sellise kordusesituse puhul näidata toimuvast ülevaatlikku pilti, kordusesituste failid on väga suured (tihti enam kui 100 MB) ning kordusesituste juures ei salvestata interaktiivselt muutuvat informatsiooni, nagu statistika.

### 2.3.1 Edasiarendus: MUSTA KASTI PÕHINE KORDUSESITUS

Arendajalt saadud süsteemi tööpõhimõtte järgi selgus, et stsenaariumid on väga tugevalt kinni ettemääratud arvutustes. See tähendab piltikult, et sama stsenaariumi sama kaader võrdub alati täpselt sama kaadriga duplikaat-stsenaariumis. Sellest järeldub, et stsenaariumis toimuvat saavad peamiselt muuta ainult Musta Kasti kaudu antud käsud, ehk siis treenitava üksuse tegevus.

Järelikult oleks vähem ressursinõudlik ja võimalusterohkem kordusesituste lahendus samuti Mustast Kastist saadud informatsiooni põhine. Salvestades kasutajate tegevust, mida Must Kast esitab ServeAlfonsile, võimaldaks reaajas näidata ServeAlfonsi kasutajaliideses stsenaariumitegevusest ülevaatlikku pilti, näidata samal ajal muutuvat statistikat ning hoida kordusesituste failid võimalikult väikesed - isegi saadetavad mailide teel raportitena.

Musta Kasti tegevuse salvestamiseks kordusesituste tarbeks nõuab süsteem samas väiksemaid muudatusi ja lisasid. Kordusesituse salvestamisel peab salvestama ka toimuva stsenaariumi enda faili, et kordusesitus vastaks kindlasti eelnevale stsenaariumi koodile ning lukustama ServeAlfonsi editori ja Miinikonsooli käsud, et esitletavat kordusesitust ei häiritaks.

**Peamised eelised:** Väikesed ja kiiresti süsteemis laetavad kordusesituste failid, mida kerge vajaduse korral kasutajalt kasutajale või tellijalt arendajale saata. Teiste edariarendustega koostöös on võimalik jõuda korraliku kordusesituse süsteemini, mis võimaldab täielikku edasi ja tagasi kerimise funktsionaalsust.

**Võimalikud tagasilöögid:** Arendustükkel võib olla ettearvamatult pikkusega ja nõua pikemat testimist, tõenäoliselt peaks osa süsteemist siiski olema salvestatud filtreeritud netiliikluse kaudu, näiteks miinipilduja konsooli kalkulatsioonide tulemusel saadetud miiniplahvatuste asukoht on varieeruv, olukord mis tekib random arvu genereerimise tõttu millega simuleeritakse ettearvamatuid loodustingimusi ja muid varieeruvaid parameetreid.

### 2.3.2 Edasiarendus: KERIMISVÕIMALUSED

Stsenaariumite kordusesitusi pole võimalik simulaatoris tehniliste probleemideta edasi-tagasi kerida. Nii graafikamootori võrguliikluse salvestamise lahendusel, kui ka punktis 2.4.1 toodud Musta Kasti tegevuse salvestamisel eraldi lahendada olukord, kus teatud hetkelised sündmused - nagu miiniplahvatused ja kattesuitsud - kestaksid stsenaariumis ka edasi ja tagasi kerides visuaalsete ja heliga seotud probleemideta.

Kordusesitusi testides selfus, et kuna plahvatused ja suitsuefektid esitatakse Ogre 3D graafikamootoris andes käsu laadida ettemääratud koordinaadile particle-süsteem<sup>10</sup> või mängida positsioonil helifail, siis on kordusesitustes salvestatud ainult see hetk, millal particle-süsteem luuakse või helifail ette mängitakse. See tähendab, et alustades stsenaariumi kordusmängimist üks kaader peale suitsumiini plahvatust, tekitab olukorra, kus suitsumiinile kuuluvat particle-süsteemi ei näe.

Lahendus, mis kordusesitustega koos probleemideta töötada saaks, oleks peamise kordusesituse kõrvele salvestatav kordusesitus, mis salvestab ainult particle süsteemi ning helifailide ajalise informatsiooni. Kuna mõlemat on võrreldes üldise kordusesituses salvestatud informatsiooniga vähe, siis võib mõlemad eraldada üheks iseseisvaks kaadrinumbrist sõltuvaks ja selleläbi sorteeritud vooks, mida saab kiiresti edasi kerida ilma pika ajalise pausita. Tagasi kerides toimuks tegelikult salvestatud voo kiire edasikerimine alguspunktist.

See lahendus oleks sarnane kiirendatud edasikerimisfunktsiooniga, mida käsitletud punktis 2.2.1.

**Peamised eelised:** Korralik võimalus kerida salvestatud stsenaariumitegevust ka tagasi. Tegu on kiire lahendusega olgugi, et ajahetke muutmisel genereeritakse iga muutmise käigus kiirkorras ära valitud ajahetkeni kõik particle süsteemid ja helifailid.

---

<sup>10</sup> Particle Süsteem on 2D ja 3D primitiivseid osakesi reeglite põhjal genereeriv süsteem, mida kasutatakse suitsu, leekide ja erinevate uduefektide tekitamisel virtuaalses 3D keskkonnas.

