CAASD

Working Notes on ASSAP Design 12th Set

4 April 2006

Roxaneh Chamlou

Outline

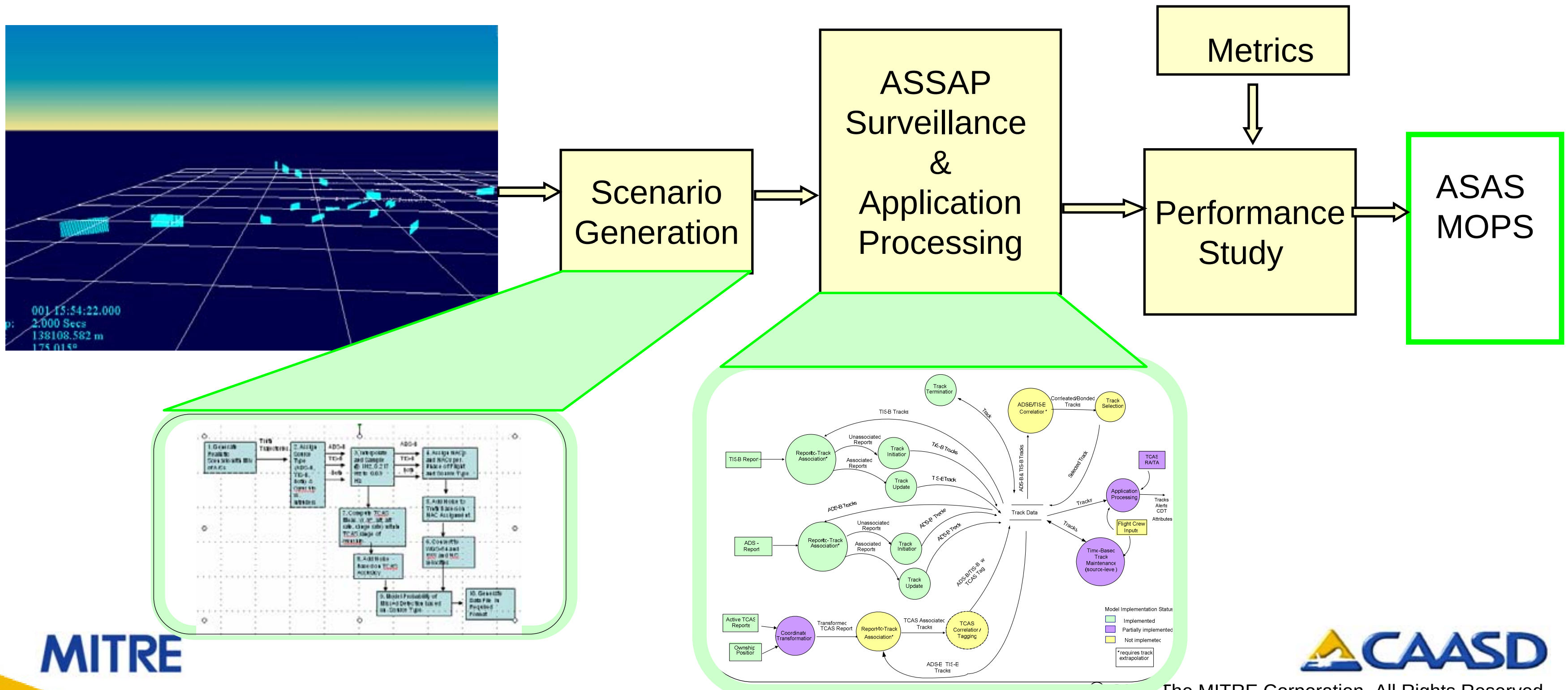
- High Level ASSAP Requirements
- Approach to MOPS Development
- Block Diagrams Depicting
  - Track Association/Update for each Track Type
  - TCAS to ADS-B or TIS-B Correlation/Tagging
  - ADS-B to TIS-B Track-to-Track Correlator
- Overview of Sequence of Operations Event Driven by ADS-B/TIS-B Reports
- Overview of Proposed Filtering for Track Update
  - Degenerate Kalman Filter
- Coordinate Frame for Prediction
  - Transformations from WGS-84 to ownship ENU
  - Transformation from ownship ENU to WGS-84
  - Transformation from range, bearing, altitude to ENU
- Report-to-Track Association Logic for Spatial Association/Consistency Check
  - Defining the Candidate Window
  - Defining the Association Window
  - (removed) Metric for Association - Statistical Distance
  - Report-to-Track Resolution Logic
  - Track Merge Logic for Correlated Track with Different IDs
  - Track Termination (for Least Demanding Application)
- Background-Track-Correlation Logic
- Track-to-Track (ADS-B to TIS-B) Correlation/Bonding Logic
- Best of Track Selection
- TCAS Tagging Logic

High Level ASSAP Requirements

- **Surveillance Processing:**
  - **Establish Tracks from ADS-B and TIS-B traffic reports.**
  - **Correlate traffic from different surveillance sources (ADS-B, TIS-B, TCAS).**
  - **Select best source between ADS-B and TIS-B for application processing.**
  - **Indicate when a track is supported by TCAS (TCAS-tag to be used by CDTI).**
  - **Delete tracks that are beyond the maximum allowable coast time for any ASA application.**
- **Application Processing**
  - **Determine the appropriateness of track information for various applications (i.e., normal, degraded, not used).**
  - **Perform alerting functions (e.g., CD).**
  - **Maintain the interface to/from the CDTI Display and Control Panel subsystem.**

Approach to MOPS Specifications

- **Build simulation of ASSAP. Architectural approach:**
  - **Implement Distributed Tracking, Best-Of Track Selection, TCAS Tagging.**
- **Define and develop realistic scenarios.**
  - **Start with TASF scenarios as truth trajectories, simulate ADS-B, TIS-B, TCAS**
- **Use simulation results to develop MOPS level requirements.**
  - **Define metrics, Conduct Performance Study, Develop Requirements.**

