SESAM-gruppen i Programvarusäkerhet har studerat ett antal systemsäkerhetsanalysmetoder inom ett mikroprojekt benämnt 'Säkanalysmetoder' (se *Spec_SafetyAnalyses*).

Syftet med studien har bl a varit att utreda vilka av dessa analysmetoder som speciellt lämpar sig för programvarusystem vid olika skeden i systemutvecklingen samt på vilket vis dessa kan bidra till att identifiera de olika typer av riskkällor som kan föreligga i ett system. Ett resultat av studien är denna sammanställning, vilken är avsedd att även kunna användas som självstudiematerial.

Projektet inleddes med att framställa följande studieunderlag:
- **Faktablad**, som kortfattat beskriver de successiva systemsäkerhetsanalysfaserna samt de analysmetoder som avses provas under dessa.
- **Typexempel** (i detta fall ett delsystem) på vilket dessa metoder kan appliceras.

Metodikproven genomfördes under gruppmöten, då även andra punkter avhandlades, varför endast minimal tid kunde avdelas för analys. Efter varje analysmöte kördes därför en mail-stafett bland analysdeltagarna för att ge utrymme för kompletteringar och 'avslut' av studerad teknik före nästföljande möte och metodprov. Givetvis innebar detta att provning och resultat på inget vis kunde slutföras helt, men övningarna kom ändock att bidra till ökad kunskap och insikt i systemsäkerhetsområdets olika analysmetoder.

Ansatsen inför första analysprovrundan var, att PHL-analys redan genomförts på övergripande systemnivå och nu nått raketstolssystemet 'Ejection system'. Detta delsystem fick tjäna som typexempel, vilket sedan användes för en partiell identifiering av systemets riskkällor under olika analysfaser. Beskrivningar av typexempel och övrigt ingångsmaterial samt delresultat för de olika analysfaserna återfinns nedan. Förenklingar har gjorts, bl a betr. systembeskrivningar samt när det gäller att återge de iterationer i analysförlopp och mail-utväxling som genomlöpts:

I) PHL-analys (se faktablad *PHL_fakta*)

I:1 Första analysmötet

Ingångsmaterial:

1) Historik räddningssystem (*HistorikRäddnsystFlyg*)
2) Konceptuell systembeskrivning (*EjectionSyst_Conceptual*)
3) Konceptuell design (*RaketstolEnvelopUtg1_1*)
4) Olycksutfall/bilder (*A6BadEject1*, *A6BadEject2*, *A6BadEject3*, *F16BallOut*)
5) Olycksscenarier/film (*Acc4Mig29*, *EjectToGround*, *EjectFromHarrier*, *JetAirShowCrash*, *LyckligaPiloter*)
6) Generell riskkällelista (*GenHzChecklist*)
7) Checklista programvaruriskkällor (*GenSwHzChecklist*)

Presentation av delsystem och tillämpningens utvecklingshistorik. Genomgång av olycksscenarier (filmer), deltagarnas erfarenhetsbank, generella riskkällelistor samt aktuellt system för riskkälleidentifiering (se *Ant_M14_06*).

Resultat:

8) Preliminär riskkällelista (*PHL_Eject1*).

I och med att riskkällelistan under efterföljande mail-stafett kom att detaljeras med skattade sannolikheter för olika olycksutfall, övergick PHL-analysen till PHA (se II). Möjligheten att ge trovärdiga sannolikhetsskattningar kan vara ytterst begränsad _ i synnerhet beträffande systemets programvarudelar _ men i detta fall förelåg viss statistik, vilken ansågs kunna ligga till grund för skattningarna. Layouten för den PHA-lista som togs fram blev därför inte helt renodlad och kan behöva justeras ytterligare i samband med slutsummering av detta studieprojekt.
Frågeställningar under detta möte resulterade även i modifieringar av tidigare designbeskrivningar.

II) PHA (se faktablad *PHA_fakta*):

Resultat efter mail-stafett #1:

9) Preliminär riskkällelista (*PHL_Eject5*).
10) Kompletterad design (*RaketstolEnvelopUtg3*)

II:1 Andra analysmötet

Diskussioner under detta möte kom bl a att gälla distinktionen riskkälla_riskkälleorsak samt möjligheten/nyttan av att beakta riskkällans ’värsta troliga konsekvens’ snarare än ’värsta möjliga konsekvens’*(*se *Ant_M15_06).*

Resultat:

11) Preliminär riskkällelista (*PHA_Eject6_1*).
12) Funktionsbeskrivning (*BkFknsbeskrRaketstolUtg1_1*)
13) Kompletterad design (*RaketstolEnvelopUtg4_2*)

Riskkälleidentifiering innebär att ytterligare säkerhetskrav samt designrestriktioner kan formuleras. Dessa leder till designmodifieringar, där identifierade riskkällor _ så långt praktiskt möjligt _ elimineras/undviks samt de resterande hanteras genom kontroller och varningar, där utbildning och träning utgör ett sista led i åtgärdslistan.