**Võimalikud tagasilöögid:** Väga suurte miinipilduja pommituste puhul võib particle süsteemide ja heli edasikerimisel tekkida tarkvaralisi probleeme, samas tarkvaraliselt vähendades otsingu hetkel toimuvate particle süsteemide poolt genereeritud elementide hulka ning otsingu hetkel limiteerida erinevate heliefektide arv peaks probleemi elimineerima.

## 2.4 ANIMATSIOONIMOOTOR

Alfons 2M arenduse käigus üks enim probleeme tekitanud koht oli ServeAlfonsi ja graafikamootorite hoidmine omavahel sünkroonis, et sõduri animatsioon ühelt ekraani servalt teisele üle minnes oleks alati sama. Sarnaselt sõduri asukohtadega, saadab ServeAlfons graafikamasinasse 30 korda sekundis ka sõduri animatsioonikaadri. Sõdurite animatsioonifailid on valmistatud seda reeglit silmas pidades, omades 30 kaadrit sekundis biped-animatsiooni<sup>11</sup>.

Olemasolevat animatsioonilahendust uurides selgus, et ServeAlfons kontrollib animatsioone tsüklite kaupa, animatsioon ei vahetu ennem eelmise animatsiooni lõppu, ehk siis ServeAlfons hoiab iga sõduri kohta meeles tema hetkel kestvat animatsiooni, selle kaadri numbrit ning sellele järgnevat animatsiooni. Näiteks saades käsu joosta, asetatakse see järgmiseks animatsiooniks. Reaalselt käivitatakse uus animatsioon alles siis, kui eelmise animatsiooni tsükkel on lõppenud, mis tagab sujuva ülemineku ühest animatsioonist teise.

Animatsiooni üleminekuid on Alfons 2M'is kahte liiki, ühed on üleminekud mis on manuaalselt animaatori poolt valmistatud, teine üleminek on kahe erineva animatsiooni vaheline sujuv üleminek, kus interpoleeritakse kahe animeeritud osa asukohad vastavalt üleminekule ja saadakse uus kokkusulatatud animatsioon.

See lahendus samas tähendab, et võrreldes üldise 30-muutust-sekundis lähenemisega ServeAlfonsis on animatsioon väga aeglase reaktsiooniga, tekitades hetki, kus sõdur võib küll kuuliga pihta saada, aga langeb maha alles märgatav aeg hiljem.

### 2.4.1 Edasiarendus: RAG DOLL FÜÜSIKA

Olemasoleva animatsioonisüsteemi ümbertegemine nõuaks pikka arendusperioodi, aga enim tellijale probleeme tekitanud olukord - kus pihtasaanud sõdur langeb alles animatsioonitsükli lõpus - oleks lahendatav kui pihtasaamise hetkel anda sõduri animatsiooni-skeleti üle kontroll Open

---

<sup>11</sup> Biped (ning Skeletal) animatsioon on karakteri animatsioon, milles on animeeritud primitiivset kere, kere osad mõjutavad seda ümbritseva detailsema mudeli tippe ning tõlgendavad kere animatsiooni detailsele mudelile.

Dynamics Engine rag-doll füüsikale, mis on toetatud Ogre 3D graafikamootori poolt. Rag-doll füüsika tähendab olukorda, kus virtuaalse sõduri animatsiooni-skeleti - mis muidu kontrollitud eelnevalt määratud animatsioonipooside poolt - võtab üle füüsikamootor ning vastavalt gravitatsioonikonstandile ning liikumissuunale määratud muutujale paiskab sõduri maapinnale.

Rag-doll füüsika samas ei tohiks mõjutada miinide plahvatusi või suuremat massilist protsessi, mis võib haarata gruppi virtuaalseid üksuseid korraga. Mitme sõduri rag-doll füüsika sisselülitumine sama lühikese ajahetke jooksul võib tekitada graafikamootorile raskuseid ja aeglustada süsteemi töökiirust.

***Peamised eelised:*** Lasu hetkel reaalsemad vastase sõdurite pihtasaamised.

***Võimalikud tagasilöögid:*** Erinevad tuntud rag-doll füüsika probleemid, mis võivad tekitada stsenaariumi käigus probleeme. Collision Detection<sup>12</sup> maapinnaga ning erinevad inimluustikule võimatud poosid on peamised mured.

---

12 Collision Detection on kahe objekti vaheline pörketuvastuse kontroll.

### 3. SERVEALFONS'I FRONT-END<sup>13</sup>

ServeAlfonsi stsenaariumimootor esitleb stsenaariumis toimuvat operaatorile eraldi läbi OpenGL'i<sup>14</sup> renderdatud illustratiivse maailmapildi ja omab samas ka erifunktsionaalsust, milles luuakse uusi stsenaariumeid, muudetakse olemasolevaid stsenaariumeid või muudetakse-kohandatakse stsenaariumit, mis hetkel käimas.

ServeAlfonsi operaatoripoolne kasutajaliides annab operaatorile võimaluse näha tehnilist ja samas illustratiivset ülevaadet stsenaariumis toimuvast. Kasutajale kuvatakse detailne informatsiooni virtuaalsete üksuste liikumiste ja staatuse kohta. Liides omab ka võimalust operaatoril ServeAlfonsi akna sees hiire-koordinaate kasutades otseselt testida stsenaariumis toimuvat: operaator võib hiirega määratud asukohal tekitada relvade laske, mida ServeAlfons tõlgendab sarnaselt Musta Kasti käskudele ja saadab sellekohase informatsiooni edasi graafikamootoritele.

