



## Module Ops.Auftrag

### Global(s)

#### Global

#### AUFTAG

## The AUFTAG Concept

The AUFTAG class significantly simplifies the workflow of using DCS tasks.

#### #AUFTAG AUFTAG

*A warrior's mission is to foster the success of others. -- Morihei Ueshiba*

## The AUFTAG Concept

The AUFTAG class significantly simplifies the workflow of using DCS tasks.

You can think of an AUFTAG as document, which contains the mission briefing, i.e. information about the target location, mission altitude, speed and various other parameters. This document can be handed over directly to a pilot (or multiple pilots) via the Ops.FlightGroup#FLIGHTGROUP class. The pilots will then execute the mission.

The AUFTAG document can also be given to an AIRWING. The airwing will then determine the best assets (pilots and payloads) available for the job.



One more up the food chain, an AUFTRAG can be passed to a COMMANDER. The commander will recruit the best assets of AIRWINGs, BRIGADEs and/or FLEETs and pass the job over to it.

# Airborne Missions

Several mission types are supported by this class.

## Anti-Ship

An anti-ship mission can be created with the [AUFTRAG.NewANTISHIP\(\)](#) function.

## AWACS

An AWACS mission can be created with the [AUFTRAG.NewAWACS\(\)](#) function.

## BAI

A BAI mission can be created with the [AUFTRAG.NewBAI\(\)](#) function.

## Bombing

A bombing mission can be created with the [AUFTRAG.NewBOMBING\(\)](#) function.

## Bombing Runway

A bombing runway mission can be created with the [AUFTRAG.NewBOMBRUNWAY\(\)](#) function.

## Bombing Carpet

A carpet bombing mission can be created with the [AUFTRAG.NewBOMBCARPET\(\)](#) function.

## Strafing

A strafing mission can be created with the [AUFTRAG.NewSTRAFING\(\)](#) function.

## CAP

A CAP mission can be created with the [AUFTRAG.NewCAP\(\)](#) function.



## **Escort**

An escort mission can be created with the [AUFTRAG.NewESCORT\(\)](#) function.

## **FACA**

An FACA mission can be created with the [AUFTRAG.NewFACA\(\)](#) function.

## **Ferry**

Not implemented yet.

## **Ground Escort**

An escort mission can be created with the [AUFTRAG.NewGROUNDESCORT\(\)](#) function.

## **Intercept**

An intercept mission can be created with the [AUFTRAG.NewINTERCEPT\(\)](#) function.

## **Orbit**

An orbit mission can be created with the [AUFTRAG.NewORBIT\(\)](#) function.

## **GCICAP**

An patrol mission can be created with the [AUFTRAG.NewGCICAP\(\)](#) function.

## **RECON**

An reconnaissance mission can be created with the [AUFTRAG.NewRECON\(\)](#) function.

## **RESCUE HELO**

An rescue helo mission can be created with the [AUFTRAG.NewRESCUEHELO\(\)](#) function.

## **SEAD**

An SEAD mission can be created with the [AUFTRAG.NewSEAD\(\)](#) function.



## Tanker

A refueling tanker mission can be created with the [AUFTRAG.NewTANKER\(\)](#) function.

## TROOPTRANSPORT

A troop transport mission can be created with the [AUFTRAG.NewTROOPTRANSPORT\(\)](#) function.

## CARGOTRANSPORT

A cargo transport mission can be created with the [AUFTRAG.NewCARGOTTRANSPORT\(\)](#) function.

## HOVER

A mission for a helicoptre or VSTOL plane to Hover at a point for a certain amount of time can be created with the [AUFTRAG.NewHOVER\(\)](#) function.

## Ground Missions

### ARTY

An arty mission can be created with the [AUFTRAG.NewARTY\(\)](#) function.

### GROUNDATTACK

A ground attack mission can be created with the [AUFTRAG.NewGROUNDATTACK\(\)](#) function.

## Assigning Missions

An AUFTRAG can be assigned to groups (FLIGHTGROUP, ARMYGROUP, NAVYGROUP), legions (AIRWING, BRIGADE, FLEET) or to a COMMANDER.

### Group Level

#### Flight Group

Assigning an AUFTRAG to a flight group is done via the [Ops.FlightGroup#FLIGHTGROUP.AddMission](#) function. See FLIGHTGROUP docs for details.



## Navy Group

Assigning an AUFTRAG to a navy group is done via the [Ops.NavyGroup#NAVYGROUP.AddMission](#) function. See NAVYGROUP docs for details.

## Legion Level

Adding an AUFTRAG to an airwing is done via the [Ops.Airwing#AIRWING.AddMission](#) function. See AIRWING docs for further details. Similarly, an AUFTRAG can be added to a brigade via the [Ops.Brigade#BRIGADE.AddMission](#) function.

## Commander Level

Assigning an AUFTRAG to a commander is done via the [Ops.Commander#COMMANDER.AddMission](#) function. The commander will select the best assets available from all the legions under his command. See COMMANDER docs for details.

## Chief Level

Assigning an AUFTRAG to a commander is done via the [Ops.Chief#CHIEF.AddMission](#) function. The chief will simply pass on the mission to his/her commander.

## Transportation

TODO

## Events

The AUFTRAG class creates many useful (FSM) events, which can be used in the mission designers script.

TODO

## Examples

TODO



## #number **\_AUFTRAGSNR**

Global mission counter.

# Type(s)

## Type AUFTRAG

**AUFTRAG** , extends **Core.Fsm#FSM**

### Fields and Methods inherited from AUFTRAG

#### Description

[AUFTRAG:AddAsset\(Asset\)](#)

Add asset to mission.

[AUFTRAG:AddConditionFailure\(ConditionFunction, ...\)](#)

Add failure condition.

[AUFTRAG:AddConditionPush\(ConditionFunction,....\)](#)

Add push condition.

[AUFTRAG:AddConditionStart\(ConditionFunction,....\)](#)

Add start condition.



Add SUCCESS condition.

#### AUFTRAG:AddLegion(Legion)

Add LEGION to mission.

#### AUFTRAG:AddOpsGroup(OpsGroup)

Add a Ops group to the mission.

#### AUFTRAG:AddRequiredPayload(Payload)

Add a required payload for this mission.

#### AUFTRAG:AddTransportCarriers(Carriers)

Add carriers for a transport of mission assets.

#### AUFTRAG:AssignCohort(Cohort)

**[LEGION, COMMANDER, CHIEF]** Assign a legion cohort to the mission.

#### AUFTRAG:AssignEscortCohort(Cohort)

**[LEGION, COMMANDER, CHIEF]** Assign an escort cohort.

#### AUFTRAG:AssignEscortLegion(Legion)

**[LEGION, COMMANDER, CHIEF]** Add an escort Legion.



**[LEGION, COMMANDER, CHIEF]** Assign a legion to the mission.

AUFTRAG:AssignSquadrons(Squadrons).

**[LEGION, COMMANDER, CHIEF]** Assign airwing squadron(s) to the mission.

AUFTRAG:AssignTransportCohort(Cohort).

**[LEGION, COMMANDER, CHIEF]** Assign a transport cohort.

AUFTRAG:AssignTransportLegion(Legion).

**[LEGION, COMMANDER, CHIEF]** Assign a transport Legion.

AUFTRAG:Cancel().

Triggers the FSM event "Cancel".

AUFTRAG.Category

AUFTRAG:CheckGroupsDone().

Check if all groups are done with their mission (or dead).

AUFTRAG.CheckMissionCapability(MissionTypes, Capabilities, All).

Check if a mission type is contained in a list of possible capabilities.

AUFTRAG.CheckMissionCapabilityAll(MissionTypes, Capabilities).



#### AUFTRAG.CheckMissionType(MissionType, Capabilities).

Check if a mission type is contained in a list of possible capabilities.

#### AUFTRAG.CheckMissionType(MissionType, PossibleTypes).

Checks if a mission type is contained in a table of possible types.

#### AUFTRAG.ClassName

Name of the class.

#### AUFTRAG:CountMissionTargets()

Count alive mission targets.

#### AUFTRAG:CountOpsGroups()

Count alive OPS groups assigned for this mission.

#### AUFTRAG:CountOpsGroupsInStatus(Status)

Count OPS groups in a certain status.

#### AUFTRAG.DCStask

DCS task structure.

#### AUFTRAG:DelAsset(Asset)



#### AUFTRAG:DeleteOpsGroup(OpsGroup).

Remove an Ops group from the mission.

#### AUFTRAG:Done().

Triggers the FSM event "Done".

#### AUFTRAG:EvalConditionsAll(Conditions).

Check if all given condition are true.

#### AUFTRAG:EvalConditionsAny(Conditions).

Check if any of the given conditions is true.

#### AUFTRAG:Evaluate().

Evaluate mission outcome - success or failure.

#### AUFTRAG:Executing().

Triggers the FSM event "Executing".

#### AUFTRAG:Failed().

Triggers the FSM event "Failed".

#### AUFTRAG:GetAssetByName(Name).



AUFTRAG:GetAssetDataTable(AssetName).

Get asset data table.

AUFTRAG:GetCasualties()

Get casualties, *i.e.* number of own units that died during this mission.

AUFTRAG:GetDCSMissionTask()

Get DCS task table for the given mission.

AUFTRAG:GetGroupData(opsgroup)

Get flight data table.

AUFTRAG:GetGroupEgressWaypointUID(opsgroup)

Get Egress waypoint UID of OPS group.

AUFTRAG:GetGroupStatus(opsgroup)

Get ops group mission status.

AUFTRAG:GetGroupWaypointCoordinate(opsgroup)

Get mission (ingress) waypoint coordinate of OPS group

AUFTRAG:GetGroupWaypointIndex(opsgroup)

AUFTRAG:GetGroupWaypointTask(OPSgroup).

Get mission waypoint task of OPS group.

AUFTRAG:GetImportance().

Get mission importance.

AUFTRAG:GetKills().

Get kills, i.e.

AUFTRAG:GetLegionStatus(Legion).

Get LEGION mission status.

AUFTRAG:GetMissionEgressCoord().

Get the mission egress coordinate if this was defined.

AUFTRAG:GetMissionHoldingCoord().

Get the mission holding coordinate if this was defined.

AUFTRAG:GetMissionIngressCoord().

Get the mission ingress coordinate if this was defined.

AUFTRAG:GetMissionTaskforMissionType(MissionType).

AUFTRAG:GetMissionWaypointTypeList(MissionTypeTypes)

Get coordinate of target.

AUFTRAG:GetMissionWaypointCoord(group, randomradius, surfacetypes)

Get coordinate of target.

AUFTRAG:GetName()

Get mission name.

AUFTRAG:GetNumberOfRequiredAssets()

Get number of required assets.

AUFTRAG:GetObjectve(RefCoordinate, Coalitions)

Get mission objective object.

AUFTRAG:GetOpsGroups()

Get all OPS groups.

AUFTRAG:GetOpsTransport()

Get the attach OPS transport of the mission.

AUFTRAG:GetPriority()



#### AUFTRAG:GetRequiredAssets()

**[LEGION, COMMANDER, CHIEF]** Get number of required assets.

#### AUFTRAG:GetTargetCoordinate()

Get coordinate of target.

#### AUFTRAG:GetTargetDamage()

Get target damage.

#### AUFTRAG:GetTargetData()

Get target.

#### AUFTRAG:GetTargetDistance(FromCoord)

Get distance to target.

#### AUFTRAG:GetTargetHeading()

Get heading of target.

#### AUFTRAG:GetTargetInitialLife()

Get target life points.

#### AUFTRAG:GetTargetInitialNumber()



#### AUFTRAG.GetTargetLife()

Get target life points.

#### AUFTRAG:GetTargetName()

Get name of the target.

#### AUFTRAG:GetTargetType()

Get type of target.

#### AUFTRAG:GetTargetVec2()

Get 2D vector of target.

#### AUFTRAG:GetType()

Get mission type.

#### AUFTRAG.GroupStatus

#### AUFTRAG:IsAircraft()

Check if mission is for aircraft (airplanes and/or helicopters).

#### AUFTRAG:IsAirplane()

Check if mission is for airplanes.



~~CHECK IF MISSION WAS CANCELLED.~~

#### AUFTRAG:IsDone()

Check if mission is done.

#### AUFTRAG:IsExecuting(AllGroups)

Check if mission is EXECUTING.

#### AUFTRAG:IsGround()

Check if mission is for ground units.

#### AUFTRAG:IsHelicopters()

Check if mission is for helicopters.

#### AUFTRAG:IsNaval()

Check if mission is for naval units.

#### AUFTRAG: IsNotOver()

Check if mission is NOT over.

#### AUFTRAG: IsOver()

Check if mission is over.



CHECK IF MISSION IS PLANNED.

AUFTRAG:IsQueued(Legion)

Check if mission is QUEUED at a LEGION mission queue.

AUFTRAG:IsReadyToCancel()

Check if mission is ready to be cancelled.

AUFTRAG:IsReadyToGo()

Check if mission is ready to be started.

AUFTRAG:IsReadyToPush()

Check if mission is ready to be pushed.

AUFTRAG:IsRequired(Legion)

Check if mission is REQUESTED.

AUFTRAG:IsScheduled()

Check if mission is SCHEDULED.

AUFTRAG:IsStarted()

Check if mission is STARTED.



~~CHECK IF MISSION WAS A SUCCESS.~~

#### AUFTRAG:IsUrgent()

Check if mission is "urgent".

#### AUFTRAG.Nassets

Number of requested warehouse assets.

#### AUFTRAG.NassetsLegMax

Number of required warehouse assets for each assigned legion.

#### AUFTRAG.NassetsLegMin

Number of required warehouse assets for each assigned legion.

#### AUFTRAG.NassetsMax

Max. number of required warehouse assets.

#### AUFTRAG.NassetsMin

Min. number of required warehouse assets.

#### AUFTRAG.Nassigned

Number of assigned groups.



Max number of required carrier assets.

#### AUFTRAG.NcarriersMin

Min number of required carrier assets.

#### AUFTRAG.Ncasualties

Number of own casualties during mission.

#### AUFTRAG.Ndead

Number of assigned groups that are dead.

#### AUFTRAG.Nelements

Number of elements (units) assigned to mission.

#### AUFTRAG.NescortMax

Max. number of required escort assets for each group the mission is assigned to.

#### AUFTRAG.NescortMin

Min. number of required escort assets for each group the mission is assigned to.

#### AUFTRAG:New(Type)

Create a new generic AUFTRAG object.



**[GROUND, NAVAL]** Create an ALERT 1 mission.

AUFTRAG:NewALERT5(MissionType).

**[AIR]** Create an ALERT 5 mission.

AUFTRAG:NewAMMOSUPPLY(Zone).

**[GROUND]** Create a AMMO SUPPLY mission.

AUFTRAG:NewANTISHIP(Target, Altitude).

**[AIR]** Create an ANTI-SHIP mission.

AUFTRAG:NewARMORATTACK(Target, Speed, Formation).

**[OBSOLETE]** Create a ARMORATTACK mission.

AUFTRAG:NewARMOREDGUARD(Coordinate, Formation).

**[GROUND]** Create an ARMORED ON GUARD mission.

AUFTRAG:NewARTY(Target, Nshots, Radius, Altitude).

**[GROUND, NAVAL]** Create an ARTY mission ("Fire at point" task).

AUFTRAG:NewAUTO(EngageGroup).

Create a mission to attack a group.



**[AIR]** Create a AVALO mission.

AUFTRAG:NewBAI(Target, Altitude)

**[AIR]** Create a BAI mission.

AUFTRAG:NewBARRAGE(Zone, Heading, Angle, Radius, Altitude, Nshots)

**[GROUND, NAVAL]** Create an BARRAGE mission.

AUFTRAG:NewBOMBCARPET(Target, Altitude, CarpetLength)

**[AIR]** Create a CARPET BOMBING mission.

AUFTRAG:NewBOMBING(Target, Altitude, EngageWeaponType)

**[AIR]** Create a BOMBING mission.

AUFTRAG:NewBOMBRUNWAY(Airdrome, Altitude)

**[AIR]** Create a BOMBRUNWAY mission.

AUFTRAG:NewCAP(ZoneCAP, Altitude, Speed, Coordinate, Heading, Leg, TargetTypes)

**[AIR]** Create a CAP mission.

AUFTRAG:NewCAPGROUP(Grp, Altitude, Speed, RelHeading, Leg, OffsetDist, OffsetAngle, UpdateDistance, TargetTypes, EngageRange)

**[AIR]** Create a CAP mission over a (moving) group.



**[AIR, GROUND, NAVAL]** Create a CAPTUREZONE mission.

AUFTRAG:NewCARGOTRANSPORT(StaticCargo, DropZone).

**[AIR ROTARY]** Create a CARGO TRANSPORT mission.

AUFTRAG:NewCAS(ZoneCAS, Altitude, Speed, Coordinate, Heading, Leg, TargetTypes).

**[AIR]** Create a CAS mission.

AUFTRAG:NewCASENHANCED(CasZone, Altitude, Speed, RangeMax, NoEngageZoneSet, TargetTypes).

**[AIR]** Create a CASENHANCED mission.

AUFTRAG:NewESCORT(EscortGroup, OffsetVector, EngageMaxDistance, TargetTypes).

**[AIR]** Create an ESCORT (or FOLLOW) mission.

AUFTRAG:NewEWR(Zone).

**[GROUND]** Create an EWR mission.

AUFTRAG:NewFAC(FacZone, Speed, Altitude, Frequency, Modulation).

**[AIR, GROUND]** Create a FAC mission.

AUFTRAG:NewFACA(Target, Designation, DataLink, Frequency, Modulation).

**[AIR]** Create a FACA mission.



**[GROUND]** Create a FULL SUPPORT MISSION.

AUFTRAG:NewFromTarget(Target, MissionType).

Create a mission to attack a TARGET object.

AUFTRAG:NewGCICAP(Coordinate, Altitude, Speed, Heading, Leg).

**[AIR]** Create a Ground Controlled CAP (GCICAP) mission.

AUFTRAG:NewGROUNDATTACK(Target, Speed, Formation).

**[GROUND]** Create a GROUNDATTACK mission.

AUFTRAG:NewGROUNDESCORT(EscortGroup, OrbitDistance, TargetTypes).

**[AIR/HELO]** Create a GROUNDESCORT (or FOLLOW) mission.

AUFTRAG:NewHOVER(Coordinate, Altitude, Time, Speed, MissionAlt).

**[AIR ROTARY]** Create an HOVER mission.

AUFTRAG:NewINTERCEPT(Target).

**[AIR]** Create an INTERCEPT mission.

AUFTRAG:NewLANDATCOORDINATE(Coordinate, OuterRadius, InnerRadius, Time, Speed, MissionAlt, CombatLanding, DirectionAfterLand).

**[AIR ROTARY]** Create an LANDATCOORDINATE mission.



**[GROUND, NAVAL]** Create a mission to do nothing.

AUFTRAG:NewONGUARD(Coordinate)

**[GROUND, NAVAL]** Create an ON GUARD mission.

AUFTRAG:NewORBIT(Coordinate, Altitude, Speed, Heading, Leg)

**[AIR]** Create an ORBIT mission, which can be either a circular orbit or a race-track pattern.

AUFTRAG:NewORBIT\_CIRCLE(Coordinate, Altitude, Speed)

**[AIR]** Create an ORBIT mission, where the aircraft will go in a circle around the specified coordinate.

AUFTRAG:NewORBIT\_GROUP(Group, Altitude, Speed, Leg, Heading, OffsetVec2, Distance)

**[AIR]** Create an ORBIT mission, where the aircraft will fly a circular or race-track pattern over a given group or unit.

AUFTRAG:NewORBIT\_RACETRACK(Coordinate, Altitude, Speed, Heading, Leg)

**[AIR]** Create an ORBIT mission, where the aircraft will fly a race-track pattern.

AUFTRAG:NewPATROLZONE(Zone, Speed, Altitude, Formation)

**[AIR, GROUND, NAVAL]** Create a PATROLZONE mission.

AUFTRAG:NewPATROL\_RACETRACK(Coordinate, Altitude, Speed, Heading, Leg, Formation)

**[AIR]** Create an enhanced orbit race track mission.



**[GROUND]** Create a REARMING mission.

AUFTRAG:NewRECON(ZoneSet, Speed, Altitude, Adinfinitum, Randomly, Formation)

**[AIR, GROUND, NAVAL]** Create a RECON mission.

AUFTRAG:NewRECOVERYTANKER(Carrier, Altitude, Speed, Leg, RelHeading, OffsetDist, OffsetAngle, UpdateDistance)

**[AIRPANE]** Create a RECOVERY TANKER mission.

AUFTRAG:NewRESCUEHELO(Carrier)

**[AIR ROTARY]** Create a RESCUE HELO mission.

AUFTRAG:NewSEAD(Target, Altitude)

**[AIR]** Create a SEAD mission.

AUFTRAG:NewSTRAFING(Target, Altitude, Length)

**[AIR]** Create a STRAFING mission.

AUFTRAG:NewSTRIKE(Target, Altitude, EngageWeaponType)

**[AIR]** Create a STRIKE mission.

AUFTRAG:NewTANKER(Coordinate, Altitude, Speed, Heading, Leg, RefuelSystem)

**[AIR]** Create a TANKER mission.



**[AIR ROTARY, GROUND]** Create a TROOP TRANSPORT mission.

AUFTRAG.Ngroups

AUFTRAG.Nkills

Number of (enemy) units killed by assets of this mission.

AUFTRAG.Nrepeat

Number of times the mission is repeated.

AUFTRAG.NrepeatFailure

Number of times mission is repeated if failed.

AUFTRAG.NrepeatSuccess

Number of times mission is repeated if successful.

AUFTRAG:OnAfterCancel(From, Event, To)

On after "Cancel" event.

AUFTRAG:OnAfterDone(From, Event, To)

On after "Done" event.



On after "Executing" event.

#### AUFTRAG:OnAfterFailed(From, Event, To)

On after "Failed" event.

#### AUFTRAG:OnAfterPlanned(From, Event, To)

On after "Planned" event.

#### AUFTRAG:OnAfterQueued(From, Event, To)

On after "Queued" event.

#### AUFTRAG:OnAfterRepeat(From, Event, To)

On after "Repeat" event.

