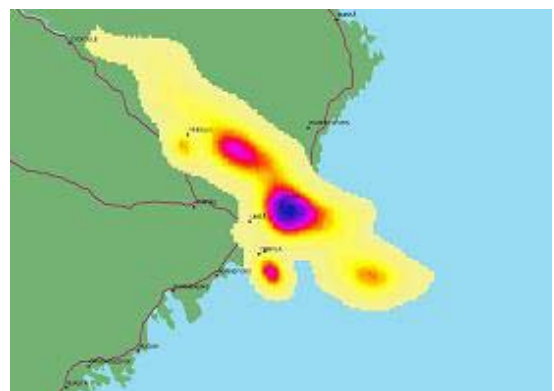
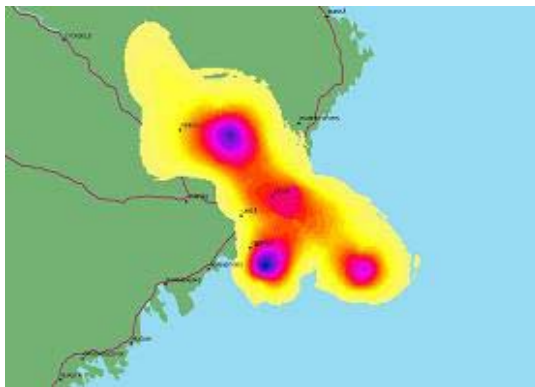


Magnus Danielson, Jan Burman, Jan-Olov Moberg

Utveckling av ett GIS-verktyg för analys av mätprecision samt resursallokering



TOTALFÖRSVARETS FORSKNINGSPINSTITUT

NBC-skydd
901 82 Umeå

FOI-R--0924--SE

Oktober 2003

ISSN 1650-1942

Metodrapport

Magnus Danielson, Jan Burman, Jan-Olov Moberg

Utveckling av ett GIS-verktyg för analys av mätprecision samt resursallokering

Utgivare Totalförsvarets Forskningsinstitut - FOI NBC-skydd 901 82 Umeå	Rapportnummer, ISRN FOI-R—0924--SE	Klassificering Metodrapport
	Forskningsområde 3. Skydd mot massförstörelsevapen	
	Månad, år Oktober 2003	Projektnummer A4521/E4812/I4210/A413
	Verksamhetsgren 1. Forskning för regeringens behov	
	Delområde 33 NBC-studier	
	Författare/redaktör Magnus Danielson Jan Burman Jan-Olov Moberg	
Projektledare Ronny Bergman		Godkänd av
Uppdragsgivare/kundbeteckning		Tekniskt och/eller vetenskapligt ansvarig Kenneth Lidström, Jan-Olov Moberg, Jan Burman
Rapportens titel Utveckling av ett GIS-verktyg för analys av mätprecision samt resursallokering		
Sammanfattning (högst 200 ord) <p> Detta arbete beskriver hur ett GIS-verktyg utvecklats för ArcView 8.2. På FOI används och utvecklas metoder för att simulera nedfall av NBC-utsläpp. Dessa simulerade fält används för att bygga upp övningsscenarior för militärt och civilt försvar. För att kunna planera dessa övningar behövs ett verktyg där mätstrategier kan prövas och mätresultaten kan visas på ett överskådligt sätt. Arbetet beskriver hur ett verktyg designas och implementeras i VisualBasic 6.0 och ArcView 8.2. I verktyget konstrueras fält som beskriver hur det simulerade beläggningsfältet betar sig över tiden. Mätstrategier beskrivs genom att ett tänkt fordon färdas med bestämd hastighet över en grupp markerade linjer och gör mätningar med en bestämd frekvens. Resultatet av mätningarna kan analyseras och resultatet visas som: interpolerat fält, absolut fel, relativt fel, dos eller dosrat per mätstrategi. Det är alltså möjligt att se hur mätprecisionen påverkas av antalet mätningar samt se hur mätresurserna kan användas för att få ett så bra resultat som möjligt. Verktyget är designat för att det ska bli så flexibelt och utbyggbart som möjligt. </p>		
Nyckelord GIS, ArcView, verktyg, anpassning, beläggningsfält, VisualBasic, mätprecision, resursallokering, interpolering, IDW, Spline, Kriging		
Övriga bibliografiska uppgifter	Språk Svenska	
ISSN 1650-1942	Antal sidor: 22 s.	
Distribution enligt missiv	Pris: Enligt prislista	

Issuing organization FOI – Swedish Defence Research Agency NBC Defence SE-901 82 Umeå	Report number, ISRN FOI-R—0924--SE	Report type Methodology report
	Programme Areas 3. Protection against Weapons of Mass Destruction	
	Month year October 2003	Project no. A4521/E4812/I4210/A413
	General Research Areas 1. Policy Support to the Government	
	Subcategories 33 NBC Studies	
Author/s (editor/s) Magnus Danielson Jan Burman Jan-Olov Moberg	Project manager Ronny Bergman	
	Approved by	
	Sponsoring agency	
	Scientifically and technically responsible Kenneth Lidström, Jan-Olov Moberg, Jan Burman	
Report title (In translation) Development of a GIS tool for analysis of resource allocation and precision of measurement		
Abstract (not more than 200 words) <p>This work describes the development of a GIS-tool for ArcView 8.2. FOI is using and developing methods for simulating release of NBC agents. These simulated fields are used to develop training scenarios for military and civil defense. Scenario planning requires a tool for the examining and displaying of the results of measurements in these NBC-fields. This paper describes the design and implementation in VisualBasic 6.0 and ArcView 8.2. The tool enables the user to generate fields describing the simulated release of NBC agents over time. Measurement strategies are described by letting an imagined vehicle travel along a set of geographical lines with a fixed speed and performing measurements at a certain interval. The result of the measurements can be analyzed and displayed as: interpolated field, absolute error, relative error, load or accumulated load. It is therefore possible to examine how the precision in the measurements is affected by the number and density of measurements. The tool is designed to be as flexible and extendable as possible</p>		
Keywords GIS, ArcView, tool, add-in, Radiation field, VisualBasic, resource allocation, precision of measurement, interpolation, IDW, Spline, Kriging		
Further bibliographic information	Language Swedish	
ISSN 1650-1942	Pages 22 p.	
	Price acc. to pricelist	