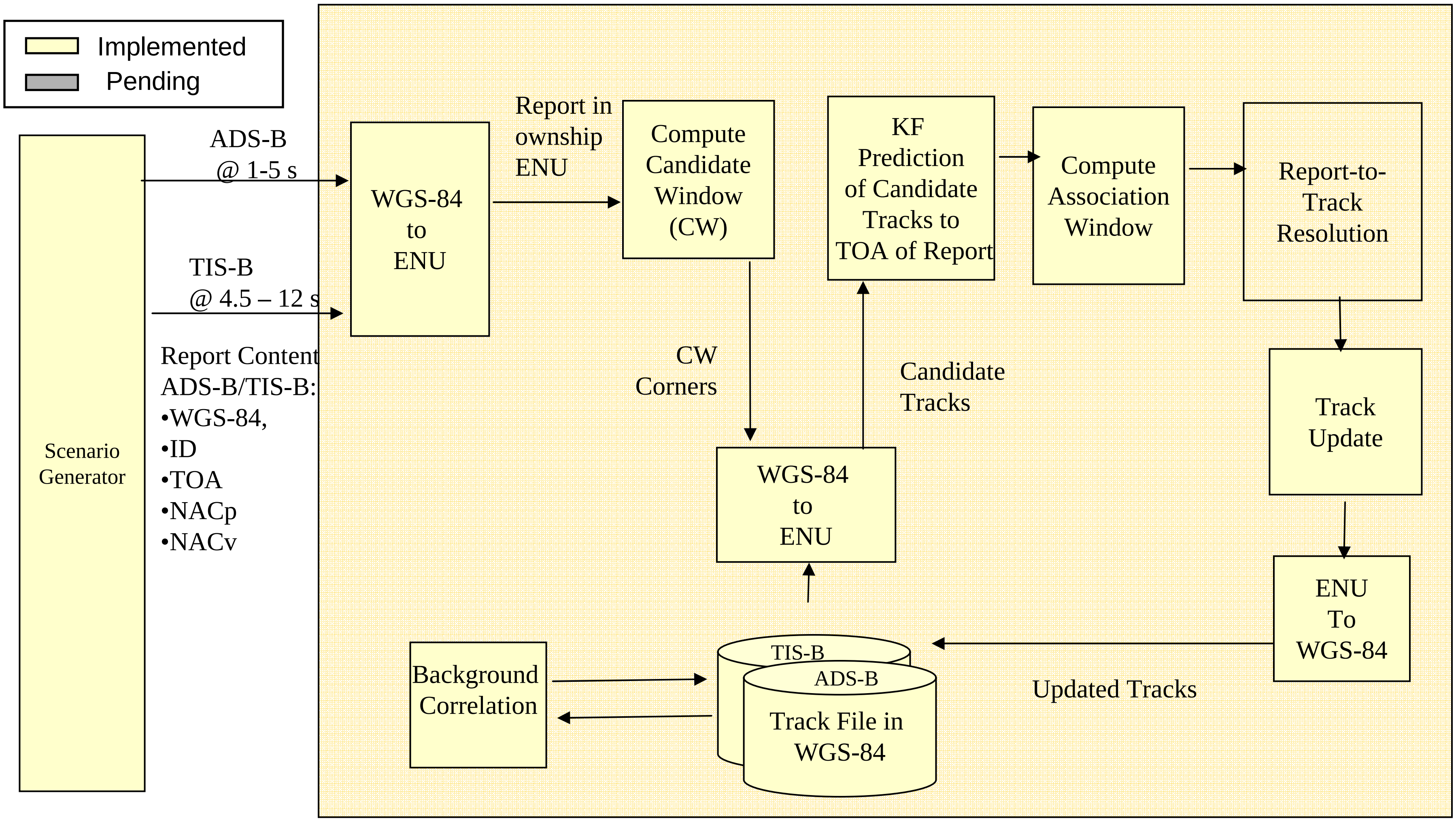


MITRE

CAASD

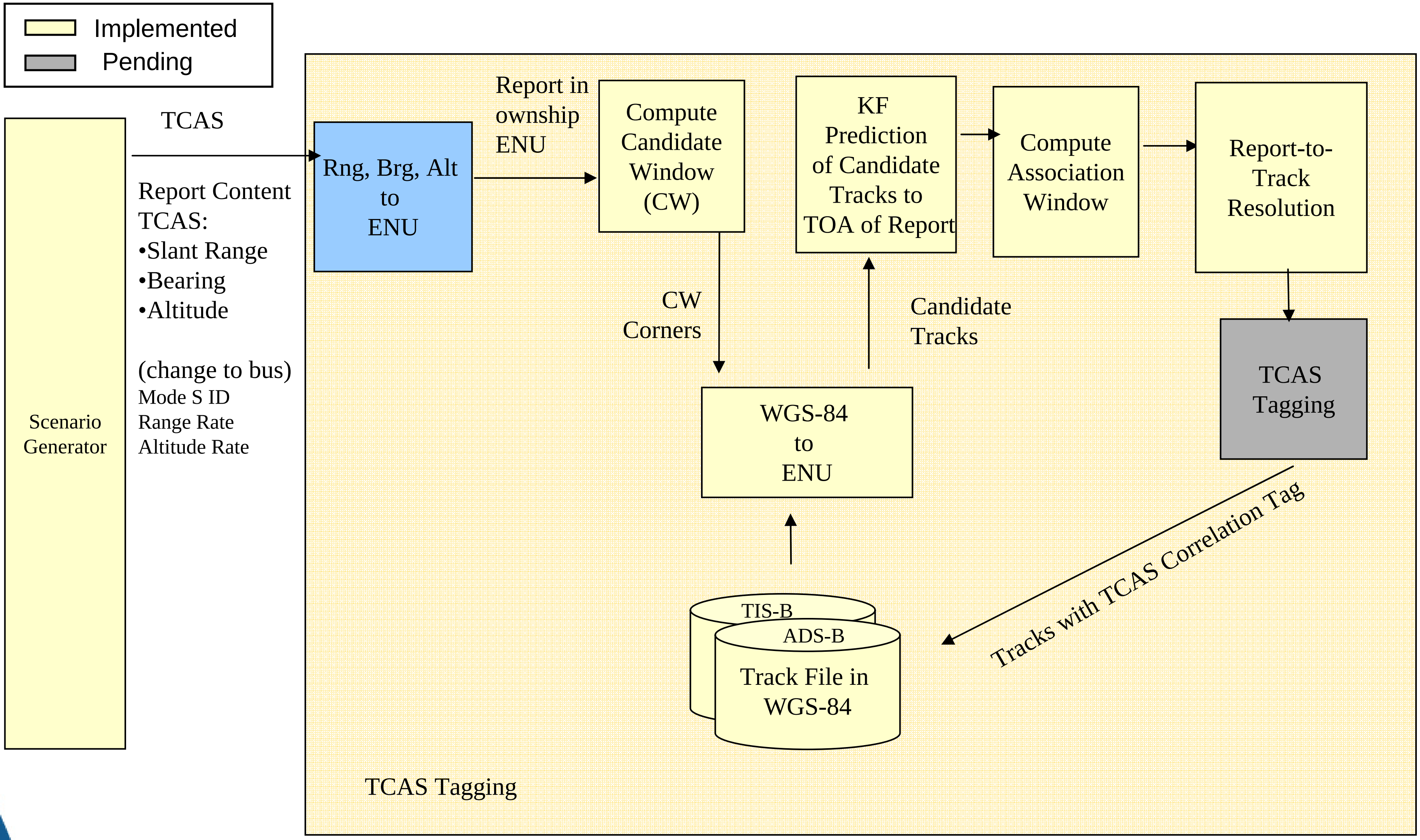
F084-B06-009

Process of Track Association/Update for each Track Type



Track Association/Resolution/Update

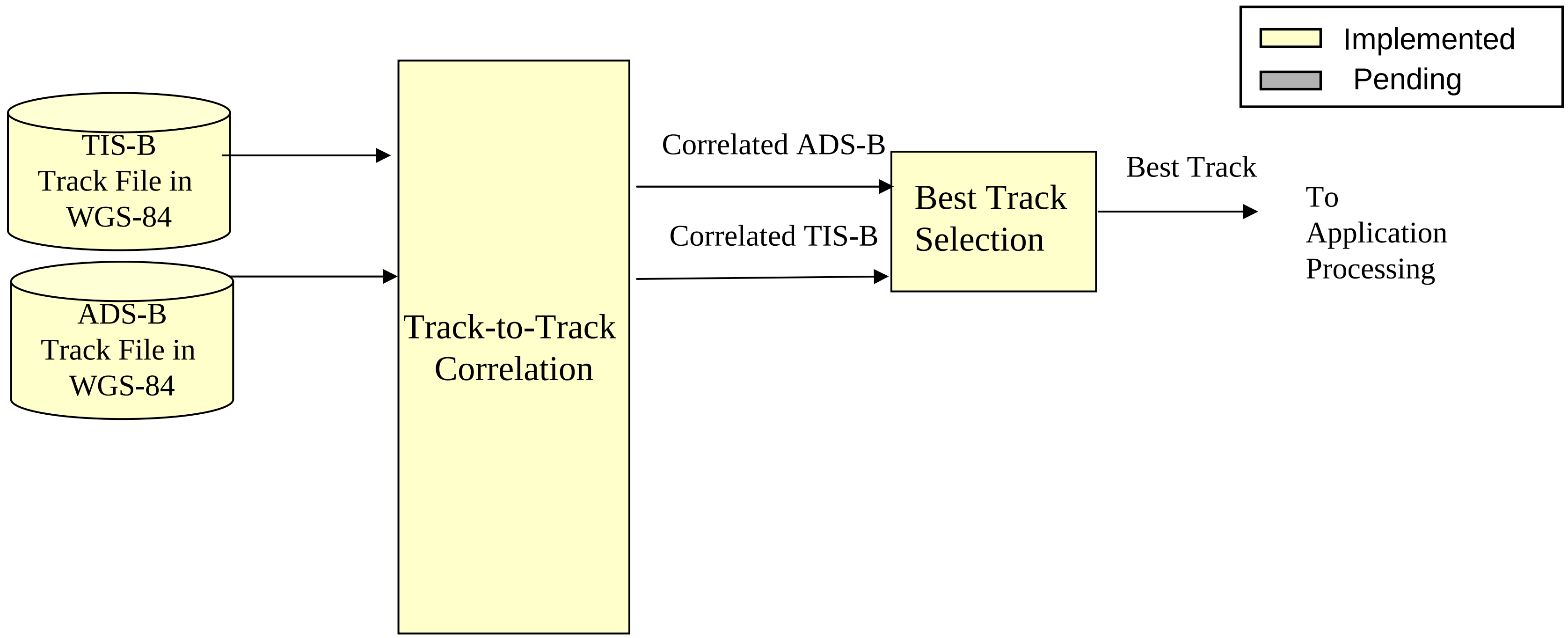
Process of TCAS to ADS-B or TIS-B Correlation/Tagging