II:2 Tredje analysmötet (HAZOP-prov)

Nytt material inför mötet:

14) HAZOP-teknik (*HAZOP_fakta)*
15) Allmänna nyckelord/ledord *(Guidewords)*
16) HAZOP-tabell, ett delvis ifyllt exempel *(Hazop_Eject0)*
17) Felhanteringssystem, ett nytt delsystem för HAZOP-prov (*FelhantSys_HAZOP*)

Under detta möte tillämpades HAZOP-tekniken _ en systematisk genomgång av varje studienod och detaljgranskning av dess parametrar för att m h a givna nyckelord finna avvikelser från avsedd konstruktion och drift. Dessa noteras i en speciell HAZOP-tabell. De avvikelser, som därvid bedöms kunna utgöra ett hot mot systemsäkerheten införs dessutom i PHA:s riskkällelista.
Studien inskränktes i detta fall till dataflöde 7 och speciellt parametern ’Höjd över mark’. HAZOP-teknikens allmänna nyckelord tolkades för denna parameter och en HAZOP-tabell skapades för de ledord, som därvid befanns vara relevanta *(*se *Ant_M16_06)*.

Resultat:

18) HAZOP-ledord för Ejection system (*Guidew_EjectSyst1_1)*

19) HAZOP-tabell, påbörjad för studerat delsystem (*Hazop_Eject1_2*)

Resultat efter mail-stafett #2:
20) HAZOP-tabell, kompletterad m a p resterande nyckelord (*Hazop_Eject1_3*)
21) PHA-listan, kompletterad med säkerhetskritiska avvikelser (*PHA_Eject6_2*)

II:3 Fjärde analysmötet (FMECA-prov)
Nytt material inför mötet:
22) FMECA-metoden *(FMECA_fakta)*
23) A Hierarchical FMEA Unified System Model for Comprehensive Hazard Analysis, B. J. Czerny et al, ISSC´06.
24) Felhanteringssystem/-modul anpassat för FMECA-prov (*FelhantModul_FMECA*)

Syftet med en FME(C)A (som kan appliceras både på produkt och process), är att för varje ingående komponent finna potentiella felmod/felsätt samt att undersöka vilken effekt dessa har lokalt o på systemnivå, för att kunna föreslå konstruktionsändringar prioriterade efter allvarlighetsgrad.
En uppdaterad variant av Felhanteringssystemet utgjorde underlag för FMECA-provet. Att i detalj beskriva en modul inom detta hade visat sig alltför omfattande, varför förberedelserna fick inskränka sig till en mer översiktlig skiss av olika funktionsmoden och ett antagande att varje funktionsmod svarar mot en modul.
Slutsatsen från detta analysmöte blev, att antingen var underlaget inte tillräckligt detaljerat, eller också passar inte metoden på ännu ej implementerad programvara.
Mot detta skulle tala, att en variant av metoden tillämpats, där strukturell och funktionell FMEA fått utgöra basen för en hierarkiskt FMEA inledd på övergripande systemnivå (se 23 ovan). Vid närmare eftertanke fann vi f ö, att de tillämpningar av metoden som publicerats snarare verkar analysera felmod hos hårdvara än programvara (ventil fastnad i visst läge). Detta ledde till definition av en hemläxa (se 26).
De felmod för programvara som identifierades under mötet var mer av generell natur (deadlock, ohanterade exceptions, minnesbrist etc) än unika för studerat system. När det gäller att finna de generella feltillstånden/ felsätten, har dock programvaran andra, specialiserade och mer effektiva analystekniker att ta till.
Resultatet av denna analys, kan tyckas misslyckad i så måtto, att inga unika felmod identifierades, men slutsatsen kan ha sitt värde (om den inte motsägs vid förnyat prov).
FMECA-tabellen nedan visar en tidig ansats snarare än ett uppstrukturerat, mer komplett resultat *(jfr Ant_M17_06)*.

Resultat:
25) FMECA-tabell påbörjad för studerat system/modul *(FMECA_1)*
26) Hemläxa: Leta efter fall, där FME(C)A visat sig effektiv i att identifiera felmod i själva programvaran.

II:4 Femte analysmötet (FTA-prov)
Nytt/uppdaterat material inför mötet:
27) FTA-metoden *(FTA_fakta)*
28) PHA-listan kompletterad med riskkällor identifierade under FMECA (*PHA_Eject6_3*)
29) Retrenchment and the Generation of Fault Trees, SafeComp´06 (FTA_fakta:ref 11).

Hemläxa enl punkt 26: Ur publicerade tillämpningar framgår att det är den funktionella FMEA-varianten som visat sig tillämpbar för programvara. Analysen inriktas mot att studera resultat/utdata/beteende hos ingående programvarufunktioner för att finna typiska felaktigheter i dessa, snarare än i implementerad kod. De exempel som återfanns rörde dock mest övergripande funktioner (före uppdelning i mjuk- resp hårdvara). Som exempel på typiska felmod hos programvara nämns stoppad exekvering/ systemkrasch, överskriden resursförbrukning, uppstartsproblem samt att utförd (eller överförd) funktion/ kommando/ meddelande sker inte-alls, felaktigt,vid-fel-tidpunkt etc.
Den slutsats som drogs efter denna genomgång var att de felmod som kan identifieras antingen är av generell karaktär, där programvarutekniken erbjuder andra, mer specifika o effektiva analysmetoder eller också rör interaktionsdelar, där FMEA inte synes tillföra något utöver vad som går att få fram med HAZOP *(jfr Ant_M18_06)*.

