# White Box approach for a Trajectory Prediction Tool

Common Trajectory Modeling
CCOM – Action Plan 16

Round table on Potential of OSS in ATM

(Jointly organized by CALIBRE and Eurocontrol)

Garcia Avello C. & Swierstra S. DAS-ATS EUROCONTROL HQ

December 7, 2005

# Trajectory Prediction and ATM

- TP is present in most of the key components of the ATM system

*On the Ground*

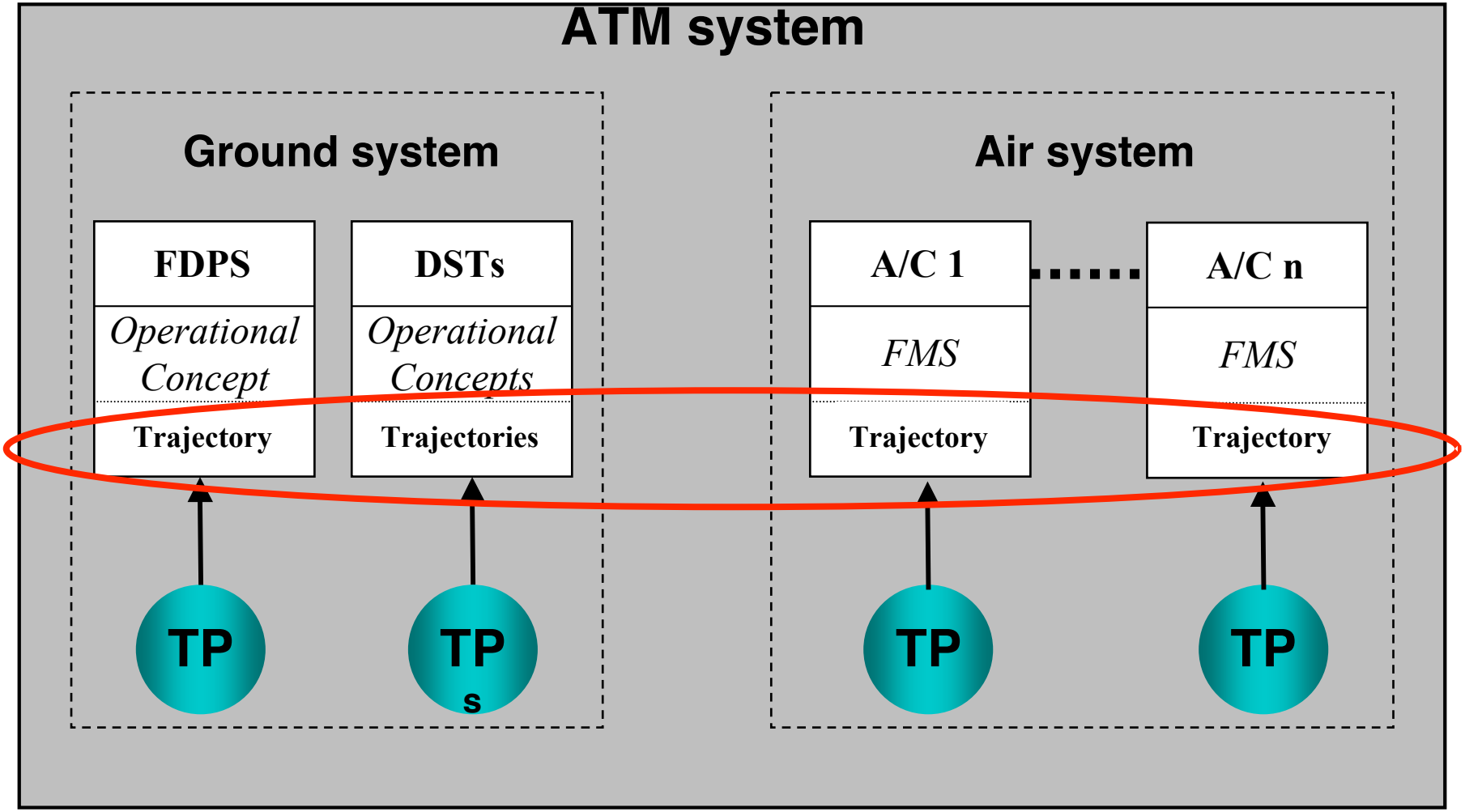
- CFMU for preflight and real time flow management
- FDPS for identifying sectors to be flown
- FDPS for estimation of sector entry exit times
- To support new tools dealing with the planning and/or separation of traffic
- Within Airline Operations Centers for optimum city pair flight planning