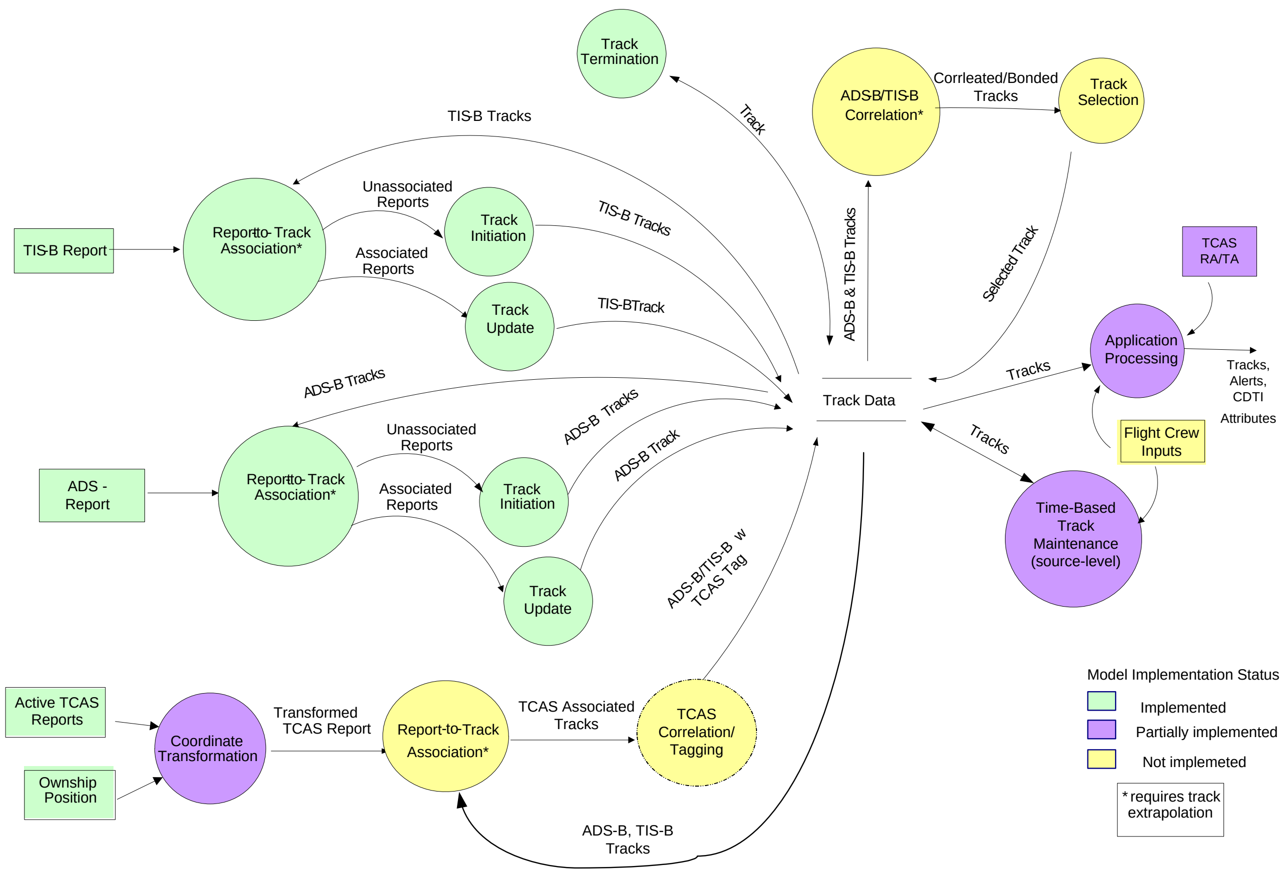
MITRE

CAASD

ADS-B to TIS-B Track-to-Track Correlator



Functional Diagram

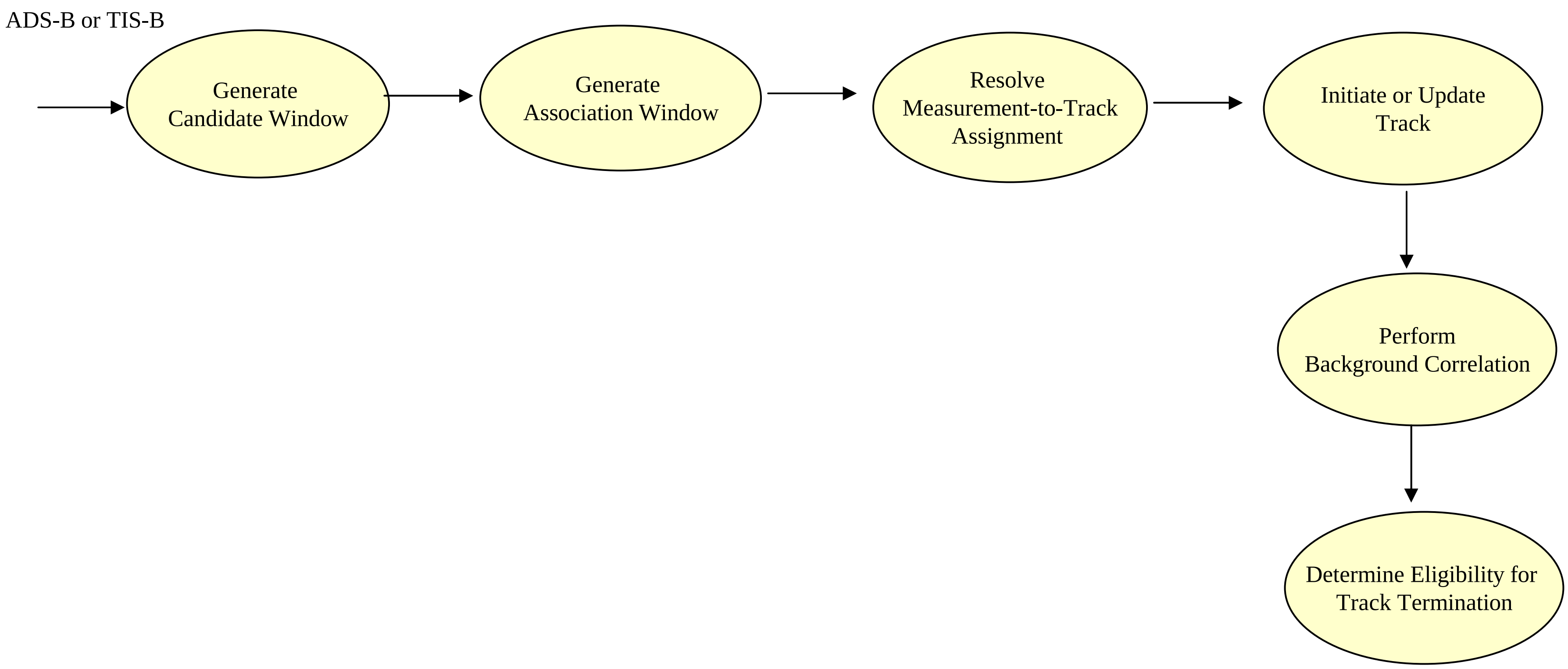


MITRE

CAASD

F084-B06-009

Sequence of Operations Event Driven by ADS-B or TIS-B Reports



Overview of Proposed Filtering for Track Update

- Use a degenerate Kalman Filter that
  - Predicts state (position, velocity) and state uncertainty (NACp, NACv) to current time (i.e., between ADS-B/TIS-B updates) using standard Kalman Filter equations
  - Updates the state and state uncertainty differently than the standard Kalman Filter which "blends" the predicted and measured parameters using the Kalman gain
    - State update = received report state
    - State uncertainty = received report uncertainty
      - The containment radius, R_c, indexed by NACp is a 2-D 95% uncertainty bound. To derive the 1-σ radius for the covariance matrix multiply the R_c by a factor of 0.4085.