#### AUFTRAG:OnAfterRequested(From, Event, To)

On after "Requested" event.

#### AUFTRAG:OnAfterScheduled(From, Event, To)

On after "Scheduled" event.

#### AUFTRAG:OnAfterStarted(From, Event, To)

On after "Started" event.



ON AFTER SUCCESS EVENT.

#### AUFTRAG:OnEventUnitLost(EventData)

Unit lost event.

#### AUFTRAG:Planned()

Triggers the FSM event "Planned".

#### AUFTRAG:Queued()

Triggers the FSM event "Queued".

#### AUFTRAG:RemoveLegion(Legion)

Remove LEGION from mission.

#### AUFTRAG:Repeat()

Triggers the FSM event "Repeat".

#### AUFTRAG:Requested()

Triggers the FSM event "Requested".

#### AUFTRAG:Scheduled()

Triggers the FSM event "Scheduled".



Set alarm state for this mission.

#### AUFTRAG:SetAllowAfterburner()

Set that (jet) aircraft are generally allowed to use afterburner.

#### AUFTRAG:SetAllowAfterburnerExecutePhase()

Set that (jet) aircraft are allowed to use afterburner in mission execution phase.

#### AUFTRAG:SetAssetsStayAlive(Switch)

**[LEGION, COMMANDER, CHIEF]** Set that only alive (spawned) assets are considered.

#### AUFTRAG:SetDuration(Duration)

Set time how long the mission is executed.

#### AUFTRAG:SetEPLRS(OnOffSwitch)

Set EPLRS datalink setting for this mission.

#### AUFTRAG:SetEmission(OnOffSwitch)

Set emission setting for this mission.

#### AUFTRAG:SetEnableMarkers(Coalition)

Enable markers, which display the mission status on the F10 map.



Set engage attribute.

#### AUFTRAG:SetEngageAsGroup(Switch)

Set whether target will be attack as group.

#### AUFTRAG:SetEngageDetected(RangeMax, TargetTypes, EngageZoneSet, NoEngageZoneSet)

Enable to automatically engage detected targets.

#### AUFTRAG:SetEvaluationTime(Teval)

Set time interval between mission done and success/failure evaluation.

#### AUFTRAG:SetFormation(Formation)

Set formation for this mission.

#### AUFTRAG:SetGroupEgressWaypointUID(opsgroup, waypointindex)

Set Egress waypoint UID for OPS group.

#### AUFTRAG:SetGroupStatus(opsgroup, status)

Set opsgroup mission status.

#### AUFTRAG:SetGroupWaypointCoordinate(opsgroup, coordinate)

Set mission (ingress) waypoint coordinate for OPS group.



Set mission (ingress) waypoint ID for OPS group.

AUFTRAG:SetGroupWaypointTask(opsgroup, task)

Set mission waypoint task for OPS group.

AUFTRAG:SetICLS(Channel, Morse, UnitName)

Set ICLS beacon channel and Morse code for this mission.

AUFTRAG:SetImmortal(OnOffSwitch)

Set immortality setting for this mission.

AUFTRAG:SetIngressCoordinate(coordinate)

[Air] Set mission (ingress) waypoint coordinate for FLIGHT group.

AUFTRAG:SetInvisible(OnOffSwitch)

Set invisibility setting for this mission.

AUFTRAG:SetLegionStatus(Legion, Status)

Set LEGION mission status.

AUFTRAG:SetMissionAltitude(Altitude)

Set mission altitude.



Set the mission egress coordinate.

AUFTRAG:SetMissionHoldingCoord(Coordinate, Altitude, Speed, Duration)

[Air] Set the mission holding coordinate.

AUFTRAG:SetMissionIngressCoord(Coordinate, Altitude, Speed)

[Air] Set the mission ingress coordinate.

AUFTRAG:SetMissionRange(Range)

Set max mission range.

AUFTRAG:SetMissionSpeed(Speed)

Set mission speed.

AUFTRAG:SetMissionWaypointCoord(Coordinate)

[NON-AIR] Set the mission waypoint coordinate from where the mission is executed.

AUFTRAG:SetMissionWaypointRandomization(Radius)

Set randomization of the mission waypoint coordinate.

AUFTRAG:SetName(Name)

Set mission name.



ATTACH OPS TRANSPORT TO THE MISSION.

#### AUFTRAG:SetPriority(Prio, Urgent, Importance).

Set mission priority and (optional) urgency.

#### AUFTRAG:SetProhibitAfterburner().

Set that (jet) aircraft are generally **not** allowed to use afterburner.

#### AUFTRAG:SetProhibitAfterburnerExecutePhase().

Set that (jet) aircraft are **not** allowed to use afterburner in mission execution phase.

#### AUFTRAG:SetPushTime(ClockPush).

Set mission push time.

#### AUFTRAG:SetROE(roe).

Set Rules of Engagement (ROE) for this mission.

#### AUFTRAG:SetROT(rot)

Set Reaction on Threat (ROT) for this mission.

#### AUFTRAG:SetRadio(Frequency, Modulation).

Set radio frequency and modulation for this mission.



**[LEGION, COMMANDER, CHIEF]** Set how many mission assets get reinforced in their number groups below the minimum number of required assets of the mission (c.f. SetRequiredAssets() function).

#### AUFTRAG:SetRepeat(Nrepeat)

**[LEGION, COMMANDER, CHIEF]** Set how many times the mission is repeated.

#### AUFTRAG:SetRepeatOnFailure(Nrepeat)

**[LEGION, COMMANDER, CHIEF]** Set how many times the mission is repeated if it fails.

#### AUFTRAG:SetRepeatOnSuccess(Nrepeat)

**[LEGION, COMMANDER, CHIEF]** Set how many times the mission is repeated if it was successful.

#### AUFTRAG:SetRequiredAssets(NassetsMin, NassetsMax)

**[LEGION, COMMANDER, CHIEF]** Define how many assets are required to do the job.

#### AUFTRAG:SetRequiredAttribute(Attributes)

**[LEGION, COMMANDER, CHIEF]** Set required attribute(s) the assets must have.

#### AUFTRAG:SetRequiredCarriers(NcarriersMin, NcarriersMax, Categories, Attributes, Properties)

**[LEGION, COMMANDER, CHIEF]** Set number of required carrier groups if an OPSTRANSPORT assignment is required.

#### AUFTRAG:SetRequiredEscorts(NescortMin, NescortMax, MissionType, TargetTypes, EngageRange)



#### AUFTRAG:SetRequiredProperty(Properties)

**[LEGION, COMMANDER, CHIEF]** Set required property or properties the assets must have.

#### AUFTRAG:SetRequiredTransport(DeployZone, NcarriersMin, NcarriersMax, DisembarkZone, Categories, Attributes, Properties)

**[LEGION, COMMANDER, CHIEF]** Attach OPS transport to the mission.

#### AUFTRAG:SetReturnToLegion(Switch)

**[LEGION, COMMANDER, CHIEF]** Set whether assigned assets return to their legion once the mission is over.

#### AUFTRAG:SetTACAN(Channel, Morse, UnitName, Band)

Set TACAN beacon channel and Morse code for this mission.

#### AUFTRAG:SetTeleport(Switch)

Set that mission assets are teleported to the mission execution waypoint.

#### AUFTRAG:SetTime(ClockStart, ClockStop)

Set mission start and stop time.

#### AUFTRAG:SetVerbosity(VerbosityLevel)

Set verbosity level.



Set number of weapons to engage.

#### AUFTRAG:SetWeaponType(WeaponType)

Set weapon type used for the engagement.

#### AUFTRAG.SpecialTask

#### AUFTRAG:Started()

Triggers the FSM event "Started".

#### AUFTRAG:Status()

Triggers the FSM event "Status".

#### AUFTRAG:Stop()

Triggers the FSM event "Stop".

#### AUFTRAG:Success()

Triggers the FSM event "Success".

#### AUFTRAG.TargetType

#### AUFTRAG.Texecuting

Time stamp (abs) when mission is executing. Is #nil on start.



Mission abs. time stamp, when mission was over.

AUFTRAG.Tpush

Mission push/execute time in abs. seconds.

AUFTRAG.TrackAltitude

AUFTRAG.TrackFormation

AUFTRAG.TrackPoint1

AUFTRAG.TrackPoint2

AUFTRAG.TrackSpeed

AUFTRAG.Tstart

Mission start time in abs. seconds.

AUFTRAG.Tstarted

Time stamp (abs) when mission is started.

AUFTRAG.Tstop

Mission stop time in abs. seconds.



### AUFTRAG:UpdateMarker()

Update mission F10 map marker.

### AUFTRAG: AddAssets(Assets)

Add assets to mission.

### AUFTRAG: DetermineAuftragType(Target)

Create a mission to attack a group.

### AUFTRAG: GetDCSAttackTask(Target, DCStasks)

Get DCS task table for an attack group or unit task.

### AUFTRAG: GetMissionWaypointCoordSet()

Get coordinate which was set as mission waypoint coordinate.

### AUFTRAG: GetRequest(Legion)

Get request from legion this mission requested assets from.

### AUFTRAG: GetRequestID(Legion)

Get request ID from legion this mission requested assets from

### AUFTRAG: IsNotReinforcing()

AUFTAG: IsReinforcing()

Check if reinforcement is still ongoing.

AUFTAG: NewRELOCATECOHORT(Legion, Cohort)

**[PRIVATE, AIR, GROUND, NAVAL]** Create a mission to relocate all cohort assets to another LEGION.

AUFTAG: SetLogID()

Set log ID string.

AUFTAG: SetRequestID(Legion, RequestID)

Set request ID from legion this mission requested assets from

AUFTAG: TargetFromObject(Object)

Create target data from a given object.

AUFTAG: Cancel(delay)

Triggers the FSM event "Cancel" after a delay.

AUFTAG: Done(delay)

Triggers the FSM event "Done" after a delay.

AUFTAG: Executing(delay)



AUFTAG: Failed(delay).

Triggers the FSM event "Failed" after a delay.

AUFTAG: Planned(delay).

Triggers the FSM event "Planned" after a delay.

AUFTAG: Queued(delay).

Triggers the FSM event "Queued" after a delay.

AUFTAG: Repeat(delay).

Triggers the FSM event "Repeat" after a delay.

AUFTAG: Requested(delay).

Triggers the FSM event "Requested" after a delay.

AUFTAG: Scheduled(delay).

Triggers the FSM event "Scheduled" after a delay.

AUFTAG: Started(delay).

Triggers the FSM event "Started" after a delay.

AUFTAG: Status(delay).



#### AUFTRAG\_Stop(delay)

Triggers the FSM event "Stop" after a delay.

#### AUFTRAG: Success(delay)

Triggers the FSM event "Success" after a delay.

#### AUFTRAG.alert5MissionType

Alert 5 mission type. This is the mission type, the alerted assets will be able to carry out.

#### AUFTRAG.artyAltitude

Altitude in meters. Can be used for a Barrage.

#### AUFTRAG.artyAngle

Shooting angle in degrees (for Barrage).

#### AUFTRAG.artyHeading

Heading in degrees (for Barrage).

#### AUFTRAG.artyRadius

Radius in meters.

#### AUFTRAG.artyShots



#### AUFTRAG.assets

Warehouse assets assigned for this mission.

#### AUFTRAG.attributes

Generalized attribute(s) of assets.

#### AUFTRAG.auftragsnummer

Auftragsnummer.

#### AUFTRAG.carrierAttributes

Generalized attribute(s) of transport assets.

#### AUFTRAG.carrierCategories

#### AUFTRAG.carrierProperties

DCS attribute(s) of transport assets.

#### AUFTRAG.categories

Mission categories.

#### AUFTRAG.chief



### AUFTRAG.commander

The COMMANDER managing this mission.

### AUFTRAG.conditionFailure

If all conditions are true, the mission is cancelled.

### AUFTRAG.conditionFailureSet

### AUFTRAG.conditionPush

If all conditions are true, the mission is executed. Before, the group(s) wait at the mission execution waypoint.

### AUFTRAG.conditionStart

Condition(s) that have to be true, before the mission will be started.

### AUFTRAG.conditionSuccess

If all conditions are true, the mission is cancelled.

### AUFTRAG.conditionSuccessSet

### AUFTRAG.dTevaluate

Time interval in seconds before the mission result is evaluated after mission is over.



MISSION DURATION IN SECONDS.

AUFTRAG.durationExe

Mission execution time in seconds.

AUFTRAG.engageAltitude

Engagement altitude in meters.

AUFTRAG.engageAsGroup

Group attack.

AUFTRAG.engageDirection

Engagement direction in degrees.

AUFTRAG.engageLength

Length of engage (carpet or strafing) in meters.

AUFTRAG.engageMaxDistance

Max engage distance.

AUFTRAG.engageQuantity

Number of times a target is engaged.



### AUFTRAG.engageTarget

Target data to engage.

### AUFTRAG.engageTargetTypes

Table of target types that are engaged in the engagement zone.

### AUFTRAG.engageWeaponExpend

How many weapons are used.

### AUFTRAG.engageWeaponType

Weapon type used.

### AUFTRAG.engageZone

*Circular* engagement zone.

### AUFTRAG.engageDetectedEngageZones

### AUFTRAG.engageDetectedNoEngageZones

### AUFTRAG.engageDetectedOn

### AUFTRAG.engageDetectedRmax



### AUFTRAG.enrouteTasks

Mission enroute tasks.

### AUFTRAG.escortCohorts

Cohorts explicitly requested for providing escorting assets.

### AUFTRAG.escortEngageRange

Engage range in nautical miles (NM).

### AUFTRAG.escortGroup

The group to be escorted.

### AUFTRAG.escortGroupName

Name of the escorted group.

### AUFTRAG.escortLegions

Legions explicitly requested for providing escorting assets.

### AUFTRAG.escortMissionType

Escort mission type.

### AUFTRAG.escortTargetTypes



#### AUFTRAG.escortVecs

The 3D offset vector from the escorted group to the escort group.

#### AUFTRAG.facDatalink

FAC datalink enabled.

#### AUFTRAG.facDesignation

FAC designation type.

#### AUFTRAG.facFreq

FAC radio frequency in MHz.

#### AUFTRAG.facModu

FAC radio modulation 0=AM 1=FM.

#### AUFTRAG.failurecondition

#### AUFTRAG.groupdata

Group specific data.

#### AUFTRAG.hoverAltitude

#### AUFTRAG.hoverTime



ICLS setting.

#### AUFTRAG.importance

Importance.

#### AUFTRAG.legionReturn

If true, assets return to their legion (default). If false, they will stay alive.

#### AUFTRAG.legions

Assigned legions.

#### AUFTRAG.lid

Class id string for output to DCS log file.

#### AUFTRAG.marker

F10 map marker.

#### AUFTRAG.markerCoaliton

Coalition to which the marker is displayed.

#### AUFTRAG.markerOn

If true, display marker on F10 map with the AUFTRAG status.



MISSION altitude in meters.

#### AUFTRAG.missionEgressCoord

Mission egress waypoint coordinate.

#### AUFTRAG.missionEgressCoordAlt

#### AUFTRAG.missionEgressCoordSpeed

#### AUFTRAG.missionFraction

Mission coordinate fraction. Default is 0.5.

#### AUFTRAG.missionHoldingCoord

#### AUFTRAG.missionHoldingCoordAlt

#### AUFTRAG.missionHoldingCoordSpeed

#### AUFTRAG.missionHoldingDuration

#### AUFTRAG.missionIngressCoord

Mission Ingress waypoint coordinate.

#### AUFTRAG.missionIngressCoordAlt



### AUFTRAG.missionRange

Mission range in meters. Used by LEGION classes (AIRWING, BRIGADE, ...).

### AUFTRAG.missionSpeed

Mission speed in km/h.

### AUFTRAG.missionTask

Mission task. See ENUMS.MissionTask.

### AUFTRAG.missionWaypointCoord

Mission waypoint coordinate.

### AUFTRAG.missionWaypointRadius

Random radius in meters.

### AUFTRAG.name

Mission name.

### AUFTRAG:onafterAssetDead(From, Event, To, Asset)

On after "AssetDead" event.

### AUFTRAG:onafterAssign(From, Event, To)

AUFTRAG:onafterCancel(From, Event, To).

On after "Cancel" event.

AUFTRAG:onafterDone(From, Event, To).

On after "Done" event.

AUFTRAG:onafterElementDestroyed(From, Event, To, OpsGroup, Element).

On after "ElementDestroyed" event.

AUFTRAG:onafterExecuting(From, Event, To).

On after "Execute" event.

AUFTRAG:onafterFailed(From, Event, To).

On after "Failed" event.

AUFTRAG:onafterGroupDead(From, Event, To, OpsGroup).

On after "GroupDead" event.

AUFTRAG:onafterPlanned(From, Event, To).

On after "Planned" event.

AUFTRAG:onafterQueued(From, Event, To, Airwing).

AUFTRAG:onafterRepeat(From, Event, To).

On after "Repeat" event.

AUFTRAG:onafterRequested(From, Event, To)

On after "Requested" event.

AUFTRAG:onafterScheduled(From, Event, To)

On after "Schedule" event.

AUFTRAG:onafterStarted(From, Event, To)

On after "Start" event.

AUFTRAG:onafterStatus(From, Event, To)

On after "Status" event.

AUFTRAG:onafterStop(From, Event, To)

On after "Stop" event.

AUFTRAG:onafterSuccess(From, Event, To)

On after "Success" event.

AUFTRAG:onbeforeRepeat(From, Event, To)



#### AUFTRAG.operation

Operation this mission is part of.

#### AUFTRAG.opstransport

OPS transport assignment.

#### AUFTRAG.optionAlarm

Alarm state.

#### AUFTRAG.optionCM

Counter measures.

#### AUFTRAG.optionECM

ECM.

#### AUFTRAG.optionEPLRS

EPLRS datalink.

#### AUFTRAG.optionEmission

Emission is on or off.

#### AUFTRAG.optionFormation



#### AUFTRAG.optionImmortal

Immortal is on/off.

#### AUFTRAG.optionInvisible

Invisible is on/off.

#### AUFTRAG.optionROE

ROE.

#### AUFTRAG.optionROT

ROT.

#### AUFTRAG.optionRTBammo

RTB on out-of-ammo.

#### AUFTRAG.optionRTBfuel

RTB on out-of-fuel.

#### AUFTRAG.orbitAltitude

Orbit altitude in meters.

#### AUFTRAG.orbitDeltaR



#### AUFTRAG.orbitHeading

Orbit heading in degrees.

#### AUFTRAG.orbitLeg

Length of orbit leg in meters.

#### AUFTRAG.orbitOffsetVec2

2D offset vector.

#### AUFTRAG.orbitSpeed

Orbit speed in m/s.

#### AUFTRAG.orbitVec2

2D orbit vector.

#### AUFTRAG.patrolData

Patrol data.

#### AUFTRAG.payloads

User specified airwing payloads for this mission. Only these will be considered for the job!

#### AUFTRAG.prio



#### AUFTRAG.priorityLevel

#### AUFTRAG.prohibitABExecute

#### AUFTRAG.properties

DCS attribute(s) of assets.

#### AUFTRAG.radio

Radio freq and modulation.

#### AUFTRAG.refuelSystem

Refuel type (boom or probe) for TANKER missions.

#### AUFTRAG.reinforce

#### AUFTRAG.repeated

Number of times mission was repeated.

#### AUFTRAG.repeatedFailure

Number of times mission was repeated after a failure.

#### AUFTRAG.repeatedSuccess

Number of times mission was repeated after a success.



The ID of the queued warehouse request. Necessary to cancel the request if the mission was cancelled before the request is processed.

#### AUFTRAG.specialCohorts

User specified cohorts assigned for this mission. Only these will be considered for the job!

#### AUFTRAG.specialLegions

User specified legions assigned for this mission. Only these will be considered for the job!

#### AUFTRAG.status

Mission status.

#### AUFTRAG.statusChief

Mission status of the CHIEF.

#### AUFTRAG.statusCommander

Mission status of the COMMANDER.

#### AUFTRAG.statusLegion

Mission status of all assigned LEGIONS.

#### AUFTRAG.successcondition



TACAN setting.

#### AUFTRAG.targetHeading

Heading of target in degrees.

#### AUFTRAG.teleport

Groups are teleported to the mission ingress waypoint.

#### AUFTRAG.transportCohorts

Cohorts explicitly requested for providing transport carrier assets.

#### AUFTRAG.transportDeployZone

Deploy zone of an OPSTRANSORT.

#### AUFTRAG.transportDisembarkZone

Disembark zone of an OPSTRANSORT.

#### AUFTRAG.transportDropoff

Coordinate where to drop off the cargo.

#### AUFTRAG.transportGroupSet

Groups to be transported.



Regions explicitly requested for providing transport carrier assets.

#### AUFTRAG.transportPickup

Coordinate where to pickup the cargo.

#### AUFTRAG.transportPickupRadius

Radius in meters for pickup zone. Default 500 m.

#### AUFTRAG.type

Mission type.

#### AUFTRAG.updateDCSTask

If true, DCS task is updated at every status update of the assigned groups.

#### AUFTRAG.urgent

Mission is urgent. Running missions with lower prio might be cancelled.

#### AUFTRAG.verbose

Verbosity level.

#### AUFTRAG.version

AUFTRAG class version.



### AUFTRAG:AddEndState(State)

Adds an End state.

### AUFTRAG:AddProcess(From, Event, Process, ReturnEvents)

Set the default #FSM\_PROCESS template with key ProcessName providing the ProcessClass and the process object when it is assigned to a Wrapper.Controllable by the task.

### AUFTRAG:AddScore(State, ScoreText, Score)

Adds a score for the FSM to be achieved.

### AUFTRAG:AddScoreProcess(From, Event, State, ScoreText, Score)

Adds a score for the FSM\_PROCESS to be achieved.

### AUFTRAG:AddTransition(From, Event, To)

Add a new transition rule to the FSM.

### AUFTRAG.CallScheduler

### AUFTRAG.Events

### AUFTRAG:GetCurrentState()

Get current state.



Returns the End states.

---

[AUFTRAG:GetProcess\(From, Event\)](#)

---

[AUFTRAG:GetProcesses\(\)](#)

Returns a table of the SubFSM rules defined within the FSM.

