

LiU-ITN-TEK-A--08/002--SE

Väderpresentation för flygbranschen

Olof Landahl

2008-01-15



Linköpings universitet
TEKNISKA HÖGSKOLAN

LiU-ITN-TEK-A--08/002--SE

Väderpresentation för flygbranschen

Examensarbete utfört i medieteknik
vid Tekniska Högskolan vid
Linköpings universitet

Olof Landahl

Handledare Daniel Tagesson
Examinator Mikael Jern

Norrköping 2008-01-15

Upphovsrätt

Detta dokument hålls tillgängligt på Internet – eller dess framtida ersättare – under en längre tid från publiceringsdatum under förutsättning att inga extraordinära omständigheter uppstår.

Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för ickekommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns det lösningar av teknisk och administrativ art.

Upphovsmannens ideella rätt innefattar rätt att bli nämnd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart.

För ytterligare information om Linköping University Electronic Press se förlagets hemsida <http://www.ep.liu.se/>

Copyright

The publishers will keep this document online on the Internet - or its possible replacement - for a considerable time from the date of publication barring exceptional circumstances.

The online availability of the document implies a permanent permission for anyone to read, to download, to print out single copies for your own use and to use it unchanged for any non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional on the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility.

According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement.

For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its WWW home page: <http://www.ep.liu.se/>

Sammanfattning

Denna rapport beskriver mitt examensarbete som utfördes på företaget Carmenta AB i Göteborg, hösten 2007. Examensarbetet är den slutgiltiga examinationen för min civilingenjörsutbildning i Medieteknik på Linköpings Universitet.

Arbetet gick ut på att undersöka standarder för presentation av väderprognoser för flygtrafik och att implementera lämplig standard i Carmentas kartverktyg SpatialAce. Jag har framställt en studie som går igenom grundläggande teori för flygtrafik och väder, dataformat för lagring och distribution av meteorologisk information samt standarder för visualisering av flygväder. Utvärderingen av studien visar att en så kallad Significant Weather Chart, eller SIGWX, är den främsta visualiseringsstandarden. Denna standard har jag implementerat med hjälp av SpatialAce och programmeringsspråket C#. Dessutom har jag visualiserat vindprognoser enligt de standarder som finns. Detta gjordes också med SpatialAce men i programmeringsspråket C++. Resultatet är en demoapplikation som presenterar dessa prognoser på en karta. Applikationen tillåter även navigering och annan interaktion med kartan.

Innehållsförteckning

1 Introduktion	1
1.1 Syfte	1
1.2 Bakgrund.....	1
1.3 Mål	1
1.4 Rapporten	2
2 Teori	3
2.1 Flygtrafik.....	3
2.1.1 Före start	3
2.1.2 I luften (en route)	3
2.1.3 Landning	4
2.2 Väder.....	4
2.2.1 Luftens egenskaper	4
2.2.2 Moln	5
2.2.3 Nederbörd	5
2.2.4 Isbildning	6
2.2.5 Åska	6
2.2.6 Luftmassor & fronter	6
2.2.7 Vind & jetströmmar	7
2.2.8 Turbulens & vindskjuvning	7
2.2.9 Cykloner.....	8
3 Meteorologiska data.....	9
3.1 Insamling.....	9
3.1.1 Satellit	9
3.1.2 Väderstation	9
3.1.3 Flygplan	9
3.1.4 Väderballonger.....	10
3.2 Distribution	10
3.2.1 Institutioner & organisationer	10
3.2.2 Datakällor.....	11
3.3 Format	12
3.3.1 Binära format	12
3.3.1.1 BUFR	12
3.3.1.2 GRIB	16
3.3.2 Textformat.....	18
3.3.2.1 SIGMET, AIRMET	18
3.3.2.2 METAR & TAF	19
3.3.2.3 CREX.....	20
3.4 Utvärdering	20
3.4.1 Dataformat	20
3.4.1 Datakälla	21
4 Visualisering	22
4.1 SIGWX	22
4.1.1 Fronter	23

4.1.2 Jetströmmar	23
4.1.3 Turbulens (CAT).....	24
4.1.4 Molnområden	24
4.1.5 Tropopaus	25
4.1.6 Vulkan	25
4.1.7 Strålning	26
4.1.8 Tropiska cykloner	26
4.1.9 Övriga symboler.....	26
4.2 Vindvisualisering	27
4.3 Alternativa visualiseringar	28
4.4 Utvärdering	30
5 SpatialAce	31
5.1 Konfiguration	31
5.2 Arkitektur	33
5.3 CustomDataset	33
6 Implementering	34
6.1 SigwxLib	35
6.2 WindLib	37
6.3 SigwxConfig	38
6.4 SigwxApplication	38
7 Resultat	40
7.1 Visuellt resultat	40
7.1.1 SIGWX	40
7.1.2 Vindprognoser.....	43
7.1.3 Applikation	45
7.2 Struktur och prestanda	46
7.3 Förbättringar	47
7.4 Slutsats	48
8 Referenser	49
Appendix A – Förkortningar	51
Appendix B – Bilder.....	52
Appendix C – Implementering av jetströmmar	59

Figur- och tabellförteckning

Figurer

Figur 3.1. Ett exempel på hur en deskriptor D_0 tolkas beroende på F, X och Y.	14
Figur 4.1. Figuren visar exempel på fronter. Från vänster; kallfront, varmfront och ocklusionsfront.....	23
Figur 4.2. Ett exempel på en jetström som befinner sig på FL320 med en hastighet på 90 knop. Den ökar i styrka till 120 knop och det vertikala djupet, FL290 till FL340 anges. Hastigheten sjunker till 100 knop och där jetströmmen slutar är den 80 knop.....	24
Figur 4.3. Ett CAT-område med referenssiffran 2 som hänvisar till en informationsruta. Den visar att området innebär risk för måttlig turbulens mellan FL250 och FL370.	24
Figur 4.4. Molnområde med ett antal egenskaper. ISOL = Isolated, EMBD = Embedded, CB = Cumulonimbus. Området har sin topp på FL350 och XXX betyder att molnbasen ligger under FL100.	25
Figur 4.5. Figuren visar tre sätt att presentera tropopausen. Den första är standardsymbolen och de två andra används då tropopausen sjunker respektive höjs markant mot omgivningen.	25
Figur 4.6. Informationsruta för vulkanen Etna på Sicilien.	25
Figur 4.7. Symbolen för strålning.	26
Figur 4.8. En cyklon representeras av en symbol och eventuellt dess namn. Cyklonen i figuren befinner sig på norra halvklotet. En cyklonsymbol på södra halvklotet hade varit spegelvänd.....	26
Figur 4.9. En vindsymbol som visar en sydvästlig riktning och en hastighet på 80 knop.	27
Figur 4.10. En vindprognoskarta över norra Europa. Bilden är tagen från [12] som har rättigheterna till bilden.	27
Figur 4.11. Turbulensdata på 36 000 fots höjd visualiseras med en färgskala från grönt till rött. På USA:s östkust finns till exempel ett område med extrem turbulens. Bilden är tagen från [9] som har rättigheterna till bilden.	29
Figur 4.12. En SIGMET-rapport utfärdas över ett område i Norge. Området markeras med rött på kartan. När användaren klickar på, alternativt håller musen över, området dyker meddelandet upp i sitt ursprungliga textformat.	29
Figur 5.1. En pipeline i SpaceLab som bygger upp konfigurationen. En ReadOp läser av en shapefil som renderas av en RenderOp. Lagret presenteras i vyn som ritar ut objekten i ett fönster.	31
Figur 5.2. Attributet "name" sätter texten på textvisualiseraren.....	39
Figur 5.3. En pipeline med Visualizers för linjer och text.	39
Figur 5.4. Figuren visar resultatet av exemplet med bilvägen. Attributen bestämmer text, färg och bredd på vägen.	39
Figur 5.5. SpatialAce-kärnan kommunicerar med applikationen via script-, Java- och COM-gränssnitt.....	33
Figur 6.1. En förenklad bild av min idé för SigwxLib. En BUFR-fil hämtas från SADIS FTP, avkodas med en BufrReader. Ur det tolkade meddelandet skapas väderobjekt som	

omvandlas till GeoObjects. När konfigurationen ska uppdatera vyn skickas en query till datasetet som returnerar alla GeoObjects i den aktuella vyn.....	37
Figur 6.2. Strukturen för min demoapplikation och modul. Modulen består av två dll-filer och en konfigurationsfil som är sammankopplade via varsin Proxy. Applikationen använder dessa tre komponenterna för att visualisera prognoserna.....	39
Figur 7.1. En jämförelse av min visualisering (a) med motsvarande prognoskarta från Met Office (b). Kartorna visar en prognos för 12 december 2007, klockan 18:00. Kartorna har inte samma referenssystem vilket gör att de ser något olika ut. Met Office har rättigheterna till figur 7.1b.	41
Figur 7.2. Informationsrutan pekar utanför molnområdet eftersom mittpunkten räknas fram från områdets maximi- och minimikoordinater.....	48
Figur 7.3. När koordinaterna för jetströmmen är felaktiga kan följande fenomen uppstå. En förklarande linje i (a) visar hur koordinaterna är definierade och (b) visar slutresultatet.	49
Figur 7.4. Den tunna linjen visar hur originallinjen ser ut och prickarna representerar kärnpunkter för jetströmmen. De tjocka linjerna visar hur slutresultatet blir. Eftersom symbolernas placering inte tar hänsyn till splines blir deras inbördes placering fel.	49
Figur 7.5. Den mjuka linjen får plötsligt en skarp krökning på grund av datumlinjen....	43
Figur 7.6. Defekter i molnlinjerna förekommer ofta i kartorna.	43
Figur 7.7. Min vindprognoskarta över Sverige. En större version finns i Appendix B (figur 10.6).	44
Figur 7.8. En vindprognoskarta med temperaturer över Europa från [12] som har rättigheterna till bilden. En större version finns i Appendix B (figur 10.3).....	44
Figur 7.9. Mellan -180° och -120° longitud ritas inte vindsymbolerna ut.....	45
Figur 7.10. Min demoapplikation som visar prognoskartor. En större version finns i Appendix B (figur 10.4).....	45
Figur 10.1. En SIGWX från [12], skapad av SMHI som har rättigheterna till denna bild.	53
Figur 10.2. En SIGWX från Met Office, SADIS FTP som också har rättigheterna till denna bild.....	54
Figur 10.3. En vindprognoskarta från LFV [12] som har rättigheterna till denna bild....	55
Figur 10.4. Min demoapplikation som visar prognoskartor.....	56
Figur 10.5. Figuren förklarar funktionerna i applikationen.	56
Figur 10.5. Figuren förklarar funktionerna i applikationen.	57
Figur 11.1. I (a) presenteras problemet när en linje som korsar datumlinjen istället tar den långa vägen runt jordklotet. För att lösa problemet delas linjen upp i två delar enligt (b).	62
Figur 11.2. I (a) visualiseras jetströmmarna efter de punkter som finns definierade i BUFR-filerna. I (b) används NURB splines för att rita ut strömmarna.....	65
Figur 11.3. Tabellen använder attributet "speed" för att välja rätt hastighetssymbol. Varje RasterSymbol länkar till en bild i formatet PNG.....	66
Figur 11.4. Fem Visualizers används för att visualisera jetströmmar; SpeedSymbolVis, TextVisJetFL, TextVisJetVerticalDepth, SpeedChangeSymbolVis och JetLineVis.	66

Tabeller

Tabell 3.1. Tabellen visar en sammanfattning av ett BUFR meddelande. Alla sektioner är inte kompletta för att göra exemplet mer överskådligt.	15
Tabell 3.2. Tabellen visar en sammanfattning av ett GRIB meddelande. Alla sektioner är inte kompletta för att göra exemplet mer överskådligt.	18
Tabell 4.1. I tabellen presenteras några vanliga vädersymboler i en SIGWX.	27
Tabell 4.2. Symboler för molndistribution i en Station Plot.	28
Tabell 4.3. Prioriteringen av visualiseringar i mitt projekt.	30

1 Introduktion

Rapporten redogör för mitt examensarbete i Medieteknik på Linköpings Universitet. Arbetet utfördes hösten 2007 för Carmenta AB i Göteborg och gick ut på att visualisera väder för flygtrafik. Följande avsnitt förklarar syftet med projektet, dess bakgrund och vilka mål jag ställde upp tillsammans med företaget. En kort beskrivning av rapportens kapitel ges också.

1.1 Syfte

Piloter tar stor hänsyn till vädret i planeringen av en flygtur samt under färden. De har tillgång till avancerade planeringsverktyg och här kommer presentationen av väder in i bilden. Under flygfärden är det en stor fördel att kunna se väderprognoser för sträckan. Syftet med arbetet var att ta fram en standardiserad visualisering av väderprognoser för flygtrafik en route. Resultatet ska sedan användas för att förkorta ledtiden för framtida projekt på företaget.

1.2 Bakgrund

Carmenta AB är ett IT-företag med ca 50 anställda i Göteborg och Stockholm. De specialiserar sig främst på applikationer innehållande digitala kartor. De har egna produkter för utveckling av kartor och GIS-system men de tar även fram kundspecifika lösningar och erbjuder konsulttjänster inom GIS, som står för Geographic Information System. Carmenta har många kunder inom flyg- och försvarsbranschen. Visualisering av flygväder är därför ett intressant område för företaget att titta närmare på vilket resulterade i ett examensarbete.

1.3 Mål

Arbetet var relativt fritt och utvecklades under arbetets gång. Ett tidigt satt mål var dock att utföra en grundlig studie i standarder för presentation av flygväder samt att implementera en lämplig standard i företagets utvecklingsverktyg SpatialAce. Dessutom skulle en demoapplikation som presenterar resultatet konstrueras.

Efter att studien var genomförd och visualiseringsstandarder hade utvärderats specificerades målet mer tydligt. Det slutgiltiga målet kan sammanfattas med följande punkter:

- Framställ en omfattande studie där standarder för lagring, distribution och visualisering av flygväder undersöks och utvärderas.
- Implementera den främsta visualiseringsstandarden (som i studien visade sig vara Significant Weather Chart) med kartverktyget SpatialAce och programmeringsspråket C#. Implementeringen ska fungera som en modul som kan användas för att visa aktuella prognoskartor i en applikation.
- Bygg en applikation som använder ovan nämnda modul. Applikationen ska tillåta navigering och andra relevanta funktioner, till exempel lagerhantering och val av prognos.
- I mån av tid, implementera även andra visualiseringsstandarder med SpatialAce.

1.4 Rapporten

Studien innefattar kapitel 2 – 4. Kapitel 2 går igenom grundläggande teori inom flyg och väder. Kapitel 3 undersöker vilka dataformat som används för lagring och distribution av väderprognoser. Standarder för visualisering av flygväder beskrivs i kapitel 4. Kapitel 5 förklarar SpatialAce som används i implementeringen av min visualisering och applikation. Implementeringen tas upp i kapitel 6 där metod, problem och lösningar presenteras. I kapitel 7 redovisas resultatet med förslag på förbättringar. Referenser presenteras i kapitel 8 och sedan följer ett antal bilagor. I appendix A förklaras förkortningar som förekommer i rapporten, appendix B innehåller bilder och appendix C tar upp ett detaljerat exempel av implementeringen.

2 Teori

För att visualisera väder måste man ha en grundläggande förståelse för hur olika meteorologiska fenomen uppkommer och utvecklas. För att visualisera väder specifikt för flygtrafik krävs dessutom att man har vissa kunskaper om pilotens situation före och under flygningen. Därför börjar jag med att redogöra för grundläggande teori inom flygtrafik och väder.

2.1 Flygtrafik

Varje dag lyfter över 50 000 kommersiella passagerarplan runt om i världen. Dessutom tillkommer privata och militära flygningar. För att bibehålla säkerheten med så många plan i luften, och på flygplatserna, finns det många regler och standarder som man måste följa.

Piloten och annan personal måste planera varje flygtur ordentligt och se till att start, färd och landning blir säker för passagerare på planet och för annan flygtrafik som befinner sig i närheten.

Här ger jag en mycket kort och icke-komplett genomgång av hur en flygtur med passagerare kan planeras och utföras. Fokus ligger givetvis på hur flygväder hanteras men en del andra viktiga moment tas också upp. För en mer utförlig beskrivning rekommenderas [1] eller annan litteratur om flygtrafik.

2.1.1 Före start

Piloterna analyserar väderprognoser och -rapporter, till exempel METAR (Meteorological Aerodrome Report), TAF (Terminal Aerodrome Forecast) och SIGWX (Significant Weather Chart), för sträckan, destinationen och alternativa flygplatser. Mer information om dessa prognoser och rapporter finns i avsnitt 3.3.2 och 4.1. Alternativa flygplatser måste vara bestämda i förväg om något skulle hända i luften och planet tvingas nödlanda. Med hjälp av datorn tar man fram en färdplan som beräknar hur mycket bränsle som krävs för resan. Trafikledningscentralen får färdplanen och ger färdtillstånd. En flygnivå, eller flight level (FL), tilldelas flygplanet för att förhindra kollisioner och turbulens från andra plan i luften. FL anges i hundra fot och talar om vilken höjd flygplanet ska hålla sig på under färden. När passagerarna har gått ombord och piloterna har gått igenom sina checklistor godkänner kaptenen att flygplanet är i flygbart skick och att väderförhållandena tillåter flygningen.

2.1.2 I luften (en route)

När flygplanet väl är i luften och har hamnat på sin FL säger man att planet är en route. Flygplanet manövreras automatiskt enligt färdplanen och piloterna får kontinuerligt

information om höjd, hastighet, bränslemängd, med mera. Piloten kan när som helst avbryta autopiloten och styra planet manuellt.

Flygplanet har en rad instrument som kan läsa in meteorologisk information som till exempel temperatur, luftfuktighet, vindhastighet och lufttryck. Radar kan användas för att upptäcka till exempel åskväder. Vattendroppar reflekterar radarpulsen och ger utslag på skärmen i cockpit. Utslaget växer med nederbördens storlek och hagel ger störst utslag. Under flygturen får piloten uppdaterade rapporter och prognoser för väder och annat som kan påverka färden.

2.1.3 Landning

För varje flygplats finns en manual för hur flygplatsen ser ut och hur inflygning och landning ska utföras. Dessutom kan flygbolagen ha olika bestämmelser och krav för att en inflygning ska få ske. Ett viktigt krav som måste uppfyllas är väderminima. Det är bestämt i förväg och är det mått på hur dåligt vädret (framför allt sikten) får vara innan en inflygning måste avbrytas. Om vädret är för dåligt för att landa kan man antingen avvakta genom att cirkulera runt ett varv och göra ett nytt försök eller bege sig till en av de förbestämda alternativa flygplatserna, där vädret tillåter inflygning och landning.

2.2 Väder

För flygtrafiken är väderobservationer och prognoser av största vikt. Den dagliga flygverksamheten styrs av vädret och flygbranschen har i många år varit den största enskilde användaren av meteorologisk information. För att förstå vädrets inverkan på flyg krävs en storskalig förståelse för hur vädret fungerar, från marken och havet, en bra bit upp i atmosfären.

Jag börjar med att förklara några viktiga egenskaper hos luften för att sedan kort redogöra för meteorologiska fenomen och hur de påverkar ett flygplan i luften.

2.2.1 Luftens egenskaper

Dagpunkten är ett begrepp som har stor inverkan på hur moln bildas. Det är benämningen på den lufttemperatur där vattnet i luften kondenserar. Därför kallas den höjd där lufttemperaturen når dagpunkten för *kondensationsnivån*.

Atmosfären är indelad i olika lager. *Troposfären* sträcker sig från marken upp till *tropopausen* där stratosfären tar vid. Dess höjd varierar med årstid och lufttryck men ligger på ca 8 km höjd vid polerna och 16 km vid ekvatorn. De flesta väderfenomen utspelar sig under tropopausen och åskmoln byggs ofta upp till denna höjd. Därför blir åskväder i regel mycket kraftigare nära ekvatorn än vid polerna.

Lufttryck mäts i hPa och varierar med höjden. Vid havsytan är trycket mellan 930 och 1050 hPa och det avtar med ungefär 100 hPa/m vilket gör att ett flygplans hastighet och

höjd över havet kan räknas ut från aktuellt tryck. Varm luft har lägre tryck än kall vilket ger fel mätvärden. Vid varm luft och högt tryck kommer höjd- och hastighetsmätarna visa lägre än verkligheten.

Lufttemperaturen beror givetvis på klimat, årstid och aktuellt väder men i regel sjunker den med höjden. Minskat lufttryck gör att luften expanderar. Energin som går åt för att expandera luften leder till lägre temperatur, ca 1°C för var hundra meter. Hos fuktigare luft sjunker temperaturen ungefär hälften så fort.

2.2.2 Moln

Moln består av mycket små vattendroppar, alternativt iskristaller på höga höjder. Oftast bildas moln genom att varm luft stiger. När den varma luften stiger, expanderar den och temperaturen sjunker. När luften har stigit så högt att temperaturen når daggpunkten (vid kondensationsnivån) kondenserar vattnet i luften och moln bildas.

Stigande varm luft kan orsakas av att solen värmer upp marken. Marken avger värme till den närliggande luften som då stiger. Det kallas ytuppvärmning. Vertikala barriärer, som kullar och berg, samt fronter, konvergens och turbulens kan också orsaka stigande varm luft och molnbildning. Förekomst av salter, rök och smuts i luften ökar luftens förmåga att kondensera.

