

HPG Mission System Guide



Dies basiert auf einer Kopie der HYPE Online Dokumentation.

Für dieses Dokument habe ich verwendet:

- [Mission System](#)
- H145 H:Var (aus H145 User Guide V 2.1.5)
- [H145 Mission System Documentation](#)

Wenn Sie etwas falsch finden, können Sie mich oder Dave (@davux) informieren.
Für Hilfe bei der Arbeit mit Missionen, fragen Sie bitte die Community (creators_chat auf HYPE Discord zum Beispiel) und nicht mich. Ich bin kein Ersteller, ich habe nur an diesem Dokument gearbeitet :).

Viel Erfolg und „have fun“
D-VRGL

Inhaltsverzeichnis

| | |
|---|----|
| HPG Mission System Guide..... | 1 |
| Inhaltsverzeichnis..... | 2 |
| Einführung und Überblick..... | 9 |
| Erstellen von Missionspaketen..... | 10 |
| Mission Index..... | 10 |
| Grundlagen der Missionsentwicklung..... | 10 |
| Mission Format..... | 10 |
| Arbeitsablauf der Entwicklung..... | 10 |
| Abschnitte der Missions-Metadaten..... | 11 |
| Abschnitte über Missionsdaten..... | 11 |
| Überblick über die Ausführung der Mission..... | 12 |
| COMMAND..... | 12 |
| COMMANDLIST..... | 12 |
| QUERY..... | 13 |
| DATAQUERY..... | 13 |
| Arbeiten mit SimVars und L:Vars..... | 14 |
| Luftfahrzeug-Simulationsvariablen (A:Vars)..... | 14 |
| Lokale Variablen (L:Vars)..... | 14 |
| Senden und Empfangen von Events..... | 15 |
| Events Senden..... | 15 |
| Wichtige empfangene Ereignisse..... | 16 |
| Empfang von Ereignissen..... | 16 |
| Arbeiten mit Daten..... | 16 |
| Interpretation von Zeichenketten..... | 17 |
| Optionen zur Datenspeicherung..... | 18 |
| Daten Tabellen..... | 18 |
| Tabllen API..... | 18 |
| Interaktion mit dem Anwender..... | 19 |
| Hauptanzeige Widgets..... | 19 |
| Widgets für Briefing und Versand..... | 20 |
| Karten-Widgets..... | 20 |
| Widget-Testprogramm..... | 21 |
| Arbeiten mit AI-Objekten..... | 23 |
| Objekte erstellen und löschen..... | 23 |
| Objekt-Eigenschaften..... | 23 |
| Objekt bewegen..... | 24 |
| Objekt Wegpunktnavigation..... | 24 |
| Fliegende Objekte..... | 24 |
| Erstellen von Objektpaketen von Drittanbietern..... | 24 |
| Geräusche und Text zu Sprache..... | 24 |
| Built-in sounds..... | 24 |
| Voice Server..... | 24 |
| Einrichtung eines kompatiblen Sprachservers (Fortgeschrittene)..... | 25 |
| Voice Server Test Programm..... | 26 |
| OpenStreetMap Data..... | 27 |

| | |
|--|----|
| OSM Entwickler Workflow..... | 27 |
| Beispiele für OSM-Abfragen..... | 28 |
| OpenStreetMap APIs..... | 29 |
| Beispiel für nahe gelegene Krankenhaus-Hubschrauberlandeplätze anzeigen..... | 30 |
| Beispiel für nahegelegene Umspannwerke anzeigen..... | 30 |
| Objekte auf Bahnübergängen..... | 31 |
| Testprogramm für das Straßennetz..... | 32 |
| Testprogramm für Wasserpolygone..... | 33 |
| Gebäude Test..... | 34 |
| Bedienung des Hebezeugs..... | 36 |
| Hoist test program..... | 37 |
| Multiplayer Missionen..... | 38 |
| Befehle beim Beenden des Servers..... | 38 |
| Gemeinsam genutzte Daten..... | 38 |
| MultiplayerClient..... | 39 |
| Multiplayer-Beispiel für einfache Punktevergabe..... | 40 |
| Multiplayer: Web Client..... | 45 |
| list (WebConfig)..... | 45 |
| map_point (WebConfig)..... | 45 |
| map_line (WebConfig)..... | 46 |
| event (WebConfig)..... | 46 |
| Dialog widgets..... | 46 |
| WEB COMMANDS..... | 46 |
| WEB QUERY..... | 47 |
| Unterstützung für mehrere Sprachen..... | 48 |
| Translation Test Programm..... | 49 |
| Server (Remote) Missionen..... | 49 |
| Vom Luftfahrzeug an den Server gesendete Befehle..... | 50 |
| Vom Server an das Luftfahrzeug gesendete Befehle..... | 50 |
| API Reference - COMMAND..... | 51 |
| #comment..... | 51 |
| sleep..... | 51 |
| wait_for..... | 51 |
| if..... | 52 |
| while..... | 52 |
| for_each..... | 52 |
| try..... | 53 |
| switch..... | 53 |
| trigger..... | 54 |
| call_macro..... | 54 |
| return..... | 54 |
| break..... | 55 |
| continue..... | 55 |
| private_macros..... | 55 |
| create_thread..... | 55 |
| create_event_handler..... | 56 |
| throw_error..... | 56 |
| modify_array..... | 56 |
| reload_mission..... | 56 |

| | |
|------------------------------|----|
| load_mission..... | 57 |
| create_object..... | 57 |
| destroy_object..... | 58 |
| track_object..... | 58 |
| drive_object..... | 58 |
| move_object..... | 59 |
| point_object..... | 59 |
| set_drive_data..... | 59 |
| set_df..... | 60 |
| set_carls_radio..... | 60 |
| set_tfm_radio..... | 61 |
| set_rescuetrack..... | 61 |
| open_door..... | 62 |
| close_door..... | 62 |
| create_fire..... | 62 |
| launch_missile..... | 63 |
| designate_target..... | 63 |
| set_route..... | 63 |
| set_map..... | 63 |
| wait_modal..... | 64 |
| set_modal..... | 64 |
| set_message..... | 64 |
| set_progressbar..... | 65 |
| set_dispatch..... | 65 |
| set_briefing_dialog..... | 65 |
| set_dispatch_dialog..... | 65 |
| scroll_to_briefing_item..... | 66 |
| scroll_to_dispatch_item..... | 66 |
| set_objective_title..... | 66 |
| set_hover_display..... | 66 |
| create_user_action..... | 66 |
| destroy_user_action..... | 67 |
| trigger_user_action..... | 67 |
| set_user_poi..... | 67 |
| create_route..... | 68 |
| draw_route..... | 68 |
| copy_stringtoken..... | 68 |
| open_url..... | 68 |
| copy_location..... | 69 |
| open_location..... | 69 |
| create_location..... | 69 |
| query_data..... | 70 |
| query_country..... | 71 |
| osm_query_data..... | 71 |
| osm_get_parent_ways..... | 72 |
| osm_get_connected_nodes..... | 72 |
| osm_get_nodes..... | 72 |
| osm_get_all_ways..... | 73 |
| osm_get_all_nodes..... | 73 |

| | |
|---------------------------------------|----|
| osm_get_closest_nodes..... | 73 |
| osm_is_point_within_way..... | 73 |
| osm_get_area_of_area..... | 73 |
| open_table..... | 74 |
| save_table..... | 74 |
| clear_table..... | 74 |
| play_audio..... | 74 |
| play_guidance_message..... | 75 |
| connect_voice_server..... | 75 |
| speak..... | 76 |
| Debugger & Fernsteuerungsbefehle..... | 76 |
| cancel_debugger..... | 76 |
| remote_notify..... | 76 |
| teleport_to..... | 76 |
| fetch..... | 77 |
| set_shared_data..... | 77 |
| ebug_write..... | 77 |
| hoist_control..... | 77 |
| API Reference - QUERY..... | 78 |
| text..... | 78 |
| var..... | 78 |
| object/var..... | 78 |
| location/var..... | 80 |
| bearing..... | 80 |
| has_location..... | 80 |
| resolve_location..... | 80 |
| has_object..... | 81 |
| has_user_action..... | 81 |
| has_mission..... | 81 |
| has_macro..... | 81 |
| no_resolve..... | 81 |
| resolve_icon..... | 82 |
| static..... | 82 |
| has_static..... | 82 |
| has_global..... | 82 |
| global..... | 82 |
| has_route..... | 83 |
| route..... | 83 |
| create_array..... | 83 |
| create_struct..... | 83 |
| struct..... | 83 |
| js:get..... | 84 |
| js:create_async_function..... | 84 |
| js:function..... | 84 |
| js:new..... | 84 |
| json:stringify..... | 85 |
| json:parse..... | 85 |
| json:copy..... | 85 |
| object:keys..... | 85 |

| | |
|---|-----|
| string:split..... | .85 |
| string:join..... | .86 |
| create_number..... | .86 |
| has_local..... | .86 |
| local..... | .86 |
| gamevar..... | .86 |
| table..... | .87 |
| param..... | .87 |
| has_param..... | .87 |
| rand..... | .87 |
| add..... | .87 |
| add360..... | .88 |
| compare360..... | .88 |
| subtract..... | .88 |
| multiply..... | .88 |
| divide..... | .88 |
| right_shift..... | .88 |
| left_shift..... | .89 |
| xor..... | .89 |
| remainder..... | .89 |
| exponent..... | .89 |
| round..... | .89 |
| toFixed..... | .90 |
| floor..... | .90 |
| ceil..... | .90 |
| abs..... | .90 |
| Math. ... Funktionen..... | .90 |
| clamp..... | .91 |
| scale..... | .91 |
| require..... | .91 |
| and..... | .91 |
| or..... | .92 |
| not..... | .92 |
| typeof..... | .92 |
| isNaN..... | .93 |
| parseInt..... | .93 |
| parseFloat..... | .93 |
| if..... | .93 |
| switch..... | .93 |
| convert..... | .94 |
| fn.HOIST_SEND_TO_GROUND..... | .94 |
| fn.HOIST_REEL_UP_AND_STOW..... | .94 |
| fn.HOIST_REEL_UP..... | .95 |
| fn.hoist_get_reel_distance:ft..... | .95 |
| fn.hoist_get_distance_from_ground:ft..... | .95 |
| fn.score_bambi_dump..... | .95 |
| fn.all_fires_extinguished..... | .96 |
| fn.has_remote_notify..... | .96 |
| fn.is_voice_server_connected..... | .96 |

| | |
|--|-----|
| fn.create_guid..... | 96 |
| fn.create_date..... | 96 |
| fn.get_time_string..... | 96 |
| fn.get_mission_objects..... | 97 |
| fn.get_aircraft_moniker..... | 97 |
| fn.is_any_sling_object_coupled..... | 97 |
| fn.get_sling_object_type..... | 97 |
| fn.get_mission_icons..... | 97 |
| fn.create_multiplayer_connection..... | 97 |
| API Referenz - LOCATION..... | 98 |
| LOCATIONREF..... | 98 |
| bearing..... | 98 |
| bearing2..... | 99 |
| location_alter..... | 99 |
| closest..... | 99 |
| Besondere Standorte..... | 99 |
| Beispiele..... | 100 |
| Konvertierte Funktion aus JS Beispiel..... | 100 |
| Beispiel für die Erkennung von Szenerien..... | 100 |
| Zufälliges Element aus der statischen Liste holen..... | 101 |
| CARLS Radio Test Program..... | 102 |
| Testprogramm für Remote Dispatcher..... | 103 |
| RescueTrack Test Program..... | 112 |
| SDK H:Events..... | 116 |
| Home Cockpit SDK..... | 116 |
| Overhead Panel..... | 116 |
| Engine Control Panel (ECP)..... | 118 |
| Autopilot Control Panel (APCP)..... | 118 |
| Cyclic Control..... | 119 |
| Collective Control..... | 119 |
| H145M Weapons..... | 120 |
| Cabin..... | 120 |
| Misc..... | 121 |
| Hoist..... | 121 |
| Center Console WXRCF..... | 122 |
| Search Light..... | 122 |
| Landing Light..... | 122 |
| Center Console HISLCP..... | 122 |
| Tablet..... | 123 |
| Hype Radio App..... | 124 |
| Equipment Setup..... | 124 |
| MFDs..... | 125 |
| IESI..... | 127 |
| Center Console Other..... | 127 |
| Sensor Pod..... | 128 |
| System Failures..... | 128 |
| H145 Mission System Documentation..... | 129 |
| Grundlegende Details zu der Mission..... | 129 |
| Missionen von einem Server laden..... | 129 |

| | |
|---|-----|
| Erstellen von Missionspaketen..... | 130 |
| Missions Abschnitte..... | 130 |
| OBJECT..... | 131 |
| Spezielle Objektvariablen..... | 131 |
| THREAD..... | 132 |
| OBJECTIVE..... | 132 |
| Commands..... | 133 |
| Dynamic Object Library..... | 133 |
| H145 Crew..... | 133 |
| H145 verletzter Mensch..... | 134 |
| H145 Winkender Zivilist..... | 134 |
| H145 Leuchtsignal..... | 135 |
| Erstellen von benutzerdefinierten dynamischen Objekten..... | 135 |
| Mission Server..... | 136 |
| Vom H145 an den Missionsserver gesendete Befehle..... | 136 |
| Vom Server an den H145 gesendete Befehle..... | 136 |

Einführung und Überblick



Das HPG-Missionssystem ist eine in das Flugzeug integrierte Plattform, die eine hochgradige Orchestrierung von Missionsszenarien ermöglicht.

Missionen sind kleine Textdateien, die einem Computerprogramm ähnlich sind. Diese Programme haben Zugriff auf den Simulator, das Benutzerflugzeug und das Netzwerk und ermöglichen so fesselnde und realistische Szenarien, die die Funktionen der Flugzeugvarianten nutzen.

| Bestandteil | |
|----------------------------|--|
| Flight Simulator SDK | Zugriff auf Variablen und Ereignisse innerhalb der Simulation. |
| HPG Aircraft SDK | Zugriff auf Variablen und Ereignisse innerhalb des HPG-Luftfahrzeugs. |
| AI Object Management | Erstellen und verwalten Sie KI-Objekte am Boden und in der Luft. |
| OpenStreetMap Data Queries | Leistungsstarke APIs zur Abfrage von Informationen über die Umwelt. |
| Pilot Interfaces | Schnittstelle zur Missions-App oder zu Touchpoints im Cockpit wie den Funkgeräten oder Rescue Track. |
| Sound & Text-To-Speech | Abspielen von Sounddateien und dynamische Text-zu-Sprache-Ausgabe |
| Network Communication | Führen Sie Missionen über das Netzwerk aus und kommunizieren Sie, um Multiplayer-Funktionen zu ermöglichen |
| Debugger & Editor | Testen Sie Missionen mit dem integrierten Debugger für eine schnelle Entwicklung |
| Templates | Erstellen von Missionsvorlagen, die vom Benutzer mit einem grafischen Editor bearbeitet werden |

Erstellen von Missionspaketen

Die Missionen können zu jedem anderen Community-Paket hinzugefügt werden.

1. Erstellen Sie einen `hpgmission`-Ordner innerhalb Ihres Pakets, und legen Sie eine Ordnerhierarchie mit Ihren json-Missionsdateien darunter.

Alle Inhalte (Ordner und json-Dateien) unterhalb von `hpgmission` über alle Gemeinschaftspakete hinweg werden im `Missionsindex` zusammengeführt.

Mission Index

Es gibt zwei Möglichkeiten, den Missionsindex des Flugzeugs zu aktualisieren. Sie müssen dies tun, bevor das Flugzeug neue Missionen oder aktualisierte Missionen sieht.

1. `Tools\Update Mission Index.cmd`.
2. Hype Operations Center -> `Refresh Index`.
3. Wenn Ihr Flugzeug im Einsatz ist, besteht der letzte Schritt darin, in der Missions-App zu aktualisieren, um den neuen Index zu übernehmen.

Hinweis: Wenn Sie den direkt angeschlossenen `Scenario Developer` verwenden, umgehen Sie den Index und laden neue Missionen direkt über den Editor.

Grundlagen der Missionsentwicklung

Mission Format

Missionen sind JSON-Dateien. Sie sollten einen JSON-Validator wie [jsonlint](#) verwenden, um sicherzustellen, dass das Dateiformat gültig ist. Das Hype Operations Center enthält auch das Tool `Scenario Developer`, das die Formatierung überprüft, wenn Sie Änderungen vornehmen.

Arbeitsablauf der Entwicklung

Missionsentwickler sollten den `Missionseditor -> Szenarioentwickler` im Hype Operations Center verwenden. Klicken Sie auf `Verbinden mit dem Missionseditor (Mission Editor)` auf dem PC und dann auf `Speichern`, um das Skript in das Flugzeug zu laden. Sobald das Luftfahrzeug verbunden ist, wird die Mission durch `Speichern (save)` automatisch neu geladen.

Die Mission kann ganz oder teilweise in HOC getestet und anschließend in einer einfachen .json-Datei zur Weitergabe an die Endnutzer gespeichert werden. Bei Verwendung des Editors wird der Missionsindex umgangen.