---

[AUFTRAG:GetScores\(\)](#)

Returns a table with the scores defined.

---

[AUFTRAG:GetStartState\(\)](#)

Returns the start state of the FSM.

---

[AUFTRAG:GetState\(\)](#)

Get current state.

---

[AUFTRAG:GetSubs\(\)](#)

Returns a table with the Subs defined.

---

[AUFTRAG:GetTransitions\(\)](#)

Returns a table of the transition rules defined within the FSM.

---

[AUFTRAG:Is\(State\)](#)



#### AUFTRAG.LoadCallbacks(CallBackTable).

Load call backs.

#### AUFTRAG:New()

Creates a new FSM object.

#### AUFTRAG.Scores

#### AUFTRAG:SetProcess(From, Event, Fsm)

#### AUFTRAG:SetStartState(State)

Sets the start state of the FSM.

#### AUFTRAG. EndStates

#### AUFTRAG. EventSchedules

#### AUFTRAG. Processes

#### AUFTRAG. Scores

#### AUFTRAG. StartState

#### AUFTRAG. Transitions



Add to map.

AUFTRAG: call\_handler(step, trigger, params, EventName)

Call handler.

AUFTRAG: create\_transition(EventName)

Create transition.

AUFTRAG: delayed\_transition(EventName)

Delayed transition.

AUFTRAG: eventmap(Events, EventStructure)

Event map.

AUFTRAG: gosub(ParentFrom, ParentEvent)

Go sub.

AUFTRAG: handler(EventName, ...)

Handler.

AUFTRAG: isendstate(Current)

Is end state.

Sub maps.

### AUFTRAG:can(e)

Check if can do an event.

### AUFTRAG:cannot(e)

Check if cannot do an event.

### AUFTRAG.current

### AUFTRAG.endstates

### AUFTRAG:is(State, state)

Check if FSM is in state.

### AUFTRAG.options

### AUFTRAG.subs

## Type AUFTRAG.Capability

### AUFTRAG.Capability

#### Fields and Methods inherited from AUFTRAG.Capability



Type of mission.

#### AUFTRAG.Capability.Performance

Number describing the performance level. The higher the better.

## Type AUFTRAG.Category

### AUFTRAG.Category

#### Fields and Methods inherited from AUFTRAG.Category

##### Description

#### AUFTRAG.Category.AIRCRAFT

Airplanes and helicopters.

#### AUFTRAG.Category.AIRPLANE

Airplanes.

#### AUFTRAG.Category.ALL

#### AUFTRAG.Category.GROUND

Ground troops.



Menüpunkt.

### AUFTRAG.Category.NAVAL

Naval grouse.

## Type AUFTRAG.Condition

### AUFTRAG.Condition

#### Fields and Methods inherited from AUFTRAG.Condition

##### Description

#### AUFTRAG.Condition.arg

Optional arguments passed to the condition callback function.

#### AUFTRAG.Condition.func

Callback function to check for a condition. Should return a #boolean.

## Type AUFTRAG.GroupData

### AUFTRAG.GroupData

#### Fields and Methods inherited from AUFTRAG.GroupData

##### Description



The warehouse asset.

#### AUFTRAG.GroupData.opsgroup

The OPS group.

#### AUFTRAG.GroupData.status

Group mission status.

#### AUFTRAG.GroupData.waypointEgressUID

Egress Waypoint UID.

#### AUFTRAG.GroupData.waypointcoordinate

Ingress waypoint coordinate.

#### AUFTRAG.GroupData.waypointindex

Mission (ingress) Waypoint UID.

#### AUFTRAG.GroupData.waypointtask

Waypoint task.

#### AUFTRAG.GroupData.wpegresscoordinate

Egress waypoint coordinate.



## AUFTRAG.GroupStatus

### Fields and Methods inherited from AUFTRAG.GroupStatus

#### Description

##### AUFTRAG.GroupStatus.CANCELLED

Mission was cancelled.

##### AUFTRAG.GroupStatus.DONE

Mission task of the Ops group is done.

##### AUFTRAG.GroupStatus.EXECUTING

Ops group is executing this mission.

##### AUFTRAG.GroupStatus.PAUSED

Ops group has paused this mission, e.g. for refuelling.

##### AUFTRAG.GroupStatus.SCHEDULED

Mission is scheduled in a FLIGHGROUP queue waiting to be started.

##### AUFTRAG.GroupStatus.STARTED

Ops group started this mission but it is not executed yet.



## AUFTRAG.SpecialTask

### Fields and Methods inherited from AUFTRAG.SpecialTask

#### Description

##### AUFTRAG.SpecialTask.AIRDEFENSE

Air defense.

##### AUFTRAG.SpecialTask.ALERT5

Alert 5 task.

##### AUFTRAG.SpecialTask.AMMOSUPPLY

Ammo Supply.

##### AUFTRAG.SpecialTask.ARMORATTACK

##### AUFTRAG.SpecialTask.ARMOREDGUARD

On guard with armor.

##### AUFTRAG.SpecialTask.BARRAGE

Barrage.

##### AUFTRAG.SpecialTask.CAPTUREZONE



#### AUFTRAG.SpecialTask.EWR

Early Warning Radar.

#### AUFTRAG.SpecialTask.FERRY

Ferry mission.

#### AUFTRAG.SpecialTask.FORMATION

AI formation task.

#### AUFTRAG.SpecialTask.FUELSUPPLY

Fuel Supply.

#### AUFTRAG.SpecialTask.GROUNDATTACK

Ground attack.

#### AUFTRAG.SpecialTask.HOVER

Hover.

#### AUFTRAG.SpecialTask.NOTHING

Nothing.

#### AUFTRAG.SpecialTask.ONGUARD



#### AUFTRAG.SpecialTask.PATROLRACETRACK

Patrol Racetrack.

#### AUFTRAG.SpecialTask.PATROLZONE

Patrol zone task.

#### AUFTRAG.SpecialTask.REARMING

Rearming.

#### AUFTRAG.SpecialTask.RECON

Recon task.

#### AUFTRAG.SpecialTask.RECOVERYTANKER

Recovery tanker.

#### AUFTRAG.SpecialTask.RELOCATECOHORT

Relocate cohort.

## Type AUFTRAG.Status

### AUFTRAG.Status

#### Fields and Methods inherited from AUFTRAG.Status



Mission was cancelled.

AUFTRAG.Status.DONE

Mission is over.

AUFTRAG.Status.EXECUTING

Mission is being executed.

AUFTRAG.Status.FAILED

Mission failed.

AUFTRAG.Status.PLANNED

Mission is at the early planning stage and has not been added to any queue.

AUFTRAG.Status.QUEUED

Mission is queued at a LEGION.

AUFTRAG.Status.REQUESTED

Mission assets were requested from the warehouse.

AUFTRAG.Status.SCHEDULED



#### AUFTAG.Status.STARTED

Mission has started but is not executed yet.

#### AUFTAG.Status.SUCCESS

Mission was a success.

## Type AUFTAG.Success

### AUFTAG.Success

#### Fields and Methods inherited from AUFTAG.Success

##### Description

#### AUFTAG.Success.DAMAGED

Target was damaged.

#### AUFTAG.Success.DESTROYED

Target was destroyed.

#### AUFTAG.Success.ENGAGED

Target was engaged.

#### AUFTAG.Success.SURVIVED



# Type AUFTRAG.TargetData

## AUFTRAG.TargetData

### Fields and Methods inherited from AUFTRAG.TargetData

#### Description

##### AUFTRAG.TargetData.Lifepoints

Total life points.

##### AUFTRAG.TargetData.Lifepoints0

Initial life points.

##### AUFTRAG.TargetData.Name

Target name.

##### AUFTRAG.TargetData.Ninitial

Number of initial targets.

##### AUFTRAG.TargetData.Target

Target Object.

##### AUFTRAG.TargetData.Type



# Type AUFTRAG.TargetType

## AUFTRAG.TargetType

### Fields and Methods inherited from AUFTRAG.TargetType

#### Description

##### AUFTRAG.TargetType.AIRBASE

Target is an AIRBASE.

##### AUFTRAG.TargetType.COORDINATE

Target is a COORDINATE.

##### AUFTRAG.TargetType.GROUP

Target is a GROUP object.

##### AUFTRAG.TargetType.SETGROUP

Target is a SET of GROUPs.

##### AUFTRAG.TargetType.SETUNIT

Target is a SET of UNITS.

##### AUFTRAG.TargetType.STATIC



#### AUFTRAG.Type.UNIT

Target is a UNIT object.

## Type AUFTRAG.Type

### AUFTRAG.Type

#### Fields and Methods inherited from AUFTRAG.Type

##### Description

#### AUFTRAG.Type.AIRDEFENSE

Air defense.

#### AUFTRAG.Type.ALERT5

Alert 5.

#### AUFTRAG.Type.AMMOSUPPLY

Ammo supply.

#### AUFTRAG.Type.ANTISHIP

Anti-ship mission.

#### AUFTRAG.Type.ARMORATTACK



#### AUFTRAG.Type.ARMOREDGUARD

On guard - with armored groups.

#### AUFTRAG.Type.ARTY

Fire at point.

#### AUFTRAG.Type.AWACS

AWACS mission.

#### AUFTRAG.Type.BAI

Battlefield Air Interdiction.

#### AUFTRAG.Type.BARRAGE

Barrage.

#### AUFTRAG.Type.BOMBCARPET

Carpet bombing.

#### AUFTRAG.Type.BOMBING

Bombing mission.

#### AUFTRAG.Type.BOMBRUNWAY



#### AUFTRAG.Type.CAP

Combat Air Patrol.

#### AUFTRAG.Type.CAPTUREZONE

Capture zone mission.

#### AUFTRAG.Type.CARGOTRANSPORT

Cargo transport.

#### AUFTRAG.Type.CAS

Close Air Support.

#### AUFTRAG.Type.CASENHANCED

Enhanced CAS.

#### AUFTRAG.Type.ESCORT

Escort mission.

#### AUFTRAG.Type.EWR

Early Warning Radar.

#### AUFTRAG.Type.FAC



#### AUFTRAG.Type.FACA

Forward AirController airborne mission.

#### AUFTRAG.Type.FERRY

Ferry mission.

#### AUFTRAG.Type.FUELSUPPLY

Fuel supply.

#### AUFTRAG.Type.GCICAP

Similar to CAP but no auto engage targets.

#### AUFTRAG.Type.GROUNDATTACK

Ground attack.

#### AUFTRAG.Type.GROUNDESCORT

Ground escort mission.

#### AUFTRAG.Type.HOVER

Hover.

#### AUFTRAG.Type.INTERCEPT



#### AUFTRAG.Type.LANDATCOORDINATE

Land at coordinate.

#### AUFTRAG.Type.NOTHING

Nothing.

#### AUFTRAG.Type.ONGUARD

On guard.

#### AUFTRAG.Type.OPSTRANSPORT

Ops transport.

#### AUFTRAG.Type.ORBIT

Orbit mission.

#### AUFTRAG.Type.PATROLRACETRACK

Patrol Racetrack.

#### AUFTRAG.Type.PATROLZONE

Patrol a zone.

#### AUFTRAG.Type.REARMING



#### AUFTRAG.Type.RECON

Recon mission.

#### AUFTRAG.Type.RECOVERYTANKER

Recovery tanker.

#### AUFTRAG.Type.RELOCATECOHORT

Relocate a cohort from one legion to another.

#### AUFTRAG.Type.RESCUEHELO

Rescue helo.

#### AUFTRAG.Type.SEAD

Suppression/destruction of enemy air defences.

#### AUFTRAG.Type.STRAFING

Strafing run.

#### AUFTRAG.Type.STRIKE

Strike mission.

#### AUFTRAG.Type.TANKER



#### AUFTAG.Type.TROOP TRANSPORT

Troop transport mission.

## Type Ops.Auftrag

### Type AUFTAG

AUFTAG class.

#### Field(s)

##### #AUFTAG.Category AUFTAG.Category

#string **AUFTAG.ClassName**

Name of the class.

##### #table AUFTAG.DCStask

DCS task structure.

##### #AUFTAG.GroupStatus AUFTAG.GroupStatus

#number **AUFTAG.Nassets**

Number of requested warehouse assets.

##### #table AUFTAG.NassetsLegMax

Number of required warehouse assets for each assigned legion.

##### #table AUFTAG.NassetsLegMin



Max. number of required warehouse assets.

#number **AUFTRAG.NassetsMin**

Min. number of required warehouse assets.

#number **AUFTRAG.Nassigned**

Number of assigned groups.

#number **AUFTRAG.NcarriersMax**

Max number of required carrier assets.

#number **AUFTRAG.NcarriersMin**

Min number of required carrier assets.

#number **AUFTRAG.Ncasualties**

Number of own casualties during mission.

#number **AUFTRAG.Ndead**

Number of assigned groups that are dead.

#number **AUFTRAG.Nelements**

Number of elements (units) assigned to mission.

#number **AUFTRAG.NescortMax**

Max. number of required escort assets for each group the mission is assigned to.

#number **AUFTRAG.NescortMin**

Min. number of required escort assets for each group the mission is assigned to.



Number of (enemy) units killed by assets of this mission.

### #number **AUFTRAG.Nrepeat**

Number of times the mission is repeated.

### #number **AUFTRAG.NrepeatFailure**

Number of times mission is repeated if failed.

### #number **AUFTRAG.NrepeatSuccess**

Number of times mission is repeated if successful.

### #AUFTRAG.SpecialTask **AUFTRAG.SpecialTask**

### #AUFTRAG.TargetType **AUFTRAG.TargetType**

### #number **AUFTRAG.Texecuting**

Time stamp (abs) when mission is executing. Is #nil on start.

### #number **AUFTRAG.Tover**

Mission abs. time stamp, when mission was over.

### #number **AUFTRAG.Tpush**

Mission push/execute time in abs. seconds.

### **AUFTRAG.TrackAltitude**

### **AUFTRAG.TrackFormation**

### **AUFTRAG.TrackPoint1**

### **AUFTRAG.TrackPoint2**

### **AUFTRAG.TrackSpeed**

**#number AUFTRAG.Tstarted**

Time stamp (abs) when mission is started.

**#number AUFTRAG.Tstop**

Mission stop time in abs. seconds.

**#AUFTRAG.Type AUFTRAG.Type****#string AUFTRAG.alert5MissionType**

Alert 5 mission type. This is the mission type, the alerted assets will be able to carry out.

**#number AUFTRAG.artyAltitude**

Altitude in meters. Can be used for a Barrage.

**#number AUFTRAG.artyAngle**

Shooting angle in degrees (for Barrage).

**#number AUFTRAG.artyHeading**

Heading in degrees (for Barrage).

**#number AUFTRAG.artyRadius**

Radius in meters.

**#number AUFTRAG.artyShots**

Number of shots fired.

**AUFTRAG.assetStayAlive****#table AUFTRAG.assets**



Generalized attribute(s) of assets.

#### #number **AUFTRAG.auftragsnummer**

Auftragsnummer.

#### #table **AUFTRAG.carrierAttributes**

Generalized attribute(s) of transport assets.

#### **AUFTRAG.carrierCategories**

#### #table **AUFTRAG.carrierProperties**

DCS attribute(s) of transport assets.

#### #table **AUFTRAG.categories**

Mission categories.

#### Ops.Chief#**CHIEF AUFTRAG.chief**

The CHIEF managing this mission.

#### Ops.Commander#**COMMANDER AUFTRAG.commander**

The COMMANDER managing this mission.

#### #table **AUFTRAG.conditionFailure**

If all conditions are true, the mission is cancelled.

#### #boolean **AUFTRAG.conditionFailureSet**

#### #table **AUFTRAG.conditionPush**

If all conditions are true, the mission is executed. Before, the group(s) wait at the mission execution waypoint.



### #table **AUFTRAG.conditionSuccess**

If all conditions are true, the mission is cancelled.

### #boolean **AUFTRAG.conditionSuccessSet**

### #number **AUFTRAG.dTevaluate**

Time interval in seconds before the mission result is evaluated after mission is over.

### #number **AUFTRAG.duration**

Mission duration in seconds.

### #number **AUFTRAG.durationExe**

Mission execution time in seconds.

### #number **AUFTRAG.engageAltitude**

Engagement altitude in meters.

### #boolean **AUFTRAG.engageAsGroup**

Group attack.

### #number **AUFTRAG.engageDirection**

Engagement direction in degrees.

### #number **AUFTRAG.engageLength**

Length of engage (carpet or strafing) in meters.

### #number **AUFTRAG.engageMaxDistance**

Max engage distance.

### #number **AUFTRAG.engageQuantity**



### Ops.Target#TARGET **AUFTRAG.engageTarget**

Target data to engage.

### #table **AUFTRAG.engageTargetTypes**

Table of target types that are engaged in the engagement zone.

### #number **AUFTRAG.engageWeaponExpend**

How many weapons are used.

### #number **AUFTRAG.engageWeaponType**

Weapon type used.

### Core.Zone#ZONE\_RADIUS **AUFTRAG.engageZone**

*Circular* engagement zone.

### **AUFTRAG.engageddetectedEngageZones**

### **AUFTRAG.engageddetectedNoEngageZones**

### #boolean **AUFTRAG.engageddetectedOn**

Set parameters.

### **AUFTRAG.engageddetectedRmax**

### **AUFTRAG.engageddetectedTypes**

### #table **AUFTRAG.enrouteTasks**

Mission enroute tasks.

### #table **AUFTRAG.escortCohorts**



Engage range in nautical miles (NM).

#### Wrapper.Group#GROUP **AUFTRAG.escortGroup**

The group to be escorted.

#### #string **AUFTRAG.escortGroupName**

Name of the escorted group.

#### #table **AUFTRAG.escortLegions**

Legions explicitly requested for providing escorting assets.

#### #string **AUFTRAG.escortMissionType**

Escort mission type.

#### #table **AUFTRAG.escortTargetTypes**

Target types that will be engaged.

#### DCS#Vec3 **AUFTRAG.escortVec3**

The 3D offset vector from the escorted group to the escort group.

#### #boolean **AUFTRAG.facDatalink**

FAC datalink enabled.

#### #number **AUFTRAG.facDesignation**

FAC designation type.

#### #number **AUFTRAG.facFreq**

FAC radio frequency in MHz.

[REDACTED]

[REDACTED]

[REDACTED]

### #boolean **AUFTRAG.failurecondition**

#### #table **AUFTRAG.groupdata**

Group specific data.

#### **AUFTRAG.hoverAltitude**

#### **AUFTRAG.hoverTime**

#### Ops.OpsGroup#OPSGROUP.Beacon **AUFTRAG.icls**

ICLS setting.

#### #number **AUFTRAG.importance**

Importance.

#### #boolean **AUFTRAG.legionReturn**

If true, assets return to their legion (default). If false, they will stay alive.

#### #table **AUFTRAG.legions**

Assigned legions.

#### #string **AUFTRAG.lid**

Class id string for output to DCS log file.

#### Wrapper.Marker#MARKER **AUFTRAG.marker**

F10 map marker.

#### #number **AUFTRAG.markerCoaliton**

Coalition to which the marker is dispayed.

#### #boolean **AUFTRAG.markerOn**



Mission altitude in meters.

#### Core.Point#COORDINATE **AUFTRAG.missionEgressCoord**

Mission egress waypoint coordinate.

#### **AUFTRAG.missionEgressCoordAlt**

##### #number **AUFTRAG.missionFraction**

Mission coordinate fraction. Default is 0.5.

#### **AUFTRAG.missionHoldingCoord**

#### **AUFTRAG.missionHoldingCoordAlt**

#### Core.Point#COORDINATE **AUFTRAG.missionIngressCoord**

Mission Ingress waypoint coordinate.

##### #number **AUFTRAG.missionRange**

Mission range in meters. Used by LEGION classes (AIRWING, BRIGADE, ...).

##### #number **AUFTRAG.missionSpeed**

Mission speed in km/h.

##### #string **AUFTRAG.missionTask**

Mission task. See ENUMS.MissionTask.

#### Core.Point#COORDINATE **AUFTRAG.missionWaypointCoord**

Mission waypoint coordinate.

##### #number **AUFTRAG.missionWaypointRadius**



Mission name.

#### Ops.Operation#OPERATION **AUFTAG.operation**

Operation this mission is part of.

#### Ops.OpsTransport#OPSTRANSPORT **AUFTAG.opstransport**

OPS transport assignment.

#### #number **AUFTAG.optionAlarm**

Alarm state.

#### #number **AUFTAG.optionCM**

Counter measures.

#### #number **AUFTAG.optionECM**

ECM.

#### #boolean **AUFTAG.optionEPLRS**

EPLRS datalink.

#### #boolean **AUFTAG.optionEmission**

Emission is on or off.

#### #number **AUFTAG.optionFormation**

Formation.

#### #boolean **AUFTAG.optionImmortal**

Immortal is on/off.

**#number AUFTRAG.optionROE**

ROE.

**#number AUFTRAG.optionROT**

ROT.

**#number AUFTRAG.optionRTBammo**

RTB on out-of-ammo.

**#number AUFTRAG.optionRTBfuel**

RTB on out-of-fuel.

**#number AUFTRAG.orbitAltitude**

Orbit altitude in meters.

**#number AUFTRAG.orbitDeltaR**

Distance threshold in meters for moving orbit targets.

**#number AUFTRAG.orbitHeading**

Orbit heading in degrees.

**#number AUFTRAG.orbitLeg**

Length of orbit leg in meters.

**DCS#Vec2 AUFTRAG.orbitOffsetVec2**

2D offset vector.

**#number AUFTRAG.orbitSpeed**



2D orbit vector.

#### Ops.Airwing#AIRWING.PatrolData **AUFTRAG.patrolData**

Patrol data.

#### #table **AUFTRAG.payloads**

User specified airwing payloads for this mission. Only these will be considered for the job!

#### #number **AUFTRAG.prio**

Mission priority.

#### #boolean **AUFTRAG.prohibitAB**

#### #boolean **AUFTRAG.prohibitABExecute**

#### #table **AUFTRAG.properties**

DCS attribute(s) of assets.

#### Ops.OpsGroup#OPSGROUP.Radio **AUFTRAG.radio**

Radio freq and modulation.