FTA-provet bestod i att bygga ett logiskt felträd utgående från någon av de riskkällor/ vådahändelser, som tidigare identifierats vid ovanst PHL/ PHA-analyser. Syftet är att utreda bakomliggande orsaker samt att försöka skatta sannolikheterna för dessa, för att bedöma var risklindrande designändringar är nödvändiga _alternativt för att verifiera att systemsäkerheten är tillgodosedd.
Eftersom det åtgår ett träd per vådahändelse o systemmod, inskränkte vi oss till topphändelsen 'Oavsiktlig utskjutning' vid systemmod 'Underhåll' (Haz-27 i riskkällelista enl. punkt 28 ovan). Systemgränsen bestämdes av vårt studiesystem 'Ejection system' (ett utskjutningssystem för raketstol). Det stoppkriterium som fick begränsa analysens omfattning var, att avbryta vidare nedbrytning av trädets noder, där sannolikhetsbidraget bedömdes vara försumbart eller där underliggande noder förutsätter detaljerad källkod. Processen att bygga felträdet ledde till klargörande diskussioner o därmed ökad insikt om vilka delar som stod för de största riskbidragen. En grov, erfarenhetsbaserad skattning av nodhändelsernas sannolikheter (p) gjordes.
En del av FTA:s styrkor för programvara framgick av detta prov, t ex att kunna

- värdera programvarans bidrag till topphändelsen m h a känslighetsanalys o ansatsen p=0 resp =1,
- identifiera de mest kritiska delarna (där designändringar nödvändiga),
- indikera var/hur risklindring är möjlig (för *CutSets* _händelsekedjor mot toppen_ en strävan mot att ersätta eller-grindar med &-grindar o därmed en summa av sannolikheter med en produkt _givet oberoende delhändelser). Att införa diversitet (vilket motsvaras av &-grindar) är m a o ett effektivt sätt att minska riskbidraget från kritiska delar.

FTA-provet genomfördes ned till programvarunivå, men bör vid ett senare tillfälle ev. fortsättas ned i programvarustrukturen (givet tillräckligt detaljerat underlag kan tas fram). Det finns ex där FTA tillämpats på programvara liksom mallar för vanliga språkelement (tilldelning, if, while, proceduranrop etc).

Det logiska komplementet till Felträd o dess *Cut Sets* (dvs *Success Trees* resp *Path Sets*) diskuterades även. Efter diverse efterforskningar fann vi, att dessa begrepp inte återfinns i beskrivningar av FTA-metodik eller dess verktyg, utan snarare torde användas för implementering av FTA-verktyg vid validering av resulterande felträd. Vi beslöt därför att inte utvidga FTA-provet med ett matchande ST-prov.

Resultat:

30) Felträd ’Räddningssystem flygplan’ (*Raketstol_FTA_Haz27_Utg1_1*)

II:5 Sjätte analysmötet (FTA-prov på programvara)
Nytt/uppdaterat material inför mötet:
31) Rättning av felträd under punkt 30 (*Raketstol_FTA_Haz27_Utg2_1*): Tillägg av mikroswitch.
32) Kompletterad design (*RaketstolEnvelopUtg5*): Programfunktioner i ’Räddningssystem Flygplan’ detaljerad inför en SW-FTA.

Med utgångspunkt från nod ’Programvarufel’ i felträd enl punkt 31 tog varje deltagare fram sin variant av underliggande noder, varefter samtliga förslag diskuterades. Även om felträden skiljde, var de relativt likartade och förfiningen inskränkte sig till slutnoder av typ felaktiga data (terräng-/nav-/indata etc), felaktig algoritm etc *(jfr Ant_M19_06)*.

Resultat:
33) Programvarudelen av felträd ’Räddningssystem flygplan’ (*Raketstol_FTA_Haz27_SW_Utg3*) med två utkast till mjukvaruFTA sist i filen.

II:6 Sammanfattning analysmöte 1-6 (SESAM-redovisning)
De erfarenheter och slutsatser som Programvarusäkerhetsgruppen dragit ur ovanstående analysprov presenterades för SESAM-medlemmarna i januari 2007.
En tidig insikt blev värdet av, att _utöver PHL-analysens traditionella ingångsmaterial _ kunna visa filmade scenarier av applikationen i normala resp olycksdrabbade förlopp.
Till de mer överraskande resultaten beträffande metodernas användbarhet på programvarusystem hör styrkan hos HAZOP, att på ett systematiskt sätt stötta sökandet efter avvikelser från avsedd design av systemets interaktionsdelar (meddelanden, signaler, flödeselement etc).
Lika förvånande var, att den i andra sammanhang så väletablerade FME(C)A-metoden inte övertygade i sin förmåga, att hos programvara finna felmod av mer systemspecifik karaktär.
När det gäller FTA noterades vikten av, att upprättat felträd inte avslutas med basnoder av typ ’Fel i programvara’. För meningsfull FTA på programvara fordras, att analysen drivs vidare mot mer systemspecifika defekter, utan att därför gå ända ned i kodstrukturen.
Analysresultatens användbarhet ökas avsevärt, om ovanstående feltyp detaljeras mot basnoder av typ ’Fel i terräng-/navigeringsdata’, ’Felaktig navigeringsberäkning/-algoritm’.
En positiv bieffekt av projektet är de faktablad, som producerats. Hit hör t ex en unik riskkällelista för programvarusystem.
Ytterligare slutsatser ges i mötesanteckningar *(se Ant_M20_07)*. OH-material till detta möte finns länkat till motsvarande dagordning *(se Dagordn_SESAM_jan2007)*.