Abschnitte der Missions-Metadaten

| Section | Description |
|-------------|--|
| title | Text für den Benutzer, um die Mission zu identifizieren. |
| id | id wird verwendet, um mit <code>load_mission</code> API zur Mission zu wechseln. Muss eindeutig sein. |
| start_info | Legt die Startpositionen auf der Karte fest. |
| briefing | Konfigurieren Sie die Informationen, die der Benutzer beim Start der Mission sehen soll. |
| aircraft | Optional. Gibt, falls vorhanden, eine Reihe von unterstützten Luftfahrzeugen an. <code>["H145"]</code> |
| applicable | Optional. Gibt, falls vorhanden, eine Reihe von unterstützten Varianten an. <code>["CIVILCARGO", "MILITARYCARGO"]</code> |
| api_version | Nicht überprüft mit v1 Missionen. Alle Missionen sind API Version 0.1. |

Abschnitte über Missionsdaten

Jeder der folgenden Abschnitte entspricht einem Datensatz für verschiedene Arten von Daten. In der Regel können Sie statische Informationen im Voraus definieren oder APIs aufrufen, um Daten während der Mission zu erstellen/zu bearbeiten/zu entfernen.

| Section | Description |
|--------------|--|
| locations | Standorte (lat/lon) |
| events | Events (Event handlers) |
| objects | Objekte (live objects) |
| routes | Routen (lists of locations) |
| threads | Ausführungsthreads |
| stringTokens | Ersetzen einer Zeichenkette durch eine andere |
| userActions | Befehle, mit denen der Benutzer interagieren kann |
| icons | data-uri's, die 44x44 PNG-Bilder zur Verwendung auf der Karte darstellen |
| macros | Funktionen, die die Mission verwenden kann (wiederverwendbarer Code) |
| data | Statische Angaben |

Überblick über die Ausführung der Mission

Eine Mission ist im Wesentlichen ein Computerprogramm. Missionen bestehen aus einer Reihe von Befehlen, die mit Daten innerhalb des Simulators und im Netz arbeiten können.

Ein sehr einfacher Auftrag sieht folgendermaßen aus:

```
{
  "title": "My simple mission",
  "objectives": [
    {
      "title": "Done",
      "commands": [
        {"sleep": "forever"}
      ]
    }
  ]
}
```

title: Der Titel der Mission wird vom Benutzer verwendet, um sie in Listen zu identifizieren und sie aus der Missionsbibliothek auszuwählen.

objectives: Ziele enthalten einfach einen anderen **Titel** (das Ziel, das unten in der Missions-App angezeigt wird) und Befehle (**comands**) (eine **COMMANDLIST**), die beim Start der Mission automatisch ausgeführt werden.

Diese Mission enthält nur einen Befehl, {"sleep": "forever"}, der das System anweist, mit dem Warten zu beginnen und niemals fortzufahren. Dieser **BEFEHL** verhindert, dass die Mission endet.

COMMAND

COMMAND ist der grundlegende Befehl im Missionssystem, der immer in Form einer **COMMANDLIST** ausgeführt wird. **QUERY** wird sehr häufig verwendet und ist ein Bestandteil eines Befehls, aber kein eigenständiger Befehl.

Jeder der möglichen Befehle ist im Kapitel [COMMANDS](#) aufgeführt. Befehle, die eine **QUERY** benötigen, können einen beliebigen Ausdruck aus dem Abschnitt [QUERY](#) verwenden.

COMMANDLIST

Eine **COMMANDLIST** ist eine Liste von Befehlen, die nacheinander ausgeführt werden, wobei das Ende jedes Befehls abgewartet wird, bevor es weitergeht.

```
[
  COMMAND1,
  COMMAND2,
  COMMAND3
]

[
  {"set_message":{"text": "hello world"}},
  {"sleep": 1},
  {"#comment": "my hello world program"}
]
```

QUERY

Eine **QUERY** kann aus anderen **QUERY** zusammengesetzt werden, was zu einem Ausdruck führt, der z.B. einen Wert abrufen und einen anderen Wert dazu addiert.

Jeder der unten aufgeführten Befehle eignet sich als **QUERY**, ebenso wie Zahlen und Strings.

Beispiel:

```
1
11.5
{"var":["L:TEST", "number"]}
"hello"
{"text": "hello {0}", "params": [ QUERY, ... ]}
```

DATAQUERY

Eine Datenabfrage ist eine OSM Overpass API-Abfrage. Überprüfen Sie Ihre Abfragen auf [Overpass Turbo](#). Optional können Abfragen mit **Logik/Gruppen** (logic/groups) nachbearbeitet werden.

Beispiel:

```
"[out:json]; node({{bbox}})[man_made=silo]; out center;"

{
  query:
  "[out:json];(area({{bbox}})[amenity=hospital];area({{bbox}})
[eroway=helipad]); out center;",
  "groups": [
    {amenity: "hospital"},
    {eroway: "helipad"}
  ],
  logic: {"intersection": 0.2}
}
```

Arbeiten mit SimVars und L:Vars

Die Informationen aus dem Simulator und dem HPG-Flugzeug sind jederzeit abrufbar.

Luftfahrzeug-Simulationsvariablen (A:Vars)

A:Vars sind Daten aus der MSFS-Simulation, einschließlich der Umgebung und der Standardsysteme.

Ein paar beliebte Beispiele sind:

| Variable | Beschrei |
|-------------------------|--------------------------------|
| PLANE ALTITUDE, feet | Höhe über dem Erdboden. |
| GPS GROUND SPEED, knots | Geschwindigkeit über Grund |
| RADIO HEIGHT, feet | Funkhöhe (Höhe über dem Boden) |

Die [vollständige Liste der Simulationsvariablen](#) in MSFS. Jeder Variable auf dieser Seite kann ein A: vorangestellt werden, was jedoch nicht notwendig ist, da dies das Standard-Variablenpräfix ist.

Examples:

```
{"set_message":{"text":"my variable is: {0}", "params": [{"var":["PLANE ALTITUDE", "feet"]}]}}
```

Note that **A:Vars** are **usually** read-only. Some **A:Vars** however can be written directly, like the transponder:

Beispiel:

```
{"set":{"var":["TRANSPONDER STATE:1", "enum"], "value": 1}}
```

Lokale Variablen (L:Vars)

L:Vars sind wesentlich flexibler als A:Vars.

- Durch einfaches Schreiben in eine L:Var wird diese erstellt, wenn sie vorher nicht existierte.
- Es steht den Entwicklern frei, L:Vars für alles zu verwenden, was sie benötigen.

Sie haben Zugriff auf die HPG Aircraft SDK **L:Vars** und können Ihre eigenen L:Vars für Ihre eigene Anwendung verwenden. Beachten Sie, dass die meisten HPG Aircraft SDK **L:Vars** schreibgeschützt sind.

Beispiel:

```
{"set_message":{"text":"my variable is: {0}", "params": [
  {"var":["L:H145_SDK_VARIANT_ID", "number"]}
]}}
```

Beispiel:

```
{"set":{"var":["L:MY_MISSION_VAR", "number"]}, "value": 99}
```

Senden und Empfangen von Events

Sie werden viele Ereignisse senden, die verschiedene Aktionen innerhalb des Simulators und des HPG-Flugzeugs auslösen. Sie werden nicht so viele Ereignisse empfangen, sondern nur dann, wenn das System Sie über ein Ereignis informieren muss.

Events Senden

Es gibt zwei Haupttypen von Ereignissen, **K**: Ereignisse und **H**: Ereignisse. **K**: (für Keyboard) Events sind im Wesentlichen die gleichen Control Bindings, die Sie in den MSFS-Einstellungen binden können. **H**: (für HTML) Ereignisse sind Ereignisse, die vom Entwickler definiert werden, d. h. die Liste ist das HPG Aircraft SDK.

| Event Prefix | List | Provider |
|---------------|---|-----------|
| K: (Keyboard) | Simulation Ereignis-IDs | Microsoft |
| H: (Html) | HPG H145 SDK Events | HPG |

Beispiel:

```
{"trigger": "K:TOGGLE_NAV_LIGHTS"}
{"trigger": "H:H145_SDK_OH_PITOT_1_TOGGLE"}
```

Wenn Sie den Wert eines K: Ereignisses setzen möchten, verwenden Sie stattdessen **set**.

Wichtige empfangene Ereignisse

Dies ist eine Liste häufiger Ereignisse (ohne Anspruch auf Vollständigkeit), auf die Sie reagieren können:

| Event Name | Beschreibung |
|---|--|
| <code>ON_MISSION_ABORTING</code> | Wird kurz vor dem Entladen der Mission aufgerufen. Sie sollten hier nur kritische Aktionen durchführen. |
| <code>H145_SDK_CARGO_COUPLE_FAILED</code> | Wird aufgerufen, wenn die Taste zum Verbinden der Ladung gedrückt wurde, aber kein Objekt an- oder abgekoppelt werden konnte. |
| <code>H145_SDK_CARGO_COUPLE_ACTIVATED</code> | Wird aufgerufen, wenn die Taste zum Verbinden der Ladung gedrückt wurde und anschließend ein Objekt erfolgreich verbunden wurde. |
| <code>H145_SDK_CARGO_DECOUPLE_ACTIVATED</code> | Wird aufgerufen, wenn die Taste zur Verbindung der Ladung gedrückt wurde und ein Objekt anschließend erfolgreich abgekoppelt wurde. |
| <code>H145_SDK_HELITOUCH_IGNITE_ACTIVATED</code> | Wird aufgerufen, wenn der Heli- Brenner sofort ein neues Feuer erzeugen will. (Dies wird wiederholt, aber mit der richtigen Rate, damit Sie Feuer erzeugen können) |
| <code>H145_SDK_BAMBI_BUCKET_FILL_ACTIVATED</code> | Wird aufgerufen, wenn sich der Bambi-Behälter zu füllen beginnt. |
| <code>H145_SDK_BAMBI_BUCKET_DUMP_ACTIVATED</code> | Wird aufgerufen, wenn das Ventil des Bambi-Behälter geöffnet wird. |
| <code>H145_SDK_BAMBI_BUCKET_VALVE_CLOSED</code> | Wird aufgerufen, wenn das Ventil des Bambi-Behälters geschlossen wird. |

Empfang von Ereignissen

Sie können über `H`-Ereignisse informiert werden, indem Sie einen Event-Handler erstellen. Sie können den Handler auch im Voraus in der Ereignistabelle (`events` table) definieren. Beachten Sie, dass Sie nicht mehr als einen Event-Handler für ein bestimmtes Ereignis haben dürfen, und dass der letzte Handler gewinnen wird.

Beispiel:

```
{ "create_event_handler": "H145_SDK_BAMBI_BUCKET_DUMP_ACTIVATED", "commands": [
  { "set_message": { "text": "bambi dumped!" } }
]}
```

Hinweis: Das Arbeiten im Cockpit (Umlegen von Schaltern) erzeugt für die meisten Steuerelemente NICHT das "erwartete" SDK-Ereignis.

Arbeiten mit Daten

Im Rahmen Ihrer Mission müssen Sie auf Daten zugreifen und diese speichern.

Die einfachste Form der Speicherung ist der **Param**. Params sind eine Sammlung von Schlüssel/Wert-Paaren, die mit Ihrem Ausführungskontext verbunden sind. Wenn Sie eine Mission zum ersten Mal starten, laufen Sie auf dem Haupt-Thread des Ziels und **params** ist leer. Wenn Sie Daten erhalten und darauf verweisen müssen, können Sie diese Daten einem Parameter wie **my_param** zuordnen. Der Schlüssel **my_param** wird verwendet, um auf einige Daten zu verweisen, auf die dann jederzeit zugegriffen oder geschrieben werden kann. Wenn Sie ein Makro mit **call_macro** aufrufen, werden die params explizit übergeben, was bedeutet, dass Sie standardmäßig keine Kopie der vorhandenen params erhalten, es sei denn, Sie übergeben jeden einzelnen mit seinem Namen. Der Sinn von params ist, dass Sie einen lokalen exklusiven Speicherplatz haben, um Ihre Daten zu verwalten.

Die zweitunterste Form sind **Locals**. Locals sind in der gesamten Mission verfügbar, funktionieren aber ansonsten genau wie Params. **Locals** sind gemeinsam genutzte Daten, die auch im **Briefing** und im **dispatch** verfügbar sind, wo Params nicht verfügbar sind. Sie werden **Locals** sehr oft verwenden und sie sind auch nützlich für die Fehlersuche, so dass Sie zum Beispiel einen sonst "unzugänglichen" Param-Wert verwenden können. Locals befinden sich immer noch innerhalb der Missionsplattform und verursachen keine zusätzlichen Aufwände oder Zeit beim Lesen/Schreiben.

Wenn Sie große Datenmengen zwischen den Einsätzen speichern müssen, verwenden Sie die **Tabellen-API**. Tabellen sind Schlüssel/Wert-Paare, die auf die Festplatte geladen und gespeichert werden, und Sie können so viele Daten in einer Tabelle speichern, wie Sie benötigen.

Wenn Sie statische Informationen haben, die Sie im Voraus mit Ihrer Mission definieren, ruft der Befehl **static** jeden Pfad aus dem Datenabschnitt Ihrer Mission ab. Dies ist ein großartiger Ort, um übergeordnete Konfigurationsoptionen zu definieren, damit jemand Ihre Mission anpassen kann.

Vermeiden Sie die Verwendung globaler Daten, stattdessen kann eine Tabelle für alle benötigten permanenten Daten verwendet werden.

Sie können Daten in **L:Vars** lesen und speichern, die Sie dann auch mit dem MSFS-Verhaltensfenster anzeigen können, allerdings haben L:Vars einen kleinen Leistungsnachteil gegenüber **params** und **locals**.

Interpretation von Zeichenketten

Lokale und statische Variablen sowie Parameter (ohne Präfix) können mit Strings und geschweiften Klammern referenziert werden.

In den folgenden Fällen befindet sich **my_id** in **params**, **locals** bzw. statischen Daten.

Beispiel:

```
"object{my_id}"  
"object{local:my_id}"  
"object{static:my_id}"
```

Optionen zur Datenspeicherung

- `param` - isoliert zu jedem Makro+Kind-Thread und dem Ziel-Thread+Kind-Thread.
- `local` - für die Mission und zwischen Missionen gemeinsam genutzt, wenn `reload_mission` und `load_mission` verwendet wird.
- `global` - eine globale Variable lesen, die persistent ist. Vermeiden Sie es, Daten hier zu speichern, wenn Sie stattdessen eine Tabelle verwenden können!
- `table` - Lesen von Tabellendaten, die aufbewahrt werden (eine Gruppe von Schlüsseln kann unter einem Tabellennamen gespeichert werden)
- `static` - statische Daten aus der Missionsdatentabelle lesen, z. B. Konfigurationseinstellungen
- `location` Orte aus der Tabelle der Einsatzorte lesen
- `var` - `L:Vars`: global für MSFS, sichtbar im Verhaltensfenster und `A:Vars`: MSFS Aircraft SDK-Variablen.

Daten Tabellen

Datentabellen werden zum Speichern und Abrufen von Informationen verwendet. Datentabellen werden optional auf der Festplatte gespeichert, so dass Sie über dauerhafte Informationen verfügen können.

Datentabellen werden durch ihren Namen identifiziert, der dann als Dateiname auf der Festplatte verwendet wird. Jede Tabelle ist ein JSON-Objekt mit Schlüsseln.

Sie können auf Ihre Tabellen auf der Festplatte unter `%LocalAppData%\Packages\Microsoft.FlightSimulator_8wekyb3d8bbwe\LocalState\packages\hpg-airbus-h145-ap\work` zugreifen.

Tabellen API

Beginnen Sie mit `open_table` mit dem `Namen` der Tabelle, die Sie verwenden wollen. In diesem Schritt wird entweder eine Tabelle von der Festplatte geladen oder eine neue, leere Tabelle für Sie erstellt. Wenn Sie alles aus einer Tabelle entfernen möchten, können Sie `clear_table` jederzeit nach dem Öffnen der Tabelle verwenden.

Danach können Sie mit Set-Befehlen einen bestimmten `Schlüssel` in der Tabelle festlegen. Jeder `Schlüssel` in der Tabelle ist eindeutig, und Sie können alle Daten innerhalb dieses Schlüssels speichern. Wenn Sie zweimal in denselben Schlüssel schreiben, werden die vorherigen Daten überschrieben, und der alte Inhalt geht verloren.

Sie können Daten aus einer `Tabelle` lesen, indem Sie eine Tabelle mit dem angegebenen `Schlüssel` verwenden

Wir öffnen eine Tabelle mit dem Namen `test1`, setzen `item1` auf `99` und speichern sie. Danach geben wir `item1` mit `set_message` auf dem Bildschirm aus.

```

{"open_table": "test1"}
{"set": {"table": "test1", "key": "item1"}, "value": 99},
{"save_table": "test1"},
{"set_message": {"text": "The contents of the item1 key are: {0}", "params": [
  {"table": "test1", "key": "item1"}
]}}

```

Der Inhalt des Schlüssels item1 ist: 99, wie erwartet.
Entfernen Sie nun die Zeilen zum Speichern und Ändern der Tabelle:

```

{"open_table": "test1"}
{"set_message": {"text": "The contents of the item1 key are: {0}", "params": [
  {"table": "test1", "key": "item1"}
]}}

```

Und Sie werden sehen, dass das Ergebnis unverändert ist, da die Tabelle von der Festplatte geladen wurde.