Innehållsförteckning

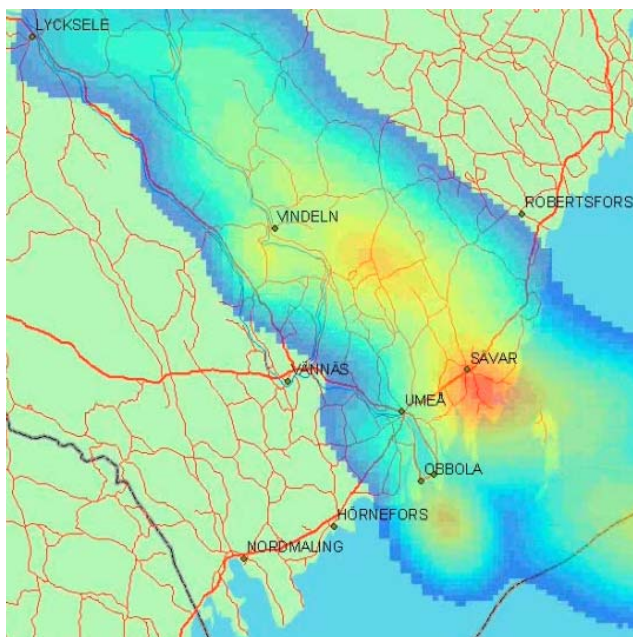
1 Inledning	5
1.1 Bakgrund.....	5
1.2 Uppgift.....	5
1.3 Avgränsningar.....	6
1.4 Rapportens disposition.....	6
2 Beskrivning av verktyg.....	6
2.1 GIS	6
2.1.1 Vektorer	6
2.1.2 Raster	7
2.1.3 Utbyggnad.....	7
2.2 Visual Basic	8
2.3 Programmering med Visual Basic och ArcView.....	8
2.3.1 Malldokument.....	8
2.3.2 Tillägg.....	8
2.3.3 Applikation	8
2.3.4 Slutsats	9
3 Design och implementering.....	9
3.1 Funktionella krav	9
3.3 Arkitektur.....	10
3.3.1 Funktionernas paketering.....	10
3.3.2 Skiktad arkitektur.....	10
3.3.3 Metodbaserad arkitektur	11
3.4 Design	11
3.4.1 Presentationsskiktet.....	11
3.4.2 Logikskiktet	12
3.4.3 Databasskiktet.....	14
4 Diskussion.....	14
4.1 Egenskaper och resultat	14
4.2 In och utdata.....	14
4.3 Arkitektur.....	14
4.4 Användarvänlighet.....	15
4.4.1 Linjestrategi	15
4.4.2 Analysdialog	15
4.4.3 Beräkningshastighet.....	15
4.5 Beräkningar.....	16
4.5.1 Sampling	16
4.5.2 Analys	16
4.6 Verktyg	16
4.6.1 ArcView 8.2.....	16
4.6.2 VisualBasic 6.0	17
4.7 Var finns kända problem.....	17
4.7.1 Accessdatabas	17
4.7.2 Enheter	17
4.9 Framtiden	17
4.9.1 Uppdatering från verkliga mätsonder	17
4.9.2 Alternativa interpoleringsmetoder	18
5 Referenser	19
6 Ordförklaringar	20
6 Ordförklaringar	20
Bilaga A: Dataflödesdiagram.....	21

1 Inledning

1.1 Bakgrund

Geografiska informationssystem, GIS, har de senaste åren blivit allt mer populära för att ta fram beslutsunderlag. GIS är ett system där användaren kan skapa, analysera och producera digitala kartor. FOI, Totalförsvarets forskningsinstitut, har startat ett projekt kallat NBC GIS. Projektet syftar till att initiera forskning och skapa beredskapstillämpningar för att analysera utsläpp, spridning, nedfall och av NBC-stridsmedel, d.v.s. nukleära, biologiska och kemiska stridsmedel. Resultat från den naturvetenskapliga beskrivningen av exponering och effekter utgör nödvändiga indata för en fortsatt samhällsvetenskaplig analys. Ett sådant verktyg för mångvetenskapligt studium av verkan på samhället och dess sårbarhet är uppenbart attraktivt och gynnsamt för att väldefinierat angripa komplexa problem.

Detta arbete är kopplat till en del i det ovan beskrivna projektet. Föreliggande rapport beskriver ett datorbaserat hjälpmedel i GIS-miljö där mätstrategier ska kunna skapas och analyseras med avseende på resursallokering och precision. Detta är intressant för spel och övning inom totalförsvaret och civilt strålskydd. Mätstrategierna appliceras på fält, som kan vara utsläpp och spridning av radioaktiv, kemisk eller biologisk agens. Det resulterande mätdatat ska kunna belysa ett flertal olika strategiska frågor. Exempel på intressanta strategiska frågor som verktyget ska kunna ge svar på är; hur bör begränsade mätningsresurser disponeras för att så tillförlitligt som möjligt återge beläggingsbilden eller hur påverkas precisionen i den mätbaserade beläggingsbilden av disponibel tid och alternativa vägval? Ett exempel på ett beläggingsfält visas i Figur 1.



Figur 1. Exempel på radioaktivt beläggingsfält över Umeås närområde.

1.2 Uppgift

Målet är att designa ovan beskrivna verktyg med hjälp av personal kopplade till projektet NBC GIS och därefter implementera en körbar version. Det verktyg som rapporten beskriver är tänkt som en förstudie där användarvänlighet och funktionalitet ska kunna undersökas för en framtida skarp version, samt fungera som en utvärdering av utvecklingsmiljön. Verktyget ska kunna användas för analys av; precision i mätningar, resursallokering samt dosbelastning. Verktygets egenskaper ska vara; öppet för nya tillägg samt generellt användbart för flera tillämpningsscenarios.

1.3 Avgränsningar

Ett flertal avgränsningar finns för att arbetet ska kunna utföras på planerad tid. Den första avgränsningen är att indata i form av rasterfält skapas av andra verktyg. Det finns flera simuleringsverktyg som används på FOI, gemensamt för dem är att de kan producera utdata i matrisform. Dessa matriser kan konverteras till ett rasterformat som verktyget kan använda. Den andra avgränsningen gäller för konstruktionen av mätstrategierna. Verktyget utgår från befintlig geografisk data som vägar och städer, dessa data förutsätts finnas tillgängliga i ArcView-kompatibelt format. Ytterligare en avgränsning är att endast de med ArcView integrerade interpoleringsfunktionerna, Inverse distant weighting, Spline och Kriging kan användas vid skattning av fältet. Slutligen behöver verktyget endast fungera under Microsoft Windows™ operativsystem.

1.4 Rapportens disposition

Avsnitt två innehåller beskrivningar av de två mjukvaror som använts. Denna översiktliga beskrivning av programmeringsmiljö och GIS-plattformen ligger till grund för den senare programutvecklingen som presenteras i avsnitt tre. Där beskrivs programspecifikationen, hur mjukvaran implementerats samt hur programmet är uppbyggt. En diskussion om resultat och användbarhet tas upp i avsnitt fyra. Ord, definitioner och förkortningar förklaras i avsnitt fem.

2 Beskrivning av verktyg

2.1 GIS

GIS-verktyg är program som låter användaren skapa, bearbeta, analysera och presentera geografisk information. De har historiskt sett krävt avancerade expertanvändare men med tiden har programmen utvecklats till verktyg för beslutsfattare och mindre avancerade användare. Två separata GIS-funktioner har växt samman de sista åren. I det ena spåret behandlas geografisk information som vektorer. Dessa vektorer kan beskriva punkter, linjer, och polygoner. I det andra spåret används raster eller grafiska bilder. Båda kan kombineras med vissa restriktioner eller konverteras fritt mellan varandra.

2.1.1 Vektorer

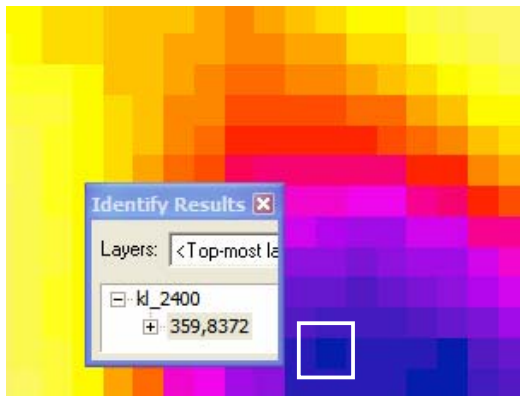
En geografisk punkt kan till exempel vara centrum av en stad i ett stort perspektiv eller toppen av ett berg. Linjer kan representera vägar, floder och andra nätstrukturer. Polygoner används för att representera en yta såsom kommuner, sjöar eller tomtmark. GIS-miljön tillhandahåller en uppsättning elementära funktioner för att hantera och bearbeta dessa element genom att kombinera dem på ett strukturerat sätt. Operationerna kan till exempel vara *union* som ger den totala gemensamma ytan, eller *intersect* som ger den överlappande ytan. GIS-miljön kan alltså hjälpa till med att hitta områden som uppfyller vissa kriterier, t.ex. för en fabrik. En sådan fråga kan formuleras som; visa all yta som ligger 200 meter från vägar, marken ska bestå av lera, 1 km till närmaste vattentäkt, minst en brandstation innanför området och den totala ytan ska vara minst 20 ha. I Figur 2 finns Umeå med dess omkringliggande vägnät uppbyggt av vektorer. Vektorer är lämpligt att använda för att beskriva diskret data som marktyper, t.ex. lera, sand, mo och mjåla.