Det finns många typer av moln, med olika höjder, egenskaper och former. De som är av intresse i en väderprognos för flyg är framför allt så kallade Cumulus- och Cumulonimbusmoln (CB-moln). Dessa moln bildas ofta genom ytuppvärmning och de kan sträcka sig högt upp, ända upp till tropopausen. De bildas snabbt och innebär ofta ostadigt väder i form av hårda vindar, turbulens och nederbörd.

Förutom det ostadiga vädret som moln kan föra med sig innebär de försämrad sikt för piloten. Täta moln kan även störa radar och radiosignaler som används vid navigering.

2.2.3 Nederbörd

Regn, snö och hagel bildas från iskristaller i CB-moln. Iskristallerna kolliderar med varandra och bildar snöflingor, som av sin tyngd faller neråt. Om temperaturen är under 0°C förblir nederbörden i form av snöflingor, annars faller regndroppar.

Hagel uppstår ofta i samband med åska. Det sker genom att iskristaller åker upp och ner, på grund av vertikala vindar i molnet, och samlar på sig vatten som fryser. När de blir tillräckligt stora och tunga faller de ner till marken som hagel.

Regn och snö ger sämre sikt och kan störa radiosignaler men innebär annars ingen större fara för flygplan i luften. Däremot kan hagel ge allvarliga konsekvenser. Om stora mängder hagel åker in i jetmotorerna kan dessa skadas och i värsta fall leda till haveri.

2.2.4 Isbildning

När temperaturen ligger runt 0°C finns det risk för isbildning, särskilt i moln där luftfuktigheten är hög. Detta är en av de största farorna för ett flygplan. Is som bildas på planets noga utformade vingar och stjärtparti kan påverka flygegenskaperna för planet och försämra vingarnas lyftkraft. Dessutom kan is lossna från planet och flyga in i jetmotorerna.

2.2.5 Åska

Ett CB-moln kan utvecklas till åskmoln som i regel sträcker sig upp till tropopausen. I tropiska regioner, där tropopausen ligger betydligt högre än vid polerna, blir åskmolnen som störst. Förutom kraftig nederbörd så uppstår elektriska laddningar i molnet som ger upphov till åska.

Hagel, som kan bildas i åskmoln, faller neråt och kolliderar med stigande iskristaller. Vid dessa kollisioner kan elektroner hoppa över till det nedfallande haglet. Detta betyder att de stigande iskristaller, som har blivit av med en elektron, är positivt laddade och de fallande hagelpartiklarna, som har fått en extra elektron, är negativt laddade. Det bildas en attraktion mellan molnets övre och undre skikt och när den attraktionen blir tillräckligt stark sker en urladdning. De flesta blixtar sker inom molnet men ibland kan de slå ner i marken.

Åskmoln skapar instabilitet och andra väderfenomen kan uppstå ur detta, till exempel turbulens och hagel. Dessa fenomen kan innebära stor fara för flygplan (se avsnitt 2.2.3 och 2.2.8).

Atmosfäriska störningar och elektriska urladdningar som sker vid åska kan störa radiosignaler. Blixtar som slår ner i planet påverkar dock inte flygningen.

2.2.6 Luftmassor & fronter

En luftmassa är en stor samling luft med liknande temperatur och luftfuktighet. Det tar lång tid för en luftmassa att ändra temperatur. Därför är det främst i områden där luften håller sig relativt stilla under en längre tid som kalla och varma luftmassor bildas.

I tropiska områden bildas varma luftmassor och vid polerna kalla luftmassor. Om luften färdas över land blir den ofta torr. Färdas den över vatten blir den istället fuktig. Dessa luftmassor, med kall eller varm luft, som är torr eller fuktig, påverkar mycket av vädret på vår planet.

Då två luftmassor möts bildas en front. Om en kall luftmassa tränger undan en varm är det en kallfront och vice versa. Kall luft är tyngre och därför pressas alltid den varma luftmassan ovanför, vilket leder till molnbildning och nederbörd.

Normalt ger en kallfront, vid marken, lägre temperatur, lägre daggpunkt och förändrad vindriktning. Tropopausen sänks. En varmfront leder till ökad temperatur och daggpunkt samt förändrad vind. Tropopausen höjs.

Kallfronter färdas snabbare vilket gör att de ofta hinner ifatt en varmfront och det bildas en okklusionsfront där de möts. När den kalla luften hinner ifatt stiger den varma luften i en, på norra halvklotet, motsols rotation (se avsnitt 2.2.7). Det leder till att varm luft ligger mellan två kalla luftmassor. Det har bildats ett lågtryck som ger instabilt väder.

Flygplan påverkas inte direkt av frontsystem, men däremot indirekt eftersom de ger upphov till molnbildning och nederbörd samt vind- och temperaturförändringar.

2.2.7 Vind & jetströmmar

Vindhastighet kan påverkas av många faktorer. Angränsande luftmassor med olika temperatur har olika lufttryck som vill jämnas ut. Fronter innebär därför ofta starka vindar.

Jordens rotation påverkar också vindens rörelser enligt Coriolis kraft¹. Det är denna kraft som gör att luften roterar motsols på norra halvklotet, och medsols på södra halvklotet, runt ett lågtryck. Denna rotation och även vindhastigheten ökar ju högre upp i troposfären man kommer.

En extrem typ av vind, som är en kraftig smal luftström, kallas jetström. De finns strax under tropopausen och ofta i frontband, mellan kall och varm luft. För att kallas jetström måste vindhastigheten vara över 90 km/h men ibland når de upp till över 200 km/h. Hastigheten beror på temperaturgradienten, det vill säga temperaturförändringen mellan luftmassorna.

Även jetströmmar påverkas av Coriolis kraft. Kraften har ingen inverkan runt ekvatorn så det finns inga jetströmmar förrän några grader latitud norr- eller söderut från ekvatorn. De kraftigaste jetströmmarna går från väst till öst och är i regel formade som en slingrande flod.

Vind påverkar flygplan på många sätt. Start- och landningssträckan kan till exempel bli längre eller kortare. När man lägger upp en planerad flygväg måste man ta hänsyn till vindar. Dels kan resan ta längre tid och dra mer bränsle, men man får också ändra flygriktning för att motverka sidovindar.

Man kan självklart utnyttja vindarna till sin fördel, främst jetströmmar. Genom att färdas i jetströmmen, längs med dess riktning, kan man spara mycket bränsle och flygtid.

2.2.8 Turbulens & vindskjuvning

Ett meteorologiskt fenomen som är vanligt att man stöter på under en flygtur är turbulens. Turbulens är vindvirvlar där vindhastigheten ofta är stor och riktningen varierar åt olika håll. Fenomenet uppstår ofta i moln, speciellt åskmoln, men även i klar luft, även kallat CAT (Clear-Air Turbulence). I klar luft skapas turbulens när kall och varm luft blandas. Därför är detta vanligt i områden runt jetströmmar. Det kan även bildas turbulens från ett

¹ Coriolis kraft: Den kraft som påverkar ett objekt som förflyttar sig på ett annat roterande objekt, givet att observatören roterar med det roterande objektet. I detta fall är det luften som förflyttar sig och jorden som är det roterande objektet.

annat flygplan. Därför är det viktigt att man planerar flygfärden så att man inte hamnar i turbulens från ett annat plan.

CAT går varken att upptäcka med blotta ögat eller med radar. Därför dyker det ofta upp helt oväntat. Turbulens påverkar främst ett plan vertikalt och är oftast bara ett litet obehag för flygpassagerarna. Men under extrem turbulens kan passagerare skadas, och dödsfall har skett, när människor kastats runt i planet.

Någonting som kan uppfattas som turbulens är vindskjuvning ("wind shear"). Vindskjuvning ändrar snabbt hastighet och riktning och kan uppstå på många sätt men är vanligast vid frontsystem, lågtryck och höga berg. De bildas ofta i så kallade konvergensmoln vid oväder och kan ibland gå vertikalt, rakt ner i marken. Framför allt dessa vertikala vindskjuvningar, så kallade fallvindar, kan vara förödande vid start och landning. Därför undviker man alltid konvergensmoln vid start och landning.

2.2.9 Cykloner

Det finns olika benämningar på extrema oväder med starka vindar, till exempel cyklon, tyfon och tornado. Alla beskriver samma väderfenomen men vilket namn man använder beror på var i världen ovädet uppstår. I rapporten kommer endast cyklon att användas som benämning även om den uppstår i områden där den skulle kallas tyfon eller tornado. De kraftigaste oväder som finns på vår jord är tropiska cykloner. De ger det mest extrema vädret och gör ofta stor skada där den sveper fram.

Tropiska cykloner bildas runt lågtryck över tropiska hav med hög vattentemperatur. Skillnaden från ett vanligt lågtryck är att värmen från havet ger ofantligt mycket energi till det kraftiga ovädet som ofta har en stor utsträckning. Cyklonens rotation är ett resultat av Coriolis kraft och en bit ut från cyklonens centrum, där moln börjar bildas, kan vindhastigheter på över 60 m/s uppmätas.

Cykloner är vanligast i Stilla Havet, norr om ekvatorn.

Man undviker alltid att flyga genom dessa områden. Det är inte ovanligt att många flyg ställs in när en tropisk cyklon drar förbi.

3 Meteorologiska data

Idag samlar man in stora mängder meteorologiska data. Eftersom vädret har stor inverkan på mycket i vårt samhälle är det viktigt att informationen är noggrann och uppdateras ofta för att man ska kunna göra korrekta prognoser. Mätningar görs över hela världen; på land, ute på havet och uppe i luften. En rad olika meteorologiska institutioner sköter distributionen av data och formaten varierar.

Här redogör jag kort för insamling och distribution av data för att sedan fördjupa mig i olika format för flygväderrapporter och -prognoser.

3.1 Insamling

Insamlingen av väderdata görs på många sätt. För att kunna göra noggranna prognoser för flygtrafik gör man mätningar både nere på marken och uppe i luften, från lägre höjder, ända upp i rymden.

För att kort redogöra för insamling av meteorologiska data tar jag upp de vanligaste sätten att samla in data, specifikt för flygtrafik. Observera att det finns betydligt fler metoder än de som tas upp här.

3.1.1 Satellit

Satelliter fångar in data via ett antal instrument, som radiometer, spektrometer, interferometer och radar. Data från spektrometern och interferometern används för automatiska numeriska beräkningar, som görs av mycket kraftfulla datorer, och kallas NWP (Numerical Weather Prediction). Resultatet blir omfattande prognoser och information om temperatur, luftfuktighet, tryck och vind.

Infraröda satellitbilder mäter temperatur, antingen för ett moln eller för jordytan.

3.1.2 Väderstation

Från automatiska och bemannade väderstationer samlar man in data från marken. Det görs bland annat med hjälp av radar som kan mäta nederbörd och vertikala vindar i atmosfären samt andra instrument som mäter temperatur, luftfuktighet, tryck och vindar. De flesta flygplatser gör väderobservationer och prognoser för området på och runt flygplatsen.

3.1.3 Flygplan

Det är viktigt att komplettera information från markstationer och satelliter med mätningar som är gjorda i flyghöjd. Detta görs enklast från flygplan i luften. Via AMDAR (Aircraft Meteorological Data Relay) samlar flygplan in väderdata och skickar ner till marken via radio eller satellit. Tanken med AMDAR är att flygplan ska samla in så pass mycket

meteorologisk information att man kontinuerligt kan ta fram väderprognoser med hjälp av NWP (se avsnitt 3.1.1) och därmed ersätta dagens väderrapportering AIREP (Aircraft Report).

PIREP (Pilot Report) är rapporter från piloter. Piloter som åker genom områden med potentiellt farliga väderförhållanden, till exempel isbildning, turbulens och dålig sikt, rapporterar dessa observationer ner till marken. Informationen bekräftar ofta redan tillgängliga prognoser men ibland kan det vara värdefull information om nyuppkomna väderfenomen i området. Det är därför viktigt att piloter alltid varnar för observerade faror.

3.1.4 Väderballonger

Väderballonger har använts länge för att samla in information om temperatur, tryck och luftfuktighet i atmosfären. Detta är, precis som flygplan, ett viktigt komplement till markstationer och satelliter. Över hela världen skickas väderballonger upp dagligen för att utföra mätningar som skickas ner till marken.

3.2 Distribution

Eftersom vädret inte bryr sig om landsgränser är meteorologiska data och prognoser något som rör samtliga länder på jorden. Därför finns det institutioner och organisationer över hela världen som samarbetar för att samla in och distribuera data.

För att klargöra vilka institutioner och organisationer som sätter standarder och sköter distributionen av väderdata för flygtrafik följer en kort genomgång av de viktigaste organen.

3.2.1 Institutioner & organisationer

WMO (World Meteorological Organization) är en del av FN och arbetar för ett brett samarbete mellan världens länder när det gäller klimatfrågor, naturkatastrofer samt standardisering och distribution av väderdata. De har, tillsammans med ICAO (International Civil Aviation Organization), tagit fram standardformat för flygspecifika väderdata.

ICAO, som också är en del av FN, har till uppgift att arbeta för säkerhet och effektivitet i flygbranschen. De arbetar fram rapporter och standarder som ska vara oberoende av enskilda länders intressen. Tillsammans med WMO tar de fram standarder för flygväder.

Met Office (Meteorological Office) är Storbritanniens nationella meteorologiska institut, men även ett av världens två WAFC (World Area Forecast Centre), utsedda av ICAO. WAFC:s har till uppgift att ta fram och distribuera internationella väderprognoser för flygtrafik, enligt ICAO:s konventioner. De tar in meteorologiska data från hela världen

och via satellitsystemet SADIS (Satellite Distribution) distribuerar de data och prognoser till användare inom flygbranschen.

NWS (National Weather Service) är USA:s meteorologiska institut och världens andra WAFC. Med hjälp av ISCS (International Satellite Communications System) kompletterar de SADIS i distributionen av data och prognoser för flygtrafik världen över.

EUMETSAT (European Organisation for the Exploitation of Meteorological Satellites) är ett europeiskt samarbete för insamling av väderdata från satelliter. Organisationen är ansvarig för planering, uppskjutning och underhåll av vädersatelliter.

De distribuerar vidare informationen till nationella meteorologiska institut, framför allt i Europa, men även övriga världen. Medlemsländernas respektive meteorologiska institut äger licensen för en del av satellitdata i det landet. SMHI och Met Office är Sveriges respektive Storbritanniens licensinnehavare av EUMETSAT:s data.

3.2.2 Datakällor

Väderdata kan bland annat fås från Met Office, EUMETSAT och NWS. De erbjuder tjänster, gratis eller licenserade, för den som vill ha tillgång till meteorologiska data och prognoser.

Met Office tillhandahåller dels en trådlös tjänst som heter SADIS 2G och en Internet-baserad FTP-tjänst SADIS FTP [10]. Här kan man få tillgång till de vanligaste dataformaten och rapporterna, som BUFR (Binary Universal Form for Representing data), GRIB (Gridded Binary data format), AIRMET (Airmens Meteorological Information), METAR och TAF. Dessa format och rapporter förklaras i avsnitt 3.3. Det är via SADIS FTP som Carmenta får sina väderdata från och det är framför allt dessa data jag har haft tillgång till under studien. Därför kommer jag att applicera fotnoter där det finns möjlighet att närmare specificera vad som finns på SADIS FTP.

De flesta prognoser uppdateras var sjätte timma, och gäller för tiderna 00:00, 06:00, 12:00 och 18:00. Andra prognoser och rapporter uppdateras varje timma eller sporadiskt, när det sker förändringar.

EUMETSAT har också trådlösa tjänster och även en FTP-tjänst [11]. De tillhandahåller satellitdata där en del är gratis och andra kräver licenser. De väderdata som erbjuds är inte specifik för flygtrafik men format som BUFR och GRIB (se avsnitt 3.3) finns representerade. Dessa format anses av WMO vara "essential data" och därför är de gratis.

ADDS (Aviation Digital Data Service), som drivs av NWS, har en uppsjö av gratis vädertjänster för flygtrafik (främst för USA). Gratis satellit- och radarbilder, textbaserade väderrapporter som PIREP, METAR och SIGMET (se avsnitt 3.3) samt grafiska prognoser för turbulens, vindar, isbildning, med mera finns att tillgå på ADDS:s hemsida [9].

LFV (Luftfartsverket) lägger ut METAR, TAF, NOTAM och SIGWX (se avsnitt 3.3) på sin hemsida [12].

SMHI tillhandahåller också väderdata efter önskemål men den är inte specifik för flygtrafik.

3.3 Format

Det finns många sätt att representera väderprognoser. En uppsjö av format för meteorologiska rapporter används inom flygbranschen och även om många överlappar varandra så har de alla sina syften, kriterier och begränsningar. Vilka format man väljer beror till exempel på regler, flygplanets beskaffenhet och pilotens erfarenhet.

Vissa format är gjorda för att läsas direkt av piloten före eller under en flygning och andra är gjorda för att lagra och distribuera stora mängder meteorologiska data. Den gemensamma nämnaren är att de innehåller väderrapporter som i första hand är ämnat flygtrafik.

Här följer en relativt grundlig genomgång av de vanligaste formaten som används.

3.3.1 Binära format

För att kunna lagra, och framför allt distribuera, meteorologiska data har WMO tagit fram två format som är noga utformade för att vara flexibla, konsekventa och kompakta. Båda är binära format och kräver en avkodare som tolkar den binära koden och antingen skriver ut informationen som läsbar text eller ritar ut informationen på en karta. Formaterna kallas BUFR och GRIB och är flitigt använda för visualisering av flygväder.

3.3.1.1 BUFR

BUFR står för ”Binary Universal Form for Representing data” och är ett standardformat för att distribuera och lagra meteorologiska data. Det används bland annat för att lagra data för skapandet av SIGWX, en väderprognoskarta för flygtrafik (se avsnitt 4.1).

BUFR är ett tabellbaserat format som inte nödvändigtvis måste användas för meteorologisk information. Dess flexibilitet tillåter kodning av andra sorters data. Detta kräver dock tillgång till, för ändamålet, rätt tabeller. Tabellerna finns inte inkluderade i BUFR-meddelandet utan de måste skickas separat.

WMO har planer på att göra GRIB2 (se avsnitt 3.3.1.2) till nytt standardformat för skapandet av prognoskartor.

Ett BUFR-meddelande är indelat i sex sektioner som består av oktetter. En oktett är en sekvens på åtta bitar och kan representera text, numeriska värden eller referens till tabell. Längder på sektioner anges i antal oktetter.

De sex sektionerna i ett BUFR-meddelande förklaras nedan och en sammanfattning presenteras i tabell 3.1.

Sektion 0: Indikator

Sektionen innehåller texten "BUFR" (ASCII tecken), för att visa att meddelandet är kodat i BUFR. Sen följer hela meddelandets längd (i antal oktetter) och vilken BUFR-version det är. Den senaste är version 3².

Sektion 1: Identifikation

Börjar med sektionens längd. Sedan följer bland annat information om meddelandets källa, datum och tidpunkt för dess skapande.

Sektion 2: Valfri

Här kan valfri information inkluderas om det behövs. Denna sektion används mycket sällan.

Sektion 3: Databeskrivning

Sektionens längd följs bland annat av två flaggor (bitar) som indikerar om det är observerade data eller inte samt om det är komprimerat eller inte. Icke-observerade data kan till exempel vara en prognos.

Sedan följer själva kärnan i BUFR-formatet; deskriptorerna.

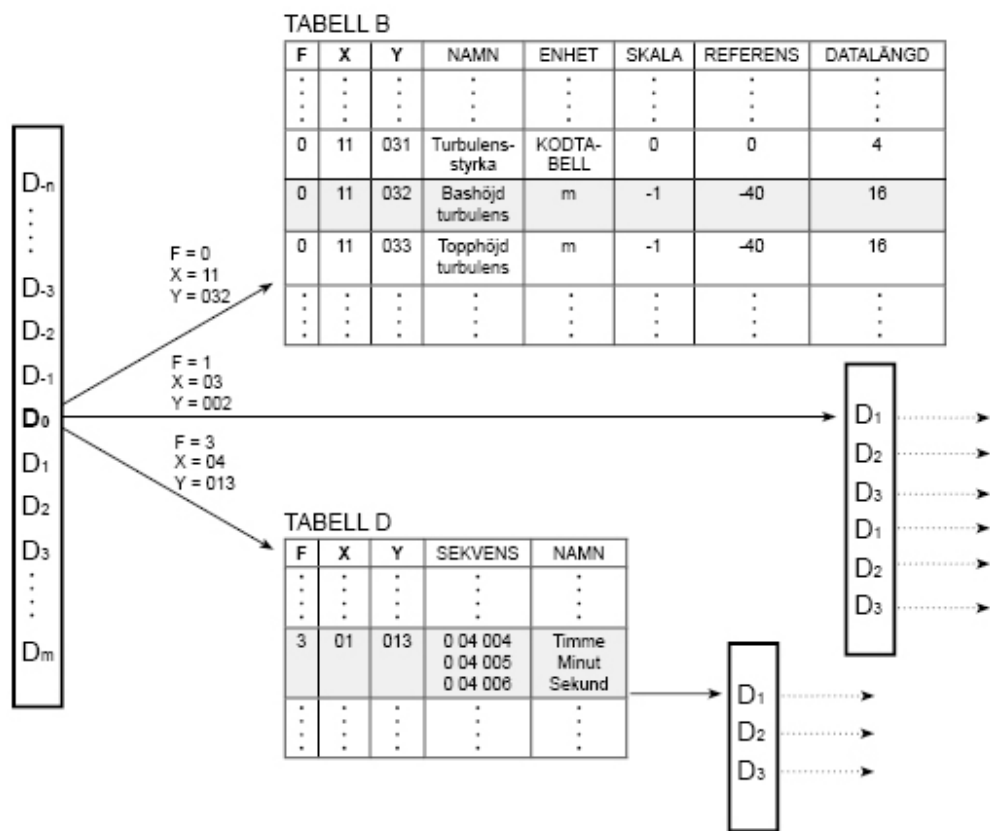
En deskriptor består av två oktetter och man brukar dela upp de i F (2 bitar), X (6 bitar) och Y (8 bitar). F och X utgör alltså den första oktetten, och Y utgör den andra, i en deskriptor. Deskriptorn, i sin grundläggande form (då F = 0), talar via en tabelluppslagning om vilken typ av data den beskriver och hur det ska tolkas. Tabellen kallas tabell B och där ska alla databeskrivningar, som kan tänkas komma till användning, finnas representerade.

