

3651.9892 -605.8146

3660.7362 -0.8520

ECAC.sls

ECAC_ooo_600 + ECAC ooo 600

9.6 Daily Flight List

The daily flight list used in NEST is the same daily flight list used by the DDR2. It's defined as all flights crossing the [Extended IFPZ Area](#) for at least one second on the given day. The size of the extended IFPZ Area ensures that it captures all the necessary flights to allow calculating realistic traffic counts that reflect the actual counts and flight lists observed in the airspace on the given day, and also that the counts for all airspaces are mutually consistent.

9.7 File Formats

This topic provides the file format specifications for all text-based data files used either by NEST or by the external modules delivered with NEST. All of these files are text files (ASCII) that can be edited with a standard text editor.

NEST dataset text files

A subset of these files are used to define the full NEST dataset. These files are used by the [NEST file builder](#) to create the AIRAC.nest binary data files. This subset of dataset files is also used in the [import](#) and [export](#) contexts.

File naming conventions

The naming conventions for dataset text files are outlined here.

Lat / Lon coordinate conventions

In most NEST files coordinates are described in Decimal-Minutes. It is easy to convert between decimal minutes and degrees, minutes and seconds using Excel (see below).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	3.063,49	249,85		51	N	03	29		04	E	09	50		51 N 3 29	4 E 9 50
2	3.398,15	-1.024,29		56	N	38	09		17	W	04	17		56 N 38 9	17 W 4 17
3	-681,25	1.010,47		11	S	21	15		18	E	50	28		11 S 21 15	16 E 50 28
4	2.956,16	13.405,13		49	N	16	09		223	E	25	07		49 N 16 9	223 E 25 7
5	-1.476,28	-5.443,79		24	S	36	16		90	W	43	47		24 S 36 16	90 W 43 47
6	-167,98	6.092,54		02	S	47	58		101	E	32	32		2 S 47 58	101 E 32 32

Decimal Minutes to Degrees, Minutes & Seconds

Insert function (fx);

- Column D fx = INT(ABS(A1)/60)
- Column E fx =IF(A1<0;"S";"N")
- Column F fx = INT(MOD(ABS(A1);60))
- Column G fx = INT((ABS(A1)-INT(ABS(A1)))*60)
- Column I fx = INT(ABS(B1)/60)
- Column J fx = IF(B1<0;"W";"E")
- Column K fx = INT(MOD(ABS(B1);60))
- Column L fx = INT((ABS(B1)-INT(ABS(B1)))*60)
- Column N fx =D1&" "&E1&" "&F1&" "&G1&
- Column O fx =I1&" "&J1&" "&K1&" "&L1&

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	59	51	04	N	12	35	40	E		3591,066667	755,6666667			
2	59	46	36	N	13	25	42	E			3586,6	805,7		
3	59	37	59	N	15	18	05	E		3577,983333	918,0833333			
4	59	31	22	N	15	18	39	E		3571,366667		918,85		
5	59	25	36	N	15	22	54	E			3565,6	922,9		
6	58	36	46	N	13	15	23	E		3516,766667	795,3833333			

Degrees, Minutes & Seconds to Decimal Minutes

Insert function (fx);

- Column J fx =(A3*60)+B3+(C3/60)
- Column K fx =(E3*60)+F3+(G3/60)

9.7.1 aco - Airspace Colour file

Airspace COlour	
extensio n	aco
origin	SAAM
separato r	blank

sort	no sort
comment	Describe several named display features and colour for a set of 3D airblocks

#	Field	Type	Size	Comment
1	version	num	~	version of format (=1)
2	name of airblock	char	~	name can be: sector:airblock
3	airblock low level	num	3	expressed in Flight Level. CAUTION: padded with o !!!
4	airblock top level	num	3	expressed in Flight Level. CAUTION: padded with o !!!
5	number of colours	num	~	give the number of following different colours associated to the airblock
6	version of colour	num	~	version of colour (=1)
7	name of colour	num	~	identify the name of the colour (like "overload"). "?" is the default colour name
8	feature of colour	num	~	indicate the way to display the colour (transparent, top solid, ...), see below
9	colour flag	num	1	1 mean the colour is described here, 0 means no colour (comes from ARE or RANDOM)
10	colour type coding	num	1	"C"
11	transparency of colour	num	~	min 0, max 255
12	red of colour	num	~	min 0, max 255
13	green of colour	num	~	min 0, max 255
14	blue of colour	num	~	min 0, max 255

Field 8, feature of colour code:

indicates if the features for this airspace come from the TDV line (common to all airspace of the file) or from the airspace itself. Each feature has 3 values (except flight level):

0 means the feature comes from TDV line

1 means the feature comes from the airspace itself and is set to ON

2 means the feature comes from the airspace itself and is set to OFF

bit 0 and 1: top flag

bit 2 and 3: lighting flag

bit 4 and 5: transparent flag

bit 6: bottom level flag (0 means come from TDV if possible (!=999), 1 means come from the airspace header field 5)

bit 8: top level flag (0 means come from TDV if possible (!=999), 1 means come from the airspace header field 6)

bit 10: label flag (0 means come from TDV if possible (!=9), 1 means label is displayed, 2 label is not displayed (see also content of the label)

bit 12: soft lock flag (0 means come from TDV if possible (see lower/upper case of the airspace file name in TDV volume format), 1 mean soft lock, 2 means no soft lock (normal)

Examples:

192: means only bottom and top level are set from the airspace itself, the rest comes from TDV line

208: means bottom and top level and transparent ON for the airspace itself, rest is from TDV

224: means bottom and top level and transparent OFF for the airspace itself, rest is from TDV

Example:

```
1 LIPPMB1:A000165 000 085 1 1 ? 192 1 C 64 214 229 202
1 LIPPMB1:A000166 000 085 1 1 ? 192 1 C 64 214 229 202
1 LIPPMB1:A000167 000 085 1 1 ? 192 1 C 64 214 229 202
1 LIPPMB1:A000168 085 135 1 1 ? 192 1 C 64 214 229 202
1 LIPPMB1:A000169 085 135 1 1 ? 192 1 C 64 214 229 202
1 LIPPMB1:A000170 135 175 1 1 ? 192 1 C 64 214 229 202
1 LIPPMB1:A000171 135 175 1 1 ? 192 1 C 64 214 229 202
```

9.7.2 adinfo - AirportCapaCurfew

Airport Capacity and Curfew File

Filename: "FFF_yyyyMM.adinfo" (FFF: STATFOR "MTF" / "STF" forecast type, yyyy: year using four characters, MM: month using two characters)

Function: Defines airport capacities and curfews

[First Line](#)

Field	Value
1	“Airport” single airport ID header
2	“Year” year header
3	“Open Hour” airport opening hour header
4	“Close Hour” airport closing hour header

5	“Capacity” airport capacity header
---	------------------------------------

Subsequent Lines

Field	Value
1	single airport ID
2	year of data, airport capacities and curfews for future years represent projected data for the airport
3	airport opening hour
4	airport closing hour
5	airport capacity

Example:

Airport;Year;Open Hour;Close Hour;Capacity

EGLL;2009;3;23;89

EGLL;2010;3;23;90

EGLL;2011;3;23;90

EGLL;2012;3;23;90

EGLL;2013;3;23;90

EGLL;2014;3;23;90

EGLL;2015;3;23;90

EGLM;2009;0;24;999

EGLM;2010;0;24;999

EGLM;2011;0;24;999

EGLM;2012;0;24;999

EGLM;2013;0;24;999

EGLM;2014;0;24;999

EGLM;2015;0;24;999

9.7.3 Aircraft Performance Profile Map

Aircraft Performance Profile Map				
#	Field	Type	Size	Comment
First line				
1	file format id			Version of the file format, formatted as n.n
2	generation date	date		Date at which the file was generated, formatted as dd/mm/yyyy.
3	comment	char	till eol	A comment field. Normally contains indications on the origins of the data.
Second line				
#	Field	Type	Size	Comment
1	map count	num		Number of mappings in the file.
Mapping Entry				
#	Field	Type	Size	Comment
1	aircraft id	char		An aircraft id code followed by SR (or LR if multiple profiles are given for a single aircraft type). SR means Short Range, LR Long Range
2	performance index	num		Base 1 index in the performance data file. Indicates which performance profile family is associated for this aircraft type.

Example (1 flight 10 first lines of body):

```
3,3 20-08-2003 Based on BADA 3.4 + CFMU V31
2999
32SR 161
35SR 161
37SR 161
39SR 161
A109SR 162
A10ASR 163
A10LR 118
A10SR 117
A119SR 162
```

9.7.4 Aircraft Performance Profile Data

Aircraft Performance Profile Data	
extension	dat.txt

origin	SAAM performance data file		
separator	whitespace		
sort			
comment	For aircraft profile processing, used with "Aircraft Performance Map". Contains Family performance. Note: reference to performance is made via index number (first perf has index 1...) see Aircraft Performance Map		

#	Field	Type	Size	Comment
First line				
1	file format id			Version of the file format, formatted as n.n
2	generation date	date		Date at which the file was generated, formatted as dd/mm/yyyy.
3	comment	char	till eol	A comment field. Normally contains indications on the origins of the data.
Second line				
1	profile count	num		Number of profiles in the file.
HEADER				
1	name	char	4	NOT an ID, facilitate searches (generally first aircraft name having this perf)
2	nb line	num		total number of following body lines
3	nb_climb_line	num		total number of following climbing lines
4	nb_line_descent	num		total number of following descend lines
5	optimum_FL	num		optimum' flight level (never higher than max flight level given by the table)
BODY				
1	length	num		length expressed in NM from departure or to arrival (goes from 0 to N, step 1)
2	level climb	num		climbing level at distance 'length' from departure
3	level descent	num		descent level at distance 'length' from arrival
4	time climb	float		time in minute when climbing to reach NM length from departure
5	time descent	float		time in minute when descending from top of descent to arrival
Cruise speeds				
1	size	num		Number of entries
2 +2 n	level	num		FL
3 +2 n	cruise speed	num		speed in knots at 'level'

Example (1 flight 10 first lines of body):

3, 3 20-08-2003 Based on BADA 3.4 + CFMU V31

```

183
A306___.PTF 168 167 122 370
0   0   0   0   23.2341
1   8   4   0.3   22.9341
2   15  7   0.6   22.6341
3   23  11  0.9   22.3341
4   30  14  1.2   22.0341
5   38  18  1.5   21.7341
6   45  22  1.8   21.4341
7   53  25  2.1   21.1341
25 0 0 5 0 10 0 15 0 20 0 30 230 40 233 60 272 80 280 100 289 120 297 140 306 160
389 180 401 200
-> 413 220 425 240 438 260 452 280 466 290 468 310 464 330 459 350 455 370 453
390 453

```

9.7.5 ALL_FT+

ALL_FT+ File

Filename: "yyyyMMdd.ALL_FT+" (yyyy: year, MM:month using two characters, dd: day using two characters)

Function: Defines a list of flight trajectories for a given day

This version refers to the V4, which is currently the last version distributed by DDR2

Column N°	Identification	Info
0	departureAerodromeIcaoId : AerodromeIcaoId	ICAO location identifier of an AERODROME
1	arrivalAerodromeIcaoId : AerodromeIcaoId	ICAO location identifier of an AERODROME
2	aircraftId : Aircraftid	Aircraft identification
3	aircraftOperatorIcaoId : AircraftOperatorIcaoId	
4	aircraftTypeIcaoId : AircraftTypeIcaoId	
5	aobt : DateAndHms	Date and time expressed as YYYYMMDDHHMMSS
6	ifpsId : IfpsId	Internal number of a flight within the IFPS system, not unique as it is recycled every some days (see field 23)
7	iobt : DateAndHms	Date and time expressed as YYYYMMDDHHMMSS
8	originalFlightDataQuality : FlightDataQuality	[NON MFD RPL FPL PFD] Origin of the most up-to-date information (from MFD, RPL, PFD or FPL) received from the external world
9	flightDataQuality : FlightDataQuality	[NON MFD RPL FPL PFD]
10	source : Source	[UNK FPL RPL AFI MFS FNM AFP DIV]
11	exemptionReasonType : ExemptionReasonType	["NEXE" "EMER" "SERE" "HEAD" "AEAP" "MEDE" "FIRE"] Reason for which a flight is exempted

		from a regulation
12	exemptionReasonDistance : ExemptionReasonDistance	["NEXE" "LONG"]
13	lateFiler : StatusYesNo	
14	lateUpdater : StatusYesNo	
15	northAtlanticFlight : StatusYesNo	
16	cobt : DateAndHms	
17	eobt : DateAndHms	
18	lobt : DateAndHms	
19	flightState : FlightState	[NE PL PS PR SR FI FS SI TA AA CA TE]
20	previousToActivationFlightState : FlightState	[NE PL PS PR SR FI FS SI TA AA CA TE]
21	suspensionStatus : SuspensionStatus	[NS ST SM RC DC TV NR RV MS AS]
22	tactId : TactId	Unique flight Identifier
23	samCtot : DateAndHms	
24	samSent : StatusYesNo	
25	sipCtot : DateAndHms	
26	sipSent : StatusYesNo	
27	slotForced : StatusYesNo	
28	mostPenalizingRegulationId : RegulationId	
29	regulationsAffectedByNrOfInstances : NrOfInstances	
30	excludedFromNrOfInstances : NrOfInstances	
31	lastReceivedAtfmMessageTitle : AtfmMessageTitle	[DES ERR FCM FUM FLS REA RFI RJT RRN RRP SAM SIP SLC SMM SPA SRJ SRM SWM UNK]
32	lastReceivedMessageTitle : MessageTitle	[ABI ACH ACT APL ARR CHG CNL COR CRE DEP DLA ERR EST FDI FPL FSA MFS PAC RPL UNK]
33	lastSentAtfmMessageTitle : AtfmMessageTitle	[DES ERR FCM FUM FLS REA RFI RJT RRN RRP SAM SIP SLC SMM SPA SRJ SRM SWM UNK]
34	manualExemptionReason : ManualExemptionReason	[N S R]
35	sensitiveFlight : StatusYesNo	
36	readyForImprovement : StatusYesNo	
37	readyToDepart : StatusYesNo	
38	revisedTaxiTime : TimeElapsedInSec	[000000 ... 999999]
39	tis : TimeElapsedInSec	
40	trs : TimeElapsedInSec	
41	toBeSentSlotMessageTitle : MessageTitle	[ABI ACH ACT APL ARR CHG CNL COR CRE DEP DLA ERR EST FDI FPL FSA]

		[MFS PAC RPL UNK]
42	toBeSentProposalMessageTitle : MessageTitle	[ABI ACH ACT APL ARR CHG CNL COR CRE DEP DLA ERR EST FDI FPL FSA MFS PAC RPL UNK]
43	lastSentSlotMessageTitle : MessageTitle	[ABI ACH ACT APL ARR CHG CNL COR CRE DEP DLA ERR EST FDI FPL FSA MFS PAC RPL UNK]
44	lastSentProposalMessageTitle : MessageTitle	[ABI ACH ACT APL ARR CHG CNL COR CRE DEP DLA ERR EST FDI FPL FSA MFS PAC RPL UNK]
45	lastSentSlotMessage : DateAndHms	Date and time expressed as YYYYMMDDHHMMSS
46	lastSentProposalMessage : DateAndHms	Date and time expressed as YYYYMMDDHHMMSS
47	flightCountOption : CountOption	[P N]
48	normalFlightTactId : TactId	Unique identifier associated to a flight
49	proposalFlightTactId : TactId	Unique identifier associated to a flight
50	operatingAircraftOperatorIcaoId : AircraftOperatorIcaoId	
51	reroutingWhy : ReroutingWhy	[N T M C A O]
52	reroutedFlightState : ReroutedFlightState	[P E T R V N]
53	runwayVisualRange : RunwayVisualRange]	[001,998] = actual RVR value - Minimum visible range in meters for a flight to land
54	numberIgnoredErrors : NumberIFPSIgnoredErrors	Describes the number of errors that are ignored by the IFPS operator on the FTFM
55	arcAddrSource : ArcAddrSource	[N M F D A C]
56	arcAddr : HexDigit[6]	[1,9] [A...F]
57	ifpsRegistrationMark : RegistrationMark	Free format comment
58	flightType : ICAOFlightType	[S N G M X]
59	aircraftEquipment : AircraftEquipmentSet	aircraftEquipment is identical to ICAO field10a
60	cdmStatus : CdmStatus	[N c e t s a]
61	cdmEarlyTtot : DateAndHms	
62	cdmAoTtot : DateAndHms	
63	cdmAtcTtot : DateAndHms	
64	cdmSequencedTtot : DateAndHms	
65	cdmTaxiTime : TimeElapsedMmmmSs	4 numeric for the minutes followed by 2 numeric for the seconds. This 'mmmmss' format is identical to the one of AccuracyValue
66	cdmOffBlockTimeDiscrepancy : StatusYesNo	

67	cdmDepartureProcedureId : AdexpTerminalProcedureId	Identifier of an ICAO or non-ICAO terminal procedure (SID or STAR). It has variable length and so is only suited for ADEXP and ALL FT formats.
68	cdmAircraftTypeId : CdmAircraftType	
69	cdmRegistrationMark : CdmRegistrationMark	
70	cdmNoSlotBefore : DateAndHms	Date and time expressed as YYYYMMDDHHMMSS. - cdmNoSlotBefore is the minimum take-off time computed by the CDM system, taking into account the Ctot sent by ETFMS. This noSlotBefore is the Ttot sent to ETFMS in a T-DPI message with dpiStatus "S". For subsequent slot recomputation, CASA is requested to search a slot from the noSlotBefore.
71	cdmDepartureStatus : DepartureStatus	[K D]
72	ftfmEetFirNrOfInstances : NrOfInstances	
73	ftfmEetFirList : AllFtEetFir[0..ftfmEetFirNrOfInstances]	
74	ftfmEetPtNrOfInstances : NrOfInstances	
75	ftfmEetPtList : AllFtEetPt[0..ftfmEetPtNrOfInstances]	
76	ftfmAiracCycleReleaseNumber : AiracCycleReleaseNumber	Numeric[4]
77	ftfmEnvBaselineNumber : EnvBaselineNumber	Numeric[6]
78	ftfmDepartureRunway : RunwayDirection	Numeric[2] followed by 'L' (for left) or 'R' (for right) or 'C' (for center) or ''
79	ftfmArrivalRunway : RunwayDirection	Numeric[2] followed by 'L' (for left) or 'R' (for right) or 'C' (for center) or ''
80	ftfmReqFlightlevelSpeedNrOfInstances : NrOfInstances	Numeric[4]
81	ftfmReqFlightlevelSpeedList : ReqFlightLevelSpeed[0 .. ftfmReqFlightlevelSpeedNrOfInstances]	a)The reqflightlevelspeed elements are separated by a Space. b)Within an reqflightlevelspeed element, the fields are separated by a ':'.
82	ftfmConsumedFuel : ConsumedFuel	[-99999999 ... 99999999]
83	ftfmRouteCharges : RouteCharges	[-99999999 ... 99999999]
84	ftfmAllFtPointNrOfInstances : NrOfInstances	
85	ftfmAllFtPointProfile : AllFtPointProfile[0 .. ftfmAllFtPointProfileNrOfInstances] Filed TACT model -1- Point Profile consistency_rules : 1.This defines the structure of ONE point profile element. The AllFtPointProfile itself contains several instances of this element. 2.Coded according to the standard rules applying to AllFt repeating groups -> a)The point profile elements are separated by a Space. b)Within a point profile element, the fields are separated by a ':';	

	<p>3) pointDistance is the distance over the trajectory measured from the first bound (usually ADEP) to the point.</p> <p>4) geoPointId shall always be present and represents the latitude and longitude of the point or vector.</p> <p>5) In case pointType = G, S or V then the value of 'point' shall be empty.</p> <p>6) In case pointType = G, S then the value of route shall be 'DCT';</p> <p>7) In case pointType = V then the value of relDist shall contain the ratio of the vector point distance from the segment start point divided by the segment length, expressed as a percentage, shall be empty otherwise.</p> <p>8) In a qfAllFtPointProfile 'point' and 'route' shall always be empty, 'pointType' shall be G or V.</p> <p>9) isVisible: Y indicates that IFR/GAT/IFPSTART applies from point to the next point, N indicates that either we are entering a VFR, OAT, IFPSTOP or STAY portion or a combination of the 4. The value N is also used in the qfAllFtPointProfile to indicate that coverage following this point (plot) is insufficient.;</p>
86	rtfmAllFtAirspaceNrOfInstances : NrOfInstances
87	rtfmAllFtAirspaceProfile : AllFtAirspaceProfile[0 .. rtfmAllFtAirspaceNrOfInstances] Filed TACT model -1- Sector Profile consistency_rules : 1.This defines the structure of ONE airspace profile element.The AllFtAirspaceProfile itself contains several instances of this element. 2.Coded according to the standard rules applying to AllFt repeating groups -> a)The airspace profile elements are separated by a Space. b)Within an airspace profile element, the fields are separated by a '!'. c)When AirspaceType is BCYL then airspacelD will indicate from which cylinder intersection and from which bound of the profile. E.g. F40 meaning the 40 Nm cylinder from the first (usually ADEP) side of the profile. L100 meaning the 100 Nm cylinder from the last side of the profile. d)AirspaceType is BCYL will be computed for the rtfm rftm ctfm qf sct srr sur AllFtCircleIntersections fields. 3) entry exitDistance is the distance over the trajectory measured from the first bound (usually ADEP) to the entry or exit of the airspace. ;
88	rtfmAllFtCircleIntersectionsNrOfInstances : NrOfInstances
89	rtfmAllFtCircleIntersections : AllFtAirspaceProfile[0 .. rtfmAllFtCircleIntersectionsNrOfInstances]
90	rtfmAiracCycleReleaseNumber : AiracCycleReleaseNumber Numeric[4]
91	rtfmEnvBaselineNumber : EnvBaselineNumber Numeric[6]
92	rtfmDepartureRunway : RunwayDirection Numeric[2] followed by 'L' (for left) or 'R' (for right) or 'C' (for center) or ''
93	rtfmArrivalRunway : RunwayDirection Numeric[2] followed by 'L' (for left) or 'R' (for right) or 'C' (for center) or ''
94	rtfmReqFlightlevelSpeedNrOfInstances : NrOfInstances
95	rtfmReqFlightlevelSpeedList : ReqFlightLevelSpeed[0 .. rtfmReqFlightlevelSpeedNrOfInstances]
96	rtfmConsumedFuel : ConsumedFuel [-99999999 ... 99999999]
97	rtfmRouteCharges : RouteCharges [-99999999 ... 99999999]
98	rtfmAllFtPointNrOfInstances : NrOfInstances
99	rtfmAllFtPointProfile : AllFtPointProfile[0 .. rtfmAllFtPointProfileNrOfInstances]
100	rtfmAllFtAirspaceNrOfInstances : NrOfInstances

101	rtfmAllFtAirspaceProfile : AllFtAirspaceProfile[0 .. rtfmAllFtAirspaceProfileNrOfInstances]	
102	rtfmAllFtCircleIntersectionsNrOfInstances : NrOfInstances	
103	rtfmAllFtCircleIntersections : AllFtAirspaceProfile[0 .. rtfmAllFtCircleIntersectionsNrOfInstances]	
104	ctfmAiracCycleReleaseNumber : AiracCycleReleaseNumber	Numeric[4]
105	ctfmEnvBaselineNumber : EnvBaselineNumber	Numeric[6]
106	ctfmDepartureRunway : RunwayDirection	Numeric[2] followed by 'L' (for left) or 'R' (for right) or 'C' (for center) or ''
107	ctfmArrivalRunway : RunwayDirection	Numeric[2] followed by 'L' (for left) or 'R' (for right) or 'C' (for center) or ''
108	ctfmReqFlightlevelSpeedNrOfInstances : NrOfInstances	
109	ctfmReqFlightlevelSpeedList : ReqFlightLevelSpeed [0 .. ctfmReqFlightlevelSpeedNrOfInstances]	
110	ctfmConsumedFuel : ConsumedFuel	[-99999999 ... 99999999]
111	ctfmRouteCharges : RouteCharges	[-99999999 ... 99999999]
112	ctfmAllFtPointNrOfInstances : NrOfInstances	
113	ctfmAllFtPointProfile : AllFtPointProfile[0 .. ctfmAllFtPointProfileNrOfInstances]	
114	ctfmAllFtAirspaceNrOfInstances : NrOfInstances	
115	ctfmAllFtAirspaceProfile : AllFtAirspaceProfile[0 .. ctfmAllFtAirspaceProfileNrOfInstances]	
116	ctfmAllFtCircleIntersectionsNrOfInstances : NrOfInstances	
117	ctfmAllFtCircleIntersections : AllFtAirspaceProfile[0 .. ctfmAllFtCircleIntersectionsNrOfInstances]	
118	noCPGCPFReason : NonEligibleReason	[N X B I C E]
119	scrObt : DateAndHms	
120	scrConsumedFuel : ConsumedFuel	[-99999999 ... 99999999]
121	scrRouteCharges : RouteCharges	[-99999999 ... 99999999]
122	scrAllFtPointNrOfInstances : NrOfInstances	
123	scrAllFtPointProfile : AllFtPointProfile[0 .. scrAllFtPointProfileNrOfInstances]	
124	scrAllFtAirspaceNrOfInstances : NrOfInstances	
125	scrAllFtAirspaceProfile : AllFtAirspaceProfile[0 .. scrAllFtAirspaceProfileNrOfInstances]	
126	scrAllFtCircleIntersectionsNrOfInstances : NrOfInstances	
127	scrAllFtCircleIntersections : AllFtAirspaceProfile[0 .. scrAllFtCircleIntersectionsNrOfInstances]	
128	srrObt : DateAndHms	
129	srrConsumedFuel : ConsumedFuel	[-99999999 ... 99999999]
130	srrRouteCharges : RouteCharges	[-99999999 ... 99999999]
131	srrAllFtPointNrOfInstances : NrOfInstances	
132	srrAllFtPointProfile : AllFtPointProfile[0 .. srrAllFtPointProfileNrOfInstances]	

133	srrAllFtAirspaceNrOfInstances : NrOfInstances	
134	srrAllFtAirspaceProfile : AllFtAirspaceProfile[0 .. srrAllFtAirspaceProfileNrOfInstances]	
135	srrAllFtCircleIntersectionsNrOfInstances : NrOfInstances	
136	srrAllFtCircleIntersections : AllFtAirspaceProfile[0 .. srrAllFtCircleIntersectionsNrOfInstances]	
137	surObt : DateAndHms	
138	surConsumedFuel : ConsumedFuel	[-99999999 ... 99999999]
139	surRouteCharges : RouteCharges	[-99999999 ... 99999999]
140	surAllFtPointNrOfInstances : NrOfInstances	
141	surAllFtPointProfile : AllFtPointProfile[0 .. surAllFtPointProfileNrOfInstances]	
142	surAllFtAirspaceNrOfInstances : NrOfInstances	
143	surAllFtAirspaceProfile : AllFtAirspaceProfile[0 .. surAllFtAirspaceProfileNrOfInstances]	
144	surAllFtCircleIntersectionsNrOfInstances : NrOfInstances	
145	surAllFtCircleIntersections : AllFtAirspaceProfile[0 .. surAllFtCircleIntersectionsNrOfInstances]	
146	dctObt : DateAndHms	
147	dctConsumedFuel : ConsumedFuel	[-99999999 ... 99999999]
148	dctRouteCharges : RouteCharges	[-99999999 ... 99999999]
149	dctAllFtPointNrOfInstances : NrOfInstances	
150	dctAllFtPointProfile : AllFtPointProfile[0 .. dctAllFtPointProfileNrOfInstances]	
151	dctAllFtAirspaceNrOfInstances : NrOfInstances	
152	dctAllFtAirspaceProfile : AllFtAirspaceProfile[0 .. dctAllFtAirspaceProfileNrOfInstances]	
153	dctAllFtCircleIntersectionsNrOfInstances : NrOfInstances	
154	dctAllFtCircleIntersections : AllFtAirspaceProfile[0 .. dctAllFtCircleIntersectionsNrOfInstances]	
155	cpfObt : DateAndHms	
156	cpfConsumedFuel : ConsumedFuel	[-99999999 ... 99999999]
157	cpfRouteCharges : RouteCharges	[-99999999 ... 99999999]
158	cpfAllFtPointNrOfInstances : NrOfInstances	
159	cpfAllFtPointProfile : AllFtPointProfile[0 .. cpfAllFtPointProfileNrOfInstances]	
160	cpfAllFtAirspaceNrOfInstances : NrOfInstances	
161	cpfAllFtAirspaceProfile : AllFtAirspaceProfile[0 .. cpfAllFtAirspaceProfileNrOfInstances]	
162	cpfAllFtCircleIntersectionsNrOfInstances : NrOfInstances	
163	cpfAllFtCircleIntersections : AllFtAirspaceProfile[0 .. cpfAllFtCircleIntersectionsNrOfInstances]	
164	aircraftidIATA : AircraftidIATA	Aircraft Identification.containing AO in IATA (2 or 3 letters), Flight number (1 to 4 numbers) and

		optional suffix (0 or 1 letter), eventually padded with spaces to fit within 8 characters
165	<u>intentionFlight : StatusYesNo</u>	
166	<u>intentionRelatedRouteAssignmentMethod : IntentionRelatedRouteAssignmentMethod</u>	[HIF HIC RPL HSF RCT PFI USR CHE SHR OPT]
167	<u>intentionUID : IntentionUID</u>	Numeric[16]
168	<u>intentionEditionDate : DateAndHms</u>	Date and time expressed as YYYYMMDDHHMMSS
169	<u>intentionSource : IntentionSource</u>	[S O E R V A L D]
170	<u>associatedIntentions : AssociatedIntentions</u>	Alphabetic[8]
171	<u>enrichmentOutput : EnrichmentOutput</u>	[I H M S]
172	<u>eventID : EventType</u>	Alphabetic[3]
173	<u>eventTime : DateAndHms</u>	Date and time expressed as YYYYMMDDHHMMSS
174	<u>flightVersionNr : Numeric[8]</u>	Numeric[8]
175	<u>ftfmNrTvProfiles : NrOfInstances</u>	Numeric[4]
176	<u>ftfmTvProfile : AllFtTvProfile[0 .. ftfmNrTvProfiles]</u>	a)The traffic volume profile elements are separated by a Space. b)Within a traffic volume profile element, the fields are separated by a ':'
177	<u>rtfmNrTvProfiles : NrOfInstances</u>	Numeric[4]
178	<u>rtfmTvProfile : AllFtTvProfile[0 .. rtfmNrTvProfiles]</u>	a)The traffic volume profile elements are separated by a Space. b)Within a traffic volume profile element, the fields are separated by a ':'
179	<u>ctfmNrTvProfiles : NrOfInstances</u>	Numeric[4]
180	<u>ctfmTvProfile : AllFtTvProfile[0 .. ctfmNrTvProfiles]</u>	a)The traffic volume profile elements are separated by a Space. b)Within a traffic volume profile element, the fields are separated by a ':'

PROFILE	DESCRIPTION	FORMAT
FTFM profile (model 1)	Filed Tactical Flight Model	ftfmAllFtPointProfile
		ftfmAllFtAirspaceProfile
		ftfmAllFtCircleIntersections
RTFM profile	Regulated Tactical Flight Model (by ATFM Measures)	rtfmAllFtPointProfile

