NATS Multi-user Simulation

Version: beta 1.4

NATS release Beta 1.4 supports multiple users in collaborative simulations. This multi-user feature is a simulation mode in which multiple users operate different aircraft in the NAS at the same time. Different types of clients (normal user clients and external aircraft clients) log in and modify aircraft states and controls individually in the same simulation environment. The simulation is processed in real-time clocking, not fast-time. The following description and examples are provided to aid the user to better understand this feature, and how it may be used in future investigations.

1. Types of Aircraft Control

Two types of aircraft are supported:

1.1 Regular aircraft

Aircraft operate according to the flight plans in the Master TRX file. After NATS engine processes the TRX file, these valid aircraft can be used in simulation.

1.2. External aircraft

Aircraft from external simulators (Xplane, for example). Users have to execute specific functions to create aircraft profile data and send the state/control data to the NATS Server. NATS engine will propagate new state based on previous state and control data. Clients can send real-time state/control updates at every time step to NATS Server. NATS Server will collect the data from all users and apply them to create the new state vector for all the aircraft operating in the NAS.

2. Definition of Users

All users must be defined in

```
NATS_Server/share/user.conf
```

Three types of roles are defined in this file: Administrator, Simulation_Admin and Normal_User. In each type of roles, multiple users can be configured with authentication IDs. For example,

```
[Administrator]
admin

[Simulation_Admin]
sim1

[Normal_User]
user1:AC_ID_1, AC_ID_2
user2
```

Rule for user login:

- If the NATS Client and Server installed on the same machine:
  - If a client does not need to login, NATS Server automatically grants Administrator permission to it.

- If a client calls the login function, it will be logged-in in the role according to its authentication ID.

- If the NATS Client and Server installed on different machines:
  Clients must login with authentication id in order to execute functions.

Rules for assigning aircraft (refer to example `user.conf` above):

- Administrator can define how aircraft are assigned to users by putting aircraft IDs followed by ":" sign for each user. Multiple aircraft can be assigned to one user by using "," sign to separate different aircraft IDs.
- When `SimulationInterface.load_aircraft` function is executing, TRX file and `user.conf` content will be used to assign the ownership of aircraft.
- Aircraft which are not assigned in `user.conf` file can be requested to get the ownership by normal users by calling function
  `SimulationInterface.request_aircraft(String ac_id).`

3. Sample Code for Demonstration

Multi-user sample can be found in the directory
`NATS_Client/sample/DEMO_MultiUser`

Three types of users are illustrated in the directory:

- Client_Admin: Samples of operations belonging to the Administrator role.
- Client_NormalUser: Samples of operations of a normal user who operates an aircraft already in the TRX file to get state data and set new values back to the NATS Server.
- Client_ExternalAircraft: Sample of as XPlane simulator to send data to the NATS Server.

The three clients must be executed from independent terminal consoles to represent different users.

The following steps illustrate the demonstration example.

Pre-simulation step:

- NATS Server uses default TCP port 2017 as a connection point. If you need NATS Server to listen to different port, please modify NATS_Server/run script to set a different port number.
- NATS Client uses default TCP port 2018 as a connection point.

  Configure firewall setting so that the NATS Server and Client can be accessed through their corresponding ports.

Open three terminal console windows and run sample files in the following order.

Execution Order	Terminal Console A Client_Admin	Terminal Console B Client_NormalUser	Terminal Console C Client_ExternalAircraft
1	`python sample/DEMO_MultiUser/Client_Admin_Step1_beta1.4.py`		

	This program is to log in as the Administrator and load the 10-flight sample TRX file.		
2	python sample/DEMO_MultiUser/Client_Admin_Step2_beta1.4.py This program is to log in as the Administrator and start real-time simulation.		
3		python sample/DEMO_MultiUser/Client_NormalUser_Step1_beta1.4.py This program is to log in as a Normal User and set aircraft state.	
4			python sample/DEMO_MultiUser/Client_ExternalAircraft_beta1.4.py This program is to log in as a Normal User, create external aircraft profile and send aircraft state to the NATS Server.
5	Wait for Terminal B and C to finish. python sample/DEMO_MultiUser/Client_Admin_Step3_beta1.4.py This program is to log in as Administrator, stop real-time simulation, and output the trajectory data.		
6		python sample/DEMO_MultiUser/Client_NormalUse	

		r_Step2_DownloadFile_beta1.4.py This program is to request the download of the latest trajectory file.	

After `Client_Admin_Step3_beta1.4.py` finishes, the trajectory result file is located in the `NATS_Server` directory. You can view the details of the aircraft state data from it.

Also, the client can download the latest trajectory file from NATS Server. Sample code:
`Client_NormalUser_Step2_DownloadFile_beta1.4.py`

4. Description of How the NATS Multi-user Simulation Works

4.1 Preparation before starting NATS Server

- All users submit their flight plans in the standard TRX format to the Administrator. These aircraft are considered to be "owned" by individual users.
- Administrator collects all flight plan records and combines them into a single TRX file.
- Administrator provides unique authentication codes to all users. Copy of these codes and aircraft ownership are maintained in the `user.conf` file, and are used to ensure the states and controls of the aircraft owned by the user can only be modified by that authenticated user.

4.2 Starting server. Before simulation

- Administrator loads the TRX file.
- External aircraft can be created by calling the function `externalAircraft_create_trajectory_profile`
- Normal user can request access to individual aircraft by inputting the aircraft ID. The requested aircraft ID must exist on NATS Server. List of all aircraft IDs in the NATS can be obtained using the function `AircraftInterface.getAllAircraftId()`

4.3 During the simulation

Simulation time is synchronized with UTC (Universal Ttime)

Default time step = 30 seconds

When simulation time = 0

- External aircraft
  The latitude, longitude, altitude, airspeed, course angle, rate of climb/descent are set from the initial data provided when creating external aircraft profile.
- Regular aircraft (from TRX)
  - The states and control of aircraft which are assigned to specific users can be updated by them by calling the function to set new states and controls. These new states will overwrite the values in the NATS server.

When simulation time = 0+DT

- External aircraft
  NATS uses the states and controls to calculate new state. Client program can call function `externalAircraft_inject_trajectory_state_data` to send state data. The States and controls sent by the client will overwrite the NATS propagated state.
- Regular aircraft (from TRX)
  - They will be propagated no matter they are assigned to users or not.
  - The states and control of aircraft which are assigned to specific users can be updated by them by calling the function to set new states and controls. These new states will overwrite the values in the NATS server.
- The NATS Server allows 5 seconds to listen for client inputs. Any client state data (aircraft setter functions or external aircraft injecting function) will be ignored if it is not sent within the 5-second listening window.

4.4 After simulation

- Administrator writes trajectory result to a file.
- All clients can download this file by calling function
  `requestDownloadTrajectoryFile(String localFilePathName).`