ServeAlfonsi teine kasutajaliides on tavakasutaja poolt varjatud stsenaariumi editori keskkond, mis sisaldab lisaks operaatori funktsionaalsusele veel võimaluse muuta teekonnapunktide asukohti, lisada virtuaalseid üksuseid ja neid siduda teekonnapunktidega ning neile lisada tegevuse skripte ja kõiki tehtud muudatusi salvestada. ServeAlfonsi stsenaariumi loomise tarkvara on kirjutamise hetkel arendusprogramm ning nõuab eelnevalt detailset dokumentatsiooniga tutvumist ning Python programmeerimise kogemust stsenaariumite loomisel.

Kontrollides kasutajaliidese komponendi osakaalu stsenaariumi jooksul selgus, et ServeAlfonsi front-end pole süsteemile elutähtis osa ja simulaator töötab ilma ServeAlfonsi visuaalse OpenGL pildita ilma probleemideta. OpenGL visuaalne kasutajaliides on samas vajalik stsenaariumite loomisel ja muutmisel ja selle tarbeks vajab arvuti ning operatsioonisüsteem OpenGL tuge.

---

<sup>13</sup> Front-end on tarkvara pealmine osa, mis sisaldab endas kasutajaliidest, sisestamisvorme, menüüsid ja visuaalset keskkonda, millega antakse kasutaja poolt käsked süsteemi back-end'ile.

<sup>14</sup> OpenGL on tarkvara programmeerimise liides 2D ja 3D graafika esitamiseks tarkvaras.

### **3.1 KASUTAJALIIDES**

Operaatori poolt kasutatavas ServeAlfonsi kasutajaliideses puuduvad menüüd ja selgesti eristatud funktsionaalsus. See vähene, mis operaatoril võimalik, on saavutatav ainult eraldi ära märgitud klaviatuuri nuppudel ja nõuab operaatorilt keskkonnaga eelnevalt kohanemist. Operaatori ServeAlfonsi otsene tundmine samas pole Alfons 2M lahingusimulaatori kasutamiseks vajalik ja on mõeldud operaatoritele, kes tehnilisemat pilti stsenaariumi toimumise ajal näha soovivad või selle kaudu arendajatele täpsemat tagasisidet anda tahavad.

ServeAlfonsi kasutajaliides on veel keerulisem, kui ServeAlfonsit kasutada stsenaariumite muutmiseks ja loomiseks, mis annavad kasutajale kümneid erinevaid klaviatuuri kaudu sisestatavaid käsked, mis lugemata dokumentatsiooni tekitas programmi kasutamisel raskuseid ka sellega seotud arendajale.

Üldine informatsioon - teed, teekonnapunktid, üksused ja soomukid - on kuvatud ServeAlfonsi kasutajale probleemideta, samal ajal kui spetsiifiline informatsioon on kasutajale esitatud raskestiloetavates tekstiblokkides, mille täpsemad tähendused on jällegi arusaadavad vaid dokumentatsiooniga tutvudes. Seetõttu eeldab stsenaariumite muutmist ja uute stsenaariumite loomist pidevalt konsulteerimist dokumentatsiooniga.

Lisaks puudub otsene seos graafikamootoris oleva 3D maailma ning ServeAlfonsis esitatud tasandilise kaardi vahel. Nende kahe vaheline ainus tõlgendatav seos on see, et mõlemad kasutavad 2x2 km suuruses ala ja kasutavad sama koordinaatsüsteemi. See tähendab, et 3D maailma elemente - nagu hooneid, puid ja maju - ei kuvata ServeAlfonsis. ServeAlfonsi tasandiline kaart on kaetud puhtalt sama ala aerofotoga, mis uute elementide lisandumisel 3D ruumis nõuab stsenaariumi arendajalt rohkem testimist ja seetõttu aega.

#### **3.1.1. Edasiarendus: HIIREPÕHINE NAVIGATSIOON JA MENÜÜD**

ServeAlfonsi kasutajaliidesepõhine programm valmistati esialgselt puhtalt arendusprogrammiks, millega ei tellija ega operaator kunagi tegelema hakka, aga arenduse lõpufaasis avaldas operaatorile ServeAlfonsi ülevaatlik stsenaariumipilt piisavalt huvi, et sellest arendati lühikese aja jooksul välja väiksemate funktsioonidega operaatori tugiprogramm. Lähenedes ServeAlfonsi kasutajaliidese kasutamisele operaatori perspektiivist lähtudes leidsin, et kasutajaliidese kasutamine on mitmeti ebamugav ja kohmakas.

Programmi kasutamine sellises võrgustikus on raskendatud ja nõuab pikaajalist harjumist. Eelnevatele kogemustele põhinedes leidsin, et suuresti parandaks olukorda hiirepõhine funktsionaalsus, kus erinevad operaatorikäsud - miiniplahvatus, üksuste ümbersuunamine või elimineerimine ja kaardil liikumine - oleks teostatavad puhtalt hiirega, mitte klaviatuuri nuppe kasutades, kuna ülejäänud süsteem vajab klaviatuuri kasutamist minimaalselt. Icoonide ning mugava hiirefunktsionaalsusega kasutajaliidesed on samal ajal kasutajale kergemini ja kiiremini omandatavad.