Overview of Proposed Filtering for Track Update (cont'd)

Kalman Filter Equations

State and Measurement Equations

$z_k = y_k - H_k \tilde{x}_k$ Innovation = Measurement - Prediction

$\hat{x}_k = \tilde{x}_k + K z_k$ State Estimate

$\tilde{x}_{k+1} = \Phi_k \hat{x}_k + f_{k+1}$ State Prediction

Kalman Gain and Error Covariance Equations

$K_k = \tilde{P}_k H_k^T \left[H_k \tilde{P}_k H_k^T + R_k \right]^{-1}$ Kalman Gain

$\hat{P}_k = \left[1 - K_k H_k \right] \tilde{P}_k$ Estimate Covariance

$\tilde{P}_{k+1} = \Phi_k \hat{P}_k \Phi_k^T + Q_k$ Prediction Covariance

Degenerate Kalman Filter Equations

$z_k = y_k - H_k \tilde{x}_k$ Innovation = Measurement - Prediction

$\hat{x}_k = \tilde{x}_k + K z_k$ State Estimate

$\tilde{x}_{k+1} = \Phi_k \hat{x}_k + f_{k+1}$ State Prediction

Kalman Gain not computed

Estimate Covariance not computed

$\tilde{P}_{k+1} = \Phi_k R_k \Phi_k^T + Q_k$ Prediction Covariance

where $R_{k_i} = \begin{bmatrix} \sigma^2{}_{NAC_P} & 0 \\ 0 & \sigma^2{}_{NAC_V} \end{bmatrix}$, i : ENU coordinates

for East and North coordinates, NAC is the circular NAC,

for the Up coordinate, NAC is the vertical NAC

$\Phi_{k_i} = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix}$

$Q_{k_i} = 0$

MITRE CAASD

Coordinate Transformation for Prediction of ADS-B and TIS-B: WGS-84 to ENU

- **Track files stored in WGS-84 Coordinates**
- **Track prediction performed in ownship Cartesian East-North-Up (ENU) coordinates**

1. Transform from WGS-84 to ECEF

$$\begin{bmatrix} X_e \\ Y_e \\ Z_e \end{bmatrix} = \begin{bmatrix} (R_n + h)\cos(\mu)\cos(L) \\ (R_n + h)\cos(\mu)\sin(L) \\ \left(R_n(1-\varepsilon^2) + h\right)\sin(\mu) \end{bmatrix}$$

where

$R_b = 6356752.3142(\mathit{meters})$ earth's semiminor axis

$h = \mathrm{W}GS - 84$ altitude of target

$L = \mathrm{WGS} - 84$ longitude of target

$\mu = \mathrm{WGS}$-84 latitude of target

$\varepsilon^2 = 0.00669437999013(\mathit{unitless})$ ecentricity squared

$$R_a = \sqrt{\frac{{R_b}^2}{1-\varepsilon^2}}$$

$$R_n = \frac{R_a}{\sqrt{1-\varepsilon^2 \sin^2 \mu}}$$

2.Transform from ECEF to ENU

$$\begin{bmatrix} X_E \\ Y_N \\ Z_U \end{bmatrix} = M^T \left(\begin{bmatrix} X_e \\ Y_e \\ Z_e \end{bmatrix} - T \right)$$

where

$$M = \begin{bmatrix} -\sin(L_{A/C}) & -\cos(L_{A/C})\sin(\mu_{A/C}) & \cos(L_{A/C})\cos(\mu_{A/C}) \\ \cos(L_{A/C}) & -\sin(L_{A/C})\sin(\mu_{A/C}) & \sin(L_{A/C})\cos(\mu_{A/C}) \\ 0 & \cos(\mu_{A/C}) & \sin(\mu_{A/C}) \end{bmatrix}$$

$$T = \begin{bmatrix} (R_{nR} + h_{A/C})\cos(\mu_{A/C})\cos(L_{A/C}) \\ (R_{nR} + h_{A/C})\cos(\mu_{A/C})\sin(L_{A/C}) \\ \left(R_{nR}(1+\varepsilon^2) + h_{A/C}\right)\sin(\mu_{A/C}) \end{bmatrix}$$

$h_R = \mathrm{WGS} - 84$ ownship altitude

$L_R = \mathrm{WGS}$-84 ownship longitude

$\mu_R = \mathrm{WGS}$-84 ownship latitude

$R_{nR} = \dfrac{R_a}{\sqrt{1-\varepsilon^2 \sin^2 \mu_R}}$ Prime vertical at ownship latitude

Note, M^T denotes the transpose of M

MITRE

CAASD

Coordinate Transformation for Prediction of TCAS: Relative Range, Relative Bearing, Altitude to ENU

TCAS Report: Relative Range (r), Relative Bearing (θ) and Baro Altitude (alt)

Ownship Position: ENU (derived from WGS-84)

Transformation from {r,θ, alt} to ENU:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_{ENU} = HPRBE \begin{bmatrix} 0 \\ r_s \\ 0 \end{bmatrix}$$

where

r_s = target range from sensor

H is the heading, *hdng*, rotation matrix

$$H = \begin{bmatrix} \cos(hdng) & \sin(hdng) & 0 \\ -\sin(hdng) & \cos(hdng) & 0) \\ 0 & 0 & 1 \end{bmatrix}$$

P is the ownship pitch angle, *p*, rotation matrix, positive noise down

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(p) & \sin(p) \\ 0 & -\sin(p) & \cos(p) \end{bmatrix}$$

R is the ownship roll angle rotation matrix, positive is left wing down

R = [1 0 0;0 1 0; 0 0 1];

B is bearing angle, *brg*, rotation matrix wrt sensor, positive is clockwise from nose

$$B = \begin{bmatrix} \cos(brg) & \sin(brg) & 0 \\ -\sin(brg) & \cos(brg) & 0) \\ 0 & 0 & 1 \end{bmatrix}$$

E is elevation angle, *el*, rotation matrix wrt sensor, positive is up

$el = rad2\deg(a\sin((ownship\_u - alt)/rng));$

$$E = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(el) & -\sin(el) \\ 0 & \sin(el) & \cos(el) \end{bmatrix}$$

Coordinate Frame for Storing into Track File
Transformation from ENU to WGS-84

- **Track updates, candidate and association window corners are computed in ENU and need to be translated to WGS-84 coordinates**

1. Transformation from ENU to ECEF

$$\begin{bmatrix} X_e \\ Y_e \\ Z_e \end{bmatrix} = \begin{bmatrix} (R_n + h)\cos(\mu_{own})\cos(L_{own}) \\ (R_n + h)\cos(\mu_{own})\sin(L_{own}) \\ \left(R_n(1-\varepsilon^2) + h\right)\sin(\mu_{own}) \end{bmatrix} + \begin{bmatrix} -\sin(L_{own}) & -\cos(L_{own})\sin(\mu_{own}) & \cos(L_{own})\cos(\mu_{own}) \\ \cos(L_{own}) & -\sin(L_{own})\sin(\mu_{own}) & \sin(L_{own})\cos(\mu_{own}) \\ 0 & \cos(\mu_{own}) & \sin(\mu_{own}) \end{bmatrix} \begin{bmatrix} E \\ N \\ U \end{bmatrix}$$

where

ENU are the target's east, north, up coordinates measured in Cartesian coordinates centered on ownship