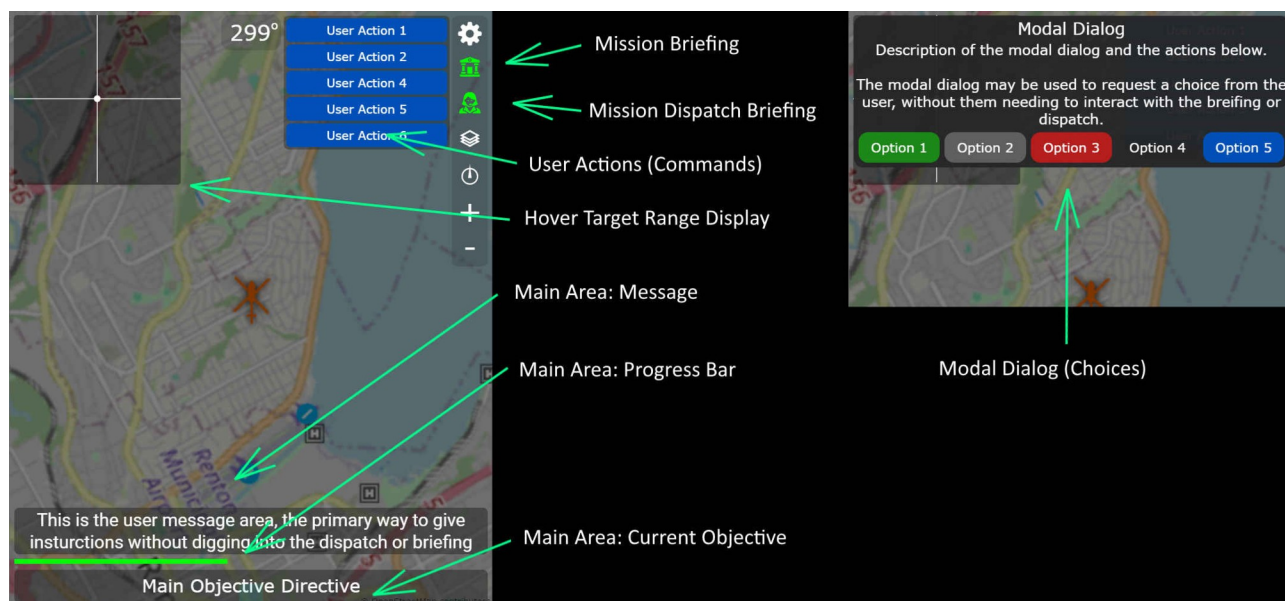
Wir können diese Persistenz nutzen, um Einstellungen, Protokolle der Mission oder was auch immer wir wollen, zu speichern. Versuchen Sie, `save_table` nicht zu schnell aufzurufen, da es die Festplatte belastet.

Interaktion mit dem Anwender

Die Missions-App ist auf dem Tablet verfügbar und dient zum Starten von Missionen sowie zur Interaktion mit der Mission während der gesamten Dauer ihrer Ausführung.

Hauptanzeige Widgets

Die Widgets der Hauptanzeige sind oben auf der Missionskarte verfügbar. Sie können die Elemente jederzeit nach Bedarf konfigurieren.



Jedes Widget wird in dem folgenden Testprogramm behandelt.

Zu Beginn können Sie `set_message` verwenden.

```
{"set_message": "hello world"}
```

Diese Meldung wird am unteren Rand der Missions-App angezeigt.

Widgets für Briefing und Versand

Diese Widgets sind für die `Einsatz-` und `Briefing-Seiten` verfügbar. Das `Briefing` ist statisch und wird mit der Mission definiert, während die `Meldung` dynamisch ist und jederzeit geändert werden kann.

| Element | Beschreibung |
|-------------------------------|--|
| <code>#comment</code> | menschenlesbare Beschreibung, kein Effekt |
| <code>title</code> | Großen Überschriftentext anzeigen |
| <code>text</code> | Text mit verschiedenen Formatierungen anzeigen |
| <code>textbox</code> | Ermöglichen Sie dem Benutzer die Eingabe von Freitext. |
| <code>buttonbar</code> | Erstellen einer Reihe von Schaltflächen |
| <code>buttonbar.button</code> | Klickbare Schaltfläche erstellen (mit <code>select_condition</code>) |
| <code>link</code> | Anklickbaren Link erstellen |
| <code>image</code> | Bild anzeigen |
| <code>describe_icon</code> | Zeichnen Sie ein kleines Bild mit Text rechts daneben |
| <code>iframe</code> | Anzeigen eines IFrames |
| <code>slider</code> | Anzeige eines Schiebereglers, mit dem der Benutzer aus einer Reihe von Werten auswählen kann |
| <code>progressbar</code> | Anzeige eines Wertebereichs |

Jedes Widget hat eine `show_condition` und eine `disabled_condition`:

- `show_condition`: Optional. QUERY, die bestimmt, ob das Element sichtbar sein soll.
- `disabled_condition`: Optional. QUERY, die bestimmt, ob das Element nicht interaktiv und sichtbar deaktiviert sein soll.
- `select_condition`: Optional. QUERY, die bestimmt, ob das Element visuell ausgewählt werden soll. (nur `buttonbar.button`)

Karten-Widgets

Mit Kartenwidgets können Sie Elemente auf der Missionskarte platzieren (sie sind auch auf DMAP/NAVD auf den Flugzeug-MFDs verfügbar).

Sie können Punkte mit Symbolen und/oder Text hinzufügen und Linien zeichnen. Sie können auch einen präzisen Entfernungskreis um einen Punkt ziehen.

Beispiel:

```
{
  "copy_location": {"bearing": 330, "dist": 500, "to": "P1"},
  "copy_location": {"bearing": 30, "dist": 500, "to": "P2"},
  "copy_location": {"bearing": 120, "dist": 500, "to": "P3"},
  "copy_location": {"bearing": 240, "dist": 500, "to": "P4"},
  "set_map": {"add": {"line": {"points": ["P1", "P2", "P3", "P4", "P1"], "stroke": {"color": "#4287f5", "width": 4}}}},
  "set_map": {"add": {"point": {"location": "P1", "text": "waypoint text"}}},
  "set_map": {"add": {"point": {"location": "P4", "icon": "ki_helipad"}}},
}
```

Sie können die Kartenauswahl des Benutzers überwachen, indem Sie den Ort `$MISSION_SELECTED_POI_LOCATION` und `L:MISSION_SELECTED_POI_TYPE` (Null, wenn nicht ausgewählt) überprüfen.

Widget-Testprogramm

Dieses Programm enthält ein Beispiel für jedes der Widgets.

```
{
  "title": "Widget Test Program",
  "briefing": [
    {
      "title": "Briefing Title",
      "text": "paragraph text",
      "text": "paragraph text with params {0} {1}", "params": [99, 100],
      "text": "red text", "color": "red",
      "text": "centered text", "align": "center",
      "text": "Text input: (value: {0})", "params": [{"local": "my_textbox"}],
      "textbox": "my_textbox",
    },
    {
      "text": "Button bar (button clicked count={0})", "params": [{"local": "test_local"}],
      "buttonbar": [
        {
          "title": "add", "commands": [
            {
              "set": {"local": "test_local", "value": {"add": [{"local": "test_local"}, 1]}}
            ]
          },
          "title": "subtract", "commands": [
            {
              "set": {"local": "test_local", "value": {"add": [{"local": "test_local"}, -1]}}
            ]
          },
          {
            "title": "my button (with conditions)",
            "commands": [
              {
                "set": {"local": "test_local", "value": {"add": [{"local": "test_local"}, 1]}}
              ],
              "show_condition": {"require": {"local": "test_local"}, "gt": 0},
              "select_condition": {"require": {"local": "test_local"}, "eq": 2},
              "disabled_condition": {"require": {"local": "test_local"}, "eq": 3}
            ]
          }
        ]
      ],
    },
    {
      "link": "my link", "commands": [
        {
          "set": {"local": "test_local", "value": {"add": [{"local": "test_local"}, 1]}}
        ]
      ],
    },
    {
      "image": "https://www.hypeperformancegroup.com/cdn/shop/files/HPG_Solid_Transparent_180x.png",
      "describe_icon": "fire_station", "description": "icon description here",
      "iframe": "https://davux.com/docs", "height": 400,
      "slider": {"min": 0, "max": 100, "var": ["L:TEST", "number"] },
      "progressbar": {"min": 0, "max": 100, "color": "red", "var": ["L:TEST", "number"] },
      "#comment": "you may add comments as needed"
    ],
  ],
  "objectives": [
    {
      "title": "Done",
      "commands": [
        {
          "set": {"local": "test_local", "value": 0},
          "set": {"var": ["L:TEST", "number"], "value": 45},
          "set_modal": {}
        },
        {
          "create_user_action": {
            "id": "user_action_1",
            "title": "User Action 1",
            "click_commands": [ {"destroy_user_action": "user_action_1"} ]
          },
        },
        {
          "create_user_action": {
            "id": "user_action_2",
            "title": "User Action 2",
            "click_commands": [ {"destroy_user_action": "user_action_2"} ]
          },
        },
        {
          "create_user_action": {
            "id": "user_action_3",
          }
        }
      ]
    }
  ]
}
```

```

        "title": "User Action 3",
        "click_commands": [ {"destroy_user_action": "user_action_3"} ]
    },
    {"create_user_action": {
        "id": "user_action_3",
        "title": "User Action 4",
        "click_commands": [ {"destroy_user_action": "user_action_4"} ]
    },
    {"create_user_action": {
        "id": "user_action_5",
        "title": "User Action 5",
        "click_commands": [ {"destroy_user_action": "user_action_5"} ]
    },
    {"create_user_action": {
        "id": "user_action_6",
        "title": "User Action 6",
        "click_commands": [ {"destroy_user_action": "user_action_6"} ]
    },
    {"set_hover_display": {"target": "$USER", "range": 0.1}},
    {"set_message": "This is the user message area, the primary way to give instructions without digging into the dispatch or briefing"},
    {"set_objective_title": "Main Objective Directive"},
    {"set_progressbar": {"min": 0, "max": 100, "var": ["L:TEST", "number"], "color": "green"}},
    {"set_dispatch": [
        {"text": "Example dispatch"}
    ]},
    {"set_modal": {
        "title": "Modal Dialog",
        "text": "Description of the modal dialog and the actions below.<br /><br /> The modal dialog may be used to request a choice from the user, without them needing to interact with the briefing or dispatch.",
        "options": [
            {"text": "Option 1", "style": "primary", "commands": [
                {"comment": "use a sleep 0 here to make sure button with empty list still executes"},
                {"sleep": 0}
            ]},
            {"text": "Option 2", "style": "secondary", "commands": [
                {"comment": "use a sleep 0 here to make sure button with empty list still executes"},
                {"sleep": 0}
            ]},
            {"text": "Option 3", "style": "danger", "commands": [
                {"comment": "use a sleep 0 here to make sure button with empty list still executes"},
                {"sleep": 0}
            ]},
            {"text": "Option 4", "style": "subtle", "commands": [
                {"comment": "use a sleep 0 here to make sure button with empty list still executes"},
                {"sleep": 0}
            ]},
            {"text": "Option 5", "style": "", "commands": [
                {"comment": "use a sleep 0 here to make sure button with empty list still executes"},
                {"sleep": 0}
            ]}
        ]
    },
    {"copy_location": {"bearing": 330, "dist": 500, "to": "P1"},
    {"copy_location": {"bearing": 30, "dist": 500, "to": "P2"},
    {"copy_location": {"bearing": 120, "dist": 500, "to": "P3"},
    {"copy_location": {"bearing": 240, "dist": 500, "to": "P4"},
    {"set_map": {"add": {"line": {"points": ["P1", "P2", "P3", "P4", "P1"], "stroke": {"color": "#4287f5", "width": 4}}}},
    {"set_map": {"add": {"point": {"location": "P1", "text": "waypoint text"}}},
    {"set_map": {"add": {"point": {"location": "P4", "icon": "fire_station"}}},
    {"sleep": "forever"}
}
],
"icons": {
    "fire_station":
"data:image/png;base64,iVBORw0KGgoAAAANSUHEUgAAACAAAAAAYAAAFp2gQAAAAAXNSR0IArs4c6QAAAAARnQUBAACAjwv8YQAAAAJcEhZcwAADsMAAA7ACdvwGQAAAbCSURBVFH5Vl9bFNVFL+v3X/rTtZ2a90G2dhWBqxsYf0gIVNMi0xRhpM0D9IMHwae4mJaICyGkiQoJEMeMhUQ01MZE/JSY0BYyowwMYIpuYBEG7ZRMGw9d1a6/n3L731te+0q5sqPglP86955x77n3n3Xc/
OpI0aMvTVmq323fyDUUCoWLL4p08BJkTU17sejVQC2UEhdLHFQsEqNYACwQ6Q9v6c97CYsf1cSB47i32Ejy8+1HQIieT0kF8J06x8wERoJVKCCNY0+mgDZTjleIHKMceDEFMFdLx2209aGNBoSy75ypw0GsIf5ME8eiYht1IRCJJZrsr0rePM0hHTK8V4Plz7MzVMFENRmsx2IapIDEiX0yq11TPKUIj8//
ZMQsQ4SFhQUBAIZZuZ0TS4tEJhZwdbv1f0CUpo6yCuNJ29ikDgLuW32+t4XyrG00KeCxDnYlE0CDh3punh2K/Yf0eCQpyFx1n57h+j/
DJJAI2zqKEgyZKnzabenImizBSEeCynIGcPnw+6G0RL2er05vCp21nBwL0Bg40vc+0p8saxUKic8H08ayumhQakJvwsxqMD2Y2UUV5WGVZYEm0glG2urpakSsyL07nwxvDwUK7BYBjTarWvDpTDA0NZefm5gZnpxm/T88l6wJYXl5+SG6FTAf4jba1tTxYieQTLQRQu0E5W1nxTKW2800kSBq4yJa3GfL0eSrz6+p98bqGqqrE5QeQNLB/
cADXIjERnZJ813L2K8pYfeFfLr6FFEkd4+pGYUhhAwQCr/Fws010v5cBwWQNDDEXndVsbh0KTZFXi1WYvxk1B0IdtRDACZMvmYn/j8wmUyXYG3+lK/
eP9xu92Iq+IYYYY0Ib9q0+f4WItegw0kCwGLs+KKYGo0rsoKio6D2QmW0sakXcXcEUQMhYnUajCe3YSVMH5dSwwCxfpAEQLLAEWlggcXFw+ALEDQ6MFFIzDhUlp9MFYPT
i+sww+
+rVrdU6u396ua+vr6Csr0wms8wQ0tTvq1raDh5sP80rRwDvMLSNVlacZ9gqYgB9sp0kxrnBfQ0WE4G2JMGtYFavLAFivGSBoZCK8pUA0vc+KIkcDzQKADLo1Go6QeQwGx5
bnL8YN/ebDtI9GNdC/4BmmhJuaFmq04j7DILZFJA8M/27CZcPy7K/kwydcxJwHAcHiI1piHb1LQvUmYi/
oRmchRvgBw0YVUQTdsM9ZCFHXZXXJoI6cwwKHFHA6RG0QgxJ4HiGef87iYHT5IusJZR5KQuJmQYLeJbuXr9BxavI9cavZ6NuxGi8yqsESNqmmhXMWwVhB0EGLUYzjLWWayY
fdnI9PnVzc7N4QpXyU/efXqHw7EdjtZ4f8JXwgj7f6qqun00cFb5w4LkwlPdvgonAXqt0DNcPpUxIwghHnj6n01kL5QeHxbt2qwsLczf09XqxcAmDqnbL0ethD3iz/
wL61WkXNT+iTvBwWuB/
htuQF8qzj5qaGq4ysw50PUNQlR2AwBnNloonJuT6v+bJ+sUsjnFdLpdrCZQ2B2RSAAemZwwGy8pBxrJgvo4e07mInuv30Nd2z40DaeI9NBnxqAjXx0/
hsls09dSATJLKysqn4Hvfh6ps0AdFHLzZbP5mxYraeVBPDbz3gqCZ3n0zBd6ZQdYe3Ms5uw8MvdIuktjht7ezfj74eRSKSHVzN0PddwMDUyUyCrGUNBwdHy/
dHhEb7Kg8mmLG6HDK88fVz4KV6dGsmmRITjjsAkiz9RZMwpjpv+SRkw61PCKZcfcfctcydS9mGY1T9XX1++vs8fQtq4Fxp49M5rASiMesVMTYcvp4wrjTx8P/
0W1saWkduXbwdidLAQcgsxZ3vEdz+wUG0BxWenh44eFQ/Bs/7RwYRjH91kiY407iThnFhkoKfCNe297KknTy2k/f0ueu5sBF2xczHt63Vr/yl7A0702Zj/
sPb51EK0YDgqHP70kP16omwQ1WxnCxoR1StI3yhFtsZG0f/wX0X1lgwvR4cjBwZiIE/P2oSaTdeL21jYj4kzCrkM/+bxNoZ0+lN8ZmaF+32dyFuCd/
F7Ga45ML5jerGuB0Kfjsh6PaKPB3MMAA0B9PeGrJ5Gc+UyCYLcfVxwtfcyQ8MY0YBjSgqAdeX5JZXNEMd183Gj3VtekhTssfx9ZMDw303hrTF4Lc7nGG9KiaQDzsrKuo0
/NwJHeZueyy/3XBo3mfBPZ2QsY7JLNPw6kMhXmXJoJH1iYDvhn8Y64CH7K86w+trSYKgv2vYLytbv0q/zII+RuQqIYHxZ+HtgAAAABJRUSERkJggg=="
}
}

```

Arbeiten mit AI-Objekten



Erstellen und verwalten Sie Objekte außerhalb des Luftfahrzeugs. Flugzeuge und Objekte sind für die Gestaltung verfügbar.