Hiirepõhine menüü aitaks samamoodi kaasa ka uute stsenaariumite loomisel ning vähendaks vajadust pidevalt kasutada kasutajaliidese poolset dokumentatsiooni. Stsenaariumite arendamise juures oleks mugavam navigatsioonisüsteem samuti kasulik ka siis kui stsenaariumi arendus nõuab suurel määral programmikoodi kirjutamist.

***Peamised eelised:*** Mugav, kiiresti omandatav ja kergesti kasutatav.

***Võimalikud tagasilöögid:*** Arendusprotsess on pikk ja nõuab eraldi kasutajaliidese jaoks arendatud süsteemi disaini, et omada tõelist eelist olemasoleva klaviatuuripõhise süsteemi ees.

### **3.1.2. Edasiarendus: VISUAALSED TÄIENDUSED**

Nii operaatoril, kui ka stsenaariumi arendajal, on keeruline välja lugeda spetsiifilist informatsiooni virtuaalsete üksuste kohta. Operaatori jaoks oleks kasulikud lihtsustatud kastid, milles näidatud üksuse number ja meeste hetke tegevus ja meeste arv.

Stsenaariumi arendajale oleks lisaks eelnevale veel kasulikud visuaalsed abivahendid, mis näitaksid ära valitud üksusega seotud teekonnapunktid ja Python skriptid. Sedasorti visuaalsed toetuselemendid kiirendaksid stsenaariumite loomise ja muutmise protsessi märgatavalt, kuna lahingusimulaatoris on stsenaariumeid, kus kuue üksusega on seotud parkümmend erinevat teed ja sadu teekonnapunkte ning sadu väikseid Python skripte.

ServeAlfons talletab virtuaalsete üksuste ID'sid väga kõrgete numbritena, sadades tuhandetes, mida on tihti raske meelde jätta, kui stsenaariumi arendaja peab meeles hoidma mitme üksuse ID'd. Lahendus oleks ID'd katta visuaalsete ajutiste ID'dega nii, et ka Pythoni skript seda tõlgendaks läbi väikse funktsiooni. Süsteemisiseselt võib olla üksuse ID igasugune number, aga stsenaariumit muutes, luues või stsenaariumitegevust vaadates esitataks üksuseid ajutiste ID'dega või nimedega (näiteks tähed ABC). Lahendus parandaks olukorda ja ServeAlfonsi kasutamiskiirust märgatavalt.

**Peamised eelised:** Visuaalselt kiiremini omandatav piltlik ülevaade süsteemis toimuvast. Vähem tehnilisi tekstikaste ja rohkem ikoone teevad ServeAlfonsi kasutamise ka operaatorile atraktiivsemaks ära võtmata midagi ServeAlfonsi funktsionaalsuselt.

**Võimalikud tagasilöögid:** Sarnaselt edasiarenduse 3.2.1'ga nõuab eraldi kasutajaliidese jaoks loodud süsteemi disaini ja mitmeid graafikaelemente.

### **3.1.3 Edasiarendus: SERVEALFONS'I SEOS GRAAFIKAMOOTORIGA**

Vähendamaks vajadust stsenaariumite visuaalse testimise järgi, aitaks stsenaariumite muutmisel ja loomisel palju kaasa olukord, kus ServeAlfonsi illustratiivsel kaardil oleks esitatud 3D kaardil olevate hoonete, puude ja muude objektide asukohad. Järeldusele jõudsin põhinedes mitme kuu pikkusel kogemusel ServeAlfonsit kasutades stsenaariumite loomiseks sõltusin tugevalt graafikamootoris esitatud pildi olemasolust, mis omakorda sundis tähelepanu jagama kahe erineva programmi vahel.

Sedasorti lisa oleks eriti vajalik olukorras, kus stsenaariumisse on lisatud elemente, mida reaalselt Meeri alal ei eksisteeri ja mis muidu nõuaksid mitmekordset testimist graafikamootori kaudu stsenaariumi arendamisprotsessis. Esialgsete stsenaariumite juures sellelaadset olukorda ei esinenud, aga võimalus dünaamiliselt maailmasse objekte lisada oli tellija üks nõudeid.

Graafikamootori süsteemi uurimisel selgus, et virtuaalse maailma 3D objektide asukohad on määratud XML<sup>15</sup> põhistes nimekirjades. Kuna graafikamootori ja ServeAlfonsi kaardid töötavad sama koordinaatsüsteemi põhjal siis oleks võimalik kasutada XML'is salvestatud andmeid filtreeritult objektide positsioonide esitamiseks ServeAlfonsi kasutajaliideses. Lahendus tõstaks stsenaariumite muutmise ja uute loomise töökiirust ning võimaldaks ka esitatud informatsiooni kategoriseerida, näiteks seadistada ServeAlfons näitama ainult teatud tüüpi objektide, nagu puud või hooned, asukohti.