$R_b = 6356752.3142(\textit{meters})$ earth's semiminor axis

$h_{own} = WGS-84$ altitude of ownship

$L_{own} = \text{WGS}-84$ longitude of ownship

$\mu_{own} = \text{WGS-84}$ latitude of ownship

$\varepsilon^2 = 0.00669437999013(\textit{unitless})$ ecentricity squared

$$R_a = \sqrt{\frac{R_b^{\,2}}{1-\varepsilon^2}}$$

$$R_n = \frac{R_a}{\sqrt{1-\varepsilon^2 \sin^2 \mu}}$$

2. Transformation from ECEF toWGS-84

$$\mu \approx \text{atan}\left(\frac{Z_e + \varepsilon^2 R_b \sin^3\theta}{P - \varepsilon^2 R_a \cos^3\theta}\right) \quad \text{WGS-84 latitude of target}$$

$$L = \text{atan}\,2(Y_e, X_e) \quad \text{WGS-84 longitude of target}$$

$$h = \frac{P}{\cos\mu} - R_n \quad \text{WGS}-84 \text{ altitude of target}$$

where

$$\varepsilon^2 = \frac{R_a^{\,2} - R_b^{\,2}}{R_b^{\,2}}$$

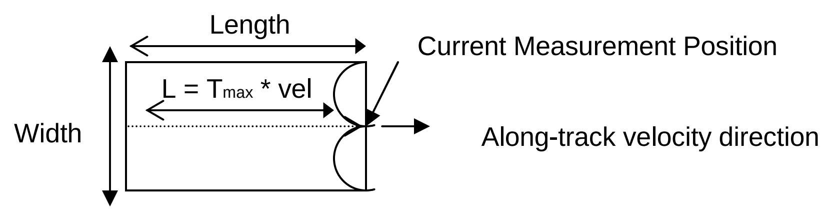
$$P = \sqrt{X_e^{\,2} + Y_e^{\,2}} \quad \text{(distance from the } Z_e \text{axis)}$$

$$\theta \approx \text{atan}\left(\frac{Z_e R_a}{P R_b}\right)$$

$$R_n = \frac{R_a}{\sqrt{1-\varepsilon^2 \sin^2 \mu}}$$

Report-to-Track Association – Candidate Window

- Convert new report (in WGS-84) to local ENU
- Compute a *Candidate Window* (CW) centered on the new report
  - The length of the CW is defined in the along-track velocity direction
    - Length = along_track_velocity * Tmax, where Tmax is source (TIS-B, ADS-B) dependant, e.g., to account for a miss
      - Tmax = 2.1 sec for ADS-B
      - Tmax = 24.1 sec for TIS-B
  - The width of the CW is defined in the cross-track velocity direction
    - Width = 2 x Cross section of arc assuming a constant velocity (equal to the measurement speed) turn at 3deg/sec

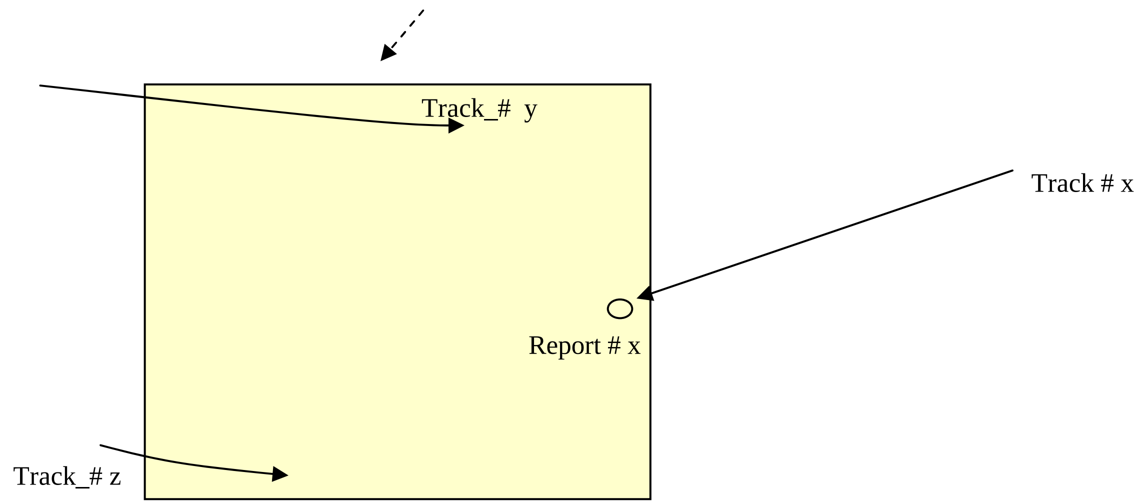


- Convert the four corners of the CW computed in ownship ENU to WGS-84 latitude and longitude. Also compute Altitude window size by extrapolating the altitude back in time using altitude rate information embedded in the report.

Report-to-Track Association – Candidate Window (cont'd)

- **Search the track file for all existing tracks within the CW defined in WGS-84.**
- **For each track in the CW, make sure a report field exists for the report # under investigation. If not, add a new field for that report.**

Candidate Window: x_{report}



Track.num	Track.SameCorrAdd	Track.SameCorrPattern. *this*
Track # x	# x	11111
	# z	101
	# a	1000
	# x	101
Track # z	# z	111111111
	# x	0001
Track # y	# y	1111

Reports #x and #z have Previously associated with tracks #z and #x, respectively.

Report-to-Track Association – Association Window

- **For each report, compute a report-to-track *Association Window* (AW) centered on the report, i.e., for every track that falls inside the CW. The size of the AW is calculated from the track uncertainty, measurement uncertainty and potentially an additional lag:**

$$x_{window} = \pm\left(3 * \sqrt{P_{\varepsilon_{xx}}} + \Lambda_x\right)$$
$$y_{window} = \pm\left(3 * \sqrt{P_{\varepsilon_{yy}}} + \Lambda_y\right)$$
$$z_{window} = \pm\left(3 * \sqrt{P_{\varepsilon_{zz}}} + \Lambda_z\right)$$

  - **$P\varepsilon_{xx}$, $P\varepsilon_{yy}$ are covariances of the residual vector between the measurement and predicted track position**

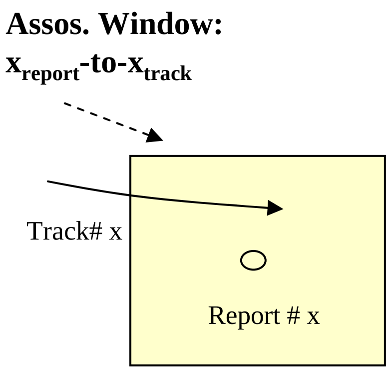
$$P_{\varepsilon_{xx}} = P_{M_{xx}} + P_{P_{xx}}$$
$$P_{\varepsilon_{yy}} = P_{M_{yy}} + P_{P_{yy}}$$
$$P_{M_{xx}}, P_{M_{yy}} \text{ are measurement variances } \left(\mathrm{NAC}_p^2\right)$$
$$P_{P_{xx}}, P_{P_{yy}} \text{ are predicted track variances}$$

  - **Λ_x , Λ_y, predicted lags for the track at the time of the measurement**
- **Predict the position of all the tracks inside the CW to the time of the report.**
  - **Convert the tracks from WGS-84 to ENU**
  - **Predict its trajectory assuming straight line motion.**

- **Determine if the predicted position of the track - found in the CW and used to compute the AW – in addition falls inside the AW for that report-to-track pair.**
  - **If the predicted track falls inside the AW, set a "1" in the *track.SamePattern* (M/N pattern) for that track (see later slide)**
  - **Else set a "0" in the *track.SamePattern* for that track.**
- **For all tracks except track with the same ID as the report that fall outside the CW**
  - **If there is no field for that report in the structure for that track, create a new field and set the *track.SamePattern* to "0".**
  - **Else, update the *track.SameCorrPattern* by adding a "0" to the end of the pattern.**