Beroende på värdet i F kan deskriptorn även dupliceras och bilda en sekvens med nya deskriptorer. F kan anta värdena 0-3, X 0-63 och Y 0-255. En grafisk förklaring finns i figur 3.1.

- F = 0 : Här används deskriptorn i sin grundläggande form, som en referens till en extern tabell B där X anger vilken klass deskriptorn tillhör och Y anger en post i den klassen. Från den refererade posten får man ut en beskrivning av data, närmare bestämt dess namn, enhet, skala, referensvärde och längd. En förklaring av dessa begrepp finns senare i detta avsnitt.
- F = 1 : I detta fall används deskriptorn som en kopian där X anger hur många av de föregående deskriptorerna som ska ingå i den kopierade sekvensen och Y anger hur många gånger den ska repeteras.
- F = 2 : Deskriptorn fungerar då som en operator men detta används sällan.
- F = 3 : Deskriptorn länkar till en extern tabell D där X anger kategori och Y vilken post som ska användas. Tabell D används för att lagra vanliga

² Version 3 används av Met Office (SADIS).

sekvenser av deskriptorer. Till exempel kan en post i tabell D leda till en sekvens av deskriptorer som anger tid; det vill säga år, månad, dag, timma, minut (en sekvens med fem deskriptorer).



Figur 3.1. Ett exempel på hur en deskriptor D₀ tolkas beroende på F, X och Y.

Sektion 4: Data

Efter den angivna sektionslängden finns data lagrad. Det är dessa data som beskrivs av deskriptorerna i sektion 3.

Sektion 5: Slutmarkering

Här markeras slutet på meddelandet med “7777” (ASCII tecken).

Sektion 0 (Indikator)	"BUFR" Meddelandets längd Version
Sektion 1 (Identifikation)	Sektionens längd BUFR huvudtabell Källa (Vilka har skapat meddelandet) . . .

	År (Tidpunkt för meddelandets skapande) Månad Dag Timme Minut
Sektion 2 (Valfri)	(Används ej)
Sektion 3 (Data beskrivning)	Sektionens längd Antal data subset Flaggor (indikerar om det är observerad data och om den är komprimerad) Deskriptorer . . .
Sektion 4 (Data)	Sektionens längd Data . . .
Sektion 5 (Slut)	"7777"

Tabell 3.1. Tabellen visar en sammanfattning av ett BUFR meddelande. Alla sektioner är inte kompletta för att göra exemplet mer överskådligt.

Kolumnerna i tabell B anger som sagt namn, enhet, skala, referensvärde och datalängd. Här följer en förklaring för dessa begrepp och hur de hänger ihop med motsvarande data.

Namnet anger helt enkelt vad meddelandets data beskriver, till exempel *longitud*, *hastighet*, eller ett *specifikt väderfenomen*.

Enheten anger vilken SI-enhet som används, till exempel *K* (*kelvin*) eller *m/s*. I fallet där deskriptorn beskriver ett väderfenomen refererar enhetsvärdet istället till en kodtabell där datavärdet refererar till en post i kodtabellen. Posten kan till exempel innehålla "*Warm front at the surface*" eller "*Clear-air turbulence*".

Attributet skala anger precisionen på datavärdena. Om precisionen är 2 har datavärdet vid kodningen skalats med 10^2 vilket man måste skala ner igen för att få ut rätt värde (se ekvation 3.2).

Referensvärdet anger det minsta möjliga värdet för den aktuella tabellposten och ser till att det kodade värdet är garanterat positivt. Detta värde måste också användas vid avkodning (se ekvation 3.2).

Datalängd anger hur många bitar datasekvensen för aktuell deskriptor tar upp. Detta krävs för att rätt data ska lokaliseras i sektion 4.

Det kodade datavärdet som är lagrat i sektion 4 har tagits fram med ekvation 3.1.

$$\text{kodat_värde} = \text{korrekt_värde} * 10^{\text{skala}} - \text{referensvärde} \quad \text{Ekvation 3.1}$$

Det betyder att den måste avkodas enligt ekvation 3.2.

$$\text{korrekt_värde} = (\text{kodat_värde} + \text{referensvärde}) * 10^{-\text{skala}} \quad \text{Ekvation 3.2}$$

För ytterligare information om BUFR hänvisar jag till [4] och [5].

3.3.1.2 GRIB

GRIB (GRIdded Binary data format) är ännu ett standardformat för meteorologisk och hydrologisk data. Det är binärt, precis som BUFR, men formatet är inriktat på numeriska värden representerat i ett rutnät (en så kallad grid).

GRIB används dels inom flygväder för att distribuera och lagra information om temperatur, lufttryck, vindhastighet och luftfuktighet³ på olika höjder i atmosfären men även för båttrafik där man istället lagrar information om havets beskaffenheter.

I dagens läge används i regel GRIB1⁴ (GRIB version 1) men WMO har för avsikt att göra GRIB2 (version 2) till standardformat, både för vinddata och för prognoskartor. GRIB2 har en kraftigare komprimering av filer och ger därför snabbare överföringshastigheter. Dessutom är det mer flexibelt och tillåter därför möjligheter för nya användningsområden. I denna utökade version står GRIB för "General Regularly-distributed Information in Binary form".

Redogörelsen för GRIB som följer nedan gäller version 1. Utförligare information om GRIB1 och GRIB2 finns i [5] och [6].

Ett GRIB-meddelande är indelat i sex sektioner och börjar ungefär likadant som ett BUFR-meddelande. Här följer en genomgång av de olika sektionerna och en sammanfattning i tabell 3.2.

Sektion 0: Indikator

Sektionen börjar med "GRIB" (ASCII), sen hela meddelandets längd i antal oktetter och till sist vilken version som används.

Sektion 1: Definitioner

Börjar med sektionens längd och sen följer information om källa, grid identifiering, tidpunkt för meddelandets skapande, och mycket annat.

Sektion 2: Gridbeskrivning (valfri)

Här kan man, om man inte vill välja en fördefinierad grid (vilket görs i sektion 1), definiera en egen grid. I så fall anger man som vanligt sektionens längd, och sedan antal vertikala koordinatparametrar, plats, datarepresentation och till sist gridbeskrivningen.

³ Met Office (SADIS) testar för närvarande (december 2007) att representera isbildning, turbulens och CB-moln med GRIB på sin FTP [8].

⁴ Met Office (SADIS) använder GRIB1.

Sektion 3: Bitmap (valfri)

En bitmap kan användas för att maska bort irrelevanta områden på griden, där det till exempel saknas värden.

Sektion 4: Data

Här anges de numeriska datavärdena i griden där var och en är uppdelad i fyra variabler; X (kodat värde), R (referensvärde), E (skalfaktor) och D (decimalskalningsfaktor). Om Y är det korrekta datavärdet så kan ekvationen för avkodningen av data skrivas som ekvation 3.3.

$$Y * 10^D = R + (X * 2^E) \quad \text{Ekvation 3.3}$$

Referensvärdet R är det minsta möjliga värdet för Y vilket garanterar att det kodade värdet inte blir negativt. Skalfaktorn E anger antal bitar, och D justerar precisionen⁵, för värdet.

Sektion 5: Slutmarkering

Meddelandet avslutas med "7777" (ASCII).

Sektion 0 (Indikator)	"GRIB" Meddelandets längd Version
Sektion 1 (Definitioner)	Sektionens längd Parametertabell, version Källa Process ID (beror på källan) Grid identifiering (enligt tabell) . . . År (tidpunkt för meddelandets skapande) Månad Dag Timme Minut
Sektion 2 (Valfri grid- beskrivning)	Sektionens längd Antal vertikala koordinatparametrar Referens till lista med vertikala koordinatparametrar Typ av datarepresentation Gridbeskrivning . . .
Sektion 3 (Valfri bitmap)	Sektionens längd Antal oanvända bitar i slutet av sektionen Flagga (indikerar om egen bitmap ska användas) Bitmap

⁵ Met Office (SADIS) sätter alltid D till 0 i sina GRIB-filer.

	<ul style="list-style-type: none"> . . .
Sektion 4 (Data)	Sektionens längd Flaggor Antal oanvända bitar i slutet av sektionen Binära skalfaktorn E Referensvärdet R Antal bitar varje datavärde är packat till Data <ul style="list-style-type: none"> . . .
Sektion 5 (Slut)	"7777"

Tabell 3.2. Tabellen visar en sammanfattning av ett GRIB meddelande. Alla sektioner är inte kompletta för att göra exemplet mer överskådligt.

ICAO har tagit fram en egen variant av GRIB som kallas ”Thinned” GRIB⁶ och distribueras endast av de två WAFC i London och Washington. Ett sådant meddelande är indelat i åtta områden, så kallade oktanter, för att dels minska storleken på filerna men framför allt för att hindra översampling vid polerna. Detta uppnås genom att gradvis minska antalet datapunkter längs med longitudlinjen när latituden närmar sig polerna. Det betyder att griddens som datavärdena placeras ut på blir icke-uniform. Den nödvändiga gridbeskrivningen finns därför definierad i sektion 2 i meddelandet. Antal datapunkter för en longitudlinje kan fås ur ekvation 3.4. Vid polerna, där latituden är 90 respektive -90, specificeras två punkter och längs med ekvatorn, där latituden är 0, finns 73 punkter.

$$n = 2.0 + \left(\frac{90.0}{1.25} \right) * \cos(\text{latitud}) \quad \text{Ekvation 3.4}$$

3.3.2 Textformat

Det finns en rad olika rapporter angående flygväder som presenteras i textform. De har till uppgift att informera om aktuellt och prognostiserat väder samt varna för ändringar i tidigare prognoser.

De flesta rapporter är uppbyggda av förkortningar, som alla piloter kan tolka. Rapporterna är därför relativt korta.

3.3.2.1 SIGMET, AIRMET

SIGMET (SIGNificant METeorological information) är en kortfattad beskrivning av existerande eller prognostiserade faror för flygplan i luften som till exempel isbildning,

⁶ Met Office (SADIS) använder sig av ”Thinned GRIB”.

turbulens, sandstormar och vulkanisk aska. En ”convective” SIGMET rapporterar om faror i samband med oväder, till exempel isbildning, turbulens och vindskjuvning.

AIRMET (Airmens Meteorological Information) är ett textmeddelande, liknande SIGMET, som rapporterar potentiellt farligt väder för små flygplan eller flygplan med begränsade förutsättningar. AIRMET täcker mindre farligt väder än SIGMET, som medelstark turbulens och isbildning, markvindar på 30 knop eller utbredd begränsad sikt. Detta är viktigt för små flygplan eftersom de både saknar den utrustning och mätinstrument som finns i större flygplan. Dessutom framförs de i regel av mindre erfarna piloter.

SIGMETs och AIRMETs skickas ut antingen på fasta tider eller sporadiskt. Informationen till en SIG- och AIRMET kommer ofta från så kallade PIREPs (se avsnitt 3.1.3).

Nedan följer ett exempel på ett AIRMET-meddelande som är hämtat från [7]. SIGMET ser ungefär likadant och finns också som exempel på [7]. Jag går inte in på några detaljer utan förklarar grundläggande vad meddelandet innehåller.

```
MIAP WAS 151900
AIRMET PAPA 2 VALID UNTIL 160100
GA FL
FROM SAV TO JAX TO CTY TO TLH TO SAV
MDT TURBC BLO 100 EXPCD.
CONDS IPVG AFT 160000Z
```

Meddelandet talar om:

- Var och när rapporten gjordes
- Vilken rapport i ordningen det är beträffande denna fara
- Hur länge rapporten gäller
- I vilket område det gäller
- Meteorologiska detaljer om faran
- Prognosen för faran

Förutom SIGMET och AIRMET finns det även mer specifika varningsrapporter om bland annat vulkanisk aska, tropiska cykloner eller strålningsolyckor.

3.3.2.2 METAR & TAF

Det görs specifika observationer och prognoser för alla flygplatser. Detta presenteras i två format som kallas METAR och TAF. De tar främst upp förhållanden som vind, sikt, moln, temperatur och dagpunkt. Dessa rapporter utfärdas vid olika tider för olika flygplatser.

METAR (METeorological Aerodrome Report) presenterar aktuellt väder vid flygplatsen. TAF (Terminal Aerodrome Forecast) är en prognos som gäller ett antal timmar framåt.

Båda rapporterna skrivs på ett liknande sätt. Nedan presenteras ett exempel på en TAF från LFV [12], 19 september för Landvetter flygplats.

```
ESGG 191130Z 191221 22014G25KT 9000 -RA  
BKN008 TEMPO 1321 2000 RADZ BKN004=
```

Meddelandet talar om:

- För vilken flygplats det gäller
- Vilken tid prognosen gjordes
- Hur länge prognosen gäller
- Vindriktning och vindhastighet i knop
- Sikt i meter
- Väder
- Molndistribution samt typ och höjd
- Om, och i så fall hur, vädret kommer ändras samt vilken tid det förväntas ske

3.3.2.3 CREX

CREX (Character form for the Representation and EXchange of data) är ett standardformat från WMO. Det liknar BUFR i funktion och uppbyggnad men är textbaserad till skillnad från binära format. CREX är således ett mer användarvänligt format än BUFR men lämpar sig inte lika bra för överföring av data. CREX kan till exempel användas när det saknas datorresurser som kan hantera binära data. Eftersom CREX sällan används tänker jag inte fördjupa mig i formatet utan hänvisar till [4] och [5].

Trots att CREX inte används i stor utsträckning idag kan formatet få en betydande roll i framtiden som distributionsformat för meddelanden i textform, till exempel SIGMET.

3.4 Utvärdering

Studien i distribution och lagring av meteorologiska data gav mycket bra information och en hel del insikter om vilka dataformat som bör användas vid visualisering av flygväder och vilken datakälla om bör användas. Nedan följer en utvärdering som redogör för dessa insikter.

3.4.1 Dataformat

BUFR är ett utmärkt format för att representera meteorologisk information i ett, av meddelandet, definierat område. Flexibiliteten gör att man kan definiera ett molnområde med koordinater för dess utbredning och ett antal egenskaper. Möjligheten att använda sekvenser gör det också lätt att definiera vektorobjekt, som till exempel jetströmmar. På grund av dessa egenskaper och med hjälp av påtryckningar från WMO så har BUFR blivit standardformatet för att lagra och distribuera så kallade SIGWX (se avsnitt 4.1).

Därför är BUFR ett självklart val som datakälla för visualisering av SIGWX. Nackdelen är att det binära formatet kräver en implementerad avkodare som tolkar koden. Utvecklingen av en sådan avkodare kan bli komplicerad och ta lång tid.

GRIB är ett format som fungerar bra för numerisk information som existerar över stora områden, till exempel temperatur, luftfuktighet och vindhastighet. Mätningar som utförs med fasta mellanrum över ett område översätts enkelt till datavärden i en grid och kan därför kodas som GRIB. Om man dessutom gör mätningar på flera höjdnivåer får man ut ett antal GRIB-meddelanden som tillsammans ger en tredimensionell representation av atmosfärens beskaffenheter.

Information om vind, temperatur och tryck i en specifik punkt kan givetvis vara intressant men det är mönstret som uppstår för ett område av punkter som är det mest intressanta. Detta gör att en representation i gridformat är idealisk för en sådan presentation. GRIB bör därför användas för visualisering av framför allt vinddata (se avsnitt 4.2), men även temperatur och lufttryck. Nackdelen är, precis som med BUFR, att implementeringen av en avkodare kan vara komplicerad och ta lång tid.

Eftersom WMO arbetar för att införa GRIB2 som standard för både SIGWX-data och den information som idag finns i GRIB1, kan det vara en bra idé att så småningom börja titta mer på detta format. Men än så länge är det BUFR och GRIB1 som är standard och bör därför användas.

Idag skickas väderrapporter, som SIGMET och AIRMET, i regel som ren text. I framtiden kan det däremot bli standard att distribuera de i formatet CREX. WMO har planer på detta och därför bör man även titta mer på CREX för eventuellt framtida bruk.

Flygplatsväder som finns representerat som METAR och TAF är någonting som används vid start och landning. Syftet och målet med arbetet är att visualisera väder för flygtrafik som är en route. Därför är dessa rapporter inget som behöver tas med i min vädervisualisering.

3.4.1 Datakälla

Konkurrensen för distribution av meteorologiska data är i stort sett obefintlig. Viljan att dela med sig av information verkar stor och tack vare organisationer som ICAO, WMO, Met Office, NWS och EUMETSAT är systemen för datadistribution väl genomtänkta.

De enda källor för stora mängder flygspecifika väderdata jag kunde hitta var Met Office och NWS, det vill säga de två WAFC som har blivit utsedda av ICAO. De följer även standarder och uppdateringar från WMO och ICAO.

Andra källor, som EUMETSAT och SMHI, har tjänster för distribution av väderdata men inte specifikt för flygtrafik.

Eftersom Met Office och NWS har tillgång till samma data är Met Office (SADIS) den självklara källan till data.

4 Visualisering

Människans visuella perception är ett effektivt verktyg för att snabbt ta in och tolka stora mängder information. Genom att representera aktuella och prognostiserade väderfenomen med grafiska symboler på en karta kan piloten snabbt få en överblick av vädersituationen längs en flygsträcka. Den grafiska presentationen kan kompletteras med text och siffror för att ge mer detaljerad information om väderfenomen.

I detta avsnitt tar jag upp de vanligaste standarder som finns och även några alternativa visualiseringar för flygväder. Ytterligare information och exempel på väderpresentationer finns i [2].

4.1 SIGWX

SIGWX (SIGnificant Weather Chart) är en väderprognoskarta för flygtrafik. Den finns som hjälpmedel för att planera flygresan och som väderinformation en route. SIGWX kan jämföras med de prognoskartor vi ser i tidningar och tv men skiljer sig givetvis i vilket väder de visar. En ”vanlig” prognoskarta visar det väder som påverkar oss människor nere på marken, framför allt i form av temperatur, moln och nederbörd. En SIGWX visar däremot vädret tusentals meter upp i luften där det påverkar flygplan som är en route, till exempel jetströmmar, turbulens och isbildning.

Förr var det standard att lagra SIGWX i ett faxformat som heter T.4 men det formatet är i stort sett taget ur bruk. Idag vill WMO och ICAO att BUFR ska vara det standardformat man använder för att lagra och distribuera SIGWX. Då krävs det att man har en BUFR-avkodare som kan tolka informationen och koppla till rätt symboler på prognoskartan.

En SIGWX görs antingen för en specifik höjd, eller för ”high level” (SWH) och ”medium level” (SWM)⁷. SWM är relevant för flyghöjder mellan FL100 och FL450 vilket motsvarar 10 000 och 45 000 fot (3 050 – 13 720 meter). SWH innefattar höjder mellan FL250 och FL630, det vill säga 25 000 och 63 000 fot (7 620 – 19 200 meter). De presenterar samma väder men tillgången till data kan variera⁸.

I [13] finns tydliga riktlinjer för vilka väderfenomen som bör finnas med på en SIGWX och hur de ska visualiseras. I följande avsnitt presenteras dessa väderfenomen och deras standardpresentation.

I Appendix B presenteras två exempel på SIGWX, en från Met Office (figur 10.2) och en från SMHI (figur 10.1).

⁷ Met Office (SADIS) använder SWM och SWH.

⁸ Till exempel täcker inte Met Office:s (SADIS) BUFR-filer alla områden för medium level data.

4.1.1 Fronter

Fronter representeras, precis som på vanliga prognoskartor, med linjer som markerar frontens position och utbredning på marken. På linjen finns symboler som visar om det är en kall-, varm- eller ocklusionsfront. Symbolerna markerar också åt vilket håll fronten är vänd åt. Använder man färg ska varmfronten vara röd, kallfronten blå och ocklusionsfronten lila (se figur 4.1).



Figur 4.1. Figuren visar exempel på fronter. Från vänster; kallfront, varmfront och ocklusionsfront.

Fronter har endast en indirekt påverkan på flygplan i luften (se avsnitt 2.2.6) och är därför inte prioriterat att ta med på prognoskartan⁹.

4.1.2 Jetströmmar

En av de viktigaste presentationerna i en SIGWX är jetströmmar. De visualiseras som en linje med en pil som markerar strömmens riktning. En jetström visas endast om dess hastighet överstiger 80 knop. Därför är hastigheten runt 80 knop både där linjen börjar och slutar.