Objekte erstellen und löschen

Die Objekte müssen zunächst bei MSFS registriert werden, bevor sie erstellt werden können.

Erstellen Sie ein Objekt mit `create_object`. Sie wählen einen `Namen`, mit dem Sie das Objekt referenzieren, während es existiert (um seine Eigenschaften zu aktualisieren und es später zu entfernen), und einen `Titel`, der der Name ist, der aus `aircraft.cfg` oder `sim.cfg` stammt.

Entfernen Sie ein Objekt, wenn Sie fertig sind, indem Sie `destroy_object` verwenden. Wenn Sie die Objekte nicht löschen, werden sie aufgeräumt, wenn die Mission endet.

Beispiel:

```
{ "create_object": {  
  "name": "my_object",  
  "title": "HPG Airbus H145 Ambulance",  
  "location": "$USER"  
}},  
{ "sleep": 60 },  
{ "destroy_object": "my_object" }
```

Objekt-Eigenschaften

Legen Sie Eigenschaften von Objekten fest, um sie zu gestalten.

`VAR 1` wird häufig so konfiguriert, dass der Animationszustand für das Objekt festgelegt wird.

Beispiel:

```
{"set": {"object": "my_object", "var": "MODE"}, "value": 1}
```

Objekt bewegen

Verwenden Sie `move_object`, um die Position und Ausrichtung eines Objekts sofort zu ändern.

Objekt Wegpunktnavigation

Verwenden Sie `drive_object`, um ein Objekt mit einer bestimmten Geschwindigkeit entlang einer Reihe von Wegpunkten zu bewegen.

Fliegende Objekte

TODO

Erstellen von Objektpaketen von Drittanbietern

Lesen Sie die [frühere Dokumentation](#)

Geräusche und Text zu Sprache

Es ist möglich, die Hauptsound-Datei `sound.xml` zu überschreiben, aber diese Methode wird nicht empfohlen, da nur ein Addon "gewinnt", wenn es mehrere gibt, die dies versuchen. Außerdem können Addons veralten, wenn sich die `sound.xml` des Flugzeugs ändert. Wo es möglich ist, kann der Voice Server zur Verwaltung von Sounds verwendet werden.

Built-in sounds

`play_audio` kann verschiedene eingebaute Töne abspielen.

Voice Server

Der Sprachserver ist ein externes Programm, das ausgeführt werden muss, während der Benutzer die Mission ausführt. Der Voice Server nimmt Befehle entgegen und spielt anschließend passende Sounds ab (außerhalb von MSFS).

Um den Sprachserver zu benutzen, muss er zunächst eingebunden werden. Rufen Sie `connect_voice_server` mit einer `on_connected` `COMMANDLIST` auf. Beachten Sie, dass Sie den Fall, dass der Sprachserver nicht verfügbar ist, angemessen behandeln sollten.

Beispiel:

```
{ "connect_voice_server": {
  "on_connected": [
    { "speak": "Speech activated." }
  ],
  "on_disconnected": [
    { "set_message": { "text": "No voice server available" } }
  ]
}}
```

Wenn Sie die obigen Schritte ausführen, sehen Sie entweder `No voice server available` oder hören `Speech activated`.

Sie können jederzeit überprüfen, ob der Sprachserver verbunden ist, indem Sie das boolesche Ergebnis von `{ "fn": "is_voice_server_connected" }` verwenden.

Sobald die Verbindung hergestellt ist, können Sie den Sprechbefehl in verschiedenen Varianten senden:

1. `{ "speak": "hello world" }` In diesem Fall sprechen wir einfach einen Text. `speak` kehrt sofort nach dem Senden des Befehls zurück.
2. `{ "speak": { "text": "hello world: {0}", "params": [99] } }` Hier verwenden wir `text/params`, um eine Zeichenkette zu erstellen.
3. `{ "speak": "hello world", "interrupt": 1 }`, Hier verwenden wir die `Interrupt`-Funktion, um alle aktiven Sprach-/Audioübertragungen abzubrechen und sofort mit der Wiedergabe dieser neuen Nachricht zu beginnen.
4. `{ "speak": "hello.wav", "is_audio_file": 1 }` Hier spielen wir den Sound `hello.wav` (aus dem Audio-Ordner) ab, die Direktive `is_audio_file` teilt dem Server mit, dass der Text, der normalerweise Sprache ist, stattdessen ein Dateiname ist.

Einrichtung eines kompatiblen Sprachservers (Fortgeschrittene)

Der Server muss auf `localhost:5997` laufen und vom Typ `Websocket` sein.

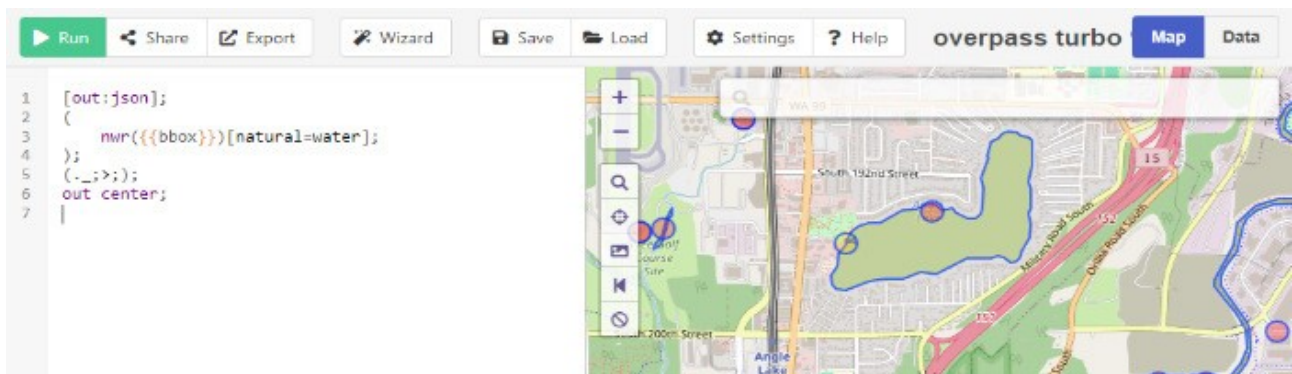
Nachrichten:

- `{ "Text": "", "FileName": "test1.wav", "Interrupt": false }`
- `{ "Text": "hello your name", "FileName": "", "Interrupt": false }`
- `{ "Text": "stop text", "FileName": "", "Interrupt": true }`

Voice Server Test Programm

```
{
  "title": "Voice Server Test Program",
  "briefing": [
    { "title": "Voice Server Test Program",
      {
        "buttonbar": [ { "title": "Connect to voice server", "commands": [ { "call_macro": "connect_voice" } ] },
        "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 0 }
      },
      {
        "text": "Not connected to voice server.",
        "color": "red",
        "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 0 }
      },
      {
        "text": "Connected to voice server.",
        "color": "green",
        "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 1 }
      },
      { "text": "-----" },
      { "text": ".wav file test1", "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 1 } },
      {
        "buttonbar": [ { "title": "Play", "commands": [
          { "speak": "test1.wav", "is_audio_file": 1 }
        ] },
        "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 1 }
      },
      { "text": "speech recognition test text", "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 1 } },
      {
        "buttonbar": [ { "title": "Speak", "commands": [
          { "speak": "speech recognition test text" }
        ] },
        "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 1 }
      },
      { "text": "INTERRUPT: stop text", "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 1 } },
      {
        "buttonbar": [ { "title": "Speak", "commands": [
          { "speak": "stop text", "interrupt": 1 }
        ] },
        "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 1 }
      },
      { "text": { "text": "Custom Message: {0}", "params": [ { "local": "message_textbox" } ] }, "show_condition": { "require":
        { "fn": "is_voice_server_connected" }, "eq": 1 } },
      { "textbox": "message_textbox",
        {
          "buttonbar": [ { "title": "Speak", "commands": [
            { "speak": { "local": "message_textbox" } }
          ] },
          "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 1 }
        },
      { "text": { "text": "Format Message: hello {0}", "params": [ { "local": "message_textbox" } ] }, "show_condition": { "require":
        { "fn": "is_voice_server_connected" }, "eq": 1 } },
      {
        "buttonbar": [ { "title": "Speak", "commands": [
          { "speak": { "text": "hello {0}", "params": [ { "local": "message_textbox" } ] } }
        ] },
        "show_condition": { "require": { "fn": "is_voice_server_connected" }, "eq": 1 }
      }
    ],
    "macros": {
      "connect_voice": [
        { "connect_voice_server": {
          "on_connected": [
            { "speak": "Speech activated." }
          ],
          "on_disconnected": [
            { "set_message": "No voice server available" }
          ]
        }
      ]
    }
  ],
  "objectives": [
    {
      "title": "Done",
      "commands": [
        { "set": { "local": "message_textbox", "value": "your name" },
          { "call_macro": "connect_voice" },
          { "sleep": "forever" }
        ]
      }
    ]
  ]
}
```

OpenStreetMap Data



OpenStreetMap- oder OSM-Daten sind weltweit verfügbar. OSM-Elemente definieren und kennzeichnen die Merkmale der Welt. OSM verfügt über drei Haupttypen von Elementen:

| Element | Remarks |
|----------|---|
| node | Ein Knoten stellt einen einzelnen Punkt dar. Manchmal werden kleine Objekte als Knoten markiert, aber bei größeren Objekten wird ihr Umfang stattdessen durch einen Weg definiert. Ein Knotenpunkt kann ein Punkt auf dem Straßen- oder Schienennetz, ein Baum oder ein Krankenhausgebäude sein. Sie müssen vorsichtig sein, da es immer häufiger vorkommt, dass Features mit einem Weg anstelle des Mittelpunkts gekennzeichnet werden. Durch Hinzufügen des Mittelpunkts (<code>out_center</code>) können Sie OSM bitten, Ihnen den Mittelpunkt eines Weges zu geben, was manchmal ebenfalls hilfreich sein kann. |
| way | Eine Trasse (way) ist ein Container für eine Liste von Knotenpunkten. Straßen und Eisenbahnen werden aus Wegen erstellt, die wiederum aus Knoten erstellt werden. Sie können den Weg und seine Metadaten sowie die vollständige Liste der in diesem Weg enthaltenen Knoten abrufen. Ein Gebiet ist ein geschlossener Weg. |
| relation | Eine Relation ist eine Möglichkeit, Knoten und Wege in eine Art Container einzubinden, der mit Tags versehen sein kann. Eine Relation kann für eine administrative Grenze wie eine Stadt oder ein Bundesland verwendet werden. |

OSM Entwickler Workflow

Sie sollten:

1. Entdecken Sie Features, mit denen Sie arbeiten möchten. Besuchen Sie [OpenStreetMap](#) und verwenden Sie das Tool "Query Features" (rechter Rand), um OSM-Daten an einem bestimmten Ort zu untersuchen.
2. Erstellen Sie die Abfrage, die diese Merkmale abrufen. Besuchen Sie [Overpass Turbo](#), um mit Abfragen zu arbeiten und sie an verschiedenen Stellen der Karte zu testen.

3. Wählen Sie die Art und Weise, wie Sie die Daten im Rahmen der Mission verwenden möchten. Sie könnten `create_location` und dann `query_random_result` (alle Ergebnisse der Abfrage abrufen, eines zufällig auswählen) oder `query_closest_result` (bei 100 Metern beginnen und erweitern, bis Sie das erste Ergebnis finden) verwenden.

Sobald Sie die gewünschten Daten gefunden und eine funktionierende Abfrage erstellt haben, integrieren Sie diese in die Mission und testen sie.

Beispiele für OSM-Abfragen

Alle folgenden Abfragen verwenden den Begrenzungsrahmen `bbox`. Vergessen Sie nicht, dass Sie die Anführungszeichen " wie folgt auslassen müssen: `\` in JSON.

Ermitteln Sie nahe gelegene Krankenhäuser (`amenity=hospital` tag). Wir erhalten Knoten- (`node`) und Weg- (`way`) Ergebnisse, da Krankenhäuser häufig auf jedem Weg getaggt sind. `Out center` fügt jedem Weg einen Schwerpunkt hinzu, der es uns ermöglicht, sie ähnlich wie Knoten zu behandeln.

```
[out:json];
(
  node({{bbox}})[amenity=hospital];
  way({{bbox}})[amenity=hospital];
);
out center;
```

Gliederung der Abfrage:

1. `[out:json];`: Konfigurieren Sie für die JSON-Ausgabe. Immer gleich.
2. `(` und `)`: Verbindung, die die beiden Gruppen von Ergebnissen in einer Liste zusammenfasst.
3. Ermitteln der Knoten innerhalb der Ansicht, die dem Tag entsprechen `amenity=hospital`.

```
node({{bbox}})[amenity=hospital];
```

- 4: Suche nach Wegen innerhalb der Ansicht, die dem Tag entsprechen `amenity=hospital`.

```
way({{bbox}})[amenity=hospital];
```

- 5: `out center`: Für Wege fügen Sie zu jedem Ergebnis einen Breiten- und Längenschwerpunkt (`lat/lon`) hinzu.

OpenStreetMap APIs

Es gibt leistungsfähige APIs für die Arbeit mit Daten von OpenStreetMap. Sie können nahe gelegene POIs entdecken sowie Polygone und andere Beziehungen untersuchen.

HINWEIS: Sie sollten immer `bbox` verwenden, das für Sie durch den angegebenen Ort/Radius ersetzt wird. Konvertieren Sie jeden Code, der `around` verwendet, in `bbox`.

| API | V1 or V2 | Anmerkung |
|------------------------------|----------|---|
| <code>create_location</code> | V1 | Erstellen Sie einen Ort aus einer Liste von Zonen. Immer noch nützlich. |
| <code>query_data</code> | V1 | Nützlich für Datenbankabfragen. |
| <code>query_country</code> | V1 | Ermittelt das Land für einen bestimmten Ort. Ebenfalls nützlich. |
| <code>osm_query_data</code> | V2 | Abfrage und anschließende Verarbeitung beliebiger OSM OverpassAPI-Daten |

Wenn Sie die V2-API verwenden, werden Sie diese Funktionen nutzen:

| API | Anmerkung |
|--------------------------------------|--|
| <code>osm_get_parent_ways</code> | Ermitteln der Wege, auf denen ein bestimmter Knoten existiert |
| <code>osm_get_connected_nodes</code> | Ermittelt die Knoten vor und nach diesem Knoten in der Art und Weise, in der er vorliegt |
| <code>osm_get_nodes</code> | Ermitteln Sie die Knoten auf diese Weise |
| <code>osm_get_all_ways</code> | Alle Wege innerhalb des Datensatzes ermitteln |
| <code>osm_get_all_nodes</code> | Alle Knoten innerhalb des Datensatzes ermitteln |
| <code>osm_get_closest_nodes</code> | Erhalten Sie eine geordnete Liste von Knoten, geordnet nach Entfernung |
| <code>osm_is_point_within_way</code> | Ermittelt einen booleschen Wert, der angibt, ob der Punkt innerhalb eines bestimmten geschlossenen Weges liegt |
| <code>osm_get_area_of_area</code> | Ermitteln der Fläche in Metern ² eines bestimmten geschlossenen Weges |

Beispiel für nahe gelegene Krankenhaus-Hubschrauberlandeplätze anzeigen

Dieses Beispiel verwendet `query_data`, um nahegelegene Elemente in der Datenbank `DB:H_HOSPITAL` (Krankenhaus mit Hubschrauberlandeplatz) zu finden.

```
{
  "title": "Show nearby hospital helipads",
  "author": "davux3",
  "api_version": 0.1,
  "aircraft": ["H145"],

  "objectives": [
    {
      "title": "Done",
      "commands": [

        {"query_data": {
          "query": "DB:H_HOSPITAL",
          "location": "$USER",
          "radius": 5000,
          "minRadius": 0,
          "bypass_commands": [
            {"#comment": "$ITEMS contains an array of results: type=way, lat, lon, tags. by convention center is copied down into
lat/lon"},
            {"for_each":{"param":"$ITEMS"},"do":[
              {"set":{"param":"loc"},"value":{"create_array":[
                {"struct":{"param":"$item"},"path":"lat"},
                {"struct":{"param":"$item"},"path":"lon"}
              ]}},
              {"set_map":{"add":{"point":{"location":{"param":"loc"},"icon":"ki_helipad"}}}}
            ]}
          ]
        }},
        {"sleep": "forever"}
      ]
    }
  ]
}
```

Beispiel für nahegelegene Umspannwerke anzeigen

Dieses Beispiel verwendet `query_data`, um nahe gelegene OSM-Elemente zu finden, die `power=substation` entsprechen.

```
{
  "title": "Show nearby power substations",
  "author": "davux3",
  "api_version": 0.1,
  "aircraft": ["H145"],
  "objectives": [
    {
      "title": "Done",
      "commands": [

        {"query_data": {
          "query": "[out:json]; (node({bbox})[power=substation]; area({bbox})[power=substation]); out center;",
          "location": "$USER",
          "radius": 2500,
          "minRadius": 0,
          "bypass_commands": [
            {"#comment": "$ITEMS contains an array of results: type=way, lat, lon, tags. by convention center is copied down into
lat/lon"},
            {"for_each":{"param":"$ITEMS"},"do":[
              {"set":{"param":"loc"},"value":{"create_array":[
                {"struct":{"param":"$item"},"path":"lat"},
                {"struct":{"param":"$item"},"path":"lon"}
              ]}},
              {"set_map":{"add":{"point":{"location":{"param":"loc"},"icon":"ki_helipad"}}}}
            ]}
          ]
        }},
        {"sleep": "forever"}
      ]
    }
  ]
}
```

Objekte auf Bahnübergängen

Dieses Beispiel greift auf die Straßen- und Schienendaten zu, um einen guten Ort für einen Unfall zwischen einem Zug und einem Schulbus zu ermitteln.