(model 2)		
		rtfmAllFtAirspaceProfile
		rtfmAllFtCircleIntersections
CTFM profile (model 3)	Current Tactical Flight Model	ctfmAllFtPointProfile
		ctfmAllFtAirspaceProfile
		ctfmAllFtCircleIntersections
CPF_REF profiles	<p>CPF_REF: The Correlated Position reports for a Flight.</p> <p>(1) The system shall supply point and airspace profiles that will follow as closely as possible the actual trajectory and airspace intersections flown by the aircraft.</p> <p>(2) The Correlated Position reports for a Flight a.k.a. CPF that is stored in the ETFMS oplog database (or archived file) will serve as the basis for these profiles. Correlated Positions reports or CPRs for short are radar data plots enhanced with flight data so that they can be correlated with the flight. The ETFMS at present stores a filtered list of CPRs that can serve as a model of the actual flown trajectory (CPF). Filtering of CPRs is required for reasons of bogus and redundant data.</p>	cpfAllFtPointProfile
		cpfAllFtAirspaceProfile
		cpfAllFtCircleIntersections
CPG_GEN profiles	CPG_GEN: The set of SCR, SRR, SUR and DCT as generated by the Path Generator tool in Network Operations Systems.	
SRC	SCR: Shortest Constrained Route; the profile corresponding to the shortest route available at the time of	scrAllFtPointProfile

	flight, with all restrictions validated and using CDRs, if open at the time. This type of route is meant to be IFPS compliant and can be filed in a flight plan that will pass the IFPS without need for manual error correction.	
		scrAllFtAirspaceProfile
		srcAllFtCircleIntersections
SRR	SRR: Shortest RAD restrictions applied Route; the profile corresponding to the shortest route available at the time of flight, with all RAD restrictions validated and using CDRs, assuming all CDRs are open at the time. This type of route will not pass IFPS validation.	srrAllFtPointProfile
		srrAllFtAirspaceProfile
		srrAllFtCircleIntersections
SUR	SUR: Shortest Unconstrained Route; the profile corresponding to the shortest route available at the time of flight, with no RAD applied and using CDRs, assuming all CDRs are open at the time. This type of route will not pass IFPS validation.	surAllFtPointProfile
		surAllFtAirspaceProfile
		sucAllFtCircleIntersections
DCT	DCT: Direct Route; the profile corresponding to the Great Circle route for the portion that is not frozen (see 2.4.3(5) 2.4.3(7) of FB220 - Support to Flight Efficiency Plan document). This type of route will not pass IFPS validation.	dctAllFtPointProfile
		dctAllFtAirspaceProfile
		dctAllFtCircleIntersections

Example:

LYBE;LFPB;YUTPC;;C500;20171105213100;AA69751831;20171105213000;FPL;FPL;FPL;NEXE;NEXE;
N;N;N;;20171105213000;20171105213000;AA;FI;NS;482873;;N;;N;N;;0;0;;FPL;N;N;Y;N;;600;300;;;;;N;0;

0;;N;;0;C;4C4DE2;YUTPC;N;DFGORSY;N;;;;N;;;;K;1;LDZOFIR:9;0;;431;58134;LYBE30;LFPB25;1;F2
 80:N0346;0;901;306;48;20171105214000:LYBE:TUVAR1V:3:0:A:444910N0201825E::Y
 20171105214032::DCT:10:2:V:444944N0201707E:5:Y
 20171105214214::DCT:35:7:V:445110N0201353E:18:Y
 20171105214255::DCT:50:11:V:445218N0201118E:28:Y
 20171105214340::DCT:70:15:V:445327N0200842E:38:Y
 20171105214427::DCT:90:21:V:445509N0200449E:54:Y
 20171105214517::DCT:110:26:V:445635N0200135E:67:Y
 20171105214703::DCT:150:38:V:450000N0195349E:97:Y
 20171105214711:*50LY:TUVAR1V:153:39:D:450017N0195310E::Y
 20171105214901::DCT:190:52:V:450145N0194328E:20:Y
 20171105215117::DCT:230:70:V:450346N0193002E:48:Y
 20171105215234::DCT:250:80:V:450454N0192234E:63:Y
 20171105215400::DCT:270:92:V:450615N0191336E:82:Y
 20171105215447::DCT:280:99:V:450702N0190823E:92:Y
 20171105215520:TUVAR:M19:280:104:W:450736N0190439E::Y
 20171105215921:ADULA:M19:280:141:W:451614N0183831E::Y
 20171105220428:NASSY:M19:280:188:W:452648N0180559E::Y
 20171105221316:VBA:L863:280:269:N:454452N0170848E::Y
 20171105222600:BEDOX:DCT:280:386:W:461558N0154934E::Y
 20171105223300::DCT:280:471:V:463108N0144633E:27:Y
 20171105225426:UMVEG:UQ856:280:704:W:471242N0115348E::Y
 20171105230500::DCT:280:819:V:472041N0102241E:68:Y
 20171105230930:GAMSA:DCT:280:874:W:472430N0093907E::Y
 20171105232141:HOC:UL613:280:1023:N:472800N0073956E::Y
 20171105232245:NATLI:UL613:280:1036:W:472931N0073026E::Y
 20171105232300::DCT:280:1039:V:472954N0072805E:3:Y
 20171105233117:LUL:DCT:280:1129:N:474118N0061744E::Y
 20171105234225::DCT:280:1250:V:480330N0044542E:66:Y
 20171105234244::DCT:270:1254:V:480414N0044240E:68:Y
 20171105234708::DCT:170:1304:V:481325N0040438E:95:Y
 20171105234803:TRO:TRO8E:150:1313:V:481504N0035747E::Y
 20171105234831::DCT:150:1317:V:481631N0035529E:12:Y
 20171105234927::DCT:130:1326:V:481948N0035019E:38:Y
 20171105235024::DCT:110:1336:V:482326N0034434E:68:Y
 20171105235121::DCT:90:1345:V:482642N0033924E:94:Y
 20171105235136:EPURI:TRO8E:85:1347:W:482726N0033815E::Y
 20171105235151::DCT:80:1349:V:482810N0033704E:13:Y
 20171105235316::DCT:80:1360:V:483214N0033036E:87:Y
 20171105235330:OKABO:TRO8E:75:1362:W:483258N0032925E::Y
 20171105235343::DCT:70:1364:V:483332N0032801E:2:Y
 20171106000105::DCT:70:1417:V:484822N0025057E:61:Y
 20171106000210::DCT:50:1426:V:485053N0024440E:71:Y
 20171106000356::DCT:25:1439:V:485432N0023534E:86:Y
 20171106000426::DCT:20:1442:V:485522N0023329E:89:Y
 20171106000508::DCT:15:1444:V:485556N0023205E:91:Y
 20171106000603::DCT:10:1448:V:485703N0022917E:96:Y
 20171106000653::DCT:5:1450:V:485736N0022753E:98:Y
 20171106000724:LFPB::2:1452:A:485810N0022629E::Y:54;20171105214000:LYBETA:20171105214954:ES
 :444910N0201825E:450234N0193816E:3:206:0:59
 20171105214000:LYBETMA:20171105214954:AUA:444910N0201825E:450234N0193816E:3:206:0:59
 20171105214000:LY:20171105215520:NAS:444910N0201825E:450736N0190439E:3:280:0:104
 20171105214000:LYBAFIR:20171105215520:FIR:444910N0201825E:450736N0190439E:3:280:0:104
 20171105214954:LYBALN:20171105215520:ES:450234N0193816E:450736N0190439E:206:280:59:104
 20171105214954:LYBACTA:20171105215520:AUA:450234N0193816E:450736N0190439E:206:280:59:104
 20171105215520:LDZOTOT:20171105222600:ES:450736N0190439E:461558N0154934E:280:280:104:386
 20171105215520:LD:20171105222600:NAS:450736N0190439E:461558N0154934E:280:280:104:386
 20171105215520:LDZOCTA:20171105222600:AUA:450736N0190439E:461558N0154934E:280:280:104:386
 20171105215520:LDZOLN:20171105222600:ES:450736N0190439E:461558N0154934E:280:280:104:386
 20171105215520:LDZOFIR:20171105222600:FIR:450736N0190439E:461558N0154934E:280:280:104:386

20171105222600:LJLAFIR:20171105223300:FIR:461558N0154934E:463155N0144722E:280:280:386:471
 20171105222600:LJ:20171105223300:NAS:461558N0154934E:463155N0144722E:280:280:386:471
 20171105222600:LJLACTA:20171105223523:AUA:461558N0154934E:463641N0142813E:280:280:386:497
 20171105222600:LJLA2529:20171105223523:ES:461558N0154934E:463641N0142813E:280:280:386:497
 20171105223300:LOVVFIR:20171105230500:FIR:463155N0144722E:472058N0102248E:280:280:471:819
 20171105223300:LO:20171105230500:NAS:463155N0144722E:472058N0102248E:280:280:471:819
 20171105223300:EDDDCTA:20171105230604:AUA:463155N0144722E:472150N0101229E:280:280:471:8
 32
 20171105223300:EDDDALL2:20171105230604:ES:463155N0144722E:472150N0101229E:280:280:471:832
 20171105223523:LSAGEUROP1:20171105224950:ES:463641N0142813E:470419N0123125E:280:280:497:
 654
 20171105223523:LOVVCTA:20171105225426:AUA:463641N0142813E:471242N0115348E:280:280:497:70
 4 20171105223523:LOVWW1:20171105225426:ES:463641N0142813E:471242N0115348E:280:280:497:704
 20171105223703:LSASTCTA:20171105235843:AUA:463957N0141456E:484341N0030258E:280:70:515:140
 0 20171105223703:LSASFPLS:20171105235843:ES:463957N0141456E:484341N0030258E:280:70:515:1400
 20171105224950:LOVV1CTA:20171105230539:AUA:470419N0123125E:472130N0101627E:280:280:654:8
 27
 20171105224950:LOVVINN1:20171105230539:ES:470419N0123125E:472130N0101627E:280:280:654:827
 20171105225426:EDMMTEG275:20171105225717:ES:471242N0115348E:471503N0112919E:280:280:704:
 735
 20171105225426:EDMMTEG:20171105225717:ES:471242N0115348E:471503N0112919E:280:280:704:735
 20171105225426:LOVVINN:20171105230500:ES:471242N0115348E:472058N0102248E:280:280:704:819
 20171105225426:EDMMALL:20171105230807:ES:471242N0115348E:472326N0095238E:280:280:704:857
 20171105225426:EDMMCTA:20171105230807:AUA:471242N0115348E:472326N0095238E:280:280:704:8
 57
 20171105225717:EDMMSTA:20171105230052:ES:471503N0112919E:471753N0105828E:280:280:735:774
 20171105230052:EDMMZUG:20171105230807:ES:471753N0105828E:472326N0095238E:280:280:774:85
 7
 20171105230500:EDUUUIR:20171105230604:FIR:472058N0102248E:472150N0101229E:280:280:819:832
 20171105230500:ED:20171105230604:NAS:472058N0102248E:472150N0101229E:280:280:819:832
 20171105230604:LOVVFIR:20171105230930:FIR:472150N0101229E:472430N0093907E:280:280:832:874
 20171105230604:LO:20171105230930:NAS:472150N0101229E:472430N0093907E:280:280:832:874
 20171105230807:LSAZUTA:20171105232317:AUA:472326N0095238E:473018N0072546E:280:280:857:10
 42
 20171105230807:LSAZM2:20171105232317:ES:472326N0095238E:473018N0072546E:280:280:857:1042
 20171105230930:LSASUIR:20171105232300:FIR:472430N0093907E:472955N0072806E:280:280:874:1039
 20171105230930:LS:20171105232300:NAS:472430N0093907E:472955N0072806E:280:280:874:1039
 20171105231340:LSAZSTRMZ:20171105231929:ES:472556N0085821E:472731N0080133E:280:280:925:99
 6
 20171105232300:LFFFUIR:20171105234605:FIR:472955N0072806E:481121N0041355E:280:194:1039:1292
 20171105232300:LF:20171106000724:NAS:472955N0072806E:485810N0022629E:280:2:1039:1452
 20171105232317:LFEEUH:20171105233403:ES:473018N0072546E:474700N0055507E:280:280:1042:1159
 20171105232317:LFEEECTA:20171105234146:AUA:473018N0072546E:480233N0045123E:280:280:1042:1
 243
 20171105233403:LFEEUF:20171105234146:ES:474700N0055507E:480233N0045123E:280:280:1159:1243
 20171105233833:LFFFSTREAM:20171106000724:ES:475609N0051801E:485810N0022629E:280:2:1208:14
 52 20171105234146:LFFFAR:20171105235136:ES:480233N0045123E:482726N0033815E:280:85:1243:1347
 20171105234146:LFFFCTA:20171106000724:AUA:480233N0045123E:485810N0022629E:280:2:1243:1452
 20171105234605:LFFFIR:20171106000724:FIR:481121N0041355E:485810N0022629E:194:2:1292:1452
 20171105234631:LFFFLOW:20171106000724:ES:481215N0041004E:485810N0022629E:184:2:1297:1452
 20171105235136:LFFFEGA:20171106000724:ES:482726N0033815E:485810N0022629E:85:2:1347:1452
 20171106000141:LFPGITMW:20171106000426:ES:484950N0024735E:485524N0023332E:59:20:1422:1442
 ;0;;;;;;431;58134;LYBE30;LFPB25;1;F280:N0346:0;1061;302;92;20171105214100:LYBE:DCT:3:0:A:44
 4910N0201825E::Y 20171105214132::DCT:10:1:V:444843N0201850E:11:Y
 20171105214216::DCT:20:2:V:444815N0201915E:22:Y
 20171105214238::DCT:25:3:V:444748N0201939E:33:Y
 20171105214312::DCT:35:4:V:444720N0202004E:44:Y
 20171105214352::DCT:50:6:V:444625N0202054E:67:Y
 20171105214437::DCT:70:8:V:444530N0202143E:89:Y
 20171105214452::DCT:74:9:G:444503N0202208E::Y

20171105214532::DCT:90:13:V:444551N0201917E:67:Y
20171105214542::DCT:90:14:V:444603N0201835E:83:Y
20171105214552::DCT:92:15:G:444615N0201752E::Y
20171105214612::DCT:100:17:V:444658N0201648E:8:Y
20171105214640::DCT:110:20:V:444803N0201512E:20:Y
20171105214728::DCT:120:25:V:444952N0201233E:40:Y
20171105214736::DCT:120:26:V:445013N0201201E:44:Y
20171105214755::DCT:126:28:V:445057N0201057E:52:Y
20171105214806::DCT:130:29:V:445118N0201025E:56:Y
20171105214820::DCT:135:31:V:445202N0200921E:64:Y
20171105214845::DCT:144:34:V:445307N0200746E:76:Y
20171105214916::DCT:152:38:V:445434N0200538E:92:Y
20171105214932:LABUD:DCT:154:40:W:445517N0200434E::Y
20171105214956::DCT:158:43:V:445610N0200231E:43:Y
20171105215004::DCT:160:44:V:445628N0200149E:57:Y
20171105215028:DEVDA:DCT:160:47:W:445721N0195946E::Y
20171105215051::DCT:166:50:V:445734N0195730E:7:Y
20171105215123::DCT:173:54:V:445751N0195429E:16:Y
20171105215147::DCT:181:57:V:445804N0195214E:22:Y
20171105215218::DCT:186:61:V:445821N0194913E:31:Y
20171105215231::DCT:190:63:V:445830N0194742E:36:Y
20171105215241::DCT:193:64:V:445834N0194657E:38:Y
20171105215341::DCT:204:72:V:445908N0194055E:56:Y
20171105215433::DCT:218:79:V:445938N0193539E:71:Y
20171105215530::DCT:230:87:V:450013N0192937E:89:Y
20171105215554::DCT:235:90:V:450025N0192721E:96:Y
20171105215607:SMI:DCT:237:92:N:450034N0192551E::Y
20171105215622::DCT:240:94:V:450101N0192429E:6:Y
20171105215629::DCT:240:95:V:450115N0192348E:10:Y
20171105215704::DCT:246:100:V:450223N0192023E:26:Y
20171105215800::DCT:254:108:V:450412N0191454E:52:Y
20171105215906::DCT:262:117:V:450614N0190845E:81:Y
20171105215936::DCT:267:121:V:450709N0190601E:94:Y
20171105215950:TUVAR:M19:269:123:W:450736N0190439E::Y
20171105220124::DCT:280:136:V:451038N0185528E:35:Y
20171105220405:ADULA:M19:280:160:W:451614N0183831E::Y
20171105220801::DCT:280:195:V:452406N0181417E:74:Y
20171105220927:NASSY:M19:280:207:W:452648N0180559E::Y
20171105221810::DCT:280:280:V:454305N0171427E:90:Y
20171105221906:VBA:L863:280:288:N:454452N0170848E::Y
20171105222750::DCT:280:363:V:460448N0161801E:64:Y
20171105223247:BEDOX:DCT:280:405:W:461558N0154934E::Y
20171105223325::DCT:280:412:V:461713N0154423E:2:Y
20171105224148::DCT:280:490:V:463108N0144633E:27:Y
20171105224343::DCT:280:506:V:463359N0143441E:32:Y
20171105225343::DCT:280:599:V:465035N0132544E:61:Y
20171105230401::DCT:280:693:V:470721N0121603E:91:Y
20171105230719:UMVEG:UQ856:280:723:W:471242N0115348E::Y
20171105231407::DCT:280:785:V:471700N0110441E:36:Y
20171105232006::DCT:280:838:V:472041N0102241E:68:Y
20171105232413::DCT:280:879:V:472332N0095012E:92:Y
20171105232543:GAMSA:DCT:280:893:W:472430N0093907E::Y
20171105233429::DCT:280:975:G:473030N0083435E::Y
20171105233906::DCT:280:1018:V:473543N0080048E:26:Y
20171105234222::DCT:280:1052:V:473950N0073405E:46:Y
20171105234714::DCT:280:1097:V:474516N0065843E:73:Y
20171105235150::DCT:280:1142:G:475043N0062321E::Y
20171106000127::DCT:280:1235:V:480822N0051315E:66:Y
20171106000152::DCT:270:1239:V:480907N0051014E:69:Y
20171106000157::DCT:269:1240:V:480919N0050928E:70:Y

20171106000222::DCT:269:1244:V:481004N0050627E:73:Y
 20171106000355::DCT:242:1261:V:481318N0045339E:85:Y
 20171106000407::DCT:242:1263:V:481341N0045208E:86:Y
 20171106000451::DCT:228:1271:V:481512N0044606E:92:Y
 20171106000539::DCT:210:1280:V:481654N0043919E:99:Y
 20171106000551::DCT:205:1282:G:481717N0043749E::Y
 20171106000636::DCT:190:1290:V:481842N0043136E:9:Y
 20171106000651::DCT:185:1292:V:481904N0043002E:11:Y
 20171106000708::DCT:180:1295:V:481936N0042742E:15:Y
 20171106000923::DCT:180:1315:V:482310N0041210E:38:Y
 20171106000953::DCT:170:1320:V:482403N0040817E:43:Y
 20171106001055::DCT:150:1331:V:482601N0035944E:56:Y
 20171106001157::DCT:130:1341:V:482748N0035157E:67:Y
 20171106001301::DCT:110:1352:V:482945N0034324E:80:Y
 20171106001406::DCT:90:1362:V:483132N0033538E:91:Y
 20171106001500::OKABO:TRO8E:75:1370:W:483258N0032925E::Y
 20171106001520::DCT:70:1373:V:483348N0032719E:3:Y
 20171106002158::DCT:70:1421:V:484715N0025345E:57:Y
 20171106002311::DCT:50:1431:V:485003N0024646E:68:Y
 20171106002508::DCT:25:1446:V:485415N0023616E:84:Y
 20171106002542::DCT:20:1449:V:485505N0023411E:88:Y
 20171106002629::DCT:15:1451:V:485539N0023247E:90:Y
 20171106002729::DCT:10:1455:V:485646N0022959E:94:Y
 20171106002900:LFPB::2:1460:A:485810N0022629E::Y:59;20171105214100:LYBETA:20171105215348:ES
 :444910N0201825E:445914N0194011E:3:206:0:73
 20171105214100:LYBETMA:20171105215348:AUA:444910N0201825E:445914N0194011E:3:206:0:73
 20171105214100:LY:20171105215950:NAS:444910N0201825E:450736N0190439E:3:269:0:123
 20171105214100:LYBAFIR:20171105215950:FIR:444910N0201825E:450736N0190439E:3:269:0:123
 20171105215348:LYBALN:20171105215950:ES:445914N0194011E:450736N0190439E:206:269:73:123
 20171105215348:LYBACTA:20171105215950:AUA:445914N0194011E:450736N0190439E:206:269:73:123
 20171105215950:LDZOTOT:20171105223247:ES:450736N0190439E:461558N0154934E:269:280:123:405
 20171105215950:LD:20171105223247:NAS:450736N0190439E:461558N0154934E:269:280:123:405
 20171105215950:LDZOCTA:20171105223247:AUA:450736N0190439E:461558N0154934E:269:280:123:405
 20171105215950:LDZOLN:20171105223247:ES:450736N0190439E:461558N0154934E:269:280:123:405
 20171105215950:LDZOFIR:20171105223247:FIR:450736N0190439E:461558N0154934E:269:280:123:405
 20171105223247:LJLAFIR:20171105224148:FIR:461558N0154934E:463155N0144722E:280:280:405:490
 20171105223247:LJ:20171105224148:NAS:461558N0154934E:463155N0144722E:280:280:405:490
 20171105223247:LJLACTA:20171105224448:AUA:461558N0154934E:463641N0142813E:280:280:405:516
 20171105223247:LJLA2529:20171105224448:ES:461558N0154934E:463641N0142813E:280:280:405:516
 20171105224148:LOVVFIR:20171105232006:FIR:463155N0144722E:472058N0102248E:280:280:490:838
 20171105224148:LO:20171105232006:NAS:463155N0144722E:472058N0102248E:280:280:490:838
 20171105224148:EDDDCTA:20171105232124:AUA:463155N0144722E:472150N0101229E:280:280:490:8
 51
 20171105224148:EDDDALL2:20171105232124:ES:463155N0144722E:472150N0101229E:280:280:490:851
 20171105224448:LSAGEUROP1:20171105230150:ES:463641N0142813E:470419N0123125E:280:280:516:
 673
 20171105224448:LOVVCTA:20171105230719:AUA:463641N0142813E:471242N0115348E:280:280:516:72
 3
 20171105224448:LOVWW1:20171105230719:ES:463641N0142813E:471242N0115348E:280:280:516:723
 20171105224644:LSASTCTA:20171106002010:AUA:463957N0141456E:484341N0030258E:280:70:534:140
 8
 20171105224644:LSASFPLS:20171106002010:ES:463957N0141456E:484341N0030258E:280:70:534:1408
 20171105230150:LOVV1CTA:20171105232054:AUA:470419N0123125E:472130N0101627E:280:280:673:8
 46
 20171105230150:LOVVINN1:20171105232054:ES:470419N0123125E:472130N0101627E:280:280:673:846
 20171105230719:EDMMTEG275:20171105231043:ES:471242N0115348E:471503N0112919E:280:280:723:
 754
 20171105230719:EDMMTEG:20171105231043:ES:471242N0115348E:471503N0112919E:280:280:723:754
 20171105230719:LOVVINN:20171105232006:ES:471242N0115348E:472058N0102248E:280:280:723:838
 20171105230719:EDMMALL:20171105232355:ES:471242N0115348E:472326N0095238E:280:280:723:876
 20171105230719:EDMMCTA:20171105232355:AUA:471242N0115348E:472326N0095238E:280:280:723:8
 76

9.7.6 are - Newmaxo ASCII Region file

Newmaxo ascii region file	
extension	are
origin	newmaxo, SAAM, CAPAN
separator	blank
sort	Body is sorted by point sequence (for SAAM it should be preferably clockwise to determine inside from outside)
comment	Describe a 2D/3D piece of airspace. Has a header and a body. For SAAM, normally linked to an "sls" file see Converting SAAM format to/from Gasel format

#	Field	Type	Size	Comment
1	nb_point	num	~	contains the number of lines (=vertices) of the following body
2	latitude	num	~	in minutes (decimal for SAAM), location of the label
3	longitude	num	~	in minutes (decimal for SAAM), location of the label

4	flights	num	~	value 1 (can be negative), can be o
5	bottom_level	num	~	low level of the volume in flight level (FL) (see below), can be o
6	top_level	num	~	high level of the volume in flight level (FL) (see below), can be o
7	surface	num	~	value 2 (can be negative), can be o
8	sector_num	num	~	value 3 (can be negative), can be o
9	flight time	num	~	value 4 (multiplied by 100 to get 2 decimals, can be negative), can be o
10	traffic density	num	~	value 5, can be o
11	x mileage	num	~	Label content for SAAM (see below), can be o
12	rte extens.	num	~	Feature code for SAAM (see below), can be o
13	value 1	num	~	Color code for SAAM (see below), can be o
14	value 2	num	~	value 6, can be o
15	name	char	~	name or code of the volume (max 12 char for newmaxo, max 24 character for SAAM) (see below)
Body, contains vertices coordinates (the polygon must be closed: first point = last point)				
1	latitude	num	~	in minutes (in minutes decimal for SAAM)
2	longitude	num	~	in minutes (in minutes decimal for SAAM)

Additional comments:

Field 5 & 6: low/high level: if low and high levels are swapped, the volume is marqued negative (to be subtracted)	
Field 11, label content code	<p>Indicates the content of the label:</p> <p>bit 0: sector name display (if only sector is display then only one airblock of the sector will be chosen for display)</p> <p>bit 1: airblock name display (all airblock have a label)</p> <p>bit 2:min/max FL display (of the airblock by default or of the sector if sector displayed)</p> <p>bit 3 to 7: value display (bit 3=value1, bit 4=value2, bit 5=value3, bit 6=value4, bit 7=value5)</p>
Field 12, feature code	<p>indicates if the features for this airspace come from the TDV line (common to all airspace of the file) or from the airspace itself. Each feature has 3 values (except flight level):</p> <p>0 means the feature comes from TDV line</p> <p>1 means the feature comes from the airspace itself and is set to ON</p> <p>2 means the feature comes from the airspace itself and is set to OFF</p> <p>bit 0 and 1: top flag</p> <p>bit 2 and 3: lighting flag</p> <p>bit 4 and 5: transparent flag</p> <p>bit 6: bottom level flag (0 means come from TDV if possible (!=999), 1 means come from the airspace header field 5)</p> <p>bit 8: top level flag (0 means come from TDV if possible (!=999), 1 means come from the airspace header field 6)</p> <p>bit 10: label flag (0 means come from TDV if possible (!=9), 1 means label is displayed, 2 label is not displayed (see also content of the label))</p> <p>bit 12: soft lock flag (0 means come from TDV if possible (see lower/upper</p>

	<p>case of the airspace file name in TDV volume format), 1 mean soft lock, 2 means no soft lock (normal)</p> <p>Examples:</p> <p>192: means only bottom and top level are set from the airspace itself, the rest comes from TDV line</p> <p>208: means bottom and top level and transparent ON for the airspace itself, rest is from TDV</p> <p>224: means bottom and top level and transparent OFF for the airspace itself, rest is from TDV</p>
Field 13, color code	<p>If this field is set to 0 (ZERO) it means the color for this airspace is random if no other color are defined in the TDV or an ACO file, else it represents the value of the color for the airspace itself.</p> <p>The coding is (for each component varying between 0 and 255): $\text{red} * 16777216 + \text{green} * 65536 + \text{blue} * 256 + \text{transparent}$ look at color transformation in this excel file.</p> <p>If level of transparency is 0 (whatever the values for R,G & B are) then the level of transparency is taken from TDV file.</p> <p>The most black most transparent (=totally translucid) for an individual airspace is then: 16843009.</p> <p>The most black less transparent (=totally black) for an individual airspace is then: 16843263.</p> <p><u>Try the 2 "color transformation" tables on the left side.</u></p>
Field 15, name convention	<p>The name <u>might</u> have 2 parts, separated by a semi-colon.</p> <p>If the name have 2 parts separated by semi-colon (this happened when ".are" is alone, so no associated sls), in that case, the first part is the name of the sector (or the group),</p> <p>the second part is the name of the airblock (or piece of airspace which is the element that belongs to the group) described by the data itself.</p> <p>Example: sEBBRTMA:036EB</p> <p>If the name has one part, (an associated ".sls" should exist) it might represent the name of the airblock which is used in the SLS file, in that case it MUST be the same name between ARE an SLS !!!</p> <p>Example: LFO34</p>

COLOR TRANSFORMATION	
color input:	4294901862
red result:	255
green result:	255
blue result:	0
trans. result:	102
red input:	255
green input:	255
blue input:	10
trans. Input:	102
color result:	4294904422
FEATURE CODE TRANSFORMATION	
feature code input:	

top flag result:	
lighting flag res.:	
trans. flag res.:	
bottom lev. flag res.:	
top lev. flag res.:	
label flag result:	
soft lock flag res.:	
top flag input:	
lighting flag inp.:	
trans. flag inp.:	
bottom lev. flag inp.:	
top lev. flag inp.:	

Example:

```

14 2799 925 0 0 660 0 0 0 0 0 0 0 0 0 0 LJ
2.799.925
2792 932
2797 943
2785.91 954.62
2780 942
2774 947
2781 957
2784 964
2784 975
2784 977
2790 974
2794 982
2790 992
2799 925

```

9.7.7 arp - Airport

Airport definition	
extensio n:	arp
origin:	manual input
separato r:	blank
sort:	no
comment:	defines the geographical location of airports used in the assignment process.