*On the Air*

- FMC needs a trajectory for applying its guidance (L-nav, V-nav)

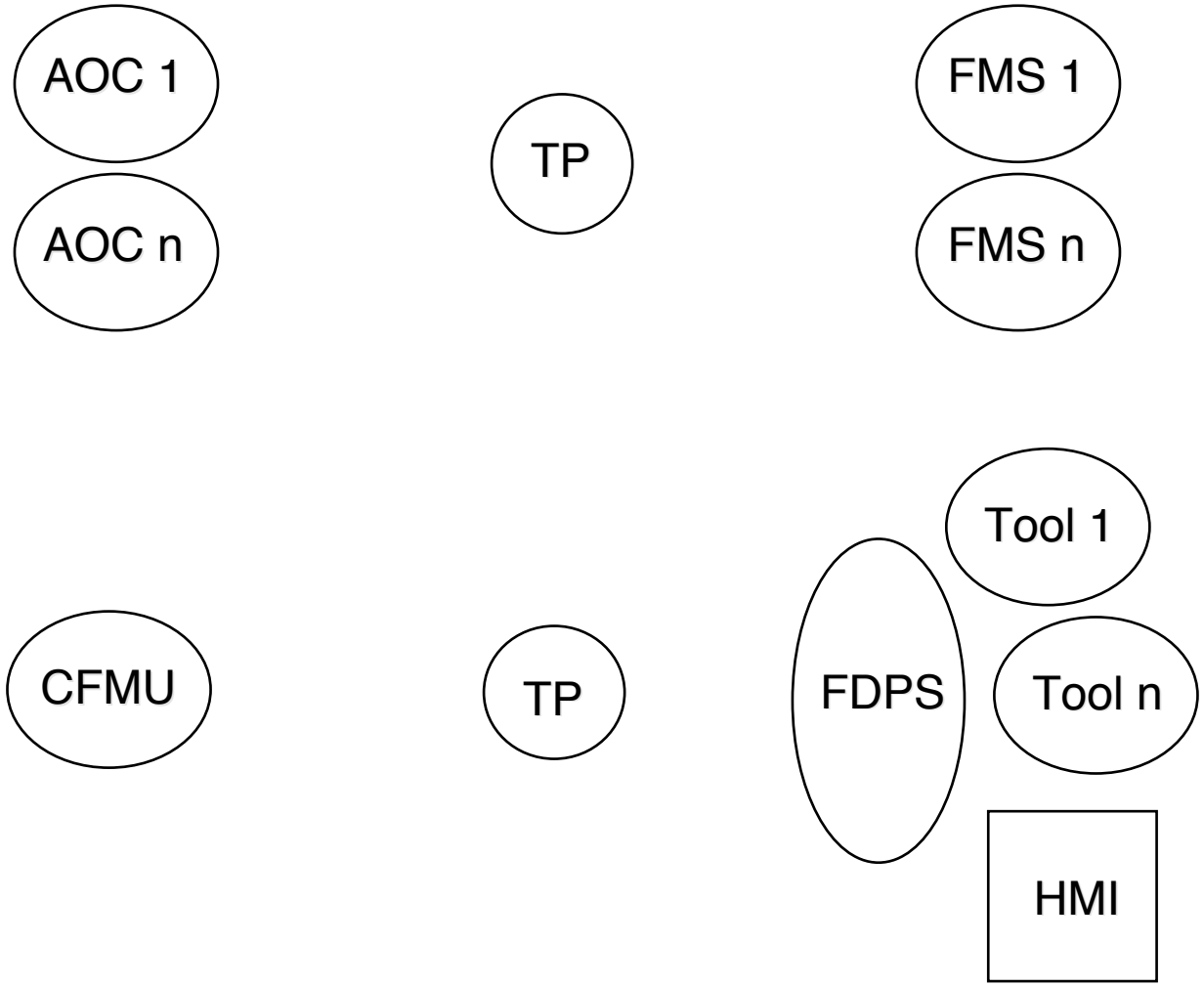
Future ATM is referred to as “Trajectory based ATM”