Figur 2. Umeå med angränsande vägnät, exempel på vektorer.

2.1.2 Raster

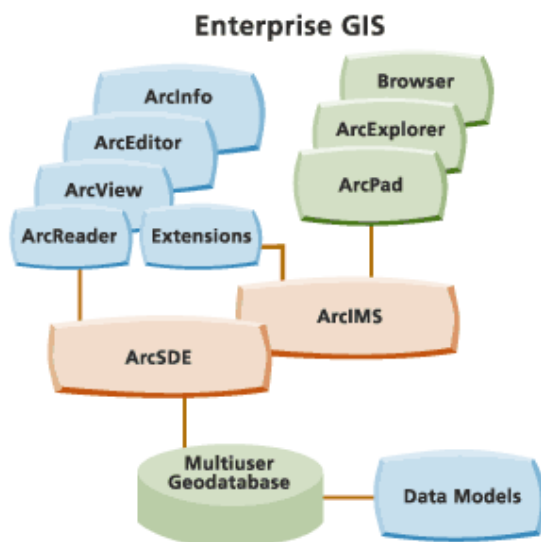
Det andra spåret behandlar geografisk information i form av raster, en matris med en definierad cellstorlek och ett värde som beskriver den geografiska informationen. Varje cell representerar alltså en given yta, t.ex. 300 meter x 300 meter. Alla celler i matrisen ligger kant i kant och värdet har ofta en enhet som är av kontinuerlig typ t.ex. [mm regn/h]. Raster används ofta för att beskriva gradvisa övergångar, t.ex. är saltkoncentrationen i mark mer lämplig att beskriva med raster än polygoner. Flera raster kan sedan kombineras för att nå ny information. Om hypotesen är att erosion beror av regn, jordtyp och markens lutning kan dessa klassificeras lämpligt. T.ex. ge varje cell i regnrastret ett värde mellan 1 och 10 beroende på regnmängd. Ge varje cell värde i jordrastret ett värde mellan 1 och 10 beroende på erosionskänslighet. Sätt slutligen ett värde i markrastret mellan 1 och 10 beroende på markens lutning. Denna information kan sedan kombineras genom en lämplig matematisk operation, t.ex. $\text{lutning} * \text{regn} * \text{jordtyp} = \text{potentiell erosion}$. I Figur 3 visas ett beläggningsfält där varje cell i rastret har ett värde. I figuren ser vi att den markerade cellen har värdet 359,8372.



Figur 3. Ett beläggningsfält i rasterformat.

2.1.3 Utbyggnad

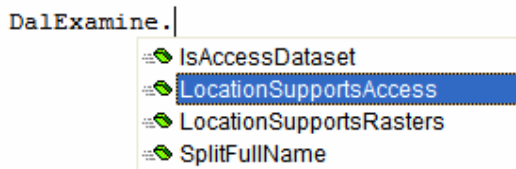
Till det här projektet används ArcView som är en av produkterna i ESRI:s ArcGIS familj. Den egentliga styrkan i produkten ArcView, som är ett GIS-verktyg, sitter inte i den medföljande funktionaliteten utan i den stora mängden tillägg som utökar funktionaliteten inom det specialområde som användaren är intresserad av. Det kan till exempel vara vattenmodellering, erosionsmodellering eller biotoputbredning. Den öppna arkitekturen gör ArcView till ett lämpligt verktyg att bygga ut i det aktuella projektet. Utöver ArcView finns en serie andra produkter i ArcGIS-familjen, se Figur 4. Detaljerna kring dessa produkter lämnas utanför denna presentation, för mer information se ESRI:s websida <http://www.esri.com>.



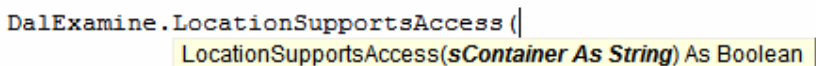
Figur 4. Produkter i ArcGIS familjen från ESRI.

2.2 Visual Basic

Visual Basic är en populär utvecklingsmiljö för operativsystemet Microsoft Windows™. Verktøjets styrka ligger i enkelheten att snabbt kunna designa, implementera och testa både program med och utan grafiska gränssnitt. Miljön döljer komplicerade och intrikata samband från operativsystemet, t.ex. fönsterhantering och händelser. Det är möjligt att bygga objektorienterade program och programmeraren får hjälp av IntelliSense och Statecompletion, se Figur 5 och Figur 6. IntelliSense gör att alla tillgängliga variabler, metoder och egenskaper visas för användaren. Det minskar behovet att söka i referenslitteratur för att finna korrekt namn samt minskar risken för felstavningar som orsakar kompileringsfel. StateCompletion gör att alla in och ut parametrar för en metod visas, vilket också minskar behovet att söka i referenslitteraturen. Nackdelarna är att där programmeringsmiljön förenklar för programmeraren, främst vid skapandet av fönster, skapas ibland också hinder för full kontroll vilket kan vara en nackdel för den mycket avancerade programmeraren. Eftersom den slutliga produkten ska köras i Microsoft Windows™-miljö blir frågan om operativsystemberoendet överflödig. Det helt avgörande skälet att använda VisualBasic i det här arbetet är den överlägsna produktivitet man får gentemot andra liknande miljöer och språk. Hastigheten på den slutliga produkten blir oftast tillräckligt hög för att inte behöva gå över till C++. Det egentliga arbetet utförs oftast av ArcView, VisualBasic koden ser bara till att rätt information hämtas från användaren och skickas till ArcView i rätt följd.



Figur 5. IntelliSense i VisualBasic



Figur 6. StateCompletion i VisualBasic

2.3 Programmering med Visual Basic och ArcView

Tre alternativ finns att bygga ut ArcView: malldokument, tillägg eller eget gränssnitt. Varje alternativ har sina klara för- och nackdelar.

2.3.1 Malldokument

Malldokument kan med fördel användas när funktionaliteten behövs för att skapa nya dokument av en speciell typ, t.ex. en bygglovsansökan. Tekniken är identisk för mallar i Office-paketet, Word, Excel och Power Point. Nackdelen är att funktionaliteten kopplas till ett dokument och inte till programmet. Om funktionaliteten krävs för en annan typ av dokument blir arbetet för användaren besvärligt. För att bygga malldokument i ArcView 8.0 och senare versioner används Visual Basic for Applications, VBA. Utvecklingsmiljön är integrerad i värdprogrammet, i detta fall ArcView. Från VBA kan man bara bygga projekt till värdprogrammet medan Visual Basic tillåter att man bygger helt andra typer av projekt. Språket och syntaxen är dock helt identisk i Visual Basic och i VBA.

2.3.2 Tillägg

Det andra alternativet är att skapa ett tillägg till ArcView. Då integreras funktionaliteten med ArcView och användaren kan använda den på all typ av data som funktionen hanterar. Ett typexempel här är rättstavningsfunktionen i Office, en funktion som ska fungera på samma sätt oavsett om dokumentet är en ansökan eller ett kontrakt. Nackdelen med ett tillägg är att funktionen måste installeras på användarens dator och kan inte enkelt skickas vidare som malldokumentet. Dessa tillägg kan bara byggas från Visual Basic miljö.