1. Finden Sie einen Bahnübergang in der Nähe
2. Anschluss der Straßen und Schienen an diesen Übergang
3. Stellen Sie einen Zug auf den Bahnübergang und einen Bus, der aus Richtung der Straße in den Zug hineinfährt.

```
{
  "title": "Train Level Crossing Test",
  "api_version": 0.1,
  "aircraft": ["H145"],
  "locations": {
    "LOC": "$USER"
  },
  "briefing": [
    { "buttonbar": [ { "title": "teleport to accident", "commands": [ { "call_macro": "adv_teleport", "params": { "loc": "accident_location" } } ] } ] } ]
  ],
  "macros": {
    "adv_teleport": [
      { "#comment": "param: loc",
        "trigger": "K:SLEW_ON",
        "sleep": 1,
        "teleport_to": { "param": "loc" },
        "set": { "var": ["A:PLANE ALTITUDE", "feet"], "value": -1000 },
        "set": { "var": ["A:PLANE HEADING DEGREES TRUE", "degrees"], "value": { "param": "hdg" } },
        "sleep": 2,
        "trigger": "K:SLEW_OFF"
      }
    ],
    "create_closest_railway_crossing_accident_scene": [
      { "#comment": "param - location",
        "#comment": "first get the closest result and get the node ID",
        "create_location": "accident_location", "zones": [
          { "zone": {
              "zone_type": "query_random_result",
              "radius": 10000,
              "query": "[out:json];node({bbox})[railway='level_crossing'];out center;",
              "location": { "param": "location",
                "commands": [ { "set": { "param": "railway_crossing_node_id", "value": { "param": "$LOCATION:ID" } } ]
              }
            }
          ]
        },
        "#comment": "query nearby highway and railway (very close because it's already the exact position). We get all the nodes in those ways",
        "osm_query_data": [
          "[out:json];(way({bbox})[railway]; way({bbox})[highway]);(._;>);out;",
          "location": "accident_location",
          "size": 1,
          "result": "my_data"
        ],
        "#comment": "find all the connected nodes and put them into highway and railway lists",
        "set": { "param": "highway_nodes", "value": { "create_array": [] } },
        "set": { "param": "railway_nodes", "value": { "create_array": [] } },
        "#comment": "get the nodes connected to our root (intersection) node. these are nodes before/after the root node, on any ways that the node is apart of",
        "osm_get_connected_nodes": { "param": "railway_crossing_node_id", "data": { "param": "my_data",
          "result": "my_nodes_connected_to_nearest_node",
          "for_each": { "param": "my_nodes_connected_to_nearest_node", "do": [
            { "#comment": "draw a line with a different color for each leg of the intersection",
              "set": { "param": "first_way_key", "value": { "struct": { "object": keys: { "struct": { "param": "$item", "path": "_ways" } }, "index": 0 } } },
              "set": { "param": "is_road", "value": { "struct": { "param": "$item", "path": "_ways.{first_way_key}.tags.highway" } },
                "#comment": { "text": "is_road: {0}", "params": [ { "json:stringify": { "param": "is_road" } } ] } } },
              "if": { "param": "is_road", "ne": null, "then": [
                { "modify_array": { "param": "highway_nodes", "append": { "create_array": [ { "struct": { "param": "$item", "path": "lat" } } ] } },
                { "struct": { "param": "$item", "path": "lon" } } ] } },
              "else": [
                { "modify_array": { "param": "railway_nodes", "append": { "create_array": [ { "struct": { "param": "$item", "path": "lat" } } ] } },
                { "struct": { "param": "$item", "path": "lon" } } ] } } ] } } ],
        "return": { "create_struct": {
          "nodeId": { "param": "railway_crossing_node_id",
            "highway_nodes": { "param": "highway_nodes",
              "railway_nodes": { "param": "railway_nodes",
                "data": { "param": "my_data",

```

```

    "location": {"resolve_location": "accident_location"}
  }}}
  ]
},
"objectives": [
  {
    "title": "Initializing...",
    "commands": [
      {
        "#comment": "find the closest railway/road level crossing",
        "call_macro": "create_closest_railway_crossing_accident_scene", "params": {
          "location": "$USER"
        }, "result": "crossing_ret",
      },
      {
        "#comment": "get the extracted data",
        "set": {"param": "railway_crossing_node_id", "value": {"struct": {"param": "crossing_ret"}, "path": "nodeId"}},
        "set": {"param": "railway_crossing_location", "value": {"struct": {"param": "crossing_ret"}, "path": "location"}},
        "set": {"param": "highway_nodes", "value": {"struct": {"param": "crossing_ret"}, "path": "highway_nodes"}},
        "set": {"param": "railway_nodes", "value": {"struct": {"param": "crossing_ret"}, "path": "railway_nodes"}},
      },
      {
        "#comment": "visualize the results by putting an icon at the crossing and drawing a line to the nearby highway/railway nodes",
        "set_map": {"add": {"point": {"location": {"param": "railway_crossing_location"}, "icon": "ki_waypoint_blue"}}},
        "set_map": {"add": {"line": {"points": {"create_array": [
          {"param": "railway_crossing_location"},
          {"struct": {"param": "highway_nodes"}, "index": "0"}
        ]}, "stroke": {"create_struct": {"color": "red", "width": 2}}}}},
        "set_map": {"add": {"line": {"points": {"create_array": [
          {"param": "railway_crossing_location"},
          {"struct": {"param": "railway_nodes"}, "index": "0"}
        ]}, "stroke": {"create_struct": {"color": "blue", "width": 2}}}}},
      },
      {
        "#comment": "calculate bearings",
        "set": {"param": "train_brg", "value": {"bearing": {"from": {"param": "railway_crossing_location"}, "to": {"struct": {"param": "railway_nodes"}, "index": "0"}}}},
        "set": {"param": "crash_brg", "value": {"bearing": {"from": {"param": "railway_crossing_location"}, "to": {"struct": {"param": "highway_nodes"}, "index": "0"}}}},
      },
      {
        "#comment": "place train and crash objects",
        "create_object": {"name": "train", "title": "Airbus H145 Train", "location": {"bearing2": {"param": "train_brg"}, "dist": 20, "object": {"param": "railway_crossing_location"}}},
        "track_object": {"object": "train", "icon": "ki_helipad"},
      },
      {
        "create_object": {"name": "crash1", "title": "Airbus H145 School bus", "location": {"bearing2": {"param": "crash_brg"}, "dist": -7, "object": {"param": "railway_crossing_location"}}},
        "track_object": {"object": "crash1", "icon": "ki_helipad"},
      },
      {"sleep": "forever"}
    ]
  }
]
}

```

Testprogramm für das Straßennetz

Dieses Beispiel zeigt:

1. Abfrage eines zweiten Teils des Straßennetzes
2. Zeichnet rote Linien auf alle Straßen
3. Findet die nächstgelegene Straßenkreuzung zum LOC-Standort.
4. Zieht eine andersfarbige Linie von jedem Schenkel des Schnittpunkts zur Mitte.

```

{
  "title": "Road Network Test",
  "api_version": 0.1,
  "aircraft": ["H145"],
  "data": {
    "colors": ["hotpink", "blue", "orange", "yellow", "green", "purple"]
  },
  "locations": {
    "LOC": {"bearing": 270, "dist": 500}
  },
  "objectives": [
    {
      "title": "Initializing...",
      "commands": [
        {
          "#comment": "Query a block of road network data and save it into my_data",
          "osm_query_data":
            "[out:json];way({bbox})[highway~\"^(motorway|trunk|primary|secondary|unclassified|residential|living_street|service|tertiary|motorway|trunk|primary|secondary|tertiary)_link\"]$\";(.;>);out;",
          "location": "LOC",
          "size": 600,
          "result": "my_data"
        },
      ],
    }
  ],
}

```



```

    {"#comment":"extract the list of ways into my_ways, and then loop over them and draw them all on the map"},
    {"osm_get_all_ways":{"param":"my_data"}, "result":"my_ways"},
    {"for_each":{"param":"my_ways"},"do":[
      {"#comment":"create a list and then get a list of all the nodes in my_ways. store that list of nodes into
my_nodes_on_way"},
      {"set":{"param":"node_location_list"}, "value":{"create_array":[]}},
      {"osm_get_nodes":{"struct":{"param":"$item"}, "path":"id"}, "data":{"param":"my_data"}, "result":"my_nodes_on_way"},
      {"#comment":"create a [lat,lon] array from each node and put it into the results list"},
      {"for_each":{"param":"my_nodes_on_way"},"do":[
        {"modify_array":{"param":"node_location_list"}, "append":{"create_array":[
          {"struct":{"param":"$item"},"path":"lat"},
          {"struct":{"param":"$item"},"path":"lon"}
        ]}},
        {"sleep":0.001}
      ]},
      {"#comment":"draw the road"},
      {"set_map":{"add":{"line":{"points":{"param":"node_location_list"}, "stroke":{"color":"red", "width":2}}}}}
    ]},

    {"#comment":"get an ordered list of all the nodes by their distance from LOC"},
    {"osm_get_closest_nodes": "LOC", "data": {"param":"my_data"}, "result":"my_closest_nodes"},

    {"#comment":"go through the result nodes and pick only those with at least 2 parents (an intersection)"},
    {"set":{"param":"closest_node"},"value":null},
    {"for_each":{"param":"my_closest_nodes"},"do":[
      {"osm_get_parent_ways":{"struct":{"param":"$item"},"path":"id"}, "data": {"param":"my_data"}, "result":"parents"},
      {"if":{"struct":{"param":"parents"}, "path":"length"},"gt": 1,"then":[
        {"set":{"param":"closest_node"},"value":{"param":"$item"}},
        {"break":1}
      ]},
      {"#comment":"do get parent and check for more than one for an intersection"},
      {"sleep":0.001}
    ]},
    {"set":{"param":"closest_location"}, "value":{"create_array":[
      {"struct": {"param":"closest_node"},"path":"lat"},
      {"struct": {"param":"closest_node"},"path":"lon"}
    ]}},

    {"set_map":{"add":{"point":{"location":{"param":"closest_location"}, "icon":"ki_waypoint_blue"}}}},

    {"#comment":"get the nodes connected to our root (intersection) node. these are nodes before/after the root node, on any
ways that the node is apart of"},
    {"osm_get_connected_nodes":{"struct":{"param":"closest_node"},"path":"id"}, "data": {"param":"my_data"},
"result":"my_nodes_connected_to_nearest_node"},
    {"for_each":{"param":"my_nodes_connected_to_nearest_node"},"do":[
      {"#comment":"draw a line with a different color for each leg of the intersection"},
      {"set_map":{"add":{"line":{"points":{"create_array":[
        {"param":"closest_location"},
        {"create_array":[
          {"struct":{"param":"$item"},"path":"lat"},
          {"struct":{"param":"$item"},"path":"lon"}
        ]}
      ]}, "stroke":{"create_struct":{"
"color":{"struct":{"static": "colors"}, "index": {"param":"$index"}}, "width":2}}}}}
    ]},

    {"#comment":"save some debug stuff"},
    {"set":{"local":"node_location_list2"},"value":{"param":"node_location_list2"}},
    {"set":{"local":"my_closest_nodes"},"value":{"param":"my_closest_nodes"}},
    {"set":{"local":"my_ways"},"value":{"param":"my_ways"}},
    {"set":{"local":"my_data"},"value":{"param":"my_data"}},

    {"sleep": "forever"}
  ]
}
}
}

```

Testprogramm für Wasserpolygone

Sie müssen sich im Wesentlichen auf einem See befinden, damit dies funktioniert. Ändern Sie LOC so, dass Sie sich in der Mitte eines Sees befinden, oder vergrößern Sie die Fläche. Sie sollten nur ein Gewässer finden, da sonst die Gebietsmeldung überschrieben wird.

Dieses Beispiel:

1. Abfrage nahegelegender Wasserpolygone
2. Zeichnet eine Reihe von Punkten auf die Karte, um festzustellen, ob sie sich innerhalb oder außerhalb des Wassers befinden

```

{
  "title": "get very nearby water and get area",
  "api_version": 0.1,
  "aircraft": ["H145"],
  "locations": {
    "LOC": "$USER"
  },
  "objectives": [
    {
      "title": "Initializing...",
      "commands": [
        {
          "#comment": "Query a block of road network data and save it into my_data",
          "osm_query_data": {
            "out": "json",
            "nw": "{bbox}",
            "natural": "water",
            "location": "LOC",
            "size": 200,
            "result": "my_data"
          },
          "#comment": "extract the list of ways into my_ways, and then loop over them and draw them all on the map",
          "osm_get_all_ways": {
            "param": "my_data",
            "result": "my_ways"
          },
          "for_each": {
            "param": "my_ways",
            "do": [
              "set": {
                "param": "way",
                "value": {
                  "param": "$item"
                }
              },
              "#comment": "create a list and then get a list of all the nodes in my_ways. store that list of nodes into my_nodes_on_way",
              "set": {
                "param": "node_location_list",
                "value": {
                  "create_array": []
                }
              },
              "osm_get_nodes": {
                "struct": {
                  "param": "$item",
                  "path": "id",
                  "data": {
                    "param": "my_data",
                    "result": "my_nodes_on_way"
                  }
                },
                "#comment": "create a [lat,lon] array from each node and put it into the results list",
                "for_each": {
                  "param": "my_nodes_on_way",
                  "do": [
                    "modify_array": {
                      "param": "node_location_list",
                      "append": {
                        "create_array": [
                          {
                            "struct": {
                              "param": "$item",
                              "path": "lat"
                            },
                            {
                              "struct": {
                                "param": "$item",
                                "path": "lon"
                              }
                            }
                        ]
                      },
                      "sleep": 0.001
                    }
                  ]
                },
                "osm_get_area_of_area": {
                  "struct": {
                    "param": "way",
                    "path": "id",
                    "data": {
                      "param": "my_data",
                      "result": "way_area"
                    }
                  },
                  "set_message": {
                    "text": "way size {way_area} meters"
                  },
                  "#comment": "draw the road",
                  "for_each": {
                    "create_array": 10,
                    "do": [
                      "set": {
                        "param": "dist",
                        "value": {
                          "multiply": [
                            {
                              "param": "$index",
                              "value": 30
                            }
                          ]
                        }
                      },
                      "for_each": {
                        "create_array": 36,
                        "do": [
                          "set": {
                            "param": "brg",
                            "value": {
                              "multiply": [
                                {
                                  "param": "$index",
                                  "value": 10
                                }
                              ]
                            }
                          },
                          "osm_is_point_within_way": {
                            "struct": {
                              "param": "way",
                              "path": "id",
                              "location": {
                                "bearing": {
                                  "param": "brg"
                                },
                                "dist": {
                                  "param": "dist"
                                }
                              },
                              "data": {
                                "param": "my_data",
                                "result": "is_in"
                              }
                            },
                            "if": {
                              "param": "is_in",
                              "eq": 1,
                              "then": [
                                "set_map": {
                                  "add": {
                                    "point": {
                                      "location": {
                                        "bearing": {
                                          "param": "brg"
                                        },
                                        "dist": {
                                          "param": "dist"
                                        }
                                      },
                                      "icon": "ki_helipad"
                                    }
                                  }
                                },
                                "else": [
                                  "set_map": {
                                    "add": {
                                      "point": {
                                        "location": {
                                          "bearing": {
                                            "param": "brg"
                                          },
                                          "dist": {
                                            "param": "dist"
                                          }
                                        },
                                        "icon": "ki_waypoint_blue"
                                      }
                                    }
                                  }
                                ]
                              },
                              "sleep": 0.001
                            }
                          ]
                        }
                      },
                      "set_map": {
                        "add": {
                          "line": {
                            "points": {
                              "param": "node_location_list"
                            },
                            "stroke": {
                              "color": "red",
                              "width": 2
                            }
                          }
                        }
                      },
                      "sleep": "forever"
                    ]
                  }
                }
              ]
            }
          }
        ]
      }
    }
  ]
}

```

Gebäude Test

1. Ermittelt die nahegelegenen Gebäude und umrandet sie in Rot.

```

{
  "title": "Get nearby buildings and outline them",
  "api_version": 0.1,
  "aircraft": ["H145"],
  "locations": {
    "LOC": "$USER"
  },
  "objectives": [
    {

```

```

    "title": "Initializing...",
    "commands": [
      {
        "#comment": "Query a block of road network data and save it into
my_data",
        "osm_query_data": {
          "out": "json",
          "way": "way({{bbox}})[building];(._;>);out;",
          "location": "LOC",
          "size": 100,
          "result": "my_data"
        }
      },
      {
        "#comment": "extract the list of ways into my_ways, and then loop over
them and draw them all on the map",
        "osm_get_all_ways": {
          "param": "my_data",
          "result": "my_ways"
        },
        "for_each": {
          "param": "my_ways",
          "do": [
            {
              "#comment": "create a list and then get a list of all the nodes in
my_ways. store that list of nodes into my_nodes_on_way",
              "set": {
                "param": "node_location_list",
                "value": {
                  "create_array": []
                }
              },
              "osm_get_nodes": {
                "struct": {
                  "param": "$item",
                  "path": "id",
                  "data": {
                    "param": "my_data",
                    "result": "my_nodes_on_way"
                  }
                },
                "#comment": "create a [lat,lon] array from each node and put it into
the results list",
                "for_each": {
                  "param": "my_nodes_on_way",
                  "do": [
                    {
                      "modify_array": {
                        "param": "node_location_list",
                        "append": {
                          "create_array": [
                            {
                              "struct": {
                                "param": "$item",
                                "path": "lat"
                              },
                              {
                                "struct": {
                                  "param": "$item",
                                  "path": "lon"
                                }
                              }
                            ]
                          }
                        }
                      },
                      "sleep": 0.001
                    }
                  ]
                },
                "#comment": "draw the building",
                "set_map": {
                  "add": {
                    "line": {
                      "points": {
                        "param": "node_location_list"
                      }
                    }
                  }
                },
                "stroke": {
                  "color": "red",
                  "width": 2
                }
              }
            }
          ]
        }
      },
      {
        "sleep": "forever"
      }
    ]
  }
}
}
}

```

Bedienung des Hebezeugs



Es gibt V1- und V2-APIs für die Handhabung des Hebezeugs.