Längs med linjen visas hastigheten på ett sätt som påminner om romerska siffror. Ett streck motsvarar 10 knop och ett ifyllt V motsvarar 50 knop. Hastighetssymbolerna placeras ovanför linjen på norra halvklotet och under på södra halvklotet. Om avståndet mellan symbolerna blir för litet kan man istället markera en hastighetsförändring på 20 knop som två parallella streck som skär strömlinjen vinkelrätt.

Förutom utbredning, riktning och hastighet är jetströmmens höjd intressant. Höjden anges i FL (flight level) och presenteras som text under hastighetssymbolerna. Där jetströmmen är 120 knop eller mer anger man även det vertikala djupet på strömmen. Djupet anges också i FL och talar om på vilken höjd, under och ovanför, som strömmens hastighet går under 80 knop. Ett exempel på en jetströmspresentation finns i figur 4.2.

⁹ Met Office har beslutat att markfronter och tropiska konvergenszoner inte längre ska finnas med på deras SIGWX. Informationen om konvergenszoner och fronter i BUFR-filerna på SADIS FTP togs bort 7 november, 2007.



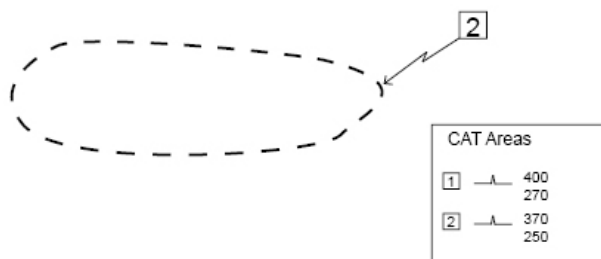
Figur 4.2. Ett exempel på en jetström som befinner sig på FL320 med en hastighet på 90 knop. Den ökar i styrka till 120 knop och det vertikala djupet, FL290 till FL340 anges. Hastigheten sjunker till 100 knop och där jetströmmen slutar är den 80 knop.

4.1.3 Turbulens (CAT)

Turbulensområden i klar luft, även kallat CAT (Clear-air turbulence) presenteras med streckade linjer. Området kan antingen vara stängt, då start- och slutpunkt ligger i samma koordinat, eller öppet.

Information om kartans samtliga CAT-områden samlas i en gemensam ruta i anslutning till SIGWX-presentationen. En siffra, som pekar på området, refererar till informationsrutan där styrka och höjd specificeras. Styrkan anges med en symbol och höjden med siffror.

Symbolen för turbulensstyrka är ett horisontellt streck med en upphöjning i mitten. Ett tak ovanför upphöjningen visar att styrkan är hård. Höjden anges med siffror som specificerar områdets bas och topp, mätt i hundratals fot. Ett turbulensområde presenteras i figur 4.3.

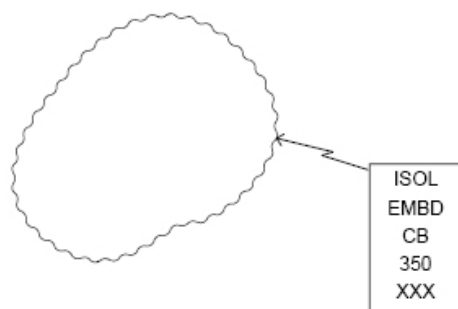


Figur 4.3. Ett CAT-område med referenssiffran 2 som hänvisar till en informationsruta. Den visar att området innebär risk för måttlig turbulens mellan FL250 och FL370.

4.1.4 Molnområden

Ett molnområde ritas ut som böljande linjer och kan vara öppet eller stängt. I anslutning till området anges information om molnets distribution, typ, bas och höjd (se figur 4.4). Molninformationen ser något annorlunda ut för en medium level SIGWX. I regel ritas endast CB-moln ut men cumulusmoln med fara för turbulens och isbildning kan också presenteras¹⁰. Information om turbulens och isbildning finns med i en medium level SIGWX och visas med standardsymboler och text (se avsnitt 4.1.3 och 4.1.9) i anslutning till molnområdet. Precis som för turbulens anges isbildningen med bas och höjd mätt i FL.

¹⁰ Met Office (SADIS) tog, 7 november 2007, bort information om icke CB-moln som inte innehåller fara för turbulens och isbildning.



Figur 4.4. Molnområde med ett antal egenskaper. ISOL = Isolated, EMBD = Embedded, CB = Cumulonimbus. Området har sin topp på FL350 och XXX betyder att molnbasen ligger under FL100.

4.1.5 Tropopaus

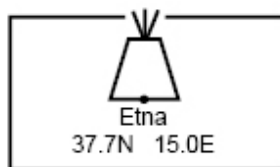
Tropopausens höjd anges i boxar över hela kartan. Höjden mäts som vanligt i FL. Där tropopaushöjden skiljer sig markant mot omgivningen presenteras den något annorlunda (se figur 4.5).



Figur 4.5. Figuren visar tre sätt att presentera tropopausen. Den första är standardsymbolen och de två andra används då tropopausen sjunker respektive höjs markant mot omgivningen.

4.1.6 Vulkan

Aktiva vulkaner presenteras med en symbol placerad vid dess position. Vulkanens namn, koordinater, och eventuell tidpunkt för senaste utbrott, anges i en separat informationsruta enligt figur 4.6. Då symbolen för vulkan kolliderar med annan visualisering på kartan har vulkansymbolen alltid högst prioritet och bör alltså renderas sist.



Figur 4.6. Informationsruta för vulkanen Etna på Sicilien.

4.1.7 Strålning

Vid risk för strålning, till exempel en radioaktiv läcka, ritas en symbol ut i aktuellt område (se figur 4.7). Information om koordinater och tidpunkt för olyckan presenteras i en separat ruta. Efter vulkaner har strålningssymbolen högst prioritet på kartan.



Figur 4.7. Symbolen för strålning.

4.1.8 Tropiska cykloner








En tropisk cyklon presenteras med en symbol och eventuellt cyklonens namn. Symbolen är vänd åt olika håll beroende på om cyklonen befinner sig på södra eller norra halvklotet. Namnet står bredvid symbolen. I figur 4.8 presenteras cyklonen Katrina som en symbol. Efter vulkaner och strålning har cyklonsymbolen högst prioritet.






Figur 4.8. En cyklon representeras av en symbol och eventuellt dess namn. Cyklonen i figuren befinner sig på norra halvklotet. En cyklonsymbol på södra halvklotet hade varit spegelvänd.

4.1.9 Övriga symboler

De väderfenomen och visualiseringar som har presenterats hittills är de vanligaste och bör alltid finnas med i en SIGWX, bortsett från fronter och konvergenszoner. Det finns dock fler symboler som kan tas med i kartan. Några av dessa symboler presenteras i tabell 4.1.

	Måttlig turbulens
	Kraftig turbulens
	Måttlig isbildning
	Kraftig isbildning
	Åska
	Regn
	Duggregn

	Snö
	Kraftig nederbörd
	Hagel

Tabell 4.1. I tabellen presenteras några vanliga vädersymboler i en SIGWX.

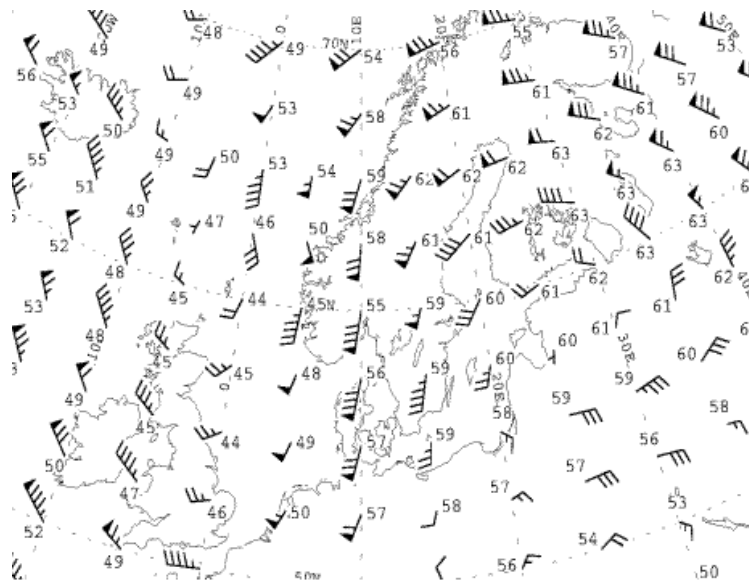
4.2 Vindvisualisering

Piloter har i regel tillgång till vindprognoser före och under flygningen. Vindar på höga höjder, så kallade ”upper winds”, kan presenteras på flera sätt. Däremot brukar samma vindsymbol användas i de flesta presentationerna. Symbolen visar riktning och hastighet på samma sätt som en jetström presenteras i SIGWX (se avsnitt 4.1.2) men är kortare, smalare och saknar pil (se figur 4.9).



Figur 4.9. En vindsymbol som visar en sydvästlig riktning och en hastighet på 80 knop.

Den vanligaste visualiseringen av vindar är en karta med vindsymboler utplacerade över hela kartan, som i ett rutnät. Kartan gäller för ett visst höjdsikt, ofta en specifik FL. Information om tropopaus och temperatur kan också finnas med. I figur 4.10 visas ett exempel på en vindkarta. I Appendix B finns en större bild som visar en vindprognos (figur 10.3). Dessa prognoskartor lagras i regel i formatet GRIB.



Figur 4.10. En vindprognoskarta över norra Europa. Bilden är tagen från [12] som har rättigheterna till bilden.

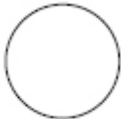





En presentation som är mer specifik för en flygtur kan anges med liknande symboler men i en tabell. Tabellen visar vindarna i olika höjdsikt, vid ett antal platser längs med flygsträckan. Även här kan man specificera temperatur och tropopaus.

Andra vindvisualiseringar, till exempel strömlinjer och animationer, finns men används sällan i flygbranschen.

4.3 Alternativa visualiseringar

SIGWX och vindprognoser är de vanligaste väderpresentationer som piloten använder sig av före och under en flygning. Det finns dock ett antal visualiseringar som är mindre vanliga.

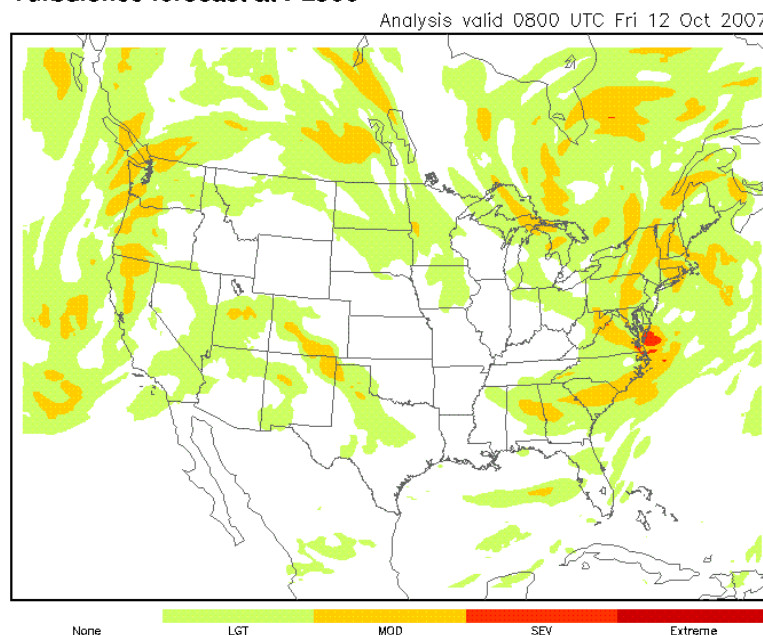
Ett exempel är ”Station Plot” som framför allt visar vädret på marken. Detta kan vara användbart för att presentera vädret för en flygplats vid start och landning. En Station Plot är en karta med standardsymboler som anger moln och vind men även temperatur, daggpunkt, tryck, med mera kan presenteras med siffror i anslutning till symbolerna. En symbol visar hur stor del av himlen som täcks av moln (se tabell 4.2). Tre symboler för olika höjdsikt (lågt, mitten och högt) representerar molntyp på respektive höjd. Standardsymbolen för vind (se avsnitt 4.2) finns också med.

					
Klart väder	Spridda moln (ca 25 %)	Delvis molnigt (ca 50 %)	Mest molnigt (ca 75 %)	Mulet	Himlen täckt (t.ex. dimma)

Tabell 4.2. Symboler för molndistribution i en Station Plot.

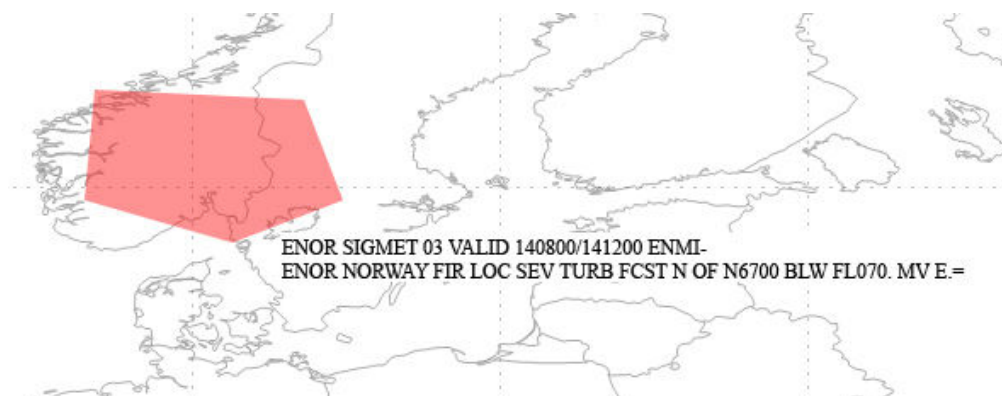
En enkel visualisering för numeriska värden, som temperatur, tryck, luftfuktighet och sikt är att helt enkelt koppla värdena till en färgskala som täcker området på en karta. Ett exempel på en sådan visualisering finns i figur 4.11.

Turbulence forecast at FL360



Figur 4.11. Turbulensdata på 36 000 fots höjd visualiseras med en färgskala från grönt till rött. På USA:s östkust finns till exempel ett område med extrem turbulens. Bilden är tagen från [9] som har rättigheterna till bilden.

Det finns inga klara standarder för att presentera rapporter, som till exempel SIGMET, AIRMET och ASHTAM (se avsnitt 3.3.2), på en karta. Eftersom det är svårt att göra en konsekvent avkodare till dessa meddelanden kan man istället behålla rapporten i textform och endast rita ut en symbol på kartan som hänvisar till textmeddelandet. Ett exempel på hur detta kan se ut finns i figur 4.12.



Figur 4.12. En SIGMET-rapport utfärdas över ett område i Norge. Området markeras med rött på kartan. När användaren klickar på, alternativt håller musen över, området dyker meddelandet upp i sitt ursprungliga textformat.

4.4 Utvärdering

Som tidigare nämnts är SIGWX och vindvisualisering de vanligaste sätten att presentera flygväder grafiskt. Det är därför naturligt att använda båda dessa i en flygvädervisualisering. Det finns tydliga standarder för hur de ska se ut vilket betyder att piloter är vana att tolka dessa presentationer.

Förutom visualiseringen av vindprognoser och SIGWX är det en stor vinst i att inkludera väderrapporter som SIGMET, AIRMET i en väderpresentation. Rapporterna uppdateras ofta vilket ger ett bra komplement till väderprognoserna som uppdateras var sjätte timma. Rapporternas format är svåra att avkoda och koppla till en visualisering på grund av deras inkonsekvens. De är dessutom till för att läsas av piloter och bör därför representeras som text som i figur 4.12.

Andra visualiseringar kan givetvis också vara till nytta men bör få lägre prioritet. Prioriteringsordningen sammanfattas i tabell 4.3.

Prioritet	Visualisering	Lagringsformat
1	SIGWX	BUFR
2	Vindprognos	GRIB
3	SIGMET/AIRMET	Text
4	Övrig visualisering	-

Tabell 4.3. Prioriteringen av visualiseringar i mitt projekt.

5 SpatialAce

SpatialAce är ett kartverktyg utvecklat av Carmenta för att underlätta vid skapandet av framför allt GIS-applikationer. Tanken är att det ska vara enkelt och flexibelt att använda. Enkelheten representeras främst av SpaceLab, en användarvänlig editor där man bygger sin konfiguration som sedan kan användas i en applikation. Konfigurationen specificerar hur data ska tolkas och presenteras på skärmen.

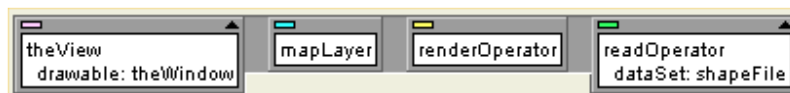
Flexibiliteten ligger i att SpatialAce stöder ett stort antal programmeringsspråk och miljöer som .NET, C#, Java, C++, m.fl. Dessutom kan egna komponenter utformas och användas i konfigurationen.

5.1 Konfiguration

En konfiguration anger hur geografiska data ska presenteras och den definieras i ett scriptspråk som heter Pilsner. Man kan givetvis skriva i scriptspråket direkt men enklast är att använda sig av SpaceLab som är en grafisk editor för att skapa konfigurationer. Konfigurationen byggs upp av en pipeline med ett antal komponenter. Eftersom SpatialAce har en mängd olika komponenter tillgängliga kommer jag främst att nämna de som jag har använt i min implementering.

En enkel konfiguration kan ha en pipeline med följande komponenter (se även figur 5.1):

- **ReadOp:** En operator som tolkar information från ett dataset, till exempel en databas eller fil, och skapar så kallade *GeoObjects* som skickas vidare i pipelinen.
- **RenderOp:** Denna operator hanterar visualiseringen genom att, via så kallade *Visualizers*, översätta *GeoObjects* till presentationsobjekt.
- **Layer:** Ett lager i kartan håller en kedja av operatorer.
- **View:** Här definieras vyn för kartan och även specificering av geografiskt referenssystem samt grafiskt API (GDI eller OpenGL). Detta är slutdestinationen för presentationsobjekten som visas i vyn.



Figur 5.1. En pipeline i SpaceLab som bygger upp konfigurationen. En ReadOp läser av en shapefil som renderas av en RenderOp. Lagret presenteras i vyn som ritar ut objekten i ett fönster.

SpatialAce kan läsa ett stort antal filformat och databaser; till exempel SHP, TIFF, MySQL och Oracle. Dessutom kan man skapa ett eget dataset, ett så kallat *CustomDataset*, för att läsa av filformat som inte redan stöds av SpatialAce. *CustomDataset* specificerar hur fildata ska tolkas för att skapa *GeoObjects*.

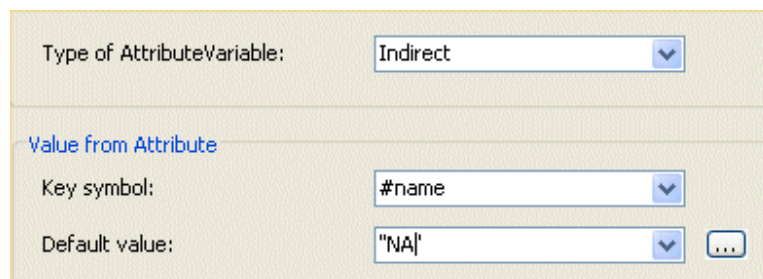
Varje *GeoObject* innehåller koordinatpunkter och attribut. Ett *GeoObject* kan till exempel vara en punkt, linje, polygon eller raster. Attributen beskriver objektets egenskaper.

Som exempel kan en bilväg på en karta representeras av ett *GeoObject* i form av en samling koordinater som bildar en linje. Dessutom har objektet attributen "roadtype", "width" och "name".

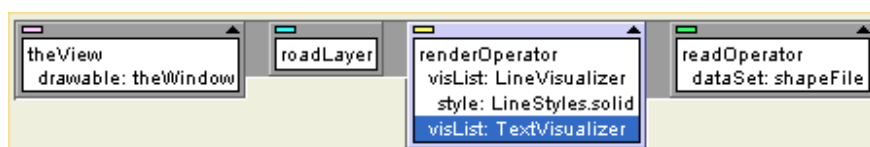
Med hjälp av *Visualizers* kan SpatialAce visualisera linjekoordinaterna och dess attribut på valfritt sätt. Olika *Visualizers* finns till förfogande, till exempel för linjer, symboler, raster och text. Man kan även programmera en egen *CustomVisualizer* om tillgängliga komponenter inte räcker till.

Genom att använda *Visualizers* för linje- och textobjekten kan man rita ut vägen som en linje med text som beror av objektets attribut. Linjens tjocklek kan då bestämmas av attributet "width" medan "roadtype" sätter en bestämd färg på linjen. Dess namn plockas från attributet "name" (se figur 5.2) och kan skrivas ut längs med vägen.

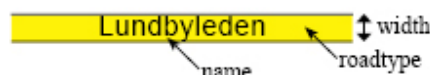
Visualiseringen samlas i ett lager som läggs i vyn (se figur 5.3). Resultatet av exemplet finns i figur 5.4.



Figur 5.2. Attributet "name" sätter texten på textvisualiseraren.



Figur 5.3. En pipeline med Visualizers för linjer och text.