#### #number **AUFTRAG.refuelSystem**

Refuel type (boom or probe) for TANKER missions.

#### **AUFTRAG.reinforce**

#### #number **AUFTRAG.repeated**

Number of times mission was repeated.

#### #number **AUFTRAG.repeatedFailure**



Number of times mission was repeated after a success.

#### #table **AUFTRAG.requestID**

The ID of the queued warehouse request. Necessary to cancel the request if the mission was cancelled before the request is processed.

#### #table **AUFTRAG.specialCohorts**

User specified cohorts assigned for this mission. Only these will be considered for the job!

#### #table **AUFTRAG.specialLegions**

User specified legions assigned for this mission. Only these will be considered for the job!

#### #string **AUFTRAG.status**

Mission status.

#### #string **AUFTRAG.statusChief**

Mission status of the CHIEF.

#### #string **AUFTRAG.statusCommander**

Mission status of the COMMANDER.

#### #table **AUFTRAG.statusLegion**

Mission status of all assigned LEGIONS.

#### #boolean **AUFTRAG.successcondition**

#### Ops.OpsGroup#OPSGROUP.Beacon **AUFTRAG.tacan**

TACAN setting.

#### #number **AUFTRAG.targetHeading**



Groups are teleported to the mission ingress waypoint.

#### #table **AUFTRAG.transportCohorts**

Cohorts explicitly requested for providing transport carrier assets.

#### Core.Zone#ZONE **AUFTRAG.transportDeployZone**

Deploy zone of an OPSTRANSPORT.

#### Core.Zone#ZONE **AUFTRAG.transportDisembarkZone**

Disembark zone of an OPSTRANSPORT.

#### Core.Point#COORDINATE **AUFTRAG.transportDropoff**

Coordinate where to drop off the cargo.

#### Core.Set#SET\_GROUP **AUFTRAG.transportGroupSet**

Groups to be transported.

#### #table **AUFTRAG.transportLegions**

Legions explicitly requested for providing transport carrier assets.

#### Core.Point#COORDINATE **AUFTRAG.transportPickup**

Coordinate where to pickup the cargo.

#### #number **AUFTRAG.transportPickupRadius**

Radius in meters for pickup zone. Default 500 m.

#### #string **AUFTRAG.type**

Mission type.



### #boolean **AUFTRAG.urgent**

Mission is urgent. Running missions with lower prio might be cancelled.

### #number **AUFTRAG.verbose**

Verbosity level.

### #string **AUFTRAG.version**

AUFTRAG class version.

## Function(s)

### **AUFTRAG:AddAsset(Asset)**

Add asset to mission.

#### Defined in:

AUFTRAG

#### Parameter:

[Functional.Warehouse#WAREHOUSE.Assetitem](#) **Asset**

The asset to be added to the mission.

---

#### Return value:

[#AUFTRAG:](#)

self



Add failure condition.

## Defined in:

AUFTRAG

## Parameters:

#function **ConditionFunction**

If this function returns true, the mission is cancelled.

---

...

Condition function arguments if any.

---

## Return value:

#AUFTRAG:

self

# AUFTRAG:AddConditionPush(ConditionFunction, ...)

Add push condition.

## Defined in:

AUFTRAG

## Parameters:

#function **ConditionFunction**



Condition function arguments if any.

## Return value:

#AUFTRAG:

self

# AUFTRAG:AddConditionStart(ConditionFunction, ...)

Add start condition.

## Defined in:

AUFTRAG

## Parameters:

#function **ConditionFunction**

Function that needs to be true before the mission can be started. Must return a #boolean.

...

Condition function arguments if any.

## Return value:

#AUFTRAG:

self



Add success condition.

## Defined in:

AUFTRAG

## Parameters:

#function **ConditionFunction**

If this function returns true, the mission is cancelled.

---

...

Condition function arguments if any.

---

## Return value:

#AUFTRAG:

self

---

## AUFTRAG:AddLegion(Legion)

Add LEGION to mission.

## Defined in:

AUFTRAG

## Parameter:

[Ops.Legion](#) #LEGION Legion

### #AUFTRAG:

self

## AUFRAG:AddOpsGroup(OpsGroup)

Add a Ops group to the mission.

### Defined in:

AUFRAG

### Parameter:

Ops.OpsGroup#OPSGROUP **OpsGroup**

The OPSGROUP object.

### Return value:

### #AUFTRAG:

self

## AUFRAG:AddRequiredPayload(Payload)

Add a required payload for this mission.

Only these payloads will be used for this mission. If they are not available, the mission cannot start. Only available for use with an AIRWING.

### Defined in:



### Ops.Airwing#AIRWING.Payload **Payload**

Required payload.

---

## **Return value:**

#AUFTRAG:

self

---

## **AUFTRAG:AddTransportCarriers(Carriers)**

Add carriers for a transport of mission assets.

## **Defined in:**

AUFTRAG

## **Parameter:**

Core.Set#SET\_OPSGROUP **Carriers**

Set of carriers. Can also be a single group.

---

## **Return value:**

#AUFTRAG:

self

---

## **AUFTRAG:AssignCohort(Cohort)**

**[LEGION, COMMANDER, CHIEF]** Assign a legion cohort to the mission.



## AUFTAG

### Parameter:

[Ops.Cohort#COHORT Cohort](#)

The cohort.

---

### Return value:

[#AUFTAG:](#)

self

---

## AUFTAG:AssignEscortCohort(Cohort)

**[LEGION, COMMANDER, CHIEF]** Assign an escort cohort.

### Defined in:

AUFTAG

### Parameter:

[Ops.Cohort#Cohort Cohort](#)

The cohort.

---

## AUFTAG:AssignEscortLegion(Legion)

**[LEGION, COMMANDER, CHIEF]** Add an escort Legion.



## Parameter:

### Ops.Legion#LEGION Legion

The legion.

## AUFTAG:AssignLegion(Legion)

**[LEGION, COMMANDER, CHIEF]** Assign a legion to the mission.

Only cohorts of this legion will be considered for the job. You can assign multiple legions.

## Defined in:

AUFTAG

## Parameter:

### Ops.Legion#LEGION Legion

The legion.

## Return value:

### #AUFTAG:

self

## AUFTAG:AssignSquadrons(Squadrons)

**[LEGION, COMMANDER, CHIEF]** Assign airwing squadron(s) to the mission.

Only these squads will be considered for the job.



## Parameter:

#table **Squadrons**

A table of SQUADRON(s). **Has to be a table {}** even if a single squad is given.

## Return value:

#AUFTRAG:

self

## AUFRAG:AssignTransportCohort(Cohort)

**[LEGION, COMMANDER, CHIEF]** Assign a transport cohort.

## Defined in:

AUFRAG

## Parameter:

Ops.Cohort#Cohort **Cohort**

The cohort.

## AUFRAG:AssignTransportLegion(Legion)

**[LEGION, COMMANDER, CHIEF]** Assign a transport Legion.

## Defined in:



## Ops.Legion#LEGION Legion

The legion.

## AUFRAG:Cancel()

Triggers the FSM event "Cancel".

### Defined in:

AUFRAG

## AUFRAG:CheckGroupsDone()

Check if all groups are done with their mission (or dead).

### Defined in:

AUFRAG

### Return value:

#boolean:

If true, all groups are done with the mission.

## AUFRAG.CheckMissionCapability(MissionTypes, Capabilities, All)

Check if a mission type is contained in a list of possible capabilities.



## Parameters:

### #table **MissionTypes**

The requested mission type. Can also be passed as a single mission type #string.

---

### #table **Capabilities**

A table with possible capabilities Ops.Auftrag#AUFTRAG.Capability.

---

### #boolean **All**

If true, given mission type must be included in ALL capabilities. If false or nil, it must only match one.

---

## Return value:

### #boolean:

If true, the requested mission type is part of the possible mission types.

---

## AUFTRAG.CheckMissionCapabilityAll(MissionTypes, Capabilities)

Check if a mission type is contained in a list of possible capabilities.

## Defined in:

AUFTRAG

## Parameters:

### #table **MissionTypes**



A table with possible capabilities `Ops.Auftrag#AUFTRAG.Capability`.

## Return value:

#boolean:

If true, the requested mission type is part of the possible mission types.

# AUFTRAG.CheckMissionCapabilityAny(MissionTypes, Capabilities)

Check if a mission type is contained in a list of possible capabilities.

## Defined in:

AUFTRAG

## Parameters:

#table **MissionTypes**

The requested mission type. Can also be passed as a single mission type #string.

#table **Capabilities**

A table with possible capabilities `Ops.Auftrag#AUFTRAG.Capability`.

## Return value:

#boolean:

If true, the requested mission type is part of the possible mission types.



Checks if a mission type is contained in a table of possible types.

## Defined in:

AUFTRAG

## Parameters:

#string **MissionType**

The requested mission type.

#table **PossibleTypes**

A table with possible mission types.

## Return value:

#boolean:

If true, the requested mission type is part of the possible mission types.

## AUFTRAG:CountMissionTargets()

Count alive mission targets.

## Defined in:

AUFTRAG

## Return value:

#number:



## AUFRAG:CountOpsGroups()

Count alive OPS groups assigned for this mission.

### Defined in:

AUFRAG

### Return value:

#number:

Number of alive OPS groups.

## AUFRAG:CountOpsGroupsInStatus(Status)

Count OPS groups in a certain status.

### Defined in:

AUFRAG

### Parameter:

#string **Status**

Status of group, e.g. AUFRAG.GroupStatus.EXECUTING.

### Return value:

#number:

Number of alive OPS groups.



## Defined in:

AUFTAG

## Parameter:

[Functional.Warehouse#WAREHOUSE.Assetitem](#) **Asset**

The asset to be removed.

## Return value:

[#AUFTAG](#):

self

# AUFTAG:DelOpsGroup(OpsGroup)

Remove an Ops group from the mission.

## Defined in:

AUFTAG

## Parameter:

[Ops.OpsGroup#OPSGROUP](#) **OpsGroup**

The OPSGROUP object.

## Return value:

[#AUFTAG](#):



## AUFRAG:Done()

Triggers the FSM event "Done".

### Defined in:

AUFRAG

## AUFRAG:EvalConditionsAll(Conditions)

Check if all given condition are true.

### Defined in:

AUFRAG

### Parameter:

#table **Conditions**

Table of conditions.

### Return value:

#boolean:

If true, all conditions were true. Returns false if at least one condition returned false.

## AUFRAG:EvalConditionsAny(Conditions)

Check if any of the given conditions is true.

### Defined in:



## #table **Conditions**

Table of conditions.

---

## **Return value:**

#boolean:

If true, at least one condition is true.

---

## **AUFRAG:Evaluate()**

Evaluate mission outcome - success or failure.

## **Defined in:**

AUFRAG

## **Return value:**

#AUFRAG:

self

---

## **AUFRAG:Executing()**

Triggers the FSM event "Executing".

## **Defined in:**

AUFRAG



## Defined in:

AUFRAG

# AUFRAG:GetAssetByName(Name)

Get asset by its spawn group name.

## Defined in:

AUFRAG

## Parameter:

#string **Name**

Asset spawn group name.

## Return value:

[Functional.Warehouse#WAREHOUSE.Assetitem](#):

Asset.

# AUFRAG:GetAssetDataByName(AssetName)

Get asset data table.

## Defined in:

AUFRAG



Name of the asset.

## Return value:

#AUFTRAG.GroupData:

Group data or *nil* if OPS group does not exist.

## AUFTRAG:GetCasualties()

Get casualties, *i.e.* number of own units that died during this mission.

## Defined in:

AUFTRAG

## Return value:

#number:

Number of dead units.

## AUFTRAG:GetDCSMissionTask()

Get DCS task table for the given mission.

## Defined in:

AUFTRAG

## Return value:



## AUFRAG:GetGroupData(opsgroup)

Get flight data table.

### Defined in:

AUFRAG

### Parameter:

Ops.OpsGroup#OPSGROUP **opsgroup**

The flight group.

### Return value:

#AUFRAG.GroupData:

Flight data or nil if opsgroup does not exist.

## AUFRAG:GetGroupEgressWaypointUID(opsgroup)

Get Egress waypoint UID of OPS group.

### Defined in:

AUFRAG

### Parameter:

Ops.OpsGroup#OPSGROUP **opsgroup**

The OPS group.



Waypoint UID.

## AUFRAG:ListGroupStatus(opsgroup)

Get ops group mission status.

### Defined in:

AUFRAG

### Parameter:

[Ops.OpsGroup#OPSGROUP](#) **opsgroup**

The OPS group.

### Return value:

#string:

The group status.

## AUFRAG:ListGroupWaypointCoordinate(opsgroup)

Get mission (ingress) waypoint coordinate of OPS group

### Defined in:

AUFRAG

### Parameter:



## Return value:

Core.Point#COORDINATE:

Waypoint Coordinate.

## AUFRAG:ListGroupWaypointIndex(opsgroup)

Get mission (ingress) waypoint UID of OPS group.

### Defined in:

AUFRAG

### Parameter:

Ops.OpsGroup#OPSGROUP **opsgroup**

The OPS group.

## Return value:

#number:

Waypoint UID.

## AUFRAG:ListGroupWaypointTask(opsgroup)

Get mission waypoint task of OPS group.

### Defined in:



### Ops.OpsGroup#OPSGROUP opsgroup

The OPS group.

## Return value:

### Ops.OpsGroup#OPSGROUP.Task:

task Waypoint task. Waypoint task.

## AUFTAG:GetImportance()

Get mission importance.

## Defined in:

AUFTAG

## Return value:

#number:

Importance. Smaller is higher.

## AUFTAG:GetKills()

Get kills, i.e.

number of units that were destroyed by assets of this mission.

## Defined in:

AUFTAG



Number of units destroyed.

## AUFRAG:GetLegionStatus(Legion)

Get LEGION mission status.

### Defined in:

AUFRAG

### Parameter:

[Ops.Legion#LEGION Legion](#)

The legion.

### Return value:

#string:

status Current status.

## AUFRAG:GetMissionEgressCoord()

Get the mission egress coordinate if this was defined.

### Defined in:

AUFRAG

### Return value:



## AUFRAG:GetMissionHoldingCoord()

Get the mission holding coordinate if this was defined.

### Defined in:

AUFRAG

### Return value:

[Core.Point#COORDINATE](#):

Coordinate Coordinate or nil.

## AUFRAG:GetMissionIngressCoord()

Get the mission ingress coordinate if this was defined.

### Defined in:

AUFRAG

### Return value:

[Core.Point#COORDINATE](#):

Coordinate Coordinate or nil.

## AUFRAG:GetMissionTaskforMissionType(MissionType)

Get DCS task table for an attack group or unit task.



## Parameter:

#string **MissionType**

Mission (AUFTAG) type.

## Return value:

#string:

DCS mission task for the auftrag type.

# AUFTAG:GetMissionTypesText(MissionTypes)

Get coordinate of target.

First unit/group of the set is used.

## Defined in:

AUFTAG

## Parameter:

#table **MissionTypes**

A table of mission types.

## Return value:

#string:

Comma separated list of mission types.



Get coordinate of target.

First unit/group of the set is used.

## Defined in:

AUFRAG

## Parameters:

[Wrapper.Group#GROUP group](#)

Group.

---

#number **randomradius**

Random radius in meters.

---

#table **surfacetypes**

Surface types of random zone.

## Return value:

[Core.Point#COORDINATE:](#)

Coordinate where the mission is executed.

## AUFRAG:GetName()

Get mission name.

## Defined in:

#string:

Mission name, e.g. "Auftrag Nr.1".

## AUFRAG:GetNumberOfRequiredAssets()

Get number of required assets.

### Defined in:

AUFRAG

### Return value:

#number:

Numer of required assets.

## AUFRAG:GetObjectve(RefCoordinate, Coalitions)

Get mission objective object.

Could be many things depending on the mission type.

### Defined in:

AUFRAG

### Parameters:

Core.Point#COORDINATE RefCoordinate

(Optional) Reference coordinate from which the closest target is determined.



## Return value:

Wrapper.Positionable#POSITIONABLE:

The target object. Could be many things.

## AUFRAG:GetOpsGroups()

Get all OPS groups.

### Defined in:

AUFRAG

## Return value:

#table:

Table of Ops.OpsGroup#OPSGROUP or {}.

## AUFRAG:GetOpsTransport()

Get the attach OPS transport of the mission.

### Defined in:

AUFRAG

## Return value:

Ops.OpsTransport#OPSTRANSPORT:

The OPS transport assignment attached to the mission.



Get mission priority.

## Defined in:

AUFTAG

## Return value:

#number:

Priority. Smaller is higher.

# AUFTAG:GetRequiredAssets()

**[LEGION, COMMANDER, CHIEF]** Get number of required assets.

## Defined in:

AUFTAG

## Return values:

#number:

Min. number of required assets.

#number:

Max. number of required assets.

# AUFTAG:GetTargetCoordinate()

Get coordinate of target.



## Return value:

Core.Point#COORDINATE:

The target coordinate or *nil*.

## AUFRAG:GetTargetDamage()

Get target damage.

### Defined in:

AUFRAG

## Return value:

#number:

Damage in percent.

## AUFRAG:GetTargetData()

Get target.

### Defined in:

AUFRAG

## Return value:

Ops.Target#TARGET:

The target object. Could be many things.



Get distance to target.

## Defined in:

AUFRAG

## Parameter:

Core.Point#COORDINATE FromCoord

The coordinate from which the distance is measured.

## Return value:

#number:

Distance in meters or 0.

# AUFRAG:GetTargetHeading()

Get heading of target.

## Defined in:

AUFRAG

## Return value:

#number:

Heading of target in degrees.

# AUFRAG:GetTargetInitialLife()



AUFRAG

## Return value:

#number:

Number of initial life points when mission was planned.

# AUFRAG:GetTargetInitialNumber()

Get initial number of targets.

## Defined in:

AUFRAG

## Return value:

#number:

Number of initial life points when mission was planned.

# AUFRAG:GetTargetLife()

Get target life points.

## Defined in:

AUFRAG

## Return value:

#number:



## AUFRAG:GetTargetName()

Get name of the target.

### Defined in:

AUFRAG

### Return value:

#string:

Name of the target or "N/A".

## AUFRAG:GetTargetType()

Get type of target.

### Defined in:

AUFRAG

### Return value:

#string:

The target type.

## AUFRAG:GetTargetVec2()

Get 2D vector of target.

### Defined in:



### DCS#VEC2:

The target 2D vector or *nil*.

## AUFRAG:GetType()

Get mission type.

### Defined in:

AUFRAG

### Return value:

#string:

Mission type, e.g. "BAI".

## AUFRAG:IsAircraft()

Check if mission is for aircraft (airplanes and/or helicopters).

### Defined in:

AUFRAG

### Return value:

#boolean:

If true, mission is for aircraft.



## Defined in:

AUFRAG

## Return value:

#boolean:

If true, mission is for airplanes.

## AUFRAG:IsCancelled()

Check if mission was cancelled.

## Defined in:

AUFRAG

## Return value:

#boolean:

If true, mission was cancelled.

## AUFRAG:IsDone()

Check if mission is done.

## Defined in:

AUFRAG



If true, mission is done.

## AUFRAG:IsExecuting(AllGroups)

Check if mission is EXECUTING.

The first OPSGROUP has reached the mission execution waypoint and is not executing the mission task.

### Defined in:

AUFRAG

### Parameter:

#boolean **AllGroups**

(Optional) Check that all groups are currently executing the mission.

### Return value:

#boolean:

If true, mission is currently executing.

## AUFRAG:IsGround()

Check if mission is for ground units.

### Defined in:

AUFRAG



If true, mission is for ground units.

## AUFRAG:IsHelicopters()

Check if mission is for helicopters.

### Defined in:

AUFRAG

### Return value:

#boolean:

If true, mission is for helicopters.

## AUFRAG:IsNaval()

Check if mission is for naval units.

### Defined in:

AUFRAG

### Return value:

#boolean:

If true, mission is for naval units.

## AUFRAG: IsNotOver()



AUFRAG

## Return value:

#boolean:

If true, mission is NOT over yet.

## AUFRAG:IsOver()

Check if mission is over.

This could be state DONE, CANCELLED, SUCCESS, FAILED.

## Defined in:

AUFRAG

## Return value:

#boolean:

If true, mission is over.

## AUFRAG:IsPlanned()

Check if mission is PLANNED.

## Defined in:

AUFRAG

## Return value:



## AUFRAG:IsQueued(Legion)

Check if mission is QUEUED at a LEGION mission queue.

### Defined in:

AUFRAG

### Parameter:

[Ops.Legion#LEGION Legion](#)

(Optional) Check if mission is queued at this legion.

### Return value:

#boolean:

If true, mission is queued.

## AUFRAG:IsReadyToCancel()

Check if mission is ready to be cancelled.

- Mission stop already passed.
- Any stop condition is true.

### Defined in:

AUFRAG

### Return value:



## AUFRAG:IsReadyToGo()

Check if mission is ready to be started.

- Mission start time passed.
- Mission stop time did not pass already.
- All start conditions are true.

### Defined in:

AUFRAG

### Return value:

#boolean:

If true, mission can be started.

## AUFRAG:IsReadyToPush()

Check if mission is ready to be pushed.

- Mission push time already passed.
- **All** push conditions are true.

### Defined in:

AUFRAG

### Return value:

#boolean:

If true, mission groups can push.



The mission request out to the WAREHOUSE.

## Defined in:

AUFTAG

## Parameter:

[Ops.Legion#LEGION Legion](#)

(Optional) Check if mission is requested at this legion.

## Return value:

#boolean:

If true, mission is requested.

## AUFTAG:IsScheduled()

Check if mission is SCHEDULED.

The first OPSGROUP has been assigned.

## Defined in:

AUFTAG

## Return value:

#boolean:

If true, mission is queued.



The first OPSGROUP is on its way to the mission execution waypoint.

## Defined in:

AUFTAG

## Return value:

#boolean:

If true, mission is started.

## AUFTAG:.IsSuccess()

Check if mission was a success.

## Defined in:

AUFTAG

## Return value:

#boolean:

If true, mission was successful.

## AUFTAG:IsUrgent()

Check if mission is "urgent".

## Defined in:

AUFTAG



If true, mission is "urgent".