NORMAL LINE				
#	Field	Type	Size	Comment
1	airport name	char	4	ICAO airport name
2	latitude	num	~	in minutes (in minutes decimal for SAAM)
3	longitude	num	~	in minutes (in minutes decimal for SAAM)

4	FIR name	char	5/7	FIR name associated to the airport, with "_FIR" as a suffix. This field is optional. "NOFIR" is used when no FIR information is available.
---	----------	------	-----	--

Example:

```

EGSX 3103.267 9.350 EGTT_FIR
EGSY 3203.650 -83.300 EGTT_FIR
EGSZ 3620.000 -102.000 EGTX_FIR
EGTA 3106.883 -56.550 EGTT_FIR
EGTB 3096.700 -48.483 EGTT_FIR

```

9.7.8 as4

AS4				
extensio n	as4			
origin	SAAM filter so6ToAs4			
separato r	blank			
sort	yes: on "time begin segment" then on "time end segment"			
commen t	ascii file used as an input in the SAAM/3Dviewer for traffic animation			
#	Field	Type	Size	Comment
1	flighth key	num		could be used to retrieve information on the flight (not used)
2	sequence	num		give the sequence of segment route for a given flight (not used)
3	segment name	char	max 11	pointName1_pointName2 (currently ignored)
4	lat begin segment	float		in minute decimal
5	lon begin segment	float		in minute decimal
6	FL begin segment	float		in flight level with decimal
7	time begin segment	float		in second decimal
8	lat end segment	float		in minute decimal
9	lon end segment	float		in minute decimal
10	FL end segment	float		in flight level with decimal
11	time end segment	float		in second decimal

Example:

```

13972 15 VALPO_LI7 2750.1 685.9 350 0 2754.5 678.35 350 1.12
16185 9 $alYh_ES21 3582.7 899.55 350 0 3585.2 849.8 307 1.13
11907 19 %LK6_VLM 2969.1 913.3 191.2 0 2982.7 904.6 142.6 2
4912 4 $adEt_REDFA 3123.4 129.2 290 0 3127.7 149.7 290 2
7754 3 $afKv_RASDA 1963.9 1857.8 350 0 1986.7 1856.6 350 3
7876 10 $afPA_DVR 3074.3 56.4 370 0 3070.5 82.7 333.2 3.2
13968 20 %ED88_ED15 2979.7 506 350 0 3002 447.81 350 5
2441 19 PON_EVX 2945 122.1 330 0 2941 73.9 330 5
4888 7 EXMOR_BHD 3070 -201 370 0 3023 -209 370 6
3277 12 AGN_ANETO 2633.4 52 370 0 2562 33.8 370 9

```

9.7.9 ase - ASCII Segment file

Ascii SEgment				
#	Field	Type	Size	Comment
1	flight count	float		normally number of flights using this route segment, could be a load
2	segment parity	num		0=NO, 1=ODD, 2=EVEN, 3=ODD_LOW, 4=EVEN_LOW, 5=ODD_HIGH, 6=EVEN HIGH
3	segment type	num		0=NO, 1=NORMAL, 2=ARRIVAL, 3=DEPARTURE (permanent rte segment) 20=NO, 21=NORMAL, 22=ARRIVAL, 23=DEPARTURE (CDR Generic) 40=NO, 41=NORMAL, 42=ARRIVAL, 43=DEPARTURE (CDR 1) 60=NO, 61=NORMAL, 62=ARRIVAL, 63=DEPARTURE (CDR 2) 80=NO, 81=NORMAL, 82=ARRIVAL, 83=DEPARTURE (CDR 3) 100=NO, 101=NORMAL, 102=ARRIVAL, 103=DEPARTURE (CDR 1+2) 120=NO, 121=NORMAL, 122=ARRIVAL, 123=DEPARTURE (CDR 1+3)
4	lat begin segment	float		in minute decimal (only in minute for Newmaxo)
5	lon begin segment	float		in minute decimal (only in minute for Newmaxo)
6	lat end segment	float		in minute decimal (only in minute for Newmaxo)

7	lon end segment	float		in minute decimal (only in minute for Newmaxo)
8	segment name	char	max 11	routePointNameBegin_routePointNameEnd (separator can only be "_")

Example:

```

373 2 3 2941.1 145.9 2919.25 119.02 WDG_RBT
376 5 1 2825.3 466 2831 480 BERSU_RIVEL
384 2 1 3049 505 3014 556 %ED24_GELNI
387 5 41 2859 399 2877 400 ARPUS_MIRGU
445 2 1 3078 36 3070.85 82 DET_DVR

```

Note: segment ARPUS_MIRGU is a CDR

9.7.10 asx - ASCII Segment Extended file

Ascii Segment eXtended	
extension	asx
origin	so6ToNetwork SAAM module or SAAM Network editor or NM data translator
separator	blank
sort	No sort
comment	Describe a route network (possibly loaded), each line represent a route segment or a single point or an airport

#	Field	Type	Size	Comment
1	flight count	float		normally number of flights using this route segment, could be a load
2	segment parity	num		0=NO, 1=ODD, 2=EVEN, 3=ODD_LOW, 4=EVEN_LOW, 5=ODD_HIGH, 6=EVEN_HIGH
3	item type	num		less than 900 means a SEGMENT being: ROUTE segments: 0=NO, 1=NORMAL, 2=ARRIVAL, 3=DEPARTURE (permanent ROUTE) 20=NO, 21=NORMAL, 22=ARRIVAL, 23=DEPARTURE (CDR Generic ROUTE) 40=NO, 41=NORMAL, 42=ARRIVAL, 43=DEPARTURE (CDR 1 ROUTE) 60=NO, 61=NORMAL, 62=ARRIVAL, 63=DEPARTURE (CDR 2 ROUTE) 80=NO, 81=NORMAL, 82=ARRIVAL, 83=DEPARTURE (CDR 3 ROUTE) 100=NO, 101=NORMAL, 102=ARRIVAL, 103=DEPARTURE (CDR 1+2 ROUTE) 120=NO, 121=NORMAL, 122=ARRIVAL, 123=DEPARTURE (CDR 1+3 ROUTE) STAR segments: 200=NO, 201=NORMAL, 202=ARRIVAL (permanent

				STAR) 220=NO, 221=NORMAL, 222=ARRIVAL (CDR generic STAR) 240=NO, 241=NORMAL, 242=ARRIVAL (CDR 1 STAR) 260=NO, 261=NORMAL, 262=ARRIVAL (CDR 2 STAR) 280=NO, 281=NORMAL, 282=ARRIVAL (CDR 3 STAR) 200=NO, 201=NORMAL, 202=ARRIVAL (CDR 1+2 STAR) 220=NO, 221=NORMAL, 222=ARRIVAL (CDR 1+3 STAR) SID segments: 400=NO, 401=NORMAL, 403=DEPARTURE (permanent SID) 420=NO, 421=NORMAL, 423=DEPARTURE (CDR generic SID) 440=NO, 44
4	lat begin segment	float		in minute decimal
5	lon begin segment	float		in minute decimal
6	lat end segment	float		in minute decimal, equals to field 4 in case of single point or airport
7	lon end segment	float		in minute decimal, equals to field 5 in case of single point or airport
8	item name	char	max 11	routePointNameBegin_routePointNameEnd for segment or pointName or AirportName for single point. Letters not accepted are: ? / \ " ` ~ \$ # ^ & () - + = { } [] , < > space Special case of _ is used as a separator for segment
9	airway name	char		for instance UZ225, ? if no airway name
10	Lower Level Limit	num	3	3 digits expressed in FL
11	Upper Level limit	num	3	3 digits expressed in FL
14	airway seg. sequ.	num		only for airway: provide segments sequencing and possible break, starts at 1. Set to 0 for other items or other cases.
15	Layer Project ID	num		linked to Layer data file else 0

Example (extract that do not correspond to any offical data):

373	2	3	2941.1	145.9	2919.25	119.02	BENSU_WOOD	UA47	295	999
1	0									

9.7.11 awk, flc2 - Rule file

Rule file	
extension	awk or flc2
origin	Edited by the Network Rule Editor , used by the assignment and profile modules.
separator	N.A.
sort	Visualization order in the editor
comment	Describe a set of RAD rules. A rule is defined by a header line and multiple definition lines. They are organized in sections for display purposes.

Section header line:

#	Field	Type	Size	Comment
1	#SECTION_X	text		Section token identifier: X has to be replaced by the depth of the section. Note: The first section of the file should have a depth of 1.
2	-	char	1	
3	name	text		The name of the section

Rule header line:

#	Field	Type	Size	Comment
1	#RAD	text		Rule token identifier. Allow identifying a rule line.
2	id	text		The ID of the RAD rule. Should be unique among the file.
3	- properties -	text		Optional. A list of properties expressed as a list of key=value separated from the rest of the line using two "- " token. <ul style="list-style-type: none"> • The key should not contain any spaces or carriage return. • The value should comply to the following rules: <ul style="list-style-type: none"> ◦ Values can be any text. ◦ If it contains white-spaces, it has to be surrounded by quotes or escaped using a "\". ◦ Quotes have to be escaped. ◦ Carriage returns should be replaced by "
". ◦ Minus signs, "-", have to be escaped.

				<i>Example:</i> - PERIOD="MON-TUE 06.00 10.00" POINTNAME=KOK - - OP_GOAL="Link route for EGKK deps to DVR" -
4	AIRAC XXXX	text		Optional. XXXX have to be replaced with the number of the AIRAC cycle associated with the rule.
5	Comments	text		Optional. A free text describing the rule. Carriage return are replaced by " "

Rule definition (multiple lines):

The definition of the rule. The syntax depend of the file type.

- See [Segment Management syntax](#) for awk files.
- See [Flight Level Constraints syntax](#) for flc2 files.

End line

At the end, file can have a line starting by "#END".

Everything after this token, as well as everything before the first section of the file, won't be part of any rules. But any line written in this part of the file must start by "#" to be processed correctly by the assignment or the profile.

Example:

```
#SECTION_1 - Pan-European RADAN (RAD Annex and SAAM Required)

#SECTION_2 - BL BELARUS

#SECTION_3 - SAAM Required

#RAD SAAM_BLdod0001 - SAAM_REQ="SAAM Required" -
routes <FL195
    if ((RFL>195) && ! (DEP ~ /UMMS|UMMM/)) { REGLE=REGLE
" - BUGOR DERAD - DERAD PIROS - PIROS DERAD - DERAD BUGOR"}

#RAD SAAM_BLdod0002 - SAAM_REQ="SAAM Required" - ATS -
routes one/be directional
```

```

        if (RFL>305) { REGLE=REGLE " - OSMUS BESAK - BESAK
NELIM - NELIM RUDKA" }
if (RFL>275) { REGLE=REGLE " - GORAT UGNAL" }
if (RFL>285) { REGLE=REGLE " - *GRD SOTET" }
if (RFL>305) { REGLE=REGLE " - MNS GOVIK - GOVIK RUDKA" }
if (RFL>205) { REGLE=REGLE " - BERIS DEDOK - DEDOK BERIS - DEDOK
MGLbl - MGLbl DEDOK - MGLbl SLEDA - SLEDA MGLbl - SLEDA MAMIN - MAMIN
SLEDA" }
if (RFL>205) { REGLE=REGLE " - BRT SURUG - SURUG NELIM - NELIM GRINA
- GRINA *GRD - *GRD GRINA - GRINA NELIM - NELIM SURUG - SURUG BRT" }

#SECTION_2 - DS DENMARK / SWEDEN

#SECTION_3 - RAD Annex

#RAD DS2080 AIRAC 1404
if (ARR ~ /EKBI|EKEB|EKVJ/) { REGLE=REGLE " - KUGAL
VES" }

#RAD DS2037 AIRAC 1404
if (DEP ~ /EKBI|EKEB|EKVJ/) { REGLE=REGLE " - VES
LANUL" }

#RAD DS2116 AIRAC 1404 Simplified for CDRs
if (DEP ~ /EVRA/) && (ARR ~ /ENAL/) && (RFL<285)
{REGLE=REGLE " > NEKET NEKLA > NEKLA TEB > TEB LINSA > LINSA ELTOK >
ELTOK PERAX > PERAX BOR > BOR GEVRU > GEVRU SUVAR"}
#if (DEP ~ /EVRA/) && (ARR ~ /ENAL|ENBR|ENZV/) && (RFL<285)
{REGLE=REGLE " > NEKET NEKLA > NEKLA TEB > TEB BEDLA > BEDLA IBGAX >
IBGAX EBURI > EBURI TEKVA > TEKVA ESEBA"}
if (DEP ~ /EVRA/) && (ARR ~ /ENZV/) && (RFL<285) {REGLE=REGLE " >
NEKET NEKLA > NEKLA TEB > TEB BEDLA > BEDLA KSD > KSD OKSAT > OKSAT
VATEX"}
if (DEP ~ /EVRA/) && (ARR ~ /ENBR/) && (RFL<285) {REGLE=REGLE " >
RASEL NIKEG > NIKEG NILUG > NILUG TRS > TRS IBGAX > IBGAX EBURI >
EBURI TEKVA > TEKVA ESEBA"}

#SECTION_3 - RAD App 5

#RAD ES2109 AIRAC 1404 AIRAC1213
if (DEP ~ /ESSA/) && (ARR ~ /ESSB/) {REGLE=REGLE " +
SA706 B1210" }

#RAD ES2110 AIRAC 1404 AIRAC1213
if (DEP ~ /ESSB/) && (ARR ~ /ESSA/) {REGLE=REGLE " +
NT211 TEB >TEB" }

#SECTION_3 - SAAM Required

#RAD SAAM_DSsge0001 - SAAM_REQ="SAAM Required" - AIRAC
1404 Validation (old EK2050)

```

```

        if (DEP ~ /EKCH|EKFR|EKGH|EKHE|EKHK|EKRH|EKRK|EKRS |
EKVL/) { REGLE=REGLE " + LANGO ALASA" }

#RAD SAAM_EK000062EB - SAAM_REQ="SAAM Required" -
AIRAC 1404 OLD
        if (DEP ~ /EKCH/) {REGLE=REGLE " - CDA ODARU - BISTA
CDA"}
if (DEP ~ /EKCH/) && (RFL < 245) { REGLE= REGLE " - SORGA MIKSI "}
if (DEP ~ /EKCH/) && (ARR=="EKKA") { REGLE=REGLE " - KAS07 DOBEL" }
if (DEP ~ /EKCH/) && (ARR=="EKYT") { REGLE=REGLE " - GOLGA ASLID" }

#END
#General comment not part of any rules

```

9.7.12 cdr - Conditional Routes file

CDR (conditional routes)	
extension	cor
origin	SAAM network editor
separator	semicolon ";"
sort	no sort
comment	<p>Each line defines a time and level window CLOSURE for a given CDR for a given scenario.</p> <p>All existing CDR of type 2 and 3 (by default closed) must be described in this file with at least one "default" scenario line.</p> <p>If "bottom level" equals "top level" equals 660, the CDR is OPEN for the given scenario.</p> <p>If a CDR exists and is not described in this file it is considered OPEN (case for CDR type 1 by default open).</p>

#	Field	Type	Size	Comment
1	version	num		has value '1'
2	first point name	char	max 5	ICAO point name
3	second point name	char	max 5	ICAO point name
4	bottom level	num		division FL (values go from FL5 to FL5 with max 660, min 0)
5	top level	num		division FL (values go from FL5 to FL5 with max 660, min 0)
6	start time	char	5	format HH:MM padded with 0
7	end time	char	5	format HH:MM padded with 0
8	scenario name	char		specific "default" word is valid for any scenario not defined. Scenario name could also be a date with format "DD/MM/YYYY".

Example:

```

1;SUPAM;ARNEM;660;660;00:00;12:00;default
1;LOT;REMBA;0;660;00:00;24:00;default

```

```

1;LOT;REMBA;0;185;08:00;19:00;Any Friday
1;LOT;REMBA;185;285;10:00;12:30;Any Friday
1;SIGEN;TESGA;345;660;10:30;17:30;Any Friday

```

Note:

- SUPRA_ARNEM is open by default,
- LOT_REMBA is closed by default, it is partially open for "Any Friday" scenario
- SIGEN_TESGA is not described in a "default" scenario so it is considered open outside "Any Friday" scenario

9.7.13 cfg - Configuration

NAME	Configuration			
extension:	cfg			
origin:	gasel, SAAM			
separators:	;	(semicolon)		
sort:	by family name then by config name			
comment:				
FIRST LINE				
#	Field	Type	Size	Comment
1#	char		1	
2file type	text			= CONFIGURATION
3version	num			= 1
4NM AIRAC cycle	num			
5AIRAC start date	date			format is yyyyMMdd
6AIRAC end date	date			format is yyyyMMdd
7number of record	num			
8datasource	text			
#	Field	Type	Size	Comment
1ACC name	text			
2Configuration name	text			
3Sector name	text			

Example:

```
#;CONFIGURATION;1;383;20140206;20140305;24398;EAR_P
BIRDCTA;CONF1;BIRDSS
BIRDCTA;CONF1;BIRDNO
BIRDCTA;CONF1;BIRDSS
BIRDCTA;CONF1;BIRDVGIA
BIRDCTA;CONF1;BIRDWE
BIRDICTA;CNF1;BIRDFIS
BIRDTOCA;CONF1;BIRDEST
BIRDTOCA;CONF1;BIRDVGIA
DAAACTA;CONF1;DAAASC
DAAACTA;CONF1;DAAASE
DAAACTA;CONF1;DAAASS
DAAACTA;CONF1;DAAASW
DAAACTA;CONF1;DAAATC
DAAACTA;CONF1;DAAATNE
DAAACTA;CONF1;DAAATORAN
DAAATCTA;CNF1;DAAATNEX
DTTCCTA;CONF2;DTTCNS
DTTCCTA;CONF2;DTTCSS
```

9.7.14 conf - Conflict file

Conflict	
extension	conf
origin	produced by snapshot.exe with So6 file
separator	blank
sort	no
comment	each line contains 2 conflicting flights. This file is an input for conflict analysis: clh, conf2tdp, densities ...

#	Field	Type	Size	Comment
1	key of flight 1	num		field 17 from so6
2	key of flight 2	num		field 17 from so6
3	time first contact	num		in second, within a day must process modulo 86400 of this value
4	conflict duration	num		in second
5	FL of flight 1 first contact	float		when first contact
6	FL of flight 2 first contact	float		when first contact
7	lat 1	float		location of flight 1 when first contact, in minute decimal
8	lon 1	float		location of flight 1 when first contact, in minute decimal
9	lat 2	float		location of flight 2 when first contact, in minute decimal

10	lon 2	float		location of flight 2 when first contact, in minute decimal
11	distance first contact	float		square NM, decimal
12	distance minimum	float		square NM, decimal
13	distance last contact	float		square NM, decimal
14	status 1 fist contact	num		same as so6 file: 0=climb, 1=descent, 2=cruise, used for type of conflict
15	status 2 fist contact	num		same as so6 file: 0=climb, 1=descent, 2=cruise, used for type of conflict
16	point 1 of seg 1	char		name of point, route segment where flight 1 is located
17	point 2 of seg 1	char		name of point, route segment where flight 1 is located
18	point 1 of seg 2	char		name of point, route segment where flight 2 is located
19	point 2 of seg 2	char		name of point, route segment where flight 2 is located
20	azimuth seg 1	float		degrees decimal, used to determine type of conflict: parallel, crossing or opposite
21	azimuth seg 2	float		degrees decimal, used to determine type of conflict: parallel, crossing or opposite
22	Time end of contact ?	float		in second
23	FL of flight 1 end contact	float		FL
24	FL of flight 2 end contact	float		FL
25	lat 1	float		location of flight 1 when end contact, in minute decimal
26	lon 1	float		location of flight 1 when end contact, in minute decimal
27	lat 2	float		location of flight 2 when end contact, in minute decimal
28	lon 2	float		location of flight 2 when end contact, in minute decimal
29	time of minimum distance	num		in second, within a day must process modulo 86400 of this value
30	FL of flight 1 min distance	float		Flight Level of Flight 1 when minimum separation distance is reached
31	status 1 min distance	num		same as so6 file: 0=climb, 1=descent, 2=cruise, used for type of conflict
32	azimuth seg 1 min distance	float		degrees decimal, used to determine type of conflict: parallel, crossing or opposite
33	FL of flight 2 min distance	float		Flight Level of Flight 2 when minimum separation distance is reached
34	status 2 min	num		same as so6 file: 0=climb, 1=descent, 2=cruise,

	distance			used for type of conflict
35	azimuth seg 2 min	float		degrees decimal, used to determine type of distance conflict: parallel, crossing or opposite

Example (1 line):

```
105973632 105973659 1183077320 23 309.312500 317.047607 3016.445745 344.487062
3041.733769 348.262501...
... 645.253413 509.266157 750.890224 0 0 BULUX SOPOK PINUS LNO 135.40 208.62
1183077540 332.439026 334.212128...
... 3011.124959 388.244208 3016.377921 346.203506 1183077430 322.600006 0
104.68 325.545441 0 173.04
```

9.7.15 cos - Opening Scheme

NAME	OpeningScheme			
extensio	cos			
n:				
origin:	gassel, SAAM			
separato	(semicolon)			
r:				
sort:	by period name then by family name then by starting time			
commen				
t:				
FIRST LINE				
#	Field	Type	Size	Comment
1#	char		1	
2file type	text			= OPENING SCHEME
3version	num			= 1
4NM AIRAC cycle	num			
5AIRAC start date	date			format is yyyyMMdd
6AIRAC end date	date			format is yyyyMMdd
7number of record	num			
8datasource	text			
#	Field	Type	Size	Comment
1Date	date			format is dd/MM/yyyy
2ACC name	text			
3Start time	time			format is hh:mm
4End time	time			format is hh:mm
5Configuration name	text			
6Origin of information	char		1	E=Env T=Tact

Example:

```
#;OPENING_SCHEME;1;383;20140206;20140305;16556;EAR_P
06/02/2014;BIRDCTA;00:00;23:59;CONF1;E
06/02/2014;BIRDDICTA;00:00;23:59;CNF1;E
06/02/2014;BIRDTOCA;00:00;23:59;CONF1;E
06/02/2014;DAAACTA;00:00;23:59;CONF1;E
06/02/2014;DAAATCTA;00:00;23:59;CNF1;E
06/02/2014;DTTCCTA;00:00;23:59;CONF2;E
06/02/2014;EBBUCTA;00:00;05:39;CE1W1;E
06/02/2014;EBBUCTA;05:40;06:19;CE2W2H;E
06/02/2014;EBBUCTA;06:20;07:59;C3HW3;E
06/02/2014;EBBUCTA;08:00;10:19;C3EW3;E
06/02/2014;EBBUCTA;10:20;15:29;CE2W2H;E
06/02/2014;EBBUCTA;15:30;17:59;C3HW3;E
06/02/2014;EBBUCTA;18:00;19:59;C3EW2H;E
06/02/2014;EBBUCTA;20:00;21:59;CE2W1;E
06/02/2014;EBBUCTA;22:00;23:59;CE1W1;E
06/02/2014;EDDDCTA;00:00;23:59;CONF1;E
```

9.7.16 cost - Route Charge Cost file

Route Charge Cost	
version	1
extension	cost
origin	SAAM from Route Charge Module
separator	space
sort	by flight ID then by Country code
comment	Provide Route Charge Cost per flight per crossed country in EURO

#	Field	Type	Size	Comment
1	flight ID	num		Unique flight ID coming from input so6/t5 file
2	country code	char	2	ICAO code
3	cost	num		Route Charge Cost in Euro with at least 2 decimals (6 significant digits)

Example:

```
139729486 ED 356.159
139729486 LB 230.25
139729486 LC 86.7351
139729486 LH 148.107
139729486 LO 234.767
139729486 LT 365.319
139729486 LY 190.189
```

```

139729486 X2 19.3752
139729487 ED 48.1162
139729487 LH 181.893
139729487 LO 242.215
139729487 LR 385.217
139729487 UG 204.097
139729487 UK 262.747

```

9.7.17 crco - CRCO Intersection file

CRCO Intersection	
version	3
extensio n	crco
origin	SAAM from Route Charge Module
separator	blank
sort	flight id and chronological in flight
comment	Intersection summary with distance according to CRCO rules. Built on so6 and t5.

#	Field	Type	Size	Comment
1	country name	char		name is coming from T5 file sector name
2	callsign	char		
3	ICAO aircraft type	char		ICAO code
4	flight id	char		identifies flight in so6
5	entry lat	float		in minutes with 4 decimals
6	entry lon	float		in minutes with 4 decimals
7	exit lat	float		in minutes with 4 decimals
8	exit lon	float		in minutes with 4 decimals
9	distance	float		in kilometres with 8 decimals, 20km correction may have been applied
10	entry time	hhm mss		time of entry HHMMSS
11	entry date	yym mdd		date of entry YYMMDD
12	entry level	int		level of entry in FL with 3 decimals
13	exit time	hhm mss		time of exit HHMMSS
14	exit date	yym mdd		date of exit YYMMDD
15	exit level	int		level of exit in FL in FL with 3 decimals

Example :

```

ED N182QS B737 1523 3217.8167 599.3000 3018.8009 726.39836 396.03131430 090500
020322 0.000 095918 020322 270.000
LK N182QS B737 1523 3018.8009 726.3983 3019.0000 726.9864 0.78728520 095918 020322
270.000 095922 020322 270.000
LK N182QS B737 1523 3018.8009 726.3983 3019.0000 726.9864 0.78728520 111918 020322
270.000 111922 020322 270.000
ED N182QS B737 1523 3019.0000 726.9864 3217.8167 599.3000 395.93214016 111922
020322 270.000 121700 020322 0.000

```

9.7.18 eep - Entry Exit Point

EEP (Entry Exit Point)	
extension	eep
origin	Manual user input to specify entry/exit points (gates) in a Free-Route project area
separator	space
sort	no sort
comment	Each line defines the entry or exit points for a flow or a set of flows overpassing specific points. Blank lines are possible. It helps users to group flows into Free-routes "gates".

#	Field	Type	Size	Comment
1	entry/exit tag	char		"I" for Input, "O" for output (this code is obsolete). Note: // is used for comment
2	flow origin	char		ICAO Airport name, can be "*" for any airport, a "?" replace any letter. Airport group can be created like: EDDF EDDL
3	flow destination	char		ICAO Airport name, can be "*" for any airport, a "?" replace any letter. Airport group can be created like: EDDF EDDL
4	entry/exit point name	char		entry/exit in free route area will be done via this point (it must exist)
5	route point names	char		list of point names, space separated, which individually gives a route condition to the flow to be routed to the entry/exit point

Example:

```

//DEP EDDF added by TCH 02/03/2005 (this is a comment)
I EDDF * RUDOT BITBU DIK MAKIK
I EDDF * DITAM ABAXA MASIR
O EDDF * KOLAG MIMVA RAVLO

```

Note, how to interpret this example:

Departures from EDDF must:
"- enter the area via RUDOT if they cross BITBU or DIK or MAKIK"

"- enter the area via DITAM if they cross ABAXA or MASIR"
 "- exit the area via KOLAG if they cross MIMVA or RAVLO"

9.7.19 equi

equi				
extension	equi			
#	Field	Type	Size	Comment
1	flow ID	num		initialised with first flight ID (smallest) of the flow
2	flight ID	num		as found in expand file
3	time departure	num	4	HHMM padded with 0
4	date departure	num	6	YYMMDD padded with 0
5	callsign	char		
6	origin	char	4	ICAO code
7	destin	char	4	ICAO code
8	aircraft type	char		
9	RFL	num		

Example (10 flights / 7 flows):

```

41 41 1335 050630 PLK225 ULLI EKCH T154 340
51 51 1501 050630 RGL231 LPPT GMMN B190 210
61 61 1944 050630 RAM533 GMMN DIAP B738 350
101 101 0638 050630 LBC220 LATI LIRF B461 280
121 121 0638 050630 REA03CK EIDW EICK AT43 180
1061 1061 2224 050630 ADR650 LJLJ LTBA A320 370
121 151 1043 050630 REA15CK EIDW EICK AT72 180
41 221 1334 050630 PLK225 ULLI EKCH T154 340
51 231 1454 050630 RGL231 LPPT GMMN B190 210
241 241 0217 050701 EXS6498 EDDK LIME A30B 290

```

9.7.20 exp - Expand file

Expand	
extension	exp
origin	traffic demand data
separator	blank

sort	no			
comment	input for SAAM modeling (assignment ...)			
#	Field	Type	Size	Comment
1	origin	char		ICAO code (airport)
2	destination	char	4	ICAO code (airport)
3	not used	num	2	"oo"
4	aircraft type	char	4	
5	RFL	num	3	Warning: generally the value is calculated based on the maximum flight level reached.
6	zone origin	char	4	ICAO code (could be the same as airport)
7	zone destin	char	4	ICAO code (could be the same as airport)
8	flight ID	num		SAMAD ID or SAAM ID or ... (must be uniq)
9	date departure	num	6	YYMMDD
10	time departure	num	4	HHMM
11	time arrival	num	4	NOT USED = "!!!!"
12	callsign	char	till 7	
13	company	char	3	generally the 3 first letters of the callsign

Example (10 flights):

```
BGSF EKCH 00 B767 290 BGSF EKCH 367836 980626 1445 !!!! SAS292 SAS
BIKF EDDF 00 B757 370 BIKF EDDF 328474 980626 0745 !!!! ICE520 ICE
BIKF EDDK 00 CL60 290 BIKF EDDK 371896 980626 1516 !!!! GAF208 GAF
BIKF EDDK 00 B73S 270 BIKF EDDK 337460 980626 1740 !!!! ICE774 ICE
BIKF EDDL 00 B757 370 BIKF EDDL 346578 980626 0105 !!!! LTU1929 LTU
BIKF EDDM 00 B757 350 BIKF EDDM 351193 980626 0145 !!!! CMM759 CMM
BIKF EFHK 00 B73S 330 BIKF EFHK 328169 980626 0730 !!!! ICE342 ICE
BIKF EGLL 00 B757 290 BIKF EGLL 364422 980626 0750 !!!! ICE450 ICE
BIKF EGPF 00 B73S 290 BIKF EGPF 364457 980626 1200 !!!! ICE430 ICE
BIKF EGUN 00 C130 210 BIKF EGSS 366363 980626 0800 !!!! BLIGH49 ZZZ
```

9.7.21 exp2

Exp2 is a combination of Expand file (exp) and Flight Info file (flf)

Expand	
extension	exp
origin	traffic demand data
separator	semicolon ";"

sort	no
comment	input for SAAM modeling (assignment ...)

#	Field	Type	Size	Comment
1	origin	char		ICAO code (airport)
2	destination	char	4	ICAO code (airport)
3	not used	num	2	"oo"
4	aircraft type	char	4	
5	RFL	num	3	Generally extracted from filled flight plan (not calculated)
6	zone origin	char	4	ICAO code (could be the same as airport)
7	zone destin	char	4	ICAO code (could be the same as airport)
8	flight ID	num		SAMAD ID or SAAM ID or ... (must be uniq)
9	date departure	num	6	YYMMDD
10	time departure	num	4	HHMM
11	time arrival	num	4	NOT USED = "!!!!"
12	callsign	char	till 7	
13	company	char	3	generally the 3 first letters of the callsign

Example (10 flights):

```
BGSF EKCH 00 B767 290 BGSF EKCH 367836 980626 1445 !!!! SAS292 SAS
BIKF EDDF 00 B757 370 BIKF EDDF 328474 980626 0745 !!!! ICE520 ICE
BIKF EDDK 00 CL60 290 BIKF EDDK 371896 980626 1516 !!!! GAF208 GAF
BIKF EDDK 00 B73S 270 BIKF EDDK 337460 980626 1740 !!!! ICE774 ICE
BIKF EDDL 00 B757 370 BIKF EDDL 346578 980626 0105 !!!! LTU1929 LTU
BIKF EDDM 00 B757 350 BIKF EDDM 351193 980626 0145 !!!! CMM759 CMM
BIKF EFHK 00 B73S 330 BIKF EFHK 328169 980626 0730 !!!! ICE342 ICE
BIKF EGLL 00 B757 290 BIKF EGLL 364422 980626 0750 !!!! ICE450 ICE
BIKF EGPF 00 B73S 290 BIKF EGPF 364457 980626 1200 !!!! ICE430 ICE
BIKF EGUN 00 C130 210 BIKF EGSS 366363 980626 0800 !!!! BLIGH49 ZZZ
```

Flight inFo	
extensio n	flf
origin	cfmu
separato r	blank
sort	no
commen t	input for SAAM traffic filtering and trajectories matching

#	Field	Type	Size	Comment
1	Flight SAAM ID	num		uniq for the day

2	Flight SAMAD ID	num		uniq for ever
3	TACT ID	num		uniq for ever, CFMU ID
4	SSR_CODE	char		
5	REGISTRATION	char		
6	Planned Date departure	num	6	YYMMDD padded with o's
7	Planned Time departure	num	4	HHMM padded with o's
8	ATFM DELAY	num		
9	REROUTING STATE	char		
10	most pen reg	char		if no regulation 'X' is present
11	type	char		letters indicating type of flight (military ?)
12	equipment	char		letters indicating equipment of flight
13	ICAO equipment	char		
14	COM equipment	char		
15	NAV equipment	char		
16	SSR equipment	char		
17	SURVIVAL equip.	char		
18	PERSONS ON BOARD	num		o means no information
19	top FL	num		
20	max RFL	num		
21	FLT PLN SOURCE	char		

Example:

```

1 118814379 0086774620080901 BB8093375720080901 @ RA86926 080901 0040 0 @ @ N SRWY @ @ S 340 340 FPL
2 118814380 0086257920080901 AA6817941420080901 @ OOJAM 080901 0040 0 @ @ S SFHRWY @ @ S 370 370 FPL
3 118814381 0087092420080901 AA6818588420080901 @ N795AN 080901 0040 0 @ @ S SHIJRWXYZ @ RNVD1E2A1 SD 410 410 MFS
4 118814382 0086115020080901 BB8092860920080901 @ GFCLG 080901 0040 0 @ @ N SDHIRWXYZ @ RNP10 RNP5 S 380 380 FPL
5 118814383 0087102220080901 BB8093626320080901 @ SUGBE 080901 0040 0 @ @ S SDEPRWY @ @ S 370 370 FPL
6 118814384 0083939520080901 BB8092043720080901 @ DAXLF 080901 0045 0 @ @ N SDHIRWXYZ TCAS @ S 380 380 FPL
7 118814385 0086830220080901 AA6818393420080901 @ GBNLN 080901 0045 0 @ @ S SIPRWXYZ @ RNVD1E2A1 RNP10 S 370 370 MFS
8 118814386 0087086220080901 BB8093614520080901 @ GENLJ 080901 0045 0 @ @ S SIPRWXYZ @ RNVD1E2A1 RNP10 S 370 370 MFS
9 118814387 0085611020080901 BB8092448320080901 @ TCOGJ 080901 0045 0 @ @ S SDEHILRWY @ @ S 340 340 FPL
10 118814388 0086824920080901 AA6818390020080901 @ FGLZI 080901 0045 0 @ @ S SJPRWXYZ @ RNVD1E2A1 SD 370 370 FNM
11 118814389 0086215520080901 BB8092950520080901 @ GBYAO 080901 0045 0 @ @ S SEHIRWXYZ @ RNP10 S 360 380 FPL
12 118814390 0086258720080901 BB8092987920080901 @ ECKHU 080901 0045 0 @ @ S SJRWXYZ @ RNP10 C 380 380 FPL
13 118814391 0086298020080901 AA6817971720080901 @ PHSHL 080901 0045 0 @ @ N SGZ @ TCAS S 30 30 FPL
14 118814392 0086840920080901 BB8093430420080901 @ ECJVO 080901 0045 0 @ @ N SDGHRY @ @ S 280 280 FPL
15 118814393 0087017720080901 BB8093575320080901 @ JYAIIC 080901 0045 0 @ @ S SHIRWXY @ @ C 380 380 FPL
16 118814394 0086867220080901 BB8093449320080901 @ CSTGU 080901 0045 0 @ @ N SDHIRWXY @ @ C 370 370 FNM
17 118814395 0086749620080901 AA6818328220080901 @ PHBFE 080901 0045 0 @ @ S SDGHIJPRWXYZ @ RNAV1 RNAV5 RNP10 R S 350 350 MFS
18 118814396 0086825720080901 BB8093419220080901 @ TCJHA 080901 0045 0 @ @ S SDHIPRWXY @ @ S 360 360 FPL
19 118814397 0086885320080901 BB8093464120080901 @ ZKNBV 080901 0045 0 @ @ S SGHIJPRWYZ @ RNP4 RNP10 RNAV1 RNA SD 381 381 FPL
20 118814398 0086901520080901 AA6818445920080901 @ GCIVZ 080901 0045 0 @ @ S SIPRWXYZ @ RNVD1E2A1 RNP10 S 350 370 MFS
21 118814399 0086690120080901 AA6818274320080901 @ DATUF 080901 0045 0 @ @ S SFGIRWXY @ @ S 370 370 FPL

```

EXP2:

	#E XP	#L oc	Field	Type	Comment
--	----------	----------	-------	------	---------

	2				
EXP	1	1	origin (ADEP)	char[4]	ICAO code (airport) International Civil Aviation Organisation
	2	2	destination (ADES)	char[4]	ICAO code (airport)
	3	3	not used	Numeric[2]	"oo"
	4	4	aircraft type	char[4]	
	5	5	RFL	Numeric[3]	Requested Flight level
	6	6	zone origin	char[4]	ICAO code (could be the same as airport)
	7	7	zone destin	char[4]	ICAO code (could be the same as airport)
	8	8	flight ID	Numeric	SAMAD ID or SAAM ID or ... (must be uniq)
	9	9	date departure	Numeric[6]	YYMMDD (ETD not EOBT)
	10	10	time departure	Numeric[4]	HHMM (ETD not EOBT)
	11	11	time arrival	Numeric[4]	NOT USED = "!!!!"
	12	12	callsign	char[<=7]	
	13	13	company	char[3]	generally the 3 first letters of the callsign
SEPARATOR	14	NA	Separator	Numeric[1]	value is o
UUID	15	NA	Universal Unique ID	char[TO DO]	CALLSIGN-ADEP-ADES-EOBD-EOBT
	16	NA	Fips cloned	Alphabetic[1]	Y, N
SEPARATOR	17	NA	Separator	Numeric[1]	value is o
FLF	18	1	Flight SAAM ID	Numeric	uniq for the day
	19	2	Flight SAMAD ID	Numeric	uniq for ever
	20	3	TACT ID	Numeric	CFMU ID : Check with TACTID on ALL-FT
	21	4	SSR_CODE	char	plan identifiant (12bits mode A), Secondary Surveillance Radar
	22	5	REGISTRATION	char	
	23	6	Planned Date departure	Numeric[6]	YYMMDD padded with o's
	24	7	Planned Time departure	Numeric[4]	HHMM padded with o's
	25	8	ATFM DELAY	Numeric	Air Traffic Flow Management Delay : Difference between calculated by CASA take off time (CTOT) and estimated take off time (ETOT).;
	26	9	REROUTING	char	

		STATE		
	27 10	most pen reg	char	if no regulation 'X' is present : Check with mostPenalizingRegulationId
	28 11	type	char	letters indicating type of flight (military ?)
	29 12	equipment	char	letters indicating equipment of flight
	30 13	ICAO equipment	char	International
	31 14	COM equipment	char	
	32 15	NAV equipment	char	
	33 16	SSR equipment	char	Secondary Surveillance Radar equipement
	34 17	SURVIVAL equip.	char	
	35 18	PERSONS ON BOARD	Numeric	0 means no information
	36 19	top FL	Numeric	
	37 20	max RFL	Numeric	Requested Flight Level
	38 21	FLT PLN SOURCE	char	
SEPARATOR	39 NA	Separator	Numeric[1]	value is 0
ALLFT	40 6	aobt	Numeric[14] YYYYMMDDHHMMSS	Actual Of Block in Time
	41 7	ifpsId	Alphabetic[2] Digit[8] XX12345 678	Prisme (Integrated Initial Flight Plan Processing System)
	42 8	iobt	Numeric[14] YYYYMMDDHHMMSS	Initial Of Block in Time FLF (planned time departure)
	43 9	originalFlightData Quality	Alphabetic[3]	NON, PFD, RPL, FPL
	44 10	flightDataQuality	Alphabetic[3]	FLF (flt pln source) , NON, PFD, RPL, FPL
	45 11	source	Alphabetic[3]	UNK, FPL, RPL, AFI, MFS, FNM, AFP, DIV
	46 13	exemptionReason Type	Alphanumeric[4]	NEXE, EMER, SERE, HEAD, AEAP, empty
	47 14	exemptionReason Distance	Alphanumeric[4]	NEXE, LONG, empty Note : in NEST the flight exemption is limited to exempted or not exempted states. Not exempted corresponds to NEXE in fields 46 and 47 . All other values correspond to exempted. During export, NEST writes empty fields in 46 and 47 for exempted flights.