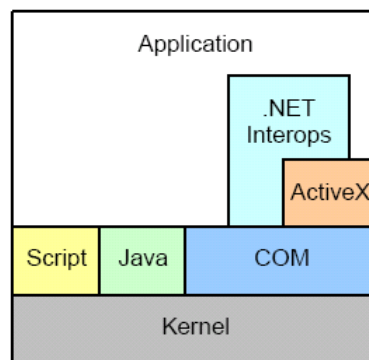
Figur 5.4. Figuren visar resultatet av exemplet med bilvägen. Attributen bestämmer text, färg och bredd på vägen.

Detta var ett enkelt exempel hur simpel geografisk data kan visualiseras med SpatialAce. Det finns givetvis en uppsjö av visualiseringsverktyg och operatorer att använda sig av för att nå ett önskat resultat.

5.2 Arkitektur

Konfigurationen använder sig av SpatialAce teknologi och skapar endast en karta som kan integreras i en applikation. Applikationen som använder kartan måste dock byggas på egen hand. SpatialAce är utvecklat i C++ men när applikationen ska skapas kan man välja flera olika programspråk. Via ett gränssnitt kan språken kommunicera med konfigurationen.

Det finns tre olika gränssnitt som möjliggör kommunikation mellan applikationen och konfigurationen; scriptspråk, Java och COM (se figur 5.5). Språk i Microsofts utvecklingsmiljö .NET kan använda sig av så kallade Primary Interop Assemblies och ActiveX-kontrollen SpaceX för att kommunicera med SpatialAce-kärnan. SpaceX är det lättaste sättet att integrera kartan i en applikation. Det underlättar även för interaktion med kartan.



Figur 5.5. SpatialAce-kärnan kommunicerar med applikationen via script-, Java- och COM-gränssnitt.

Jag kommer inte att gå in mer i detalj om arkitekturen för SpatialAce utan hänvisar istället till lämplig litteratur om .NET, Java, ActiveX, COM, och så vidare.

5.3 CustomDataset

Ett *CustomDataset* kan skapas när man vill använda ett filformat, som inte redan stöds av SpatialAce, för att förse en konfiguration med data. *CustomDataset* kan sedan användas i konfigurationen via en *DataSetProxy*.

CustomDataset använder ett interface, *ICustomDataSet*, som innehåller basfunktioner för ett *CustomDataset*. Det finns ett antal funktioner som måste finnas med i klassen. Förutom dessa metoder ska datasetet givetvis kunna tolka det önskade filformatet och översätta det till *GeoObjects*.

6 Implementering

Utifrån den utförda studien (kapitel 2 - 4) skulle jag implementera en passande presentation av flygväder i Carmentas kartverktyg SpatialAce. Detta utfördes med programmeringsspråken C# och C++, i Microsoft Visual Studio 2005 samt i SpaceLab. Jag använde även hjälpmedel som Paint Shop Pro 7 och Adobe Photoshop för att skapa rastersymboler till konfigurationen, SmartFTP 2.5 för att ladda ner filer från SADIS FTP samt SpaceExplorer för förhandsgranskning av min konfiguration. För en förklaring av .NET, C# och C++ hänvisar jag till lämplig litteratur i ämnet. SpatialAce förklaras närmare i kapitel 5.

Syftet med examensarbetet var att implementera standarder för flygväderpresentation. I min studie kom jag fram till att SIGWX var den främsta standarden. Implementeringen av SIGWX var därför en stor del av målet med arbetet och jag lade ner mest tid på detta. Applikationen som visar prognoserna var också en del av målet. När implementeringen av SIGWX och applikationen var gjord hade jag tid över och beslöt mig för att även göra vindprognoser. Denna implementering lade jag ner betydligt mindre tid på och den presenteras inte lika utförligt.

Implementeringen av båda dessa standarder genomfördes på ungefär samma sätt. Först gäller det att hantera data, att få den meteorologiska informationen ur de binära filerna. Sedan ska denna information tolkas och skapa objekt för konfigurationen. Konfigurationen använder objekten för att visualisera vädret på en karta. Denna karta integreras, med en ActiveX-kontroll, i applikationen.

Följande delar implementerades:

- SigwxLib.dll (bibliotek för SIGWX)
- WindLib.dll (bibliotek för vindprognoser)
- SigwxConfig.p (konfiguration)
- SigwxApplication.exe (applikation)

SigwxLib, WindLib och SigwxConfig kan användas som modul för att visualisera flygväder i en applikation.

6.1 SigwxLib

Biblioteket SigwxLib innehåller följande klasser, skrivna i C#:

SIGWX:

- SigwxDataset
- Jet
- JetSegment
- Cloud
- CloudArea
- MCloud
- MCloudArea
- Cat
- CatArea
- Tropopause
- TropopausePoint
- Vts
- VtsPoint
- UndefinedArea

BUFR:

- BufrBitStream
- BufrBitString
- BufrDataElement
- BufrDecodeException
- BufrDecoder
- BufrMessage
- BufrParseException
- BufrRawMessage
- BufrReader
- BufrTableB
- BufrTableBEntry
- BufrTableD
- BufrTableDescriptor
- DataElementManager

EXTRA:

- FileHandler
- FTPFactory
- Calc

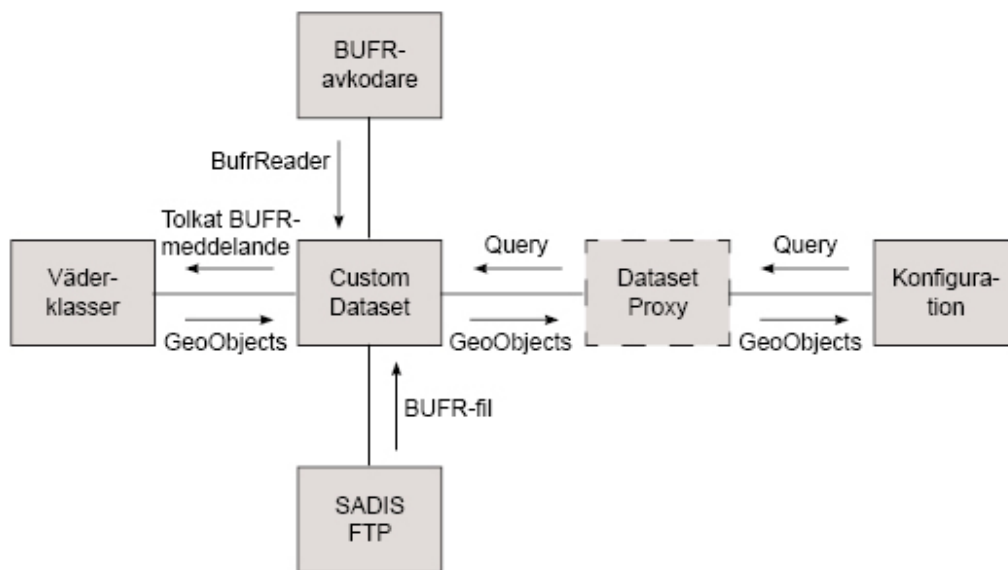
Klasserna under SIGWX skapar *GeoObjects* som sedan används i konfigurationen. *SigwxDataset* är ett *CustomDataset* som kommunicerar med konfigurationen via en proxy (se avsnitt 5.3). De övriga SIGWX-klasserna kallar jag väderklasser. Väderklasserna beskriver de väderfenomen som finns tillgängliga i BUFR-filerna, det vill säga det väder som finns med i prognosen. Varje väderfenomen representeras av två klasser, till exempel *Cat* och *CatArea*. *CatArea* representerar väderobjektet turbulensområde. Ett *Cat*-objekt skapar alla *CatArea*-objekt (turbulensområden), omvandlar de till *GeoObjects* och samlar de sedan i en vektor som skickas till *SigwxDataset*.

UndefinedArea är inget riktigt väderobjekt utan skapar de linjer som representerar områden utan information i en medium level SIGWX.

Klasserna under BUFR är skapade av Carmenta och används för att avkoda BUFR-filer. Jag har dock gjort en del förändringar i koden för att anpassa den till mitt projekt. I *SigwxDataset* skapas ett *BufrReader*-objekt för varje väderfenomen. Det används sedan för att koda av rätt fil och returnera ett *BufrMessage*-objekt som är det avkodade meddelandet. Det är detta objekt som används av väderklasserna för att skapa *GeoObjects*.

Förutom SIGWX- och BUFR-klasser finns tre hjälpklasser. *FileHandler* sköter filhanteringen. Genom att använda ett *FTPFactory*-objekt kopplar programmet upp sig mot SADIS FTP. Den laddar ner BUFR-filerna för senaste prognosen och sparar de på den lokala hårddisken. Det är dessa filer som sedan används av *SigwxDataset*. Eftersom datavärdena i BUFR-filerna inte alltid är definierade i samma enhet som för en SIGWX måste de konverteras, till exempel från m/s till knop. Detta görs med hjälpklassen *Calc*. Där finns även funktioner för att räkna ut avstånd, vinkel, med mera. Hur *SigwxLib* fungerar och kommunicerar med konfigurationen sammanfattas i figur 6.1.

När *GeoObjects* skapas i ett *CustomDataset* får de sina koordinater och attribut som sedan används i konfigurationen för att visualisera objekten (se exemplet i avsnitt 5.1). Samtliga väderobjekt skapas på liknande sätt men en del variationer förekommer beroende på olika utformning av BUFR-meddelanden samt hur de ska visualiseras. I Appendix C tar jag upp skapandet av jetströmmar som exempel eftersom de har den mest komplicerade visualiseringen av alla objekten. Jag går igenom metod, problem, lösningar och kod från avkodningen av BUFR-meddelandet till den färdiga grafiken på skärmen.



Figur 6.1. En förenklad bild av min idé för SigwxLib. En BUFR-fil hämtas från SADIS FTP, avkodas med en BufrReader. Ur det tolkade meddelandet skapas väderobjekt som omvandlas till GeoObjects. När konfigurationen ska uppdatera vyn skickas en query till datasetet som returnerar alla GeoObjects i den aktuella vyn.

6.2 WindLib

De viktiga klasserna i WindLib, tillsammans med deras header-filer, är följande

- GRIB
- GRIBDataset
- GRIBReader
- GRIBThinned

Precis som i SigwxLib innehåller WindLib ett *CustomDataset*. I detta fall heter det GRIBDataset och är implementerat i C och C++. Programkoden använder ett API i C för avkodning av GRIB-filer. Wgrib, som API:t heter, finns på [14] och går under GPL (General Public License) vilket betyder att koden får användas men inte ändras vid kommersiell användning. Wgrib fungerar inte att kompilera i Windows och därför användes en så kallad "wrapper" som översätter koden till C++ som i sin tur kan kompileras i Windows.

Jag fick göra en hel del förändringar och tillägg eftersom avkodaren inte stöder "thinned" GRIB, vilket är det format som finns tillgängligt på SADIS FTP (se avsnitt 3.3.1.2).

Den största förändringen i koden var att definiera gridden, det vill säga ta fram positionerna för datapunkterna. Klassen GRIBThinned tar, med hjälp av information i GRIB-meddelandet och ekvation 3.4, fram korrekta koordinater för en oktant. Världskartan är indelad i åtta oktanter.

Jag använde även metoder som tunnar ut antalet datapunkter, för att undvika en ”grötig” visualisering, när man zoomar ut. Dessutom implementerade jag kod som endast tar med de symboler som befinner sig innanför den aktuella vyn. Dessa funktioner förbättrar både visualiseringen och prestanda.

6.3 SigwxConfig

Den tredje delen i min implementering var konfigurationen (se avsnitt 5.1). I SigwxConfig används tre olika dataset som tillhandahåller data för bakgrundskartan, SIGWX och vindar.

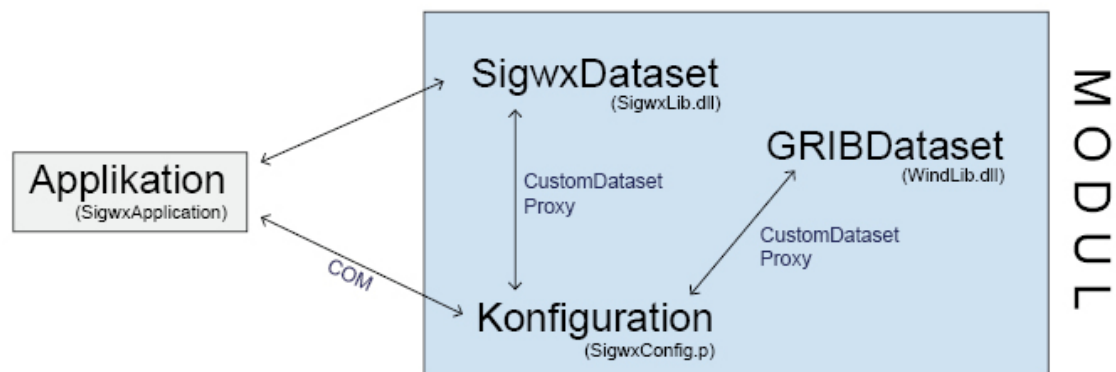
De geografiska data som används för att rita ut världskartan är i formatet VPF (Vector Product Format). Eftersom SpatialAce stöder detta format var det lätt att lägga till en operator som omvandlar vektordata till polygon-, linje- och punktoobjekt för länder, gränser respektive städer. Objekten visualiseras med passande färger och ger en bakgrundskarta för vädret att visualiseras på.

För visualiseringen av SIGWX används det tidigare nämnda SigwxDataset som via en dataset proxy förser konfigurationen med *GeoObjects* (se figur 6.1). När konfigurationens vy ändras skickas en Query till SigwxDataset som returnerar de *GeoObjects* som finns i den nya vyn. Objekten renderas med olika *Visualizers* som använder dess koordinater och attribut. Jag tänker inte gå in mer på hur konfigurationen ser ut, utan hänvisar till Appendix C som går in mer i detalj vid skapandet av jetströmmar.

Vindprognosen får sina objekt från GRIBDataset på samma sätt som för SIGWX. Den stora skillnaden är att SIGWX har flera olika *Visualizers* för att visualisera de olika väderobjekten men vindprognosen har endast en *Visualizer* eftersom den endast består av vindsymboler.

6.4 SigwxApplication

Applikationen är implementerad i Visual Studio och använder en ActiveX-kontroll och COM-gränssnittet (se avsnitt 5.2) för att kommunicera med konfigurationen. Kommunikationen med SigwxDataset sker direkt mellan klasserna men GRIBDataset nås endast via konfigurationen. Strukturen för dessa beroenden kan förklaras med figur 6.2.



Figur 6.2. Strukturen för min demoapplikation och modul. Modulen består av två dll-filer och en konfigurationsfil som är sammankopplade via varsin Proxy. Applikationen använder dessa tre komponenterna för att visualisera prognoserna.

Kommunikationen mellan applikation, dataset och konfiguration ger möjlighet till interaktion. Jag har lagt till grundläggande navigeringsverktyg i applikationen. Några finns som färdiga objekt i SpatialAce och för de andra har jag själv implementerat funktionen.

Jag har även implementerat andra funktioner. Hantering av prognoser kommunicerar med FileHandler i SigwxLib för att ladda ner nya prognoser. Val av område i kartan ändrar position och referenssystem för vyn i konfigurationen. Lagerhanteringen visar eller döljer lager i konfigurationen. Slutligen har jag implementerat en panel där man väljer höjd för vindprognosen. Mer information om hur applikationen ser ut och fungerar finns i nästa kapitel.

7 Resultat

Resultatet från mitt arbete utgörs dels av min studie (kapitel 2-4) men framför allt av implementeringen (kapitel 6). Utvärderingen som låg till grund för implementeringen finns i avsnitt 4.4.

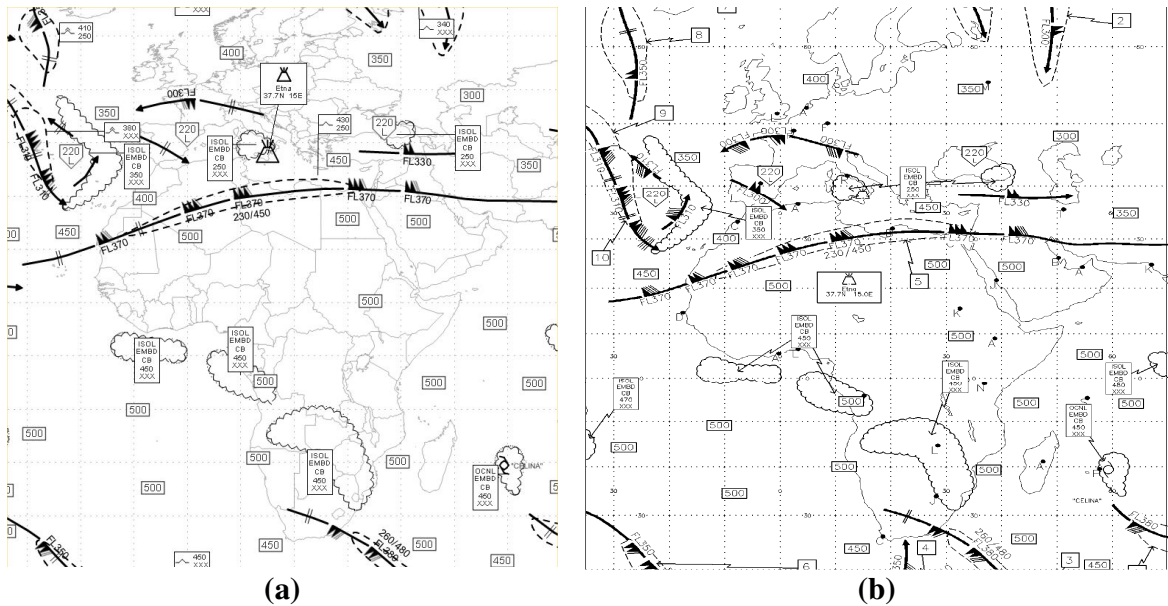
Presentation av väder enligt SIGWX standarden hade högst prioritet och var också det jag lade ner mest arbete på. Visualisering av vindar är också en viktig standard men den fick mindre utrymme i mitt arbete. I kommande avsnitt presenteras det visuella och tekniska resultatet av min implementering samt problem som uppstod under arbetets gång och förslag på vidare arbete och förbättringar.

7.1 Visuellt resultat

Mitt examensarbete gick ut på att visualisera väder och det visuella resultatet är givetvis viktigast. En prognoskarta av typen SIGWX har tydliga riktlinjer för hur de olika väderfenomenen ska visualiseras. Det är därför relativt lätt att utvärdera det visuella resultatet genom att jämföra mina prognoskartor med existerande kartor som är skapade från samma data. Eftersom WAFC London och WAFC Washington tillhandahåller kartor av typen SIGWX som är skapade från samma data som jag har haft tillgång till kommer jag använda dessa som referens till mina resultatbilder. Även vindprognoserna kommer att jämföras med andra prognosbilder. De avvikningar och fel som finns i mina kartor kommer motiveras och förklaras.

7.1.1 SIGWX

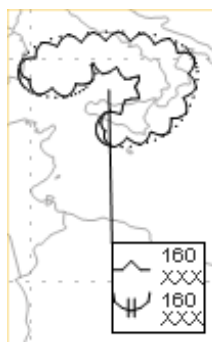
I figur 7.1 presenteras en jämförelse mellan min prognoskarta och en karta från Met Office. Min implementering använder ett annat referenssystem än Met Office vilket gör att kartorna skiljer sig något i utseende, speciellt närmare polerna.



Figur 7.1. En jämförelse av min visualisering (a) med motsvarande prognoskarta från Met Office (b). Kartorna visar en prognos för 12 december 2007, klockan 18:00. Kartorna har inte samma referenssystem vilket gör att de ser något olika ut. Met Office har rättigheterna till figur 7.1b.

En allmän avvikelse i mina kartor är att pilarna som refererar från moln- och turbulensinformation till respektive objekt i mitt fall inte är pilar, och i många fall inte ens finns med. Dels anser jag inte att rutorna är i vägen när de lappar över ett väderområde och dessutom var pilen svårare än väntat att implementera i SpatialAce. I vissa fall, när många molnområden ligger tätt ihop kan det dock vara svårt att veta vilken ruta som tillhör vilket moln.

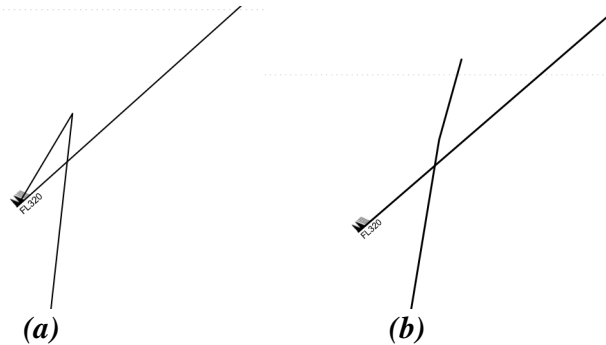
Informationsrutorna skapas dessutom i en punkt i mitten av ett område. Punktens koordinater räknas ut från områdets maximi- och minimikoordinater vilket gör att områden som kröker sig mycket kan få en ruta som pekar utanför området (se figur 7.2).



Figur 7.2. Informationsrutan pekar utanför molnområdet eftersom mittpunkten räknas fram från områdets maximi- och minimikoordinater.