Lahendus oleks peale stsenaariumi arendaja kasulik ka süsteemi operaatorile, kellel seetõttu lihtsam hinnata üksuste asukohti maastikul. Eriti kui tegu on uue stsenaariumiga, mida operaator veel treeningüksusega läbi harjutanud pole ja puudub võimalus eelnevatele kogemustele tuginedes hinnata toimuvat, teadmata kus ja millal ekraanil üks või teine üksus asuda võib.

---

15 XML (Extensible Markup Language) on üldotstarbeline markeerimise keel, mille eesmärgiks on struktureeritud info jagamine erinevate infosüsteemide vahel

**Peamised eelised:** Tõstab ServeAlfonsi kasutamist veel iseseisvamaks, võimaldades uute stsenaariumite esialgset arendust, vajaduseta pidevalt tehtud muudatusi testida ja üle vaadata graafikamootoris – protsess, mille saab sellisel juhul jätta eraldi kõige lõppu.

**Võimalikud tagasilöögid:** Süsteemi programmeerija Elmo Trolla sõnul on OpenGL läbi Pythoni tekitanud probleeme ja süsteemi aeglustumist. Laadides sisse suure maailma XML faili ja sealt filtreeritud markerid virtuaalsele kaardile, võib protsess aeglustada süsteemi tööd nii andmete sisse laadides, kui ka ServeAlfonsit kasutades. Samas läbi korraliku sisselaadimise filtri ei tohiks süsteemi kiirus olla takistuseks.

## **4. SERVEALFONS JA STSENAARIUMID**

### **4.1 STSENAARIUMITE ARENDAMISE LIIDES**

ServeAlfonsis stsenaariumite muutmine või uute stsenaariumite loomine sõltub otseselt ServeAlfonsi stsenaariumite konfiguratsioonifailist. ServeAlfons loeb sisse parameetrid stsenaariumi kohta ja kuvab kasutajale teekonnapunktid ning virtuaalsed üksused kaardile. Kasutaja võib lisada uusi üksuseid või pikendada/lühendada stsenaariumi kaardil olevaid teid vastavalt vajadusele.

Stsenaariumite muutmine ServeAlfonsis ei erine stsenaariumite käivitamise ja testimisega: mõlemad toimuvad samas arenduskeskkonnas. See võimaldab stsenaariumitegevust jälgida ja testida arendamisega samal ajal.

#### **4.1.1 Edasiarendus: SÕLTUMATUS NOTEPAD TEKSTIREDAKTORIST**

ServeAlfonsi võimalus muuta ja salvestada stsenaariumeid ja tehtud muudatusi näha, ilma vajaduseta stsenaariumeid uuesti laadida, vajab kasutamiseks Notepad tekstiredaktorit ja seetõttu Windows operatsioonisüsteemi. Näiteks testides tarkvara Mac OS'il võimaldas tarkvaral probleemideta töötada hetkeni, mil kasutaja üritab muuta või lisada Pythin skripte teekonnapunktidele.

Tegu on arenduse käigus tehtud kiire lahendusega. Selle asemel, et lisada tekstiredaktorite tugi erinevatele operatsioonisüsteemidele, on parem lahendus laiendada ServeAlfons eraldiseisvaks arendusplatvormiks, mis kasutaks omaenda sisemist tekstiredaktorit ja oleks kohandatud Alfons 2M stsenaariumite arendamiseks.

Sisemisel tekstiredaktoril oleks potentsiaali veel mitmete lisavõimaluste jaoks. Uurides arendusplatvorme nagu Microsoft Visual Studio, leidis mitmeid võimalusi skripti kirjutamist mugavamaks ja kiiremaks teha. Üks kasulikumaid lisisid edasiarendusel oleks võimalus muuta Python skriptides üksuste klasse, teekonnapunktide identifikaatoreid ning tegevuskäske kiiresti, ilma vajaduseta neid ennem skripti kirjutamist meelde jätta, valides neid näiteks valikmenüüst.

Teine paremat arenduskeskkonda soodustav lisavõimalus oleks skriptide andmebaas, kuhu võimalik salvestada erinevaid skripte, mida kiiresti ja kergelt stsenaariumi teekonnapunktidest lisada ja seal muuta vastavalt parameetriteid. Lahendus tõstaks stsenaariumite loomise kiirust ja efektiivsust ning vähendaks süntaksi vigade arvu märgatavalt.

**Peamised eelised:** Võimalus arendada ServeAlfonsiga stsenaariumeid erinevates Python-toega operatsioonisüsteemides, mitte ainult Windows keskkonnas. Stsenaariumi arenduse põhise funktsionaalsusega tekstiredaktor kiirendab ja lihtsustab märgatavalt arendajale stsenaariumite ehitamist.

**Võimalikud tagasilöögid:** Arendusplatvormi põhise tekstiredaktori arendus on juba ise-enesest keeruline, sellelaadse lahenduse implementeerimine ServeAlfonsi keskkonda tähendab pikka ja keerulist arendusprotsessi ja oleks nii firmale, kui ka tellijale teisejärgulise tähtsusega.