2.3.3 Applikation

Det tredje alternativet är att bygga ett helt eget gränssnitt ovanpå ArcView och bara exponera den funktionalitet som användaren anses behöva. Det kan vara lämpligt om användaren är ovan men är

samtidigt en nackdel då arbetsgången blir låst. Nya uppdateringar och tillägg till ArcView kommer inte användaren till nytta vilket är en brist. Applikationer kan endast byggas från Visual Basic miljön.

2.3.4 Slutsats

Under arbetets gång har både mallar och tillägg använts. Först har funktionaliteten testats i en mall från Visual Basic for Applications miljön. Den är starkt integrerad med ArcView och alla aktuella dokumentobjekt är lätt åtkomliga. Det går också smidigt att hoppa mellan VBA och ArcView för att testa förändringar av koden. När funktionalitet är på plats har koden flyttats och putsats i Visual Basic 6.0. Några små förändringar brukar krävas men övergången är tillräckligt problemfri. Enda skälet till att börja i VBA överhuvudtaget är för att man slipper kompilera koden. När koden kompileras får inte ArcView vara startat för att alla uppdateringar ska gå bra att genomföra. VBA kräver att ArcView är igång och man slipper alltså starta om ArcView vid varje ny testkörning.

Alla objektinstanser av ArcViews användargränssnitt når man på ett föredömligt sätt. Det är också väldigt lätt att bygga ut eller förändra den befintliga miljön genom att registrera nya användarkomponenter i ett verktyg som medföljer vid installationen av ArcView. Om man inte uppskattar hur en dialogruta fungerar kan man alltså byta ut den mot sin egentillverkade och även ArcView kommer att använda den vid nästa tillfälle.

Data till detta projekt måste finnas i en databas och en av de viktigaste klasserna kallas då FeatureClass. Det finns också en uppsättning gränssnitt till varje klass. Egentligen används den engelska termen interface oftast också på svenska, hädanefter används termen interface. För att direkt manipulera, läsa, förändra, söka och uppdatera en FeatureClass finns en uppsättning interfaces.

3 Design och implementering

3.1 Funktionella krav

Enligt diskussioner med projektgruppen för NBC GIS enades vi om följande lista på nödvändiga funktioner, se Tabell 1.

Tabell 1, Funktioner som verktyget ska implementera

Funktion	Beskrivning
Fältdefinition	Peka ut ett antal raster som ska ingå i ett spridningsfält. Varje raster måste kunna definieras i tidsdimensionen.
Mätstrategi	Peka ut flera geografiska punkter och ange en tidpunkt när och hur ofta en mätning ska ske.
Sampling	Använda mätstrategier och fältdefinitioner för att skapa en mätresultatmängd.
Analys interpolering	Interpolera fram fältet från mätdata till en specifik tidpunkt.
Analys mätpatrull	Se dos och dosrat som funktion av tiden för enskild patrull.
Analys utvärdering	Skatta hur bra en interpolering blir jämfört med simulerade data.

Varje funktion består av ett antal indata, en process samt en uppsättning utdata enligt Bilaga A. Överst i figur 12 sitter användaren som matar in data till respektive process. Varje process skapar utdata som sedan sparas i en databas eller visas för användaren. Denna generella bild används för att beskriva varje specifik process. I figur 12 visas även informationsflödet från användaren när analysprocessen används. Här har både indata och utdata angivits i figuren. En detaljerad lista över varje funktions indata, process samt utdata finns i Tabell 2.

Tabell 2, Indata och utdata för processerna

Process	Indata	Utdata
Fältdefinition	Ett eller flera raster En tidpunkt per raster	En XML-fil som beskriver fältet enligt indata
Mätstrategi	En eller flera geografiskt placerade punkter. En tidsangivelse per punkt Ett namn per patrull i strategin.	En punkt FeatureClass med attribut för tid och patrullnamn.
Sampling	En eller flera mätstrategier. En tidsförskjutning per mätstrategi Ett eller flera fält.	En FeatureClass med alla mätstrategiattribut samt ett attribut för mätvärdet.
Analys interpolering	En sampling Interpoleringsmetod <i>Tillval:</i> Ett tidsfilter. En tidsberoende funktion som påverkar mätenheten. En geografisk mask där mätvärden saknas.	Ett raster som beskriver fältskattningen.
Analys mätpatrull	Samma som analys-interpolering. En utvald mätpatrull En tidsberoende funktion som påverkar dos/dosraten.	Ett dos-tids diagram för mätpatrull
Analys utvärdering	Samma som analys-interpolering.	Antal mätpunkter Relativa felet Riktiga fältet Absoluta felet Täckt yta

3.3 Arkitektur

Enligt de funktionella och kraven ovan har följande arkitektur använts.

3.3.1 Funktionernas paketering

Slutsatsen blir att funktionaliteten ska implementeras som ett tillägg då det är önskvärt att nå befintlig och framtida funktionalitet i ArcView samt att beroendet av enskilda dokument skulle skapa problem för användaren. Nackdelen med installationen bör kunna minimeras till en hanterbar nivå.

3.3.2 Skiktad arkitektur

Tre distinkta skikt identifieras som kallas presentation, logik och data. Denna uppdelning syftar till att isolera funktionalitet från varandra så att felsökning och senare utbyggnad ska bli så enkel som möjligt.

Genom det nya verktygsfältet nås huvudfunktionsdialogerna fält, mätstrategi, sampling och analys. Dessa dialogrutor kallas användargränsnittet eller presentationsskiktet. Två uppgifter finns för presentationsskiktet, den ena är att inhämta och validera all information som användaren matar in. Den andra uppgiften är att publicera resultat och eventuella felmeddelande för användaren. Om någon av de två uppgifterna kräver en logisk analys så ska den i mesta möjliga mån utföras av det logikskiktet.

Logikskiktet i sin tur sköter all logik som behövs för att de funktionella kraven ska uppfyllas. I detta skikt sorteras, analyseras och samplas data. De egentliga processerna i Tabell 2 finns alltså implementerade

här. Logikskiktet är vidare beroende av dataskiktet som sköter kommunikationen ner till datakällan, egentligen ArcView komponenter.

Dataskiktets syfte är att tillhandahålla en enkel och väldefinierad struktur för hur data hämtas och uppdateras. När denna funktionalitet bryts ut ur logikskiktet, är det lättare att optimera och strömlinjeforma kommunikationen med datakällan.

Genom att använda denna arkitektur uppnås ett fundamentalt mål, nämligen separation. Felsökning och framtida uppdatering blir enklare när skikten är väl avgränsade. Hela systemet blir lättare att testa om separata funktioner har paketerats tillsammans. I ett framtida scenario blir det också betydligt lättare att placera logik och databasskiktet på en server och byta ut presentationsskiktet mot en webbtjänst, webservice och eller en WAP-portald (Kirtland, 1999)

3.3.3 Metodbaserad arkitektur

En svårlöst konflikt vid programmering är huruvida arkitekturen ska prioritera snabbhet vid skapandet av objekt eller minimera de resurser ett objekt kräver. I det första fallet låter man ett skapat objekt leva så länge som möjligt och åtkomsten av det blir därför snabb. Problemet är att resurser blir låsta till objektet även fast de kanske inte används. Resurser kan vara minnen, kopplingar till databaser och liknande. I det andra fallet låter man objekt leva så kort tid som möjligt och släpper alla resurser man efterfrågat. Nackdelen här är att vissa resurser tar lång tid att skapa, så som databaskopplingar. Vilken arkitektur man väljer beror på vilken miljö som programmet kom att arbeta i och vilka resurser som finns tillhanda. I det här projektet kommer ArcView-komponenterna att skapas oftare än de egentillverkade komponenterna. Det blir alltså ganska ointressant att försöka spara några millisekunder på att hålla i resurser jämfört med risken att minnet börjar ta slut eller databasen inte kan leverera flera kopplingar.