## AUFRAG:New(Type)

Create a new generic AUFRAG object.

### Defined in:

AUFRAG

### Parameter:

#string **Type**

Mission type.

### Return value:

#AUFRAG:

self

## AUFRAG:NewAIRDEFENSE(Zone)

**[GROUND, NAVAL]** Create an AIRDEFENSE mission.

### Defined in:

AUFRAG

### Parameter:



## Return value:

#AUFTRAG:

self

## AUFTRAG:NewALERT5(MissionType)

**[AIR]** Create an ALERT 5 mission.

Aircraft will be spawned uncontrolled and wait for an assignment. You must specify **one** mission type which is performed. This determines the payload and the DCS mission task which are used when the aircraft is spawned.

## Defined in:

AUFTRAG

## Parameter:

#string **MissionType**

Mission type AUFTRAG.Type.XXX. Determines payload and mission task (intercept, ground attack, etc.).

## Return value:

#AUFTRAG:

self

## AUFTRAG:NewAMMOSUPPLY(Zone)



## AUFTAG

### Parameter:

#### Core.Zone#ZONE Zone

The zone, where supply units go.

---

### Return value:

#### #AUFTAG:

self

---

## AUFTAG:NewANTISHIP(Target, Altitude)

**[AIR]** Create an ANTI-SHIP mission.

### Defined in:

AUFTAG

### Parameters:

#### Wrapper.Positionable#POSITIONABLE Target

The target to attack. Can be passed as a Wrapper.Group#GROUP or Wrapper.Unit#UNIT object.

---

#### #number Altitude

Engage altitude in feet. Default 2000 ft.

---

### Return value:



# AUFRAG:NewARMORATTACK(Target, Speed, Formation)

**[OBSOLETE]** Create a ARMORATTACK mission.

\* Note that this is actually creating a GROUNDATTACK mission!\*

## Defined in:

AUFRAG

## Parameters:

### Ops.Target#TARGET **Target**

The target to attack. Can be a GROUP, UNIT or STATIC object.

---

### #number **Speed**

Speed in knots.

---

### #string **Formation**

The attack formation, e.g. "Wedge", "Vee" etc.

## Return value:

### #AUFRAG:

self



**[GROUND]** Create an ARMORED ON GUARD mission.

## Defined in:

AUFTRAG

## Parameters:

### Core.Point#COORDINATE Coordinate

Coordinate, where to stand guard.

---

### #string Formation

Formation to take, e.g. "On Road", "Vee" etc.

---

## Return value:

#AUFTRAG:

self

## AUFTRAG:NewARTY(Target, Nshots, Radius, Altitude)

**[GROUND, NAVAL]** Create an ARTY mission ("Fire at point" task).

If the group has more than one weapon type supporting the "Fire at point" task, the employed weapon type can be set via the AUFTRAG:SetWeaponType() function.

**Note** that it is recommended to set the weapon range via the OPSGROUP:AddWeaponRange() function as this cannot be retrieved from the DCS API.

## Defined in:

AUFTRAG



Center of the firing solution.

---

**#number Nshots**

Number of shots to be fired. Default #nil.

---

**#number Radius**

Radius of the shells in meters. Default 100 meters.

---

**#number Altitude**

Altitude in meters. Can be used to setup a Barrage. Default #nil.

## Return value:

AUFTAG:

self

## AUFTAG:NewAUTO(EngageGroup)

Create a mission to attack a group.

Mission type is automatically chosen from the group category.

## Defined in:

AUFTAG

## Parameter:

Wrapper.Group#GROUP EngageGroup

## #AUFTRAG:

self

# AUFTRAG:NewAWACS(Coordinate, Altitude, Speed, Heading, Leg)

**[AIR]** Create a AWACS mission.

## Defined in:

AUFTRAG

## Parameters:

### Core.Point#COORDINATE Coordinate

Where to orbit. Altitude is also taken from the coordinate.

### #number Altitude

Orbit altitude in feet. Default is y component of Coordinate.

### #number Speed

Orbit speed in knots. Default 350 kts.

### #number Heading

Heading of race-track pattern in degrees. Default 270 (East to West).

### #number Leg

**#AUFTRAG:**

self

## AUFRAG:NewBAI(Target, Altitude)

**[AIR]** Create a BAI mission.

### Defined in:

AUFRAG

### Parameters:

Wrapper.Positionable#POSITIONABLE Target

The target to attack. Can be a GROUP, UNIT or STATIC object.

#number **Altitude**

Engage altitude in feet. Default 5000 ft.

### Return value:

**#AUFTRAG:**

self

## AUFRAG:NewBARRAGE(Zone, Heading, Angle, Radius, Altitude, Nshots)

**[GROUND, NAVAL]** Create an BARRAGE mission.



## AUFTAG

### Parameters:

#### Core.Zone#ZONE Zone

The zone where the unit will go.

---

#### #number **Heading**

Heading in degrees. Default random heading [0, 360).

---

#### #number **Angle**

Shooting angle in degrees. Default random [45, 85].

---

#### #number **Radius**

Radius of the shells in meters. Default 100 meters.

---

#### #number **Altitude**

Altitude in meters. Default 500 m.

---

#### #number **Nshots**

Number of shots to be fired. Default is until ammo is empty (#nil).

---

### Return value:

#### #AUFTAG:

self

---



**[AIR]** Create a CARPET BOMBING mission.

## Defined in:

AUFRAG

## Parameters:

### Core.Point#COORDINATE Target

Target coordinate. Can also be specified as a GROUP, UNIT or STATIC object.

---

### #number Altitude

Engage altitude in feet. Default 25000 ft.

---

### #number CarpetLength

Length of bombing carpet in meters. Default 500 m.

## Return value:

### #AUFRAG:

self

## AUFRAG:NewBOMBING(Target, Altitude, EngageWeaponType)

**[AIR]** Create a BOMBING mission.

Flight will drop bombs at a specified coordinate. See [DCS task bombing](#).



## Parameters:

### Core.Point#COORDINATE Target

Target coordinate. Can also be specified as a GROUP, UNIT, STATIC or TARGET object.

---

### #number Altitude

Engage altitude in feet. Default 25000 ft.

---

### #number EngageWeaponType

Which weapon to use. Defaults to auto, ie ENUMS.WeaponFlag.Auto. See ENUMS.WeaponFlag for options.

## Return value:

### #AUFTRAG:

self

## AUFTRAG:NewBOMBRUNWAY(Airdrome, Altitude)

**[AIR]** Create a BOMBRUNWAY mission.

## Defined in:

AUFTRAG

## Parameters:

### Wrapper.Airbase#AIRBASE Airdrome

#number **Altitude**

Engage altitude in feet. Default 25000 ft.

## Return value:

#AUFTRAG:

self

# AUFTRAG:NewCAP(ZoneCAP, Altitude, Speed, Coordinate, Heading, Leg, TargetTypes)

**[AIR]** Create a CAP mission.

## Defined in:

AUFTRAG

## Parameters:

Core.Zone#ZONE\_RADIUS ZoneCAP

Circular CAP zone. Detected targets in this zone will be engaged.

#number **Altitude**

Altitude at which to orbit in feet. Default is 10,000 ft.

#number **Speed**

Orbit speed in knots. Default 350 kts.

Core.Point#COORDINATE Coordinate



Heading of race-track pattern in degrees. If not specified, a simple circular orbit is performed.

---

#number **Leg**

Length of race-track in NM. If not specified, a simple circular orbit is performed.

---

#table **TargetTypes**

Table of target types. Default {"Air"}.

## Return value:

#AUFTRAG:

self

**AUFTRAG:NewCAPGROUP(Grp, Altitude, Speed, RelHeading, Leg, OffsetDist, OffsetAngle, UpdateDistance, TargetTypes, EngageRange)**

**[AIR]** Create a CAP mission over a (moving) group.

## Defined in:

AUFTRAG

## Parameters:

Wrapper.Group#GROUP Grp

The grp to perform the CAP over.



---

### #number **Speed**

Orbit speed in knots. Default 250 KIAS.

---

### #number **RelHeading**

Relative heading [0, 360) of race-track pattern in degrees wrt heading of the carrier. Default is heading of the carrier.

---

### #number **Leg**

Length of race-track in NM. Default 14 NM.

---

### #number **OffsetDist**

Relative distance of the first race-track point wrt to the carrier. Default 6 NM.

---

### #number **OffsetAngle**

Relative angle of the first race-track point wrt. to the carrier. Default 180 (behind the boat).

---

### #number **UpdateDistance**

Threshold distance in NM before orbit pattern is updated. Default 5 NM.

---

### #table **TargetTypes**

(Optional) Table of target types. Default {"Air"}.

---

### #number **EngageRange**

Max range in nautical miles that the escort group(s) will engage enemies. Default 32 NM (60 km).

---

## Return value:



# AUFRAG:NewCAPTUREZONE(OpsZone, Coalition, Speed, Altitude, Formation)

**[AIR, GROUND, NAVAL]** Create a CAPTUREZONE mission.

Group(s) will go to the zone and patrol it randomly.

## Defined in:

AUFRAG

## Parameters:

Ops.OpsZone#OPZONE **OpsZone**

The OPS zone to capture.

---

#number **Coalition**

The coalition which should capture the zone for the mission to be successful.

---

#number **Speed**

Speed in knots.

---

#number **Altitude**

Altitude in feet. Only for airborne units. Default 2000 feet ASL.

---

#string **Formation**

Formation used by ground units during patrol. Default "Off Road".

self

## AUFRAG:NewCARGOTRANSPORT(StaticCargo, DropZone)

**[AIR ROTARY]** Create a CARGO TRANSPORT mission.

**Important Note:** The dropoff zone has to be a zone defined in the Mission Editor. This is due to a restriction in the used DCS task, which takes the zone ID as input. Only ME zones have an ID that can be referenced.

### Defined in:

AUFRAG

### Parameters:

Wrapper.Static#STATIC StaticCargo

Static cargo object.

Core.Zone#ZONE DropZone

Zone where to drop off the cargo. **Has to be a zone defined in the ME!**

### Return value:

#AUFRAG:

self



- - - - -

**[AIR]** Create a CAS mission.

## Defined in:

AUFTAG

## Parameters:

### Core.Zone#ZONE\_RADIUS ZoneCAS

Circular CAS zone. Detected targets in this zone will be engaged.

---

#### #number Altitude

Altitude at which to orbit. Default is 10,000 ft.

---

#### #number Speed

Orbit speed in knots. Default 350 KIAS.

---

### Core.Point#COORDINATE Coordinate

Where to orbit. Default is the center of the CAS zone.

---

#### #number Heading

Heading of race-track pattern in degrees. If not specified, a simple circular orbit is performed.

---

#### #number Leg

Length of race-track in NM. If not specified, a simple circular orbit is performed.

---

#### #table TargetTypes



## Return value:

#AUFTRAG:

self

# AUFTRAG:NewCASENHANCED(CasZone, Altitude, Speed, RangeMax, NoEngageZoneSet, TargetTypes)

**[AIR]** Create a CASENHANCED mission.

Group(s) will go to the zone and patrol it randomly.

## Defined in:

AUFTRAG

## Parameters:

Core.Zone#ZONE CasZone

The CAS zone.

**#number Altitude**

Altitude in feet. Only for airborne units. Default 2000 feet ASL.

**#number Speed**

Speed in knots.

**#number RangeMax**

Max range in NM. Only detected targets within this radius from the group will be engaged. Default is 25 NM.



## #table **TargetTypes**

Types of target attributes that will be engaged. See [DCS enum attributes](#). Default {"Helicopters", "Ground Units", "Light armed ships"}.

## Return value:

[#AUFTRAG](#):

self

## AUFRAG:NewESCORT(**EscortGroup**, **OffsetVector**, **EngageMaxDistance**, **TargetTypes**)

**[AIR]** Create an ESCORT (or FOLLOW) mission.

Flight will escort another group and automatically engage certain target types.

## Defined in:

AUFRAG

## Parameters:

[Wrapper.Group#GROUP EscortGroup](#)

The group to escort.

[DCS#Vec3 OffsetVector](#)

A table with x, y and z components specifying the offset of the flight to the escorted group. Default {x=-100, y=0, z=200} for z=200 meters to the right, same altitude (y=0), x=-100 meters behind.



## #table TargetTypes

Types of targets to engage automatically. Default is {"Air"}, i.e. all enemy airborne units. Use an empty set {} for a simple "FOLLOW" mission.

## Return value:

#AUFTRAG:

self

## AUFTRAG:NewEWR(Zone)

**[GROUND]** Create an EWR mission.

## Defined in:

AUFTRAG

## Parameter:

Core.Zone#ZONE Zone

Zone where the Early Warning Radar group(s) should be stationed.

## Return value:

#AUFTRAG:

self



**[AIR, GROUND]** Create a FAC mission.

Group(s) will go to the zone and patrol it randomly and act as FAC for detected units.

## Defined in:

AUFRAG

## Parameters:

### Core.Zone#ZONE FacZone

The FAC zone (or name of zone) where to patrol.

---

### #number **Speed**

Speed in knots.

---

### #number **Altitude**

Altitude in feet. Only for airborne units. Default 2000 feet ASL.

---

### #number **Frequency**

Frequency in MHz.

---

### #number **Modulation**

Modulation.

---

## Return value:

### #AUFRAG:

self



**[AIR]** Create a FACA mission.

## Defined in:

AUFTRAG

## Parameters:

### Wrapper.Group#GROUP Target

Target group. Must be a GROUP object.

---

#### #string **Designation**

Designation of target. See AI.Task.Designation. Default AI.Task.Designation.AUTO.

---

#### #boolean **DataLink**

Enable data link. Default true.

---

#### #number **Frequency**

Radio frequency in MHz the FAC uses for communication. Default is 133 MHz.

---

#### #number **Modulation**

Radio modulation band. Default 0=AM. Use 1 for FM. See radio.modulation.AM or radio.modulation.FM.

## Return value:

### #AUFTRAG:

self



## Defined in:

AUFRAG

## Parameter:

### Core.Zone#ZONE Zone

The zone, where supply units go.

## Return value:

### #AUFRAG:

self

## AUFRAG:NewFromTarget(Target, MissionType)

Create a mission to attack a TARGET object.

## Defined in:

AUFRAG

## Parameters:

### Ops.Target#TARGET Target

The target.

### #string **MissionType**

The mission type.



self

## AUFRAG:NewGCICAP(Coordinate, Altitude, Speed, Heading, Leg)

**[AIR]** Create a Ground Controlled CAP (GCICAP) mission.

Flights with this task are considered for A2A INTERCEPT missions by the CHIEF class. They will perform a combat air patrol but not engage by themselves. They wait for the CHIEF to tell them whom to engage.

### Defined in:

AUFRAG

### Parameters:

Core.Point#COORDINATE **Coordinate**

Where to orbit.

#number **Altitude**

Orbit altitude in feet. Default is y component of Coordinate.

#number **Speed**

Orbit indicated airspeed in knots at the set altitude ASL. Default 350 KIAS.

#number **Heading**

Heading of race-track pattern in degrees. Default random in [0, 360) degrees.

#number **Leg**

## #AUFTRAG:

self

# AUFTAG:NewGROUNDATTACK(Target, Speed, Formation)

**[GROUND]** Create a GROUNDATTACK mission.

Ground group(s) will go to a target object and attack.

## Defined in:

AUFTAG

## Parameters:

### Wrapper.Positionable#POSITIONABLE Target

The target to attack. Can be a GROUP, UNIT or STATIC object.

### #number Speed

Speed in knots. Default max.

### #string Formation

The attack formation, e.g. "Wedge", "Vee" etc. Default ENUMS.FORMATION.Vehicle.Vee.

## Return value:

### #AUFTRAG:



# AUFTAG:NewGROUNDESCORT(EscortGroup, OrbitDistance, TargetTypes)

**[AIR/HELO]** Create a GROUNDESCORT (or FOLLOW) mission.

Helo will escort a **ground** group and automatically engage certain target types.

## Defined in:

AUFTAG

## Parameters:

Wrapper.Group#GROUP **EscortGroup**

The ground group to escort.

---

#number **OrbitDistance**

Orbit to/from the lead unit this many NM. Defaults to 1.5 NM.

---

#table **TargetTypes**

Types of targets to engage automatically. Default is {"Ground vehicles"}, i.e. all enemy ground units. Use an empty set {} for a simple "FOLLOW" mission.

## Return value:

#AUFTAG:

---

self



**[AIR ROTARY]** Create an HOVER mission.

## Defined in:

AUFTRAG

## Parameters:

### Core.Point#COORDINATE Coordinate

Where to hover.

---

### #number Altitude

Hover altitude in feet AGL. Default is 50 feet above ground.

---

### #number Time

Time in seconds to hold the hover. Default 300 seconds.

---

### #number Speed

Speed in knots to fly to the target coordinate. Default 150kn.

---

### #number MissionAlt

Altitude to fly towards the mission in feet AGL. Default 1000ft.

## Return value:

### #AUFTRAG:

self



## Defined in:

AUFRAG

## Parameter:

Wrapper.Positionable#POSITIONABLE Target

The target to intercept. Can also be passed as simple Wrapper.Group#GROUP or Wrapper.Unit#UNIT object.

## Return value:

#AUFRAG:

self

**AUFRAG:NewLANDATCOORDINATE(Coordinate, OuterRadius, InnerRadius, Time, Speed, MissionAlt, CombatLanding, DirectionAfterLand)**

**[AIR ROTARY]** Create an LANDATCOORDINATE mission.

## Defined in:

AUFRAG

## Parameters:

Core.Point#COORDINATE Coordinate

Where to land.



OuterRadius and (optionally) avoiding InnerRadius of the coordinate.

---

**#number **InnerRadius****

(Optional) Vary the coordinate by this many feet, e.g. get a new random coordinate between OuterRadius and (optionally) avoiding InnerRadius of the coordinate.

---

**#number **Time****

Time in seconds to stay. Default 300 seconds.

---

**#number **Speed****

Speed in knots to fly to the target coordinate. Default 150kn.

---

**#number **MissionAlt****

Altitude to fly towards the mission in feet AGL. Default 1000ft.

---

**#boolean **CombatLanding****

(Optional) If true, set the Combat Landing option.

---

**#number **DirectionAfterLand****

(Optional) Heading after landing in degrees.

## **Return value:**

---

**#AUFTRAG:**

self

## **AUFTRAG:NewNOTHING(RelaxZone)**



AUFTAG

## Parameter:

### Core.Zone#ZONE RelaxZone

Zone where the assets are supposed to do nothing.

---

## Return value:

### #AUFTAG:

self

---

## AUFTAG:NewONGUARD(Coordinate)

**[GROUND, NAVAL]** Create an ON GUARD mission.

## Defined in:

AUFTAG

## Parameter:

### Core.Point#COORDINATE Coordinate

Coordinate, where to stand guard.

---

## Return value:

### #AUFTAG:

self

---



**[AIR]** Create an ORBIT mission, which can be either a circular orbit or a race-track pattern.

## Defined in:

AUFRAG

## Parameters:

### Core.Point#COORDINATE Coordinate

Where to orbit.

---

### #number Altitude

Orbit altitude in feet above sea level. Default is y component of Coordinate.

---

### #number Speed

Orbit indicated airspeed in knots at the set altitude ASL. Default 350 KIAS.

---

### #number Heading

Heading of race-track pattern in degrees. If not specified, a circular orbit is performed.

---

### #number Leg

Length of race-track in NM. If not specified, a circular orbit is performed.

## Return value:

### #AUFRAG:

self



**[AIR]** Create an ORBIT mission, where the aircraft will go in a circle around the specified coordinate.

## Defined in:

AUFTRAG

## Parameters:

### Core.Point#COORDINATE Coordinate

Position where to orbit around.

---

### #number Altitude

Orbit altitude in feet. Default is y component of Coordinate.

---

### #number Speed

Orbit indicated airspeed in knots at the set altitude ASL. Default 350 KIAS.

## Return value:

### #AUFTRAG:

self

## AUFTRAG:NewORBIT\_GROUP(Group, Altitude, Speed, Leg, Heading, OffsetVec2, Distance)

**[AIR]** Create an ORBIT mission, where the aircraft will fly a circular or race-track pattern over a given group or unit.



## Parameters:

### Wrapper.Group#GROUP Group

Group where to orbit around. Can also be a UNIT object.

---

#### #number **Altitude**

Orbit altitude in feet. Default is 6,000 ft.

---

#### #number **Speed**

Orbit indicated airspeed in knots at the set altitude ASL. Default 350 KIAS.

---

#### #number **Leg**

Length of race-track in NM. Default nil.

---

#### #number **Heading**

Heading of race-track pattern in degrees. Default is heading of the group.

---

### DCS#Vec2 OffsetVec2

Offset 2D-vector {x=0, y=0} in NM with respect to the group. Default directly overhead. Can also be given in polar coordinates {r=5, phi=45}.

---

#### #number **Distance**

Threshold distance in NM before orbit pattern is updated. Default 5 NM.

---

## Return value:

### #AUFTRAG:



# AUFTAG:NewORBIT\_RACETRACK(Coordinate, Altitude, Speed, Heading, Leg)

**[AIR]** Create an ORBIT mission, where the aircraft will fly a race-track pattern.

## Defined in:

AUFTAG

## Parameters:

### Core.Point#COORDINATE **Coordinate**

Where to orbit.

---

### #number **Altitude**

Orbit altitude in feet. Default is y component of Coordinate.

---

### #number **Speed**

Orbit indicated airspeed in knots at the set altitude ASL. Default 350 KIAS.

---

### #number **Heading**

Heading of race-track pattern in degrees. Default random in [0, 360) degrees.

---

### #number **Leg**

Length of race-track in NM. Default 10 NM.

## Return value:

### #AUFTAG:



# AUFTAG:NewPATROLZONE(Zone, Speed, Altitude, Formation)

**[AIR, GROUND, NAVAL]** Create a PATROLZONE mission.

Group(s) will go to the zone and patrol it randomly.

## Defined in:

AUFTAG

## Parameters:

Core.Zone#ZONE Zone

The patrol zone.

---

#number **Speed**

Speed in knots.

---

#number **Altitude**

Altitude in feet. Only for airborne units. Default 2000 feet ASL.