## **4.2 STSENAARIUMITE ARENDUSE TURVALISUS**

Alfons 2M ServeAlfonsi poolt jooksuputatud stsenaariumid on süsteemilähedased skriptid, ehk siis stsenaariumitesse teekonnapunktidest kirjutatud koodiread käituvad, kui osa ServeAlfonsist ja on seetõttu väga süsteemilähedased.

See tähendab, et erinevad vead, mis võivad esineda stsenaariumi skriptides - kas siis loogikavead teekonnapunkti salvestatud skriptis või stsenaariumi kaardilt kustutatud üksus, kelle ID'ga seotud skriptid ühtäkki programmi tööd takistama hakkavad - tekitavad stsenaariumit arendades mitmeid probleeme. Kuna iga skript on ServeAlfonsi enda programmikoodi lähedane, siis pole seda võimalik täiesti eraldiseisvalt kontrollida ja seetõttu on sellega seotud vigade otsimine süsteemis samuti raskendatud.

Väga süsteemilähedane stsenaariumiarendus takistab ka ServeAlfons stsenaariumimootori laienemist ja kasutamist tellijapoolsete arendajate või isegi operaatori poolt.

### **4.2.1 Edasiarendus: FUNKTSIOONIPÕHISED SKRIPTID**

Alfons 2M stsenaariumite potentsiaal tulevikus sõltub suuresti sellest, kui kergelt ja kiiresti on võimalik luua uusi stsenaariumeid arendusmeeskonna poolt. Mida rohkem tegeleb arendaja vigade otsimise ja süntaksi kontrollimisega skriptides, seda vähem tegeleb arendaja stsenaariumi endaga.

Kuna suur osa olemasolevast stsenaariumisüsteemist nõuab kasutajalt eraldi kursuse läbimist ja omajagu teadmisi programmeerimisest, siis piirab praegune lahendus võimalust välisel arendajal või tellijal uute stsenaariumite juurde arendamist ilma selleks esialgset arendajat kaasamata.

Uurides erinevaid kasutajasõbralikke lahendusi, mis kaudselt lahingusimulaatorite keskkondadega seotud leidsin, et enim kasutatud lahendus on programmikoodi peale ehitatud trigger-event süsteem, mida kasutatakse avalikes lahingusimulaatorites nagu Operation Flashpoint ja Armed Assault. Trigger-event põhine süsteem tähendab süsteemi kus käsud ja tegevused on lihtsasti loetavad funktsioonid, mille all on detailsem programmikood. Trigger-event põhiste süsteemidega saab kergesti ja kiiresti luua detailseid stsenaariumeid, määrates selle elemente teekonnapunktidele ja täites ainult käskude ja tegevuste juurde käivad lüngad, nagu gruppide ID'd.

Trigger-event funktsioonipõhine süsteem on lahendatav ServeAlfonsis läbi kahe kihi: funktsioonid ja funktsioonimatriks.

- Funktsioonimatriksis on ühendatud teekonnapunktide ID'd, üksuste ID'd ja funktsioonid.
- Funktsioonid on baaskäsud ehk peidetud skriptid, mis täidavad kindlaks määratud ülesannet ja võimaldavad samal ajal kergesti muutujaid kontrollida.

Selline lähenemine tähendab seda, et stsenaariumis saab üksuse või teekonnapunkti kaotamisel läbi kiire funktsioonimatriksi tuvastada või elimineerida stsenaariumielemente, mis ühtäkki eraldati ülejäänud stsenaariumikoodist ja vastasel juhul muutuksid kasutuks või saavad olema potentsiaalne vea koht kus stsenaarium võib blokeeruda. Selle kaudu tuvastatud veast võib kasutajat teavitada ilma programmi sisese vea tekkimata või ka lahendada automaatselt: elimineerida või välja kommenteerida skripti osad, mis veaohklikud.

Funktsioonid ise oleks eraldi failist sisseloetud baaskäsud, mida süsteem käsitleb standardse programmi koodi funktsioonina, tõlgendades trigger-event käsus sisestatud parameetrid funktsiooniparameetriteks.

See lahendus võtaks küll ära arendajalt võimaluse süsteemi enda sees Python skripti kirjutada, aga alternatiivselt oleks arendajal võimalik endale sobivaid funktsioone lisada programmeerida koodina faili kust neid hiljem ServeAlfonsi arenduskeskkond neid sisse loeb.

Funktsioonipõhine stsenaariumite ülesehitus aitaks kaasa ka stsenaariumite arendamise laienemisele vähem programmeerimisest teadlikele kasutajatele, jällegi laiendades süsteemi potentsiaali. Aja jooksul muutuks programm tellijale kergesti kasutatavaks stsenaariumite arenduse platvormiks,

võimaldades Aldons 2M simulaatori arendajal suunata ressursse muudele simulaatori edasiarenduse osadele, vajamata edasiarenduste juures pidevat tarkvarameeskonna tööd.

**Peamised eelised:** Selgem ülevaade stsenaariumitest, uutel arendajatel võimalus süsteemiga kiiresti kohaneda. Vähem süntaksi ja muutujanimedest tekkivaid vigu. Võimalus laiendada stsenaariumite arendamine tellijapoolsetele operaatoritele või arendajatele.

**Võimalikud tagasilöögid:** Võib süsteemiga juba kohanenud arendajatele tekitada väiksemaid ebamugavusi. Pikk edasiarenduseks vajalik arendustsükkel ning testimisperiood.