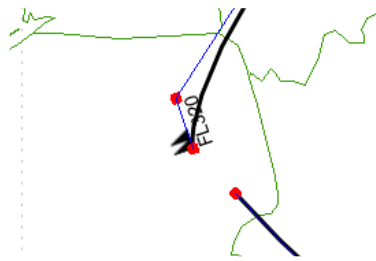
I figur 7.3 presenteras ett exempel på fenomen som kan uppstå för jetströmmar. När positionen för ett nytt jetsegment räknas ut kan det i vissa fall blir fel. Detta gör att ett

segment kan sluta efter starten på nästa segment, enligt figur 7.3b. Det är ett fel i min metod som räknar ut startkoordinaterna för ett nytt segment och jag har inte lyckats lösa det. Felet blir tydligare ju mer man zoomar in men är i regel inget som stör i prognosen.



Figur 7.3. När koordinaterna för jetströmmen är felaktiga kan följande fenomen uppstå. En förklarande linje i (a) visar hur koordinaterna är definierade och (b) visar slutresultatet.

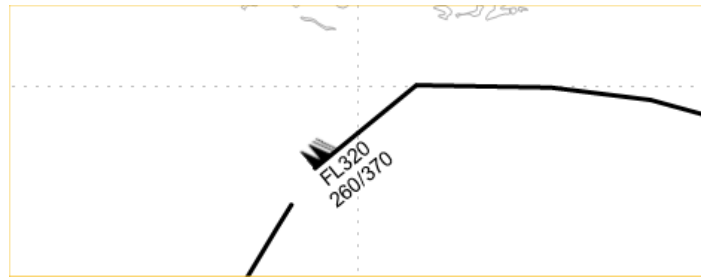
Ett annat fel som ofta uppstår i jetströmmarna är att linjen och dess hastighets- och höjdsymboler inte är perfekt placerade. Detta fel beror på att objekt längs med en linje inte tar hänsyn till om linjen använder sig av splines vilket är en bugg i SpatialAce. Objekten anpassas efter den raka originallinjen vilket kan resultera i felpacerade symboler och textrutor. Ett exempel på detta syns i figur 7.4. Detta fenomen kan även ge fel placering av symboler för hastighetsförändring.



Figur 7.4. Den tunna linjen visar hur originallinjen ser ut och prickarna representerar kärnpunkter för jetströmmen. De tjocka linjerna visar hur slutresultatet blir. Eftersom symbolernas placering inte tar hänsyn till splines blir deras inbördes placering fel.

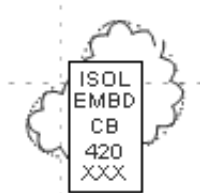
Datumlinjen¹¹ innebar en del problem för skapandet av jetströmmar. Problemen och min lösning förklaras närmare i Appendix C. Min lösning fungerar bra men kan ibland ge små defekter i de annars mjuka linjerna. När en jetström korsar datumlinjen kan det bildas en skarp krökning i linjen, vilket syns i figur 7.5. Dessa fel är i regel små och stör inte nämnvärt i presentationen.

¹¹ Datumlinjen är den longitudlinjen, på ”baksidan” av jordklotet, där -180° och 180° möts.



Figur 7.5. Den mjuka linjen får plötsligt en skarp krökning på grund av datumlinjen.

I molnvisualiseringen finns små defekter i form av oönskade linjer och prickar runt de böljande linjerna. Dessutom är de i regel inte helt slutna. Dessa fenomen syns i figur 7.6. Felen beror på klassen *ComplexLine* i SpacialAce. Det är denna klass som definierar linjemönstret för molnet. Även om molnformerna förlorar lite i skärpa är detta endast en liten detalj som inte stör så mycket.



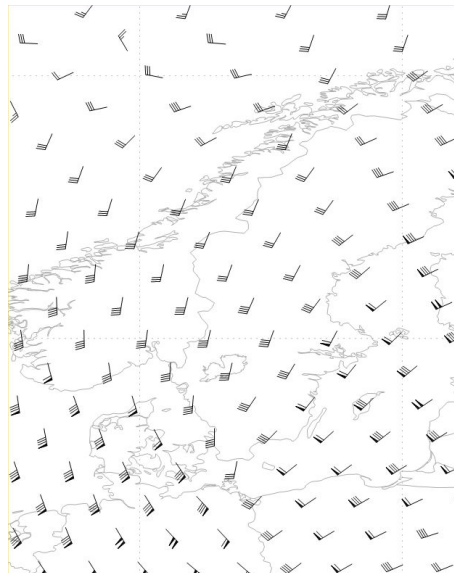
Figur 7.6. Defekter i molnlinjerna förekommer ofta i kartorna.

Enligt standarden ska information om turbulensområden hänvisas till en separat ruta i kartan (se avsnitt 4.1.3). Förteckningen över turbulensområden finns inte i min karta utan denna information presenteras istället i anslutning till respektive område. Anledningen till detta är att skapandet och uppdateringen av denna ruta hade varit mycket komplicerad att implementera. Dessutom är det enligt mitt tycke en bättre visualisering att ha informationen i anslutning till området.

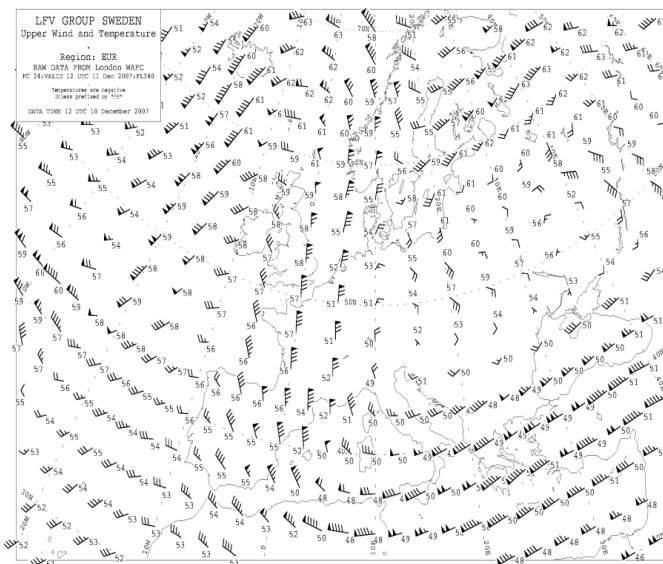
Det finns i regel två informationsrutor på kartan som presenterar allmän information om en SIGWX (se figur 10.2 i Appendix B). De finns även med i min demoapplikation (se figur 7.10) och är implementerade i Visual Studio. De följer dock inte med när kartan sparas som en bild eftersom de ligger ovanför ActiveX-kontrollen. Det är alltså endast objekten i konfigurationen som finns med i ActiveX-fönstret och därmed syns på den sparade bilden.

7.1.2 Vindprognoser

Vindprognoser har inte samma grafiska mångfald som SIGWX. Dessa kartor behöver därför inte utvärderas lika utförligt. Ett exempel på det visuella resultatet finns i figur 7.7. Denna figur kan jämföras med figur 7.8 som är en vindprognos från [12]. Notera att figurerna inte visar samma prognos.



Figur 7.7. Min vindprognoskarta över Sverige. En större version finns i Appendix B (figur 10.6).



Figur 7.8. En vindprognoskarta med temperaturer över Europa från [12] som har rättigheterna till bilden. En större version finns i Appendix B (figur 10.3).

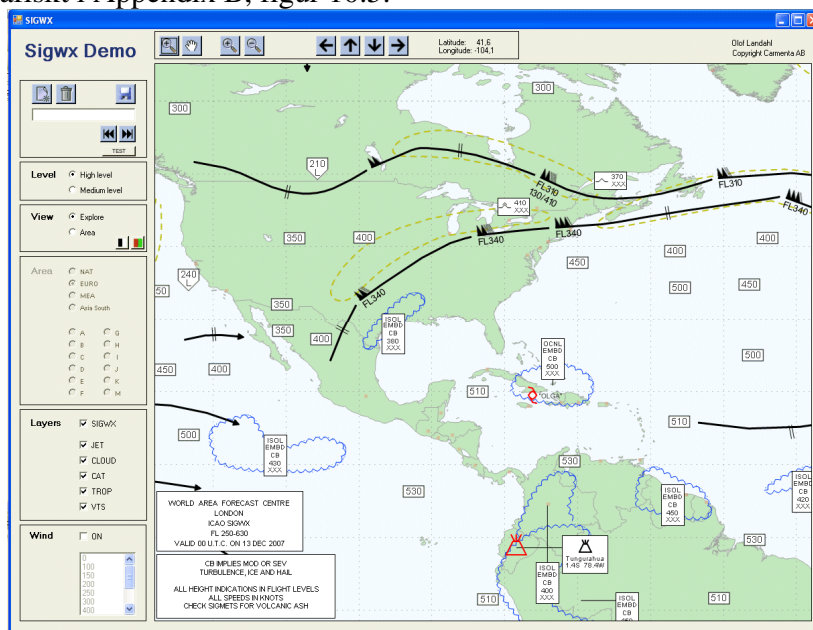
Ett problem som uppstår för vindprognoser är att vindsymboler inte ritas ut mellan -180° och -120° longitud (se figur 7.9). Detta beror med största sannolikhet på datumlinjens besvärliga egenskaper, samt de fördefinierade områdena i "thinned GRIB" men jag har inte lyckats hitta felet mer specifikt.



Figur 7.9. Mellan -180° och -120° longitud ritas inte vindsymbolerna ut.

7.1.3 Applikation

Prognoskartorna kan undersökas i en demoapplikation som jag har gjort i Visual Studio. Det finns funktioner för att navigera runt på världskartan, välja färdigdefinierade områden och andra viktiga funktioner för respektive visualiseringsstandard. Applikationen presenteras i figur 7.10 och förklaras nedan. Funktionerna i applikationen förklaras grafiskt i Appendix B, figur 10.5.



Figur 7.10. Min demoapplikation som visar prognoskartor. En större version finns i Appendix B (figur 10.4).

När applikationen startas laddas BUFR-filer för den senaste prognosen ner från SADIS FTP. Filerna avkodas, tolkas, visualiseras och visas i kartfönstret.

Översta panelen erbjuder ett antal verktyg för navigering och till höger uppdateras koordinaterna efter musens position.

Vänster panel är uppdelade i flera mindre paneler. Den första panelen hanterar filer och prognoser. Med ett tryck på den första knappen laddar användaren ner BUFR-filer för de tre senaste prognoserna och lägger på den lokala hårddisken. Nästa knapp rensar de BUFR-filer som ligger på datorn. Den tredje sparar aktuell vy som en PNG-bild.

Texttrutan visar senast utförd aktion och pilarna under bläddrar bakåt eller framåt mellan de prognoser som ligger på datorn.

I nästa panel specificeras höjdsikt, det vill säga high eller medium level. För medium level finns väderdata endast i vissa områden. Områden som saknar data markeras med streckade linjer.

Tredje panelen justerar vyn. Det finns två lägen för att välja vilket område man vill titta närmare på. Explore tillåter användaren att navigera fritt och använda de verktyg som finns. Area ger access till panelen under som innehåller färdigdefinierade områden. Dessa områden är standardiserade av ICAO men stämmer inte alltid i minsta detalj, beroende på fönstrets format och referenssystemet i mina kartor.

Det finns även två knappar i vy-panelen som justerar färgen i kartan. Från början presenteras prognoskartan i färg men föredrar man svartvitt kan man välja det.

De två sista panelerna styr över vilka lager som ska visas i kartan. I den ena panelen kan man välja att dölja eller visa specifika väderobjekt för SIGWX. Den andra panelen tillåter visning av vindprognos på en specifik höjd. Höjden specificeras i FL.

7.2 Struktur och prestanda

Andra aspekter på resultatet är den bakomliggande strukturen och prestanda. De är dock inte, i mitt fall, lika viktiga som det visuella resultatet vilket betyder att jag har lagt ner relativt lite tid på att optimera resultatet enligt dessa aspekter. Det viktiga har varit att få önskat visuellt resultat, inte att göra koden så effektiv och snabb som möjligt. Det är dock så pass viktigt att det bör utvärderas.

Den övergripande strukturen är relativt given men den detaljerade implementeringen av de olika komponenterna har stor inverkan på prestanda. Med komponenterna menar jag SigwxLib, WindLib, SigwxConfig och SigwxApplication (se figur 6.2).

Det tar relativt lång tid att starta upp applikationen men det beror givetvis på att den senaste prognosen hämtas från SADIS FTP. Finns prognosen redan på datorn används den men uppstarten går endast något snabbare. Detta beror på att programmet ändå måste koppla upp sig mot FTP:n och kontrollera om det finns färskare prognoser tillgängliga. När användaren väntar på att applikationen ska hämta de tre senaste prognoserna går det inte att samtidigt navigera i kartan vilket är en nackdel.

Samtliga interaktionsmoment tar lite tid, dels beroende på att applikationen ska kommunicera med övriga komponenter samt att applikationens objekt ska uppdateras. Framför allt är det operationer som måste leta upp objekt i konfigurationen och ändra dess egenskaper som tar lång tid. Ändå går det relativt snabbt att utföra operationer, navigera runt och växla prognoser. Datorn som jag testade min applikation på är en Pentium 4, 3.2 GHz med 2 GB minne.

7.3 Förbättringar

Givetvis finns det många förändringar och förbättringar som kan göras både visuellt och tekniskt. Det finns fler visualiseringsstandarder som man kan implementera och det går även att gå utanför ramarna och göra visualiseringar som inte är standardiserade.

Det finns ett antal visuella defekter som tas upp i avsnitt 7.1 där man kan göra förbättringar. Framför allt är det skapandet av jetströmmar som kan bli bättre. Algoritmen som ska undvika problemet med datumlinjen fungerar inte perfekt och skulle behöva några justeringar.

Felplaceringar av symboler när man använder splines är också ett svårt problem. En lösning hade varit att definiera en egen splinefunktion men det tar förmodligen relativt lång tid.

Andra detaljer som kan rättas till är defekter runt molnlinjer samt strecken mellan informationsrutor och väderobjekt som borde vara pilar.

Dessutom kan man göra en operator för SpatialAce som ser till att informationsrutor alltid hamnar innanför fönstret när ett moln- eller turbulensområde beskärs av vyn.

När det gäller vindvisualiseringen finns det inte så många detaljer som skiljer från standarden. Man skulle kunna lägga till en rastervisualisering som dyker upp när kartan är mycket utzoomad. Den skulle, med lämpliga färger, kunna visualisera var stora vindhastigheter förekommer.

Felet som gör att vindsymboler inte ritas ut mellan -180° och -120° longitud bör givetvis rättas till.

En funktion som jag gärna ville ha med i projektet var att presentera väderrapporter på kartan. Rapporter i textformat, till exempel SIGMET och AIRMET (se avsnitt 3.3.2), uppdateras i regel oftare än SIGWX och vindprognoser. De ger därför nyttig information om väderförhållanden under en flygning. Begränsningar i de filer som finns tillgängliga på SADIS FTP satte dock stopp för mina planer. Dels täcks endast vissa områden in i dessa rapporter och dessutom är de inkonsekventa i sitt språk.

I min studie kom jag fram till att väderrapporter i textformat inte lämpar sig för visualisering (se avsnitt 4.4) men en symbol som representerar meddelandet kan placeras ut på kartan (enligt figur 4.13), och via interaktion med symbolen får användaren upp meddelandet i textformat. Detta hade varit en användbar förbättring i min applikation.

Rent prestandamässigt kan en del förbättringar göras, framför allt i vindprognoserna. Tiden jag lade ner på denna del var begränsad och med C++ kan mycket effektiviseras. Ett exempel är positionerna för vindsymboler. Eftersom positionerna i grunden alltid är desamma behöver man endast räkna ut de en gång för att sedan sparas tills nästa gång kartan uppdateras. Nu räknas de ut varje gång användaren förflyttar sig på kartan vilket är onödigt.

Även SIGWX prognoserna kan förmodligen också förbättras rent tekniskt men det finns inte lika mycket utrymme för effektiviseringar i C# som i C++.

Nedladdning av prognoser från SADIS FTP låser möjligheten att navigera samtidigt. Hade FTP-hanteringen istället arbetat i en egen separat tråd kunde nedladdning och navigering ske samtidigt.

7.4 Slutsats

Syftet med detta projekt var att dels undersöka vilka standarder som finns och är lämpliga för visualisering av flygväder en route och sedan implementera en eller flera av dessa standarder. I studien kom jag fram till att SIGWX och vindprognoser är de standarder som är bäst lämpade. Detta har jag implementerat med hjälp av SpatialAce och dessutom gjort en demoapplikation där detta presenteras. Det finns givetvis förbättringar och alternativa visualiseringar som kan göra resultatet ännu bättre (se föregående avsnitt) men mitt resultat har helt klart uppfyllt målen.

MÅL:

- Framställ en omfattande studie där standarder för lagring, distribution och visualisering av flygväder undersöks och utvärderas.
- Implementera visualiseringsstandarden SIGWX med kartverktyget SpatialAce och programmeringsspråket C#. Implementeringen ska fungera som en modul som kan användas för att visa aktuella prognoskartor i en applikation.
- Bygg en applikation som använder ovan nämnda modul. Applikationen ska tillåta navigering och andra relevanta funktioner, till exempel lagerhantering och val av prognos.
- I mån av tid, implementera även andra visualiseringsstandarder med SpatialAce, i första hand vindprognoser.

RESULTAT:

- Omfattande studie där standarder för lagring, distribution och visualisering av flygväder undersöks och utvärderas.
- Modul för visualisering av SIGWX och vindprognoser (SigwxLib.dll, WindLib.dll och SigwxConfig.p).
- Applikation som använder ovan nämnda modul och innehåller navigeringsverktyg samt relevanta funktioner.

8 Referenser

- [1] **Luftens mysterier – Liten uppslagsbok om flyg å sånt**
Olle Åkerlind
Luftfartsverket, Flygtrafiktjänsten AIS/IP
Utgåva nr 2
1994
- [2] **Pilot's guide to Weather Report**
Lankford, Terry T
McGraw-Hill Professional Book Group
1999
(Linköpings Universitetsbibliotek)
- [3] **Representing WAFS Significant Weather (SIGWX) Data in BUFR**
Version 4.0
Mars 2007
<http://www.icao.int/anb/wafsopsg/WAFS%20SIGWX%20BUFR.pdf>
- [4] **Guide to WMO Table Driven Code Forms: FM 94 BUFR and FM 95 CREX**
Januari 2002
<http://www.wmo.ch/pages/prog/www/WDM/Guides/BUFRCREX-Layer3-English-only.pdf>
- [5] **WMO – Manual on Codes, International Codes, Volume 1.2**
WMO-No. 306
2001
<http://www.wmo.int/pages/prog/www/WMOCodes/ManualCodesGuides.html>
- [6] **Guide to FM 92 GRIB Edition 2**
Juni 2006
http://www.wmo.int/pages/prog/www/WMOCodes/Guides//GRIB/GRIB2_062006.pdf
- [7] **NASA – Virtual Skies**
Aviation weather tutorial
<http://virtualskies.arc.nasa.gov/weather/tutorial/tutorial6.html>
- [8] **WAFC Update**
Februari 2007
Met Office, London
http://www.metoffice.gov.uk/aviation/services/WAFC_update_brochure.pdf
- [9] **ADDs – Aviation Digital Data Service**
NWS
<http://adds.aviationweather.noaa.gov/>

- [10] Met Office: SADIS users' section**
Met Office
<http://www.metoffice.gov.uk/sadis/index.html>
- [11] EUMETSAT – Access to data**
EUMETSAT
http://www.eumetsat.int/Home/Main/Access_to_Data/index.htm?l=en
- [12] NOTAM & Flygväder – LFV**
LFV
http://www.lfv.se/templates/LFV_InfoSida_Bred____18344.aspx
- [13] Convention on International Civil Aviation, Annex 3**
Meteorological Service for International Air Navigation
15e upplagan
ICAO
- [14] WGRIB**
Climate Prediction Center (NWS)
<http://www.cpc.ncep.noaa.gov/products/wesley/wgrib.html>

Appendix A – Förkortningar

Här följer en förteckning över de förkortningar som förekommer i rapporten.