II:7 Sjunde analysmötet (STAMP/STPA-prov)
Ingångsmaterial:
34) ‘A notation supporting a Systems-Theoretic Hazard Analysis Technique’ *(STPA_Notation)*
35) ‘A new Approach to System Safety Engineering’ (**310 s)** (*http://sunnyday.mit.edu/book2.pdf).*
36) Leveson’s STPA/STAMP-tutorial (**56 Mb**) *(http://sunnyday.mit.edu/issc).*
37) STAMP/STPA-metoden (*STPA-STAMP_fakta*)
38) STPA-mall (*STPA-mall*)
39) Allmän beskrivning av farthållarsystem *(http://auto.howstuffworks.com/cruise-control.htm).*
40) ‘Adaptive Cruise Control, System Overview’

(http://sunnyday.mit.edu/safety-club/workshop5/Adaptive_Cruise_Control_Sys_Overview.pdf)

41) Designbeskrivningar CCC/ACC *(Cruise Control)*

De traditionella riskkälleanalysmetoder, som hittills varit förhärskande inom systemsäkerhetsmetodiken är inriktade mot att analysera de händelsekedjor, vilka föregår en olycka (FTA, ETA), de informationsflöden, som är involverade i ett olycksförlopp (HAZOP) eller de brister hos ingående komponenter, vilka kan utgöra hot mot systemsäkerheten (FMECA). Resultaten från dessa säkerhetsanalyser används bl a för att få fram villkor o situationer att undvika samt för att identifiera säkerhetskrav / designrestriktioner specifika för aktuellt system (till skillnad från generella säkerhetskrav giltiga för programvarusystem i allmänhet, se t ex H ProgSäk under *http://sesam.smart-lab : Arbetsgrupper: Programvarusäkerhet: Publikationer*).

STPA/STAMP är en relativt ny teknik, som studerar o modellerar hela systemet –framför allt de **styrmekanismer** (regler*loop*ar för styrning o återmatning)– som varje nivå i en systemhierarki utöver på närmast underliggande nivå. Olycksförlopp ses som en följd av bristande säkerhetsrestriktioner. Det gäller därför att identifiera dessa m h a de modeller som upprättats över system o organisation, snarare än att studera linjära o diskreta händelsekedjor, vilka kan missa emergenta egenskaper (samverkanseffekter) i de komplexa o dynamiska processer som har betydelse för utfallet av ett förlopp och som kan ha sin upprinnelse i såväl system–systemomgivning–driftsförhållanden som i organisatoriska–administrativa rutiner o regelverk för konstruktion o drift samt personal verksam i dessa. De riskkällor som identifieras m h a STPA/STAMP kommer därvid att avslöja andra typer av riskkällor och följdaktligen resultera i andra typer av åtgärder / designförändringar än de klassiska systemsäkerhetsmetoderna. Metoden kan därför ses som ett komplement till övriga analysmetoder, även om vissa angreppssätt är gemensamma (t ex är de nyckelord som leder en HAZOP-analys användbara även här).
STAMP är en grundorsaksanalys (*root cause analysis*) medan STPA är en riskkälleanalys baserad på STAMP avsedd att användas parallellt med designutvecklingen. Det finns tre olika varianter på hur säkerhetsmodellering m h a STAMP kan utföras: Dels m a p statiska (dvs tidsstabila) resp dynamiska (tidsföränderliga) styrstrukturer, dels m a p dynamiska förändringsprocesser (säkerhetsdegenererande förändringar i driftsatta system).

Sökandet efter ett tillämpningsexempel (allmänt känt av samtliga, utan att därför ingå i någon gruppmedlems yrkesmässiga verksamhet) hade pågått en längre tid. Ett ytterligare önskemål var –med tanke på gruppens begränsade utrymme för förberedelser– att finna ett inte alltför komplicerat system. Valet föll, efter förslag från prof. Nancy Leveson (metodens upphovsman), på ett adaptivt farthållningssystem.

Efter presentation av metod (34-38 ovan) samt tillämpning (39-41) inleddes en statisk STPA-analys utgående från främst bild 13 i OH-materialet (41) m h a det faktablad o den mall (37-38), som framtagits inför provningen. Inledande moment –att identifiera systemets generella krav samt omgivningsrestriktioner– kan inte sägas vara specifik för metoden, och genomfördes därför tämligen summariskt. Att identifiera riskkällor, bidragande faktorer samt motsvarande säkerhetsrestriktioner är också gemensam med övriga analysmetoder. Därefter påbörjades metodens mer specifika angreppssätt: att för varje komponent i systemet identifiera dess åtagande tillsammans med eventuella

bristfälliga styrmekanismer. Listade typexempel på brister i styrningens design resp verkställighet nyttjades därvid (jfr HAZOP:s frågeord). Några kompletteringar beträffande missade styr*loop*ar identifierades (’*Alive*’-bekräftelser, hårdkopplad bromssignal, styrkommando till radar). Trots att tillämpningen var relativt okomplicerad, fann gruppen det svårt att utan tillgång till mer detaljerade beskrivningar hitta ytterligare missar.
För att få en mer rättvisande bild av STPA/STAMP:s verkliga styrkor o särdrag togs beslutet att vid nästa analysmöte fortsätta provningen på ett nytt, mer uttalat system-av-system. Denna gång med STAMP som haveriutredningsteknik (jfr *Ant_M21_07*).