---

#string **Formation**

Formation used by ground units during patrol. Default "Off Road".

## Return value:

#AUFTAG:

---

self



**[AIR]** Create an enhanced orbit race track mission.

Planes will keep closer to the track.

## Defined in:

AUFRAG

## Parameters:

### Core.Point#COORDINATE Coordinate

Where to start the race track.

---

### #number Altitude

(Optional) Altitude in feet. Defaults to 20,000ft ASL.

---

### #number Speed

(Optional) Speed in knots. Defaults to 300kn TAS.

---

### #number Heading

(Optional) Heading in degrees, 0 to 360. Defaults to 90 degree (East).

---

### #number Leg

(Optional) Leg of the race track in NM. Defaults to 10nm.

---

### #number Formation

(Optional) Formation to take, e.g. ENUMS.Formation.FixedWing.Trail.Close, also see [Hoggit Wiki](#).



self

## AUFTAG:NewREARMING(Zone)

**[GROUND]** Create a REARMING mission.

### Defined in:

AUFTAG

### Parameter:

Core.Zone#ZONE Zone

The zone, where units go and look for ammo supply.

### Return value:

#AUFTAG:

self

## AUFTAG:NewRECON(ZoneSet, Speed, Altitude, Ad infinitum, Randomly, Formation)

**[AIR, GROUND, NAVAL]** Create a RECON mission.

### Defined in:

AUFTAG

### Parameters:



---

**#number Speed**

Speed in knots.

---

**#number Altitude**

Altitude in feet. Only for airborne units. Default 2000 feet ASL.

---

**#boolean Adinfinitum**

If true, the group will start over again after reaching the final zone.

---

**#boolean Randomly**

If true, the group will select a random zone.

---

**#string Formation**

Formation used during recon route.

---

## Return value:

**#AUFTRAG:**

self

---

**AUFTRAG:NewRECOVERYTANKER(Carrier, Altitude, Speed, Leg, RelHeading, OffsetDist, OffsetAngle, UpdateDistance)**

**[AIRPANE]** Create a RECOVERY TANKER mission.



## Parameters:

### Wrapper.Unit#UNIT Carrier

The carrier unit.

---

#### #number Altitude

Orbit altitude in feet. Default is 6,000 ft.

---

#### #number Speed

Orbit speed in knots. Default 250 KIAS.

---

#### #number Leg

Length of race-track in NM. Default 14 NM.

---

#### #number RelHeading

Relative heading [0, 360) of race-track pattern in degrees wrt heading of the carrier. Default is heading of the carrier.

---

#### #number OffsetDist

Relative distance of the first race-track point wrt to the carrier. Default 6 NM.

---

#### #number OffsetAngle

Relative angle of the first race-track point wrt. to the carrier. Default 180 (behind the boat).

---

#### #number UpdateDistance

Threshold distance in NM before orbit pattern is updated. Default 5 NM.

---



self

## AUFRAG:NewRESCUEHELO(Carrier)

**[AIR ROTARY]** Create a RESCUE HELO mission.

### Defined in:

AUFRAG

### Parameter:

Wrapper.Unit#UNIT Carrier

The carrier unit.

### Return value:

#AUFRAG:

self

## AUFRAG:NewSEAD(Target, Altitude)

**[AIR]** Create a SEAD mission.

### Defined in:

AUFRAG

### Parameters:



---

**#number Altitude**

Engage altitude in feet. Default 25000 ft.

---

## Return value:

**#AUFTRAG:**

self

---

# AUFTRAG:NewSTRAFING(Target, Altitude, Length)

**[AIR]** Create a STRAFING mission.

Assigns a point on the ground for which the AI will do a strafing run with guns or rockets. See [DCS task strafing](#).

## Defined in:

AUFTRAG

## Parameters:

**Core.Point#COORDINATE Target**

Target coordinate. Can also be specified as a GROUP, UNIT, STATIC or TARGET object.

---

**#number Altitude**

Engage altitude in feet. Default 1000 ft.

---

**#number Length**

The total length of the strafing target in meters. Default nil.

---



self

## AUFTAG:NewSTRIKE(Target, Altitude, EngageWeaponType)

**[AIR]** Create a STRIKE mission.

Flight will attack the closest map object to the specified coordinate.

### Defined in:

AUFTAG

### Parameters:

Core.Point#COORDINATE Target

The target coordinate. Can also be given as a GROUP, UNIT, STATIC or TARGET object.

#number **Altitude**

Engage altitude in feet. Default 2000 ft.

#number **EngageWeaponType**

Which weapon to use. Defaults to auto, ie ENUMS.WeaponFlag.Auto. See ENUMS.WeaponFlag for options.

### Return value:

#AUFTAG:

self



**[AIR]** Create a TANKER mission.

## Defined in:

AUFRAG

## Parameters:

### Core.Point#COORDINATE Coordinate

Where to orbit.

---

#### #number Altitude

Orbit altitude in feet. Default is y component of Coordinate.

---

#### #number Speed

Orbit indicated airspeed in knots at the set altitude ASL. Default 350 KIAS.

---

#### #number Heading

Heading of race-track pattern in degrees. Default 270 (East to West).

---

#### #number Leg

Length of race-track in NM. Default 10 NM.

---

#### #number RefuelSystem

Refueling system (0=boom, 1=probe). This info is *only* for AIRWINGs so they launch the right tanker type.

---

## Return value:



# AUFTAG:NewTROOPTRANSPORT(TransportGroupSet, DropoffCoordinate, PickupCoordinate, PickupRadius)

**[AIR ROTARY, GROUND]** Create a TROOP TRANSPORT mission.

## Defined in:

AUFTAG

## Parameters:

### Core.Set#SET\_GROUP **TransportGroupSet**

The set group(s) to be transported.

### Core.Point#COORDINATE **DropoffCoordinate**

Coordinate where the helo will land drop off the the troops.

### Core.Point#COORDINATE **PickupCoordinate**

Coordinate where the helo will land to pick up the the cargo. Default is the first transport group.

### #number **PickupRadius**

Radius around the pickup coordinate in meters. Default 100 m.

## Return value:

### #AUFTAG:

self



On after "Cancel" event.

## Defined in:

AUFTAG

## Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

## AUFTAG:OnAfterDone(From, Event, To)

On after "Done" event.

## Defined in:

AUFTAG

## Parameters:

#string **From**

From state.

---

#string **Event**



To state.

## AUFTAG:OnAfterExecuting(From, Event, To)

On after "Executing" event.

### Defined in:

AUFTAG

### Parameters:

#string **From**

From state.

#string **Event**

Event.

#string **To**

To state.

## AUFTAG:OnAfterFailed(From, Event, To)

On after "Failed" event.

### Defined in:

AUFTAG

### Parameters:

#string **Event**

Event.

#string **To**

To state.

## AUFRAG:OnAfterPlanned(From, Event, To)

On after "Planned" event.

### Defined in:

AUFRAG

### Parameters:

#string **From**

From state.

#string **Event**

Event.

#string **To**

To state.

## AUFRAG:OnAfterQueued(From, Event, To)

On after "Queued" event.



## Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

---

## AUFRAG:OnAfterRepeat(From, Event, To)

On after "Repeat" event.

## Defined in:

AUFRAG

## Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**



## AUFTAG:OnAfterRequested(From, Event, To)

On after "Requested" event.

### Defined in:

AUFTAG

### Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

## AUFTAG:OnAfterScheduled(From, Event, To)

On after "Scheduled" event.

### Defined in:

AUFTAG

### Parameters:

#string **From**

From state.



---

#string **To**

To state.

## AUFRAG:OnAfterStarted(From, Event, To)

On after "Started" event.

### Defined in:

AUFRAG

### Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

## AUFRAG:OnAfterSuccess(From, Event, To)

On after "Success" event.

### Defined in:

AUFRAG



From state.

---

#string **Event**

Event.

---

#string **To**

To state.

---

## AUFRAG:OnEventUnitLost(EventData)

Unit lost event.

### Defined in:

AUFRAG

### Parameter:

Core.Event#EVENTDATA **EventData**

---

Event data.

## AUFRAG:Planned()

Triggers the FSM event "Planned".

### Defined in:

AUFRAG



## Defined in:

AUFTAG

# AUFTAG:RemoveLegion(Legion)

Remove LEGION from mission.

## Defined in:

AUFTAG

## Parameter:

Ops.Legion#LEGION Legion

The legion.

## Return value:

#AUFTAG:

self

# AUFTAG:Repeat()

Triggers the FSM event "Repeat".

## Defined in:

AUFTAG



## Defined in:

AUFRAG

### AUFRAG:Scheduled()

Triggers the FSM event "Scheduled".

## Defined in:

AUFRAG

### AUFRAG:SetAlarmstate(Alarmstate)

Set alarm state for this mission.

## Defined in:

AUFRAG

## Parameter:

#number **Alarmstate**

Alarm state 0=Auto, 1=Green, 2=Red.

## Return value:

#AUFRAG:

self



Default is use of afterburner is allowed.

## Defined in:

AUFTAG

## Return value:

#AUFTAG:

self

# AUFTAG:SetAllowAfterburnerExecutePhase()

Set that (jet) aircraft are allowed to use afterburner in mission execution phase.

Default is use of afterburner is allowed.

## Defined in:

AUFTAG

## Return value:

#AUFTAG:

self

# AUFTAG:SetAssetsStayAlive(Switch)

**[LEGION, COMMANDER, CHIEF]** Set that only alive (spawned) assets are considered.

## Defined in:

#boolean **Switch**

If true or nil, only active assets. If false

## Return value:

#AUFTRAG:

self

# AUFTRAG:SetDuration(Duration)

Set time how long the mission is executed.

Once this time limit has passed, the mission is cancelled.

## Defined in:

AUFTRAG

## Parameter:

#number **Duration**

Duration in seconds.

## Return value:

#AUFTRAG:

self

# AUFTRAG:SetEPLRS(OnOffSwitch)



AUFTAG

## Parameter:

#boolean **OnOffSwitch**

If true or nil, EPLRS is on. If false, EPLRS is off.

---

## Return value:

#AUFTAG:

self

# AUFTAG:SetEmission(OnOffSwitch)

Set emission setting for this mission.

## Defined in:

AUFTAG

## Parameter:

#boolean **OnOffSwitch**

If true or nil, emission is on. If false, emission is off.

---

## Return value:

#AUFTAG:

self



## Defined in:

AUFRAG

## Parameter:

#number **Coalition**

The coalition side to which the markers are displayed. Default is to all.

## Return value:

#AUFRAG:

self

# AUFRAG:SetEngageAltitude(Altitude)

Set engage altitude.

This is the altitude passed to the DCS task. In the ME it is the tickbox ALTITUDE ABOVE.

## Defined in:

AUFRAG

## Parameter:

#string **Altitude**

Altitude in feet. Default 6000 ft.

## Return value:



## AUFRAG:SetEngageAsGroup(Switch)

Set whether target will be attack as group.

### Defined in:

AUFRAG

### Parameter:

#boolean **Switch**

If true or nil, engage as group. If false, not.

### Return value:

#AUFRAG:

self

## AUFRAG:SetEngageDetected(RangeMax, TargetTypes, EngageZoneSet, NoEngageZoneSet)

Enable to automatically engage detected targets.

### Defined in:

AUFRAG

### Parameters:

#number **RangeMax**



### #table **TargetTypes**

Types of target attributes that will be engaged. See [DCS enum attributes](#). Default "All".

#### Core.Set#SET\_ZONE EngageZoneSet

Set of zones in which targets are engaged. Default is anywhere.

#### Core.Set#SET\_ZONE NoEngageZoneSet

Set of zones in which targets are *not* engaged. Default is nowhere.

## Return value:

### #AUFTRAG:

self

## AUFTRAG:SetEvaluationTime(Teval)

Set time interval between mission done and success/failure evaluation.

## Defined in:

AUFTRAG

## Parameter:

### #number **Teval**

Time in seconds before the mission result is evaluated. Default depends on mission type.

## Return value:



## AUFRAG:SetFormation(Formation)

Set formation for this mission.

### Defined in:

AUFRAG

### Parameter:

#number **Formation**

Formation.

### Return value:

#AUFRAG:

self

## AUFRAG:SetGroupEgressWaypointUID(opsgroup, waypointindex)

Set Egress waypoint UID for OPS group.

### Defined in:

AUFRAG

### Parameters:

Ops.OpsGroup#OPSGROUP **opsgroup**



Waypoint UID.

## Return value:

#AUFTRAG:

self

## AUFTRAG:SetGroupStatus(opsgroup, status)

Set opsgroup mission status.

## Defined in:

AUFTRAG

## Parameters:

Ops.OpsGroup#OPSGROUP **opsgroup**

The flight group.

**#string status**

New status.

## Return value:

#AUFTRAG:

self



Set mission (ingress) waypoint coordinate for OPS group.

## Defined in:

AUFRAG

## Parameters:

[Ops.OpsGroup#OPSGROUP](#) **opsgroup**

The OPS group.

[Core.Point#COORDINATE](#) **coordinate**

Waypoint Coordinate.

## Return value:

#AUFRAG:

self

# AUFRAG:SetGroupWaypointIndex(opsgroup, waypointindex)

Set mission (ingress) waypoint UID for OPS group.

## Defined in:

AUFRAG

## Parameters:



---

#number **waypointindex**

Waypoint UID.

---

## Return value:

AUFTAG:

self

---

## AUFTAG:SetGroupWaypointTask(opsgroup, task)

Set mission waypoint task for OPS group.

## Defined in:

AUFTAG

## Parameters:

Ops.OpsGroup#OPSGROUP **opsgroup**

The OPS group.

---

Ops.OpsGroup#OPSGROUP.Task **task**

Waypoint task.

---

## AUFTAG:SetICLS(Channel, Morse, UnitName)

Set ICLS beacon channel and Morse code for this mission.



## Parameters:

#number **Channel**

ICLS channel.

---

#string **Morse**

Morse code. Default "XXX".

---

#string **UnitName**

Name of the unit in the group for which acts as ICLS beacon. Default is the first unit in the group.

---

## Return value:

AUFRAG:

self

---

## AUFRAG:SetImmortal(OnOffSwitch)

Set immortality setting for this mission.

## Defined in:

AUFRAG

## Parameter:

#boolean **OnOffSwitch**

If true or nil, immortal is on. If false, immortal is off.

self

## AUFTAG:SetIngressCoordinate(coordinate)

[Air] Set mission (ingress) waypoint coordinate for FLIGHT group.

### Defined in:

AUFTAG

### Parameter:

Core.Point#COORDINATE **coordinate**

Waypoint Coordinate.

### Return value:

#AUFTAG:

self

## AUFTAG:SetInvisible(OnOffSwitch)

Set invisibility setting for this mission.

### Defined in:

AUFTAG

### Parameter:



## Return value:

#AUFTRAG:

self

## AUFTAG:SetLegionStatus(Legion, Status)

Set LEGION mission status.

## Defined in:

AUFTAG

## Parameters:

Ops.Legion#LEGION Legion

The legion.

#string **Status**

New status.

## Return value:

#AUFTRAG:

self

## AUFTAG:SetMissionAltitude(Altitude)



## Defined in:

AUFRAG

## Parameter:

#string **Altitude**

Altitude in feet.

## Return value:

#AUFRAG:

self

# AUFRAG:SetMissionEgressCoord(Coordinate, Altitude, Speed)

Set the mission egress coordinate.

This is the coordinate where the assigned group will go once the mission is finished.

## Defined in:

AUFRAG

## Parameters:

Core.Point#COORDINATE Coordinate

Egress coordinate.

#number **Altitude**



(Optional) Speed in knots to reach this waypoint. Defaults to mission speed.

## Return value:

**#AUFTRAG:**

self

# AUFTRAG:SetMissionHoldingCoord(Coordinate, Altitude, Speed, Duration)

[Air] Set the mission holding coordinate.

This is the coordinate where the assigned group will fly before the actual mission execution starts.  
Do not forget to add a push condition, too!

## Defined in:

AUFTRAG

## Parameters:

**Core.Point#COORDINATE Coordinate**

Holding coordinate.

**#number Altitude**

(Optional) Altitude in feet. Default is y component of coordinate.

**#number Speed**

(Optional) Speed in knots to reach this waypoint and hold there. Defaults to mission speed.



continues if either a push condition is met or the time is up.

## Return value:

#AUFTRAG:

self

# AUFTRAG:SetMissionIngressCoord(Coordinate, Altitude, Speed)

[Air] Set the mission ingress coordinate.

This is the coordinate where the assigned group will fly before the actual mission coordinate.

## Defined in:

AUFTRAG

## Parameters:

Core.Point#COORDINATE Coordinate

Ingress coordinate.

#number **Altitude**

(Optional) Altitude in feet. Default is y component of coordinate.

#number **Speed**

(Optional) Speed in knots to reach this waypoint. Defaults to mission speed.

## Return value:



## AUFRAG:SetMissionRange(Range)

Set max mission range.

Only applies if the AUFRAG is handled by an AIRWING or CHIEF. This is the max allowed distance from the airbase to the target.

### Defined in:

AUFRAG

### Parameter:

#number **Range**

Max range in NM. Default 100 NM.

### Return value:

#AUFRAG:

self

## AUFRAG:SetMissionSpeed(Speed)

Set mission speed.

That is the speed the group uses to get to the mission waypoint.

### Defined in:

AUFRAG

### Parameter:



## Return value:

#AUFTRAG:

self

## AUFTAG:SetMissionWaypointCoord(Coordinate)

[NON-AIR] Set the mission waypoint coordinate from where the mission is executed.

Note that altitude is set via :SetMissionAltitude.

## Defined in:

AUFTAG

## Parameter:

Core.Point#COORDINATE Coordinate

Coordinate where the mission is executed.

## Return value:

#AUFTRAG:

self

## AUFTAG:SetMissionWaypointRandomization(Radius)

Set randomization of the mission waypoint coordinate.

Each assigned group will get a random ingress coordinate, where the mission is executed.



## Parameter:

#number **Radius**

Distance in meters. Default #nil.

---

## Return value:

#AUFTRAG:

self

## AUFTRAG:SetName(Name)

Set mission name.

## Defined in:

AUFTRAG

## Parameter:

#string **Name**

Name of the mission. Default is "Auftrag Nr. X", where X is a running number, which is automatically increased.

---

## Return value:

#AUFTRAG:

self



Mission assets will be transported before the mission is started at the OPSGROUP level.

## Defined in:

AUFTRAG

## Parameter:

[Ops.OpsTransport#OPSTRANSPORT](#) **OpsTransport**

The OPS transport assignment attached to the mission.

## Return value:

[#AUFTRAG](#):

self

## AUFTRAG:SetPriority(Prio, Urgent, Importance)

Set mission priority and (optional) urgency.

Urgent missions can cancel other running missions.

## Defined in:

AUFTRAG

## Parameters:

#number **Prio**

Priority 1=high, 100=low. Default 50.

#boolean **Urgent**



Number 1-10. If missions with lower value are in the queue, these have to be finished first.  
Default is nil.

## Return value:

#AUFTRAG:

self

## AUFTRAG:SetProhibitAfterburner()

Set that (jet) aircraft are generally **not** allowed to use afterburner.

Default is use of afterburner is allowed.

## Defined in:

AUFTRAG

## Return value:

#AUFTRAG:

self

## AUFTRAG:SetProhibitAfterburnerExecutePhase()

Set that (jet) aircraft are **not** allowed to use afterburner in mission execution phase.

Default is use of afterburner is allowed.

## Defined in:

AUFTRAG



self

## AUFRAG:SetPushTime(ClockPush)

Set mission push time.

This is the time the mission is executed. If the push time is not passed, the group will wait at the mission execution waypoint.

### Defined in:

AUFRAG

### Parameter:

#string **ClockPush**

Time the mission is executed, e.g. "05:00" for 5 am. Can also be given as a #number, where it is interpreted as relative push time in seconds.

### Return value:

#AUFRAG:

self

## AUFRAG:SetROE(roe)

Set Rules of Engagement (ROE) for this mission.

### Defined in:

AUFRAG



Mission ROE, e.g. ENUMS.ROE.ReturnFire (whiche equals 3)

## Return value:

#AUFTRAG:

self

## AUFTRAG:SetROT(rot)

Set Reaction on Threat (ROT) for this mission.

## Defined in:

AUFTRAG

## Parameter:

#number **rot**

Mission ROT, e.g. ENUMS.ROT.NoReaction (whiche equals 0)

## Return value:

#AUFTRAG:

self

## AUFTRAG:SetRadio(Frequency, Modulation)

Set radio frequency and modulation for this mission.



## Parameters:

#number **Frequency**

Frequency in MHz.

---

#number **Modulation**

Radio modulation. Default 0=AM.

---

## Return value:

AUFRAG:

self

---

## AUFRAG:SetReinforce(Nreinforce)

**[LEGION, COMMANDER, CHIEF]** Set that mission assets get reinforced if their number drops below the minimum number of required assets of the mission (*c.f.* SetRequiredAssets() function).

**Note** that reinforcement groups are only recruited from the legion (airwing, brigade, fleet) the mission was assigned to. If the legion does not have any more of these assets, no reinforcement can take place, even if the mission is submitted to a COMMANDER or CHIEF.

## Defined in:

AUFRAG

## Parameter:

#number **Nreinforce**

Number of max asset groups used to reinforce.

---



self

## AUFTAG:SetRepeat(Nrepeat)

**[LEGION, COMMANDER, CHIEF]** Set how many times the mission is repeated.

Only valid if the mission is handled by a LEGION (AIRWING, BRIGADE, FLEET) or higher level.

### Defined in:

AUFTAG

### Parameter:

#number **Nrepeat**

Number of repeats. Default 0.

### Return value:

[#AUFTAG:](#)

self

## AUFTAG:SetRepeatOnFailure(Nrepeat)

**[LEGION, COMMANDER, CHIEF]** Set how many times the mission is repeated if it fails.

Only valid if the mission is handled by a LEGION (AIRWING, BRIGADE, FLEET) or higher level.

### Defined in:

AUFTAG



Number of repeats. Default 0.