48	15	lateFiler	Alphabetic[1]	Y, N
49	16	lateUpdater	Alphabetic[1]	Y, N
50	17	northAtlanticFlight	Alphabetic[1]	Y, N
51	18	cobt	Numeric[14]	Computed Of Block in Time YYYYMMDDHHMMSS
52	19	eobt	Numeric[14]	Estimated Of Block in Time YYYYMMDDHHMMSS
53	20	flightState	Alphanumeric[2]	NE, PL, PS, PR, SR, FI, FS, SI, TA, AA, CA, TE
54	21	previousToActivationFlightState	Alphanumeric[2]	NE, PL, PS, PR, SR, FI, FS, SI, TA, AA, CA, TE
55	22	suspensionStatus	Alphanumeric[2]	NS, ST, SM, RC, TV, NR, RV
56	23	tactId	Numeric[6]	FLF (tact id) (is it the same as tactid on FLF (3))
57	24	samCtot	Numeric[14]	YYYYMMDDHHMMSS
58	25	samSent	Alphabetic[1]	Y, N
59	26	sipCtot	Numeric[14]	YYYYMMDDHHMMSS
60	27	sipSent	Alphabetic[1]	Y, N
61	28	slotForced	Alphabetic[1]	Y, N
62	29	mostPenalizingRegulationId	Alphanumeric[8]	Is it the same as most pen reg on FLF (10)
63	30	regulationsAffectedByNrOfInstances	Numeric[4]	
64	31	excludedFromNrOfInstances	Numeric[4]	
65	32	lastReceivedAtfmMessageTitle	Alphabetic[3]	DES, ERR, FCM, FUM, FLS, REA, RFI, RJT, RRN, RRP, SAM, SIP, SLC, SMM, SPA, SRJ, SRM, SWM, UNK
66	33	lastReceivedMessageTitle	Alphabetic[3]	ABI, ACH, ACT, APL, ARR, CAN, CHG, CNL, DEP, DLA, ERR, EST, FPL, FSA, MFS, PAC, PFD, RPL, UNK
67	34	lastSentAtfmMessageTitle	Alphabetic[3]	DES, ERR, FCM, FUM, FLS, REA, RFI, RJT, RRN, RRP, SAM, SIP, SLC, SMM, SPA, SRJ, SRM, SWM, UNK
68	41	manualExemptionReason	Alphanumeric[1]	N, S, R
69	42	sensitiveFlight	Alphabetic[1]	Y, N
70	43	readyForImprovement	Alphabetic[1]	Y, N

71	44	readyToDepart	Alphabetic[1]	Y, N
72	45	revisedTaxiTime	Numeric[6]	oooooooo..999999
73	46	tis	Numeric[6]	Time to Insert the Sequence , oooooooo..999999
74	47	trs	Numeric[6]	Time to Remove from Sequence , oooooooo..999999
75	48	toBeSentSlotMessageTitle	Alphabetic[3]	Related to flight progress , ABI, ACH, ACT, APL, ARR, CAN, CHG, CNL, DEP, DLA, ERR, EST, FPL, FSA, MFS, PAC, PFD, RPL, UNK
76	49	toBeSentProposalMessageTitle	Alphabetic[3]	Related to flight progress , ABI, ACH, ACT, APL, ARR, CAN, CHG, CNL, DEP, DLA, ERR, EST, FPL, FSA, MFS, PAC, PFD, RPL, UNK
77	50	lastSentSlotMessageTitle	Alphabetic[3]	Related to flight progress , ABI, ACH, ACT, APL, ARR, CAN, CHG, CNL, DEP, DLA, ERR, EST, FPL, FSA, MFS, PAC, PFD, RPL, UNK
78	51	lastSentProposalMessageTitle	Alphabetic[3]	Related to flight progress , ABI, ACH, ACT, APL, ARR, CAN, CHG, CNL, DEP, DLA, ERR, EST, FPL, FSA, MFS, PAC, PFD, RPL, UNK
79	52	lastSentSlotMessage	Numeric[14]	Related to flight progress , YYYYMMDDHHMMSS
80	53	lastSentProposalMessage	Numeric[14]	Related to flight progress , YYYYMMDDHHMMSS
81	54	flightCountOption	Alphabetic[1]	Indicates which flight plan should be / has been used when doing flight/count related operations. Used in TACT queries and replies , P, N
82	55	normalFlightTactId	Numeric[6]	
83	56	proposalFlightTactId	Numeric[6]	
84	57	operatingAircraftOperatorIcaoId	Alphabetic[3]	
85	58	reroutingWhy	Alphabetic[1]	N, M, C, A, O
86	59	reroutedFlightStatue	Alphabetic[1]	P, E, T, R, V, N
87	60	runwayVisualRange	Numeric[3]	
88	70	ftfmAiracCycleReleaseNumber	Numeric[4]	Filed Traffic Flight Model (TFM = Profile) airac..
89	71	ftfmEnvBaselineNumber	Numeric[4]	Filed Traffic Flight Model (TFM = Profile) env...
90	72	rtfmAiracCycleReleaseNumber	Numeric[4]	Regulated Traffic Flight Model (TFM = Profile) airac..

	91	73	rtfmEnvBaselineNumber	Numeric[4]	Regulated Traffic Flight Model (TFM = Profile) env...
	92	74	ctfmAiracCycleReleaseNumber	Numeric[4]	Computed Traffic Flight Model (TFM = Profile) airac..
	93	75	ctfmEnvBaselineNumber	Numeric[4]	Computed Traffic Flight Model (TFM = Profile) env...
	94	76	lastReceivedProgramMessage	Alphabetic[3..5]	DPI, EMPTY, SIZE
SEPARATOR	95	NA	Separator	Numeric[1]	value is 0

Example:

odnaRzzfdttccSUFSFFTS RP PAR nt eICNSSP tnFSaiiofseelncefp stsssssmr el 11 msrrrtt t 1111 fnpo rrrftfrrccls
 reoi Fool aiiaoeniellAS El 1 TE oyqCCASUEoaLeofor loxx aaooolr uaaiiloexaa aaaaeeei roooaaalorpeeufttttftae
 istrLmtnmlnpi ppi CRCa a FRspuAVRRpxTpbphi ueetrbbiescnpposg c s s s nnavssBBssssiroe rrnmffmfsp
 gtuc eegeee pavaggT_InnOteiCeeeVSFRPatstgrmmetttgvptCSCSttul tttusddi eettttgnpr oowAmrnA nt a
 ii sr odhddasarecrhhICSn nUp peqqqlOLFLR I i hepp Fh hioeIteteFPlauRRSaiyys SSSSSSShaoa uuai EAEi ER
 mea reteerinarlattDOTe eFT e nquuVN lNa dntet tijpa tu ndononoe tidee el tFle eeee eetlsttityr ni nr nea
 (adfsIpprgtsotSS DRdLI n euuiiAS St aIioldt Ss stttrno ec c nEiod nn nn nn CFan ieVa vr va vct
 At t gtLaain oa noAA EADTANr nippLO Co l a on eal tTi can de et xvrII ttttttollgndicBaBcBeo
 li t ii rrv rl erAM Ta i YGe tpmmeN Ur Ft nRrta ao o elis Fi i Ae e lea SPSPSPuFAgFsCacAcair
 Fo y m tta Ud MA It m S g neeqB R lia Re en tAn dz Ar v vt mfpix lrlrlrlngi Vluy sCs y sv
 Pn p uul n ID Cee T ennnuO C gC ea rt ec S i ff oe ef plpai ooo oo oothir hiaceyeceee
) (e rr i II Ndd A ntttiA E hu as i tit ne md dmt irrT tptptpCtgc ygl lcll1d
 A ee q D ee T t pR t a so c v a gctNAMMi goti Mo Mo Mo pIhr hRe ilie i P
 D u pp E .D Dl on F a t Re rtf e e o h v n es es estata taRne nRnr
 E e a a ai nD l tiu ed Oms s nte e sasa saicTft S ne eRe eo
 S I r r tt Tis i os g B fIMs s R n s l s l otaO tgl Ne Nl Ng
) D t t ay yt g n uy ne a a e a MaMaMnIcp aee ul ue ur
 uu Q pa h F l N s s g g a n ge ge ge dt e t am e ma me
 rr u en t li ar t se es t es es es Ir e s babs bs
 ee a c g t O a a T To Ts Ts s da e e s e e e s
 li e h i fl n g i i n ia ia a a t Nr e rNr M
 t t on c ett t g t g g o u N u e
 y S nsteTll le le e r mu m s
 t I a sitee e Te T I b m b s
 a dn l it it c e b e a
 t c e l l a r e r g
 e es e e o r e
 I d

EEoP2EE391!SSo Yo1181 R9 1 0P rSGi crss o45p09 X9NNUNNY99NN N89Y9Yr 1 1 DADS YY11AA AA 99 P88S NP10 00 00 0D0
 GK079GK684!AA 162 840 e coasu od 8X8OCNEE 88EE S68 8 e 4 4 EBE 444BB BB 88 66A 21 11 11 1P
 SC 60SC704!SS 874 044 g anvrr oon 010NMX 00 70 0 g 2 2 SIS 555I I I I 00 77S 52 22 22 2I
 FH 7 FH865!2 875 650 u oeeeev s 626 KEE 66 76 6 u 5 5 222 66 77 3 33 33 3
 32 9 14 2 0 eqqge 232 22 42 2 0 333 22 44
 66 2 46 6 1 quuuq 646 66 66 6 1 666 66 66
 3 0 uiuiu 151 11 1 1 0 11
 7 2 ipppi 464 44 4 4 2 44
 9 p p 474 44 4 4 44
 585 55 5 5 55
 3 3 33 3 3 33
 5 5 55 5 5 55
 EEoC3EE390!II0 No2181 R9 0 0ErNi crss 1 45p09 X9P PFEL NN99PP S89N9N 1 1 EAERNM11AA AA 99 N88I NE20 00 00 0D0
 IDoL7I1E28!CC 164 870 e coasu od 8X8FPMO 88LL T68 8 e 4 4 RCR 555CCC 88 66C 12 22 22 2P
 KD 60KE804!EE 875 044 g anvrr oon 010DELEN 00 70 0 g 2 2 RHR 888HHHHoo 77E 41 11 11 1I
 FK o FK465!5 876 655 u oeeeev s 626 RG 66 76 6 u 6 6 999 66 77 5 55 55 5

72 2	14 2	0 eqqqe	232	22	42 2 0	666	22 44
46 0	47 6	1 quuuq	646	66	76 6 1	222	66 77
	3	0 uiuiu	050	00	0 0 0		00
	8	3 ipppi	767	77	7 7 3		77
	0	p p	474	44	4 4		44
			595	55	5 5		55
			0 0	00	0 0		00
			0 0	00	0 0		00
EEoB2EE291!CCo	Y03181 R9 1 oT rGfi crss 2 45p09 x9R RRS NYY99PP S89Y9Yr 1 1 F AFS NNN11AA AA 99 P88G Ct30 oo oo oEo						
CG079CC885!AA	166 8 5 0 e coasu ool 8X8F FFEE	88SS M68 8 e 4 4 CCC	777CC CC 88 66A o3 3333 3P				
SL 30SL901!FF	876 0 1 5 g anvrr oon 01oP PLRX	00 70 0 g 2 2 MTM	222TT TT oo 77F 30 oo oo oI				
FL S FL166!2	877 6 6 0 u oeeev s 626 EE	66 76 6 u7 7	666 66 77 777777				
12 0	14 2 0 eqqqe	232	22	42 2 0	888	22 44	
26 8	48 6 1 quuuq	646	66	86 6 1	888	66 88	
	3 0 uiuiu	151	11	1 1 0		11	
	8 4 ipppi	565	55	5 5 4		55	
	1 p p	181	11	1 1		11	
		606	66	6 6		66	
		3 3	33	3 3		33	
		6 6	66	6 6		66	
EEoB3EE291!IIo	N04181 R9 1 oRrMi crss 3 45p09 x9F FAHL NNN99PP R89N9Nr 1 1 F AF RYYY11AA AA 99 N88I AF30 oo oo oEo						
IG075IC487!CC	168 8 7 0 e coasu ool 8X8P PFEO	88FR C68 8 e 4 4 UPU	888P P PP 88 66C 93 3333 3N				
KP 50KP904!EE	877 0 4 4 g anvrr oon 01oL LIAN	00 70 0 g 2 2 MLM	666LL LL oo 77E 29 99 99 9P				
FF 7 FF760!7	878 6 0 0 u oeeev s 626 DG	66 76 6 u8 8	444 66 77 9 99 99 9T				
52 7	14 2 0 eqqqe	232	22	42 2 0	111	22 44	Y
06 4	49 6 1 quuuq	646	66	96 6 1	444	66 99	
	3 0 uiuiu	151	11	1 1 0		11	
	8 5 ipppi	767	77	7 7 5		77	
	2 p p	484	44	4 4		44	
		010	00	0 0		00	
		4 4	44	4 4		44	
		5 5	55	5 5		55	
EEoC2EE290!LLo	N05182 R9 0 oVrXi crss 4 45p09 x9NNMANYY99SS T89Y9Yr 1 1 F AFS YYY222AA AA 99 P88L Cv40 oo oo oEo						
CK011GK181!TI	160 8 1 0 e coasu ool 8X8OCFEE	88FR V68 8 e 4 4 L RL	000RR RR 88 66T 84 4444 4N				
SC 30SCoo! UL	878 0 0 4 g anvrr oon 01oNNSAX	00 70 0 g 2 2 S RS	000RR RR oo 77U 19 99 99 9P				
FH o FL365!1	879 6 5 5 u oeeev s 626 PE	66 76 6 u9 9	111 66 77 1 1 1 1 1 1 T				
82 9	15 2 0 eqqqe	232	22	52 2 0	444	22 55	
86 2	40 6 1 quuuq	646	66	06 6 1	000	66 00	
9	3 0 uiuiu	050	00	0 0 0		00	
	8 6 ipppi	161	11	1 1 6		11	
	3 p p	080	00	0 0		00	
		525	55	5 5		55	
		4 4	44	4 4		44	
		8 8	88	8 8		88	
EEoB2EE190!CCo	N06182 R9 0 oNrSI i crss 5 45p09 x9P PFNL NNN99FF N89N9Nr 1 1 RCRRNNN222CC CC 99 N88C N50 oo oo oEo						
IDC74I781!M	163 8 1 0 e coasu ool 8X8F FNEO	88II R68 8 e 4 4 EAE	111AAA AA 88 66M 75 55 55 5M				
KD 68KL04!M	870 0 4 5 g anvrr oon 01oDIMXN	00 70 0 g 3 3 ANA	333NNNN oo 77M 08 88 88 8P				
FK 8 FK065!7	870 6 5 0 u oeeev s 626 EG	66 76 6 u0 0	888 66 77 3 3 3 3 3 T				
22 5	15 2 0 eqqqe	232	22	52 2 0	666	22 55	
66 9	41 6 1 quuuq	646	66	16 6 1	666	66 11	
	3 0 uiuiu	050	00	0 0 0		00	
	8 7 ipppi	161	11	1 1 7		11	
	4 p p	484	44	4 4		44	
		535	55	5 5		55	
		1 1	11	1 1		11	
		0 0	00	0 0		00	
EEoC2EE190!IIo	N07182 R9 0 oP rNi crss 6 45p09 x9R RAENYY99FF R89Y9Yr 1 1 RCRS NNN222CC CC 99 P88I MP60 oo oo oSo						
CG0L3CC387!CC	165 8 7 0 e coasu ool 8X8F FFME	88SS V68 8 e 4 4 F HF	222HHHH88 66C 56 66 66 6I				
SL 60SL103!EE	871 0 3 4 g anvrr oon 01oP PPFX	00 70 0 g 3 3 I GI	777GGGG oo 77E 97 77 77 7Z				
FL 1 FL660!3	871 6 0 0 u oeeev s 626 RE	66 76 6 u1 1	555 66 77 5 5 5 5 5 E				
62 4	15 2 0 eqqqe	232	22	52 2 0	999	22 55	

46 2	42 6	1 quuuq	646	66	26 6 1		222	66 22
	3	o uiuiu	050	00	0 0 0			00
	8	8 i pppi	767	77	7 7 8			77
	5	p p	383	33	3 3			33
			040	00	0 0			00
			1 1	11	1 1			11
			2 2	22	2 2			22
EE0B2EE390! T10 N08182 R9 o oE rGki crss7 45p09 x9F FLS L NNN99SS T89N9Nr 1 1 R CRYYYY22CC CC 99 N88T CF70 oo oo oSo								
IG071IC987! AA 167 870 e coasu odl 8X8PPIEO 88II V68 8 e 4 4 RNR 444NNNN88 66A 47 77777I								
KP 32KP205! PP 872 054 g anvrr oon 010LIVRN 00 70 0 g 3 3 NLN 111LL LL 00 77P 86 66666Z								
FF S FF360! 6 872 605 u oeeev s 626 EG 66 76 6 u 2 2 333 66 77 777777E								
o2 0 15 2 o eqgqe 232 22 52 2 0 111 22 55								
26 2 43 6 1 quuuq 646 66 36 6 1 888 66 33								
3 o uiuiu 050 00 0 0 0 00								
8 9 i pppi 767 77 7 7 9 77								
6 p p 585 55 5 5 55								
050 00 0 0 00								
5 5 55 5 5 55								
4 4 44 4 4 44								
EE0B1EE391! II0 N09182 R9 1 oRrMi crss8 45p09 x9NNUHNYYY99TT N89Y9Yr 1 1 R DRS YYY22DD DD 99 P88I AT80 oo oo oSo								
CK079GK582! CC 169 820 e coasu odl 8X80CNEE 88AA R68 8 e 4 4 RER 555EE EE 88 66C 38 88888 81								
SC 54SC200! EE 873 005 g anvrr oon 010NNMAX 00 70 0 g 3 3 P PPP 555PP PPP 00 77E 75 55555 5Z								
FH 8 FH960! 4 873 600 u oeeev s 626 KDE 66 76 6 u 3 3 000 66 77 99999 9E								
42 3 15 2 o eqgqe 232 22 52 2 0 444 22 55								
06 0 44 6 1 quuuq 646 66 46 6 1 444 66 44								
3 1 uiuiu 151 11 1 1 1 11								
8 o i pppi 262 22 2 2 0 22								
7 p p 080 00 0 0 00								
060 00 0 0 00								
4 4 44 4 4 44								
7 7 77 7 7 77 77								
EE0C1EE290! EZ0 Y01183 R9 o oT rXNi crss9 45p09 x9P PLAL NNN99AA R89N9Nr 1 1 S DS RNNM22EE EE 99 N88Z CF90 oo oo o 0								
ID017II188! LZ 0161 880 e coasu odl 8X8F FIEO 88AA V68 8 e 4 4 ALA 666RR RR 88 66Z 29 99999								
KD 36KF300! LZ 874 004 g anvrr oon 010DIVAN 00 70 0 g 3 3 MAM 888RR RR 00 77Z 65 55555								
FK 1 FK560! C 874 605 u oeeev s 626 PG 66 76 6 u 4 4 777 66 77 1 1 1 1 1								
72 F 15 2 o eqgqe 232 22 52 2 0 777 22 55								
86 4 45 6 1 quuuq 646 66 56 6 1 000 66 55								
6 3 1 uiuiu 050 00 0 0 1 00								
8 1 i pppi 868 88 8 8 1 88								
8 p p 080 00 0 0 00								
070 00 0 0 00								
2 2 22 2 2 22								
5 5 55 5 5 55 55								

9.7.22 fids

fids	
extensio	fids
n	
origin	
separator	semicolon
sort	no
comment	Used to match flight IDs from different sources for a given date, including PRISME ID, IFPS ID and TACT ID

#	Field	Type	Comment
---	-------	------	---------

1	PRISME ID	char	
2	IFPS ID	char	
3	TACT ID	char	

Example:

```
192471283;AA48238139;874290
192471069;AA48236304;874291
192471326;AA48239076;874294
192471342;AA48239146;874297
192470983;AA48218602;874299
192471340;AA48232673;874300
```

9.7.23 fld - Flow Demand file

Flow Demand				
extension	Fld			
origin	SAAM processing or CFMU data (via PRISME)			
separator	Blank (i.e. Space)			
sort	No			
comment	Used as an input in 3D density, forecast, transform fld to so6, etc ...			

#	Field	Type	Size	Comment
1	origin of flight	char	4	ICAO code
2	destination of flight	char	4	ICAO code
3	Number of flights	Num		
4	Lat 1	num		in minute decimal
6	Long 1	num		in minute decimal
7	Lat 2	num		in minute decimal
8	Long 2	num		in minute decimal

Example:

```
LEBL LEMD 21525 2477.8167 124.7000 2428.3333 -213.6500
LEMD LEBL 21405 2428.3333 -213.6500 2477.8167 124.7000
LIRF LIML 12728 2508.6667 735.1500 2727.0167 556.7667
LIML LIRF 12685 2727.0167 556.7667 2508.6667 735.1500
LFPO LFMN 12451 2923.4000 142.7667 2619.9167 432.9000
LFMN LFPO 12419 2619.9167 432.9000 2923.4000 142.7667
LFPO LFBO 12263 2923.4000 142.7667 2618.1000 82.0667
LFBO LFPO 12244 2618.1000 82.0667 2923.4000 142.7667
```

```

LEBL LEPA 11728 2477.8167 124.7000 2373.1000 164.3333
LEPA LEBL 11427 2373.1000 164.3333 2477.8167 124.7000
LFPO LFML 10921 2923.4000 142.7667 2606.2000 312.9000
LFML LFPO 10919 2606.2000 312.9000 2923.4000 142.7667

```

9.7.24 fif - Flight Info file

Flight inFo				
#	Field	Type	Size	Comment
1	Flight SAAM ID	num		uniq for the day
2	Flight SAMAD ID	num		uniq for ever
3	TACT ID	num		uniq for ever, CFMU ID
4	SSR_CODE	char		
5	REGISTRATIO N	char		
6	Planned Date departure	num	6	YYMMDD padded with o's
7	Planned Time departure	num	4	HHMM padded with o's
8	ATFM DELAY	num		
9	REROUTING STATE	char		
10	most pen reg	char		if no regulation 'X' is present
11	type	char		letters indicating type of flight (military ?)
12	equipment	char		letters indicating equipment of flight
13	ICAO equipment	char		
14	COM equipment	char		
15	NAV equipment	char		
16	SSR equipment	char		
17	SURVIVAL equip.	char		
18	PERSONS ON BOARD	num		o means no information
19	top FL	num		

20	max RFL	num		
21	FLT PLN SOURCE	char		

9.7.25 for - Statfor

STATFOR Demand Forecast File

Filename: "FFF_yyyyMM.for" (FFF: STATFOR "MTF" / "STF" forecast type, yyyy: year using four characters, MM: month using two characters)

Function: Defines the STATFOR forecast for the given time horizon (either Medium or Short term)

[First Line](#)

Field	Value
1	“RANK” STATFOR forecast growth hypothesis header
2	“Dep” departure forecast zone header (AKA airport family)
3	“Des” destination forecast zone header (AKA airport family)
4, 5, 6, ...	Forecast year header (number of fields depends on the number of forecasted years in the file – generally 7 years)

[Subsequent Lines](#)

Field	Value
1	Growth hypothesis (either “Base”, “Low” or “High”)
2	Departure forecast zone
3	Destination forecast zone
4, 5, 6, ...	Forecast year

Example:

RANK;Dep;Des;2009;2010;2011;2012;2013;2014;2015

Base;Albania;Austria;0.000;-9.052;-7.761;-6.244;-5.289;-3.992;-2.695

Base;Albania;Barcelona;0.000;99.839;99.839;99.887;99.839;99.839;99.839
 Base;Albania;Belgium;0.000;25.401;27.913;30.879;33.045;35.557;38.155
 Base;Albania;BerlinFIR:Berlin;0.000;168.423;168.423;168.485;168.360;168.360;168.329
 Base;Albania;BerlinFIR:BerlinSouth;0.000;70.422;70.422;70.540;70.422;70.422;70.422
 Base;Albania;Bosnia-Herzegovina;0.000;-5.826;-7.910;-9.481;-10.957;-11.766;-11.813
 Base;Albania;BremenFIR:BremenNorth;0.000;211.821;211.821;211.891;211.750;211.750;211.715
 Base;Albania;BremenFIR:BremenSouth;0.000;147.388;147.388;147.537;147.388;147.388;147.388;147.388
 Base;Albania;BremenFIR:Hamburg;0.000;179.622;179.622;179.738;179.564;179.564;179.535
 Base;Albania;Bulgaria;0.000;3.704;11.787;20.801;29.946;40.242;51.162

9.7.26 frp - Free Route Points file

Free Route Points	
extension	frp
origin	saam
separator	blank or tabulation
sort	no
comment	Used during assignment to generate Free Route segments . Multiple lines with different point type can exist for the same Area and Point name : one line for E/X/EX/I/EI/XI/EXI declaration, and additionally, one or two lines for A/D/AD declarations

E/X/I line:

#	Field	Type	Comment
1	Free Route Area name	char	name used in the associated sls file
2	Point type	char	E, X, EX, I, EI, XI, EXI
3	Point name	char	
4	Latitude	N Sddmmss	degrees, minutes, seconds. optional if the point is already existing in network *.ase files.
5	Longitude	E Wdddmms	degrees, minutes, seconds. optional if the point is already existing in network *.ase files.