# The TP issue: system context

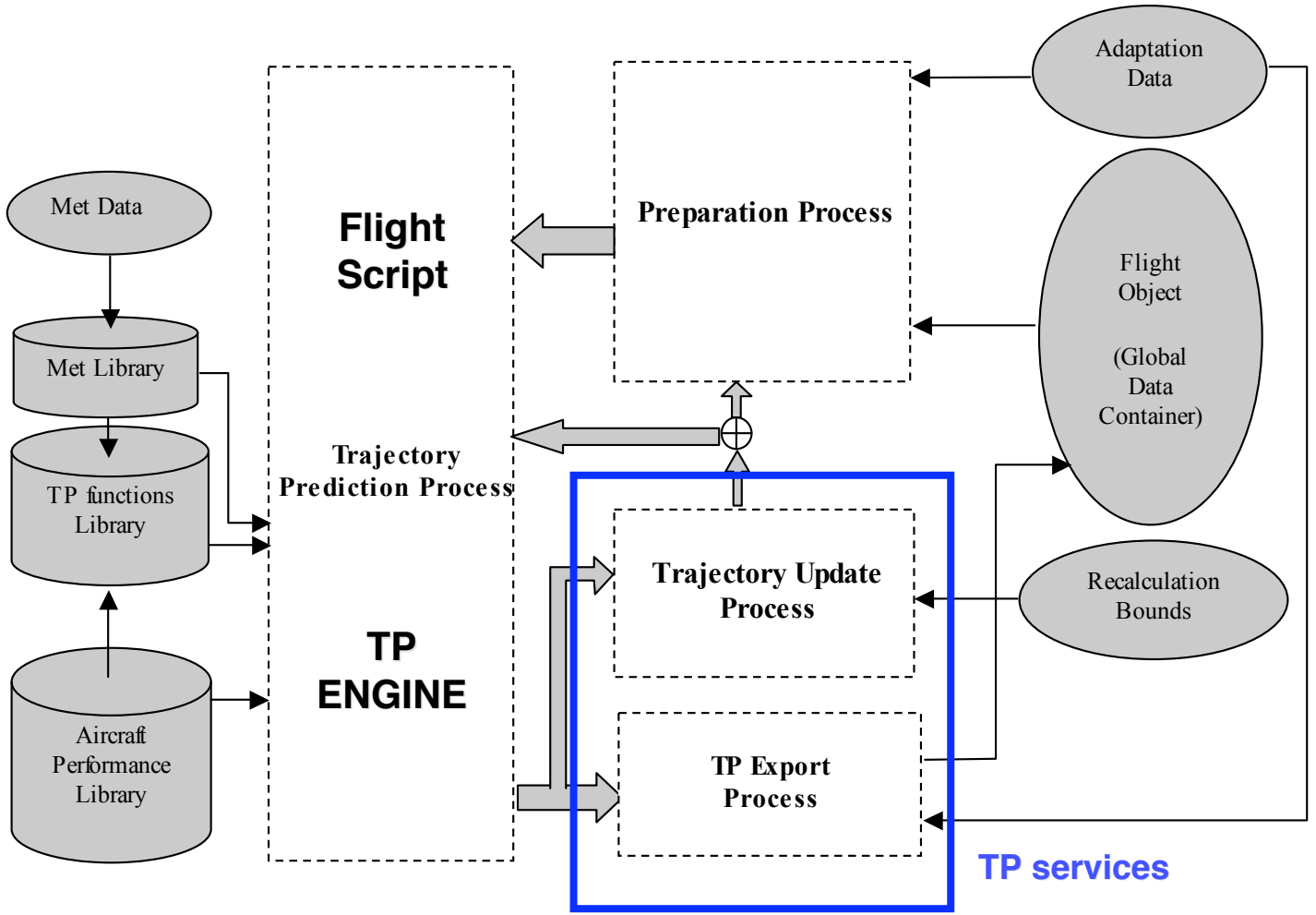



**Challenge is consistency**

# A nearly perfect ATM world …

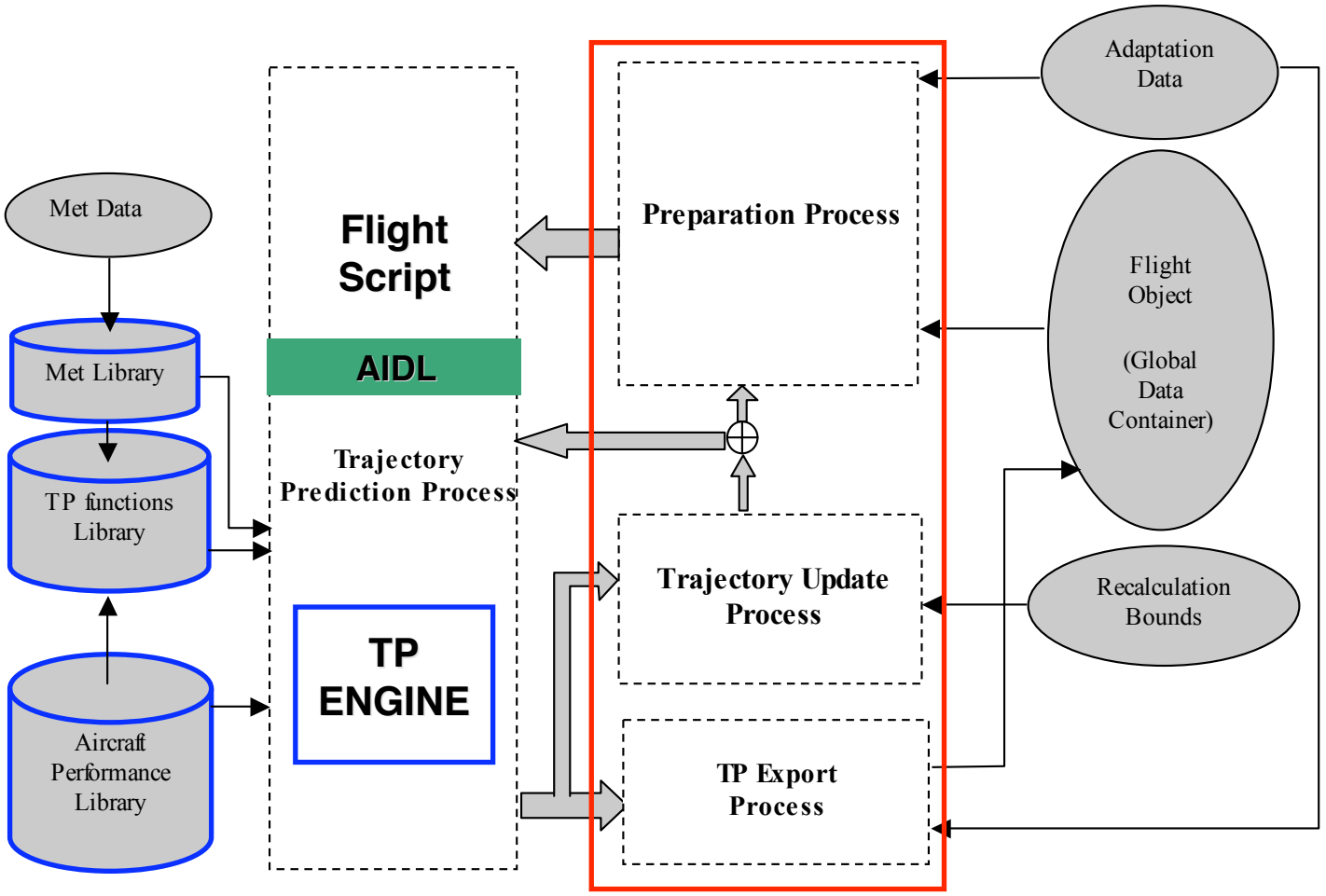


# Generic TP Structure* (air or ground)




*Defined in the Action Plan 16: Common Trajectory Prediction-Related Terminology

# OSS candidates



# Flight Script

FS allows the decoupling of the TE and the TP underlying applications

FS needs to be suitable for air and ground trajectory prediction synchronization

FS functionality and granularity is application dependent

- Open loop / Close loop
- Accuracy / response time

Common Aircraft Intent Description Language

# TP OS components

## Met Functions library

- **Calibrated air speed to True air speed conversion**

TAS = CAS/(1-6.8755856*10^- 6 * DA)^2.127940 (DA<36,089.24ft)

where D_Alt = P_alt+ (T_s/T_r)*(1.-(T_s/T)^0.2349690)

## Aircraft performance Functions library (not APMs)

- **Turn radius**

TR(feet) =TAS(kt)^2/(11.23*tan(0.01745*BA(dgr)))