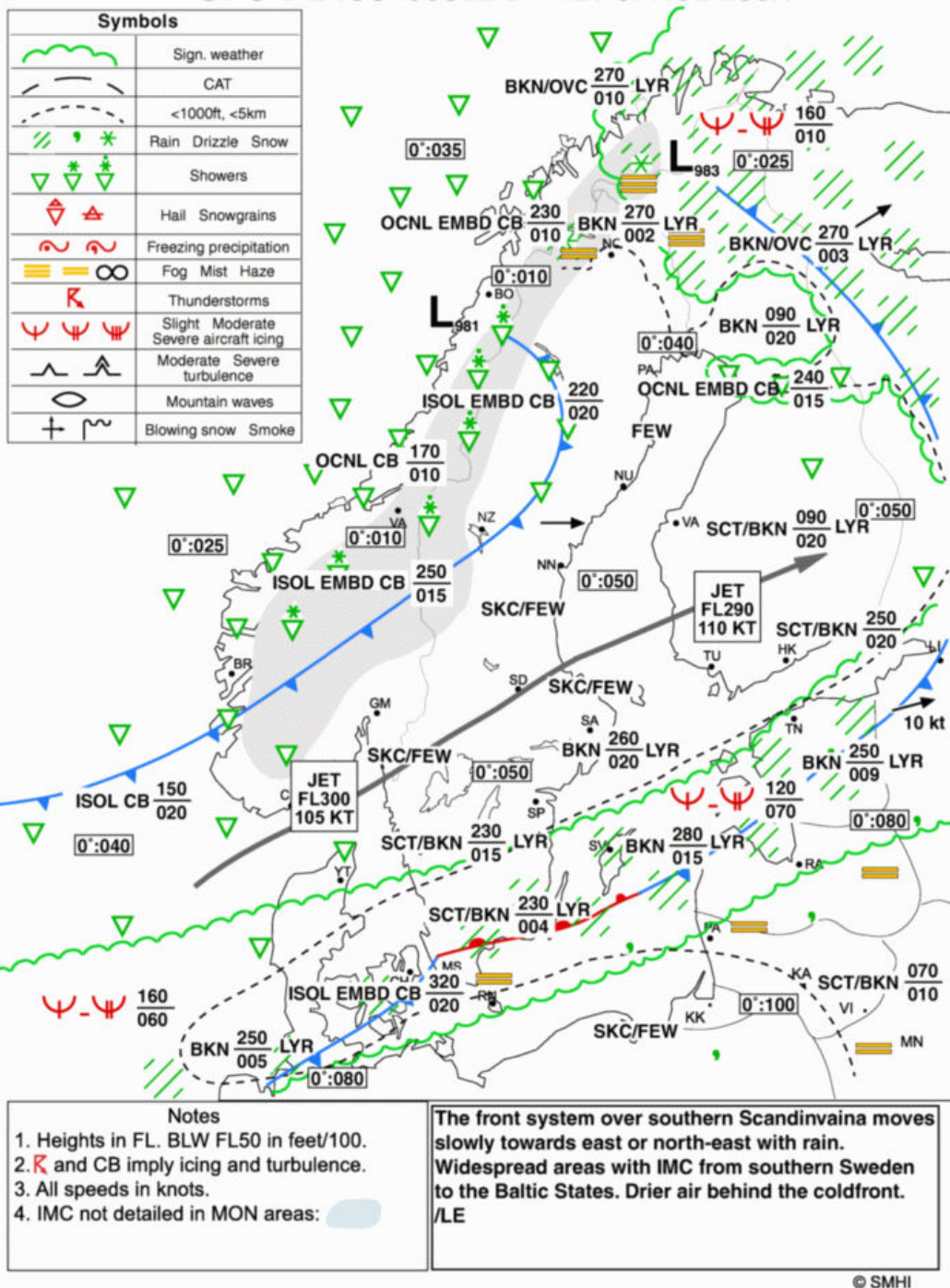
ADDs	Aviation Digital Data Service
AIREP	Aircraft Report
AMDAR	Aircraft Meteorological Data Relay
ASCII	American Standard Code for Information Interchange
BUFR	Binary Universal Format for Representing data
CAT	Clear-air turbulence
CB	Cumulonimbus
COM	Component Object Model
CREX	Character form for the Representation and Exchange of data
EUMETSAT	European Organization for the Exploitation of Meteorological Satellites
FL	Flight Level
GDI	Graphics Device Interface
GIS	Geographic Information System
GPL	General Public License
GRIB	Gridded Binary data format
ICAO	International Civil Aviation Organization
ISCS	International Satellite Communications System
LFV	Luftfartsverket
METAR	Meteorological Aerodrome Report
Met Office	Meteorological Office
NOTAM	Notice to Airmen
NWP	Numerical Weather Prediction
NWS	National Weather Service
OpenGL	Open Graphics Library
PIREP	Pilot Report
SADIS	Satellite Distribution
SIGMET	Significant Meteorological Information
SIGWX	Significant Weather Chart
SMHI	Sveriges Meteorologiska och Hydrologiska Institut
SWH	High-level SIGWX
SWM	Medium-level SIGWX
TAF	Terminal Area Forecast
VPF	Vector Product Format
WAFC	World Area Forecast Center
WMO	World Meteorological Organization

Appendix B – Bilder

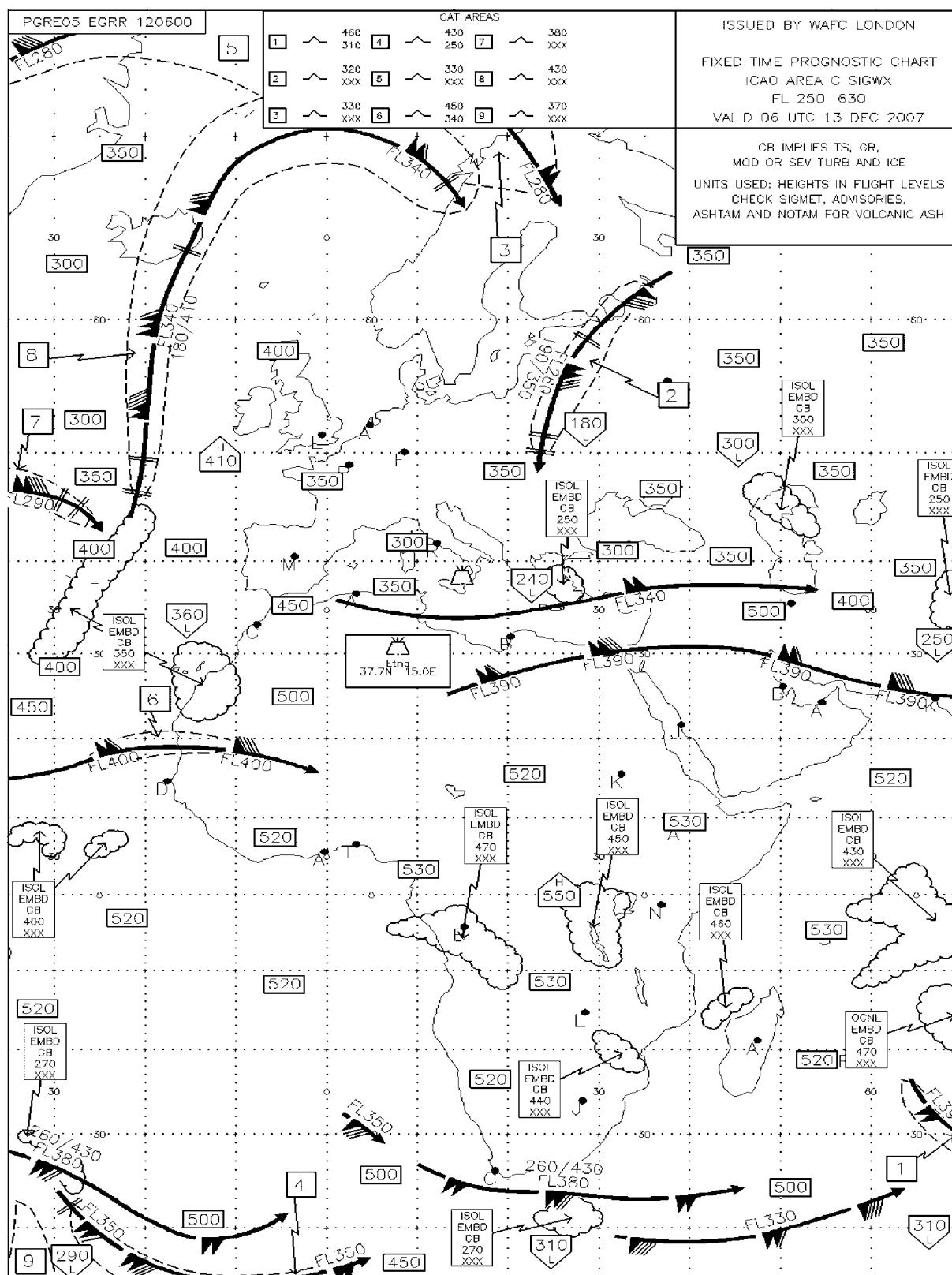
Några figurer i rapporten är för stora för att komma till sin rätt i det lilla formatet. Här finns några bilder samlade i större format.



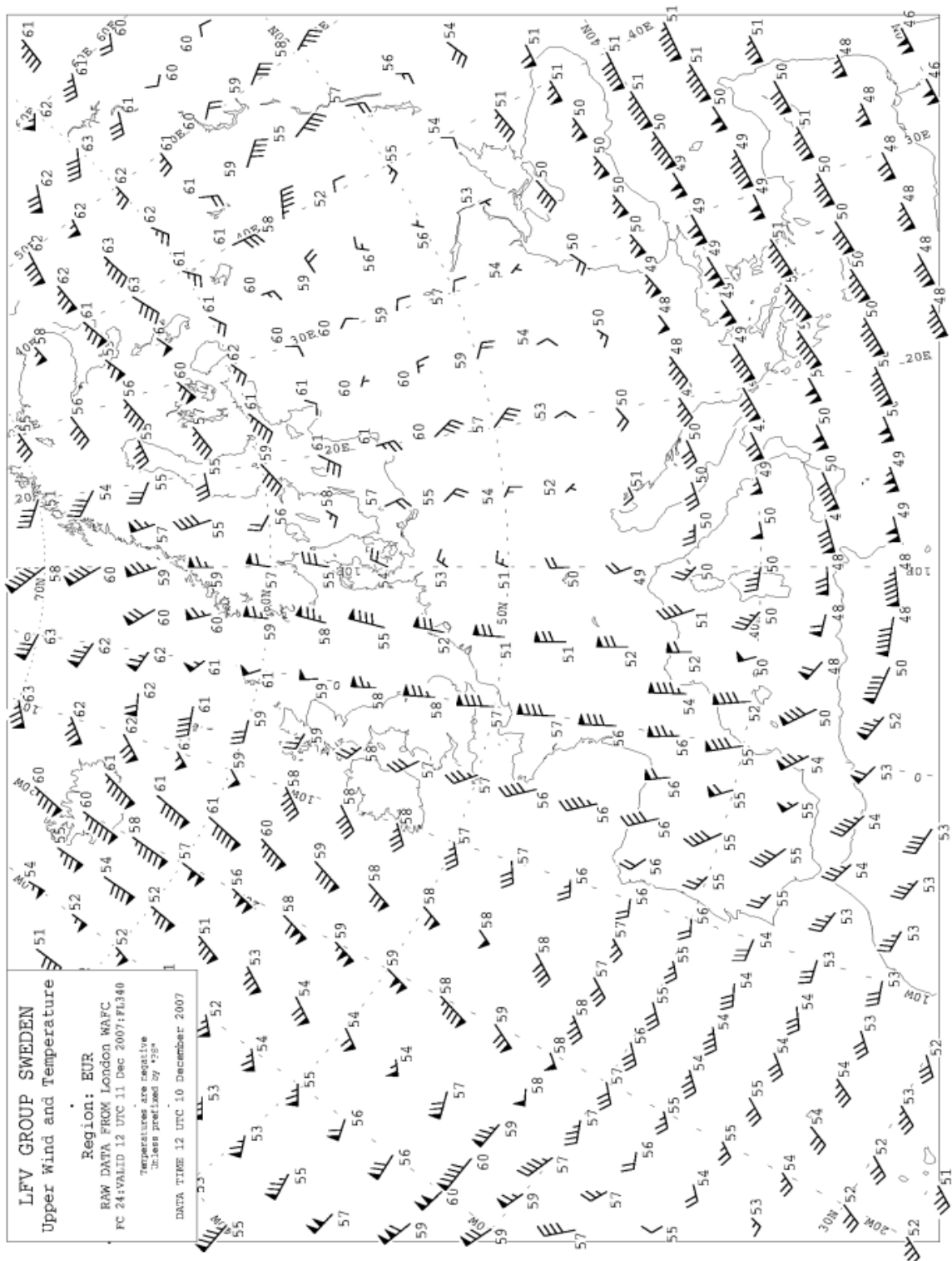
SIGWX CHART VALID 12 UTC 2007-10-17 SFC-FL400 ISSUED BY MET OFFICE ESSA



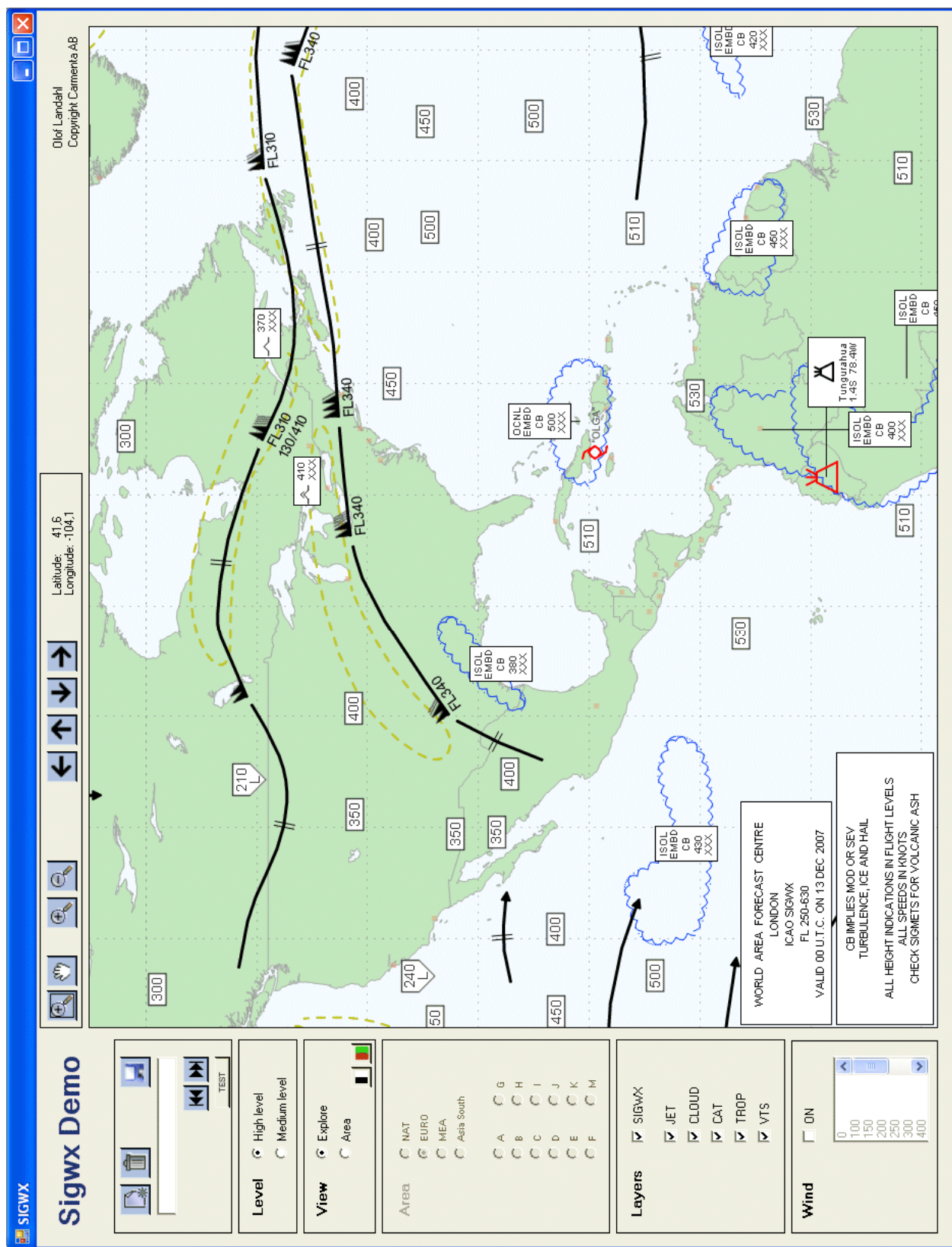
Figur 10.1. En SIGWX från [12], skapad av SMHI som har rättigheterna till denna bild.



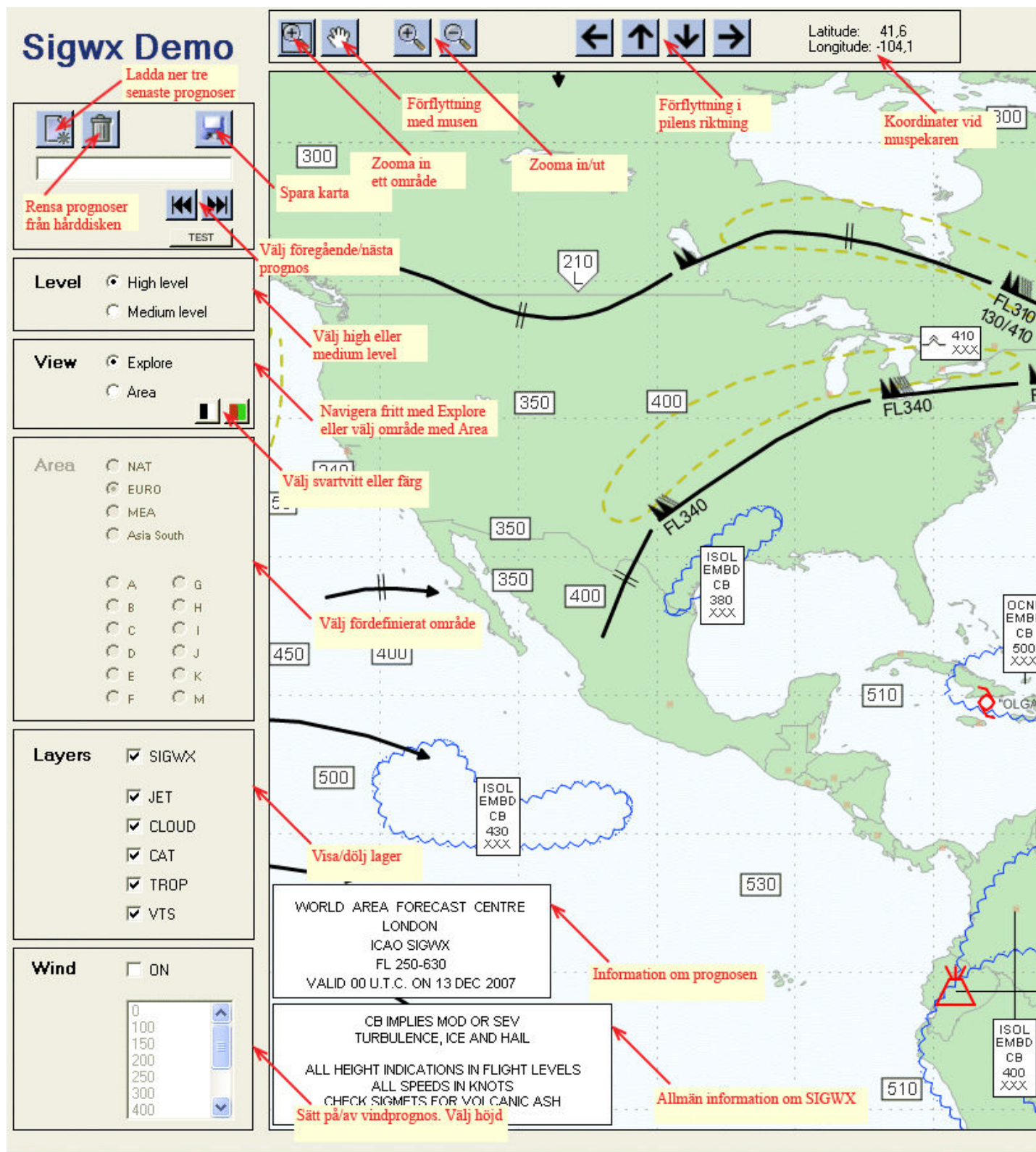
Figur 10.2. En SIGWX från Met Office, SADIS FTP som också har rättigheterna till denna bild.



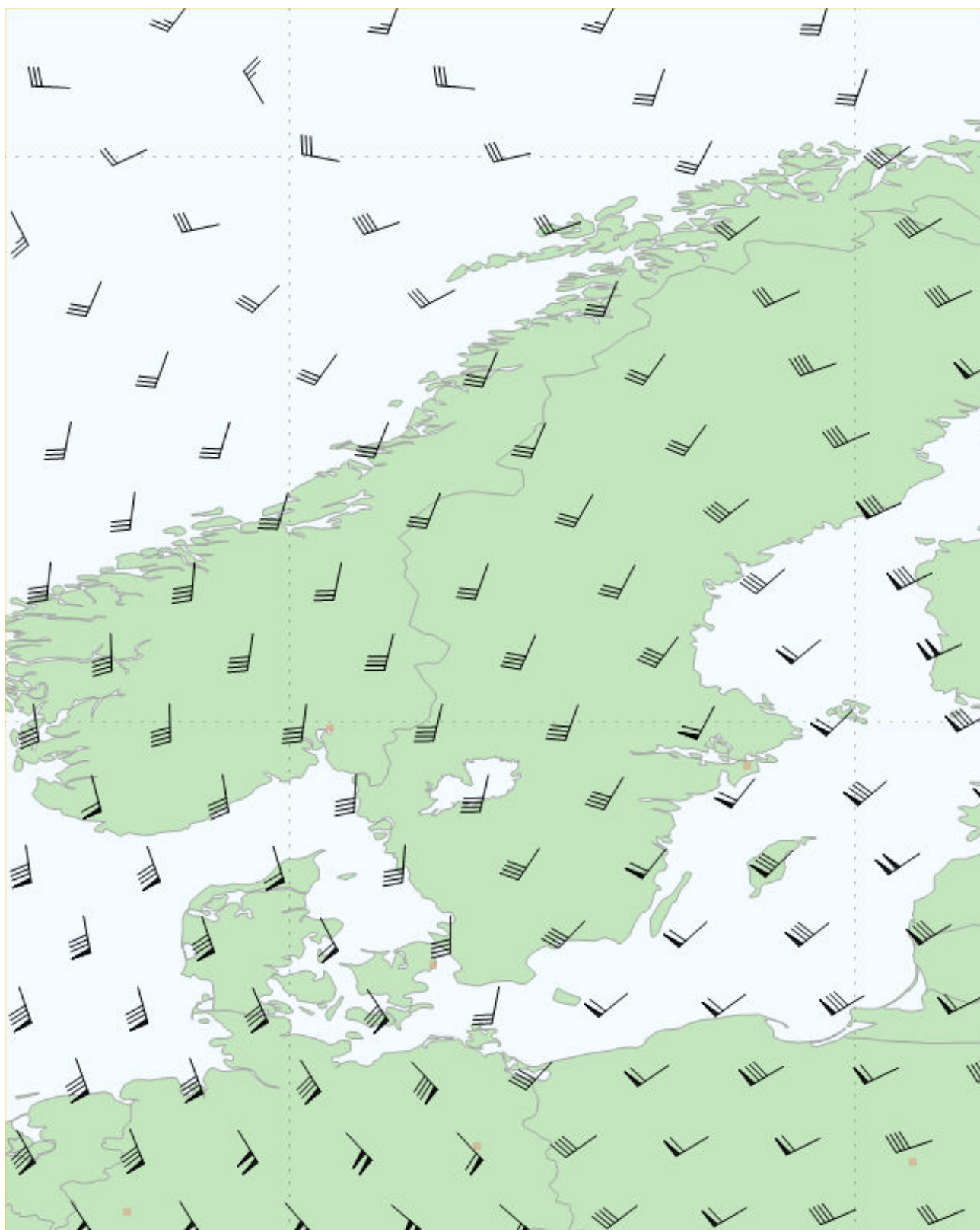
Figur 10.3. En vindprognoskarta från LFV [12] som har rättigheterna till denna bild.