A/D line:

#	Field	Type	Comment
1	Free Route Area name	char	name used in the associated sls file
2	Point type	char	D, A, AD (Free route point types is extended by combining different types on several lines, see AMUGO example.)
3	Point name	char	if not type is not including E/X/I with coordinates, then the point must be present in ase network files
4.. N	Airport name	char	Airport restriction : Free route to or from the point allowed only with this Arrival/Departure set of airports. Name can contain '.' character to match any A-Z character. The name can also be an Airport group (see soa file) Remark : if a A/D Free route point has got different airport restriction lists depending on Arrival/Departure, then, this Free route point shall be defined on two separate lines (see TROIA example below)

Example:

```

LISBOA EX RETEN
LISBOA EX ARMED
LISBOA D CP
LISBOA A LAR
LISBOA AD POR
LISBOA A TROIA LPPT LPCT
LISBOA D TROIA LPPT
LISBOA AD GAIOS LPPT
LISBOA D DIRMA LP..
LISBOA AD SNT
SWEDENNORTH EX GIKAV
SWEDENNORTH EX SOLKA
SWEDENNORTH X TIGBA
SWEDENNORTH X OBIVO
SWEDENNORTH EX XELVI
SWEDENNORTH AD OSS
SWEDENNORTH AD SKE ESNO ESNU
SWEDENNORTH AD KRA
Irland EX MIMKU
Irland EX IBROD
Irland EX GOMUP
Irland I AMPED N513400 W0111300
Irland I UNLID N511233 W0104329
Irland I LINRA N513447 W0100156
LS X CROSS N470902 E0093614
LO E CROSS N470902 E0093614
SEAFRA I AMUGO
SEAFRA A AMUGO
SEAFRA D AMUGO LDZD

```

9.7.27 gar - Airblock

NAME:	Airblock			
extension	gar			
origin:	Gasel, SAAM			
separator	;	(semicolon)		
sort	no sort			
comment	Describe 2D definition of airblocks	:		
FIRST LINE				
#	Field	Type	Size	Comment
1#	char		1	
2file type	text			= AIRBLOCK
3version	num			= 1
4NM AIRAC cycle	num			
5AIRAC start date	date			format is yyyyMMdd
6AIRAC end date	date			format is yyyyMMdd
7number of record	num			
8datasource	text			
HEADER				
#	Field	Type	Size	Comment
1A	char		1	A=Airblock
2ID of the airblock	text			
3Nb of points	num			Nb of 2D points defining the border of th airblock = number of body lines
BODY				
#	Field	Type	Size	Comment
1P	char		1	P=Point
2Latitude	num			degree decimals
3Longitude	num			degree decimals

Example:

```
#;AIRBLOCK;1;383;20140206;20140305;5652;EAR_P
A;001BI;14
P;64.633333333333; -22.96666666666667
P;64.783333333333; -22.883333333333
P;64.716666666667; -22.46666666666667
P;64.7; -22.116666666667
```

```
P;64.666666666667;-21.3
P;64.283333333333;-20.95
P;63.866666666667;-21.283333333333
P;63.716666666667;-21.216666666667
P;63.476111111111;-21.646111111111
P;63.320277777778;-22.614444444444
P;63.558333333333;-23.77
P;64.215;-24.048611111111
P;64.4975;-23.600555555556
P;64.633333333333;-22.966666666667
```

9.7.28 gr1 - Sector Load Input Chart Data file

Sector Load input chart data form at	
extensio n	gr1
origin	SAAM sector load
separato r	horizontal tabulation (displayed as → in the example below)
sort	
commen t	<p>ascii file describing the data to be displayed in charts, used by SAAM chart and combine curve.</p> <p>A gr1 file may contains several charts data. Each chart is made of: "Chart Header line1" followed by "Chart Header Line 2" followed by "Chart Body" followed by "Chart Body End"</p> <p>Note: the number of columns depends on the number of series in the graphs.</p>

Global GR1 File Header line				
#	Field	Type	Size	Comment
1	numcha rt	num		number of charts described in the file (below)

Chart Header line 1 (TAB SEPARATED)				
#	Field	Type	Size	Comment
1		char	o	empty field
2	chart name	char		name of the chart (in general the name of the airspace)
3	x_name	char		name of x axe (in general "Time")
4	y_name	char		name of y axe (in general "Number of flights")

Chart Header line 2 (TAB SEPARATED)				
#	Field	Type	Size	Comment
1		char	o	empty field
2		char	o,1	one space or empty field (ignored)
3	curve name 1	char		name of first curve (for example: "SHER")

4	curve name 2	char		name of first curve (for example: "Inst. Count")
...	...	char		...
n	curve name n	char		name of curve n

Chart Body (TAB SEPARATED) is made in general of x lines to cover 24hours by step of (24*60)/x minutes				
#	Field	Type	Size	Comment
1	chart name	char		name of the chart
2	time	int	4	expressed HHMM padded with 0
3	value curve 1	int, float		value for curve 1 (can have decimals)
4	value curve 2	int, float		value for curve 2 (can have decimals)
...	...	int, float		...
n	value curve n	int, float		value for curve n (can have decimals)

Chart Body End				
#	Field	Type	Size	Comment
1	empty	char	o	empty line (mandatory)

Example (2 charts with time step of 6 hours)

```

2
→ENGMTMA→Time→Nbr of flights
→→SHER→INST
ENGMTMA→0000→0→0
ENGMTMA→0600→0→0
ENGMTMA→1200→1→1
ENGMTMA→1800→1→1

→SLEMMCCO→Time→Nbr of flights
→→SHER→INST
SLEMMCCO→0000→1→1
SLEMMCCO→0600→1→1
SLEMMCCO→1200→1→1
SLEMMCCO→1800→1→1

```

9.7.29 gsl - Sector

NAME:	Sector		
extension	gsl		

origin:	Gasel, SAAM			
separator:	(semicolon)			
sort	no sort			
comment	Describe how elementary sectors are built from airblocks			
:				
FIRST LINE				
#	Field	Type	Size	Comment
1	#	char	1	
2	file type	text		= SECTOR
3	version	num		= 2
4	NM AIRAC cycle	num		
5	AIRAC start date	date		format is yyyyMMdd
6	AIRAC end date	date		format is yyyyMMdd
7	number of record	num		
8	datasource	text		
HEADER				
#	Field	Type	Size	Comment
1	S	char	1	S=Sector
2	Sector ID	text		
3	Sector name	text		
4	Nb of airblock	num		= number of BODY lines
5	Airspace category	char	1	
6	Sector type	text		FIR=Flight Information Region ERSA=Elementary Restricted Airspace ES=Elementary Sector ERAS=Elementary Regulated Airspace UNK= Unknown type
BODY				
#	Field	Type	Size	Comment
1	A	char	1	A=Airblock
2	Airblock name	text		
3	Operation	char	1	Always "+"
4	Lower FL	num		
5	Upper FL	num		

Example:

```
#;SECTOR;2;383;20140206;20140305;4364;EAR_P
S;BGGLFIR;SONDRESTROM FIR;1;_;FIR
A;002BG;+;0;999
S;BIFAROER;FAEROER CTA;1;_;FIR
A;011BI;+;0;200
S;BIRD;_;9;_;FIR
A;001BI;+;0;999
```

```

A;002BI;+;0;999
A;003BI;+;0;999
A;004BI;+;0;999
A;011BI;+;200;999
A;100BI;+;0;999
A;104BI;+;0;999
A;107BI;+;0;999
A;108BI;+;0;999
S;BIRDES;REYKJAVIK EAST;4;_;ES
A;003BI;+;85;999
A;100BI;+;55;999
A;105BI;+;195;999
A;110BI;+;55;999

```

9.7.30 interieur

Interieur.txt				
extensio n:	interieur.txt			
origin:	icr.exe			
separato r:	blank			
sort:	by window (field 3), then by seg sequence (field 11)			
commen t:	it is the result of intersection between 2D windows (.are file) and segments (name1 name2 lat1 lon1 lat2 lon2) Describes segment lying inside the windows			
1	segment name	char		Built with point name separated with "_"
2	area name	char		is field 15 of ".are" file. "ZZ" means no window
3	area index	num		index starts at 1
4	point name seg origin	char		
5	latitude seg origin	float		in minute decimal
6	longitude seg origin	float		in minute decimal
7	point name seg end	char		
8	latitude seg end	float		in minute decimal
9	longitude seg end	float		in minute decimal
10	segment distance	float		in NM decimal, is the total length of the segment from origin to end
11	seg sequence	num		"-1" means entire segment in the window, else segment sequence (start at 1)
12	intersec begin lat	float		where seg enters window, set to origin if seg origin is in window

13	intersec begin lon	float		where seg enters window, set to origin if seg origin is in window
14	distance begin	float		distance from origin to intersection begin, 0 if seg origin is in window
15	intersec end lat	float		where seg quits window, set to end if seg end is in window
16	intersec end lon	float		where seg quits window, set to end if seg end is in window
17	distance end	float		distance from origin to intersection end, total length if seg end is in window

Example:

```

BARIT_KOMAR ZLDZOCD1 1 BARIT 2595.756383 1071.140600 KOMAR
2697.775117 1009.029317 111.336757 2 2681.765083 1019.039467
93.823319 2697.775117 1009.029317 111.336757
KOMAR_BARIT ZLDZOCD1 1 KOMAR 2697.775117 1009.029317 BARIT
2595.756383 1071.140600 111.336757 2 2697.775117 1009.029317 0.000000
2681.765083 1019.039467 17.513438
GILUK_BOSNA ZLDZOCD1 1 GILUK 2632.000000 1088.000000 BOSNA
2662.385817 1065.994550 34.246874 2 2662.246567 1066.096383 34.089746
2662.385817 1065.994550 34.246874
KEB_BOSNA ZLDZOCD1 1 KEB 2636.000000 1107.000000 BOSNA 2662.385817
1065.994550 39.519114 2 2662.305567 1066.120783 39.398358 2662.385817
1065.994550 39.519114
BOSNA_KOMAR ZLDZOCD1 1 BOSNA 2662.385817 1065.994550 KOMAR
2697.775117 1009.029317 53.792643 -1 2662.385817 1065.994550 0
2697.775117 1009.029317 53.792643

```

9.7.31 jse - Jet Stream Effect file

Jet Stream Effect	
extension	jse
origin	produced by SAAM jetStreamAnalyser, used by jetstream (normally launch immediately after profile processing)
separator	blank
sort	no sort
comment	each line contains a city pair and the associated time shift information for the arrival

#	Field	Type	Size	Comment
1	origin	char		ICAO code
2	destination	char		ICAO code
3	time shift at the arrival	int		unit is second

Example:

```
KJFK EGLL 4120
KJFK LEMD 120
EDDF KJFK 0
```

9.7.32 lay - Layer file

Layer				
#	Field	Type	Size	Comment
1	Version	num		indicates formatting version of this file, currently = 1
2	Project ID	num		unique identifier of the project, used as a key to other SAAM files, could be derived from current system date and time
3	Scenario name	char		name of a scenario. A slash "/" introduce hierarchy between projects
4	Effective date Beg.	char	10	starting date expressed dd/mm/yyyy from which item can be operated, special case 01/01/1900 means effective date is not known and/or then item can always be operated.
5	Effective date End	char	10	ending date expressed dd/mm/yyyy from which item can be operated, special case 31/12/9999 means effective date is not known and/or item can always be operated.
6	Type	char		could be used to indicates the type of the project "NETWORK", "AIRSPACE", "MILITARY", "CONTINGENCY", "RAD"... could be empty.
7	Author	char		name of the responsible of the project
8	Description	char		brief description of the project. Could be empty.
9	Project date	char	10	date when the project started to be studied (input) in SAAM. Format: dd/mm/yyyy

Example (do not correspond to any official project):

```
1;220120091649;FAB/FABEC/German Swiss
Interface;01/04/2010;31/12/9999;BOTH;Robert;inversion of routes and
new sector shape;22/01/2009
```

9.7.33 lop - List of Options file

List of Options	
extensio n:	lop
origin:	SAAM
separato r:	blank
sort:	NO, but CDR filter requires LOP to be sorted by original flight ID (field 2) then by flight option number (field 10)
comment :	Describe 2D/3D/4D route option features Input and output for SOP, query, CDR management. Created by Route Option Generator.

#	Field	Type	Size	Comment
1	flight ID	num		* is original flight ID for flight having a ref. routing, OR * >=1000000000 for flight having a new routing option (generally longer than ref. routing)
2	original flight ID	num		original/father flight, as found in expand file
3	validation	num		1 (default) means option is valid (possibly visually checked by expert) else 0
4	choice	num		0 (default) means option was not chosen by Optimiser else 1
5	extension cost	num		route length cost of the option expressed in NM
6	global cost	num		length cost OR total cost in EURO (=length cost * N EURO/NM + CRCO cost)
7	departure	char	4	Airport ICAO code
8	arrival	char	4	Airport ICAO code
9	flow ID	num		ID of flow (set of flights having same airport or zone arrival departure), see equi file
10	flight option number	num		starts at 1 for every flight, then goes to 2 if the flight have a routing option...
11	flow demand	num	4	for MMFA, fixed to 1/14 hour (= 0.0714286)
12	flow selected	num		for MMFA, generally set to 0
13	cost delay	num		for MMFA, generally set to 1800

Example (4 original/father flights, 2 of them have 2 additional route options):

```
615768 615768 1 0 1221.3 1221.3 LDDU EIDW 27061 1 0.0714286 0 1800
644947 644947 1 0 1221.3 1221.3 LDDU EIDW 27061 1 0.0714286 0 1800
34081 34081 1 0 763.191 763.191 LDDU EDDL 34081 1 0.0714286 0 1800
```

```

1000000000 34081 1 0 772.942 772.942 LDDU EDDL 34081 2 0.0714286 0
1800
1000000001 34081 1 0 782.873 782.873 LDDU EDDL 34081 3 0.0714286 0
1800
616502 616502 1 0 763.191 763.191 LDDU EDDL 34081 1 0.0714286 0 1800
1000000002 616502 1 0 772.942 772.942 LDDU EDDL 34081 2 0.0714286 0
1800
1000000003 616502 1 0 782.873 782.873 LDDU EDDL 34081 3 0.0714286 0
1800

```

9.7.34 lox - List of Options Extended file

List of Options eXtended				
#	Field	Type	Size	Comment
1	flight ID	num		* is original flight ID for flight having a ref. routing, OR * >=1000000000 for flight having a new routing option (generally longer than ref. routing)
2	original flight ID	num		original/father flight, as found in expand file
3	validation	num		1 (default) means option is valid (possibly visually checked by expert) else 0
4	choice	num		0 (default) means option was not chosen by Optimiser else 1
5	status	num	4	status of the option (2D/3D/4D) : x1x2x3x4 with xi=0/1, x1=2D option, x2=RFL option, x3=FL constraint option, x4=4D option
6	global cost	num		length cost OR time cost or total cost in EURO
7	departure	char	4	Airport ICAO code
8	arrival	char	4	Airport ICAO code
9	flow ID	num		ID of flow (set of flights having same airport or zone arrival departure), see equi file
10	flight option number	num		starts at 1 for every flight, then goes to 2 if the flight have a routing option...
11	original departure time	num	4	HHMM
12	departure delay	num		in minutes, can be negative if departure is before original departure time

13	original RFL	num	3	
14	delta RFL	num		
15	original flight duration	num		in seconds
16	delta flight duration	num		in seconds
17	route length	num		distance of the route flown between origin & destination expressed in NM

Example (2 original/father flights, 1 of them have 16 options):

```

108471347 108471347 1 0 0000 7873 LEIB EGCC 108459205 1 0950 0 400 0
7873 0 881.702
108471420 108471420 1 0 0000 7173 EIDW EPSC 108471420 1 1005 0 350 0
7173 0 752.778
1000015654 108471420 1 0 0010 7157 EIDW EPSC 108471420 2 1005 0 350 0
7173 -16 752.778
1000015655 108471420 1 0 0100 7197 EIDW EPSC 108471420 3 1005 0 350
20 7173 24 752.778
1000015656 108471420 1 0 0110 7169 EIDW EPSC 108471420 4 1005 0 350
20 7173 -4 752.778
1000015657 108471420 1 0 0001 7173 EIDW EPSC 108471420 5 1005 -20 350
0 7173 0 752.778
1000015658 108471420 1 0 0001 7173 EIDW EPSC 108471420 6 1005 20 350
0 7173 0 752.778
1000015659 108471420 1 0 0001 7173 EIDW EPSC 108471420 7 1005 40 350
0 7173 0 752.778
1000015660 108471420 1 0 0011 7157 EIDW EPSC 108471420 8 1005 -20 350
0 7173 -16 752.778
1000015661 108471420 1 0 0011 7157 EIDW EPSC 108471420 9 1005 20 350
0 7173 -16 752.778
1000015662 108471420 1 0 0011 7157 EIDW EPSC 108471420 10 1005 40 350
0 7173 -16 752.778
1000015663 108471420 1 0 0101 7197 EIDW EPSC 108471420 11 1005 -20
350 20 7173 24 752.778
1000015664 108471420 1 0 0101 7197 EIDW EPSC 108471420 12 1005 20 350
20 7173 24 752.778
1000015665 108471420 1 0 0101 7197 EIDW EPSC 108471420 13 1005 40 350
20 7173 24 752.778
1000015666 108471420 1 0 0111 7169 EIDW EPSC 108471420 14 1005 -20
350 20 7173 -4 752.778
1000015667 108471420 1 0 0111 7169 EIDW EPSC 108471420 15 1005 20 350
20 7173 -4 752.778
1000015668 108471420 1 0 0111 7169 EIDW EPSC 108471420 16 1005 40 350
20 7173 -4 752.778
1000015669 108471420 1 0 1000 7260 EIDW EPSC 108471420 17 1005 0 350
0 7173 87 752.778

```

9.7.35 lsr - List of SOP Runs file

List of SOP Runs				
#	Field	Type	Size	Comment
1	run ID	num		number identifying the run, used in field #4 of flop file
2	mono/multi	num		mono-assignment or multi-assignment
3	type	num		MMFA or SOP or ...
4	percentage optim.	num		percentage of optimisation
5	time	num		time of run
6	date	num		date of run
7	author	char		name of person who made the run
9	study	char		name of study
	fichier option			
	fichier t5			
	fichier capa			
	default capa			
	coef			
8	comment	num		free comment

9.7.36 members - member states

Member states				
#	Field	Type	Size	Comment
	extensi on	txt		
	origin	Used in the route charge module.		
	separat or	blank		
	sort			
	comme nt	The file is called "CRCO_Members.txt"		
#	Field	Type	Size	Comment
1	sector name/ country			
2	year			Format: yyyy. Year from which the country is a member.

Example:

```
AZ 2006
EB 1998
ED 1998
EE 2007
EF 2002
EG 1998
EH 1998
EI 1998
EK 1998
EN 1998
EP 2007
ES 1998
EV 2007
```

9.7.37 mot - Military Opening Time file

Airspace Activation Time	
extensio n:	mot
origin:	manual input. Editable with the Airspace Activation Editor as well.
separato r:	blank
sort:	no
commen t:	input for SAAM processing: Route choice Based on Military Opening Tim. Version 3 is used for Assignment with Free Route Airspace profile restrictions.

#	Field	Type	Size	Comment
1	version number	num		1 (only time) or 2 (time and level) or 3 (time and type)
2	military zone name	char		must be the name appearing in first column of SLS file (normal name)
3	begin time	num	4	HHMM padded with zeros
4	end time	num	4	HHMM padded with zeros
5	lower FL	num	3	Flight Level with 0 (for instance 055) this field exists if version is 2
6	upper FL	num	3	Flight Level with 0 (for instance 055) this field exists if version is 2
5	type of opening	char	2	EX (=opening time from Entry to Exit) or E (=opening time at Entry, no restriction at Exit) this field exists if version is 3

Example (4 military zones) with the first 2 format versions:

```
1 MIL1 1000 1200
1 MIL1 1400 1700
```

```

1 MIL2 0900 1300
1 MIL3 1100 1300
1 MIL4 0900 1800

2 MIL1 1000 1200 055 245
2 MIL1 1400 1700 085 285
2 MIL2 0900 1300 000 600
2 MIL3 1100 1300 125 245
2 MIL4 0900 1800 000 285

```

Example (3 Free route areas) with the format version 3:

```

3 ED 2200 0500 EX
3 LF 2300 0600 EX
3 LI 2200 0500 E

```

Notes:

A time period (between begin time and end time) indicates Airspace activity (closed for Civil traffic in case of military airspace). For assignment, it indicates airspace opening for GAT.

A zone can have several different time period (see example with MIL1 zone).

If a zone does not appear in MOT file it is considered open for civil traffic in [Route Choice Based On Military](#) context.

In the last given examples, the zone is activated during night and the activity time window extends around midnight. To specify an activity time ending at midnight, you shall write 2400.

9.7.38 mwc - Aircraft weights

Aircraft weights	
extensi on	mwc
origin	Used in the route charge module.
separat or	blank
sort	
comme	

Aircraft weights				
nt				
#	Field	Type	Size	Comment
1	Aircraft Type	char		
2	Weight	num		Maximum Take-Off Weight (MTOW) of the aircraft (in metric tonnes)

Example:

A10	20,70
A109	2,95
A119	2,78
A124	392,00
A129	4,10
A139	6,63
A140	21,10
A148	42,78
A149	8,10
A210	0,80

9.7.39 nspsl - NEST File Builder slot list** NEST File Builder slot list file**

Filename: userSlotList.nspsl

Function: Describes a list of slot date used for NEST File Builder. Allow users to call back a list of slot used on another session and to launch the creation of NEST file with it.

Each line with 10 fields defines a part of a slot. If several lines have the same label and same date, they define the same slot. Capacity, OS and regulations data are repeated as long as data are in the same slot.

Field	Value
1	Date of the day in YYYYMMdd
2	Slot Label
3	Absolute file path of traffic file (ALL_FT+, So6, T5, Exp2)
4	Trajectory type ("None" for ALL_FT+; "Initial", "Actual", ... for so6)

5	Absolute file path of date-dependent capacity file
6	Date of the capacity day selected YYYYMMdd
7	Absolute file path of date-dependent Opening Scheme file
8	Date of the Opening Scheme day selected YYYYMMdd
9	Absolute file path of date-dependent regulation file
10	Date of the Regulation Scheme day selected YYYYMMdd

Example:

20131220;Label1;L:/TF2016/Traffic/1313_20131220_m1.so6;Initial;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.ncap;20150703;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.cos;20150703;L:/TF2016/DDR2_Data/VST_1507/1507.nreg;20150703

20131220;Label1;L:/TF2016/Traffic/1313_20131220_m3.so6;Actual;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.ncap;20150703;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.cos;20150703;L:/TF2016/DDR2_Data/VST_1507/1507.nreg;20150703

20131220;Label1;L:/TF2016/Traffic/1313_20131220_trajHyp1_avoid_LFRRM.so6;Dataset1;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.ncap;20150703;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.cos;20150703;L:/TF2016/DDR2_Data/VST_1507/1507.nreg;20150703

20131220;Label1;L:/TF2016/Traffic/1313_20131220_trajHyp2_avoid_LFFFUJ.so6;Dataset2;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.ncap;20150703;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.cos;20150703;L:/TF2016/DDR2_Data/VST_1507/1507.nreg;20150703

20131220;Label1;L:/TF2016/Traffic/1313_20131220_trajHyp3_avoid_LFFFUZ.so6;Dataset3;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.ncap;20150703;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.cos;20150703;L:/TF2016/DDR2_Data/VST_1507/1507.nreg;20150703

20150429;NewLabel;L:/TF2016/Traffic/20150429_NEST.ALL_FT+;None;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.ncap;20150722;L:/TF2016/DDR2_Data/VST_1507/AIRAC_401E.cos;20150722;L:/TF2016/DDR2_Data/VST_1507/1507.nreg;20150722

9.7.40 nact - Activation

Activation File

Filename: activation.nact

Function: Defines if and when capacities are active

Field	Value
1	Sector /TV name
2	Airspace / TV - either "AS" (airspace) or "TV" (traffic volume)
3	Activation – either "N" (non activated) or "A" (activated)

Example:

EBBUALL;AS;N
EBBUALL;TV;N
EBBUEEC;TV;N
EBBUELS;AS;N
EBBUESC;AS;N
EBBUHLC;AS;N
EBBUHUS;AS;N
EBBULUS;AS;N
EBBUNLS;AS;N
EBBUNWC;AS;N
EBBUWHS;AS;N
EBBUWLS;AS;N
EBBUWSC;AS;N
EBBUWWC;AS;N

9.7.41 narp - Airport

NAME:	Airport			
extensio n:	narp			
separato r:	;	(semicolon)		
sort:				
commen t:				
FIRST LINE				
#	Field	Type	Size	Comment
1#	char		1	
2file type	text			= AIRPORT
3version	num			= 1
4NM AIRAC cycle	num			
5AIRAC start date	date			format is yyyyMMdd
6AIRAC end date	date			format is yyyyMMdd
7number of record	num			
8datasource	text			
#	Field	Type	Size	Comment
1Airport ID	char		4	ICAO code
2Airport name	text			
3latitude	num			degree decimals
4longitude	num			degree decimals
5TIS	num			
6TRS	num			
7Taxitime	num			
8Altitude	num			Flight levels

Example:

```
#;AIRPORT;1;383;20140206;20140305;7403;EAR_P
AFIL;AFIL;40.640277777778;-73.778333333333;_:_:_;0
AGGA;AUKI/GWAUNARU'U;-8.698333333333;160.6783333333;0;0;0;0
AGGE;BALALAE;-7.083333333333;155.75;0;0;0;0
AGGH;HONIARA/HENDERSON;-9.4316666666667;160.0533333333;0;0;0;0
AGGJ;AVU AVU;-9.533333333333;160.0833333333;0;0;0;0
AGGK;KIRA KIRA;-10.45;161.89666666667;0;0;0;0
AGGL;GRACIOSA BAY;-10.723333333333;165.78;0;0;0;0
AGGM;MUNDA;-8.325;157.26333333333;0;0;0;0
AGGN;GIZO/NUSATAPE;-8.1016666666667;156.835;0;0;0;0
AGGT;SANTA ANA;-10.883333333333;162.46666666667;0;0;0;0
AGGU;MARAU SOUND;-8.083333333333;160;0;0;0;0
AGGV;SUAVANAO;-7.6;158.7;0;0;0;0
ANAU;NAURU ISLAND;-0.548333333333;166.91666666667;0;0;0;0
AYFI;FINSCHHAFEN;-6.6216666666667;147.85416666667;0;0;0;0
```

```
AYGA;GOROKA;-6.0833333333333;145.39166666667;0;0;0;0
```

9.7.42 nbak - ControllerAvailabilities

Controller Availability File

Filename: controllerAvailability.nbak

Function: Describes a list of airport sets

Function: Defines the number of controllers available to man active control positions for every 10-minute interval of the day

Field	Value
1	Date
2	ACC name
3	Start time
4	End time
5	Max number of control positions

Example:

```
10/07/2003;EDFFACC;04:00;04:29;2
10/07/2003;EDFFACC;04:30;04:59;2
10/07/2003;EDFFACC;05:00;18:59;4
10/07/2003;EDFFACC;19:00;19:59;4
10/07/2003;EDFFACC;20:00;20:29;8
10/07/2003;EDFFACC;20:30;20:59;8
10/07/2003;EDFFACC;21:00;23:59;2
10/07/2003;EDLLACC;00:00;04:59;2
```

9.7.43 ncap - Capacity

NAME	Capacity			
:				
extensi	ncap			

on:				
separat	;	(semicolon)		
or:				
sort:				
comme				
nt:				
FIRST LINE				
#	Field	Type	Size	Comment
1#	char		1	
2file type	text			= CAPACITY
3version	num			= 2
4NM AIRAC cycle	num			
5AIRAC start date	date			format is yyyyMMdd
6AIRAC end date	date			format is yyyyMMdd
7number of record	num			
8datasource	text			
#	Field	Type	Size	Comment
1Date	date			format is dd/MM/yyyy
2Element ID	text			Element on which the capacity is applied
3Start time	time			format is hh:mm
4End time	time			format is hh:mm
5Capacity	num			
6Hourly 3 capacity	num			Not used (always " ")
7Element type	text			
8Element category	char		1	Always "G"
9Env table source	char			B=body/default P=permanent T=temporary N=tact

Example:

```
#;CAPACITY;2;383;20140206;20140305;377233;EAR_P
06/02/2014;*ANT;00:00;23:59;999;_;PT;G;B
06/02/2014;*FLO1;00:00;23:59;999;_;PT;G;B
06/02/2014;*INTO;00:00;23:59;999;_;PT;G;B
06/02/2014;*KOR;00:00;23:59;999;_;PT;G;B
06/02/2014;*MUS1;00:00;23:59;999;_;PT;G;B
06/02/2014;*NSA;00:00;23:59;12;_;PT;G;B
06/02/2014;*PG3;00:00;23:59;12;_;PT;G;B
06/02/2014;ABESI;00:00;23:59;999;_;PT;G;B
06/02/2014;ABLAT;00:00;23:59;999;_;PT;G;B
06/02/2014;ABLOM;00:00;23:59;999;_;PT;G;B
06/02/2014;ABRIX;00:00;23:59;999;_;PT;G;B
06/02/2014;ABUPI;00:00;23:59;12;_;PT;G;B
06/02/2014;ABUSU;00:00;23:59;12;_;PT;G;B
06/02/2014;ADARA;00:00;23:59;999;_;PT;G;B
```

```
06/02/2014;ADILI;00:00;23:59;999;_;PT;G;B
06/02/2014;ADORO;00:00;23:59;12;_;PT;G;B
06/02/2014;AGADO;00:00;23:59;12;_;PT;G;B
06/02/2014;AGN;00:00;23:59;999;_;PT;G;B
06/02/2014;AGORI;00:00;23:59;999;_;PT;G;B
06/02/2014;AIREZ;00:00;23:59;12;_;PT;G;B
```

9.7.44 ncnc - ConfNbControllers

Number of Controllers per Configuration File

Filename: confNbControllers.ncnc

Function: Defines the number of controllers required to man each configuration

Field	Value
1	ACC name
2	Configuration name
3	Number of controllers

Example:

```
BIRDACC;CONF1;2
BIRDACC;CONF2;4
BIRDACC;CONF3A;6
BIRDACC;CONF3B;6
BIRDACC;FEED4;7
BIRDACC;CONF4;8
```

9.7.45 ndpr - Date Pair

Date Pair File

Filename: "MyDatePairs.ndpr"

Function: Defines a list of file/scenario date matches for linking date-related file data to scenario date slots during the file import process.

The following is a description of the format.

Field Separator: ; (semicolon)

Format of lines describing file/scenario date matches

Field	Value
1	File Date
2	File Date Label (Excel Import) / File Name (Text Import)
3	Scenario Date Slot ID (ID format: yyyyymmdd@label)

Example1 (Text Import):

```

01/01/2015;20150101__Capacity_Custom.ncap;20150101@
01/01/2015;20150101__ControllerAvailability_Custom.nbak;20150101@
01/01/2016;20160101_HIGH_Capacity_Custom.ncap;20160101@HIGH
01/01/2016;20160101_HIGH_ControllerAvailability_Custom.nbak;20160101
@HIGH
01/01/2016;20160101_HIGH_OpeningScheme_Custom.cos;20160101@HIGH
01/01/2016;20160101_LOW_Capacity_Custom.ncap;20160101@LOW
01/01/2016;20160101_LOW_ControllerAvailability_Custom.nbak;20160101
@LOW
01/01/2016;20160101_LOW_OpeningScheme_Custom.cos;20160101@LOW
01/01/2016;20160101_ORIG_Capacity_Custom.ncap;20160101@ORIG
01/01/2016;20160101_ORIG_ControllerAvailability_Custom.nbak;20160101
@ORIG
01/01/2016;20160101_ORIG_OpeningScheme_Custom.cos;20160101@ORIG
13/01/2016;20160113__Capacity_Custom.ncap;20160113@
13/01/2016;20160113__ControllerAvailability_Custom.nbak;20160113@
13/01/2016;20160113__OpeningScheme_Custom.cos;20160113@
18/01/2016;20160118__Capacity_Custom.ncap;20160118@
18/01/2016;20160118__ControllerAvailability_Custom.nbak;20160118@
18/01/2016;20160118__OpeningScheme_Custom.cos;20160118@
22/03/2016;20160322__Capacity_Custom.ncap;20160322@
22/03/2016;20160322__ControllerAvailability_Custom.nbak;20160322@
22/03/2016;20160322__OpeningScheme_Custom.cos;20160322@

```

Example2 (Excel Import) :

```
01/01/2015;;20150101@  
01/01/2016;ORIG;20160101@ORIG  
01/01/2016;HIGH;20160101@HIGH  
01/01/2016;LOW;20160101@LOW  
13/01/2016;;20160113@  
18/01/2016;;20160118@  
22/03/2016;;20160322@
```

Example3 (Text or Excel Import) :

```
01/01/2015;;20150101@  
01/01/2016;;20160101@ORIG  
13/01/2016;;20160113@  
18/01/2016;;20160118@  
22/03/2016;;20160322@
```

9.7.46 nflw - Flow

Traffic Flow File

Filename: flow.nflw

Function: Defines a list of traffic flows

The first line of this file defines the number of flows defined.

Each line beginning with an “F” defines a new flow with the following structure:

Field	Value
1	F
2	ID of the flow
3	Name of the flow
4	Nb of flow elements

This line is followed by several lines beginning with an "E" following this structure:
 (the Nb value described in the previous table)

Field	Value
1	E
2	ID of the element
3	Type of element
4	Role of element (either D downstream – meaning downstream of the reference location i.e. an arrival area, or U upstream – meaning upstream of the reference location i.e. a departure area)

Example:

2697

F;1+2>345;;2
 E;EH1+2;AS;U
 E;EH3+4+5;AS;D
 F;1+2>EHLE;;2
 E;EH1+2;AS;U
 E;EHLE;AD;D
 F;12345>KD;;2
 E;EH12345;AS;U
 E;EHKD;AD;D
 F;13>LAMBB;;2
 E;LAM;SP;D
 E;EGB;AZ;D

9.7.47 ngrp - Entity Group File

Entity Group File

Filename: "myEntityGroup.ngrp"

Function: Defines a list of airspace entities

The following is a description of the format used to store entity lists.