Slutsatsen blir att komponenterna kommer att bygga på en metodbaserad design. I mesta möjliga mån ska endast ett anrop ske till komponenten som sedan destrueras för att släppa resurserna så fort som möjligt. I metodanropet ska all information som komponenten behöver skickas med. Egenskaper ska alltså inte sättas på komponenten innan anropet. Om ett framtida scenario innebär att komponenterna placeras på en server kommer ännu större vinster då varje tur-och-retur anrop till servern kostar resurser. Om funktionaliteten redan är implementerad för att köras med ett anrop är mycket vunnet. En server, Microsoft Windows™ 2000 eller senare, har också stöd för object-pooling vilket löser problemet med långa instansieringstider, tiden det tar att skapa ett objekt. Servern har redan ett antal fördefinierat skapade komponenter och plockar fram dem när de efterfrågas av klienten. Denna idé påminner om det vatten som TV-kocken redan har kokandes i kastrullen. (1998, Ted Pattison)

3.4 Design

Enligt arkitekturparagrafen finns tre logiska skikt som beskrivs i detalj nedan. Presentationsskiktet hämtar och verifierar informationen innan den skickas vidare till logikskiktet. Där bearbetas och analyseras informationen. Om logikskiktet behöver läsa eller skriva information från databasen sker det genom databasskiktet.

3.4.1 Presentationsskiktet

Presentationsskiktet ansvarar för att verifiera informationen som användaren matar in. All information som skickas till logikskiktet ska vara korrekt och kontrollerad så att logikfunktionen inte behöver verifiera detta.

3.4.1.1 Fält

Om användaren ändrar tidsstämpeln kontrolleras att det inmatade värdet är en siffra större än 0. Kontroll sker också så att minst ett raster är valt. Rasterlistan sorteras med avseende på tidsstämpeln innan den skickas till logikskiktet. Endast ESRI Grid raster kan hanteras av verktyget. Andra typer av rasterdata kan enkelt konverteras till ESRI Grid raster i ArcViews eget gränssnitt.

3.4.1.2 Mätstrategi Linje

Alla tillgängliga linjelager läggs till rulltextrutan. Om användaren väljer ett annat lager i rulltextrutan kommer endast objekt i det lagret kunna väljas. Hastighet och periodicitet kontrolleras så att de representeras med en siffra/heltal. Alla objekt som väljs, läggs till i listan för utvalda objekt. Namnet plockas från det fält som angivits som DisplayField i lagrets egenskaper.

3.4.1.3 Mätstrategi Punkt

På samma sätt som i mätstrategidialogrutan finns en rulltext och en lista av utvalda objekt. Funktionen finns beskriven i 3.4.1.2. En rulltextruta används för att välja huruvida mätstrategin ska beskriva en mobil patrull eller flera stationära. För stationära mätstrategier öppnas textrutor där antal mätpunkter och periodicitet kan skrivas in. Båda dessa kontrolleras att de är siffror.

3.4.1.4 Sampling

Dialogrutan låter användaren peka ut ett eller flera fält som ska användas vid samplingen. En eller flera mätstrategier kan också pekas ut. Det kontrolleras att minst ett fält och minst en strategi har valts. Denna information skickas sedan till logikskiktet som startar samplingen. Den resultatmängd som samplats sparas ner av logikskiktet.

3.4.1.5 Analys







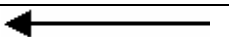
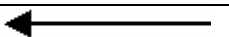
En samplingsfil väljs av användaren. Metadata hämtas som pekar ut källan till samplingen. Det kan vara en eller flera fält samt en eller flera mätstrategier. Rulltexten fylls med de fält som samplingen innehåller. Användaren kan alltså bara analysera ett fält i taget. Alla mätstrategier dyker upp i textrutorna med en kryssruta intill. En kontroll sker att minst en kryssruta är ifylld. Om tidsfiltret kopplas in kontrolleras att från- och till- tiderna är en siffra mellan 0 och 31536000. All denna information skickas till logikskiktet som startar analysen.

3.4.2 Logikskiktet

Hela logikskiktet är implementerat som en COM-komponent som beskrivs i detalj nedan.

3.4.2.1 GeoRelation

Syftet med klassen är att analysera sambandet mellan två punkter, mellan punkt och linje samt mellan linje och linje. Tre av metoderna används för att se om två punkter alternativt linjer är geografiskt identiska. En av de mer intressanta metoderna kallas GetDirection. Indata är två linjer, låt oss kalla dem linje1 och linje2. Utdata är den riktning man färdas i om man går från linje1 till linje2 med avseende på linje1, se Tabell 3. Det returnerade värdet är en Enum, en datatyp som är uppräkningsbar. Ett exempel på uppräkningsbar data är namn på bilmärken. Den enum som används här kan bara anta något av värdena DirUnkown, DirForward och DirReverse. Metoden returnerar normalt DirForward eller DirReverse och endast DirUnkown om något går fel.

Linje1	Linje2	Utdata
		DirForward
		DirForward
		DirReverse
		DirReverse

Tabell 3. De fyra alternativen som reds ut i GetDirection

3.4.2.2 FoiMetaData

Syftet med klassen är att få ett enkelt sett att hämta, uppdatera och skapa metadata för mätstrategier, rasterfält samt resultat data. Den innehåller både enkla metoder som kontrollerar att det finns metadata samt mer komplicerade metoder som hämtar metadata och skapar lätthanterliga listor av informationen. Metadata kan användas för att följa hur datat har skapats. Ett enkelt sett att titta på metadata är att använda det stylesheet, FOI.xsl som bifogats på installations CD-skivan. Se manualen till ArcView 8.2 för installationsinstruktioner av metadata. När installationen är genomförd används ArcCatalog för att titta på metadata.

3.4.2.3 LineScenario

Syftet med klassen är att skapa en mätstrategi från en sorterad men icke orienterad lista av linjer. Listan innehåller referenser till de OIDs som identifierar linjerna i en ArcView fil. OID är en förkortning för Object ID, vilket är ett unikt nummer för varje objekt. Varje punkt, linje eller polygon har ett eget unikt OID. Listan är sorterad med avseende på att den tänkta ordningen som mätpatrullen ska röra sig men icke orienterad med avseende på att de enskilda linjerna inte behöver vara vända i rätt riktning, se 3.4.2.1. Huvudmetoden skapar en mätstrategi med hänsyn till hastighet och mätfrekvens. Tre undermetoder exponeras publikt så att andra utvecklare kan återanvända funktionaliteten. De tre undermetoderna ansvarar för att bygga topologin på de icke orienterade linjerna.

3.4.2.4 PointScenario

Klassen har liknande funktion som 3.4.2.3. Det finns två huvudsakliga metoder, en stationär och en mobil. Båda skapar mätstrategier med utgångspunkt från OID's, punkter i en PolygonFeatureClass. Indata är periodtid mellan varje mätning, samt antal mätpunkter om det är en stationärt strategi. Den mobila tar mätpunkterna i turordning och räknar upp tidsattributet för varje mätpunkt. Den stationära skapar ett antal angivna mätpunkter på samma plats med givet tidsintervall. Den stationära metoden kan vara lämplig för att representera brandstationer eller liknande där man vet att mätningar görs regelbundet. Den mobila används förmodligen mer sällan och endast då ett spår skapats från en annan källa.