Resultat:
42) STPA-rapport enl mall 38 (*STPA-ACC*).

III Orsaksanalys *(Root cause analysis)*
III:1 Åttonde analysmötet (STAMP-prov)
Ingångsmaterial:
43) STAMP/STPA-metoden, v. 1.2 (*STPA-STAMP_fakta_1-2)*
44) STPA-mall, v. 1.2 (*STPA-mall_1-2)*
45) STAMP-översikt *(IG#22_STAMP.ppt)*
46) ’Flygkollisionen över Überlingen – Rapportutdrag’ *(Extract_2_Uberlingen.doc)*
47) ’Investigation Report’, BFU (114 s) *(BFU_Report_02_AX001-2_Uberlingen_Report.pdf)*
48) ’Flygolyckan satte djupa spår hos flygledare i Malmö’ *(Urklipp_LFVtidn-nr8-07.pdf)*
49) Grafer (’Timeline last 15 minutes’, ’Basic Control Structure’, ’Control Structure ACC Zurich’, ‘Orsak&Verkan) *(Uberlingen_STAMP-ansats_utg94.ppt)*

Förutom uppdaterade versioner av tidigare faktablad o översikt över STAMP (43-45 ovan), innehöll ingångsmaterialet beskrivningar av det nya tillämpningsexemplet: flygkollisionen juli 2002 över Überlingen (46-49). Speciellt framtaget inför analysmötet var grafer (49) över
(a) händelseförloppets sista kvart (49: OH 1),
(b) grundläggande styrmekanismer för ATC:er-flygoperatörer- piloter-stödsystem (OH 2)
(c) detaljerad styrstrukturen mellan ATC Zürich’s roller o dess stödsystem (OH 3) samt
(d) bakomliggande faktorer (49: OH 4). Dessa utgjordes bl a av:

(d1) En sedan länge accepterad praxis (i strid mot fastlagda procedurer) beträffande arbetsfördelning under nattpass (SMOP, *Single Man Operational Procedures*: 1 person på 5 roller),
(d2) Sektoromläggning av de övre luftrummen aktuell natt med gängse stödsystem ersatta av funktionellt mindre potenta *back-up*-system,
(d3) Bristande kunskap hos berörda ATCer om sektoromläggningen o dess konsekvenser,
(d4) Distraherande trafikledning över annan frekvens o konsol av försenat flyg för landning i grannsektorn (ARFA) kombinerat med
(d5) Kontaktproblem p g a oväntat fel i *by-pass*-telefon samt
(d6) Trafikledning egen sektor av transitflyg på kollisionskurs i övre luftrummet (SUED).
(d7) Missade larm/info p g a radiokommunikation över två olika frekvenser o konsoler.
(d8) Trafikledarens ovetskap om givna TCAS-instruktioner i resp transitplan.
(d9) Avvikande praxis i ena planet beträffande prioritering vid motstridiga råd från TCAS o trafikledare.

Analysen inleddes med en ny snabbgenomgång av STAMP (45) och i synnerhet dess 3:e variant (tidigare ej provad) – en modellering av de dynamiska förändringsprocesser ett driftsatt system kan utsättas för (45: OH 10). Flera olika typer av återmatningsloopar ingår i denna variant (positiv, negativ resp fördröjd, se 45: OH 11-13).

Händelseförloppet summerades m h a utdelat bildmaterial (49), där *'Timeline-'*, samt 'Orsak&Verkan'-bilderna visade sig speciellt användbara, utan att i egentlig mening utgöra ingredienser i STAMP-metodiken.

Första insatsen blev att förtydliga de **grundläggande styrstrukturer**na i 49:OH 2-3 på en mer övergripande nivå. Resultatet av detta blev 50:OH 1 med (inter)nationella myndigheter, organisationer, centraler, operatörer, flygplanstillverkare o stödsystem (främst TCAS) som översiktliga komponenter. Redan här kunde konstateras, att en mängd **styrmekanismer o *feed-back-loopar* saknas**, bl a:

- Samordnade rekommendationer från normgivande internationella organisationer (ICAO, EUROCAE), nationella luftfartsmyndigheter och TCAS-leveratören beträffande handhavande (t e x vid motstridiga råd från TCAS o ATCO, se d9).
- *Feed-back* till ATCO ang de råd TCAS ger i resp plan (se d8). Den senare TCAS II version 7 realiserar denna möjlighet.
- *Feed-back* på radarmonitorerna ang verklig flygplanposition i förhållande till färdplan (sektoromläggning: ATC-stödsystem enbart i *'fall-back mode'*, se d2).
- *Feed-back* till TCAS huruvida TCAS-råd (RA: *Resolution Advisory*) verkligen åtlytts.