Format the first line describing the entity file version

Field	Value
1	Version Key : "version"
2	Version Value : "1.0"

Format of lines describing entities

Field	Value
1	Entity ID : identifier of entity e.g. EBBUCTA
2	Entity Type : "ACC", "Sector", "TV", "Nav Point", "Traffic Flow"

version;1.0

DAAACTA;ACC

DTTCCTA;ACC

EBBRTMA;ACC

EBBUCTA;ACC

EBBUICTA;ACC

EBCFMCTA;ACC

BIRDVGTLIA;Sector

BIRDWE;Sector

DAAASC;Sector

DAAASE;Sector

9.7.48 nmoa - MonitoringArea

Monitoring Area File

Filename: monitoringArea.nmoa

Function: Defines a list of monitoring areas

The first line of the file defines the number of Monitoring Areas defined and other lines follow the following structure:

Field	Value
1	MA (Monitoring Area)
2	ID
3	Name
4	Nb Traffic Volumes

This line is followed by several lines (the Nb value described in the previous table) beginning with a “T” following this structure:

Field	Value
1	T (Traffic Volume)
2	ID

Example:

1

MA;ED;;16

T;EDMALPT

T;EDMALPU

T;EDMALPUT
 T;EDMCHIH
 T;EDMCHIL
 T;EDMCHILH
 T;EDMINKP
 T;EDMINKPX
 T;EDMINN
 T;EDMINNX
 T;EDMKHAPU
 T;EDMKPT
 T;EDMKPTH
 T;EDMKPTN
 T;EDMKPTNX
 T;EDMKPTS

9.7.49 nnpt - NavPoint

Navigation Point File

Filename: navPoint.nnpt

Function: Defines a list of navigation points

The first line of the file defines the number of navigation points defined and other lines follow the following structure:

Field	Value
1	ID
2	type
3	latitude in degrees
4	longitude in degrees
5	Name (underscore means no data)

Example:

17189

```
*01BP;DB;47.651388888889;19.194444444444;_
*01DC;DB;47.57;21.719166666667;_
*01RD;DB;52.155555555556;4.78694444444444;_
*02BP;DB;47.392777777778;19.524166666667;_
*02DC;DB;47.332777777778;21.502222222222;_
*03BP;DB;47.1725;19.663611111111;_
*03DC;DB;47.319166666667;21.658055555556;_
*03RD;DB;52.002222222222;4.5741666666667;_
*04BP;DB;47.244166666667;19.761388888889;_
*04DC;DB;47.340277777778;21.729444444444;_
*04RD;DB;51.871111111111;4.55694444444444;_
*05BP;DB;47.732777777778;18.788611111111;_
```

9.7.50 nreg - RegPlan

Regulation Plan File

Filename: regPlan.nreg

Function: Describes a list of regulations for all the dates

Each line beginning with 9 fields defines a new regulation with the following structure:

Field	Value
1	Date on which the regulation is linked YYYYMMDD
2	ID of the regulation
3	ID of the regulated entity
4	Type of the regulated entity (AS or TV)
5	Flag for starts the day before (Y or N)
6	Slot window width in minutes

7	Slot slice width in minutes
8	<p>Reason for the regulation using the following single letter codes:</p> <p>C ATC CAPACITY G AERODROME CAPACITY W WEATHER T EQUIPMENT (ATC) E EQUIPMENT (NOT ATC) I INDUSTRIAL ACTION M MILITARY ACTIVITY R ATC ROUTING S ATC STAFFING V ENVIRONMENTAL ISSUES D DE-ICING P SPECIAL EVENT O OTHER REASON U UNKNOWN REASON</p>
9	Nb of regulation periods

This line is followed by several lines (the Nb value described in the previous table) with 3 fields following this structure:

Field	Value
1	Time of the start of the regulation period hh:mm
2	Time of the end of the regulation period hh:mm
3	Capacity of the regulation period in flights per hour

Notes:

- If a regulation is across two dates present in the dates of the NEST file, this regulation is duplicated, once with the first date (field 1) and the flag starts the day before equal to false (field 5), once with the second date (fields 1) and the flag starts the day before equal to true (field 5). Thanks to that, the two regulations can be edited independently.

Example:

```

20170228 LGAVD28M LGAVDEP TV N 10 20 W 3
06:15 07:12 15
07:12 07:32 22
07:32 07:33 28
20170228 LGAVG28M LGAV TV N 10 20 W 1
04:40 06:11 20
20170228 LIRFA28 LIRFARR TV N 10 20 G 1
18:40 20:00 36
20170301 LLBGA01 LLBGARR TV N 10 20 G 2
23:00 00:05 0
00:05 00:45 18
20170302 LLBGA01 LLBGARR TV Y 10 20 G 2
23:00 00:05 0
00:05 00:45 18
20170228 LPWEST28 LPWEST TV N 10 20 C 1
12:00 16:40 35
20170301 LQTZG01 LQTZ TV N 10 20 S 1
22:30 04:30 0
20170302 LQTZG01 LQTZ TV Y 10 20 S 1
22:30 04:30 0
20170301 LSZHA01M LSZHARR1 TV N 10 20 G 2
09:20 10:00 34
10:00 11:00 36

```

9.7.51 nsoa - Set Of Airports

NAME	SetOfAirports			
:				
extensi on:	nsoa			
separat or:	;(semicolon)			
sort:				
comme nt:				
FIRST LINE				
#	Field	Type	Size	Comment
1#	char	1		
2file type	text			= SET_OF_AIRPORTS
3version	num			= 1
4NM AIRAC cycle	num			
5AIRAC start date	date			format is yyyyMMdd
6AIRAC end date	date			format is yyyyMMdd
7number of record	num			
8datasource	text			
HEADE				

R				
#	Field	Type	Size	Comment
1	S	char		1S=Set of Airports
2	Set of airport ID	text		
3	Set of airport name/description	text		"_ " means no data
4	Set of airport type	text		
5	Nb of airports	num		= number of BODY lines
BODY				
#	Field	Type	Size	Comment
1	A	char		1A=Airport
2	Airport name	text		

Example:

1236

S;..PR;LYPR/BKPR;TF;2

A;BKPR

A;LYPR

S;10MILDES;_;TF;904

A;EBAM

A;EBAW

A;EBBC

A;EBBE

A;EBBH

A;EBBL

A;EBBR

A;EBBT

A;EBBX

9.7.52 nsio - Airspace Skip Values

NAME	SkipInOut			
:				
extensi on:	nsio			
separat or:	;	(semicolon)		
sort:				
comme nt:				

FIRST LINE				
#	Field	Type	Size	Comment
1	Nb of airspaces	num		1= number of BODY lines
BODY				
#	Field	Type	Size	Comment
1	Airspace name	text		
2	Airspace type	text		Allowed values: ES, CS, FIR, ERSA, CRSA, AOI, AOP, AUA, CLUS, NAS, AREA, AUAG, ERAS, CRAS, UNK
3	Skip in	num		duration in minutes (999 for undefined)
4	Skip out	num		duration in minutes (999 for undefined)

Example:

8553

BG;NAS;1;5
 BGGLFIR;FIR;1;5
 BI;NAS;1;5
 BIEGOCA;ES;0;999
 BIFAROER;FIR;1;1
 BIRD;FIR;0;999
 BIRDBARDAR;ES;0;1
 BIRDCTA;AUA;1;20
 BIRDES;ES;1;20
 BIRDEST;ES;1;20
 BIRDFAXI;ES;1;1
 BIRDFIS;ES;1;1
 BIRDICTA;AUA;1;1
 BIRDNO;ES;1;1
 BIRDSS;ES;1;1
 BIRDTMA;AUA;1;999
 BIRDTOCA;AUA;0;0
 BIRDVGTIA;ES;1;1
 BIRDVGTIAT;ES;1;1

9.7.53 ntfv - Traffic Volume

NAME:	TrafficVolume			
extension:	ntfv			
separator:	(semicolon)			
sort:				
comment:				
FIRST LINE				
#	Field	Type	Size	Comment
1#	char		1	
2file type	text			= TRAFFIC_VOLUME
3version	num			= 2
4NM AIRAC cycle	num			
5AIRAC start date	date			format is yyyyMMdd
6AIRAC end date	date			format is yyyyMMdd
7number of record	num			
8datasource	text			
HEADER				
#	Field	Type	Size	Comment
1T	char		1	T=Traffic Volume
2Traffic volume ID	text			
3Traffic volume name/description	text			"_ " means no data
4Traffic volume category	num			MO=Monitorable "_"=no data
5Reference location name				
6Reference location type				AD, AS, AZ, SP, DBP
7Reference location role	char		1	D, A, G
8Nb of flow elements	text			= number of BODY lines
BODY				
#	Field	Type	Size	Comment
1F	char		1	F=Flow
2Airblock name	text			
3Upper FL	num			IN=Included (default value) IE=Included and Exempted EX=Excluded EM=Exempted

Example:

```
#;TRAFFIC_VOLUME;2;383;20140206;20140305;7799;EAR_P
T;ABSON;ABSON FLOWS;_;UKLVCTA;AS;G;5
F;BUK>AB;IN
F;BUK>AG;IN
F;DIB>AB;IN
F;INR>AB;IN
F;INR>AG;IN
T;AERODS1;EDNY ARRS VIA EDGGADS;_;EDGGADS;AS;G;1
F;>EDNY;IN
T;AERODS2;EDNY DEPS VIA EDGGADS;_;EDGGADS;AS;G;1
F;EDNY>;IN
;EX
```

9.7.54 Old SAAM are file

Newmaxo ascii region file				
extension	are			
origin:	newmaxo, SAAM, CAPAN			
separator	blank			
sort	Body is sorted by point sequence (for SAAM it should be clockwise)			
comment:	has a header and a body. First the header(one line) followed by the body (n lines).			
Header				
#	Field	Type	Size	Comment
1	nb_point	num	~	contains the number of line of the following body
2	latitude	num	~	in minutes (decimal for SAAM), location of the label
3	longitude	num	~	in minutes (decimal for SAAM), location of the label
4	flights	num	~	number of flights slot of info
5	low_level	num	~	low level of the volume in flight level (FL)
6	high_level	num	~	high level of the volume in flight level (FL)
7	surface	num	~	surface of the volume, not important can be o
8	sector_num	num	~	number of sector in the volume, not important can be o
9	flight time	num	~	flight time, not important can be o
10	traffic density	num	~	traffic density, not important can be o
11	x mileage	num	~	extra mileage, not important can be o
12	rte extens.	num	~	Store TopLev/BotLev/Trans/Light/Top codes (SAAM code)
13	value 1	num	~	Store the color+trans of the Airblock (SAAM code)
14	value 2	num	~	Store the sign of the airblock (CAPAN),

				$o = \text{positif}, 1 = \text{negatif}$
15	name	char	\sim	name or code of the volume (max 12 char for newmaxo)

Body (the polygon must be closed: first point = last point)

#	Field	Type	Size	Comment
1	latitude	num	~	in minutes (in minutes decimal for SAAM)
2	longitude	num	~	in minutes (in minutes decimal for SAAM)

Comment: the SAAM codes are:

=0 means the volume properties is set by the one found in TDV line
 =1 means TopLev or BotLev set by field #5 and #6 or value is ON
 =2 means value is OFF
 !=0 means for color only, the value of the color
 Color has 4 components varying between 0 to 255 = trans*255^3 + red*255^2 +
 green*255 + blue
 The blue component of the color, if setup, is never equals to ZERO.
 field #12 = TopLev*256 + BotLev*64 + trans*16 + Light*4 + top

Example:

9.7.55 odz - StatforFamilies

 STATFOR Forecast Families File

Filename: "FFF_yyyyMM.odz" (FFF: STATFOR "MTF" / "STF" forecast type, yyyy: year using four characters, MM: month using two characters)

Function: Defines the link between an airport and a demand forecast zone

All Lines

Field	Value
1	Forecast zone ID (AKA airport family ID)
2	Forecast zone definition (wildcard expression capturing single airports – no spaces allowed, all wildcards must be 4 letters long, valid special characters: * replaces any letter of the alphabet, - removes a single airport, + adds a single airport)
3	“Des” destination forecast zone header (AKA airport family)
4, 5, 6, ...	Forecast year header (number of fields depends on the number of forecasted years in the file – generally 7 years)

Example:

Albania,LA**

Armenia,UD**

Austria,LO**

Azerbaijan,UB**

Azores,LPFL+LPGR+LPHR+LPLA+LPPI+LPPD+LPAZ+LPSJ+LPCR

Barcelona,LEBL

BarcelonaACC,LEV

Belarus,UM**-UMKK

Belgium,EB**

BerlinFIR:Berlin,EDDB+EDDI+EDDT

BerlinFIR:BerlinNorth,EDAH+EDBH+EDBM+EDCD+EDCP+EDOP+ETNL+ETNU
+ETSH

BerlinFIR:BerlinSouth,EDDC+EDDE+EDDP+EDDN+EDAB+EDAC+EDQC+EDQD

Birmingham,EGBB

Bosnia-Herzegovina,LQ**

BremenFIR:BremenNorth,EDHL+EDXW+ETMN+EDDW+EDHK+ETNJ+EDHF

BremenFIR:BremenSouth,EDDV+ETHB+EDVE+ETNW+ETND+ETHE+EDVK

BremenFIR:Hamburg,EDHI+EDDH

Bulgaria,LB**

CanaryACC,GC**-GCLP

9.7.56 ord - Sector Order file

sector order				
#	Field	Type	Size	Comment
	extensi on	ord		
	origin	User manual input		
	separato r	N/A		
	sort	sort the lines (containing sector names) according to the order you want the sector loads to be displayed		
	commen t	each line contains the name of a sector for which the sector load has to be computed		
1	sector name	char	26	the same one as the one found in field 2 of t5

Example:

```
LFEPPLOW
EDDYLNO
```

9.7.57 pal - Color Palette

Color Palette				
#	Field	Type	Size	Comment
	extensi on	pal		
	origin			
	separat or	space		
	sort	no sort		
	commen t	Describe a list of color ass for a range of values		
1	min value	float		
2	max value	float		
3	r	num		between 0 and 255
4	g	num		between 0 and 255
5	b	num		between 0 and 255

Example:

```
-130.152 130.152 231 229 255
-81.1616 81.1616 255 220 221
81.1616 243.485 255 218 219
243.485 405.808 255 216 217
405.808 568.131 255 214 215
```

568.131	730.455	255	211	213
730.455	892.778	255	209	211

9.7.58 pin zin - Profile Input file

profin	
extensio n:	pin
origin:	SAAM kernel
separato r:	blank
sort:	no
commen t:	input of SAAM profile processing. Each flight has a header (one line) and a body (n lines).
	Note that the distances need not be linear distances and should therefore not be recalculated from coordinates.

HEADER				
#	Field	Type	Size	Comment
1	flight identifier	num		
2	origin of flight	char	4	ICAO code
3	destination of flight	char	4	ICAO code
4	aircraft type	char	4	
5	route length	float		total length of the route
6	departure time	num		expressed in minute from midnight
7	RFL	num		Optional: if RFL=0 or RFL outbound then Optimum Cruise FL is taken
8	body lines number	num		number of following body lines
9	callsign	char	<=7	
10	departure date	num	6	YYMMDD padded with 0's

BODY				
#	Field	Type	Size	Comment
1	flight identifier	num		same as the header one
2	point name	char	3 or 5	
3	length	float		length from origin to this point
4	parity	num	1	0=NO, 1=ODD, 2=EVEN, 3=ODD_LOW, 4=EVEN_LOW, 5=ODD_HIGH,

				6=EVEN_HIGH (from this point to next one)
5	level constraint	num	1 to 4	NEVER starts with oxx 999=no constraint, +nnn=not above FLnnn in climbing phase, -nnn=not above FLnnn in descending phase, if the constraint is set on the first point of the route (departure, with length=0) it means not above FLnnn in cruising phase of the flight.
6	latitude	float		expressed in minute of angle (with decimals)
7	longitude	float		expressed in minute of angle (with decimals)

Example (3 flights):

```

110 DAAG LEBL B727 331 900 0 6 DAH2012 970905
110 DAAG 0 2 999 2202 193
110 ALR 3 1 999 2200 190
110 MHN 200 2 999 2391 253
110 TASOS 298 2 999 2456 158
110 QUV 330 2 999 2477 125
110 LEBL 331 0 999 2477 124
111 DAAG LEPA B727 234 1090 330 5 DAH2000 970905
111 DAAG 0 2 999 2202 193
111 ALR 3 2 999 2200 190
111 IBZ 160 2 999 2334 88
111 MJV 227 2 999 2366 165
111 LEPA 234 0 999 2373 163
239 DTTA LICJ AT42 271 490 0 6 AZA873 970905
239 DTTA 0 1 999 2211 614
239 TUC 0 1 999 2211 614
239 MON 70 1 999 2143 637
239 TRP 229 1 999 2273 750
239 PAL 261 2 999 2281 790
239 LICJ 271 0 999 2290 786

```

9.7.59 pkg - StatsPackage

Statistics Package File

Filename: "MyStatsPackage.pkg"

Function: Defines a list of statistics used to generate actual statistics for a given scenario and date

The following is an example of a stats package containing one statistic; namely ACC hourly demand counts every hour.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Stat>
<Stat version="9">
    <Package>
        <Filename>B:/NEVAC_DATA/Import_Export/dummy.csv</Filename>

```

```

<File_Type>0</File_Type>
<Group_by_Day>0</Group_by_Day>
<Show_Scenario_Info>1</Show_Scenario_Info>
<Multiple_Output_Files>0</Multiple_Output_Files>
<Stat_Def>
    <ID>Stat_Demand_60_60</ID>
    <Type>0</Type>
    <Airspace_Def>
        <Airspace_ID>EBBUCTA</Airspace_ID>
        <Airspace_Type>0</Airspace_Type>
    </Airspace_Def>
    <Integ_Window>60</Integ_Window>
    <Sliding_Step>60</Sliding_Step>
    <Capa_Monitor>4</Capa_Monitor>
    <Peak_Analysis>0</Peak_Analysis>
    <NE_Param>2</NE_Param>
    <Airport_Demand_Param>3</Airport_Demand_Param>
    <FIPS_Param>2</FIPS_Param>
    <Baseline_Param>2</Baseline_Param>
    <RegDisplayOpt_Param>2</RegDisplayOpt_Param>
    <Backbone_Type>3</Backbone_Type>
    <Demand_Type>0</Demand_Type>
    <Count_Type>0</Count_Type>
    <Overload_Calc_Mode>3</Overload_Calc_Mode>
    <Overload_Type>2</Overload_Type>
    <Overload_Opening_Period>2</Overload_Opening_Period>
    <Flow_ID></Flow_ID>
    <Flow_Type>5</Flow_Type>
    <Raw_Indicator_Type>10</Raw_Indicator_Type>
    <Complexity_Type>10</Complexity_Type>
    <ACC_Complexity_Calc_Area>2</ACC_Complexity_Calc_Area>
    <Start_Period_Index>-1</Start_Period_Index>
    <End_Period_Index>-1</End_Period_Index>
    <Min_Alt>-1</Min_Alt>
    <Max_Alt>-1</Max_Alt>
    <Regu_Reason>14</Regu_Reason>
    <Delay_Threshold>8</Delay_Threshold>
    <Delay_Display>2</Delay_Display>
  </Stat_Def>
</Package>
</Stat>

```

9.7.60 pl6 - PointLine6 file

PointLine6	
extensio n:	pl6
origin:	produced by so6ToPl, used by plToSo6. Different from the one provided by Eatchip Database (field #14 is missing)
separato	blank

r:	
sort:	Yes on field #1 (id) then on Field #3 (seq)
comment:	same format as so6 but point by point (files are smaller)

#	Field	Type	Size	Comment
1	ID	num		flight ID
2	break	num		same number= no break, new number=break
3	sequence	num		point sequence, start at 1 for the first point
4	point name	char		
5	level	num		Unit is Flight Level
6	time	num	6	Format is HHMMSS with leading zeros
7	date	num	6	Format is YYMMDD with leading zeros
8	origin	char		airport where the flight is taking off
9	destination	char		airport where the flight is landing
10	aircraft type	char		
11	callsign	char		
12	status	num	1	to next point: 0=climb, 1=descent, 2=cruise
13	parity	num	1	to next point: 0=NO, 1=ODD, 2=EVEN, 3=ODD_LOW, 4=EVEN_LOW, 5=ODD_HIGH, 6=EVEN_HIGH
14	length ?	num	1	ALWAYS ZERO (=0)
15	latitude	float		in minute decimal
16	longitude	float		in minute decimal

Example (1 flight):

```

23 1 1 BIKF 0 072000 010626 BIKF ELLX B757 ICE614 0 1 0 3839 -1356
23 1 2 $aksP 390 074230 010626 BIKF ELLX B757 ICE614 0 1 0 3782.48 -
1066.26
23 1 3 B 390 081604 010626 BIKF ELLX B757 ICE614 2 1 0 3660 -600
23 1 4 *BORM 390 085348 010626 BIKF ELLX B757 ICE614 2 1 0 3469 -193
23 1 5 ADN 390 085938 010626 BIKF ELLX B757 ICE614 2 1 0 3438 -136
23 1 6 SKATE 390 092949 010626 BIKF ELLX B757 ICE614 2 1 0 3300 184
23 1 7 SAMON 390 093810 010626 BIKF ELLX B757 ICE614 2 1 0 3243 227
23 1 8 SPY 390 095123 010626 BIKF ELLX B757 ICE614 2 1 0 3152 291
23 1 9 $aksQ 390 100140 010626 BIKF ELLX B757 ICE614 2 1 0 3078.15
324.359
23 1 10 LNO 244 100802 010626 BIKF ELLX B757 ICE614 1 1 0 3035 343
23 1 11 LARED 208 100950 010626 BIKF ELLX B757 ICE614 1 1 0 3024 349
23 1 12 DIK 68 101539 010626 BIKF ELLX B757 ICE614 1 1 0 2992 368
23 1 13 ELLX 0 101842 010626 BIKF ELLX B757 ICE614 1 1 0 2977 372

```

9.7.61 pnl - Penalisation file

penalisation	
extension:	pnl

origin:	produced manually
separator:	one or more spaces (ASCII character 32)
sort:	no sort
comment:	each line contains the penalisation for a network segment. It is possible to have a comment line starting with "://"

#	Field	Type	Size	Comment
1	Point 1	char	26	the starting point of the segment to penalize
2	Point 2	char	26	the ending point of the segment to penalize
3	penalisation	num		penalisation (in NM)

Example :

LONTA	BEDAK	1000
LONTA	GONAR	1000
MODRA	KOGAT	1000
LINDU	OLOTA	1000
VAL	KOGAT	1000

9.7.62 pre - Presentation file

Presentation file is a text file that contains names and possibly path of several TDV files. Once a PRE file is loaded, the user can navigate from one TDV to the next either manually or automatically.

presentation				
#	Field	Type	Size	Comment
1	name and possibly path to a TDV file	char		

Example :

EB_TMA.tdv
ED_TMA.tdv
LE_TMA.tdv
LF_TMA.tdv
LS_TMA.tdv

9.7.63 qt5 - Quick T5 file

Quick T5				
extension:	QT5			
origin:	produced by SAAM 3D Density			
separator:	blank			
sort:	no sort			
comment:	each line contains best flight trajectories 3D cell penetration			
#	Field	Type	Size	Comment
1	ID	num		flight ID found in field 17 of SO6
2	sector/cell name	char		numeric code representing the name of the cell

Example (flight 1000 is crossing cells 3583, 3676 and 3770)

```

1000 3583
1000 3676
1000 3770

```

9.7.64 rlex - Route Length Extension Exception

Route Length Extension Exception				
extensio n:	rlex			
origin:	produced manually			
separator:	blank (space or tab)			
sort:	no sort			
comment:				
#	Field	Type	Size	Comment
1	ADEP	char	4	Departure airport of the route
2	ADES	char	4	Arrival airport of the route
3	allowed extension	num		Allowed extension for this airport pair (in NM)

Example :

```

DGAA KJFK 500
DGAA LPAZ 180
DNAA LTAI 180
DNCA LPPT 170
DNKN GMMN 450
DNMM KJFK 200
DNMM LPFR 250
DNMM LTBA 190
DNMM TJSJ 1200

```

```

EBAW EBBR 75
EBAW EBZW 100

```

9.7.65 SectorOpening

Sector Opening & Capacity File

Filename: sectorOpening.txt

Function: Defines the list of open sectors and their time-dependent capacities

Field	Value
1	Date
2	ACC ID
3	Configuration ID
4	TV ID
5	Capacity start time
6	Capacity end time
7	Capacity value (flights / hour)

Example:

```

13/07/2009;LIPP;Conf 2A;LIPPLOW;00:00;04:22;44
13/07/2009;LIPP;Conf 2A;LIPPUPP;00:00;04:22;45
13/07/2009;LIPP;Conf 4;LIPPNL;04:23;04:55;43
13/07/2009;LIPP;Conf 4;LIPPSL;04:23;04:55;42
13/07/2009;LIPP;Conf 4;LIPPNU4;04:23;04:55;42
13/07/2009;LIPP;Conf 4;LIPPSU4;04:23;04:55;42
13/07/2009;LIPP;Conf 6;LIPPNL;04:56;05:39;43
13/07/2009;LIPP;Conf 6;LIPPNU ;04:56;05:39;44
13/07/2009;LIPP;Conf 6;LIPPSL;04:56;05:39;42
13/07/2009;LIPP;Conf 6;LIPPSU;04:56;05:39;47
13/07/2009;LIPP;Conf 6;LIPPTN;04:56;05:39;43

```

13/07/2009;LIPP;Conf 6;LIPPTS;04:56;05:39;43
 13/07/2009;LIPP;Conf 7B;LIPPNL;05:40;06:49;43
 13/07/2009;LIPP;Conf 7B;LIPPNT;05:40;06:49;44
 13/07/2009;LIPP;Conf 7B;LIPPNTT;05:40;06:49;45
 13/07/2009;LIPP;Conf 7B;LIPPNNU;05:40;06:49;44
 13/07/2009;LIPP;Conf 7B;LIPPSL;05:40;06:49;42
 13/07/2009;LIPP;Conf 7B;LIPPSU;05:40;06:49;47
 13/07/2009;LIPP;Conf 7B;LIPPTS;05:40;06:49;43

9.7.66 seg - Segment file

seg				
extension:	seg			
origin:	used as an input for 3D density			
separator:	blank			
sort:	NO			
comment:	each line contains a 3D segment with its associated load			
#	Field	Type	Size	Comment
1	segment ID	num		uniq for a given 3D path
2	lat1	char		in minute decimal
3	lon1	float		in minute decimal
4	lat2	float		in minute decimal
5	lon2	float		in minute decimal
6	lev1	float		unit is FL
7	lev2	float		unit is FL
8	load	float		could be a floating number

Example:

```
97930540 2862.1 2123.3 2863.4 2116.3 56 38 229.647
97930540 2863.4 2116.3 2872.0333 2118.9833 38 0 167.57
97930541 2508.0167 734.3333 2508.2333 734.25 0 30 10.1385
97930541 2508.2333 734.25 2489.6333 729.6833 30 154 5219.27
```

9.7.67 sel - Sector Entry List

Sector Entry List	
extension	sel
origin	Export / SAAM to IPAS
separator	semicolon
sort	by callsign, but from Excel any other sort are possible
comment	Describes the sequence of crossed sectors (with entry level and entry

				time) flight per flight
#	Field	Type	Size	Comment
1	callsign	char		
2	origin	char		ICAO airport code
3	destination	char		ICAO airport code
4	aircraft type	char		ICAO aircraft type
3 x Fields below are repeated a certain number of times (on the same line) and sorted by increasing entry time				
5	entry time	char		formated: HH:MM:SS
6	entry level	char		formated: FLxxx
7	sector name	char		

Example (2 sectors are crossed):

BAW1234;EGLL;LFMN;B738;12:34:10;FL247;EGABC;13:05:59;FL340;LFFVU
--

9.7.68 sid - List of airport SID points

List of airport SID points				
#	Field	Type	Size	Comment
1	extension	sid		
	origin	SAAM		
	separator	blank		
	sort	no sort		
	comment			
#	Field	Type	Size	Comment
1	airport_name	char	4	ICAO airport name
2...N	point_name	char	~	Airport SID (Standard Instrument Departure) navigation point

Example:

BIKF THORI ALDAN
BIRK THORI ALDAN
DTMB MON.D
DTNH NBA.D

9.7.69 sls - Sector List file

Sector List	
extensio	sls
n:	
origin:	Airspace Editor
separato	blank
r:	
sort:	grouped by sector name
commen	input of SAAM Traffic intersection / Airspace Editor ... The link between this
t:	and newmaxo ascii region file ARE is made by volume name.

NORMAL LINE				
#	Field	Type	Size	Comment
1	sector name	char		max 19 characters
2	volume sign	char	1	always "+" indicates that the volume has to be added to the sector. Note that subtraction is deprecated and should not be used any longer.
3	volume name	char		refer to name of volume IN the ARE file (max 24 characters).
4	volume bottom level (*)	num	1 - 3	in FL, overwrite level present in ARE file for that volume (leading os are not mandatory).
5	volume top level (*)	num	1 - 3	in FL, overwrite level present in ARE file for that volume (leading os are not mandatory).

Note: previous sls format accepted meta line, this feature is not supported anymore and can not be used, instead use ".spc" format.

Example:

```
CASTOR_UP + AX 000 295
CASTOR_UP + GU 245 265
TMA + SLP 000 095
TMA + SLX 000 125
ABRON + TOP 45 660
```

9.7.70 slt - Graph filter file

Graph filter file	
extension	slt
origin	
separator	no separator since field 1 is the complete line.
sort	any order
comment	this file contains a list of sectors (one name per line). It is used by SAAM chart to hide the graph of the sectors that are not present in the filter file.

#	Field	Type	Size	Comment
1	sector name	ascii string compose of letters (a-z A-Z), digits (0-9) and special characters (_-@)		Characters after the # are considered as comment
	partial sector name: begin with			
	comment	ascii characters		First character must be #

Example:

```
# A comment always start with #
EB    # All sectors beginning with EB
LFFF
```

EDDD

9.7.71 soa - Area Definition file

Area Definition	
extension	soa
origin	SAAM
separator	blank (space or tabulation)
sort	an area has to be defined prior to any usage in another area definition (see Roissy in example below)
comment	used to simplify segment management rule writing for SAAM assignment. an area is a Set of Airports. Area Definitions appear in RAD appendix 2.

#	Field	Type	Size	Comment
1	Area name	char	>= 5	name used in segment management rules. Name must be at least 5 characters long.
2..N	Airport or sub-Area name	char	>= 4	list of airport or sub-area names, blank (space or tabulation) separated. When a sub-area is used, the associated list of airport names is used recursively. Airport name may contain "." character to replace any [A-Z] character, like in "ED.."

Example:

```

Ajaccio LFKF LFKG LFKJ LFKO
Amsterdam      EHAM EHHV EHKD EHLE EHRD EHTX
Roissy   LFPB  LFPC  LFPG  LFPT
Paris    Roissy      LFPN  LFPO  LFPV
Italy     LI..

```

9.7.72 so6 - Traffic file

SegOut6	
extension	so6
origin:	SAAM profile processing or PRISME Database or DDR
separator:	blank
sort:	by flight segment sequence from origin to destination
comment:	Describes flight 4D trajectories, 4D segment per 4D segment. This trajectory can come from different sources: - SAAM (trajectories could be shortest or cheapest). SAAM calculates vertical

	<p>profile possibly using operational constraint (from .flc2 file) and mix of BADA & Network Operations aircraft performance.</p> <ul style="list-style-type: none"> - NM via DDR (trajectories could be: M1=Model 1=last filled flight plan, M3=Model 3=flight plan enhanced with Radar Data, ...). It should be noted that NM processes the vertical profile from flight plan using possible operational constraint and their own aircraft performance. - pure Radar Data (called CPR data) - other sources. <p>This file could also contains flight options (flight ID ≥ 1000000000), in such case it should be accompanied with a ".lox" file.</p> <p>This file is used as an input in many other SAAM processes.</p>
--	--

#	Field	Type	Size	Comment
1	segment identifier	char		first point name "_" last point name (see note)
2	origin of flight	char	4	ICAO code
3	destination of flight	char	4	ICAO code
4	aircraft type	char	4	
5	time begin segment	num	6	HHMMSS padded with o's
6	time end segment	num	6	HHMMSS padded with o's
7	FL begin segment	num	1 to 3	
8	FL end segment	num	1 to 3	
9	status	char	1	0=climb, 1=descent, 2=cruise
10	callsign	char		
11	date begin segment	num	6	YYMMDD padded with o's
12	date end segment	num	6	YYMMDD padded with o's
13	lat begin segment	float		in minute decimal
14	lon begin segment	float		in minute decimal
15	lat end segment	float		in minute decimal
16	lon end segment	float		in minute decimal
17	flight identifier	num		same as the one provided in expand file (must be uniq). In case of flight option it is ≥ 1000000000
18	sequence	num		start at 1 for every new flight, incremented at each lines. IMPORTANT !
19	segment length	float		in nautical miles

20	segment parity/color	num	o=NO(grey, R=102, G=102, B=102), 1=ODD(green, R=60, G=255, B=60), 2=EVEN(blue, R=100, G=100, B=255), 3=ODD_LOW(dark green, R=0, G=200, B=0), 4=EVEN_LOW(light blue, R=160,G=160,B=255), 5=ODD_HIGH(light green, R=160, G=255, B=160), 6=EVEN_HIGH(dark blue, R=0, G=0, B=200), 7=General Purpose Red Color (R=255, G=0, B=0), 8=General Purpose Orange Color (R=255, G=128, B=0), 9=General Purpose Yellow Color (R=255, G=255, B=0)
----	----------------------	-----	--

Note: point name are 3 or 5 characters for route points and 4 characters for airports.

Route point starting with a \$ or % are technical points created to refine the 4D definition of the trajectory, like Top of Climb and Top of Descend. If the trajectory has been cut exactly at a border using the process [Traffic Punching](#) new points has been created. These point names start with #.

NM route point having coordinates but no name should be identified with ! followed by 4 digit like: !0034

Example (2 flights):