## 4.3 ILMASTIK

Alfons 2M stsenaariumid käsitlevad peale erinevate lahinguolukordade endas ka erinevaid ilmastikutingimusi ning kellaagegu. Süsteemis on toetatud erinevad eraldiseisvad staatilised ilmastikutingimused: valgustus, vihm, udu ja taustkeskkonnad, millega on võimalik kujutada kõiki kevadest-sügiseni ilmastikuolusid ning päikesetõuse, keskpäevi, õhtuid ja õist pilvisust ning kuuvalgust.

Esialgses Alfons 2M simulaatoris on ilmastik staatiline ja otseselt seotud stsenaariumitega. Stsenaariumitel on kindlad kellaajad ja ilmastikutingimused, mis vastavad tellija soovitud nõuetele.

### 4.3.1 Edasiarendus: STSENAARIUMI ILMASTIKUTABEL

Stsenaariumite võimaluste suurendamise visuaalselt märgatavaim meetod on lisada olemasolevatele stsenaariumitele erinevaid ilmastiku ja valgustustingimusi. Käesoleva edasiarenduse soovitus põhineb esialgsel Alfons 2M süsteemi disaini koosolekutelt säilinud materjalidelt kus autor soovitas sellelaadset süsteemi kui ühte stsenaariumivõimalusi laiendavat osa. Detailsem ilmastikumudel jäi esialgse arendustsükli juures kõrvale arendusperioodi ajanappuse tõttu.

Lihtsa kellaaja ja ilmastikutingimuste maatriksi põhiselt oleks võimalik ära määrata millistel kellaegadel ja millistel tingimustel stsenaariumi läbimängimine oleks taktikaliselt õige.

Kellaaja ja ilmastikutingimuste tabelist sõltuvad valikud saaks seadistada A2M Konsoolis, mille tulemusel ServeAlfons annab graafikamootorile käsu laadida vastav valgustus ja keskkond. Ilmastikutabel suurendaks oluliselt stsenaariumite läbimängimisvõimalusi Kaitseväge treeningtsükliks ja nõuaks vähest edasiarendust, olles teostatav lühikese arendustsükli vältel.

**Peamised eelised:** Kiire lahendus olemasolevate stsenaariumite valikuvõimaluste laiendamiseks. Hea ja kiire põhi 4.3.2 edasiarenduse jaoks.

**Võimalikud tagasilöögid:** Puuduvad, ainus erinevus on valikuvõimaluste laienemine.

### **4.3.2 Edasiarendus: ILMAST SÕLTUV STSENAARIUMIMUDEL**

Ilmastikuolude korralikuks realiseerimiseks on vaja rohkem, kui ainult ilmastikutabelit. Reaalses elus ja Kaitseväge koostöörupi kinnitusel muudab tugevalt erinev ilmastik drastiliselt vastase üksuste taktikalist tegevust ja stsenaariumikäiku.

Ilmastikust dünaamiliselt sõltuv stsenaarium koosneb kahest peamisest elemendist:

- Ilmastikukontrollid, mida kasutatakse ilmastiku tuvastamiseks. Sarnaselt stsenaariumis olevatele sündmuste skriptidele oleks tegu maailmapõhise üldsündmusega, mille olemasolu saab stsenaariumis kontrollida.

Ilmastikukontrollide kaudu võib kergesti mõjutada väiksemaid stsenaariumis toimuvaid sündmusi – nagu vastase jooksukiiruse erinevus pimedas võrreldes keskpäevaga. Samuti võib terve stsenaariumi tegevus muutuda märgatavalt erinevate ilmastikuolude tulemusel, näiteks vastase kasutab liikumiseks teisi teid. Lisaks võimaldavad ilmastikukontrollid määrata ära olukorrad stsenaariumites, kus vastase üksus kasutab valgusrakette, mida vastane päevavalguse ajal ei teeks.

- Üldparameetrid on erinevate ilmastikutingimuste kohta käivad kordajad vahemikus 0.0-2.0 (algseade 1.0) millega võimalik manipuleerida vastase tuleintensiivsust, tegutsemiskiirust ja teede skaalat suhtes meie üksuse kaamerapositsiooniga.

Üldparameetritega saab vastase üksused panna pimedates tingimustes automaatselt treenitavast üksusest lähemalt mööda liikuma, halva nähtavuse korral aeglasemalt treenitava üksuse positsioonide suunas tulistama või vihmade ilmaga vähendada veoki liikumiskiirust läbi raske maastiku.

**Peamised eelised:** Lisab uue dimensiooni stsenaariumisüsteemi, mille kaudu võib olemasolevaid stsenaariume vaikselt laiendada ja nendele ilmastikust tingitud lisavõimalusi külge programmeerida.

**Võimalikud tagasilöögid:** Üldparameetrid võivad teatud tingimustel tõsta stsenaariumis toimuva tegevuse välja sobivatest taktikaliselt täpsetest piiridest ja vajab pikemat stsenaariumite testimisperioodi kõikide erinevate valgus- ja ilmastikutingimustes.