P g a den korta tid, som stod till buds för analysmötet avbröts arbetet på denna STAMP-variant (***Static control structure***, jfr 45:OH 8) till förmån för prov av nästa. Givetvis kan den resulterande grafen förbättras ytterligare, förslagsvis genom inlägg av:

(i) ATC Zurich's stödsystem (*'Radar Data Processing System'*) som en översiktlig komponent mellan *'Radar'* o *'Radar display'* (med 48:OH 3s mer detaljerade struktur o uppdelning i SYCO, RCMS, ADAPT etc som en förfining av denna).

(ii) Det tekniska *team*et tillgängligt vid problem med ATC's stödsystem (främst SYMA).

Därigenom hade t ex missad info till CoC resp ATC-operatörer (se d3) samt *feed-back* från dessa till *team*et kunnat illustreras. Exempel på sådan *feed-back* hade kunnat vara, en av CoC utförd riskanalys över planerad sektoromläggning (med bl a d2 o d7 som typiska riskkällor) samt de risklindrande anvisningar o åtgärder, som CoC kunnat härleda ur sådan analys (t ex att stöd*team*et skall utföra tillförlitlighetskontroll av *stand-by-*/ reserv-system: se d5, att CoC förbjuder SMOP i utbyte mot annan lågbemanningsmodell: se d1,d4,d7).

STAMPs ***Dynamic control structure*** (jfr 45:OH 9) modellerar styrmekanismernas **förändringar** över tiden. Efterföljande övning inriktades mot att avspegla nattpassets hopslagning av 5 roller på en ATCO (se d1). Här blev snabbt uppenbart, att de olika bekräftelser o kontroller, som de olika trafikledarna utför av varandras information o uppgifter, inte kan åstadkommas med en enda person. All redundans är eliminerad. Avlastningar vid plötsligt tillkommande uppgifter blir också omöjliga. Även om vilande personer befinner sig i angränsande lokaler, finns inte tid för dessa att hinna sätta sig in i aktuell situation. Resultatet av denna övning återfinns i 50:OH 2. Även denna graf går att förbättra. I detta läge gällde dock, att hinna prova nästa modelleringsvariant, vilken i stället fokuserar **orsakerna** till de med tiden införda förändringarna i styrstrukturen.

STAMPs ***Behavioral dynamics*** (jfr 45:OH 10) söker efter **bakomliggande faktorer** till de säkerhetspåverkande förändringar, som systemets hantering o drift har genomgått. Relationspilar markerade med '+' eller '–' vid pilhuvudet används, för att ange huruvida källan bidrar till en ökning resp minskning av destinationsfaktorn. Därför riktas i OH 10 en minuspil från källan *'Budget cuts towards safety'* till destinationen *'System safety efforts'*.

En dynamisk modellering av införda förändringar inom ATC Zürich påbörjades.
Intrycket från OH 10 o övriga exempel, som Leveson presenterat, är, att resulterande grafer blir relativt likartade. Typiska ingredienser är: *'Complacency', 'Perceived safety', 'External pressure', 'Performance pressure', 'Expectations', 'Oversight', 'Budget', 'System safety effort', 'Priority of safety programs'* etc.
Angreppssättet blev därför, att utifrån Dulacs dynamiska modell över Columbia-olyckan se

1) vilka av ovanstående faktorer som är **allmängiltiga** o därför kan användas rakt av,
2) vilka som eventuellt behöver **tolkas mer specifikt** för aktuell tillämpning (i analogi med HAZOP:s anpassning av generella nyckelord till aktuell parametertyp, 15-16),
3) vilka, som är **tillämpningsunika för Dulacs fall** och därför bör utgå/ bytas ut.
4) vilka faktorer som är **tillämpningsunika i aktuellt fall** och därför tillkommer.

På så vis omtolkades *'Mission success'* till *'Separation maintained'* och *'Performance pressure'* till *'Performance pressure on ATCO'* , medan *'Launch delays', Launch rate'* ersattes med *'# Controlled flights', '#Flights / person'* etc.
Dessutom identifierades faktorer unika för Überlingen-olyckan: *'Procedure shortcuts', 'Sector change', 'Comfort', 'Authorities'* resp *'Management's priority of safety', 'Shortage of ATCOs', 'Accidents', 'CoC', '#Active persons/night'* etc (se 50:OH 3). Bra input här var Orsak&Verkan-grafen (49:OH 4).

Även detta resultat kan diskuteras ytterligare o förbättras, förslagsvis genom att låta

- *'Shortage of ATCOs'* peka även på *'Performance pressure'*
- *'Comfort'* placeras vid *'Procedure shortcuts'* o förses med pil från *'Complacency'*
- *'Comfort'* då i stället peka på *'# Inactive persons/night'*
- *Comfort'* peka på *'#Flights / person'* vilken i sin tur får peka på *'Performance pressure'*
- *'CoC'* :s inverkan på *'Comfort'* bli föremål för vidare utredning
- *'Delayed approach'* utgöra en ny faktor vid *'External pressure'*
- *'Mission success'* kompletteras med den mängd utfall som innebär lyckat genomförande av ATCOs säkerhetskritiska uppgifter (förslagsvis *'Safe approach/ departure/ climb/ descent/ transit service*', *'Situation awareness of assigned airspace'* etc*).* En negering av *'Mission success'* motsvarar p s s möjliga olycksutfall.