```

DAAG_ALR DAAG LEBL B727 150000 150045 0 21 0 DAH2012 970905 970905
2202 193 2200 190 110 1 3.12859 1
ALR_$aaaa DAAG LEBL B727 150045 151725 21 330 0 DAH2012 970905 970905
2200 190 2298.33 221.72 110 2 101.5 1
$aaaa_MHN DAAG LEBL B727 151725 153013 330 320 2 DAH2012 970905
970905 2298.33 221.72 2391 253 110 3 95.7977 1
MHN$_aab DAAG LEBL B727 153013 153632 320 320 2 DAH2012 970905
970905 2391 253 2422.74 207.23 110 4 47.25 1
$aab_TASOS DAAG LEBL B727 153632 154356 320 145 2 DAH2012 970905
970905 2422.74 207.23 2456 158 110 5 50.0106 1
TASOS_QUV DAAG LEBL B727 154356 155155 145 4 2 DAH2012 970905 970905
2456 158 2477 125 110 6 32.5458 1
QUV_LEBL DAAG LEBL B727 155155 155213 4 0 2 DAH2012 970905 970905
2477 125 2477 124 110 7 0.751456
EBBR_BRU EBBR EKBI B73V 173000 173100 0 18 0 DAN258 970905 970905
3054 269 3054 262 462 1 4.41473 1
BRU_NIK EBBR EKBI B73V 173100 173516 18 93 0 DAN258 970905 970905
3054 262 3070 251 462 2 17.4314
NIK_%EH21 EBBR EKBI B73V 173516 173842 93 150 0 DAN258 970905 970905
3070 251 3086 241 462 3 17.1783 1
%EH21_SPY EBBR EKBI B73V 173842 174958 150 285 0 DAN258 970905 970905
3086 241 3152 291 462 4 72.8294 2
SPY_GRONY EBBR EKBI B73V 174958 175618 285 339 0 DAN258 970905 970905
3152 291 3183 346 462 5 45.4656

```

```

GRONY_$aage EBBR EKBI B73V 175618 180012 339 370 0 DAN258 970905
970905 3183 346 3197.06 386.396 462 6 28 2
$aage_BEDUM EBBR EKBI B73V 180012 180059 370 370 2 DAN258 970905
970905 3197.06 386.396 3200 395 462 7 5.92438 2
BEDUM_NL1 EBBR EKBI B73V 180059 180320 370 370 2 DAN258 970905 970905
3200 395 3213 415 462 8 17.6328
NL1_$aagf EBBR EKBI B73V 180320 180802 370 370 2 DAN258 970905 970905
3213 415 3246.62 436.807 462 9 36 2
$aagf_VES EBBR EKBI B73V 180802 182220 370 103 2 DAN258 970905 970905
3246.62 436.807 3336 498 462 10 96.0673 2
VES_EKBI EBBR EKBI B73V 182220 183034 103 0 2 DAN258 970905 970905
3336 498 3344 549 462 11 29.8559 2

```

9.7.73 sos - Sector Boolean Opening Scheme file

Sector boolean Opening Scheme				
version:	1			
extensio n:	.sos			
origin:	SAAM, input & output from optimisation			
separato r:	tab			
sort:	no sort			
commen t:	Describes sector per sector, possible opening time, independently from other sectors (overlap possible in a given sector family) Number of columns is variable (below n* is 25 for 24 periods of 1 hour, or 49 for 48 periods of 1/2h, or 97 for 96 periods of 1/4h)			
Header				
#	Field	Type	Size	Comment
1	blank		0	nothing here
2	time	char		time expressed HHMM (zero padded)
...	time			
n*	time	bool		time expressed HHMM (zero padded)
Body (Matrix)				
#	Field	Type	Size	Comment
1	sector name	char		could be elementary sector or collapse sector, must be found in .cfg file
2	opening info	bool		1 means open for this period of time, 0 means closed for this period of time
....	opening info	bool		1 means open for this period of time, 0 means closed for this period of time
n*	opening info	bool		1 means open for this period of time, 0 means closed for this period of time

Example (24 periods of 1 hour; 6 sectors; displayed via excel with coloured cells):

C1_285_355	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
C1_285_660	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
C1_355_660	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
C1_C2_285_660	1	1	1	1	1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	1
C1_C2_Lshape	1	1	1	1	1	1	1	0	0	1	1	1	1	0	0	1	1	0	0	1	1	1	1	1	1
C1_C2_U_285_355	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

9.7.74 spc - Airspace

NAME:	Airspace			
extension	spc			
origin:	Gasel, SAAM			
separator:	(semicolon)			
sort	no sort, this file could be self referenced			
comment:	Describe how elementary sectors can be collapsed, can also contains description of other bigger airspace (like ACC ...)			
FIRST LINE				
#	Field	Type	Size	Comment
1#	char		1	
2file type	text			= AIRSPACE
3version	num			= 2
4NM AIRAC cycle	num			
5AIRAC start date	date			format is yyyyMMdd
6AIRAC end date	date			format is yyyyMMdd
7number of record	num			
8datasource	text			
HEADER				
#	Field	Type	Size	Comment
1A	char		1	A=Airspace
2Airspace ID	text			
3Airspace name/ description	text			
4Airspace type	text			Allowed values: CS, CRS, AUA, CLUS, NAS, AREA, AUAG, CRAS, REG, UNK
5Nb of sub-airspace	num			= number of BODY lines
6Airspace category	char		1	"T" = Test " " = no data
BODY				
#	Field	Type	Size	Comment
1S	char		1	S=Sector
2Sub-airspace name	text			
3Sub-airspace type	text			

Example:

```
#;AIRSPACE;2;383;20140206;20140305;4082;EAR_P
A;BENELUX;EB/EH;AREA;2;_
S;EB;NAS
```

```

S;EH;NAS
A;BG;GREENLAND (DENMARK);NAS;1;_
S;BGGLFIR;FIR
A;BI;ICELAND;NAS;3;_
S;BIFAROER;FIR
S;BIRD;FIR
S;ENJA;FIR
A;BICC;ICELAND AUAG;AUAG;6;_
S;BIRDCTA;AUA
S;BIRDICTA;AUA
S;BIRDITMA;AUA
S;BIRDTOMA;AUA
S;EKVGCTR;AUA
S;ENJACTR;AUA

```

9.7.75 ssl - SID and STAR list

Airspace				
extension	ssl			
origin	IPAS			
separator	blank			
sort	no sort			
comment	Optional management of suppression of SID/STAR points to/from airports within the area to be simulated is triggered via a user input text file having .ssl extension (if empty the SID/STAR management does not take place, but the conversion to IPAS will be valid)			
#	Field	Type	Size	Comment
1	airport_name	char	4	number of airspace structures defined
2	D or A	char	1	A or D for Arrival or Departure
3	point_name	char	~	last point of the SID or first point of the STAR

Example:

```
LSGG D JOHAN
```

9.7.76 star - List of airport STAR points

List of airport STAR points				
extension	star			
origin	SAAM			
separator	blank			
sort	no sort			
comment				
#	Field	Type	Size	Comment

1	airport_name	char	4	ICAO airport name
2...N	point_name	char	~	Airport STAR (Standard Terminal Arrival Route) navigation point

Example:

```
BIKF THORI METIL ASTAM KEF CELLO
BIRK THORI METIL ASTAM KEF CELLO
DTMB MON.A
DTNH NBA.A
```

9.7.77 t5 - Airspace Traffic Intersection file

T5	
extensio n	T5
origin	produced by SAAM Airspace Traffic Intersection
separato r	blank
sort	Important ! 1/ field 1 (flight ID), 2/ field 2 (sector name), 3/ field 3 (entry date/time) as numeric value
comment	each line contains entry and exit information for a flight for a sector (n lines for n entry/exit of the same flight in the same sector)

#	Field	Type	Size	Comment
1	ID	num		flight ID
2	sector name	char		
3	entry date/time	float		unit is second, starting 01/01/1970
4	exit date/time	float		unit is second, starting 01/01/1970
5	entry FL	float		unit is Flight Level decimal
6	exit FL	float		unit is Flight Level decimal
7	entry distance	float		unit is NM decimal, from beginning of segment
8	exit distance	float		unit is NM decimal, from beginning of segment
9	entry segment name	char		separator between point name is " " "<#>" as a prefix means that the flights was already present in the sector before it was opened
10	exit segment name	char		separator between point name is " ", might be different from field #9 "<#>" as a prefix means that the flight was still present in the sector after it was closed
11	total distance spent in sector	float		for that flight ID for that sector, expressed in NM decimal
12	total time spent in sector	float		for that flight ID for that sector, expressed in second decimal

Example:

```
16883 LQSB 2197702.791411 2198377 245.000000 4.000000 23.503334
90.142200 $amqx_DBK $amqx_DBK 66.6389 674.209
16964 LQSB 2182545.461068 2183300.537904 230.000000 175.296125
53.386000 12.871500 ZAG_MIRJA $amtD_SPL 51.4885 755.077
```

```

17262 LQSB 2204806.486014 2205400.493804 245.000000 62.522849
12.452444 62.947700 $amEI_DBK $amEI_DBK 50.4953 594.008
17546 LQSB 2202699.898361 2202784.662148 245.000000 206.642500
14.200000 21.871500 MIRJA_SPL MIRJA_SPL 7.6715 84.7638
18151 LQSB 2202533.576397 2204697.097729 240.000000 240.000000
21.281700 105.638700 VAL_SAR SAR_PUL 150.234 2163.52
19454 LQSB 2213785.364431 2215008.500000 240.000000 245.000000
14.792400 48.395400 $aokT_SAR SAR_SPL 88.3458 1223.14

```

9.7.78 tax - Turn Angle Exception file

penalisation				
#	Field	Type	Size	Comment
1	Point 1	char	26	defines with Point1 the incoming segment of the turn
2	Point 2	char	26	the turn point
3	Point 3	char	26	defines with Point2 the outcoming segment of the turn

Example :

```

ADOKI SKAVI KRW
ADOSA VIL02 VIL.A
ADUSO BENAK POBIX
ADUTO FERDI OKT.A
ADX POS MUROS
AGB.D RIDAR LELTA

```

9.7.79 tcr - T5 Cleaning Rule file

T5 Cleaning Rule				
#	Field	Type	Size	Comment
1	rule ID	num		rule identifier
2	line ID	num		o is header,

			1 is free comment, 2 is flight query criteria (can be multi lines), 3 is list of airspace name
3	info	variable	blank separated, content depends on line ID (see notes)

Notes:**Line ID=0 (header) fields are**

- version number (=0),
- toggle (=1 means rules is activated, =0 means rules disabled)
- date creation of the rule (DD/MM/YYYY)
- time creation of the rule (HH:MM:SS)

Line ID=1 (free comment):

- contains what user puts, can be used to search a rule

Line I [t5 - Airspace Traffic Intersection file](#) D=2 (flight query criteria):

- contains the exact values of query criteria, used to identify the flights that must be deleted from airspace
- In case of a qif criteria several line ID=2 will be present

Line ID=3 (list of airspace name):

- contains the list (blank separated) of airspace from which the set of identified flights will be deleted

Example (2 rules)

```

1 0 0 1 08/01/2004 18:01:11
1 1 Depart Bruxelles
1 2 EBBR_EG??
1 3 EBBUEST
2 0 0 1 09/01/2004 10:44:12
2 1 Depart NICE
2 2 LFMN_
2 2 BLONA
2 3 LFMMTOP LFMMSOUTH

```

9.7.80 tdv - Three Dimensional Viewer file

Three Dimensional Viewer	
extensio	tdv
n	
origin	file used to describe a 2D/3D/4D scene to visualized with SAAM

separato r	blank
sort	no, but order is important
comment	ascii file describing shapes used as an input in SAAM for 3D terrain display

Important:

- Each command must be on its own line.
- The first token (word or letter) in the line represent the command described below as the "**TDV code**" or command (see table below)
- The list of **parameters** (in bold) depends on the TDV code, they are separated by a blank/space character ! These parameters must be replaced by a number or a string of characters. If a name or a text contains a blank/space, it MUST be replaced by '@' character. If a file contains a blank/space, it must be replace by a '?'. SAAM will recognise these specific cases and will change them to a blank/space internally to make it working. See for example, the command #5 (Dx) below with "text@to@display" that will be displayed as "text to display".
- The order of commands is normally not important, except for COEF TDV code, but might impact the order of display in 2D.
- Empty lines can be present to group the commands and/or make the TDV file easier to read.
- Comments are accepted behind the # character or // double slashes. Grouping lines into comment can be done with /* to start (on a single line !) and with */ to finish (on a single line). In the example below the 5 lines in the middle are commented:

```
D AKIVO X 3598.000000 -565.000000 0.500000 C 255 0 0 0
P S AKIVO 0 3598.000000 -570.000000 0.000000 390.000000 1.000000 C
255 0 0 0
/*
L S AKIVO_STN_390_390 23 3598.000000 -570.000000 390.000000
3492.000000 -371.000000 390.000000 30.000000 C 255 0 128 255
D STN X 3492.000000 -366.000000 0.500000 C 255 0 0 0
P S STN 0 3492.000000 -371.000000 0.000000 390.000000 1.000000 C 255
0 0 0
L S RUGID_STN_390_390 23 3492.000000 -371.000000 390.000000
3442.150000 -293.917000 390.000000 30.000000 C 255 0 128 255
D RUGID X 3442.150000 -288.917000 0.500000 C 255 0 0 0
*/
P S RUGID 0 3442.150000 -293.917000 0.000000 390.000000 1.000000 C
255 0 0 0
L S FINDO_RUGID_390_390 23 3442.150000 -293.917000 390.000000
3382.150000 -207.817000 390.000000 30.000000 C 255 0 128 255
```

- A * or (*) in the table below indicates that more information are given after and below the table

#	Command type	TDV code/ Command	Parameters
1	Airspace	V	aspect(S/T)* top(L N T O)Border(B X C)Inside(I X)* file_name.are* bot_level(FL) top_level(FL)

			COLOR*
	Airspace	Vx	version(=1) aspect(S/T)* top(L N T O)Border(B X C) Inside(I X)* file_name.are* bot_level(FL) top_level(FL) COLOR(*) border_thickness border_COLOR*
2	3D Line	L	aspect(S/T)* segment_name line_shape* lat1 lon1 alt1 lat2 lon2 alt2 size COLOR*
3	3D Point	P	aspect(S/T)* point_name point_shape* lat lon alt_base height size* COLOR*
4	3D Designator	D	text@to@display aspect(F/B/T/X not implemented) lat lon alt COLOR*
5	3D Designator	Dx	version(=1) text@to@display 2nd@line lat lon alt feature_version(=1) aspect(=F/B/T/X*) frame_thick(=0/1) text_alignment(=L/C/R) post(=0/1) post_shiftX post_shiftY post_alignment(L/C/R) font_logical_name* auto_color_text(0/1) COLOR_text(*) auto_color_frame(0/1) COLOR_frame(*) auto_color_background(0/1) COLOR_background(*)
6	font	FONT	version(=1) font@logical@name size(in point, max 200) weight(100=thin, 400=normal, 900=heavy) italic(0 or 1) underline(0 or 1) strikeout(0 or 1) charset(ANSI or SYMBOL) system@font@name
7	Camera rotation (unique object)	XORIENT	angle_rotation_movement (0 means stable) angle_rotation_value (in tens of degrees decimal, approximately 0 means top of screen is north oriented, 450 means top of screen is west oriented). Caution: this is a pure euclidian angle (not north oriented). A tild value for any of the 2 parameters means current value is unchanged.
8	Camera tilt (unique object)	YORIENT	tilt_angle_movement (0 means stable) tilt_angle_value (in tens of degrees decimal: 0=top view, -450 side view) A tild value for any of the 2 parameters means current value is unchanged.
9	Camera zoom (unique object)	ZOOM	zoom_movement (0 means stable) zoom_value (0 means eye touching the object, 1000 view airport, 10000 view approach, 40000 view small country, 100000 view large country, 280000 view europe) A tild value for any of the 2 parameters means current value is unchanged.
10	Camera East/West translation (unique object)	XMOVE	x_translation_movement (0 means stable) x_translation_value (west-east oriented, +value means goes to west, -value goes to east, independently from rotation angle) A tild value for any of the 2 parameters means current value is unchanged.

11	Camera North/ South translatio n (unique object)	YMOVE	y_translation_movement (o means stable) y_translation_value (south-north oriented, +value means goes to south, -value means goes to north, independently from rotation angle) A tild value for any of the 2 parameters means current value is unchanged.
12	Camera vertical ref (unique object)	YVMOVE	ground_ref_movement (o means stable) ground_ref_value (o means vertical reference is the ground, -value goes above ground, +value goes below ground) A tild value for any of the 2 parameters means current value is unchanged.
13	Camera changes absorptio n (unique object)	AMORT	XORIENT YORIENT XMOVE YMOVE ZOOM YVMOVE (movement absorber, value with decimal, 99999 means stop movement, o means never stop movement, +value means stop gradually, -value means accelerate gradually if movement value != o). These values change the movement parameter of the associated objects. A tild value for any of the 4 parameters means current value is unchanged.
14	Cube	C	aspect(S/T)* name lat1 lon1 lat2 lon2 lat3 lon3 lat4 lon4 bot_level top_level COLOR*
15	Flight Animatio n (old version)	SANIM	file_to_animate.so6 FROM_MEMORY FROM_SOCKET* aircraft_shape* aircraft_size* COLOR* Caution: does not handle "Custom Cylinder Height" aircraft shape.
16	Flight Animatio n (new version)	SANIMx	version(=1) file_to_animate.so6 \.\.\pipe\ \mynamedpipe FROM_MEMORY FROM_SOCKET* aircraft_shape*) aircraft_size* COLOR* volume_name_to_be_moved (else NULL; if used it requires aircraft_shape value to be o and a single flight in the so6 file) Caution: version 1 does not handle "Custom Cylinder Height" aircraft shape. version(=2) file_to_animate.so6 \.\.\pipe\ \mynamedpipe FROM_MEMORY FROM_SOCKET* aircraft_shape* aircraft_size* COLOR* volume_name_to_be_moved (else NULL; if used it requires aircraft_shape value to be o and a single flight in the so6 file)
17	Density Anim	DENS	ori_lat ori_lon ori_lev size_airspace_East size_airspace_North size_airspace_Height size_cell_East size_cell_North size_cell_Height count_meth(o or 1 or 2) para thresh(Red,Orange,Yellow,Green) Notes: - a single command should be present in a TDV, and should

			appear after SANIM TDV command (due to coef processing) - count_meth: 0=exact count with classical cell display, 1 count is made with a trailing average of para length, 2 exact count with smooth color display
18	Clock Anim (unique object)	CLOCK	start (in s) end (in s) step (in s decimal) clock_wrap (=LOOP or NOLOOP*) Note: see comment below for NOLOOP.
19	Clock Anim (unique object)	CLOCKx	version (=1) start (*) end (*) [z] step (in s decimal) clock_warp (=LOOP or NOLOOP*) display_status (default is NTS*)) Notes: If step parameter is preceded with "z" time speed animation is Zoom dependent (quicker with zoom out, slower with zoom in, same speed with europe scale "HOME" pressed) If NOLOOP is set the animation stops when clock reached the end, but this mode might imply : - if presentation mode is ON with timer set to 0, clock reaching end of animation will open next TDV file automatically, allowing synchronisation of events (with following time value or not) amongst different TDV files.
20	Clock pause (unique object)	PAUSE	(no parameters, comment: linked to keyboard pause key, freeze the clock)
21	Terrain	TERRAIN	file_name.ter (specific SAAM format)
22	Vertical magnifier (unique object)	COEF	altitude_coef (default is 50, if used must appear before any other TDV command)
23	Background white color (unique object)	WHITE	
24	Background Color (unique object)	BACK_COLOUR	COLOR (*)
25	Window frame screen position (unique object)	PREF_POS	x1 y1 x2 y2 (position of the screen in window coordinates - pixels, command valid in former Unix version, not yet implemented on Windows)
26	Lambert project para (unique object)	LAMBPARA	unit (DEG/MIN) latNorth latSouth lonEast lonWest Provide a new parameter used or internal Lambert Secant projection

27	Airport	RUNWAY	name lat1 lon1 elev1(in FL) stop1(not used) lat2 lon2 elev2(in FL) stop2(not used) width(in NM) COLOR*
28	Line comment	//	
29	start block comment	/*	Note: must be on a single line
30	end block comment	*/	Note: must be on a single line
31	Route Network	NETWOR K	version(=1) display(=LOAD PARITY) network_file.ase COLOR*
32			version(=2) display(LOAD/PARITY) network_file.ase red_thresh orang_thresh green_thresh blue_thresh min_grey_thresh COLOR*
33			version(=3) display(=DEFAULT PARITY LOAD TYPE MIXED) network_file.ase red_thresh orang_thresh green_thresh blue_thresh grey_thresh size_point size_segment size_arrow label_on_off(0 or 1) label_size label_frame COLOR*
34			version(=4) display(=DEFAULT PARITY LOAD TYPE MIXED) network_file.ase red_thresh orang_thresh green_thresh blue_thresh grey_thresh size_point size_segment size_arrow label_on_off(0 or 1) label_size label_frame label_load COLOR*
35			version(=5) display(=DEFAULT PARITY LOAD TYPE MIXED) network_file.ase red_thresh orang_thresh green_thresh blue_thresh grey_thresh size_point size_segment size_arrow label_on_off(0 or 1) label_size label_frame label_load nb_load_decimal COLOR(*) version(=6) display(=DEFAULT PARITY LOAD TYPE MIXED) network_file.ase red_thresh orang_thresh green_thresh blue_thresh grey_thresh size_point size_segment size_arrow point_label_on_off(0 or 1) load_label_on_off(0 or 1) load_label_nb_deci COLOR(*) point_label_feature={version(=1) aspect(=F B T X) thick text_align(=L R C) post_on_off(0 or 1) post_x post_y post_align(=L R C) logical_font_name text_auto_color_on_off(0 or 1) text_COLOR(*) frame_auto_color_on_off(0 or 1) frame_COLOR(*) background_autocolor_on_off(0 or 1) background_COLOR(*)}

			<pre> load_label_feature={version(=1) aspect(=F B T X) thick text_align(=L R C) post_on_off(0 or 1) post_x post_y post_align(=L R C) logical_font_name text_auto_color_on_off(0 or 1) text_COLOR(*) frame_auto_color_on_off(0 or 1) frame_COLOR(*) background_autocolor_on_off(0 or 1) background_COLOR(*)} version(=7) display(=DEFAULT PARITY LOAD TYPE MIXED CDR) network_file.ase red_thresh orang_thresh green_thresh blue_thresh grey_thresh size_point size_segment size_arrow point_label_on_off(0 or 1) load_label_on_off(0 or 1) load_label_nb_deci COLOR* </pre> <pre> point_label_feature={version(=1) aspect(=F B T X) thick text_align(=L R C) post_on_off(0 or 1) post_x post_y post_align(=L R C) logical_font_name text_auto_color_on_off(0 or 1) textCOLOR* frame_auto_color_on_off(0 or 1) frame_COLOR* background_autocolor_on_off(0 or 1) background_COLOR*} load_label_feature={version(=1) aspect(=F B T X) thick text_align(=L R C) post_on_off(0 or 1) post_x post_y post_align(=L R C) logical_font_name text_auto_color_on_off(0 or 1) text_COLOR* frame_auto_color_on_off(0 or 1) frame_COLOR* background_autocolor_on_off(0 or 1) background_COLOR*} label_point_min_converg_seg label_point_max_converg_seg label_point_min_diverg_seg label_point_max_diverg_seg label_load_min_value label_load_max_value </pre>
36	Diagram	DIAGRAM	<pre> title Y_axe_70%_written_value(max 3 digits) position(0=left side, 1=right shifted, 2=more right shifted, 3=right maxi) COLOR* </pre>
37	Diagram	CURVE	<pre> title data_file_name(file extension ".tim" have a info line header) attached_diagram(0 to 3) text_position(1 to 6) data_file_column(*) factor(*) curve_display_flag(Y or N used with DIAGRAM object) attached_object(*see explanation below) COLOR* </pre> <p>Note: if the second column of ".tim" file (linked to this CURVE object) is called in the header (first line) "NB_CLOCK_TURN", the animation will be effective only if</p>

			the number of clock turns (calculated internally by SAAM) matches the value found in the second column. This particular column does not affect data_file_column positionning.
38	title on screen	TITLE	<p>version=1 aspect(S/T) text_to_display(blank=@) size_of_the_font screen_coord_x(*) screen_coord_y(*) COLOR</p> <p>version=2 aspect(S/T) text_to_display(blank=@, one '#'=numbering part*coef coming from CURVE for animation, or one'^'=current clock time does not more require CURVE) size_of_the_font screen_coord_x(*) screen_coord_y(*) shadow_x shadow_y COLOR(*)</p> <p>version=3 aspect(S/T) test_to_display(blank=@, one '#'=numbering part*coef coming from CURVE for animation, or one'^'=current clock time does not more require CURVE) destination(3D_SCENE or SCREEN) font@system@name(like "Arial") extrusion(depth of the character, 0 for SCREEN) deviation(not used should be 0) size(0 is extremely small, visible size should be around 1000.0, same value as version previous version for SCREEN) latitude(located in the middle of the text for 3D_SCENE or X coordinates on the left of the text for SCREEN*) longitude(located in the middle of the text for 3D_SCENE or Y coordinates for SCREEN*) altitude (in flight level, located in the middle of the text for 3D_SCENE, 0 for SCREEN) rotation_X(expressed in degrees, can be over 360.0, title turn around horizontal axe) rotation_Y(expressed in degrees, can be over 360.0, title turn around vertical axe) rotation_Z(expresed in degrees, can be over 360.0, title tilt) shadow_x(only for SCREEN, 0 for 3D_SCENE) shadow_y(only for SCREEN, 0 for 3D_SCENE) COLOR(*)</p>
39	picture	IMAGE	<p>version=1 aspect(S/T; T is not used) file_name.bmp size_factor(in decimal; 0=very very small, 1.0=original file size) lat lon alt</p> <p>Note: 3D location (lat lon alt) is defined at the bottom, in the center of the picture</p> <p>version=2 aspect(S/T; T is not used) destination(3D_SCENE SCREEN) file_name.bmp size_factor(in decimal; 0=very very small, 1.0=original file size) lat lon alt</p> <p>Note: if 3D_SCENE then bitmap is displayed in the scene as a 3D object (bitmap origin is centered in the lower part), else it is display on the top of the SCREEN and lat & lon respectively are x and y in relative pixels (bitmap x/y)</p>

			origin is lower left corner of the screen) and alt is not used.
			<p>version=3 aspect(S/T; T is not used) destination(3D_SCENE SCREEN) file_name.bmp size_factor(in decimal; 0=very very small, 1.0=original file size) origin(2 letters:x-axe={Left,Centre,Right}, y-axe={Top,Center,Bottom}, example: CB for Center & Bottom) lat lon alt</p> <p>Note: if 3D_SCENE then bitmap is displayed in the scene as a 3D object (default bitmap origin is managed by origin parameter), else it is display on the top of the SCREEN and lat & lon respectively are x and y in relative pixels (default bitmap origin is lower left corner of the screen) and alt is not used.</p>
40	image is 2D (unique object)	2D	No parameter, by default TDV image is 3D, this command displays it in 2D at first
41	comment	# or //	allows to comment and/or to disable TDV command, must be located at the beginning of the line to be commented
42	camera angle (unique object)	CAM_ANGLE	version(=1) angle (expressed in degrees decimal ranges from 1.0=no perspective to 179.0=very high distortion, default is 30.0)
43	2 views on screen (unique object)	SCREENSHARE	<p>version(=1) which_location(TOP_SCREEN or BOTTOM_SCREEN or LEFT_SCREEN or RIGHT_SCREEN)</p> <p>Note: used with presentation "pre" file, to display 2 TDV files in same time on the same screen</p>
44	Hidden surface display removal (unique object)	CULLFACE	No parameter. By default SAAM displays hidden surface, this command stops displaying them. In this case, and if your surface is anti-clockwise, it will be not displayed correctly. Note: special object (like transparent Title object in 3D scene) are not affected by this command.
45	Enable Stereo mode (unique object)	STEREO	<p>version(=1)</p> <p>Note 1: will display the current TDV file in stereo (default is no stereo)</p> <p>Note 2: stereo parameters can be changed with shortcut ALT +Z (see keys/effect indications on screen)</p>
46	Layer	LAYERSET	<p>version(=1) status(0 or 1) readwrite(RO or RW) path_of_directory_containing_dataset</p> <p>Note 1: the last parameter contains a sub-directory called baseline which have all data files for the baseline (files of dataset coming from DDR) and *.prj files (one per proposal).</p> <p>Note 2: several LAYERSET commands are possible except if the same project is open in mode read-write by several SAAM, to avoid concurrent access problem (see c:)</p>

			\SAAM_USER_PREF\layer_lock.txt file) Note 3: this command must be written before any other command which reference a dynamic 'oooo' file.
47	Object hierarchy	GROUP	version(=1) display(o=hide, 1=show) skname (all object that belongs to the group will have this skname appended) Start implementing TDV objects grouping, all TDV object after this GROUP will be together.
48	Object hierarchy	END	version(=1) end of previous GROUP. All objects before this END belongs to the same group.

Airspace "file_name.are" :

will contains the name of the ".are" file located in the current working directory, if located elsewhere, then a path (absolute or relative must be present). If a blank character is present in the path or in the name it must be replaced by a '?' character.

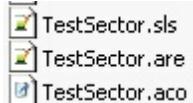
If the name and the possible leading path are all CAPITAL letters (uppercase) then the whole file is not editable (can be seen but not changed).

Special airspace files for the map are recognized by NEST as they do not need a path (NEST find them automatically in its own directory), they are:



Usually, an airspace .are file is associated with a .sls file which indicates DFL and grouping of sectors from the content of airblocks coordinates (which are described in the .are file). NEST read .sls file automatically, but they need to be together in the same directory, with the same name. Same thing with possible .aco file which describes the colour of sectors.

For instance:



Airsaces Color

If not null (= "C o o o o"), the color defined in the V or Vx objects are has the priority over the colors defined for each airspace in the ACO or ARE files.

Aspect field for Designator (Dx command) means:

- F=Frame,
- B=Background,
- T=Two(frame+background),

- X=(no frame and no background)

font_logical_name is the logical name given in the FONT command else it can be "DEFAULT" for a Fixedsys50oc.ttf font of size 12 normal thickness (which is installed automatically by SAAM)

COLOR = "C Transparent Red Green Blue" (each TRGB integer value can vary from 0 to 255)

Note: for ANIM or SANIM TDV object a Transparent value equals to 0 (zero) will display in 2D animated aircraft above all other transparent objects, and above the route network.

lat and **lon** are coordinates in minute decimal

height, **altitude** or **level** are in Flight Level decimal, size or distance are nautical miles decimal

aircraft_shape:

- 0=no aircraft displayed,
- 1=a pixel,
- 2=flat triangle,
- 3=3D triangle,
- 4=height limited cylinder,
- 5=max height cylinder,
- 6=noise contour radius fixed,
- 7=noise contour radius varying,
- 8=B737 3D Model, in that case the aircraft size must be adapted
- 9=change camera look-at point,
- 10=change camera location. For camera changes (9 or 10), as well as volume_to_be_moved different from NULL, it is recommended to use single flight so6 files (i.e one single flight so6 file for camera look at and one - possibly different - single flight so6 file for camera location)
- 12=Solar Impulse model. Note: codes running from 12 to 30 are reserved for generic manually created model. This model has always the label indicating its current FL displayed (with text 'FL' in front the value).
- 1AAA BBB. For custom cylinder height. The 3 digits (padded with 0's) AAA corresponds to the value expressed in FL above current aircraft FL, BBB corresponds to the value expressed in FL below current aircraft. AAA and BBB can be set to '000' but can not be negative. Example: 1050020 indicates a cylinder having a height of FL50 above the aircraft current FL and FL20 below.
- 13 only Aircraft FL label is displayed below.

aircraft_size: expressed in floating value.If followed by NM (ex: 50.3NM) then expressed in NM (not valid for pixel shape, look-at point and camera location) if

preceded with "z" aircraft size is Zoom dependant (bigger with zoom out, smaller with zoom in, same with Europe scale "HOME" pressed)

FROM_SOCKET If this source is provided a named pipe is created and is used for reading ARTAS info on line instead of so6 file. The name of the pipe is: "\.\.\pipe\mynamedpipe", with mynamedpipe that can be replaced by any name. Note: the writing pipe must be created before the animation is launched (see for instance prog like "writeToPipe" or "UDP_Pieter"). Several different pipes can be opened into several different SANIM objects, but always a single pipe to a single SANIM object. Note: the case of night animation is not covered and will issue an error message (see clock time)

S/T means Solid or Transparent (TDV code is "S" or "T"), for other feature see below:

Top feature code: L=normal top not lighted, N=normal top lighted, T=solid top not lighted, O=solid top lighted

Border feature code: letter B (border has its own color, by default it is black), X (no border lines) or C (border line has the color of Inside) can be added at the end of top feature code

Inside feature code: letter I (default=inside is showed) or X (inside not displayed) can be added at the end of border feature code

Line shape:

No arrow:

- 0=horizontal line,
- 1=horiz. triangle,
- 2=horiz. square,
- 3=horiz. cylinder low def,
- 4=horiz. Cyl. med. Def.,
- 5=horiz. Cyl. high def.

An arrow in the middle:

- 20=horizontal line,
- 21=horiz. triangle,
- 22=horiz. square,
- 23=horiz. cylinder low def,
- 24=horiz. Cyl. med. Def.,
- 25=horiz. Cyl. high def.

An arrow at the end of the line:

- 40=horizontal line,
- 41=horiz. triangle,
- 42=horiz. square,
- 43=horiz. cylinder low def,
- 44=horiz. Cyl. med. Def.,

- 45=horiz. Cyl. high def.

Point shape:

- 0=vertical line,
- 1=vertical triangle,
- 2=vertical square,
- 3=vertical low definition cylinder (8vertices),
- 4=vertical medium def. Cylinder (16 vert.),
- 5=vertical high def. Cylinder (32 vert.)
- 6=vertical up cone low def.,
- 7=vertical up cone medium def.,
- 8=vertical up cone high def.
- 9=Up Starfish 3 branches,
- 10=Up Starfish 4 branches,
- 11=Up Starfish 5 branches,
- 12=Up Starfish 6 branches,
- 13=Up Starfish 8 branches

Point size is expressed by default in internal unit, except when "nm" (lower case) is added at the end of it, in that case is expressed in nautical mile

TITLE screen coordinates x & y

x coordinates goes from left (minimum value is 0) to right (maximum value is circa 1270) of the screen.

y coordinates goes bottom (minimum value is 0) to top (maximum value is 900) of the screen.

They are located on the lower left corner of the string to be displayed.

CLOCK: start end time parameter. By default expressed in seconds. If value begins with H then expressed in hours minutes seconds (like H134559 for 13h45m59s)
if time start is greater than time end parameter than SAAM switch in "night" mode, and will assume the animation to run from start to midnight and midnight to end.

CLOCK Display Status NTS: 3 letters indicating how the 3 fields of the status line on screen is visible:

- In the order we have: number of aircraft='N', current time of animation (HH:MM:SS #nb clock turn)=T', current step='S'. Default is NTS.
- a 'X' instead of the field code hides the corresponding field. Examples: XTX: show only the time of animation, NXX shows only the number of aircraft animated ...

The "file_name.are" of the TDV code "V" (volume) can be written all in upper case which means the volumes of the file are "hard" locked (no possible edition, no filtering ...), if at least one character is lower case, the volumes of the files can be edited or "softly locked" (temporary user lock, filtering always available).

CURVE data_file_column parameter format is n[H][Rf] with:

- n column number where data must be read, ranges from 1 (column just after the time column) to max 100
- optional tag H means that time column is read with either HH or HHMM or HHMMSS or HHMMSS.decimal; else time column is in second decimal (default)
- optional tag R followed by parameter f means Random, with f is expressing a number of frame. It means that the value processed by SAAM is randomly created every f frame of animation, and ranges between 0 and data value read in data column n and then multiplied by factor (in between 2 frame calculations the latest value is kept). If the data value is 0 the random processing is disabled and the value is set to 0 (generally disabling the object)

CURVE factor:

- Allows to expand or reduce vertical value read from tim file. Factor = 1 means max value found in selected column of the tim file are unchanged (so reaching top of diagram, if a diagram is used).
- If the diagram is used, the formula to calculate the factor value to reach => top of the diagram is: $100 / (\text{max value found in the selected column in the tim file})$
- => 70% line of the diagram is: $70 / (\text{max value found in the selected column in the tim file})$
- => equivalent to 70% written value is: $70 / (70\% \text{ written value})$. Warning: curve might overpass the diagram !!

CURVE attached_object_name

attached_object is either "NULL" (no object attached) or the name of an object (or several objects if the name are the same, depending on object) present in the TDV file (before or after CURVE command, see column 2 "object_name" in the grid below)

When an object is attached, CURVE will change its appearance depending on the object type and on the value read from data_file_column multiplied by factor parameter found in CURVE parameter.

Below is a list of objects that can be attached with their possible graphical effect

TDV object	object_name	effect (n = value read from data_file_column * factor)	comment
L (line)	segment_name	o=disable, n=enable + thickness	max 800 different L objects having the same segment_name can be linked into a single CURVE command
P (point)	point_name	o=disable, n=enable + thickness + size + height	max 800 different P objects having the same point_name can be linked into a single CURVE command
TITLE	text_to_display	a lot of possibilities (see TIM file)	n value can be displayed in the title if '#' tag if found in "text_to_display" (idem for '^' tag for time display)
NETWORK	file_name.ase	o=disable, n=enable	
IMAGE	file_name.bmp	o=disable, n=enable + size of image	n=100 gives size given in IMAGE TDV command

			(if, in the IMAGE TDV command the size_factor is 1.0 then, with n=100, the size is the original)
D or Dx (designator)	text_to_display	o=disable, n=enable	
V or Vx (volume)	file_name.are (means all sectors found in that file) or file_name.are:sector_name (means only that sector found in that file)	o=disable, n=enable + color change	linear interpolation between TRGB color components found in V or Vx object color and found in CURVE color. n=100 means full V or Vx color is displayed n=0 means full CURVE color is displayed Note: if RGB components are the same but only T (=transparency) is changed from 0 to 255, the object will appear/disappear on the screen.
SANIM (traffic animation)	file_name.so6	o=disable, n=enable (threshold at 0.5)	Allows to toggle traffic animation on/off
CURVE	title_of_curve	depends on object to which CURVE is linked (see above and below this row)	Several CURVE objects (number depends on memory size) having the same title_of_curve name can be linked into a single CURVE command. The value found in the 2 TIM files will be multiplied, and the result will be applied to animated objects.
YMOVE (camera)	YMOVE	n is used to drive the camera (move your scene as you wish, press F12 to add current camera values in the TDV, open the TDV and copy YMOVE value)	goes in south/north direction (movement absorber is included)
XMOVE (camera)	XMOVE	n is used to drive the camera (move your scene as you wish, press F12 to add current camera values in the TDV, open the TDV and copy XMOVE value)	goes in west/east direction (movement absorber is included)
YORIENT (camera)	YORIENT	n is used to drive the camera (tilt your scene as you wish, press F12 to add current camera values in the TDV, open the TDV and copy YORIENT value)	change tilt angle (movement absorber is included)

		TDV, open the TDV and copy YORIENT value)	
XORIENT (camera)	XORIENT	n is used to drive the camera (rotate your scene as you wish, press F12 to add current camera values in the TDV, open the TDV and copy XORIENT value)	change azimuth angle (movement absorber is included)
ZOOM (camera)	ZOOM	n is used to drive the camera (zoom as you wish, press F12 to add current camera values in the TDV, open the TDV and copy ZOOM value)	change zoom (movement absorber is included)

9.7.81 tdm - Airspace Statistics

Airspace Statistics	
extensi on	tdm
origin	
separat or	tab (not space)
sort	no sort
comme nt	The file can be displayed in excel. The number of column is unlimited. Only the first one is mandatory.

First line of the file				
#	Field	Type	Size	Comment
1	Airspace header	text		
x	Column x header	text		Can contain whitespaces.

#	Field	Type	Size	Comment
1	Airspace ID	text		
x	value	float		A value associated to the sector

Example 1: Airspace Load

Sector	Number of flights	average distance (NM)	average time (min)
EDYYJEHI	316	115.3	15.6
EDYYJELO	81	70.6	9.6

EDYYKOH	120	35.9	5.5
EDYYKOL	39	22.2	3.6
EDYYLNH	259	61.3	8.6
EDYYLNL	56	38.4	5.5
EDYYMNT	159	82.9	11.4
EDYYNIL	513	57.1	7.8
EDYY_DELN		582	60.3
EDYY_DELS		1006	41.4
EDYY_RHRN		81	57.3
EDYY_RHRS		98	55.1

Example 2: Traffic density

Cell	Density
62_7_3	1
62_7_4	0
62_7_5	0
63_7_0	0
63_7_1	0
63_7_2	0
63_7_3	33
63_7_4	4
63_7_5	0
64_7_0	0
64_7_1	0
64_7_2	0
64_7_3	2
64_7_4	0
64_7_5	0
65_7_0	0
65_7_1	0
65_7_2	0
65_7_3	3
65_7_4	1
65_7_5	0
66_7_0	0
66_7_1	0
66_7_2	0
66_7_3	17
66_7_4	7

9.7.82 ter - Terrain Data file

Terrain data	
extension	ter
origin	SAAM terrain extraction module (terrain.exe)
separator	blank
sort	no, but order is important
comment	ascii file describing shapes used as an input in SAAM for 3D terrain display

#	Field	Type	Size	Comment
1	code	char		B', 'BT', 'N', 'C', 'V' or 'E'
2	parameters	list ...		depends on the code, can be empty

Code explanation with associated parameters:				
Code	Meaning	Parameters	Example	
B	Begin triangle strip		B	
BT	Begin independent triangle		BT	
N	Normal Vector	X,Y,Z normalized components	N 0.494063 0.420676 -0.760877	
C	Color	Red, green, blue component expressed between 0 and 255	C 180 255 180	
V	Vertex	lat,lon,alt expressed in minute decimal/meters	V 3355.2 -375 4.85564	
E	End		E	

Example:

