Contents

Introduction .......................................................................................................................... 3
Flight Data Monitoring Events ............................................................................................ 4
  Inside-Out Approach ........................................................................................................ 4
  Parameters and Limitations ............................................................................................. 5
Testing Your Code ............................................................................................................... 7
  Testing for False Positives .............................................................................................. 7
  Testing for False Negatives ............................................................................................. 8
Other Recommendations .................................................................................................... 8
Takeaways .......................................................................................................................... 8
Introduction

The fundamental function of Flight Data Analysis (FDA) is to extract information from flight recorded data, to enable airlines to make better decisions in order to enhance flight safety, aircraft efficiency, maintenance activities, passenger experience, and more. The analysis performed by an FDA program is dynamic and constantly changing, meaning that a problem an FDA analyst wants to address today, will not be the same problem as one to be addressed tomorrow.

Due to the dynamic nature of these analyses, it is essential that flight data analysts work with an FDA service or software provider that allows them the flexibility to create new analysis elements, such as parameters, analytical points, events, and reports. Teledyne Controls’ Services team has more than 20 years of experience in flight data analytics. We collaborate with airlines to code and implement any of these analytical elements, should our customers need it. Teledyne Controls’ FDA customers (hosted or installed) work with AirFASE®, which provides users with a flexible and powerful analysis editor module to create their own analysis elements, at any time. While AirFASE analysis editor module is relatively simple to operate, Teledyne Controls also offers regular training sessions to customers who are looking to code and implement FDA analytical elements efficiently themselves.

1 AirFASE® is a Teledyne Controls FDA software that performs the data processing and analysis functions that turn raw data into meaningful information to support flight operations monitoring, safety management and aircraft maintenance programs.
Flight Data Monitoring Events

Creating a new event is typically the most challenging FDA analysis element to construct. In general terms, an event is an algorithm that detects a particular condition inside the flight-recorded data. This condition is usually related to an exceedance, deviation, or occurrence of a particular parameter value, or of a particular aircraft behavior or procedure.

Recognizing the need for a new event is relatively straightforward but defining its requirements and understanding restrictions needed to code it is more complicated. For example, if a Flight Safety department wants to monitor deviations from Selected Altitude in the middle of cruise, they put an analyst in charge of writing an event for this condition. They assume it is as simple as comparing the aircraft’s current altitude with the selected altitude. However, when the analyst goes to write the code many questions come to mind. Should Pressure or Barometric Altitude be used for this event? When should the event use Selected Altitude from the FCC or FMC? What happens if the aircraft is moving to a different flight level on purpose? All these questions make writing this event a harder task than originally anticipated. If an FDA analyst wants to fly solo and create their own event, they can start by following these recommendations:

Inside-Out Approach

While it is preferable to define all the requirements for the implementation of a new event, you do not need to collect all these requirements to start coding your event. It is actually easier to start coding by simply implementing the essential trigger condition. Writing the most basic event detection criteria and adding the rest of the code on top later, ensures that the event code will be flexible and scalable. This is important as it allows you to adjust to new requirements that may arise later. A basic event detection criterion must be a combination of the simplest triggering exceedance condition (the when) and the most basic triggering monitoring window (the where). Don’t let unknown constraints stop you from the most important step: starting.
Monitoring problems with Radio Altitude Transducer readings is an example of where an event would be useful for an FDA analyst inside a maintenance department. The event would alert if the system detected discrepancies larger than 200ft between the left and right Radio Altitude Transducer signals. The simplest way to implement this event would be to take the two Radio Altitude values, calculate the absolute difference between them, and trigger an alert if the difference is more than 200ft during the initial climb or final approach. After defining these essential trigger conditions and monitoring window, you can add more conditions, such as validating that the aircraft’s actual radio altitude is below 2500ft or confirming that the difference in the readings existed for more than 3 seconds. Any other conditions you want to include should be added to the code at this point. After adding additional conditions, you can look for ways to make the code more efficient. You might want to combine different if-statements, reduce the number of local variables, or simply add more conditions to reduce as much as possible in the monitoring window.

**Parameters and Limitations**

One of the main advantages of FDA analysis is having a large number of parameters available to analyze. Values from these parameters are being recorded every second of the flight, which is a lot of data at your disposal, but specific parameter values are not always valid. Air Speed (CAS) is not captured below 45 or 40 knots and Radio Altitude is not reliable above 2,500 or 3,000ft. Therefore, you must use Ground Speed (GS) for Taxi-Related events, and Radio Altitude for Low Altitude related issues. Understanding when parameter values are valid or not can distinguish a good event from a bad one. The figure below shows the takeoff-initial climb segment of a flight, where the aircraft is moving on the ground, based on valid GS readings of 12 knots; however, the CAS indicates a zero value. When the aircraft is moving faster than 40 knots, the CAS starts providing valid values. It can also be seen how the radio altitude reading becomes invalid after 3000ft.
Understanding the frequency of the acquisition and recording of parameters is also essential in creating a new event. A specific standard data frame\(^1\) for the 757 fleet records EGT (Exhaust Gas Temperature) in the super frame, which means its value refreshes every 64 seconds. If you think about it, you will only detect an event if the sample in which EGT gets updated coincides with the exceedance of the EGT limit of the event. Knowing the frequency of your parameters will help you better plan what you will capture. You might need to do extrapolations or have the Teledyne Service team increase the frequency of the recorded parameter in the Aircraft Condition Monitoring System (ACMS) section of the aircraft data acquisition and management unit: DFDAU, FDIMU, or DMU.