Hoist APIs

| API | V1 or V2 | Remarks |
|--------------------------------------|----------|---|
| fn.HOIST_SEND_TO_GROUND | V1 | Schickt das Hebezeug auf den Boden, wenn die Bedingungen erfüllt sind |
| fn.HOIST_REEL_UP_AND_STOW | V1 | Aufrollen und Verstauen der Winde |
| fn.HOIST_REEL_UP | V1 | Nur das Hebezeug aufspulen |
| hoist_control | V2 | Ermöglicht die direkte Kontrolle über das Auf- und Abspulen. |
| fn.hoist_get_reel_distance:ft | V2 | Entfernung, die die Spule ausgefahren wird, unterstützt auch :m |
| fn.hoist_get_distance_from_ground:ft | V2 | Abstand des Hebezeugs zum Boden, unterstützt auch :m |

Hoist SDK Variablen

| L:Var | Remarks |
|---------------------------|---|
| L:H145_SDK_HOIST_MODE | 0: auto, 1: manual |
| L:H145_SDK_HOIST_CONTROL | Handsteuersignal |
| L:H145_SDK_HOIST_CABLE_FT | Entfernung, um die die Spule ausgefahren wird |
| L:H145_SDK_EQUIP_HOIST | 0: nicht installiert, 1: installiert |
| L:H145_SDK_OH_HOIST | Position des Kopfschalters (und Armstellung) |

Hoist SDK Event

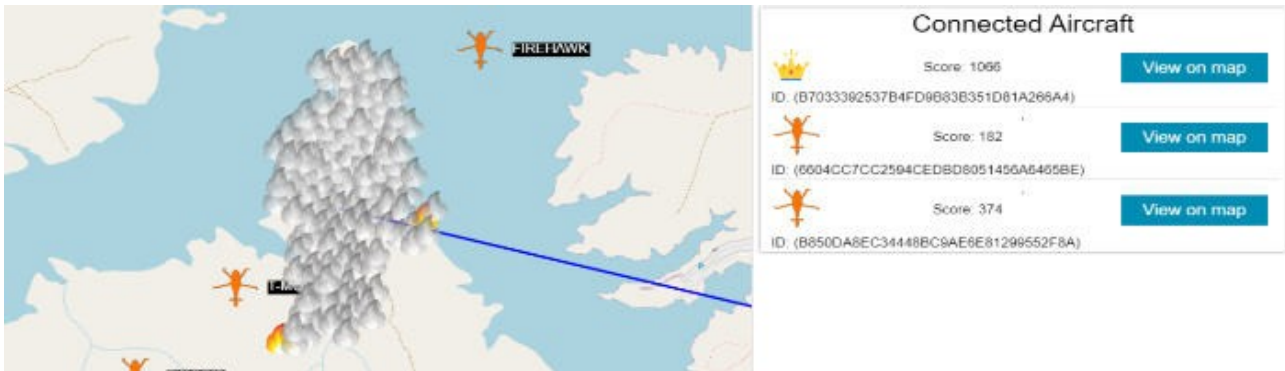
| L:Event | Remarks |
|--------------------------------------|---|
| H:H145_SDK_EQUIP_HOIST_ON | Einrichtung der Ausrüstung - Hoist ON |
| H:H145_SDK_EQUIP_HOIST_OFF | Einrichtung der Ausrüstung - Hoist OFF |
| H:H145_SDK_EQUIP_HOIST_TOGGLE | Einrichtung der Ausrüstung - Hoist TOGGLE |
| H:H145_SDK_HOIST_CONTROL_MODE_AUTO | Hoist - Hoist Mode AUTO |
| H:H145_SDK_HOIST_CONTROL_MODE_MANUAL | Hoist - Hoist Mode MANUAL |
| H:H145_SDK_HOIST_CONTROL_MOTOR_UP | Hoist - Hoist Manuelle Steuerung UP |
| H:H145_SDK_HOIST_CONTROL_MOTOR_STOP | Hoist - Hoist Manuelle Steuerung STOP |
| H:H145_SDK_HOIST_CONTROL_MOTOR_DOWN | Hoist - Hoist Manuelle Steuerung DOWN |

| L:Event | Remarks |
|---|---|
| H:H145_SDK_HOIST_CONTROL_MOTOR_MOMENTARY_UP | Hoist - Hoist Manuelle Steuerung MOMENTARY_UP |
| H:H145_SDK_HOIST_CONTROL_MOTOR_MOMENTARY_DOWN | Hoist - Hoist Manuelle Steuerung MOMENTARY_DOWN |
| H:H145_SDK_HOIST_ARM_STOW | Hoist - Hoist Arm STOW |
| H:H145_SDK_HOIST_ARM_DEPLOY | Hoist - Hoist Arm DEPLOY |
| H:H145_SDK_HOIST_EASY_STOWED | Hoist - Select Verstaut |
| H:H145_SDK_HOIST_EASY_DEPLOYED | Hoist - Select Bereitgestellt |
| H:H145_SDK_HOIST_EASY_CREW | Hoist - Select Crew |
| H:H145_SDK_HOIST_EASY_WORKER | Hoist - Select Worker |
| H:H145_SDK_HOIST_EASY_STRETCHER_ANDCREW | Hoist - Select Stetcher_And_Crew |
| H:H145_SDK_HOIST_EASY_SURVIVOR1_ANDCREW | Hoist - Select Überlebender_1_And_Crew |
| H:H145_SDK_HOIST_EASY_SURVIVOR2_ANDCREW | Hoist - Select Überlebender_2_And_Crew |
| H:H145_SDK_HOIST_EASY_TOOLBAG | Hoist - Select Werkzeugkoffer |
| H:H145_SDK_HOIST_EASY_CONTAINERS | Hoist - Select Behälter |
| H:H145_SDK_HOIST_EASY_HOSE | Hoist - Select Hose |

Hoist test program

```
{
  "title": "Hoist Test Program",
  "author": "davux3",
  "api_version": 0.1,
  "aircraft": ["H145"],
  "briefing": [
    {
      "text": "Extended length: {0} ft",
      "params": [ {"tofixed": {"fn": "hoist_get_reel_distance:ft"}, "digits": 2} ]
    },
    {
      "text": "Aircraft distance to ground: {0} ft",
      "params": [ {"tofixed": {"subtract": {"var": "RADIO HEIGHT", "feet"}}, 5.45}, {"digits": 2} ]
    },
    {
      "text": "Hoist distance to ground: {0} ft",
      "params": [ {"tofixed": {"fn": "hoist_get_distance_from_ground:ft"}, "digits": 2} ]
    }
  ],
  "buttonbar": [
    {
      "title": "Retract",
      "commands": [ {"hoist_control": "reel_up", "speed": {"scale": [{"local": "speed_slider"}, 0, 100, 0.01, 2]} ]
    },
    {
      "title": "Extend",
      "commands": [ {"hoist_control": "reel_down", "speed": {"scale": [{"local": "speed_slider"}, 0, 100, 0.01, 2]} ]
    }
  ],
  {
    "text": "Motor speed factor:",
    "slider": { "min": 0, "max": 100, "local": "speed_slider" }
  },
  "objectives": [
    {
      "title": "Done",
      "commands": [
        {
          "set": {"local": "speed_slider", "value": 10},
          "sleep": "forever"
        }
      ]
    }
  ]
}
```

Multiplayer Missionen



Multiplayer-Missionen sind Missionen, bei denen alle Mitspieler mit einem gemeinsamen Server verbunden sind. Dieser Server nimmt Befehle von Spielern entgegen und sendet Benachrichtigungen an die Spieler. Dies ermöglicht die Kommunikation von Spieler zu Spieler sowie das Hinzufügen von Web-Operatoren zur Interaktion mit den Spielern.

Die Missionen basieren auf gemeinsam genutzten Daten, die sich verändern, und die Ereignisse werden als Ergebnis dieser Veränderungen erzeugt. Alle Clients können Bereiche der Daten abonnieren und werden dann über alle Änderungen informiert. Die Konsistenz wird durch die Pufferung von Client und Server gewährleistet, die eine ordnungsgemäße Zustellung der Nachrichten ermöglicht. Clients, die ihre Verbindung unterbrechen und wiederherstellen, bekommen ihre Nachrichten automatisch zugestellt.

Befehle beim Beenden des Servers

Der Server interpretiert die gemeinsamen Daten der `terminationCommands`. Dieser Abschnitt ist speziell für Clients gedacht, um Befehle zu registrieren, die ausgeführt werden, wenn die Verbindung unterbrochen wird, und die dann vom Server gelöscht werden, nachdem eine gewisse Zeit lang keine erneute Verbindung hergestellt wurde. `terminationCommands` sind sehr wichtig, da sie sicherstellen, dass die Serverdaten nicht veraltet sind, auch wenn die Verbindung von Spielern unterbrochen wird.

Im Allgemeinen möchten Sie sich selbst (mit einer ID) zu einem `connectedAircraft` hinzufügen. Aus diesem Grund sollten Sie auch einen `terminationCommands.{id}` einrichten, der `connectedAircraft.{id}` entfernt. Auf diese Weise kann der Server beim Trennen der Verbindung automatisch für Sie aufräumen, und die Clients werden über den Löschvorgang benachrichtigt, falls sie entsprechend reagieren müssen.

Gemeinsam genutzte Daten

Gemeinsame Daten sind die Grundlage der Multiplayer-Plattform. Der Server speichert beliebige Werte, und die Clients können sich für Aktualisierungen dieser Werte anmelden. Jeder Client gibt sorgfältig Befehle zum Aktualisieren (`update`) und Löschen (`delete`) von Daten aus, wobei eine

Richtlinie (`policy`) verwendet wird, um Konflikte zu vermeiden und die Zusammenführung von Befehlen zu ermöglichen.

Sowohl im Luftfahrzeug als auch im Web können Sie `set_shared_data` verwenden, um diese Befehle zu erteilen. Im Flugzeug ist der `MultiplayerClient` Ihr Zugang zu den Multiplayer-Daten.

MultiplayerClient

`MultiplayerClient` ist der Typ des Objekts, das von `fn.create_multiplayer_connection` zurück geliefert wird. Dieses Objekt stellt die Schnittstelle zum Multiplayer-Server dar und verfügt über verschiedene Funktionen, die aufgerufen werden können, und über einen Status, auf den zugegriffen werden kann.

| Funktion | Parameter | Anmerkung |
|-----------|--|--|
| Connect | <code>url</code> , <code>userId</code> , <code>roomId</code> , <code>roomPassword</code> | Stellt eine Verbindung zum Server her. |
| Subscribe | <code>path</code> , <code>callback<object></code> | Abonnieren eines Pfades und Abrufen der aktuellen Daten per Callback |
| Get | <code>path</code> , <code>callback<object></code> | Holt einen Pfad und ruft die aktuellen Daten per Callback ab |
| Send | <code>message</code> | Sendet ein Nachrichtenobjekt |
| Close | None. | Trennen und Löschen der Verbindung |

Message Typen:

| Message Typ | Parameter | Anmerkung |
|-------------|---|---|
| read | <code>path</code> , <code>value</code> | Der Server hat einen Fehler bezüglich eines kürzlich von Ihnen gesendeten Befehls gemeldet. |
| update | <code>value</code> , <code>value</code> , <code>policy</code> | Der Server hat einen Fehler bezüglich eines kürzlich von Ihnen gesendeten Befehls gemeldet. |
| delete | <code>path</code> | Der Server sendet Ihnen Daten über gemeinsame Datenänderungen. |

| Policy | Anmerkung |
|--------------|---|
| delta | Wert ist relativ. |
| no_overwrite | Aktualisierung ignorieren, wenn der Pfad existiert. |

| Eigenschaft | Anmerkung |
|-------------|--------------------|
| Status | Connection status. |

| Event Handler | Parameter | Anmerkung |
|---------------|-----------------------------|---|
| OnError | <code>error</code> (string) | Der Server hat einen Fehler bezüglich eines kürzlich von Ihnen gesendeten Befehls gemeldet. |
| OnMessage | <code>data</code> (object) | Der Server sendet Ihnen Daten über geteilte Datenänderungen. |

Werte für den Verbindungsstatus:

| Status | Anmerkung |
|--------------|--|
| Unknown | Standardzustand, d. h. der Status vor dem Aufruf von connect. |
| Disconnected | Die Verbindung wurde unterbrochen, es wird automatisch ein neuer Verbindungsversuch unternommen. |
| Connecting | Verbindung mit dem Server herstellen. |
| LoggingIn | Server ist verbunden, Handshake läuft. |
| LoginFailed | Fehlerhaft. Die Anmeldung war nicht erfolgreich. |
| Connected | Zurzeit verbunden. |

Multiplayer-Beispiel für einfache Punktevergabe

Bei dieser Mission wird eine Punktetabelle erstellt, und jedes Flugzeug hat eine Schaltfläche, mit der die Punktzahl für den jeweiligen Spieler festgelegt werden kann.

Zusätzliche Merkmale:

