

scenario	→	declaration* instruction+
declaration	→	let \$name = { value (, value)* }, let \$name = [(Float Integer) , (Float Integer)] ,
instruction	→	alter target scope trigger? parameters alter target scope trigger? trajectory hide target scope trigger? hideParameter? create scope trajectory createParameters? replay target scope parameters? delay target scope delayParameter saturate target scope trigger? saturationParameters rotate target scope trigger? rotateParameter cut target scope trigger?
target	→	plane filter all_planes filter?
filter	→	satisfying String (and String)*
scope	→	at time from time until time at time for time
time	→	Integer seconds
trigger	→	triggered_by String
value	→	String numberOffset constant
numberOffset	→	(>> <<)? number
number	→	Integer Float recordingValue
recordingValue	→	Float * recordingParameter
parameters	→	with_values parameter (and parameter)*
parameter	→	parameterType = (Float Integer String) parameterType (+= -= ++= --= *=) (Float Integer)

saturationParameters	→	with_values saturationParameter (and saturationParameter)*
saturationParameter	→	saturationParameterType = (String Integer Float)
hideParameter	→	with_frequency = Integer
delayParameter	→	with_delay time
createParameters	→	with_values createParameter (and createParameter)*
createParameter	→	createParameterType = (String Integer Float)
rotateParameter	→	with_angle (Integer Float)
trajectory	→	with_waypoints [waypoint (, waypoint)*]
waypoint	→	(Float, Float) with_altitude Integer at time
parameterType	→	ALTITUDE CALLSIGN EMERGENCY GROUND_SPEED ICAO LATITUDE LONGITUDE SPI SQUAWK TRACK
saturationParameterType	→	ICAO AIRCRAFT_NUMBER
createParameterType	→	ICAO CALLSIGN SQUAWK EMERGENCY ALERT SPI
recordingParameter	→	REC_DURATION ALT_DURATION REC_NBR_AIRCRAFT
constant	→	\$name_constant