3.4.2.5 EvaluateRaster

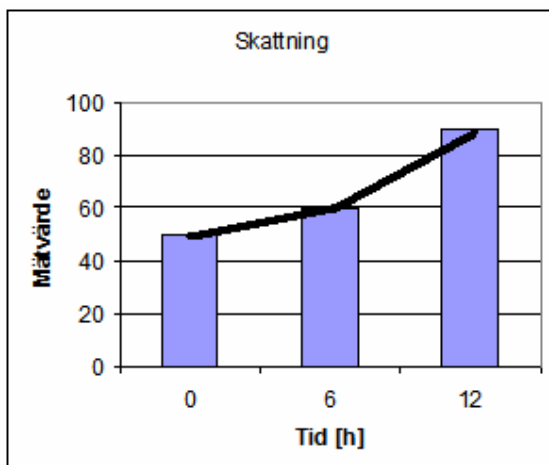
Klassen samlar de metoder som behövs för att kunna interpolera fram raster från en filtrerad sampling samt göra beräkningar på det skattade rastret. De tre interpoleringsmetoderna, IDW, Kriging samt Spline använder sig av de inbyggda interpoleringsfunktioner som finns i Spatial Analyst. Indata är en sampling, egentligen en PolygonFeatureClass där ett av punkternas attribut beskriver mätvärdet. Utdata är ett raster med samma utbredning som de yttersta punkterna i PolygonFeatureClassen.

3.4.2.6 BufferCalculator

För att kunna bedöma hur stor yta som täckts av alla mätstrategierna dras en buffer runt varje punkt med 20 meters radie. Dessa ytor överlagras sedan där de täcker varandra. Därefter kan den totala täckta ytan beräknas. Metoden tar som indata den filtrerade PolygonFeatureClassen som också används vid interpoleringen. Den egentliga radiens storlek varierar med olika faktorer. Radien skattas utifrån den joniserande strålningens räckvidd. Om vi tänker oss en radioaktiv beläggning som finns överst på markytan så ser detektorn en radie av 50 - 100 meter eller längre om detektorn sitter en meter ovan mark. Om vi flyger så minskar "konen" d.v.s. den bidragande arean, (Finck 1992). Om radioaktiviteten har trängt ner i marken minskar radien motsvarande för att kanske i extrema fall vara under 10 meter, p.g.a. av den mängd markmaterial strålningen måste passera. Detta innebär att radien måste skattas inför varje mätupdrag med en hypotes var radioaktiviteten befinner sig och på vilken höjd detektorn är. Så markens densitet, nedträngningsdjupet för radioaktiviteten, mängden luft mellan radioaktiviteten och detektorn samt avståndet mellan dessa påverkar den egentliga radien. I denna version kan radien inte ändras.

3.4.2.7 Sampler

Klassen samlar de metoder som krävs för att kunna sampla ett antal fält med ett antal mätstrategier. Indata är ett fält samt en eller flera mätstrategier. Om tidpunkten för samplingen ligger mellan två raster i fältet görs en linjär skattning av värdet mellan dessa två tidpunkter, se Figur 7. Varje stapel representerar värdet för en cell i ett raster. I figuren finns tre raster för tiderna 0, 6 och 12 timmar. Om en mätning ska utföras efter 9 timmar kommer samplerfunktionen returnera ett värde på ungefär 73. Utdata är en PointFeatureClass som innehåller alla använda mätstrategier samt mätvärde för varje punkt.



Figur 7. Linjärskattning av mätvärden mellan raster i samplerklassen.

3.4.3 Databasskiktet

Databasskiktet är byggt som en COM-komponent som exponerar funktionalitet för att förenkla och optimera läsandet och skrivandet av data från ArcView-filer.

3.4.3.1 WorkspaceSource

Syftet med klassen är att tillhandahålla en källa för att skapa ett Workspace objekt. I ArcView finns flera sätt, några snarlika och med samma funktion och några som ska hanteras väldigt annorlunda. Att endast ha en klass för att skapa Workspace-objekt underlättar därför programmering och felsökning. I ett framtida scenario där andra källor än ArcView ska användas kan Workspace-objektet anpassas för dessa nya krav och logikskiktet dra nytta av detta direkt eller med smärre justeringar. I nuvarande version kan Workspace-objektet endast hantera Accessdatabaser samt rasterkataloger.

3.4.3.2 DataSet

DataSet-klassen är en utgångspunkt för att öppna och spara specifika georelaterade filer som raster och FeatureClasser. Den används också för att uppdatera information i FeatureClassen genom InsertFeature metoden.

3.4.3.3 GxFilterXml

Syftet med klassen är att bygga ut funktionaliteten i ArcView-inbyggda dialogrutor för att öppna och spara filer. Klassen tillåter användaren att hantera xmlfiler som i den här applikationen representerar metadata.

3.4.3.4 DataCopy

På grund av en begränsning i ArcViews inbyggda klasser måste ibland information mellanlagras i databasen innan den används igen. Klassen innehåller metoder för att kopiera hela eller delar av en PointFeatureClass. Indata är en FeatureClass, namn på ny FeatureClass samt eventuella filter.

3.4.3.5 Dialog

Används för att visa ArcViews inbyggda filhanterare. Filer kan öppnas eller sparas av olika fördefinierade typer som FeatureClass, raster eller xml. Dialogrutan returnerar sökvägen som en sammansättning av katalogen och filnamnet. Det kan och bör ifrågasättas varför dialogfunktionen lagts i dataskiktet. Det vore kanske rimligt att placera det i presentationsskiktet eftersom det direkt kommunicerar med användaren. Skälet till dess nuvarande placering är att dataskiktet och dialogen sällan eller aldrig utnyttjas utan varandra.

3.4.3.6 Examine

Detta är en hjälpklass för att kunna hantera databasen på ett säkert sätt. De viktigaste metoderna beskrivs nedan. IsAccessDataset metoden undersöker om en sökväg till en FeatureClass verkligen är en FeatureClass i en accessdatabas. LocationsSupportsAccess undersöker om sökvägen kan användas för att skapa en ny FeatureClass. SplitFullName returnerar sökvägen eller filnamnet från en sträng där de slagits samman.

4 Diskussion

4.1 Egenskaper och resultat

Egenskaperna öppna och utbyggbara är uppfyllda genom den 3-skiktade lösningen. Det går lätt att lägga till ny funktionalitet och använda data från andra källor. Precision- och resursallokeringsanalysen kan genomföras genom att iterativt studera resultatet i analysdialogen. Där kan också dos/dosrat för respektive mätpatrull beräknas.

4.2 In och utdata

De definitioner som sattes upp i projektets inledning för in- och utdata var tillräckliga för att alla processer skulle kunna hämta och leverera information. Det kan vara lämpligt för en framtida version att fundera på exakta fältdefinitioner för varje datatyp. Om nya tillägg lätt ska kunna integreras så måste dataformatet vara väldefinierat. Här underlättar dataskiktet då det finns färdiga funktioner för att konsistent bygga mätstrategier och mätdata.

4.3 Arkitektur

Uppdelningen i flera skikt visade sig vara ett effektivt sätt att återanvända kod samt isolera problem. Tyvärr är uppdelningen inte helt genomförd då presentationsskiktet direkt tillåts kommunicera med dataskiktet. De extra klasser som skulle ha behövt implementeras skulle förmodligen minska risken för

framtida buggar samtidigt som det skulle bli väldigt små vinster i nuvarande version. Om applikationen ska byggas ut i framtiden skulle jag gärna se att separationen genomfördes helt. Då skulle en eller två klasser läggas till logikskiktet med funktioner för att öppna, uppdatera och spara data. Dessa klasser skulle alltså fungera som skal till det riktiga dataskiktet.

4.4 Användarvänlighet

Det finns två dialogrutor som är mindre användarvänliga, nämligen linjestrategi samt analys. Här kan en insats göras för att förenkla användandet av applikationen. I övrigt är dialogerna lätta och snabbhanterade.