- Flugpläne auf der Web-Client-Karte anzeigen
- Die Clients bereinigen ihre `connectedAircraft`- und `terminationCommands`-Einträge, nicht aber ihren Punktestand.

```
{
  "title": "Multiplayer Score Mission Test Program",
  "author": "davux3",
  "api_version": 0.1,
  "aircraft": ["H145"],
  "data": {
    "server_url": "wss://5ed547d.online-server.cloud/mpserver/ws",
    "create_room_url": "https://davux.com/dispatcher/",
    "webConfig": {
      "flightPlans": {
        "type": "map_line",
        "source": {"static": "flightPlans"},
        "name": "Flight Plan",
        "stroke": {"no_resolve": {"color": "#d303fc", "width": 2}},
        "icon": {"static": "icons.wp_blue"}
      },
      "connectedAircraftIcons": {
        "type": "map_point",
        "source": {"static": "connectedAircraft"},
        "name": "Connected Aircraft",
        "text": "{UserName}",
        "icon": {"static": "icons.h160_icon"}
      },
      "scoreList": {
        "type": "list",
        "source": {"static": "gameScores"},
        "title": "Game Scores",
        "emptyText": "No players have connected yet.",
        "rows": {
          "row0": {
            "1": {"text": "{UserName}"},
            "2": {"text": "Total Score: {0}", "params": [ {"round": {"param": "Score"} ]},
            "3": {"text": ""}
          }
        }
      },
      "connectedAircraftList": {
        "type": "list",
        "source": {"static": "connectedAircraft"},
        "title": "Connected Aircraft",
        "emptyText": "No aircraft are connected right now",
        "rows": {
          "row0": {
            "1": {"icon": {"static": "icons.h160_icon"}},
            "2": {"text": "{UserName}"},
            "4": {"button": "View", "commands": [ {"set_map_center": {"param": "location"}, "zoom": 16} ]}
          }
        }
      }
    }
  }
}
```



```

    }
  },
  "briefing": [
    { "#comment": [
      "MP_MODE ... 0: not set, 1: offline, 2: online"
    ] },
    { "title": "Mission Initial Setup", "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
    { "buttonbar": [
      { "title": "Offline (Single player)", "commands": [ { "set": { "local": "MP_MODE" }, "value": 1 } ] },
      { "title": "Online (Multiplayer)", "commands": [ { "call_macro": "mp_open_login_dialog" } ] }
    ], "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
    { "title": "Multiplayer (Online)", "show_condition": { "require": { "local": "MP_MODE" }, "eq": 2 } },
    { "buttonbar": [
      { "title": "View Multiplayer Status", "commands": [ { "call_macro": "mp_open_login_dialog" } ] }
    ], "show_condition": { "require": { "local": "MP_MODE" }, "eq": 2 } },
    { "title": "Game Score", "show_condition": { "require": { "local": "MP_MODE" }, "ne": 0 } },
    { "text": "My score: {local:MY_SCORE}", "show_condition": { "require": { "local": "MP_MODE" }, "ne": 0 } },
    { "buttonbar": [
      {
        "title": "Increment My Score",
        "commands": [
          { "set": { "local": "MY_SCORE" }, "value": { "add": [ { "local": "MY_SCORE" }, 1 ] } },
          { "set_shared_data": "update", "path": "gameScores.{service_auth}.Score", "value": { "local": "MY_SCORE" } }
        ]
      }
    ] },
    { "show_condition": { "require": { "local": "MP_MODE" }, "ne": 0 } }
  ],
  "events": {
    "ON_MISSION_ABORTING": {
      "commands": [ { "call_macro": "mp_aborting_mission" } ]
    }
  },
  "macros": {
    "mp_open_login_dialog": [
      { "#comment": "Show the login dialog dispatch (or multiplayer status)",
        "set_dispatch": [
          { "buttonbar": [ { "title": "<- Back to briefing", "commands": [ { "set_briefing_dialog": 1 } ] } ] },
          { "title": "Log in", "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
          { "text": "You are playing offline.", "show_condition": { "require": { "local": "MP_MODE" }, "eq": 1 } },
          { "text": "User Id: {0}", "params": [ { "local": "service_auth" } ], "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
          { "text": "User Name:", "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
          { "textbox": "mp_userName", "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
          { "text": "Room:", "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
          { "textbox": "mp_room", "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
          { "text": "Password:", "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
          { "textbox": "mp_password", "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
          { "buttonbar": [
            { "title": "Create Room (Opens on PC)", "commands": [ { "open_url": { "static": "create_room_url"?room={local:mp_room} } ] },
            { "title": "Log In", "commands": [ { "call_macro": "mp_login" } ] }
          ] },
          { "disabled_condition": { "require": { "struct": { "local": "MP_CONN" }, "path": "Status" }, "eq": "Connected" },
            "show_condition": { "require": { "local": "MP_MODE" }, "eq": 0 } },
          { "text": "MP Connection Status: {0}", "params": [
            { "struct": { "local": "MP_CONN" }, "path": "Status" }
          ], "show_condition": { "require": { "local": "MP_MODE" }, "ne": 1 } },
          { "text": "MP Server Last Error: {local:MP_LAST_ERROR}", "show_condition": { "require": { "local": "MP_MODE" }, "ne": 1 } },
          { "title": "Debug Info",
            { "text": "Multiplayer Mode: {0}", "params": [
              { "switch": { "local": "MP_MODE" }, "case": {
                "0": "Undecided",
                "1": "Offline, Singleplayer",
                "2": "Multiplayer"
              }
            ] }
          ] }
        ],
        { "#comment": "Debug MP Message: {local:MP_MSG}", "show_condition": { "require": { "local": "MP_MODE" }, "ne": 1 } }
      ],
      { "set_dispatch_dialog": 1 }
    ],
    "mp_login": [
      { "#comment": "try to make the actual connection to the server",
        "set": { "param": "service_auth", "value": { "local": "service_auth" } },
        "set": { "local": "MP_LAST_ERROR", "value": "" },
        "set": { "local": "MP_CONN", "value": { "fn": "create_multiplayer_connection" } },
        "set": { "local": "MP_CONN", "path": "OnError", "value": { "js": "create_async_function": [
          { "set": { "local": "MP_LAST_ERROR", "value": { "struct": { "param": "$args" }, "index": 0 } }
        ] } },
        "set": { "local": "MP_CONN", "path": "OnMessage", "value": { "js": "create_async_function": [
          { "set": { "param": "arg0", "value": { "struct": { "param": "$args" }, "index": 0 } },
          { "call_macro": "mp_on_message", "params": { "msg": { "param": "arg0" } } }
        ] } },
        "set": { "param": "unused", "value": { "struct": { "local": "MP_CONN" }, "function": "Connect", "params": [
          { "static": "server_url" }, { "param": "service_auth" }, { "local": "mp_room" }, { "local": "mp_password" }
        ] } }
      ]
    }
  }
}

```

```

    }},
    {"create_thread":{"commands":[
      {"wait_for":{"struct":{"local":"MP_CONN"}, "path":"Status","eq":"Connected"},
      {"#comment":"once we log in once, we're committed to multiplayer"},
      {"set":{"local":"MP_MODE"}, "value": 2},
      {"set_briefing_dialog":1},
      {"#comment":"First create terminationCommands with no_overwrite, then add an entry for us, and then populate with commands
to clear us from connectedAircraft and terminationCommands when we become stale on the server"},
      {"set_shared_data":"update",
      "path":"terminationCommands",
      "policy":"no_overwrite",
      "value":{"create_struct":{}}
      },
      {"set_shared_data":"update",
      "path":"terminationCommands.{service_auth}",
      "value":{"create_struct":{
        "removeFromConnectedAircraft":{"create_struct":{
          "type":"delete",
          "path":"connectedAircraft.{service_auth}"
        }},
        "removeFromFlightPlans":{"create_struct":{
          "type":"delete",
          "path":"flightPlans.{service_auth}"
        }},
        "removeFromTerminationCommands":{"create_struct":{
          "type":"delete",
          "path":"terminationCommands.{service_auth}"
        }}
      }}}},
      {"#comment":"make sure we have connectedAircraft table. all players must use no_overwrite when ensuring the table exists to
prevent anybody from destroying the table."},
      {"set_shared_data":"update", "path":"connectedAircraft", "policy":"no_overwrite", "value":{"create_struct":{}} },
      {"set_shared_data":"update", "path":"icons", "policy":"no_overwrite", "value":{"fn":"get_mission_icons"} },
      {"set_shared_data":"update", "path":"flightPlans", "policy":"no_overwrite", "value":{"create_struct":{}} },
      {"set_shared_data":"update", "path":"webConfig", "policy":"no_overwrite", "value":{"static":"webConfig"} },
      {"set_shared_data":"update", "path":"gameScores", "policy":"no_overwrite", "value":{"create_struct":{}} },
      {"set_shared_data":"update",
      "path":"connectedAircraft.{service_auth}",
      "value":{"create_struct":{
        "location":{"resolve_location":"$USER"},
        "UserName":{"local":"mp_userName"}
      }}}},
      {"set_shared_data":"update",
      "path":"gameScores.{service_auth}",
      "value":{"create_struct":{
        "UserName":{"local":"mp_userName"},
        "Score": 0
      }}
      },
      {"#comment":"update our location, score and flightplan (if changed) forever"},
      {"while":1,"eq":1,"do":[
        {"sleep":5},
        {"set_shared_data":"update", "path":"connectedAircraft.{service_auth}.location", "value":{"resolve_location":"$USER"} },
        {"set_shared_data":"update", "path":"gameScores.{service_auth}.Score", "value":{"local":"MY_SCORE"} },
        {"if":{"json:stringify":{"local":"$FLIGHTPLAN"}}, "ne":{"param":"FPL"},"then":[
          {"set":{"param":"FPL"},"value":{"json:stringify":{"local":"$FLIGHTPLAN"}}},
          {"set_shared_data":"update", "path":"flightPlans.{service_auth}", "value":{"create_struct":{
            "points":{"local":"$FLIGHTPLAN"}
          }}}
        ]}
      ]}
    ]},
    ],
    "mp_initialize":[
      {"#comment":"setup for multiplayer operations later"},
      {"set":{"local":"MP_LAST_ERROR"},"value":""},
      {"set":{"local":"MP_MODE"},"value":0},
      {"#comment":"MP_MODE 0: undecided, 1: offline, 2:online"},
      {"#comment":"these are for debugging only"},
      {"set":{"local":"MP_MSG"},"value":""},
      {"set":{"local":"mp_room"},"value":""},
      {"set":{"local":"mp_password"},"value":""},
      {"set":{"local":"mp_userName"},"value":{"var":["ATC AIRLINE","string"]}},
      {"#comment":"Create or access a unique ID to identify you on the server irrespective of callsign"},
      {"set":{"local":"service_auth"},"value":{"fn":"create_guid"}},
      {"create_thread":{"commands":[
        {"wait_for":{"local":"MP_MODE"},"ne":0},
        {"call_macro":"mp_begin"}
      ]}
    ],
    "mp_on_message":[
      {"#comment":"param - msg"},
      {"#comment":"handle READ, UPDATE and DELETE operations below"},
      {"set":{"param":"json"},"value":{"json:stringify":{"param":"msg"}}},
      {"switch":{"struct":{"param":"msg"}, "path":"type", "case":{"
        "read":[
          {"set":{"local":"MP_MSG"},"value":"we got an read: {json}"
        }
      ]}
    ]},

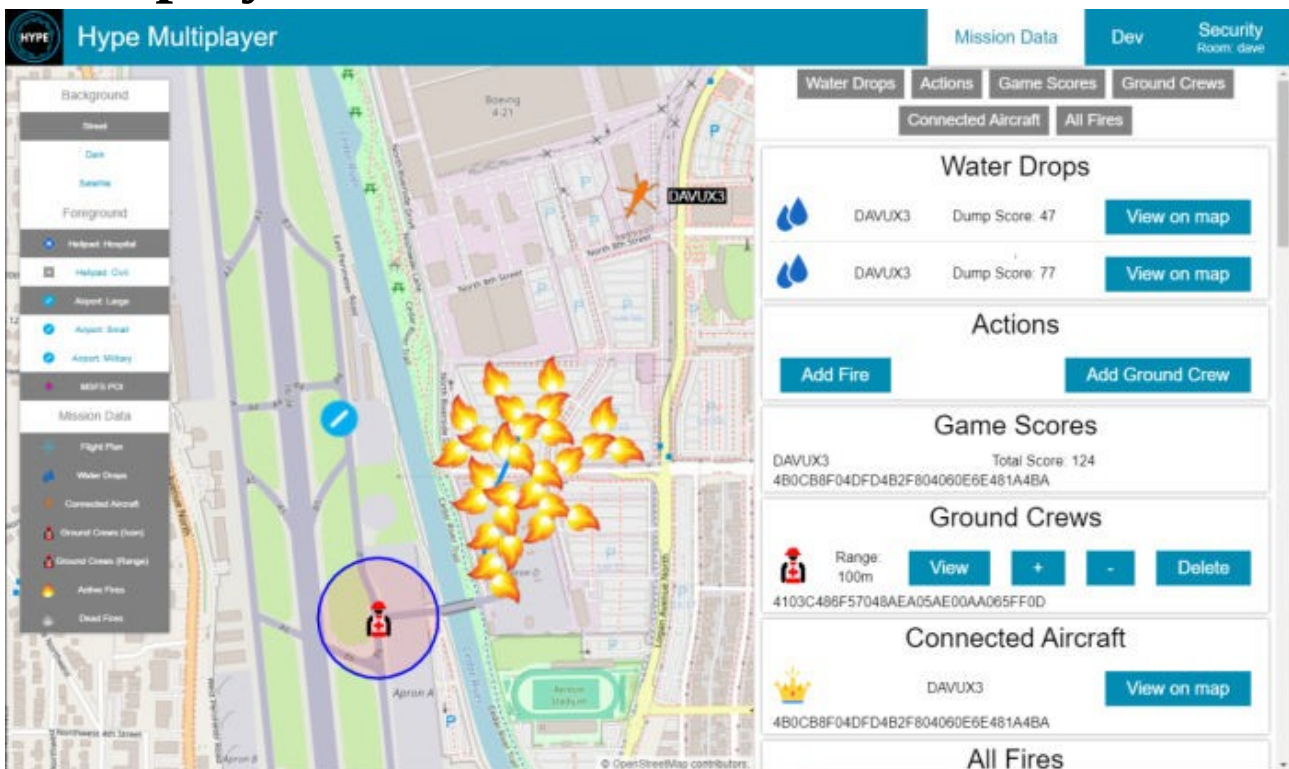
```

```
"update": [
  {"set":{"local":"MP_MSG"},"value": "we got an update: {json}"},
],
"delete": [
  {"set":{"local":"MP_MSG"},"value": "we got an delete: {json}"},
]
}],
"mp_begin":[
  {"#comment":"called once we decided if we are single or multiplayer. MP_MODE 1:offline, 2:online"},
  {"#comment":"offline case, manually run the logic and complete logic"},
],
{"set_objective_title":"Ready to play the game!"},
],
"mp_aborting_mission":[
  {"#comment":"we want to clean up our multiplayer connection if it was created"},
  {"if":{"local":"MP_CONN"},"ne":null,"then":[{"set":{"param":"unused"},"value":{"struct":{"local":"MP_CONN"},"function":"Close","params":[]}]},
]
}],
"objectives": [
  {"title": "Setup required",
  "commands": [
    {"set":{"local":"MY_SCORE"},"value":0},
    {"call_macro":"mp_initialize"},
    {"sleep":"forever"}
  ]
}
],
"icons":{
```

```
"wp_blue": "data:image/png;base64,iVBORw0KGgoAAAANSUHEuGAAAFAAAABQCAAAAH5F5I7AAAAAXNSR0IArs4c6QAAAARnQ1BAACxjwv8YQAAAAJcEhZCwAAADsIAAA7CARu0SoAAA6GURBVGd7ZsJwBRHfOaCf55IbhtvSTg3GcJwGrJqV0IaY7KaJuk0EjUe6EYNJqioWRUVQXBEFFS8QP2UYzRkKeigEIEUEVQDPH7ZmYmnoU3u6emuEadEbaPxc//+9+r+2qpu1q+t1VTVOR18E9GIovo0DIOscGy8d2G/
sCAE1q+TwxFXJx2NpJmQ6TuuCV8PS40ptfE6vVlkZC8SX892ZqjF0N2g7FL4Dbh+SLNChrQxRgIgd0oICQT8gVATDCCIFpXenVaxntLrMnj8t5wCuPR8qHfS4cJm04kMr
Lz6M6HfW2JV2526c/TYCXPAIuAUFdrTSU/9SD1PqN8XA0eu8sj+wUGLrX1onJf1Qe6e3YuYJzJKt+d7LqW155aTJjs9z+195cvd3ux60KtH/
m7ABvPwB5ASTGZLUXq1DsdBLsdPIdk0hbtL5/AI0Kmw8GpN/Dz67MpkU+sqGDRuYmLinHVTe0NF3VyhRBPkMjy1dugRqVIA/
7qqUinSoQDhVzSprJ7U5NoQM0H5xwH0Qx2bnj1pb7z0hnPmzAG74XmMakBRrdkewXBPnL9Kq64ke/pMmpg07cCB/
XreAYHdyTRNCy56FAhymLQ9CnEYiZovDZkgv9glj1PViJ2LSp0JcyGdLlSVJLj6uqq8Q1e3+InwFDubzjHpkEjt61N8jUtqa+vFPwax0Py7BPFNgby9ZsLiLaqfKcF6V0
Kd+uzbU3f0HnprwF8LcJcTz3NBo4XMoILjsZnfD3nvoI8g+0LmJjkvEXoCKW03kpcVug8I0vUhesqJDU0w7xg00dSwsDa73K0sorQ2UkyYna1f477e9RevDrWRL/
Qe2JoI1JINMLkN3k/096j1841kgcbsa2F0kKfCzKGr3IQvhiFwNhw5/YEJ708VLIwRYagBvLkVx8Wym7pgtLSZk0hJy+NEVz/0jy4UUBL/
RR0wJQhhy0frr153wyc0Xmm0HLhyjif+OdReggANwyPeXbjnmtFzFASBYSbxJHfhdU8dv2Xc/
wZelTbQzNybScpbFqS0H8QeLq0m6nuvUvUvAcxpkVRNRS0PKUmK20fn54/CHI2YpXs2cF22ZLS6f0u/W/pTgSt1cNTWYrM7jBbgLdu9z/
XqGbbq9aUxIFa8wn33yebj1gy3EH3ihxORDPVRNns1qBywCxfFM6/9+M4kt/tcuWZsectXh4/KjrMuuf824wG88acLxLtUQ91XTRBkAwEAbuXrjZvVx/
ap032W5pFAnFdn7ty5IPzi1ZFURqJhF3HVxtddYt5K1badDvFUpaiHrGyToZsmZzng/ExwcpEsY9P33yVBuYz3kDa6wMgCw7g/
LS6tqXscackkZpAr0kjuXwDKT88Q9aFa+S05cTFhKgcwtF9BpFukAYhh+2NSixTokzLIVbChRTLD9FhHYiBborHJxpT0bVFKLZxIuG7LsY9e0QC4PhaWkUjtjHRC17/
vknnV0CnHuAGiVmNlBYXZy0k+HQkuTaFiagdeikZUG052h0K9TSh0h1TXRRcmQfR3AEIYCR658CDow+TagVQ2FGzVdGxkTyhwnbiGTAaVetIGxT8K15SmAT9nkECUEdAK5
QL5cYHwJwIgaFWLXApLhgJCo9JUC5y6tGq/S6FDoerJE81snkARML0X30m1JVBi4DkznSHK1YphZY/IbapFixZS03PWJ/AihLks6ChwvIjDlt/
809T6BEWQ0GL9uueQ0Lw2q0nX7B6WtBxxd4XY0khyz446BvYv/
1uyvAvEerfDcct2QZS1Zkz3S3J8McpjMxH3IfnEpGXP8ddMIJIp0iWYtjuf80gpTwpqJj305ZdfaN1w1P4D/
gKOTJj13FR7pk47jjrofCAU18dtewm+DrAd8KwWLNjrR1hCrh7967JzBHv1QSL2A2jAeLmg0m061ubtkg0ZVYebcmJ3b/Zcp37C83TU/
bkWfL4UPCaKBXqj+5EuxohoJsch4vLZkz6EvnW2kLm0YRcz9jmuCv067yXoecchx1iLKLbsXu0T74v8QzPmz6c061I+VGTf4z29Ib35tVrKfPa7CDNzY0x/
mu3Wun69rt08nj0QNI20XPpPDZwvT7jxy3Y06ZLg7He8bu1RvSrlp/IdpjpmlULWfDnt6e9uc437drRMDAPS/cvrQv/
N077dbjNu10Mz8jK0YogJ5YfuuOHLAhoNm15a2PR1Vm785Luno+KChYQuS8TnUxLiA8eGseVmlBfy2HHTUmdxaQvxAd4/
j9X0nM1rfdLzmaAdu4e+g2g3EidiyWbhtYEiAUS93i4r9R8BF5V2Lhgx+6PuekJZn3pBKtCMYl+6reYmFpmzdW65VKkt/d2BmG986dHfPPRB+Y09w//
aBacAPDbhbugBap290IKJuf6tU40Q7VABUVSH3P9uCh0mYnKbqLW0FXLRhaSEfBe4leHabdMbuTmY0wFzQphjYcVDP2UspUQIKHf35QFwFMcVsd1UUKYESAxmftwKgcCL1W
Q0hSDPwyj0SGT0T6ESzT33u74qLawsA8HjcsjaAqSQBMXwqQ+Q6mBYXbc0ct5b4KynwQCBVnILcBSKouUwFybP4/
T0pYJARSxHdC4vDdaFThlyYxEMHRqhzmDm3IQghUSjIbgHepFYau9wFMApa2YCFX/zq46APc/b8J1szzJULv+10v6BP4PG/gpkAhe/+GEZuLqEPdCRajhXlYv/
jLTzVtZr17y50CAVPpRS0dKLB+WAnGnd/uBwDDKz/+BBva0b8vL3wdXpddSRmZglvEnVbrnG1SH5UKynaFcd16/3ZETqsr04j6CknsnzEm0rPzSdqIB2DnQe/
R9LX0nM1rfdLzmaAdu4e+g2g3EidiyWbhtYEiAUS93i4r9R8BF5V2Lhgx+6PuekJZn3pBKtCMYl+6reYmFpmzdW65VKkt/d2BmG986dHfPPRB+Y09w//
eYpTnYtXrM2Xq040b10PjJmKcrJoZTY2vNMLRX0ZuoD06LDMwRRLXALIEbTcJsgTbDgN0mVne1YafATAUHMDk7Lek8tIFwbyW0Gy+vr/
q1j5HF6GekXUJZV5YrQxgSmbACgWdTBChcThz4+mM7f9yNGYLMVXHfHvS81w47YyhqV01X23s9U/EF0g2GT2MM7VvhyLYLFzFfIoqNIBf4c12P0+/
Q0o9j0o0DatwQSD2mLUKTgnBs6B7T4Z1GIXaYg6xrJ7N8FLaTAvb+2wmIhzHM7eWf0+f7L1z4LjFRJq0bt1oszJAXadSyMcwG5JG3M5/cgbLvuYtF66Fv/
ziU+ksrj20558Vq1bg0I4YpCtLrXekjlnxLNT89FknVmo0qYG08ZnASbV540LbqXk44LmKhmwP0oj8r8k8yB2nwwQDu0YbYhLgKGTp5Fash05tdk2bGkF/
CLmTIksx88Dw7e17Gc3qfUptGmTf/1DX7nr06Igd3kwbvbo8dmJu/+4Y2YtRXLkAdwbGwa/
Sxdv45IGPRBxLBJwJNLZheKcEY2NtzLNZYmsDg1NTmxLz8Hf7XbcXsQa/
8KMc7a+zdhGHYpC3h6E644cd4wtLKzKjcrNnkzizY2EpyAhaJmdAXaa02LX5V06LBUGADERXefP01/+1D1riA29qbcghyL0LKGdenGelARjhFE/
AFYrYBpqEm0kjskFQXvos1Jqu1Vt0lntcL3bkXXhIa69RHF6Yc7+w/sSMV/G9jAmpzWqmsZIU0FMynamecGQAGTbCUyYrZMQ6czPMA/
7CT7Yp25rUBD12XrE6wGr40D0sAsrZAHANXF9DC0yGYICZS0Gcu2T7P8BpGCAVZLrLw8Bmcomp/MCXff4FP691N1AveD6P0I8cPbd01/
CjXwHtAdTtd1J0u16BADAgoiDusShaS3F64Ruc9C1YAjlqhb9Twt6T3X8SR9t6jJw8PhJ12XKDNsmUnhRvZLsf47ImxTvo2wnoJA2VEnS7XgVdQat5iue2AbYLXZLZM+
mHnGNSCKiKSchwMPTU+2gc0HCbWlaWfa9+cYWE38bwhzJLzBZEGHwHfai5mgxKDPaAWLQAOzMOFLpbU2VH6CSVG5LRq3qSouzN0LMD+/
00Ny5cWw4hej1QnHXRr1j1dh71A0ztJon1bcY850LmJnbi0FLbmeXJLF7zMQk2dz28A1BXdbFyghUu2BV3yCpLjLs9HUvMv0IAip6JgaMG5f5jXcJubFydzmjsyfy
X1ZGMT5Y0LI7oZSAVwFwXm5b4I0DKV5/d2X79m0dEwPL7l0Z5hSMHJmYGAAXGHR2pHL0Z/BpnVHLz654/
KFE6FUk2T+wPawVMQ55TWkQBEGPQ2+eFEZ+5Y0ayF1hw25YA1v00bRwAdXrUyIWA+UDYNS7pNCJ822s1brwP7zbVdsd2jJwIcJ1+IRY8UCkqTCABw4eVbVxx08EzKb4y5
2FwEjFKRgn72UUN7QhW/
vV5co+7m5dXmXyDLVfV9yRzk+tBxtgS1v6E50Vlq70LE8TE4182wk6vH2NTQum1Yrc7RpwgTysvQjgK2oYGi+AE1j729Xk0Y4uAm+SxGvMLtBQFLSUFKCa89kNYW69p6cGU
Jj/wi5ZAwDxc+rMhwGf/LX4RU5MyXNia0W53Aq3Yf6PwGLK5l7f7bVgna17hebdwFeRht01YQJ5AW3/
ACQknJXDYNTgh+kAUSA9Syrsv3hV67JkygUYPA2kph2AJhDRICqbuJ7eyLXQ0kWF6ApkJRxdKytiwo0gpa0nBE5wPVW4mScd0+nqYLtHRg4u9n4XBJEdRweI0g8VKYaLnZD
RrWzf3KMuXo61TR2AKtaQdtY6oJDDR4gbU1b0ZLr2nqyemJCYrMhw6AcB/AID93Zd+9+L6AAAAAE LFTkSuQmC",
"hi60_icon":
" data:image/png;base64,iVBORw0KGgoAAAANSUHEuGAAADGAAAA4CAYAAAHfGQuIAAAAAAXNSR0IArs4c6QAAAARnQ1BAACxjwv8YQAAAAJcEhZCwAAADsIAAA7DAdcvq
GQAAAd+SURBVfH7Z1pbVFFMdpdyhLBUG2K0bIYlmuJSLCOUY06AYIDHIZhUrIUIGLjF7Tez8AEhV1BLgZ2Bo0KNGPwDQoAYRYmgAdGQksomINAFuvFv/
39mT9t99713371viyb8kpnZr0zc2b0zJyZK25kmVACA6Qj0eMu7T4u/
ZH8Vefa6MALFJOUTB0S0SYkU0YRp4J04jp6Y06FdpaiMm7Hx06XeyXfVphhn7B5BiXQzXfRi1SYMoavSAqI66VQTLsWMGxdlR1EP2kkwWThmFXkMn9+HtkJ6cnyQfmX
eA14+XHGGZC6x0KwQKzPVLRCaHlnag9Z31kwr6Me8atxqX0YR0BseqPQeIexMwV18Hngz/
yNk5tNTX1aw2pchsje0C0XtQDjmi0AprTBhCT0FomCGpN2Eiz1b0Gf3WtSgC87JND0AV3YtG053qJMN0wqM6qGeoydZyfT2CB0VAnsy1hBEY9qhxJds4r70AZmno+GEj9X9
NjHSHrjXHPeChEfcFain0Q0PvqMCLLEXQAMj0Ce0NE0hpv+W6GTGpMX3AvJZTgnaif6jSqm1mwJHZZRNj1Hcx70afU1SABFU1TSKrupgcG7AQLoP/
6FR0fJKRyMI610b8yG7LJNY0im00sLTrGTq0LUTW0VHIhszk+XFUR5xj14wNwMIr0PoeMzMOUtdbYnq5UW3AF5DYI/
uA1PycwAKF6rS52Jd+8GzCpSYLPKGG6GJaF190esev0dXmAKx9i3aENPjYCGFPV5fTRpH8awLjF2Xb0K31jN8T8hc/xjDoz3li6k65jT/
Or04JgVg09XbXyU444c2z9pMhC0sBAmDx2aZg1zXgva5jHmuJ5JyMddJqTbvjW8I8aaawpsrYtL0LiClh9kSE4853L7m4Tfy72aiwP0152WEF8LF4BDEGYE4K3XF68Tfm7L
EkgpQbTf9GfT/vyTCH/TSX08fmgYlCg10yW3/BtR6fBkmb6WtoX5B7YDYLKXkLgeRr40pAH62ysiBp6LsePhmmCF8N1RrhXw2/KmJ8r6Zak1N8ke/
SQUC4pcDBkMw2U92Ab61e0KfXgC1m+Z9vKPaILeag4dhDCK+cMVBX5aV2zdP8Rvc3RKY2r5JMu2brMfSfLX0TtsqZRHj/
gag7FqvCYshImUzq05L0sJonzJ5Q7EY60CYpYGI6BcwQ1LFY0V7Knt4q9+I9LhAbp/pMu07eBrh/XXM6uW0FsetjTfGFEF6j/
aBSPhltxrSj563me8TUPDcGuvtgS75q0z+KpDcPp70E3biqda+Tww/
ufBEXBwerFeLNYLLANBzZdzko7vZjZGwceF9P0kEFG6wvzF0mHgLmbgfmbly95QmbT+0nwYq4WpXv+ZVnj84igrbrbdTURL6DTGzTz+WwsquKCr6nChXxu0DKFQdE7mF
8jRRBkdudrkpYXsnPrFP1Dv1H5HsvrBhoyG89r4b4rY99j14UBZpM/wds1Vshk5qunAAKc7fk/DeINJhLEcqqz/abXkz4H07BRTsmS/nzPNYCK/
F4sb3wuuADU0b03XcrT2CmHw0HcvTAVU13+20K0ZAbx0dggC1F5B1GU3LkIDYVfRgnhZukUUKayfK6i6FLz/
```

```
x2PH2odNJAONfhhpNroRwmPlLwgjcqGAt+qc11R6nsmz9SRuD7Z5DcDKljfiIkS0ELmtM9UGhHBwsc4PMLBsLS96aLYv043BohzjmljbhgMB/  
QVBuHrI+glgeU0Ikw0GLvgcwYNEb+ofcp8QEYvGv4H6dSg6pUpCw7KmYn19HuPEIobFV0vKr1N3rcZ2TPFLnQooUtVTKV7GUI+bUN1ankksqFeQK68dCwq40k0GyFKTbzZ90  
EyCLzi2xttbVcirHn88xCMlfnGnte0789DBHhPehvKkoaVgpm2BawQ7iwXT/KwmatF39M0GLwWnIzVhReY/  
qSgP89m5V8iw65cVsw5dzJa5rLkCv12QZkrxy4f56AcLtx9Mqa1ewX2uL/J2VJRnVF6T1um3Kx6wZf5XcuX+hrNKvi0Bzy9BtytwiEmcgrCwahRjNs0X57paC1ZN/AL/  
QqSD8ht4Rf+YNgVxfxyGbcnIjL2W2G25H0yF0GhSRRpB/DzLkXLJpLochvBzhHXDn0cw7HMw0Jh4G0xMdRo+LXuFEbT2g0ajnvcutEiiokoFIs/  
N9mw180LQXovJtotLJG7orZ6M6KfyhcXKGi87B0QE5ywxX00kSnCs7T9WF3FgpoNxMBMlSjlyC/  
AhRypF0EhqB22V0SauqGLS5v6zTqW0dCLiAdZBJrwjzyNgj6eUdCp4prLNLXak6hrLc6b8YiedCtbu0CLfmzhRq1yqSdUq0gan/Z/  
8Fr74U9n09mF1FnTbw32Rkgw5D3FzZtv+v6dovX5aeWa4cCG7uv0awLzJ093g9T48L5D/P/  
4VR1h00pUuwrEQe98iUks45GFxgyKACddmUctKp4HFs9sFVs7pw3dMk5Vriv0T33FPkGEKeSq6Q0CL/Ap3jTWX/GRgjAAAAAELFTkSuQmCC"  
}  
}
```