If a parameter is not available in your current data frame, look for ways to derive it or include it in by modifying the ACMS recording map. Deriving a parameter is a good way to get the parameter that you need, that is not present, by using the value of other available parameters. Let’s say you need the Static Air Temperature (SAT) to validate the behavior of an engine at a particular setting, but you don’t have the SAT. However, you do have the Total Air Temperature (TAT) and Mach number in your data frame. By using the standard isentropic flow for a calorically perfect gas\(^2\), it is possible to calculate SAT from TAT and MACH. Teledyne Controls offers regular training sessions for those looking to further understand writing code to create derived parameters.

\[
\text{SAT} = \frac{\text{TAT}}{1 + 0.2 \times \text{MACH}^2}
\]

It was also mentioned that modifying the ACMS recording map is another way to add missing parameters for your analysis. The ACMS is the function inside of the DFDAU or FDIMU in charge of acquiring the parameters coming from different sensors and LRUs. ACMS acquires thousands of parameters but only records the ones inside the non-mandatory data frame. For this reason, it is possible that a parameter you need was acquired, but for some unknown reason, not recorded. Adding a parameter into a data frame can be a simple process and is done by using Teledyne Controls’ ACMS ground configuration tools such as AGSIV and GBAT\(^3\). The hardest part will be loading the new ACMS with the new data frame into the complete fleet. To avoid unnecessary work in the process, we recommend adding several new parameters each time.

---

1 Data frame refers to the set of parameters recorded for a specific aircraft configuration and to the layout of these parameters within the ARINC 717 structure.
2 Air is considered a calorically perfect gas at subsonic speeds.
3 AGSIV and GBAT can only modify ACMS of Teledyne Controls Data Management and Acquisition units.
Testing Your Code

We often code an event based on a perfect scenario in our heads, where the parameters behave exactly the way we think they should. Think of oil quantity, we expect that its value would only experience decrements during a single flight because no one is adding oil to the engines mid-climb. However, flight-recorded data usually shows abrupt decrements in oil quantity before takeoff and huge increments between takeoff and climb (Refer to figure 2). If the event you are writing is supposed to detect oil leakages, this unexpected behavior of the oil quantity parameter could produce a false positive. This is why it is very important to test your code first by running it against recorded data to make sure it works as expected.

Testing for False Positives

False positive events are detected as a consequence of problems with the code and quality of the data. It is recommended to have a test environment where you can run your new event against a good amount of historical data; anywhere from 50 to 80 flights and, if possible, from different aircraft. The more data you run to test new events before putting them inside the production environment, the better the chances that you won’t have false positive detections.
Testing for False Negatives

Not being able to detect a real event might be worse than detecting an incorrect one. False negatives are a little trickier to test and require some creativity. During testing, try to force the detection of an event by relaxing the limits for its detection. Let’s say you are trying to monitor low duct engine pressure and want the trigger to alert when the pressure decreases below 20 PSI, but when looking at your historical flights for testing, you notice that duct pressure only goes as low as 24 PSI. To test the implementation of your event, change the triggering limit from 20 PSI to 25 PSI and keep all the other code conditions the same; you should be able to detect an event now. If the event is not detected, this is a clear indication that something else is wrong with the code. Make sure you restore the original limits after you are done testing for false negatives before moving your event to the production environment.

Other Recommendations

Once an event is created, it is important to follow an internal version control management and documentation process. As you continue making improvements to the code of your events, you might add additional trigger conditions, new specific requirements, or code improvements you didn’t consider the first time around. It is important to follow an internal version control management and documentation process once an event is created to keep track of the evolution of the event. It also helps to have a safety net you can come back to if the new change to the event didn’t work out as intended. The implementation of your code is not the same as anybody else and adding a documentation process will ensure that other users understand what you are trying to achieve with the event.

We also recommend participating in forums where other analysts, from other companies, participate and share their experiences allowing you to see different ways to address common needs. Talking to colleagues is another effective way to receive ideas on how to save time by either applying specific code techniques or avoiding problems that they already encountered and solved. Teledyne’s Data Collaboration forum is a great resource where users with any experience level share their findings: https://teledynecontrols.my.site.com/Customer/s/group/0F95c000000g6ehCAA/data-collaboration-success-group

Takeaways

Writing a new event is, without a doubt, a peculiar experience. While it may be fun, challenging, and stimulating, it can also be a very frustrating process. Don’t let the frustration prevent you from starting. Getting to see your event working as you intended is a very rewarding result. As with anything else, the more you practice, the better you will get at implementing code for flight data analysis.
Our Specialist

Edgar Salvador

Edgar is a highly experienced professional in the field of Flight Data Acquisition, Management Systems, and Flight Data Analytics currently serving as Business Development and Technical Director at Teledyne Controls. With over 19 years of expertise, he has consistently demonstrated a deep understanding of airlines’ flight data analytical needs and has used this understanding to research, create, implement, and test valid solutions for his customers. He holds an MBA, a Master’s in Aerospace Science, and a Master’s in Data Science which have provided him with a solid foundation for addressing the complex challenges and opportunities within the realm of Flight Data Analysis and Customer services.

Teledyne Controls

An integral part of the Aerospace Electronics segment, Teledyne Controls LLC is a wholly owned subsidiary of Teledyne Technologies Incorporated. Teledyne Controls is a leading manufacturer and innovator of a wide range of data management solutions designed to help aircraft operators collect, distribute, and analyze aircraft data more efficiently. Teledyne Controls maintains worldwide facilities and a global network of field representatives to support its many airlines, airframe, and military customers. To learn more about Teledyne Controls, visit: www.teledynecontrols.com.