## TP Functions library

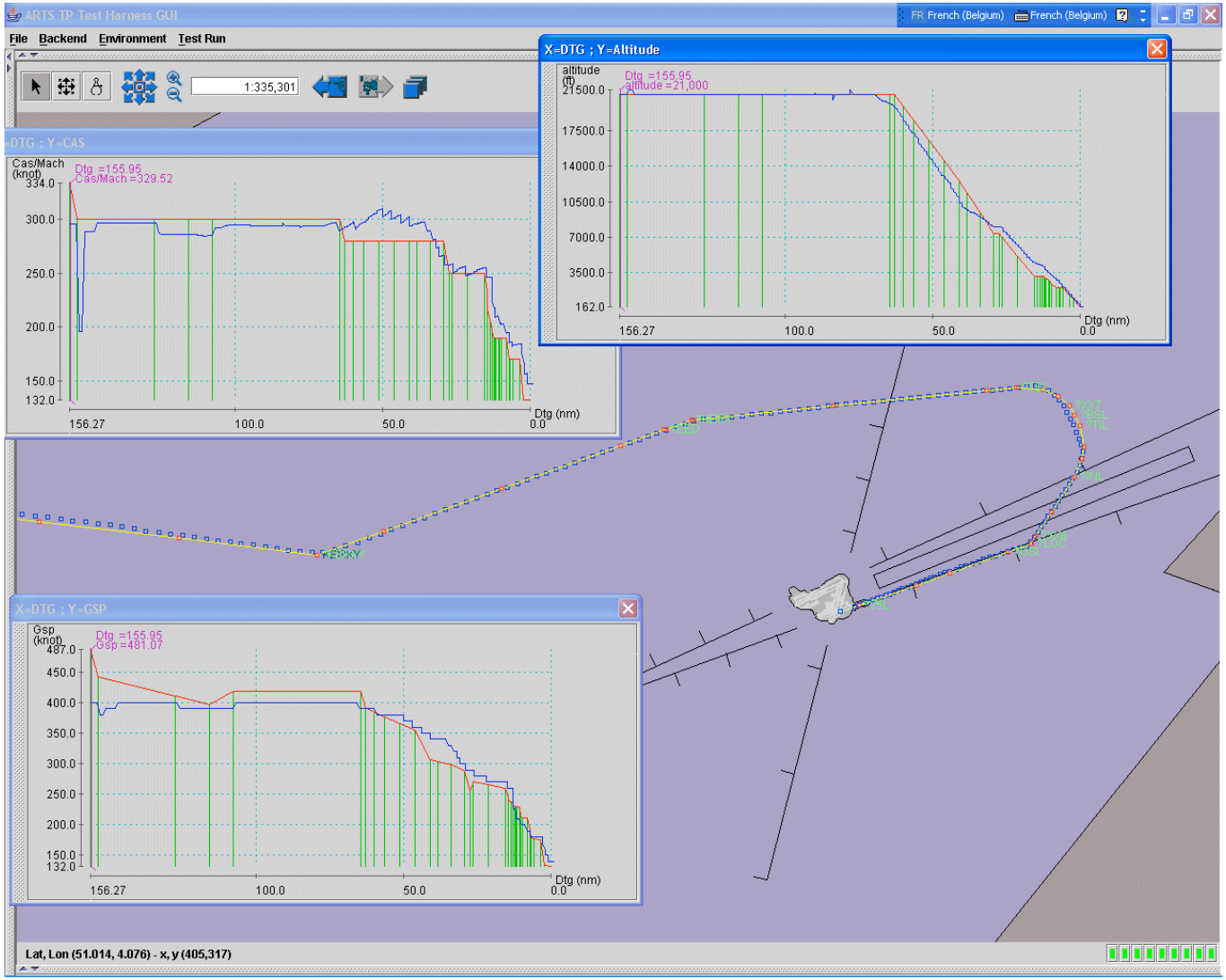
- **Great Circle navigation between 2 waypoints**

lat=atan((sin(lat1)*cos(lat2)*sin(lon-lon2) -sin(lat2)*cos(lat1)*sin(lon-lon1))/(cos(lat1)*cos(lat2)*sin(lon1-lon2)))

# The TP white boxing process steps

- *Develop OS Met and Aircarft Performances functions*
- *API for the decoupling of APM and TPs*
- *Aircraft Intent Description Language allows the decoupling of the Trajectory Engine and the application*
- *White box a TE to allow the user to understand the TP insides*
  - > *Test Harness to allow the evaluation of multiple TPs with the same reference data, validation strategy and agreed set of metric*
    - **Front End is OS**
    - **Back end is TP specific**
  - > *White Box the TE of the TP(s) linked to the Test Harness*

# Test Harness output



# TP OS benefits

*FS AIDL facilitates air and ground trajectory synchronization*

*OSS Met, Aircraft performance and TP functions facilitates TP output consistency*

*API for aircraft performance models decouples TP from model*

*OSS Trajectory Engine connected to OSS Test Harness*

- *Allows the user to run a TP and analyze performance*
- *Facilitates the identification of input parameters needed to satisfy a required level of performance*
- *Allows the user to identify tradeoffs between accuracy and response time and the reasons for it*
- *Support the user to specify a given level of TP performance to support the underlying decision support tools and quantify costs*

*Pave the way to an ATM system using an OSS TE*

*Cost, benefit from other users improvements, …*