```

B
N 0.174971 0 -0.984574
C 180 255 180
V 3349.8 -394.8 0
V 3355.2 -394.8 0
V 3349.8 -390 0.853018
N 0 -0.156032 -0.987752
V 3355.2 -390 0
N 0.107397 -0.155129 -0.982039
V 3349.8 -385.2 1.37795
N 0.046748 -0.207769 -0.97706
V 3355.2 -385.2 0.229659
N -0.106363 -0.206816 -0.972581
V 3349.8 -379.8 0.787402
N 0 -0.102739 -0.994708
V 3355.2 -379.8 0.229659
N 0.218922 -0.100247 -0.970579
V 3349.8 -375 1.87008
N 0.644724 0.369865 -0.668977
V 3355.2 -375 4.85564
N 0.494063 0.420676 -0.760877
E
B

```

9.7.83 tim - Timeline file

Timeline file	
extensio n	tim
origin	SAAM time file for animation
separato r	tab → (space is also OK but then you <u>cannot</u> have a space in the column header row)
sort	rows must be sorted ascending by time (column 1) for one day or a part of a day
colons	First line = header which contains free comment or TOKEN (see below) The number of columns is variable: first column is for the time, and then 1 column for each type of animation. In addition there can be a special column to indicate the number of clock turns. If present, this column must be column 2.

The first row is a header row. It contains the column label. The label is not used by SAAM application (but necessary for the designer for commenting the content of the column) except for TITLE object, and number of clock turn that are all enabled via tokens stored in the header row. It is very important to do not put blank or tab within a header label, better use "_" or "-" to separate words.

Each TIM file is used in one or several CURVE object(s), [see CURVE format](#).

List of recognised tokens in the header row of TIM file:

Number of clock turns:

- the token is NB_CLOCK_TURN, it must located imperatively in column 2, just after the time column. Useful for long animation, to start/end animations at Nth clock turn. Note: position of column referred in CURVE object are not changed by the presence or absence of NB_CLOCK_TURN column.

TITLE object (see [TITLE parameters](#)):

- DISPLAY to toggle on/off the display of the object (value is 0 or 1);
- LAT, LON, ALT to change respectively latitude, longitude and altitude (values are expressed in minutes decimals and in Flight Level, reference point is in the center of the Title);
- ROTX, ROTY, ROTZ will change angle of rotation for the different axes (values are expressed in degrees decimal, can be over 360.0 and negatives, note: order of rotation is important). ROTX rotates around horizontal x-axe ---, ROTY rotates around vertical y-axe | and ROTZ rotates around center z-axe o
- SIZE to change the overall size of the object (size 1.0 is extremely very small, visible size is about 1000.0);
- COLOUR to change the colour of the object. A value in the column should be between 0 and 100 to provide linear colour interpolation between the colour given to the Title object itself and the colour given to the CURVE object handling that animation, allowing to change one or several colour components: transparency, red, green and/or blue.

Notes:

- the values present in the column in the TIM file to change object properties are always linearly interpolated. If animation designer wants a sudden change than the time between the two states should be reduced to a minimum (0 or 1 second for example).
- If several animated events occurs at different times but are overlapping in the same TIM

file, a convenient way is to use the special character "~" instead of manually calculating interpolated numeric value: it will be replaced by a linearly interpolated value processed internally (see example below)

Example (time is expressed in seconds):

time	CIV_SIZE	MIL_SIZE
0	0	3000
1000	0	3000
2000	~	3000
2500	4000	~
4000	4000	0

In the example above a title object will grow linearly from size 0 at 13h to size 4000 at 20h. In the same time, a different title object will start to be reduced linearly from size 3000 at 14h to size 0 at 21h.

Data row (row 2 to ...)				
#	Field	Type	Size	Comment
1	Time	int/ float		time in seconds (max is 86400) or HH or HHMM or HHMMSS or HHMMSS.decimal (see CURVE parameter to setup time format)
2	NB_CLOCK_TURN	int		this field is optional
2 or 3	value n for anim#1	int		
3 or 4	value n for anim#2	int		
....				
i or i-1	value n for anim#i			

Example 1 (time is expressed in HHMMSS)

time	pinko_area	my_designator
000000	0	0
133000	0	0
140000	100	0
143000	100	0
150000	100	1
235959	100	1

Example 2: with a NB_CLOCK_TURN column token, 2 camera movements (free text), and one TITLE object token (SIZE)

time	NB_CLOCK_TURN	Cam_ZOOM	Cam_NS_move	SIZE
------	---------------	----------	-------------	------

020000	2	1246149	-27759	0
030000	2	1246149	-27759	30000
060000	2	300000	2258	10000
140000	2	300000	2258	30000

9.7.84 tol - Bunching Capacity (Tolerance) file

Bunching capacity (tolerance)				
extension	tol			
origin				
separator	blank			
sort	no sort			
comment	each line contains bunching capacity figures for a sector			
#	Field	Type	Size	Comment
1	sector name	char	26	the same one as the one found in field 1 of sls
2	bunching capacity figure	int		special figure 999 means infinite capacity

Note 1: blank lines are allowed to better separate sectors from each others.

Example:

```
LFEPPLOW 3
EDDYLNO 5
```

9.7.85 traj - Trajectory file

Trajectory				
extensio	traj			
n				
origin	produced by SAAM sector load (if traj check box is on), processed from t5 file and other flight information			
separato	blank (except for field #1 which is composed with multiple information)			
r				
sort	On aircraft ID, then on route point order from origin to destination			
commen	each line contains complete flight plan trajectories with sector entry and exit			
t	information			

#	Field	Type	Size	Comment
1	aircraft ID	num		complete flight ID (format is: flightID aircrafttype callsign:origin destination)
2	date/time	float		unit is s decimal, start 01 January 1970
3	flag	num	2	"-1=first flight plan point (departure) 0= sector exit point 1= normal flight plan point 2= sector entry point 3=last flight plan point (arrival)"
4	point or entry/exit sector name	char		if entry = in-sectorname, if exit = out-sectorname else route point name

5	latitude	float	unit is minute decimal
6	longitude	float	unit is minute decimal
7	level	float	unit is FL decimal

Example (1 flight):

```

17829_BA46_DLH5291:EPWA_EDDM 936974400 1 EPWA 3129 1258 0.00
17829_BA46_DLH5291:EPWA_EDDM 936974499 1 LIN2 3123 1263 53.00
17829_BA46_DLH5291:EPWA_EDDM 936974892 1 KRN2 3114 1213 189.00
17829_BA46_DLH5291:EPWA_EDDM 936975125 1 LDZ 3108 1179 255.00
17829_BA46_DLH5291:EPWA_EDDM 936975354 1 $aedG 3101.75 1143.25 300.00

17829_BA46_DLH5291:EPWA_EDDM 936976778.226916 2 in-CEATS_285_UNL
3057.751294 919.270695 300.000000
17829_BA46_DLH5291:EPWA_EDDM 936976890 1 OKX 3054 902 300.00
17829_BA46_DLH5291:EPWA_EDDM 936977565 1 RAK 3006 822 310.00
17829_BA46_DLH5291:EPWA_EDDM 936977876 1 %LK2 2980 792 310.00
17829_BA46_DLH5291:EPWA_EDDM 936977996.737125 0 out-CEATS_285_UNL
2969.950521 780.576100 304.313106
17829_BA46_DLH5291:EPWA_EDDM 936978152 1 KALOD 2957 766 297.00
17829_BA46_DLH5291:EPWA_EDDM 936978365 1 %ED43 2938 752 230.00
17829_BA46_DLH5291:EPWA_EDDM 936978661 1 MBG 2914 735 140.00
17829_BA46_DLH5291:EPWA_EDDM 936979132 1 EDDM 2887 701 0.00

```

9.7.86 txt - Corrections**Corrections File**

Filename: "MyCorrections.txt"

Function: Defines a list of entity ID aliases to be used to automatically replace IDs in the copy scenario process

The following is a description of the format used to store AIRAC compatibility corrections in the “Copy Scenario” functionality.

Field Separator: Tab Character

Format of lines describing entity aliases

Field	Value
1	Entity type : “ACC”, “Sector”, “TV”, “Nav Point”, “Traffic Flow”
2	Entity ID : identifier in target AIRAC
3	Alias : identifier to be used instead of Entity ID

Format of lines describing configuration aliases

Field	Value
1	Entity type : “Config”
2	ACC ID : ACC identifier in target AIRAC
3	Config ID : Config identifier in target AIRAC
4	Config Alias : Config identifier to be used instead of Config ID

Format of lines describing date compatibility mode

Field	Value
1	Correction Key : “dateMatching”
2	Matching mode : “byDate”, “byIndex”

Example:

```
ACC DTTCACC      DTTCCTA
ACC EBBUACC      EBBUCTA
Config      EBBUCTA      CONF4 CFE4W3
dateMatching    byDate
```

9.7.87 ur - Unit rates

Unit rates	
extensi on	ur
origin	Used in the route charge module.
separat	blank

Unit rates				
#	Field	Type	Size	Comment
1	Sector name/ country	char		
2	start date	date		yyyy/MM/dd. Usually ignored in SAAM.
3	end date	date		yyyy/MM/dd. Usually ignored in SAAM.
4	unit rate	num		Charge in Eurocents

Example:

AZ 2013/08/01	2013/08/31	896
EB 2013/08/01	2013/08/31	6799
ED 2013/08/01	2013/08/31	7665
EF 2013/08/01	2013/08/31	4979
EG 2013/08/01	2013/08/31	8399
EH 2013/08/01	2013/08/31	6553
EI 2013/08/01	2013/08/31	2835
EK 2013/08/01	2013/08/31	7358
EN 2013/08/01	2013/08/31	5473
EP 2013/08/01	2013/08/31	3534
ES 2013/08/01	2013/08/31	7673

9.8 Allowed ID characters

The table below describes the ID character restrictions applied to entity ID fields in all data edition and entity search contexts.

Entity Type	Allowed Characters	Min Length	Max Length
Navigation Point	A-Z 0-9 "\$%*."	2	5
Network Segment	A-Z 0-9 "\$%_*."	5	50
Custom Traffic Flow	A-Z 0-9 "_"	1	50
Traffic Volume	A-Z 0-9 "_&#?/-+\\"\\>@.*"	1	50
Airport	A-Z 0-9	4	4
Set of Airports	A-Z 0-9 "_&#?/-+\\"\\>@.*"	1	50
Set of Airports (FIR RAD)	A-Z 0-9 "_"	1	50
Network Zone	A-Z 0-9 "_"	1	50
Network	A-Z 0-9 "_"	1	50
Airblock	A-Z 0-9 "_+%-"	1	50
Airport Family	A-Z 0-9 "/&_-:_()."	1	50
Configuration	A-Z 0-9 "_.-"	1	50
All Entities (default)	A-Z 0-9 "_"	1	50

Notes:

- A-Z: all 26 upper-case letters
- 0-9: all 10 digits
- "_&#?/-+\\"\\>@.*": allowed character set. This example includes the SPACE " " character

9.9 Acknowledgements

Project Sponsors and Coordinators:

European Capacity Planning Taskforce,

Route Network Development Sub-Group (RNDSG)

Mr Razvan Bucuroiu

Head of Operations Planning

EUROCONTROL, Network Operations Management/Operations Planning

+32 2 729 3648

Razvan.BUCUROIU@eurocontrol.int

Mrs Stephanie Vincent

Capacity Planning Expert

EUROCONTROL, Network Operations Management/Operations Planning/Ops Performance Plan

+32 2 729 3407

stephanie.vincent@eurocontrol.int

Mr Thierry Champougny

Expert - Airspace Modelling Tools

EUROCONTROL, Network Operations Management/Operations Planning/Ops Performance Plan

Tel: +32 2 729 33 93

Thierry.champougny@eurocontrol.int

Mr Vincent Martin

Technical Project Manager of NEST Development

EUROCONTROL, Network Operations Management/Operations Planning/Ops Performance Plan

+32 2 729 3174

Vincent.martin@eurocontrol.int

NEST Development & Support Team:

- Nicolas Boulin
- David Deschaume
- Adrian Dixon
- Annika Lindberg
- Vincent Martin
- Thibaud Rabillard

NEST Contributors:

Roland Jalass, Leila Zerrouki, Marc Dalichampt, Francoise Guenin-Rambaud, Claire Leleu, Gillian Heath, Serge Manchon, Bernard Kerstenne, Elizabeth Petit, Denis Odic, Sauro Fiorani, Roman Kushnarov, Robert Falk and many others who have provided feedback to improve the tool.

Part



X

10 Frequently Asked Questions

The following is a list of recurrent business-related questions relating to the use of NEST. Please refer to the [troubleshooting](#) section for technical questions. If you do not find what you are looking for here, please don't hesitate to contact the [NEST Support Team](#).

Frequently Asked Questions

General

1. [How do I register for a NEST training course?](#)
2. [Is there a version of NEST for Mac or Linux?](#)

NEST AIRAC Data

1. [What is the difference between Initial and Actual trajectories?](#)
2. [Why do some flights have a very large flight ID compared to other flights?](#)
3. [The DDR2 provides EXP2 files, which RFL value is used for these files?](#)
4. [How is flight plan, regulation and delay data impacted when a company re-files a flight plan to avoid delay before departure?](#)
5. [Why does some network point have a *, % or a point in there name?](#)

Map features

1. [What length measure is used? Is the 3D length available?](#)

Traffic counts

1. [Why do I get different flight counts throughout the application? e.g.: more flights in the traffic browser compared to the daily entry count view?](#)
2. [What is the difference between Entries and Occupancies?](#)
3. [Why are NEST flight counts different from those displayed in CHMI?](#)
4. [Why do I see different overloads in overload charts when switching between sliding and non-sliding mode?](#)

Airspace Editing

1. [Is it possible to subtract airspaces in the airspace editor?](#)
2. [How do I create a new Free Route Airspace on my ACC?](#)

Trajectory Simulation

1. [Why don't I get any simulated flights on the new route I've created?](#)
2. [Why do I get longer simulated routes when i add new routing possibilities or shorter simulated routes when I remove existing ones?](#)
3. [Why aren't the simulated trajectories the same as the Initial trajectories in official AIRAC data, on an original scenario?](#)
4. [What RFL is used for trajectory simulations?](#)
5. [Why is the simulated trajectory not respecting a route restriction?](#)
6. [How do I run a cheapest trajectory simulation?](#)

Statistics

1. [How do I launch statistics over an entire year?](#)

ICO

1. [How does the sliding and non-sliding overload chart mode impact on the ICO algorithm?](#)
2. [What is the difference between version 2 and version 3 of ICO?](#)

Scenario Economy

1. [What is the meaning of the message about "disregarded flights" in the report?](#)

General

How do I register for a NEST training course?

NEST training sessions are advertised on the NEST news. To be directly informed of new NEST training sessions, you can send us an e-mail at

nest@eurocontrol.int, you will then receive an e-mail when new training sessions are organized. NEST trainings are organized directly by the NEST team so you have to contact us directly if you are interested.

[Back](#)

Is there a version of NEST for Mac or Linux?

NEST is only available on windows. It might be possible to install Windows on another operating system via a virtualization program such as "Virtual Box", "Vmware Fusion" or "Parallel Desktop". However, the NEST team doesn't provided any support for this kind of operation.

[Back](#)

NEST AIRAC Data

What is the difference between Initial and Actual trajectories?

Please refer to the trajectory definitions topic for more information.

[Back](#)

Why do some flights have a very large flight IDs compared to other flights?

High flight IDs depict flights that were deleted, they do not appear in PRISME data so a "normal" flight ID does not exist. It is likely that they do not appear in CHMI either as it is likely that these flights were never actually flown. These flights are maintained in AIRAC input data in order to maintain consistency with the regulation data.

[Back](#)

The DDR2 provides EXP2 files, which RFL value is used for these files?

The EXP2 file available on the DDR2 website is data that is extracted from an ALL_FT+ file containing the initial, regulated and actual trajectories. The ALL-FT + to EXP2 conversion applies the same principle as the NEST data manager, it takes the maximum RFL in the initial trajectory and stores this as a single RFL value per flight in the EXP2 file.

[Back](#)

How is flight plan, regulation and delay data impacted when a company re-files a flight plan to avoid delay before departure?

The FTFM trajectory in NEST corresponds to the last flight plan filed by the airline. Each new flight plan filed by an airline replaces the previous one. All related data such as the most penalizing regulation and delays are updated continuously.

Example

In the pre-tactical phase, a flight has a flight plan with ETOT = 14:00

At 10:00, NM attributes a ground delay to it because it is impacted by a regulated TV

At 10:10 the airline decides to change the flight plan and sends the new flight plan to NM, this becomes the new FTFM

At 10:20 NM removes the flight delay because the new route avoids the regulated TV

The flight departs at 14:00 = ETOT = CTOT, with its most penalizing regulation empty

[Back](#)

Why do some network point have a *, % or a dot in their name?

Point names that are duplicates, i.e. two or more network point with the same name, can be renamed with either * or % in front of the point name, to make sure the point name is unique.

To some point names, a ".A" or ".D" has been added after the point name. This is for trajectory simulation reasons to differentiate between when the point is used for a SID or a STAR.

There are other symbols used in the trajectory route names. These are explained in [so6 - Traffic file](#).

[Back](#)

Map features

What length measure is used, is the 3D length available?

All lengths in NEST are given with respect to the ground. They are all ground length. 3D length is not available.

[Back](#)

Traffic counts

Why do I get different flight counts throughout the application? e.g. : more flights in the traffic browser compared to the daily entry count view?

Flight counts are a product of filtering, both by airspace and by time period. Whenever the airspace filter or the time period filter are different, the flight counts are likely to be different.

The starting point for any flight count is the daily flight list, it contains all flights crossing the [Extended IFPZ Area](#) for at least one second on the given day. It's important to note that the daily flight list is not filtered based on entry time, so it contains flights entering the area on the previous day.

The daily traffic count value displayed in the entry count view is the result of filtering the daily flight list based on entry times into the selected airspace. Only flights entering the airspace from 00:00 to 24:00 are counted.

The traffic browser allows filtering by airspace only, or by both airspace and time period in parallel as in the entry count view. When using the [Custom Flow Filter](#) feature with the time period unchecked or the [Textual Query](#) feature, the filtering is by airspace only, so counts will be different compared to an entry count view. When using the Custom Flow Filter feature with the time period checked, the filtering is by both airspace and time period, so the counts will be the same as an entry count view.

[Back](#)

What is the difference between Entries and Occupancies?

[Entry counts](#) measure the number of airspace entry events that occur during a given time period. So for two consecutive non-overlapping periods, a single flight entering a sector during the first period will be counted in this period but then ignored in the second period, even though it may still be present in the sector throughout the second period.

[Occupancy counts](#) measure the number of flights present for at least one second during a given time period, irrespective of when entry events occur. So for two consecutive non-overlapping periods, a single flight entering a sector during the first period will be counted in both the first and second periods because it is present in the sector for at least a second during both periods.

Additional sector-specific [skip-in](#) and [skip-out](#) values are employed to ignore airspace entries and exits of short duration. Please refer to the [demand counting conventions](#) topic for more detailed information.

[Back](#)

Why are NEST flight counts different from those displayed in CHMI?

Firstly, a single [skip-in](#) value of 30 seconds was used for all airspaces in NEST versions up to and including v1.3.0. The [CHMI](#) tool has always used [skip-in](#) and [skip-out](#) values that vary from airspace to airspace. Using different skip-in and skip-out values can lead to different entry/exit events being ignored, and so different demand counts. As from AIRAC 1409 however, the specific airspace skip-in and skip-out values used in the CHMI tool have been added to the official AIRAC data files, and from NEST v1.3.1 onwards, these values are used in all calculations allowing this source of discrepancy to be entirely removed.

Secondly, due to a purely technical limitation in CHMI, all skip-out values are capped below 15 minutes leading to a separate source of discrepancy between CHMI and NEST counts. This capping behaviour is reproduced in NEST using the [Apply 15-minute minimum skip-out value](#) option in the scenario monitoring parameters, which is checked by default.

Using official skip-in and skip-out values (i.e. those provided by ANSPs, used in CHMI and published in post AIRAC 1409 NEST data files and used in NEST v1.3.1 onwards), and also the "Apply 15-minute minimum skip-out value" option allows NEST and CHMI demand counts for airspaces to be 100% aligned.

A final source of small discrepancies persist however for traffic volume counts. This is due to the fact that unlike NEST, CHMI does not use skip-in and skip-out

values in the evaluation of whether a flight is captured by a traffic volume flow. The NEST development team has been advised that this is considered a flaw in the CHMI design leading to some traffic volume flows to be incorrectly evaluated, so this behaviour has not been emulated in NEST resulting in a small number of traffic volume count discrepancies persisting between NEST and CHMI.

[Back](#)

Why do i see different overloads in overload charts when switching between sliding and non-sliding mode?

Sliding observation windows capture more peaks.

In non-sliding mode, successive overload observation windows do not overlap, so observed overloads are captured only once. In this mode, the daily overload value is simply the sum of all overloads detected in the discreet periods throughout the day. In this mode, demand peaks causing overloads occurring in between observation windows are skipped and therefore do not contribute to the daily overload sum value.

In sliding mode, successive overload observation windows are separated by a small [sliding step](#), thereby exploring a much larger proportion of the day and greatly reducing the possibility of skipping demand peaks. In this mode, as the overload observation windows overlap, demand peaks can be captured multiple times in successive overload observation windows. The daily overload sum is less trivial in sliding mode; overloads are summed using a technique that aims to eliminate double-counting and provide a normalised daily value, independent of the chosen integration window. A detailed explanation of the sliding overload sum technique and of how sector transitions are managed is provided in the [overload methodology](#) topic.

[Back](#)

Airspace Editing

Is it possible to subtract airspaces in the airspace editor?

Please refer to the section on airspace subtraction in the [airspace editor](#) topic.