Report-to-Track Resolution Logic

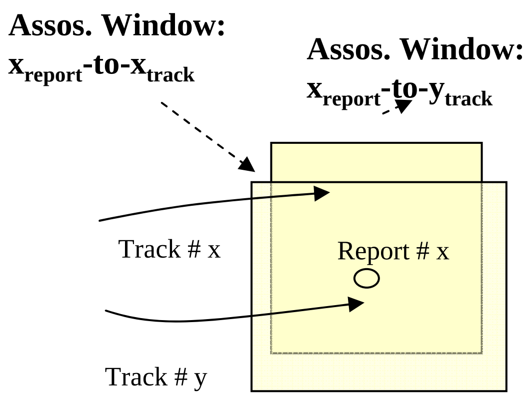
Scenario 1



Resolution Tag

Report # x	update track #x

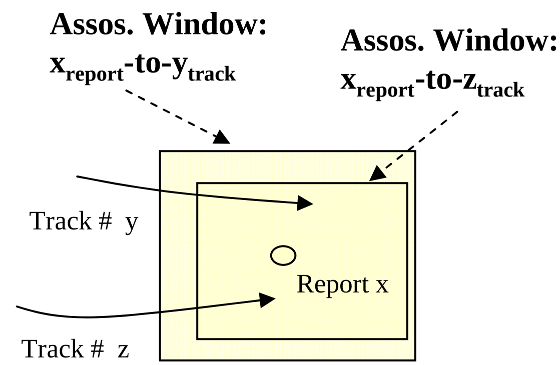
Scenario 2



Resolution Tag

Report # x	update track #x

Scenario 3



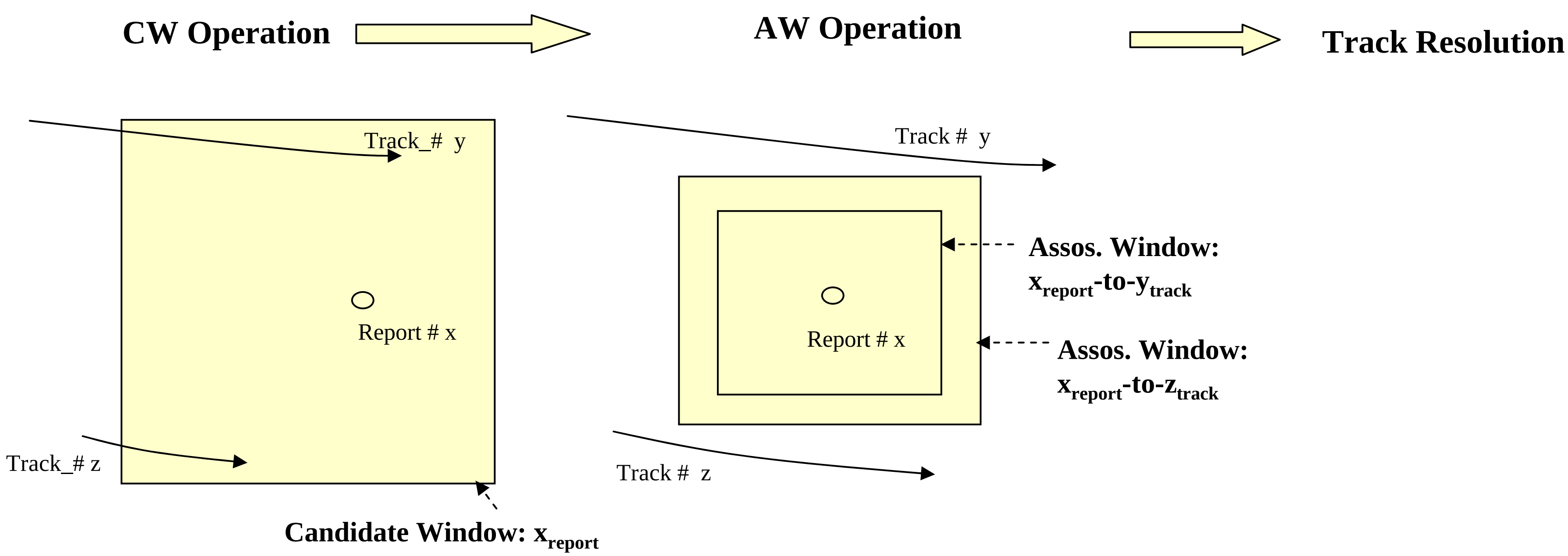
Resolution Tag

Report # x	Initiate new track #x

- Scenario 1: If only one track falls inside the association window, and that track has the same ID as the report, update the track with that report.
- Scenario2: If more than one track fall inside their respective AWs and one of the tracks has the same ID as the report, update that track with the report, even if the distance to the other track is smaller. Embed the information that the report associated with the other track(s) into the other track(s) structure and update the count (*M*) of such occurrence over the last *N* updates, denoted by *TrackSameCorrPattern*. If the occurrence count, *M*, exceeds a limit (e.g., 3) and other qualifiers are met, then merge the duplicate tracks. See more on Background Correlation - Merging Tracks slide
- Scenario 3: If none of the tracks inside their respective AWs have the same ID as the Report (e.g.,*x*) , do *not* update the track(s) (e.g., tracks *y* and *z*) with the report. The resolution for this report is to initiate a new track. However, embed this information (e.g., that the track #y associated with a Report # x) into the track structure (e.g., track *y).* If the track had previously been tagged as having correlated with Report (e.g, track #y with report #x), increment the occurrence for that address in the field *TrackSameCorrPattern*. If the occurrence count exceeds an *M/N* limit (e.g., 3/5) and other qualifiers are met, then merge the two tracks (see later).

MITRE CAASD

Report-to-Track Resolution Logic (cont'd)



Track.num	Track.SameCorrAdd	Track.SameCorrPattern.
Track # y	# x	empty
Track # z	# x	empty

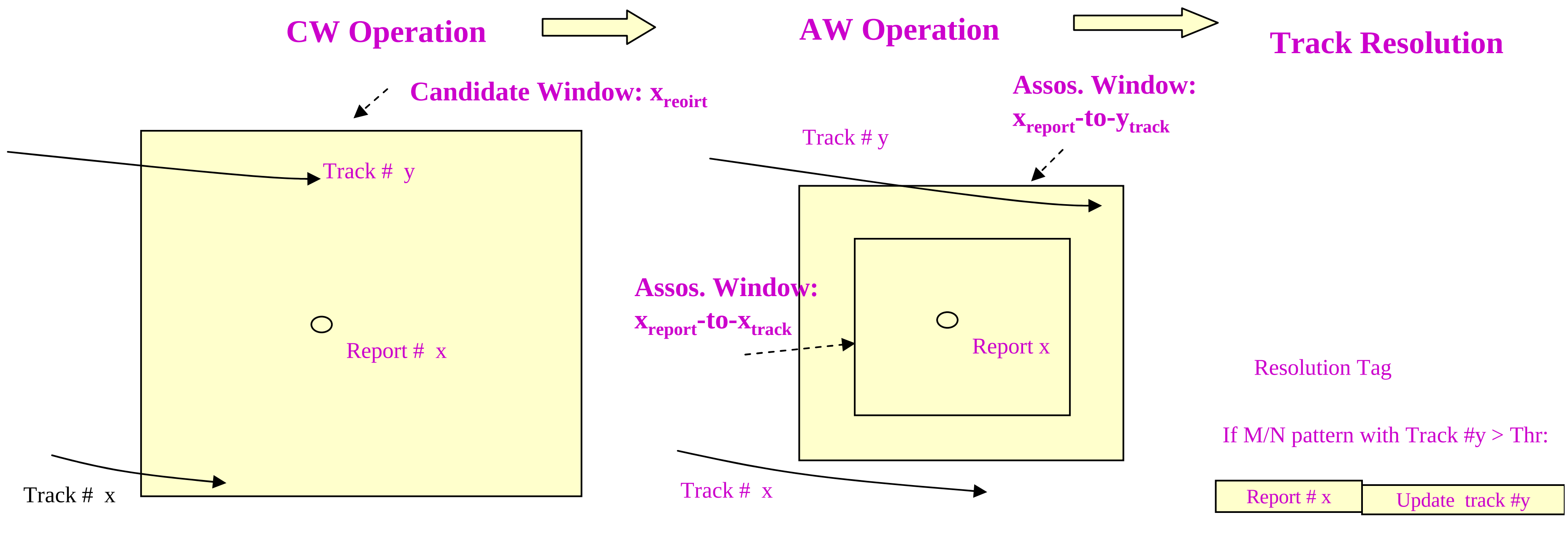
Track.num	Track.SameCorrAdd	Track.SameCorrPattern.
Track # y	# x	0
Track # z	# x	0