## Return value:

#AUFTRAG:

self

# AUFTRAG:SetRepeatOnSuccess(Nrepeat)

**[LEGION, COMMANDER, CHIEF]** Set how many times the mission is repeated if it was successful.

Only valid if the mission is handled by a LEGION (AIRWING, BRIGADE, FLEET) or higher level.

## Defined in:

AUFTRAG

## Parameter:

#number **Nrepeat**

Number of repeats. Default 0.

## Return value:

#AUFTRAG:

self



**[LEGION, COMMANDER, CHIEF]** Define how many assets are required to do the job.

Only used if the mission is handled by a **LEGION** (AIRWING, BRIGADE, ...) or higher level.

## Defined in:

AUFRAG

## Parameters:

#number **NassetsMin**

Minimum number of asset groups. Default 1.

---

#number **NassetsMax**

Maximum Number of asset groups. Default is same as NassetsMin.

---

## Return value:

AUFRAG:

self

---

# AUFRAG:SetRequiredAttribute(Attributes)

**[LEGION, COMMANDER, CHIEF]** Set required attribute(s) the assets must have.

## Defined in:

AUFRAG

## Parameter:



## Return value:

**#AUFTRAG:**

self

# AUFTAG:SetRequiredCarriers(NcarriersMin, NcarriersMax, Categories, Attributes, Properties)

**[LEGION, COMMANDER, CHIEF]** Set number of required carrier groups if an OPSTRANSORT assignment is required.

## Defined in:

AUFTAG

## Parameters:

**#number NcarriersMin**

Number of carriers *at least* required. Default 1.

**#number NcarriersMax**

Number of carriers *at most* used for transportation. Default is same as NcarriersMin.

**#table Categories**

Group categories.

**#table Attributes**

Group attributes. See GROUP.Attribute.



## Return value:

#AUFTRAG:

self

# AUFTRAG:SetRequiredEscorts(NescortMin, NescortMax, MissionType, TargetTypes, EngageRange)

**[LEGION, COMMANDER, CHIEF]** Define how many assets are required that escort the mission assets.

Only used if the mission is handled by a **LEGION** (AIRWING, BRIGADE, FLEET) or higher level.

## Defined in:

AUFTRAG

## Parameters:

#number **NescortMin**

Minimum number of asset groups. Default 1.

#number **NescortMax**

Maximum Number of asset groups. Default is same as NassetsMin.

#string **MissionType**

Mission type assets will be optimized for and payload selected, e.g. AUFTRAG.Type.SEAD. Default nil.



{"Ground Units"} for helos. Set, e.g., {"Air Defence"} for SEAD.

### #number **EngageRange**

Max range in nautical miles that the escort group(s) will engage enemies. Default 32 NM (60 km).

## Return value:

#AUFTRAG:

self

# AUFTRAG:SetRequiredProperty(Properties)

**[LEGION, COMMANDER, CHIEF]** Set required property or properties the assets must have.

These are DCS attributes.

## Defined in:

AUFTRAG

## Parameter:

#table **Properties**

Property or table of properties.

## Return value:

#AUFTRAG:

self



# Categories, Attributes, Properties)

**[LEGION, COMMANDER, CHIEF]** Attach OPS transport to the mission.

Mission assets will be transported before the mission is started at the OPSGROUP level.

## Defined in:

AUFTRAG

## Parameters:

### Core.Zone#ZONE DeployZone

Zone where assets are deployed.

---

#### #number **NcarriersMin**

Number of carriers *at least* required. Default 1.

---

#### #number **NcarriersMax**

Number of carriers *at most* used for transportation. Default is same as NcarriersMin.

---

### Core.Zone#ZONE DisembarkZone

Zone where assets are disembarked to.

---

### #table **Categories**

Group categories.

---

### #table **Attributes**

Generalizes group attributes.



## Return value:

#AUFTRAG:

self

## AUFTRAG:SetReturnToLegion(Switch)

**[LEGION, COMMANDER, CHIEF]** Set whether assigned assets return to their legion once the mission is over.

This is only applicable to **army** and **navy** groups, *i.e.* aircraft will always return.

## Defined in:

AUFTRAG

## Parameter:

#boolean **Switch**

If true, assets will return. If false, assets will not return and stay where it finishes its last mission. If nil, let asset decide.

## Return value:

#AUFTRAG:

self

## AUFTRAG:SetTACAN(Channel, Morse, UnitName, Band)



## AUFTAG

### Parameters:

#number **Channel**

TACAN channel.

---

#string **Morse**

Morse code. Default "XXX".

---

#string **UnitName**

Name of the unit in the group for which acts as TACAN beacon. Default is the first unit in the group.

---

#string **Band**

Tacan channel mode ("X" or "Y"). Default is "X" for ground/naval and "Y" for aircraft.

---

### Return value:

#AUFTAG:

self

---

## AUFTAG:SetTeleport(Switch)

Set that mission assets are teleported to the mission execution waypoint.

### Defined in:

AUFTAG



If true or nil, teleporting is on. If false, teleporting is off.

## Return value:

#AUFTRAG:

self

# AUFTRAG:SetTime(ClockStart, ClockStop)

Set mission start and stop time.

## Defined in:

AUFTRAG

## Parameters:

#string **ClockStart**

Time the mission is started, e.g. "05:00" for 5 am. If specified as a #number, it will be relative (in seconds) to the current mission time. Default is 5 seconds after mission was added.

#string **ClockStop**

(Optional) Time the mission is stopped, e.g. "13:00" for 1 pm. If mission could not be started at that time, it will be removed from the queue. If specified as a #number it will be relative (in seconds) to the current mission time.

## Return value:

#AUFTRAG:



## AUFRAG:SetVerbosity(VerbosityLevel)

Set verbosity level.

### Defined in:

AUFRAG

### Parameter:

#number **VerbosityLevel**

Level of output (higher=more). Default 0.

### Return value:

[#AUFRAG:](#)

self

## AUFRAG:SetWeaponExpend(WeaponExpend)

Set number of weapons to expend.

### Defined in:

AUFRAG

### Parameter:

#number **WeaponExpend**

How much of the weapon load is expended during the attack, e.g.  
AI.Task.WeaponExpend.ALL. Default "Auto".



self

## AUFRAG:SetWeaponType(WeaponType)

Set weapon type used for the engagement.

### Defined in:

AUFRAG

### Parameter:

#number **WeaponType**

Weapon type. Default is ENUMS.WeaponFlag.Auto.

### Return value:

[#AUFRAG:](#)

self

## AUFRAG:Started()

Triggers the FSM event "Started".

### Defined in:

AUFRAG

## AUFRAG:Status()



AUFTRAG

## AUFTAG:Stop()

Triggers the FSM event "Stop".

### Defined in:

AUFTRAG

## AUFTAG:Success()

Triggers the FSM event "Success".

### Defined in:

AUFTRAG

## AUFTAG:UpdateMarker()

Update mission F10 map marker.

### Defined in:

AUFTRAG

### Return value:

#AUFTAG:

self



## Defined in:

AUFRAG

## Parameter:

#table **Assets**

List of assets.

## Return value:

#AUFRAG:

self

# AUFRAG:\_DetermineAuftragType(Target)

Create a mission to attack a group.

Mission type is automatically chosen from the group category.

## Defined in:

AUFRAG

## Parameter:

Wrapper.Positionable#POSITIONABLE Target

Target object.

## Return value:



## AUFRAG:\_GetDCSAttackTask(Target, DCStasks)

Get DCS task table for an attack group or unit task.

### Defined in:

AUFRAG

### Parameters:

Ops.Target#TARGET Target

Target data.

#table **DCStasks**

DCS DCS tasks table to which the task is added.

### Return value:

DCS#Task:

The DCS task table.

## AUFRAG:\_GetMissionWaypointCoordSet()

Get coordinate which was set as mission waypoint coordinate.

### Defined in:

AUFRAG



Coordinate where the mission is executed or #nil.

## AUFRAG:\_GetRequest(Legion)

Get request from legion this mission requested assets from.

### Defined in:

AUFRAG

### Parameter:

[Ops.Legion#LEGION Legion](#)

The legion from which to get the request ID.

### Return value:

[Functional.Warehouse#WAREHOUSE.PendingItem](#):

Request.

## AUFRAG:\_GetRequestID(Legion)

Get request ID from legion this mission requested assets from

### Defined in:

AUFRAG

### Parameter:



## Return value:

#number:

Request ID (if any).

## AUFRAG:\_IsNotReinforcing()

Check if reinforcement is done.

### Defined in:

AUFRAG

## Return value:

#boolean:

If true, reinforcing is over.

## AUFRAG:\_IsReinforcing()

Check if reinforcement is still ongoing.

### Defined in:

AUFRAG

## Return value:

#boolean:

If true, reinforcing is ongoing.



**[PRIVATE, AIR, GROUND, NAVAL]** Create a mission to relocate all cohort assets to another LEGION.

## Defined in:

AUFRAG

## Parameters:

### Ops.Legion#LEGION Legion

The new legion.

---

### Ops.Cohort#COHORT Cohort

The cohort to be relocated.

---

## Return value:

#AUFRAG:

self

## AUFRAG:\_SetLogID()

Set log ID string.

## Defined in:

AUFRAG

## Return value:

#AUFRAG:



## AUFTAG:\_SetRequestID(Legion, RequestID)

Set request ID from legion this mission requested assets from

### Defined in:

AUFTAG

### Parameters:

#### Ops.Legion#LEGION Legion

The legion from which to get the request ID.

---

#### #number RequestID

Request ID.

---

### Return value:

#### #AUFTAG:

self

---

## AUFTAG:\_TargetFromObject(Object)

Create target data from a given object.

### Defined in:

AUFTAG

### Parameter:

#### Wrapper.Positionable#POSITIONABLE Object



## AUFRAG:\_Cancel(delay)

Triggers the FSM event "Cancel" after a delay.

### Defined in:

AUFRAG

### Parameter:

#number **delay**

Delay in seconds.

## AUFRAG:\_Done(delay)

Triggers the FSM event "Done" after a delay.

### Defined in:

AUFRAG

### Parameter:

#number **delay**

Delay in seconds.

## AUFRAG:\_Executing(delay)

Triggers the FSM event "Executing" after a delay.

### Defined in:



#number **delay**

Delay in seconds.

## AUFRAG:\_Failed(delay)

Triggers the FSM event "Failed" after a delay.

### Defined in:

AUFRAG

### Parameter:

#number **delay**

Delay in seconds.

## AUFRAG:\_Planned(delay)

Triggers the FSM event "Planned" after a delay.

### Defined in:

AUFRAG

### Parameter:

#number **delay**

Delay in seconds.



## Defined in:

AUFRAG

## Parameter:

#number **delay**

Delay in seconds.

## AUFRAG:\_\_Repeat(delay)

Triggers the FSM event "Repeat" after a delay.

## Defined in:

AUFRAG

## Parameter:

#number **delay**

Delay in seconds.

## AUFRAG:\_\_Requested(delay)

Triggers the FSM event "Requested" after a delay.

## Defined in:

AUFRAG



Delay in seconds.

## AUFRAG:\_Scheduled(delay)

Triggers the FSM event "Scheduled" after a delay.

### Defined in:

AUFRAG

### Parameter:

#number **delay**

Delay in seconds.

## AUFRAG:\_Started(delay)

Triggers the FSM event "Started" after a delay.

### Defined in:

AUFRAG

### Parameter:

#number **delay**

Delay in seconds.

## AUFRAG:\_Status(delay)



AUFTAG

## Parameter:

#number **delay**

Delay in seconds.

## AUFTAG:\_Stop(delay)

Triggers the FSM event "Stop" after a delay.

## Defined in:

AUFTAG

## Parameter:

#number **delay**

Delay in seconds.

## AUFTAG:\_Success(delay)

Triggers the FSM event "Success" after a delay.

## Defined in:

AUFTAG

## Parameter:

#number **delay**



## AUFRAG:onafterAssetDead(From, Event, To, Asset)

On after "AssetDead" event.

### Defined in:

AUFRAG

### Parameters:

#string **From**

From state.

#string **Event**

Event.

#string **To**

To state.

[Functional.Warehouse#WAREHOUSE.Assetitem](#) **Asset**

The asset.

## AUFRAG:onafterAssign(From, Event, To)

On after "Assign" event.

### Defined in:

AUFRAG

### Parameters:



---

#string **Event**

Event.

---

#string **To**

To state.

---

## AUFRAG:onafterCancel(From, Event, To)

On after "Cancel" event.

Cancells the mission.

### Defined in:

AUFRAG

### Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

---

## AUFRAG:onafterDone(From, Event, To)



## AUFTAG

## Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

---

## AUFTAG:onafterElementDestroyed(From, Event, To, OpsGroup, Element)

On after "ElementDestroyed" event.

## Defined in:

AUFTAG

## Parameters:

#string **From**

From state.

---

#string **Event**

Event.



### Ops.OpsGroup#OPSGROUP **OpsGroup**

The ops group to which the element belongs.

### Ops.OpsGroup#OPSGROUP.Element **Element**

The element that got destroyed.

## AUFRAG:onafterExecuting(From, Event, To)

On after "Execute" event.

### Defined in:

AUFRAG

### Parameters:

#string **From**

From state.

#string **Event**

Event.

#string **To**

To state.

## AUFRAG:onafterFailed(From, Event, To)

On after "Failed" event.



## Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

---

## AUFRAG:onafterGroupDead(From, Event, To, OpsGroup)

On after "GroupDead" event.

## Defined in:

AUFRAG

## Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---



### Ops.OpsGroup#OPSGROUP **OpsGroup**

The ops group that is dead now.

## AUFRAG:onafterPlanned(From, Event, To)

On after "Planned" event.

### Defined in:

AUFRAG

### Parameters:

#string **From**

From state.

#string **Event**

Event.

#string **To**

To state.

## AUFRAG:onafterQueued(From, Event, To, Airwing)

On after "Queue" event.

Mission is added to the mission queue of a LEGION.

### Defined in:



#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

---

**Airwing**

## AUFRAG:onafterRepeat(From, Event, To)

On after "Repeat" event.

### Defined in:

AUFRAG

### Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**



## AUFTAG:onafterRequested(From, Event, To)

On after "Requested" event.

### Defined in:

AUFTAG

### Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

## AUFTAG:onafterScheduled(From, Event, To)

On after "Schedule" event.

Mission is added to the mission queue of an OPSGROUP.

### Defined in:

AUFTAG

### Parameters:

#string **From**

Event.

---

#string **To**

To state.

## AUFRAG:onafterStarted(From, Event, To)

On after "Start" event.

### Defined in:

AUFRAG

### Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

## AUFRAG:onafterStatus(From, Event, To)

On after "Status" event.



## Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

---

## AUFRAG:onafterStop(From, Event, To)

On after "Stop" event.

Remove mission from LEGION and OPSGROUP mission queues.

## Defined in:

AUFRAG

## Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**



## AUFTAG:onafterSuccess(From, Event, To)

On after "Success" event.

### Defined in:

AUFTAG

### Parameters:

#string **From**

From state.

---

#string **Event**

Event.

---

#string **To**

To state.

## AUFTAG:onbeforeRepeat(From, Event, To)

On before "Repeat" event.

### Defined in:

AUFTAG

### Parameters:

#string **From**

From state.



#string **To**

To state.

## Field(s)

#AUFTRAG.Category **AUFTRAG.Category**

#string **AUFTRAG.ClassName**

Name of the class.

#table **AUFTRAG.DCStask**

DCS task structure.

#AUFTRAG.GroupStatus **AUFTRAG.GroupStatus**

#number **AUFTRAG.Nassets**

Number of requested warehouse assets.

#table **AUFTRAG.NassetsLegMax**

Number of required warehouse assets for each assigned legion.

#table **AUFTRAG.NassetsLegMin**

Number of required warehouse assets for each assigned legion.

#number **AUFTRAG.NassetsMax**

Max. number of required warehouse assets.

#number **AUFTRAG.NassetsMin**



Number of assigned groups.

#number **AUFTRAG.NcarriersMax**

Max number of required carrier assets.

#number **AUFTRAG.NcarriersMin**

Min number of required carrier assets.

#number **AUFTRAG.Ncasualties**

Number of own casualties during mission.

#number **AUFTRAG.Ndead**

Number of assigned groups that are dead.

#number **AUFTRAG.Nelements**

Number of elements (units) assigned to mission.

#number **AUFTRAG.NescortMax**

Max. number of required escort assets for each group the mission is assigned to.

#number **AUFTRAG.NescortMin**

Min. number of required escort assets for each group the mission is assigned to.

#number **AUFTRAG.Ngroups**

#number **AUFTRAG.Nkills**

Number of (enemy) units killed by assets of this mission.

#number **AUFTRAG.Nrepeat**



Number of times mission is repeated if failed.

#number **AUFTRAG.NrepeatSuccess**

Number of times mission is repeated if successful.

#AUFTRAG.SpecialTask **AUFTRAG.SpecialTask**

#AUFTRAG.TargetType **AUFTRAG.TargetType**

#number **AUFTRAG.Texecuting**

Time stamp (abs) when mission is executing. Is #nil on start.

#number **AUFTRAG.Tover**

Mission abs. time stamp, when mission was over.

#number **AUFTRAG.Tpush**

Mission push/execute time in abs. seconds.

**AUFTRAG.TrackAltitude**

**AUFTRAG.TrackFormation**

**AUFTRAG.TrackPoint1**

**AUFTRAG.TrackPoint2**

**AUFTRAG.TrackSpeed**

#number **AUFTRAG.Tstart**

Mission start time in abs. seconds.

#number **AUFTRAG.Tstarted**

Time stamp (abs) when mission is started.



## #AUFTRAG.Type **AUFTRAG.Type**

### #string **AUFTRAG.alert5MissionType**

Alert 5 mission type. This is the mission type, the alerted assets will be able to carry out.

### #number **AUFTRAG.artyAltitude**

Altitude in meters. Can be used for a Barrage.

### #number **AUFTRAG.artyAngle**

Shooting angle in degrees (for Barrage).

### #number **AUFTRAG.artyHeading**

Heading in degrees (for Barrage).

### #number **AUFTRAG.artyRadius**

Radius in meters.

### #number **AUFTRAG.artyShots**

Number of shots fired.

## **AUFTRAG.assetStayAlive**

### #table **AUFTRAG.assets**

Warehouse assets assigned for this mission.

### #table **AUFTRAG.attributes**

Generalized attribute(s) of assets.

### #number **AUFTRAG.auftragsnummer**



Generalized attribute(s) of transport assets.

## **AUFTRAG.carrierCategories**

### **#table AUFTRAG.carrierProperties**

DCS attribute(s) of transport assets.

### **#table AUFTRAG.categories**

Mission categories.

### **Ops.Chief#CHIEF AUFTRAG.chief**

The CHIEF managing this mission.

### **Ops.Commander#COMMANDER AUFTRAG.commander**

The COMMANDER managing this mission.

### **#table AUFTRAG.conditionFailure**

If all conditions are true, the mission is cancelled.

### **#boolean AUFTRAG.conditionFailureSet**

### **#table AUFTRAG.conditionPush**

If all conditions are true, the mission is executed. Before, the group(s) wait at the mission execution waypoint.

### **#table AUFTRAG.conditionStart**

Condition(s) that have to be true, before the mission will be started.

### **#table AUFTRAG.conditionSuccess**

If all conditions are true, the mission is cancelled.



Time interval in seconds before the mission result is evaluated after mission is over.

#### #number **AUFTRAG.duration**

Mission duration in seconds.

#### #number **AUFTRAG.durationExe**

Mission execution time in seconds.

#### #number **AUFTRAG.engageAltitude**

Engagement altitude in meters.

#### #boolean **AUFTRAG.engageAsGroup**

Group attack.

#### #number **AUFTRAG.engageDirection**

Engagement direction in degrees.

#### #number **AUFTRAG.engageLength**

Length of engage (carpet or strafing) in meters.

#### #number **AUFTRAG.engageMaxDistance**

Max engage distance.

#### #number **AUFTRAG.engageQuantity**

Number of times a target is engaged.

### **AUFTRAG.engageRange**

#### Ops.Target#TARGET **AUFTRAG.engageTarget**



Table of target types that are engaged in the engagement zone.

### #number **AUFTRAG.engageWeaponExpend**

How many weapons are used.

### #number **AUFTRAG.engageWeaponType**

Weapon type used.

### Core.Zone#ZONE\_RADIUS **AUFTRAG.engageZone**

*Circular* engagement zone.

### **AUFTRAG.engageddetectedEngageZones**

### **AUFTRAG.engageddetectedNoEngageZones**

### #boolean **AUFTRAG.engageddetectedOn**

Set parameters.

### **AUFTRAG.engageddetectedRmax**

### **AUFTRAG.engageddetectedTypes**

### #table **AUFTRAG.enrouteTasks**

Mission enroute tasks.

### #table **AUFTRAG.escortCohorts**

Cohorts explicitly requested for providing escorting assets.

### #number **AUFTRAG.escortEngageRange**

Engage range in nautical miles (NM).

### Wrapper.Group#GROUP **AUFTRAG.escortGroup**



Name of the escorted group.

#### #table **AUFTRAG.escortLegions**

Legions explicitly requested for providing escorting assets.

#### #string **AUFTRAG.escortMissionType**

Escort mission type.

#### #table **AUFTRAG.escortTargetTypes**

Target types that will be engaged.

#### DCS#Vec3 **AUFTRAG.escortVec3**

The 3D offset vector from the escorted group to the escort group.

#### #boolean **AUFTRAG.facDatalink**

FAC datalink enabled.

#### #number **AUFTRAG.facDesignation**

FAC designation type.

#### #number **AUFTRAG.facFreq**

FAC radio frequency in MHz.

#### #number **AUFTRAG.facModu**

FAC radio modulation 0=AM 1=FM.

#### #boolean **AUFTRAG.failurecondition**

#### #table **AUFTRAG.groupdata**



## AUFTRAG.hoverTime

Ops.OpsGroup#OPSGROUP.Beacon **AUFTRAG.icls**

ICLS setting.

## #number AUFTRAG.importance

Importance.

## #boolean AUFTRAG.legionReturn

If true, assets return to their legion (default). If false, they will stay alive.

## #table AUFTRAG.legions

Assigned legions.

## #string AUFTRAG.iid

Class id string for output to DCS log file.