## KOKKUVÕTE

Alfons 2M tarkvarasüsteemi ServeAlfonsi süsteemi ja selle edasiarendusvõimaluste analüüsis ilmneb selgelt, et esialgse tarkvaralahenduse mitmed osad on arenduse käigus jäänud rohkem kinni tarkvarameeskonna väikeste ressursside ning ajaprobleemide taha. Töö autori kogemused ja teadmised ServeAlfonsi osadest laienesid analüüsi kirjutamisel nii Python'i koodiridu uurides, kui ka lisainfot küsides arenduse sügavama poolega tegelenud arendajate käest.

Töö eesmärk, analüüsida olemasolevat ja leida mitmeid elemente, mida olemasolevas süsteemis parandada, ümber teha ning tulevikus laiendada, sai autori poolt täidetud. Autor tõi esile esialgse süsteemi analüüsi käigus avastatud süsteemi veakohti ja sellega seotult tõi välja enam kui kümme erinevat ServeAlfonsiga seotud potentsiaalset suunda süsteemi uuendamiseks ning parandamiseks arendajale ELI OÜ ning tellijale, Eesti Kaitseministeeriumile ning Kaitseväge Ühendatud Õppeasutustele.

Töö keskmes käsitletud ServeAlfonsi tarkvaralahendus - ning suur Alfons 2M lahingusimulaator milles käsitletud tarkvara pesitseb - on siiski Eesti mastaabis veel väga uus süsteem oma puudujääkide ja vigadega. Sedasorti analüüsid süsteemi tarkvara või ka riistvaraliste elementide kohta on võimalised esile tooma lahendusi - millises suunas edasi liikuda ja mida ümber teha - et jõuda parema ja edukama tulemuseni. Samas valdkonnas või ka sama lahingusimulaatori baasil on võimalik teostada edasisi analüüse ning uuringuid süsteemi erinevate osade edasiarenduse soodustamiseks. Eesti väikesel IT maastikul on ka kõige väiksem analüüs, kriitika või tõuge õigemal suunas kasulik ning väga suure tähtsusega.

Käesolev töö ja selles sisalduv komponentide analüüs ning edasiarenduse ideed lähevad tervel kujul firma tarkvara-arenduse meeskonnale.

## SUMMARY

Bachelor's thesis „Analysis of Combat Simulators ServeAlfons Scenario-engine“ is an in depth software design analysis, analysing military combat simulator Alfons 2M software component, server and scenario engine ServeAlfons, its functions, relations to other system components and ideas for future development and improvement. The thesis is written as an analysis and improvement research of the ServeAlfons software and is planned to be used for combat simulators future development by the developer company.

The thesis opens with introduction to Combat Simulators Alfons series history, its developer ELI Ltd and the Estonian Defence Forces for whom the simulator was developed. Thesis goes over the main components of the Combat Simulator before explaining more in depth about the basic structure of simulators software system components and their network connections before focusing on its main system: ServeAlfons.

ServeAlfons is separated into three categories by the author, giving the three each their own chapters. First the author discusses various elements of ServeAlfons back-end systems, analysing potential problems with its net traffic and performance, heartbeat system, playback and animation-engine components which the author considers the most performance hindering components of ServeAlfons. Author suggests various ideas how to improve said components.

In the next chapter, author delves into front-end aspect of ServeAlfons, which due to being a development tool, has a clumsy user interface and design. The author suggests a better mouse and icons based navigation system for future development, as well as offers ideas how to improve ServeAlfons visual display and connection with the graphics engine, making the software easier to grasp by either the simulators operator or the scenario software developer.

In the final chapter author focuses on battle scenario and virtual daytime and weather systems that are both handled by ServeAlfons. Author shares his ideas about how to add an extra dimension to the Combat Simulator engine through weather impact, as well as how to simplify scenario development for the simulator.

Each of the thirteen different improvements suggested are based on analytical research of the software and author lists each improvement together with a potential benefits and disadvantages list, based on authors previous experience in software development.

The author completes his set-goal to analyse ServeAlfons and bring out various ideas for its further development in year 2008.

# KASUTATUD KIRJANDUS

## ALLIKAD

[1.] **Jordan N. Fuhr** (2004), „2004 International Top 50“, Military Training Technology  
URL: <http://www.mt2-kmi.com/MT2-TOP-50.pdf> (05.05.2008)

ELI OÜ (ELI Ltd) [www.eli.ee](http://www.eli.ee)

## PROJEKTIGA SEOTUD LISAMATERJALID *(kaasas ka töö CD'l)*

A2Mi reklaamleht (Inglise) *(autor: Kristo Vaher)*

[http://www.eli.ee/private/A2M\\_reklaamleht\\_eng.pdf](http://www.eli.ee/private/A2M_reklaamleht_eng.pdf)

A2Mi reklaamleht (Eesti) *(autor: Kristo Vaher)*

[http://www.eli.ee/private/A2M\\_reklaamleht\\_est.pdf](http://www.eli.ee/private/A2M_reklaamleht_est.pdf)

Alfons 2M'i demonstratsioonivideo *(autor: Kristo Vaher)*

[http://www.eli.ee/private/A2M\\_trailer.avi](http://www.eli.ee/private/A2M_trailer.avi)

**Kontaktandmed**

Kristo Vaher

[kristo@vaher.net](mailto:kristo@vaher.net)

GSM (+372) 56604491