I en normalsituation, där SMOP ej tillämpats, skulle faktorn *'Performance pressure'* inte enbart avse en ATCO utan kunna ingå med en faktor per roll (dvs *'Performance pressure on RP', '…on RE_SUED', '…on RE_ARFA'*, osv). Varje roll skulle då kopplas till motsv utfall för *'Mission success'* (t ex *'Safe approach service for SUED', 'Safe…for ARFA'*).
Frågan är hur långt denna modelleringsvariant går att driva, utan att röra till strukturen med för många parallella snurror o korsande relationspilar. Även om det inte framgår av metodbeskrivningarna, kanske en nedbrytning av en översiktlig graf i fler detaljgrafer kan vara en framkomlig väg.

Vad skall den resulterande grafen till denna STAMP-variant användas till?
Förhoppningsvis kan den åskådliggöra de riskkällor, som en viss verksamhet kommit att inhysa, tillika med de utlösande faktorer, som kan föreligga för dessa.
För Überlingen-fallet: regelbrott (SMOP), ofullständig överlämning till avlösande arbets*team*, bristfällig informationsinhämtning hos övertagande *team* – men även – CoCs obefintliga riskanalyser av införda resursinskränkningar (personal, ekonomi, stödsystem), av avvikande praxis o av planerade systemomläggningar samt uppföljning av efterlevnad.
Relevanta frågeställningar för att bemöta dessa riskkällor är:

- Vilka styrmedel (regler, föreskrifter, procedurer, informationskanaler) kan behöva kompletteras?
- Vilken *feed-back* krävs för att försäkra sig om deras efterlevnad?

Uppslag till kompletterande styråtgärder o bekräftande återmatningar kan härledas, t ex genom översyn av vilka genvägar som utvecklats över tiden till de procedurer som föreskrivs samt analys av vilka effekter dessa kan ha på systemsäkerheten. Likaså kan en kartläggning av ev. resursförändringar (ekonomiska/ personella) samt den verkan dessa har på rådande arbetssituation o systemsäkerhet visa på behovet av ytterligare säkerhets-befrämjande åtgärder.
En sammanfattning av resultatet skulle kunna ges i form av en tabell över ’Riskkälla’ samt matchande ’Säkerhetskrav’ o ’Säkerhetsrestriktioner’ (i analogi med 45:OH 3). Komplexiteten i trafikledningsarbetet gör dock, att det torde behövas flera säkerhetskrav o säkerhetsrestriktioner för att bemöta en enda riskkälla.

Därmed avslutades STAMP-provning för denna gång. Detta kommer att återupptas på samma material vid nästa möte. Till dess kommer kompletterande information att efterfrågas från Leveson, t ex beträffande hur resultatet av *Behavioral Dynamics*-modelleringen är tänkt att användas, vilka hjälpmedel som finns framtagna etc. (jfr *Ant_M22_08*).

Resultat:
Syftet med dessa prov är, som tidigare påpekats, att med de insatser som står till buds, utröna hur pass användbar STAMP-tekniken är. Vid ovanstående analysmöte utfördes provet på ett verkligt fall, där ’facit’ nu torde föreligga. Trots detta, har analysgruppen nöjt sig med resultat o slutsatser, som ej till fullo avspeglar verkligheten. Även om de analysresultat, som sammanfattats i nedanstående grafer, avspeglar de väsentligaste faktorerna, kan de därför ej vara helt kompletta, felfria eller konsistenta.

50) Grafer (*’Operators: STAMP’s Static Control Structure’*,
’Roles within ATC Zurich: STAMP’s Static/dynamic Control Structures’
‘STAMP’s Behavioural Dynamics’) *(Uberlingen_analysresultat_1.ppt)*

III:2 Nionde analysmötet (STAMP-prov)
Ingångsmaterial:
Samma material som låg till grund för närmast föregående analysmöte (se III:1) samt resultat från detta, dock i huvudsak filer enl punkt 47 samt 50.

Mötet inleddes med en resumé över STAMP/STPA-metoderna, skillnaden mellan dessa (jfr II:7), Überlingenolyckans händelseförlopp/ aktörer / bakomliggande faktorer, tidigare

analysresultat samt frågor översända till prof Nancy Leveson inför hennes stundande Sverigeseminarium.

Föregående resultat analyserades vidare, med följande förändringar som följd:

50: OH1: Tillägg i ***'Static Control Structure'*** av den tekniska supportgruppen (representerad av *SYMA*) samt den uteblivna kommunikationsslingan (*briefings, consultation*) mellan denna o tjänstgörande *ATC*-operatör(er). Bilden kompletterades även med den obefintliga interaktionen mellan *SYMA* o kvalitetscentret (*CoC*) beträffande planer för –samt accept av– arbetet med sektoromläggningen aktuell natt.
Ytterligare ett flöde som förtydligades var den bristande samordningen i styrningen från de normgivande (inter)nationella organisationerna (typ *ICAO*, *EuroControl*) samt den ytterst långsamma o indirekta återmatning via flera nivåer, som sker till dessa samt till stödsystemstillverkare (t ex TCAS-fabrikanten) från berörda flygledarcentraler, flygoperatörer, etc. Ett sätt att poängtera segheten i denna *feedback* blev, att lägga in en 'balanseringsloop med fördröjd verkan', en symbol som egentligen hör till STAMP-varianten *'Behavioural Dynamics'*.