Resolution Tag	
Report # x	Initiate new track #x

- **In this example, Report #x is the first report on a new target and within its CW are two existing tracks, #z and #y.**
- **Since no track with the same Track ID as the Report falls inside the AW, nor do the tracks that associated with the report have a *TrackSameCorrPattern* that meets the threshold, tag this report for a new track initiation with the info imbedded in the ADS-B/TIS-B Track Report.**

- **As part of background correlation processing, periodically check to see if multiple tracks have the same ID and provide logic to**
  - **merge duplicate tracks with the same ID that meet certain criteria (see later slide on)**
  - **create a unique ID for the tracks that don't meet those criteria (see later slide)**

Association Rules and Resolution Logic - Scenario 5



Track.num	Track.SameCorrAdd	Track.SameCorrPattern.
Track # y	# x	….
Track # x	# x	10111

Track.num	Track.SameCorrAdd	Track.SameCorrPattern.
Track # y	# x	…1
Track # x	# x	101110

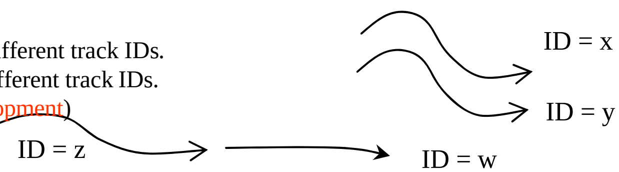
- **In this example, Report #x is <u>not</u> the first report on a target and the established Track #x along with another Track #y fall inside the CW of Report #x.**
- **Since Track #x did not fall inside the corresponding AW, add a'0'to the end of Track #x-> Report #x-> TrackSameCorrPattern.**
- **Since Track #y fell inside the corresponding AW, add a '1' to Track #y-> Report #x-> TrackSameCorrPattern.**
  - **If the updated pattern exceeds the threshold, resolve for Track #y to be updated with Report #x and note that in the future Track #y can be updated with Report #x.**
  - **If the updated pattern does not exceed the threshold, resolve for a new track to be initiated with ID #x. This will be a duplicate track #x, since one already existed. The Background Correlation logic will decide how to deal with this. (see later slide)**

Track Termination

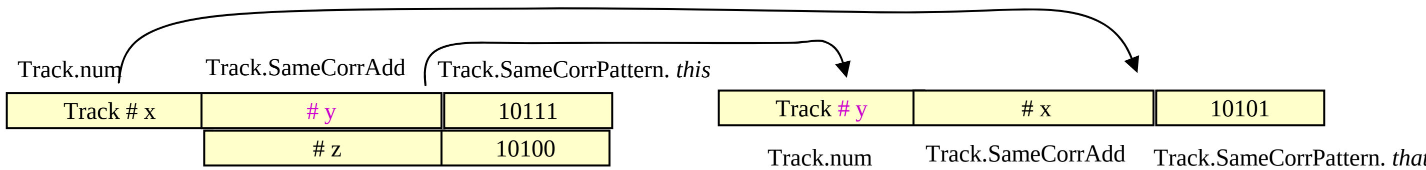
- ADS-B, TIS-B
  - Track from each individual source shall be terminated if it has not been updated for more than a set time interval which is dependant on the Service Volume or Application.
    - For TIS-B Enhanced Visual Acquisition, the track will be terminated if it has not been updated for 36.3 sec.
    - Fill out details

Background Correlation - Track Merge Logic of Correlated Tracks with Different IDs

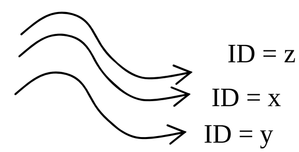
Examples:
TIS-B: Overlapping Service Volumes by CF are not networked together, resulting in more than one track with different track IDs.
TIS-B: Registration errors between multiple sensors, resulting in more than one track for the same target with different track IDs.
ADS-B: A/C switching from ICAO address to self-assigned temporary address and vice verso. (need more development)



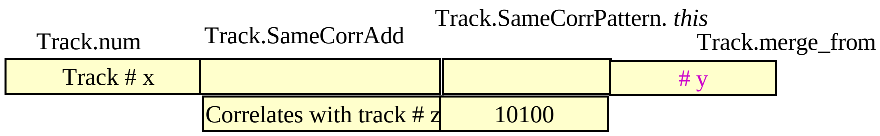
- **Delete the track with the worst NAC (e.g., track x has TOA of t_0 and track y has $t_{0+e,}$ extrapolate NAC of track x to $t_{0+e,}$) if**
  - **For a given track #x, the count for a correlated track ID #y , $M_{BG\_sameSource}$ (i.e., number of 1's in the detection pattern, example shown has 5), exceeds a threshold**
    - **The track file is searched for that track #y (i.e., track corresponding to Report #y which correlated with track #x), with the ID of the correlated track ID #x. The correlated cross-reference list of track #y is searched to find if it contains correlated track ID #x and its corresponding count.**

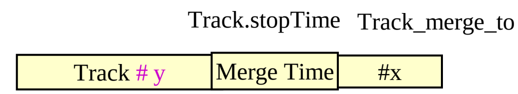


    - ***$M_{BG\_sameSource}$ _this > Thr_corr_count_this***
    - ***$M_{BG\_sameSource}$ _other > Thr_corr_count_that***
  - ***The other track, #y that is under consideration for deletion, only correlates with track #x, or correlated (in the past) with other tracks that have an all-zero detection pattern in Track.SameCorrPattern. In other words, if the track under consideration for deletion cross-correlates with other tracks, do not delete it because you may prohibit merging of dual.***
  - ***Speed difference < THR_s* and**
  - ***Ground Heading difference < THR_s* (optional)**



- **Reports with both IDs are from then on allowed to update a "merged" track.**

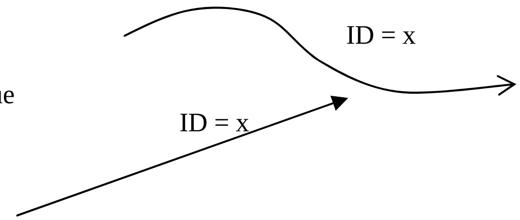




- **** "this" refers to track with same ID as report (in general)***

Background Correlation – "Splitting" Tracks of the Same ID

Example: Installer left default value or entered wrong ICAO value



- **Periodically the Background Correlator will search the track files (at a source level, i.e., for ADS-B and TIS-B tracks) to find if more that one track contain the same track ID.**
- **The criteria for deciding that multiple tracks with the same ID should be split into different track ID is**
  - **Spatial Correlation: distance between the track positions > *Thr_distance***
  - **Speed Inconsistency: difference in speed > *Thr_speed***
  - **Ground Track Inconsistency (optional): difference in Ground Track > *Thr_GT***
- ***Create a unique track number to the track with the worse NAC but allow it to be updated with reports of the previous ID #. (Both track can be updated with the same report ID. Since they are really different targets, the report will associate with only one track.)***

Before:

Track # x	T_1	Lat_1	etc
Track # x	T_2	Lat_2	etc

After:

Track # x	T_1	Lat_1	etc	
Track # w	T_2	Lat_2	etc	# x

Track.split from

Track-to-Track (ADS-B/TIS-B) Correlation/Bonding Logic

Goal: Correlate tracks from ADS-B and TIS-B sources for the same target

- Established in the following order:
  - Track.num is the same
  - Call Sign is the same (optional input)
  - Spatial correlation and velocity consistency check
    - Spatial Correlation: Establish Candidate Window on each updated ADS-B (TIS-B) track. Find candidate TIS-B (ADS-B) tracks. Extrapolate candidate TIS-B (ADS-B) tracks to the TOA of the ADS-B (TIS-B) update. Establish association window between ADS-B (TIS-B) update and each extrapolated TIS-B (ADS-B) track.. Cross-reference TIS-B and ADS-B tracks if correlation occurs and keep detection pattern (e.g., 1001111) and a count on number of occurrences over **<u>last</u>** N opportunities. If M-out-of-N correlations occur, consider the tracks bonded. Keep a list of correlated addresses for the updated Track and enter a “0” in the detection pattern for address that don’t correlate.
    - Velocity Consistency Check: Speed difference less than threshold

MITRE

CAASD

Best Track Selection Between ADS-B and TIS-B

- After correlation has been established, the local TIS-B Track.num is set to the Track.num of the correlated ADS-B track.