4.4.1 Linjestrategi

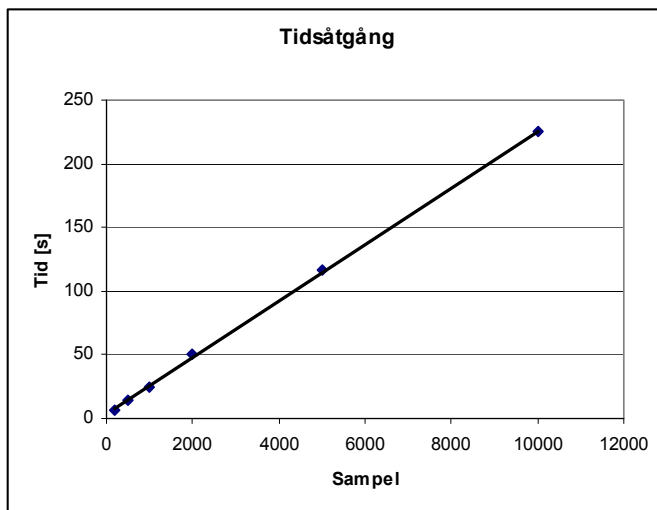
Förmodligen vill man snabbare kunna välja väg i linjestrategidialogen. Den är också något långsam vilket kan störa användaren. Det vore bekvämt att bara föra musen över vägnätet så väljs automatiskt den enklaste vägen.

4.4.2 Analysdialog

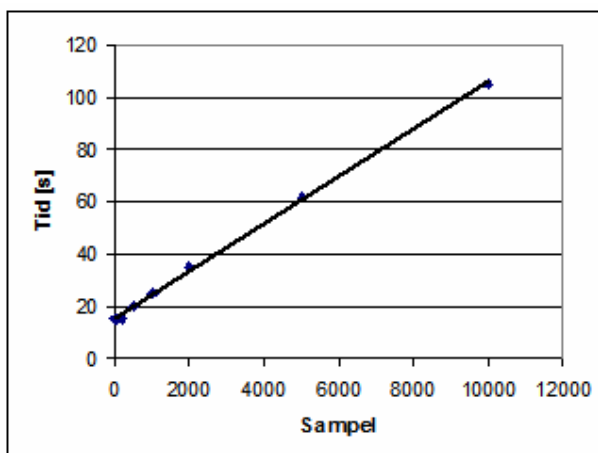
Analysdialogrutan är för komplex då många alternativa val är tillgängliga. I en framtida version vore det önskvärt att kunna individanpassa dialogrutan så att de alternativ man var intresserad av att ändra fanns tillgängligt och resterande val är gjorda på förhand. Det skulle göra den mer lätthanterlig och överskådlig.

4.4.3 Beräkningshastighet

För att verktyget ska vara enkelt att använda krävs en rimlig responstid. I Figur 8 och Figur 9 ser vi att responstiden är helt linjär mot belastningen. Detta är positivt då det betyder att verktyget inte kan överbelastas och responstiden kan förbättras med bättre hårdvara. Figur 8 och Figur 9 har producerats på PIII, 497MHz, 576Mb Ram under Microsoft Windows™ XP sp1.



Figur 8. Tidsåtgång för samplingsfunktionen.



Figur 9. Tidsåtgång för analysfunktionen.

4.5 Beräkningar

Egentligen genomförs bara två beräkningar, den ena vid samplingsstillfället och den andra vid analysen.

4.5.1 Sampling

Samplingsfunktionen interpolerar mätvärden i tidsdimensionen där det saknas riktiga data. Eftersom fälten byggs upp av diskreta raster kommer alla mätpunkter mellan dessa raster behöva interpoleras. Detta görs med en linjär interpolering. Det kan vara av intresse att studera hur beläggningen i olika fälttyper förändras över tiden med kortare tidsintervall. Man skulle då få en uppfattning om den linjära interpoleringen är tillräckligt bra eller kanske ska bytas ut mot en mer avancerad.

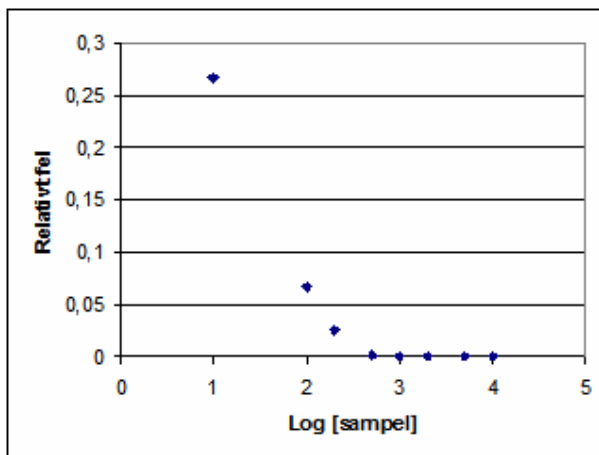
4.5.2 Analys

Skattning av felet är intressant för att kunna ge en bild av hur väl mätstrategierna faller ut. Det finns flera sätt att skatta felet och här har valts en teknik som i stort påminner om minsta kvadratskattning.

Beräkningen görs enligt ekvation 1. I ekvationen representerar s_{ij} cellvärdena i det skattade fältet och r_{ij} cellvärdena i det simulerade fältet.

$$Felet = \sqrt{\frac{\sum (s_{ij} - r_{ij})^2}{\sum r_{ij}^2}} \quad \text{Ekvation 1}$$

En av verktygets uppgifter är att analysera när det finns tillräckligt med information. Detta är nästa steg i analysen och en metod att bestämma vad som är tillräckligt är att studera hur det relativa felet ändras med avseende på antal samplingar. Denna kurva visas i Figur 10. För att skapa kurvan placerades mätpunkter ut slumpmässigt i ett definierat område. Detta område hade storleken av 100x100 celler i rastret. Analysen gjordes sedan med olika antal mätpunkter. I detta exempel tycks 1000 mätpunkter ge tillräckligt bra interpolering. Eftersom vi samtidigt vet hur snabbt den analyserar kan vi avgöra vad som är tillräckligt bra och rimligt lång väntan för resultat.



Figur 10. Relativa felet med avseende på antal sampel där ytan bestod av 100 x 100 celler.

4.6 Verktyg

De verktyg som valts kommer att påverka resultat. Det är därför intressant att ifrågasätta hur väl avpassade de var för uppgiften.

4.6.1 ArcView 8.2

Generellt sett är ArcView ett kraftfullt verktyg och är från början designat för att andra ska göra tillägg. Det finns mycket information både i tryckt och i elektronisk form. Det finns dock ett grundläggande problem med ArcView 8.2 och det är sättet som vektordata hanteras och processas. ArcView använder sig

av ett antal interfaces för att kommunicera med databasen. Dessa interfaces är inte tillräckligt väl genomarbetade. Indata och utdata till metoderna är inkonsekventa. Ett exempel på detta är de två sökfunktionerna, den ena verkar på attributdata om objekten och den andra på den spatiella informationen om objekten. Utdata från dessa metoder är av typen IFeatureCursor, ett interface där man kan steg för steg undersöka varje objekt. Det går inte att göra en ny sökning från IFeatureCursor utan att först mellanlagra informationen i en databas och sedan öppna den igen. Detta gör upprepade sökningar ineffektiva.

Data som visas på skärmen har fler och mer flexibla interfaces att arbeta mot. Flera av de inbyggda funktionerna förväntar sig också att FeatureClassen är öppnad och visas grafiskt för användaren för att de ska fungera ordentligt. Några av de inbyggda funktionerna är endast möjliga att arbeta med om användaren har markerat ett eller flera objekt. Eftersom en FeatureClass som inte visas på skärmen saknar markerade objekt kan dessa funktioner inte användas utan att blanda in användaren. Det är också i många sammanhang intressant att kunna jobba dolt i bakgrunden och bara besvara användaren med intressant och relevant information som delresultat.