50: OH2: Även grafen ***'Dynamic Control Structure'*** kompletterades med den bortrationaliserade kommunikationsslingan mellan SYMA o ATCO. I övrigt följer ju bilden, som den ritats, ej mallen för hur en dynamisk styrstruktur skall illustreras. Ett annat sätt att lösa detta framgår nedan (se 53: OH 4-6).

50: OH3: Diverse omstruktureringar gjordes av grafen ***'Behavioural Dynamics'***: Den bakomliggande faktorn *'Comfort*' flyttades – ett förtydligande av att den är en följd av *'Complacency '* – och samtidigt utgör en ingångsfaktor till *' #Active persons/ night'*, vilken i sin tur blir en av ingångsfaktorerna till *'Performance pressure'*.
'External pressure' (den tidigare identifierade ingångsfaktorn till *'Performance pressure'*) exemplifierades med ytterligare bakgrundsfaktorer: Förutom *'Sector change'* även *'Unanticipated approach'* (det försenade flyget till Friedrichshafen) samt *'Technical failure*' (den fallerade *back-up*-telefonen).
Vidare omtolkades *'Mission success'* (riskkällans motsats) till *'Separation maintained'*, *'Safe approach'*, *'Situation awareness'* osv (här går givetvis att identifiera fler exempel på lyckade uppdrag för de olika rollerna RE, RP CA, DL osv).
Ytterligare brister hos *CoC* (förutom den uteblivna riskanalysen m a p *SMOP*) lämnades därhän. I stället ägnades kvarstående tid till att från dessa 3 grafer identifiera riskkällor samt matchande säkerhetskrav, designrestriktioner o risklindringar (endast 1:a o sista kolumn i motsvarande tabell fylldes dock i, se 51: OH 4).
I och med detta avslutades aktuell STAMP-analys av Überlingenolyckan (jfr *Ant_M23_08*).

Resultat:
51) Grafer & tabell (*'Operators: STAMP's Static Control Structure'*,
'Roles within ATC Zurich: STAMP's Dynamic Control Structure',
'STAMP's Behavioural Dynamics',
'Riskkällor-Säkerhetskrav- Säkerhetsrestriktioner-Risklindringar')
(Uberlingen_analysresultat_M23.ppt)

III:3 Leveson-seminariet 2008-05-21 – 22

Ingångsmaterial:
52) STAMP-*tutorial*, 173 bilder *(Sweden-day1-print.ppt)*

Som en del i programvarusäkerhetsgruppens verksamhet hade prof Nancy G Leveson inbjudits att hålla en tvådagarskurs på temat *'System Safety in Software-Intensive Systems'*. Första dagen ägnades åt STAMP/STPA, och Leveson, som i vanlig ordning kryddade sin framställning med illustrativa fallbeskrivningar, hade lovat inkludera Überlingen-olyckan bland dessa.
Den som studerar Levesons analysresultat för detta fall (se 53: OH 2-9) finner, att studiegruppens grafer 49: OH1 samt 51: OH1-2 elegant sammanfattas av Levesons bild 53: OH3. Vidare, att Levesonbilderna 53: OH 4-6 utgör bra exempel på hur STAMP-varianten *'Dynamic Control Structure'* är tänkt att användas, för att uttrycka förändringar över tiden. Lösningen är att använda en bild per förändringsorsak, i detta fall: Degradering p g a olämplig praxis samt p g a sektoriseringsomläggning resp onormala omständigheter.

Resultat:
53) STAMP på Überlingen (extrakt av bild 1; 77; 89-96 ur ingångmaterial 52)
(Uberlingen_enl_Leveson.ppt)

Dessa kursdagar fick definiera avslut, inte bara STAMP/STPA-provningen, utan också för hela projektet 'Säkanalysmetoder' – ett mikroprojekt som i och med sin inledning i februari 2006 kom att sträcka sig över en tvåårsperiod.

Dokumenthistorik

Version	Datum	Beskrivning	Sammanställt av
1.0	06-08-18	Resultat från analysmöte 1-3.	Inga-Lill Bratteby-Ribbing
1.1	06-09-07	Resultat efter analysmöte 4.	Inga-Lill Bratteby-Ribbing
1.2	06-10-09	Resultat efter analysmöte 5.	Inga-Lill Bratteby-Ribbing
1.3	06-12-15	Resultat efter analysmöte 6.	Inga-Lill Bratteby-Ribbing
1.4	06-12-20	Uppdatering med länkfil till punkt 33 (2 st SW-FTA)	Inga-Lill Bratteby-Ribbing
1.5	07-02-28	Sammanfattning av analysmöte 1-6.	Inga-Lill Bratteby-Ribbing
1.6	07-10-22	Resultat efter analysmöte 7.	Inga-Lill Bratteby-Ribbing
1.7	07-11-05	Komplettering av analysresultat möte 7.	Inga-Lill Bratteby-Ribbing
1.8	08-03-07	Resultat efter analysmöte 8.	Inga-Lill Bratteby-Ribbing
1.9	08-05-30	Resultat efter analysmöte 9 + Leveson-seminarium.	Inga-Lill Bratteby-Ribbing