- **Best Track Selection is performed between TIS-B and ADS-B in cases where both are available.**
  - **For Gap Filler Service TIS-B will normally provide for targets that are not ADS-B equipped and no ADS-B track will be available.**
    - **The exception is when ADS-B may not be detected on the ground but the air-to-air is detected in which case both can be available to the A/C simultaneously.**
  - **In Full Picture Mode both will be available.**
- **When both ADS-B and TIS-B are available, the Track with the best NAC will be selected.**
  - **Future Applications: The exception is when TIS-B is in validation mode. If TIS-B indicates that ADS-B is invalid, TIS-B shall be selected.**

TCAS Tagging of ADS-B or TIS-B

Goal: Tag ADS-B and TIS-B tracks that correlate with TCAS measurements.

- If TCAS has Mode S ID, tag any ADS-B/TIS-B tracks that have the same ID
- Spatial correlation established on a time-based moving window
  - Establish Candidate Window on each new TCAS measurement. Candidate window is a cylinder :
    - For standard TCAS interface: The height, radius, Tmax of the candidate window are default values
      - Height = default_alt_rate* Tmax_tcas, e.g., Tmax_tcas = 2 sec to account for a miss
      - Radius = max_speed * Tmax_tcas, e.g., max_speed = 700 kts
    - For enhanced TCAS bus
      - Height = alt_rate* Tmax_tcas, e.g., Tmax_ = 12 sec for TIS-B association, 2 sec for ADS-B association
      - Radius = max_speed * Tmax_tcas, e.g., max_speed = 700 kts

Find candidate TIS-B and ADS-B tracks. Extrapolate candidate TIS-B and ADS-B tracks to the TOA (derived from time of reception) of the TCAS update. Establish association window between TCAS update and each extrapolated TIS-B and ADS-B track. Mark TIS-B and ADS-B tracks if correlation occurs and keep a count on number of occurrences over last N opportunities (Track.TCASDetectPattern, Track.TCASDetectIDs, Track.TCASDetectTimes). If M-out-of-N correlations occur, tag the ADS-B and TIS-B tracks by setting a "1" in the TCAS-Tag field. TCASDetectTimes are used to record the time of last TCAS correlation and determine if the track is still supported by TCAS by comparing the last update to the TOA of ADS-B/TIS-B track. If the difference is larger than Thr, then the TCAS-Tag is reset to zero (to indicate that the ADS-B/TIS-B track has left the TCAS detection region).

- Compute a covariance matrix for the TCAS measurement based on nominal TCAS accuracy values from **DO-185 (TCAS MOPS)**
  - **Range: 0.01 nmi**
  - **Brng:**
    - **small elevation angles: 9 deg rms with 27 deg peak**
    - **Higher elevation angles: 15 deg rms, 45 deg peak**

Track Structure for Storing data in Track File in WGS-84

- **Tracks will be stored as a structure of arrays with the following fields:**

– **Track.num**	**(track identifier)**
– **Track.type**	**(1 = TIS-B, 2 = ADS-B) (removed 3 = TCAS)**
– **Track.startTime**	**(time when the track started)**
– **Track.stopTime**	**(time when the track terminated; if void then track is still active)**
– **Track.TOA**	**(TOA of most recent update)**
– **Track.lat**	**(lat of most recent update)**
– **Track.lon**	**(lon of most recent update)**
– **Track.alt**	**(alt of most recent update)**
– **Track.ewVel**	**(east-west velocity at most recent update)**
– **Track.nsVel**	**(north-south velocity at most recent update)**
– **Track.NACp**	**(NACp of most recent update)**
– **Track.SameCorrAdd**	**(addresses of correlated tracks from same source, can be a vector)**
– **Track.SameCorrPattern**	**(Pattern of correlations for each of the addresses from same source)**
– **Track.DiffCorrAdd**	**(addresses of correlated tracks from different source, can be a vector)**
– **Track.DiffCorrPattern**	**(Pattern of correlations for each of the addresses from different source)**
– **Track.merge_to**	**(Track.num to which this track was merged)**
– **Track.merge_from**	**(Track.num of the track that was merged to this track)**
– **Track.split_from**	**(Track_num from which a split was made)**
– **Track.TCASDetectPattern**	**(pattern of detections and misses over N past TCAS Updates)**
– **Track.TCASDetectTimes**	**(correlation times of past N TCAS updates)**
– **Track.TCASDetectIDs**	**(TCAS ids that correlated over last N TCAS Updates)**
– **Track.TCAStag**	**(a Tag of “1” indicates that the track is supported by TCAS)**

Report Structure

- **TIS-B and ADS-B Tracks**
  - **Report.num**
  - **Report.type**
  - **Report.TOA**
  - **Report.lat**
  - **Report.lon**
  - **Report.alt**
  - **Report.ew_vel**
  - **Report.ns_vel**
  - **Report.NACp**
  - **Report.NACv**
- **TCAS Measurements**
  - **Report.num**
  - **Report.rng**
  - **Report.az**
  - **Report.alt**
  - **Report.ModeS**
  - **Optional: Range Rate, Altitude Rate**

Structure for each Track for Debugging in WGS-84

- **For each track record each of these parameters for each update:**

– **Track.num**	**(track identifier)**
– **Track.type**	**(1 = TIS-B, 2 = ADS-B) (removed 3 = TCAS)**
– **Track.startTime**	**(time when the track started)**
– **Track.stopTime**	**(time when the track terminated; if void then track is still active)**
– **Track.TOA**	**(TOA of most recent update)**
– **Track.lat**	**(lat of most recent update)**
– **Track.lon**	**(lon of most recent update)**
– **Track.alt**	**(alt of most recent update)**
– **Track.ewVel**	**(east-west velocity at most recent update)**
– **Track.nsVel**	**(north-south velocity at most recent update)**
– **Track.NACp**	**(NACp of most recent update)**
– **Track.SameCorrAdd**	**(addresses of correlated tracks from the same source, can be a vector)**
– **Track.SameCorrCnt**	**(count of correlations from the same source for each of the addresses above)**
– **Track.DiffCorrAdd**	**(addresses of correlated tracks from different source, can be a vector)**
– **Track.DiffCorrCnt**	**(count of correlations for each of the addresses from different source)**
– **Track.split_from**	**(Track_num from which a split was made)**
– **Track.TCASDetectPattern**	**(pattern of detections and misses over N past TCAS Updates)**
– **Track.TCASDetectTimes**	**(correlation times of past N TCAS updates)**
– **Track.TCASDetectIDs**	**(TCAS ids that correlated over last N TCAS Updates)**

Backup Slides

- **TCAS accuracies**
  - **DO-185 (TCAS MOPS)**
    - **Range: 0.01 nmi**
    - **Brng:**
      - **small elevation angles: 9 deg rms with 27 deg peak**
      - **Higher elevation angles: 15 deg rms, 45 deg peak**
  - **Rockwell's TCAS-94**
    - **Surveillance range: 35 nmi max, 14 nmi min**
    - **Surveillance capacity: 150 targets, 30 displayed**
    - **Bearing accuracy: 5 deg rms, 10 deg peak on standard ground plane**

Out-of-Sequence Logic

- **Nothing can be done about that because the TOR is the assumed TOA.**

TCAS Registration and Tracking Issue

- **Decided that there is no need to generate a track for TCAS.**
- **Decided there is no need to correct the bias between TCAS and ADS-B because the goal is to just tag the ADS-B and TIS-B tracks, not fuse the TCAS measurement into the other tracks.**