En annan konkret begränsning är oförmågan att kunna arbeta med data som kopplats loss från databasen. I princip alla funktioner som rör FeatureClass kräver en fast koppling med databasen. Det skulle vara önskvärt att kunna koppla loss data från databasen, göra multipla spatiella och eller attributsökningar, förändra informationen och sedan åter synkronisera den med databasen. För en framtida version av ArcView kan vi hoppas på att det är lika lätt att arbeta med data i bakgrunden som den som visas på skärmen.

4.6.2 VisualBasic 6.0

I stort uppfyllde utvecklingsverktyget VisualBasic 6.0 de krav som ställdes för att lösa uppgiften i examensarbetet. Produkten och resultatet var stabilt genom hela arbetet. Det gick fort att använda trots viss egen ovana med språket VisualBasic. Det som saknades var ett enkelt sätt att kontrollera hur de öppnade dialogrutorna skulle bete sig. Två mål visade sig svåra att nå samtidigt. Det första målet var att minimera de genom VisualBasic öppnade dialogrutorna när användaren minimerar ArcView. Det andra målet är att användaren ska kunna arbeta med VisualBasic-fönstret på samma gång som ArcView-fönstret. Det gick inte med min kunskap att lösa. Några liknande typer av problem uppstod där det rent tekniskt borde vara möjligt men VisualBasic-miljön satte upp en gräns för vad som kunde göras. Dessa problem var dock så ringa eller överkomliga att jag anser VisualBasic 6.0 vara en lämplig produkt för att bygga ut andra system som ArcView 8.2.

4.7 Var finns kända problem

Här beskrivs de kända problem som finns i verktyget vid tiden för rapportens färdigställande.

4.7.1 Accessdatabas

FeatureClass kan bara sparas och öppnas från en Accessdatabas. Det betyder att shapefiler eller coverages inte kan användas. Skulle andra applikationer skapa mätstrategier eller samplingsar så måste de först flyttas till en Accessdatabas. Detta kan enkelt göras i ArcCatalog.

4.7.2 Enheter

Verktyget är helt oberoende av vilka enheter som används för fälten. Däremot är all analys av kartorna beroende av att enheten är satt till meter. Detta är standard för RT 90. Är det inte gjort kan man enkelt ändra det genom ArcView.

4.9 Framtiden

Under arbetets gång uppstår alltid nya idéer som lämnas till det uppföljande projektet. De har samlats under denna rubrik

4.9.1 Uppdatering från verkliga mätsonder

Det kan vara intressant för både övning och praktisk användning att få en direkt kanal att uppdatera databasen med data från riktiga mätsonder. Det enklaste och mest användbara alternativet torde vara en webservice som kan anropas från en enhet som kan kopplas upp mot Internet. Flera enheter skulle alltså simultant kunna skicka indata till databasen och användaren på kontoret kan kontinuerligt uppdatera interpoleringen. För att uppnå detta skulle ytterligare en klass läggas till logikskiktet som sedan

uppdaterade databasen. Logikskiktet kan med små medel exponeras som en webservice. Eftersom samplingen görs direkt kan analysdialogen direkt arbeta mot de inrapporterade värdena. I praktiken kan då övningsledaren se hur beläggningsfältet interpoleras fram när mer data anländer till databasen.

4.9.2 Alternativa interpoleringsmetoder

Det kan uppstå ett behov att använda mer komplexa interpoleringsmetoder. Ett alternativ vore att använda en flervariabel version av de kända IDW, Spline eller Kriging. Då kunde annan känd information som aktuell nederbörd, temperatur och mätvärdet kombineras för att göra ännu bättre skattningar. Ett annat alternativ vore att använda helt andra interpoleringsmetoder som skraddarsyfts för en viss typ av fält. Neurala nätverk skulle kunna användas för att utnyttja redan känd information om fälttypens form och därmed interpolera fram resultatet snabbare och med högre precision. För att realisera detta måste interpoleringsmetoden implementeras och presentationskiktet uppdateras, alternativt byggas ut med ett nytt formulär.

5 Referenser

Environmental Systems Research Institute, 2001, Exploring ArcObject volume 1 & 2, New York, California

Environmental Systems Research Institute, <http://www.esri.com> , 2002-12-10

Environmental Systems Research Institute Inc., 2001, Getting Started with ArcGIS, New York, California

Finck R. Robert, 1992. High Resolution Field Gamma Spectrometry. Thesis Lund University, 1992.

Kirtland Mary, 1999, Designing Component-Based Applications, Microsoft Press, Redmond, Washington

Microsoft, <http://www.msdn.microsoft.com> , 2002-12-10

Pattison Ted, 1998, Programming Distributed Applications with COM and Microsoft VisualBasic 6.0, Redmond Washington

6 Ordförklaringar

Ord

Förklaring

ArcView	Ett GIS-verktyg från ESRI.
Beläggningsfält	En utbredning av nukleär, biologisk eller kemisk agens i tid och rum.
C++	Effektivt men komplicerat programmeringsspråk.
Cell	Minsta beståndsdelen i ett raster. Innehåller alltid ett värde och har en fix geografisk placering och utbredning
Diskret	Uppräknelig och kategoriserbar, t.ex frukt är någon av typerna banan, äpple eller päron, motsats till kontinuerlig.
Dokumentobjekt	En objektinstans av det dokument eller karta som det arbetas med för tillfället.
Dosbelastning	Akkumulerad belastning, beräknas olika beroende på agens.
FeatureClass	Vektordata som lagrats av ArcView i en databas.
Fält	Se fältdefinition och beläggningsfält.
Fältdefinition	En eller flera raster där varje raster är fixerat i tidsdimensionen.
GIS	Ett geografiskt informationssystem där användaren kan skapa, bearbeta, analysera och publicera geografisk information.
Gränssnitt	Se interface.
IDW	Inverse distant weighting, en interpoleringsmetod.
Interface	Ett gränssnitt, kan vara för programmering eller användare. Ett programmeringsgränssnitt definierar hur programmeraren skickar och tar emot information. Ett användargränssnitt definierar hur användaren kan kommunicerar med applikationen. Detta kan oftast göras med menyer, fönster, textutor, knappar och liknande.
Interpolering	Metod att skatta en formen på en yta då endast en mängd punkter inom ytan är kända.
Klass	Definitionen på ett objekt, vad det innehåller och vad det kan göra.
Kontinuerlig	Är deriverbar i alla punkter, kan endast vara numerisk, t.ex. utomhus temperaturen är kontinuerlig.
Kriging	En interpoleringsmetod.
Mätstrategi	En mängd med geografiskapunkter som har attribut för patrullnamn och tid. Patrullnamnet anger vilken patrull det är som gör mätningen och tiden anger när mätningen ska ske.
Objektinstans	En klass som skapats och finns i datorns minne i form av ett objekt.
Objektorienterad	En programmeringsteknik där information och funktion kombineras till objekt. Detta underlättar arbetet eftersom ett objekt ofta direkt kan relateras till något verkligt och funktioner grupperas med informationen som den bearbetar.
Polygoner	En månghörning.
Raster	En matris som består av ett antal celler samt har en geografisk position.
Rasterfält	Ett eller flera raster som beskriver ett beläggningsfält där varje raster är fixerat i tiden.
Spline	En interpoleringsmetod.
Spridningsfält	Se beläggningsfält.
Verktygsfält	En samling verktyg som har relaterade funktioner. Verktygsfälten visas genom att först högerklicka på menyraden i ArcView och sedan markera de verktygsfält man vill ska synas.
Visual Basic	Programmeringsspråk och verktyg för att bygga applikationer.
OID	Objekt ID, Ett unikt nummer som identifierar ett geografiskt objekt. Det kan vara en punkt, linje eller polygon.

Bilaga A: Dataflödesdiagram