Figur 10.4. Min demoapplikation som visar prognoskartor.



Figur 10.5. Figuren förklarar funktionerna i applikationen.



Figur 10.6. En vindprognos över Sverige från min demoapplikation.

Appendix C – Implementering av jetströmmar

För att ge ett mer detaljerat exempel av min implementering förklarar jag i denna bilaga hur en jetström tolkas, skapas och ritas ut. Jetström är det objekt i SIGWX-presentationen som innehåller flest grafiska element och som innebar mest problem. Därför har jag valt det som exempel. I exemplet förekommer ord och begrepp som förklaras i kapitel 2-5.

Visualiseringen av jetströmmar görs i lagret Jet i konfigurationen men för att skapa *GeoObjects* med attribut som konfigurationen behöver så används framför allt väderklasserna *Jet* och *JetSegment*.

Klassen *Jet* tolkar, med hjälp av klassen *BufrReader*, det BUFR-meddelande som innehåller information om jetströmmar. Det avkodade meddelandet innehåller en hierarki av *ArrayLists* och *BufrDataElements*. *BufrDataElement* är ett objekt som innehåller ett datavärde och dess deskriptor som beskriver det. För att iterera genom meddelandet och hitta de dataelement som är intressanta för skapandet av jetströmmar används en rekursiv funktion, *recursiveReader*.

```
// Decode the BUFR file
BufrMessage bufrMsg = _reader.HandleFile();

// Extract data recursively from the BUFR message
recursiveReader(bufrMsg);
```

Metoden *recursiveReader* går igenom ett element i taget i det avkodade BUFR-meddelandet *bufrMsg*. Objektet kan antingen vara en *ArrayList* eller ett *BufrDataElement*. Om objektet är en *ArrayList* kallas metoden rekursivt med objektet som argument. Om objektet är ett *BufrDataElement* kan den antingen innehålla data eller en ny *ArrayList*.

I BUFR-meddelandet för jetströmmar förekommer ett *BufrDataElement* som *ArrayList* alltid två gånger innan relevant data presenteras. Då anropas metoden *createJetSegments*.

```
private void recursiveReader(ArrayList elements)
{
    // Iterates through the BUFR message
    for (int i = 0; i < elements.Count; i++)
    {
        object o = elements[i];          // Stores current data element

        // Examine data element...
        if (o is BufrDataElement)
        {
            BufrDataElement de = (BufrDataElement)o;
            BufrTableBEntry desc = de.DescriptorInfo;
            object data = de.Data;
```

```

// Replicator found
if (data is ArrayList)
{
    no_of_arraylists++;

    // Extract jet data from the ArrayList...
    if (no_of_arraylists == 2)
    {
        ArrayList temp = (ArrayList) data;
        createJetSegments(temp);
        no_of_arraylists--;
    }
    // ...or iterate through the ArrayList
    else
        recursiveReader((ArrayList) data);
}

// ...or iterate through the ArrayList...
else if (o is ArrayList)
    recursiveReader((ArrayList) o);

// ...or return error message
else
    Console.WriteLine("Something's wrong");
}
}

```

Metoden *createJetSegments* tar en *ArrayList*, som i sin tur innehåller flera *ArrayLists* med relevant data för jetströmmar, och skapar jetsegment, *JetSegment*-objekt. Varje *ArrayList* innehåller en hel jetström och specificerar positioner för strömmens kärnpunkter samt dess attribut. I exemplet nedan specificeras en jetström med fyra kontrollpunkter. Trädstrukturen i exemplet visar hur ett BUFR-meddelande är uppbyggt av *ArrayLists* och *BufrDataElements*.

[0]	[0] Element Meteorological feature (10)	(Jetström)
	[1] Element Dimensional significance (1)	(Linje)
	[2] Element (no descriptor), (ArrayList)	
	[0]	(Första kärnpunkten)
	[0] Element Latitude (coarse accuracy) (La...La) (-25,13)	(Latitud)
	[1] Element Longitude (coarse accuracy) (Lo...Lo) (159,86)	(Longitud)
	[2] Element Flight level (no data)	(Höjd)
	[3] Element Wind speed (ff, fff, or fofof>fnfnfn) (no data)	(Hastighet)
	[4] Element Flight Level significance (60)	(Vertikalt djup - topp)
	[5] Element Flight level (no data)	(Höjd)
	[6] Element Flight Level significance (61)	(Vertikalt djup - bas)
	[7] Element Flight level (no data)	(Höjd)
	[8] Element Flight Level significance (no data)	
	[1]	
	[0] Element Latitude (coarse accuracy) (La...La) (-25,52)	
	[1] Element Longitude (coarse accuracy) (Lo...Lo) (162,9)	

- [2] Element Flight level (12497)
- [3] Element Wind speed (ff, fff, or fofifo>fnfnfn) (41,2)
- [4] Element Flight Level significance (60)
- [5] Element Flight level (no data)
- [6] Element Flight Level significance (61)
- [7] Element Flight level (no data)
- [8] Element Flight Level significance (no data)
- [2]
 - [0] Element Latitude (coarse accuracy) (La...La) (-25,67)
 - [1] Element Longitude (coarse accuracy) (Lo...Lo) (165,53)
 - [2] Element Flight level (no data)
 - [3] Element Wind speed (ff, fff, or fofifo>fnfnfn) (no data)
 - [4] Element Flight Level significance (60)
 - [5] Element Flight level (no data)
 - [6] Element Flight Level significance (61)
 - [7] Element Flight level (no data)
 - [8] Element Flight Level significance (no data)
- [3]
 - [0] Element Latitude (coarse accuracy) (La...La) (-25,22)
 - [1] Element Longitude (coarse accuracy) (Lo...Lo) (172,88)
 - [2] Element Flight level (no data)
 - [3] Element Wind speed (ff, fff, or fofifo>fnfnfn) (no data)
 - [4] Element Flight Level significance (60)
 - [5] Element Flight level (no data)
 - [6] Element Flight Level significance (61)
 - [7] Element Flight level (no data)
 - [8] Element Flight Level significance (no data)
- [3] Element Dimensional significance (no data)
- [4] Element Meteorological feature (no data)

Ett *BufrDataElement* beskriver antingen objektet via tabelluppslagningar eller faktiska datavärden. "Element Meteorological feature (10)" slår upp i en tabell att följande objekt är en jetström och Element "Dimensional significance (1)" talar om att objektet kan beskrivas som en linje. "Element Latitude (coarse accuracy) (La...La) (-25,13)" specificerar latitud-värdet -25,13 för en av objektets kärnpunkter.

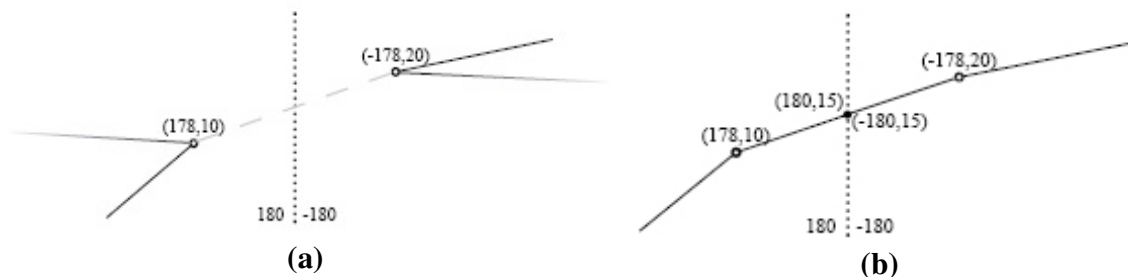
I definieringen av kärnpunkter uppstår det första problemet. En jetström är, enligt visualiseringsstandarden SIGWX, indelad i segment med ett glapp mellan varje segment. I BUFR-meddelandet definieras dock punkterna som en kontinuerlig linje. Antingen gör man jetströmmen som ett objekt där glappen tillkommer på något sätt i visualiseringen. Eller så gör man varje segment till ett eget objekt. Jag valde den senare varianten.

När en ny jetström specificeras av dataelementen initieras en ny instans av *JetSegment*. Detta objekt kan vara en komplett jetström men i regel är det, precis som namnet säger, endast ett segment.

Ett *JetSegment* består bland annat av linjens koordinater i strömmens riktning samt attribut för hastighet, höjd, vertikalt djup och ett antal booleanska variabler som talar om att det är ett jetsegment och om det är start- eller slutsegmentet i en jetström.

När en ny jetström specificeras i BUFR-meddelandet skapas ett jetsegment som också är startsegment för jetströmmen. Startsegmentet har inga hastighetssymboler utan börjar alltid på 80 knop. Nya segment i strömmen bildas då hastigheten ändras eller när den korsar datumlinjen.

När hastigheten ändras bildas antingen ett nytt segment eller, om avståndet till föregående och nästföljande hastighetsändring är för litet, ett attribut som säger att segmentet ska inkludera symbolen för en 20 knops hastighetsförändring (se avsnitt 4.1.2). Då ett segment korsar datumlinjen används en funktion som delar upp linjen i två delar. Den ena delen går från originallinjens startkoordinat till datumlinjen. Datumlinjens koordinat är givetvis longitud -180° eller 180° beroende på vilken riktning linjen har. Den andra delen går från datumlinjen till originallinjens slutkoordinat. På så vis kringgås problemet. Lösningen illustreras i figur 11.1.



Figur 11.1. I (a) presenteras problemet när en linje som korsar datumlinjen istället tar den långa vägen runt jordklotet. För att lösa problemet delas linjen upp i två delar enligt (b).

```
private JetSegment cutJet(JetSegment js, double xStart, double yStart,
double xEnd, double yEnd)
{
    JetSegment js_new = new JetSegment();

    // Calculate the y coordinate where the line cuts the datum line
    double deltaX = (180 - xStart) + (180 + xEnd);
    if (xStart < 0)
        deltaX = (180 - xEnd) + (180 + xStart);
    double angle = Math.Atan2((yEnd - yStart), deltaX);
    double t = Math.Tan(angle);
    deltaX = (180 - xStart);
    if (xStart < 0)
        deltaX = (180 + xStart);
    double yBreak = yStart + deltaX * t;

    // Add coordinates, depending on the direction of the jetstream
    if (xStart > 0)
    {
        js.AddLat(yBreak);
        js.AddLon(180.0);

        js_new.AddLat(yBreak);
        js_new.AddLon(-180.0);
        js_new.AddLat(yEnd);
    }
}
```



```

        js_new.AddLon(xEnd);
    }
    else
    {
        js.AddLat(yBreak);
        js.AddLon(-180.0);

        js_new.AddLat(yBreak);
        js_new.AddLon(180.0);
        js_new.AddLat(yEnd);
        js_new.AddLon(xEnd);
    }

    // Finish the first segment
    jetSegments.Add(js);

    // Start the next segment
    js_new.StartSegment = true;
    if (end_seg)
    {
        js_new.EndSegment = true;
        end_seg = false;
    }

    return js_new;
}

```

När ett nytt segment skapas avslutas det föregående segmentet och läggs i en lista där samtliga *JetSegment* lagras. När hela BUFR-meddelandet är avläst, och listan är komplett, är det dags att skapa *GeoObjects*.

Varje *JetSegment* i listan genererar en instans av *GeoLine*, som är ett *GeoObject*. Koordinatpunkter och attribut förs över från segmentobjektet till *GeoLine*. *GeoLine* läggs till i listan med *GeoObjects* som i SigwxDataset kommer att läggas ihop med *GeoObjects* från de andra väderobjekten och skickas till konfigurationen.

```

// Iterates through the JetSegment objects and creates the GeoObjects
for (int seg = 0; seg < jetSegments.Count; seg++)
{
    JetSegment js = jetSegments[seg];

    if ((js.Lat.Count > 1) )
    {
        // Add point coordinates
        Points ps = new Points();
        for (int i = 0; i < js.Lat.Count; i++)
        {
            Point p = new Point();
            p.Y = (double) js.Lat[i];
            p.X = (double) js.Long[i];
            p.Z = 0.0;
            ps.Add(p);
        }
    }
}

```

```

    }

    // Add attributes
    AttributeSet attr = new AttributeSet();
    if (js.StartSegment == false)
        attr["speed"] = js.Speed;
    if (js.Height != 0)
        attr["height"] = "FL" + js.Height.ToString();
    if (js.DepthBase != 0)
        attr["verticaldepth"] = js.DepthBase.ToString() + "/"
        + js.DepthTop.ToString();
    attr["type"] = js.Type;
    attr["is_jet"] = js.IsJet;
    attr["is_end"] = js.EndSegment;
    attr["has_speedline"] = js.SpeedLines;
    attr["number"] = seg;
    if ((double)js.Lat[0] < 0)
        attr["hemisphere_N"] = false;
    else
        attr["hemisphere_N"] = true;

    // Create a line object
    GeoLine gl = new GeoLine();
    gl.CreateEx(_refSys, ps);
    gl.SetAttributes(attr);

    // Add the GeoLine to the GeoObjects
    _jetObjects.Add(gl as GeoObject);
}

// Return all the Jet segments as GeoObjects
return _jetObjects;

```

När mitt *CustomDataset* har kodat av samtliga BUFR-meddelanden och skapat jetströmsobjekt, som i sin tur har genererat *GeoObjects* med koordinater och attribut, kan dessa objekt visualiseras i en *SpatialAce*-konfiguration.

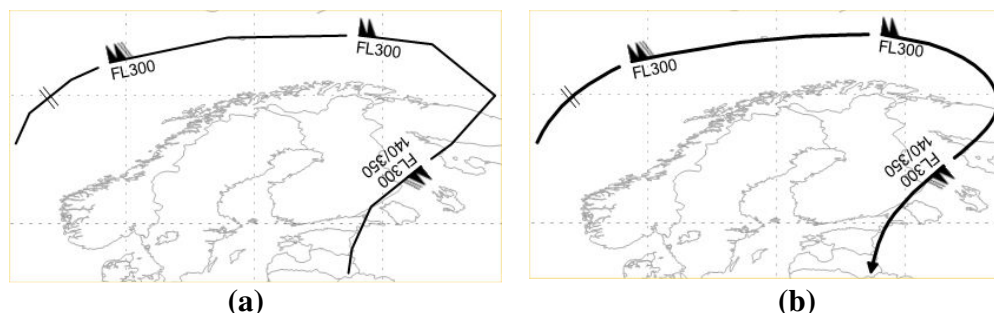
Min konfiguration är indelad i olika lager, där lagret *Sigwx* i sin tur innehåller ett lager för varje typ av väderfenomen på kartan. Varje väderlager läser data från samma dataset men använder olika *Visualizers* för att visualisera de olika objekten.

Hur en jetström bör se ut finns specificerat i avsnitt 4.1.2. Detta uppnår jag med *LineVis* som visualiserar jetsegmentets *GeoLine*-koordinater, *TextVis* som används för utritning av text och *SymbolVis* som ritar ut symboler för hastighet.

Varje jetsegment har upp till fem *Visualizers* som gör visualiseringen komplett. I figur 11.4 presenteras renderingsoperatören med dessa *Visualizers* från *SpaceLab*.

- Linjen som definierar strömmens riktning visualiseras med en *LineVis*. Dessutom används så kallade NURB splines för att göra linjens krökningar betydligt

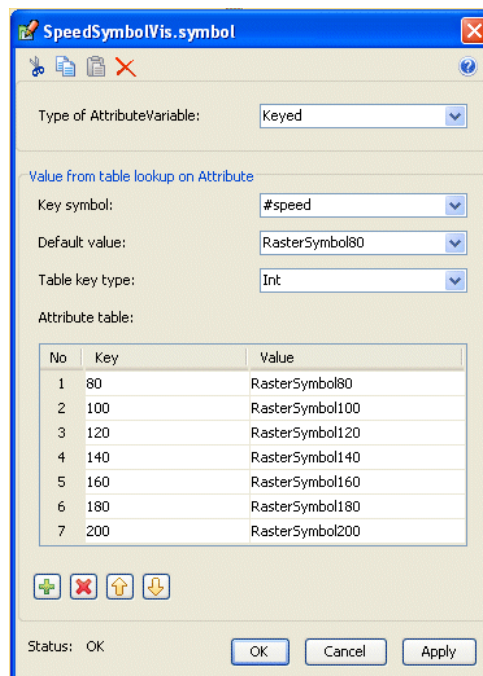
mjukare (se figur 11.2). För att inte alla andra linjer i datasetet, till exempel moln och turbulensområden ska visualiseras efter denna *LineVis* sätts ett *condition*. Det specificerar att endast *GeoLine*-objekt med attributet "is_jet" satt till *true* ska visualiseras enligt denna visualisering. Sista segmentet i varje jetström har attributet "end_seg" satt till *true* och avslutas med en pil som finns färdigdefinierad i SpatialAce. Denna metod, med booleanska attribut, används för alla *GeoObjects* i min implementering för att separera de olika väderobjekten i konfigurationen.



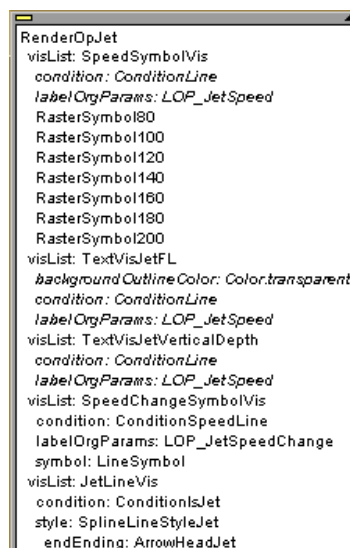
Figur 11.2. I (a) visualiseras jetströmmarna efter de punkter som finns definierade i BUFR-filerna. I (b) används NURB splines för att rita ut strömmarna.

- Symbolerna som representerar strömmens hastighet visualiseras med *SymbolVis*. En uppslagstabell används för att välja rätt symbol till hastigheten som tas från attributet "speed" (se figur 11.3). Varje symbol är en *RasterSymbol* och skapas från en PNG-fil med svarta och transparenta pixelvärden. Första segmentet har inte attributet "speed" eftersom den inte ska ha hastighetssymboler. Därefter finns en hastighetssymbol i början på varje segment i jetströmmen.
- Ett problem som uppstod vid visualiseringen av hastighetssymboler var att de, på södra halvklotet, ska speglas horisontalt mot jetströmmens axel. Eftersom detta inte gick att göra direkt i SpatialAce fick jag helt enkelt skapa en likadan pipeline för södra halvklotets jetströmmar men med andra bildfiler som speglats horisontalt. För att leda jetströmmarna till rätt pipeline (södra eller norra) inkluderar jag en operator som filtrerar fram/bort jetobjekt beroende på om attributet "hemisphere_N" är satt till *true* eller *false*.
- Symbolen som på vissa segment visar en hastighetsändring på 20 knop visualiseras också av en *SymbolVis*. Den aktiveras endast om segmentets attribut "has_speedline" är satt till *true*. Symbolen är en *VectorSymbol* som definierar två parallella streck.
- Höjden visualiseras med *TextVis* som sätter texten till jetströmmens attribut "height". Texten har utgångsposition vid linjens början och måste därför förflyttas neråt i linjens lokala koordinatsystem.

- Det vertikala djupet presenteras endast för vissa segment och det görs också med en *TextVis* på samma sätt som för höjden, fast med attributet "vertical_depth".



Figur 11.3. Tabellen använder attributet "speed" för att välja rätt hastighetssymbol. Varje RasterSymbol länkar till en bild i formatet PNG.



Figur 11.4. Fem Visualizers används för att visualisera jetströmmar; SpeedSymbolVis, TextVisJetFL, TextVisJetVerticalDepth, SpeedChangeSymbolVis och JetLineVis.