Wrapper.Marker#MARKER **AUFTRAG.marker**

F10 map marker.

## #number AUFTRAG.markerCoaliton

Coalition to which the marker is displayed.

## #boolean AUFTRAG.markerOn

If true, display marker on F10 map with the AUFTRAG status.

## #number AUFTRAG.missionAltitude

Mission altitude in meters.



## AUFTRAG.missionEgressCoordAlt

#number **AUFTRAG.missionFraction**

Mission coordinate fraction. Default is 0.5.

## AUFTRAG.missionHoldingCoord

### AUFTRAG.missionHoldingCoordAlt

Core.Point#COORDINATE **AUFTRAG.missionIngressCoord**

Mission Ingress waypoint coordinate.

#number **AUFTRAG.missionRange**

Mission range in meters. Used by LEGION classes (AIRWING, BRIGADE, ...).

#number **AUFTRAG.missionSpeed**

Mission speed in km/h.

#string **AUFTRAG.missionTask**

Mission task. See ENUMS.MissionTask.

Core.Point#COORDINATE **AUFTRAG.missionWaypointCoord**

Mission waypoint coordinate.

#number **AUFTRAG.missionWaypointRadius**

Random radius in meters.

#string **AUFTRAG.name**

Mission name.

Ops.Operation#OPERATION **AUFTRAG.operation**



OPS transport assignment.

**#number AUFTRAG.optionAlarm**

Alarm state.

**#number AUFTRAG.optionCM**

Counter measures.

**#number AUFTRAG.optionECM**

ECM.

**#boolean AUFTRAG.optionEPLRS**

EPLRS datalink.

**#boolean AUFTRAG.optionEmission**

Emission is on or off.

**#number AUFTRAG.optionFormation**

Formation.

**#boolean AUFTRAG.optionImmortal**

Immortal is on/off.

**#boolean AUFTRAG.optionInvisible**

Invisible is on/off.

**#number AUFTRAG.optionROE**

ROE.

**#number AUFTRAG.optionRTBammo**

RTB on out-of-ammo.

**#number AUFTRAG.optionRTBfuel**

RTB on out-of-fuel.

**#number AUFTRAG.orbitAltitude**

Orbit altitude in meters.

**#number AUFTRAG.orbitDeltaR**

Distance threshold in meters for moving orbit targets.

**#number AUFTRAG.orbitHeading**

Orbit heading in degrees.

**#number AUFTRAG.orbitLeg**

Length of orbit leg in meters.

**DCS#Vec2 AUFTRAG.orbitOffsetVec2**

2D offset vector.

**#number AUFTRAG.orbitSpeed**

Orbit speed in m/s.

**DCS#Vec2 AUFTRAG.orbitVec2**

2D orbit vector.

**Ops.Airwing#AIRWING.PatrolData AUFTRAG.patrolData**



User specified airwing payloads for this mission. Only these will be considered for the job!

#### #number **AUFTRAG.prio**

Mission priority.

#### #boolean **AUFTRAG.prohibitAB**

#### #boolean **AUFTRAG.prohibitABExecute**

#### #table **AUFTRAG.properties**

DCS attribute(s) of assets.

#### Ops.OpsGroup#OPSGROUP.Radio **AUFTRAG.radio**

Radio freq and modulation.

#### #number **AUFTRAG.refuelSystem**

Refuel type (boom or probe) for TANKER missions.

#### **AUFTRAG.reinforce**

#### #number **AUFTRAG.repeated**

Number of times mission was repeated.

#### #number **AUFTRAG.repeatedFailure**

Number of times mission was repeated after a failure.

#### #number **AUFTRAG.repeatedSuccess**

Number of times mission was repeated after a success.

#### #table **AUFTRAG.requestID**

[REDACTED]

[REDACTED]

[REDACTED]

// ~~table AUFTRAG.transportCohorts~~

User specified cohorts assigned for this mission. Only these will be considered for the job!

### #table **AUFTRAG.specialLegions**

User specified legions assigned for this mission. Only these will be considered for the job!

### #string **AUFTRAG.status**

Mission status.

### #string **AUFTRAG.statusChief**

Mission status of the CHIEF.

### #string **AUFTRAG.statusCommander**

Mission status of the COMMANDER.

### #table **AUFTRAG.statusLegion**

Mission status of all assigned LEGIONS.

### #boolean **AUFTRAG.successcondition**

#### Ops.OpsGroup#OPSGROUP.Beacon **AUFTRAG.tacan**

TACAN setting.

### #number **AUFTRAG.targetHeading**

Heading of target in degrees.

### #boolean **AUFTRAG.teleport**

Groups are teleported to the mission ingress waypoint.

### #table **AUFTRAG.transportCohorts**



Deploy zone of an OPSTRANSPORT.

#### Core.Zone#ZONE **AUFTRAG.transportDisembarkZone**

Disembark zone of an OPSTRANSPORT.

#### Core.Point#COORDINATE **AUFTRAG.transportDropoff**

Coordinate where to drop off the cargo.

#### Core.Set#SET\_GROUP **AUFTRAG.transportGroupSet**

Groups to be transported.

#### #table **AUFTRAG.transportLegions**

Legions explicitly requested for providing transport carrier assets.

#### Core.Point#COORDINATE **AUFTRAG.transportPickup**

Coordinate where to pickup the cargo.

#### #number **AUFTRAG.transportPickupRadius**

Radius in meters for pickup zone. Default 500 m.

#### #string **AUFTRAG.type**

Mission type.

#### #boolean **AUFTRAG.updateDCSTask**

If true, DCS task is updated at every status update of the assigned groups.

#### #boolean **AUFTRAG.urgent**

Mission is urgent. Running missions with lower prio might be cancelled.



```
#string AUFTRAG.version
```

AUFTRAG class version.

## Function(s)

### AUFTRAG:AddEndState(State)

Adds an End state.

#### Defined in:

[Core.Fsm#FSM](#)

#### Parameter:

```
#string State
```

The FSM state.

### AUFTRAG:AddProcess(From, Event, Process, ReturnEvents)

Set the default `#FSM_PROCESS` template with key ProcessName providing the ProcessClass and the process object when it is assigned to a [Wrapper.Controllable](#) by the task.

#### Defined in:

[Core.Fsm#FSM](#)

#### Parameters:

```
#table From
```



#string **Event**

The Event name.

Core.Fsm#FSM\_PROCESS Process

An sub-process FSM.

#table **ReturnEvents**

A table indicating for which returned events of the SubFSM which Event must be triggered in the FSM.

## Return value:

Core.Fsm#FSM\_PROCESS:

The SubFSM.

# AUFTAG:AddScore(State, ScoreText, Score)

Adds a score for the FSM to be achieved.

## Defined in:

Core.Fsm#FSM

## Parameters:

#string **State**

is the state of the process when the score needs to be given. (See the relevant state descriptions of the process).

#string **ScoreText**



is a number providing the score of the status.

---

## Return value:

#FSM:

self

# AUFTAG:AddScoreProcess(From, Event, State, ScoreText, Score)

Adds a score for the FSM\_PROCESS to be achieved.

## Defined in:

[Core.Fsm#FSM](#)

## Parameters:

#string **From**

is the From State of the main process.

#string **Event**

is the Event of the main process.

#string **State**

is the state of the process when the score needs to be given. (See the relevant state descriptions of the process).

#string **ScoreText**



is a number providing the score of the status.

---

## Return value:

#FSM:

self

---

## AUFRAG:AddTransition(From, Event, To)

Add a new transition rule to the FSM.

A transition rule defines when and if the FSM can transition from a state towards another state upon a triggered event.

## Defined in:

[Core.Fsm#FSM](#)

## Parameters:

#table **From**

Can contain a string indicating the From state or a table of strings containing multiple From states.

---

#string **Event**

The Event name.

---

#string **To**

The To state.

---



## Defined in:

[Core.Fsm#FSM](#)

## Return value:

#string:

Current FSM state.

## AUFRAG:GetEndStates()

Returns the End states.

## Defined in:

[Core.Fsm#FSM](#)

## Return value:

#table:

End states.

## AUFRAG:GetProcess(From, Event)

## Defined in:

[Core.Fsm#FSM](#)

## Parameters:



## AUFRAG:GetProcesses()

Returns a table of the SubFSM rules defined within the FSM.

### Defined in:

[Core.Fsm#FSM](#)

### Return value:

#table:

Sub processes.

## AUFRAG:GetScores()

Returns a table with the scores defined.

### Defined in:

[Core.Fsm#FSM](#)

### Return value:

#table:

Scores.

## AUFRAG:GetStartState()

Returns the start state of the FSM.



## Return value:

#string:

A string containing the start state.

## AUFRAG:GetState()

Get current state.

## Defined in:

[Core.Fsm#FSM](#)

## Return value:

#string:

Current FSM state.

## AUFRAG:GetSubs()

Returns a table with the Subs defined.

## Defined in:

[Core.Fsm#FSM](#)

## Return value:

#table:

Sub processes.



Returns a table of the transition rules defined within the FSM.

## Defined in:

[Core.Fsm#FSM](#)

## Return value:

#table:

Transitions.

## AUFTAG:Is(State)

Check if FSM is in state.

## Defined in:

[Core.Fsm#FSM](#)

## Parameter:

#string **State**

State name.

## Return value:

#boolean:

If true, FSM is in this state.

## AUFTAG:LoadCallBacks(CallBackTable)



[Core.Fsm#FSM](#)

## Parameter:

#table **CallBackTable**

Table of call backs.

## AUFRAG:New()

Creates a new FSM object.

## Defined in:

[Core.Fsm#FSM](#)

## Return value:

#FSM:

## AUFRAG:SetProcess(From, Event, Fsm)

## Defined in:

[Core.Fsm#FSM](#)

## Parameters:

**From**

**Event**



## AUFTAG:SetStartState(State)

Sets the start state of the FSM.

### Defined in:

[Core.Fsm#FSM](#)

### Parameter:

#string **State**

A string defining the start state.

## AUFTAG:\_add\_to\_map(Map, Event)

Add to map.

### Defined in:

[Core.Fsm#FSM](#)

### Parameters:

#table **Map**

Map.

#table **Event**

Event table.



Call handler.

## Defined in:

[Core.Fsm#FSM](#)

## Parameters:

#string **step**

Step "onafter", "onbefore", "onenter", "onleave".

---

#string **trigger**

Trigger.

---

#table **params**

Parameters.

---

#string **EventName**

Event name.

---

## Return value:

Value.

---

## AUFTAG:\_create\_transition(EventName)

Create transition.

## Defined in:



#string **EventName**

Event name.

---

## Return value:

#function:

Function.

---

# AUFRAG:\_delayed\_transition(EventName)

Delayed transition.

## Defined in:

[Core.Fsm#FSM](#)

## Parameter:

#string **EventName**

Event name.

---

## Return value:

#function:

Function.

---

# AUFRAG:\_eventmap(Events, EventStructure)

Event map.



## Parameters:

#table **Events**

Events.

---

#table **EventStructure**

Event structure.

---

## AUFTAG:\_gosub(ParentFrom, ParentEvent)

Go sub.

### Defined in:

[Core.Fsm#FSM](#)

## Parameters:

#string **ParentFrom**

Parent from state.

---

#string **ParentEvent**

Parent event name.

---

## Return value:

#table:

Subs.

---



## Defined in:

[Core.Fsm#FSM](#)

## Parameters:

#string **EventName**

Event name.

...

Arguments.

## AUFRAG:\_isendstate(�Current)

Is end state.

## Defined in:

[Core.Fsm#FSM](#)

## Parameter:

#string **Current**

Current state name.

## Return values:

#table:

FSM parent.



## AUFRAG:\_submap(subs, sub, name)

Sub maps.

### Defined in:

[Core.Fsm#FSM](#)

### Parameters:

#table **subs**

Subs.

#table **sub**

Sub.

#string **name**

Name.

## AUFRAG:can(e)

Check if can do an event.

### Defined in:

[Core.Fsm#FSM](#)

### Parameter:

#string **e**

#boolean:

If true, FSM can do the event.

#string:

To state.

## AUFRAG:cannot(e)

Check if cannot do an event.

### Defined in:

[Core.Fsm#FSM](#)

### Parameter:

#string **e**

Event name.

### Return value:

#boolean:

If true, FSM cannot do the event.

## AUFRAG:is(State, state)

Check if FSM is in state.



## Parameters:

#string **State**

State name.

**state**

## Return value:

#boolean:

If true, FSM is in this state.

## Type AUFTRAG.Capability

Mission capability.

### Field(s)

#string **AUFTRAG.Capability.MissionType**

Type of mission.

#number **AUFTRAG.Capability.Performance**

Number describing the performance level. The higher the better.

### Function(s)



Mission category.

## Field(s)

```
#string AUFTRAG.Category.AIRCRAFT
```

Airplanes and helicopters.

```
#string AUFTRAG.Category.AIRPLANE
```

Airplanes.

```
#string AUFTRAG.Category.ALL
```

```
#string AUFTRAG.Category.GROUND
```

Ground troops.

```
#string AUFTRAG.Category.HELICOPTER
```

Helicopter.

```
#string AUFTRAG.Category.NAVAL
```

Naval grous.

## Function(s)

### Type AUFTRAG.Condition

Generic mission condition.

## Field(s)



### #function **AUFTRAG.Condition.func**

Callback function to check for a condition. Should return a #boolean.

## Function(s)

## Type **AUFTRAG.GroupData**

Group specific data.

Each ops group subscribed to this mission has different data for this.

## Field(s)

### Functional.Warehouse#WAREHOUSE.Assetitem **AUFTRAG.GroupData.asset**

The warehouse asset.

### Ops.OpsGroup#OPSGROUP **AUFTRAG.GroupData.opsgroup**

The OPS group.

### #string **AUFTRAG.GroupData.status**

Group mission status.

### #number **AUFTRAG.GroupData.waypointEgressUID**

Egress Waypoint UID.

### Core.Point#COORDINATE **AUFTRAG.GroupData.waypointcoordinate**

Ingress waypoint coordinate.

### #number **AUFTRAG.GroupData.waypointindex**



Waypoint task.

#### Core.Point#COORDINATE **AUFTRAG.GroupData.wpegresscoordinate**

Egress waypoint coordinate.

## Function(s)

## Type **AUFTRAG.GroupStatus**

Mission status of an assigned group.

## Field(s)

#### #string **AUFTRAG.GroupStatus.CANCELLED**

Mission was cancelled.

#### #string **AUFTRAG.GroupStatus.DONE**

Mission task of the Ops group is done.

#### #string **AUFTRAG.GroupStatus.EXECUTING**

Ops group is executing this mission.

#### #string **AUFTRAG.GroupStatus.PAUSED**

Ops group has paused this mission, e.g. for refuelling.

#### #string **AUFTRAG.GroupStatus.SCHEDULED**

Mission is scheduled in a FLIGHGROUP queue waiting to be started.



## Function(s)

### Type AUFTRAG.SpecialTask

Special task description.

## Field(s)

#string **AUFTRAG.SpecialTask.AIRDEFENSE**

Air defense.

#string **AUFTRAG.SpecialTask.ALERT5**

Alert 5 task.

#string **AUFTRAG.SpecialTask.AMMOSUPPLY**

Ammo Supply.

#string **AUFTRAG.SpecialTask.ARMORATTACK**

#string **AUFTRAG.SpecialTask.ARMOREDGUARD**

On guard with armor.

#string **AUFTRAG.SpecialTask.BARRAGE**

Barrage.

#string **AUFTRAG.SpecialTask.CAPTUREZONE**

Capture OPS zone.

#string **AUFTRAG.SpecialTask.EWR**

Ferry mission.

**#string AUFTRAG.SpecialTask.FORMATION**

AI formation task.

**#string AUFTRAG.SpecialTask.FUELSUPPLY**

Fuel Supply.

**#string AUFTRAG.SpecialTask.GROUNDATTACK**

Ground attack.

**#string AUFTRAG.SpecialTask.HOVER**

Hover.

**#string AUFTRAG.SpecialTask.NOTHING**

Nothing.

**#string AUFTRAG.SpecialTask.ONGUARD**

On guard.

**#string AUFTRAG.SpecialTask.PATROLRACETRACK**

Patrol Racetrack.

**#string AUFTRAG.SpecialTask.PATROLZONE**

Patrol zone task.

**#string AUFTRAG.SpecialTask.REARMING**

Rearming.

```
#string AUFTRAG.SpecialTask.RECOVERYTANKER
```

Recovery tanker.

```
#string AUFTRAG.SpecialTask.RELOCATECOHORT
```

Relocate cohort.

## Function(s)

# Type AUFTRAG.Status

Mission status.

## Field(s)

```
#string AUFTRAG.Status.CANCELLED
```

Mission was cancelled.

```
#string AUFTRAG.Status.DONE
```

Mission is over.

```
#string AUFTRAG.Status.EXECUTING
```

Mission is being executed.

```
#string AUFTRAG.Status.FAILED
```

Mission failed.

```
#string AUFTRAG.Status.PLANNED
```



Mission is queued at a LEGION.

#### #string **AUFTRAG.Status.REQUESTED**

Mission assets were requested from the warehouse.

#### #string **AUFTRAG.Status.SCHEDULED**

Mission is scheduled in an OPSGROUP queue waiting to be started.

#### #string **AUFTRAG.Status.STARTED**

Mission has started but is not executed yet.

#### #string **AUFTRAG.Status.SUCCESS**

Mission was a success.

## Function(s)

## Type **AUFTRAG.Success**

Mission success.

## Field(s)

#### #string **AUFTRAG.Success.DAMAGED**

Target was damaged.

#### #string **AUFTRAG.Success.DESTROYED**

Target was destroyed.



```
#string AUFTRAG.Success.SURVIVED
```

Group did survive.

## Function(s)

# Type **AUFTRAG.TargetData**

Target data.

## Field(s)

```
#number AUFTRAG.TargetData.Lifepoints
```

Total life points.

```
#number AUFTRAG.TargetData.Lifepoints0
```

Initial life points.

```
#string AUFTRAG.TargetData.Name
```

Target name.

```
#number AUFTRAG.TargetData.Ninitial
```

Number of initial targets.

```
Wrapper.Positionable#POSITIONABLE AUFTRAG.TargetData.Target
```

Target Object.

```
#string AUFTRAG.TargetData.Type
```



# Type AUFTRAG.TargetType

Target type.

## Field(s)

```
#string AUFTRAG.TargetType.AIRBASE
```

Target is an AIRBASE.

```
#string AUFTRAG.TargetType.COORDINATE
```

Target is a COORDINATE.

```
#string AUFTRAG.TargetType.GROUP
```

Target is a GROUP object.

```
#string AUFTRAG.TargetType.SETGROUP
```

Target is a SET of GROUPs.

```
#string AUFTRAG.TargetType.SETUNIT
```

Target is a SET of UNITs.

```
#string AUFTRAG.TargetType.STATIC
```

Target is a STATIC object.

```
#string AUFTRAG.TargetType.UNIT
```

Target is a UNIT object.



## Type AUFTRAG.Type

Mission types.

### Field(s)

#string **AUFTRAG.Type.AIRDEFENSE**

Air defense.

#string **AUFTRAG.Type.ALERT5**

Alert 5.

#string **AUFTRAG.Type.AMMOSUPPLY**

Ammo supply.

#string **AUFTRAG.Type.ANTISHIP**

Anti-ship mission.

#string **AUFTRAG.Type.ARMORATTACK**

Armor attack.

#string **AUFTRAG.Type.ARMOREDGUARD**

On guard - with armored groups.

#string **AUFTRAG.Type.ARTY**

Fire at point.

#string **AUFTRAG.Type.AWACS**



Battlefield Air Interdiction.

#string **AUFTRAG.Type.BARRAGE**

Barrage.

#string **AUFTRAG.Type.BOMBCARPET**

Carpet bombing.

#string **AUFTRAG.Type.BOMBING**

Bombing mission.

#string **AUFTRAG.Type.BOMBRUNWAY**

Bomb runway of an airbase.

#string **AUFTRAG.Type.CAP**

Combat Air Patrol.

#string **AUFTRAG.Type.CAPTUREZONE**

Capture zone mission.

#string **AUFTRAG.Type.CARGOTRANSPORT**

Cargo transport.

#string **AUFTRAG.Type.CAS**

Close Air Support.

#string **AUFTRAG.Type.CASENHANCED**

Enhanced CAS.

#string **AUFTRAG.Type.EWR**

Early Warning Radar.

#string **AUFTRAG.Type.FAC**

Forward AirController mission.

#string **AUFTRAG.Type.FACA**

Forward AirController airborne mission.

#string **AUFTRAG.Type.FERRY**

Ferry mission.

#string **AUFTRAG.Type.FUELSUPPLY**

Fuel supply.

#string **AUFTRAG.Type.GCICAP**

Similar to CAP but no auto engage targets.

#string **AUFTRAG.Type.GROUNDATTACK**

Ground attack.

#string **AUFTRAG.Type.GROUNDESCORT**

Ground escort mission.

#string **AUFTRAG.Type.HOVER**

Hover.

#string **AUFTRAG.Type.INTERCEPT**



Land at coordinate.

#string **AUFTRAG.Type.NOTHING**

Nothing.

#string **AUFTRAG.Type.ONGUARD**

On guard.

#string **AUFTRAG.Type.OPSTRANSPORT**

Ops transport.

#string **AUFTRAG.Type.ORBIT**

Orbit mission.

#string **AUFTRAG.Type.PATROLRACETRACK**

Patrol Racetrack.

#string **AUFTRAG.Type.PATROLZONE**

Patrol a zone.

#string **AUFTRAG.Type.REARMING**

Rearming mission.

#string **AUFTRAG.Type.RECON**

Recon mission.

#string **AUFTRAG.Type.RECOVERYTANKER**

Recovery tanker.



#string **AUFTRAG.Type.RESCUEHELO**

Rescue helo.

#string **AUFTRAG.Type.SEAD**

Suppression/destruction of enemy air defences.

#string **AUFTRAG.Type.STRAFING**

Strafing run.

#string **AUFTRAG.Type.STRIKE**

Strike mission.

#string **AUFTRAG.Type.TANKER**

Tanker mission.

#string **AUFTRAG.Type.TROOPTRANSPORT**

Troop transport mission.

## Function(s)