Multiplayer: Web Client



Der Web-Client ist die Website, die zur Verwaltung des Multiplayer-Raums verwendet wird. Der Web-Client liest die `webConfig`-Daten aus den gemeinsam genutzten Daten und erstellt ein Spiel auf der Grundlage der Konfiguration.

Sie können Datenlisten anzeigen, auf Schaltflächen klicken und einen Dialog mit verschiedenen Listen-/Text-/Schiebereglern öffnen. Sie können auch Symbole und Text auf der Karte sowie Linien und Bereichsringe anzeigen lassen.

list (WebConfig)

Das Listen-Widget (`list`) erstellt einen Eintrag für jeden Schlüssel des Quellobjekts. Jeder Eintrag kann mehrere Zeilen haben, und jede Zeile besteht aus Elementen, des folgenden Typs:

```
text  
icon  
button
```

map_point (WebConfig)

`map_point` erzeugt ein Element für den angegebenen Ort. Das Element kann sein

1. `range` Ring. Wird immer verwendet, wenn ein Bereich (`range`) angegeben ist (auch wenn er Null ist). Nicht kompatibel mit Symbol oder Text (sie werden ignoriert)
2. `icon`. Kann ein beliebiger URI einschließlich einer Daten-URI sein.
3. `text` der neben dem Symbol angezeigt wird (das Symbol selbst ist nicht erforderlich).

siehe Kontur (`stroke`) und Füllung (`fill`) zu Stil.

map_line (WebConfig)

`map_line` erzeugt eine Linienkette aus einem gegebenen Array von Punkten.

Siehe `stroke` zur Darstellung.

event (WebConfig)

`event` kann verwendet werden, um eine Befehlsliste auszuführen, wenn Tasten innerhalb einer bestimmten Quelle hinzugefügt/entfernt/aktualisiert werden. Sie könnten damit einen Ton abspielen, einen Dialog aktivieren oder sogar Logik ausführen oder lokale Elemente aktualisieren. Die meiste Logik sollte sich innerhalb des Flugzeugs befinden, aber dieses Werkzeug ist verfügbar.

Dialog widgets

Sie können `show_dialog` verwenden, um eine Liste von Widgets in einem modalen Modus anzuzeigen. Normalerweise wird dies durch das Anklicken einer Schaltfläche ausgelöst.

```
text
icon
button
textbox
listbox
textarea
slider
```

WEB COMMANDS

Die Web-Befehle sind eine begrenzte Teilmenge des Missionsbefehlssatzes.

Web commands:

```
set_map_center
show_dialog
close_dialog
play_sound
```

Verfügbare Befehle aus dem normalen Missionsbefehlssatz:

```
set (param and local only)
sleep
if
while
switch
try
for_each
modify_array
```

```
create_thread
throw_error
debug_write
break
continue
set_shared_data
fetch
```

WEB QUERY

Die Web-Abfragebefehle sind eine begrenzte Teilmenge des Befehlssatzes für Missionsabfragen.

```
create_array
create_struct
struct
string:split
string:join
json:stringify
json:parse
json:copy
object:keys
create_number
param
has_param
static
has_static
local
has_local
rand
add
add360
subtract
multiply
right_shift
left_shift
remainder
xor
exponent
divide
round
toFixed
floor
Math.* (same)
ceil
abs
clamp
scale
and
or
not
if
switch
require
compare
text
typeof
```

```
isNaN
parseInt
parseFloat
convert (weight & length units)
no_resolve
fn.create_guid
fn.create_date
fn.get_time_string
fn.has_selected_poi
fn.selected_poi_info
fn.selected_poi_location
fn.is_dialog_open
```

Unterstützung für mehrere Sprachen

Um mehrere Sprachen in Ihrer Mission zu unterstützen, gehen Sie wie folgt vor:

1. Definieren Sie eine `data.translation`-Tabelle für jede Sprache, die Sie über die Standardsprache hinaus unterstützen möchten
2. Füllen Sie die Tabelle `data.translation.Language` mit den Schlüsseln, die die Standard-Sprachzeichenfolgen sind, und den Werten, die die für die jeweilige Sprache zu verwendende Zeichenfolge sind.

`local:$MISSION_LANGUAGE` enthält den Namen der aktuell ausgewählten Sprache. Ist sie null oder undefiniert, wird die Standardsprache verwendet, ohne dass versucht wird, in eine andere Sprache zu wechseln.

Tasten, die in der Zielsprache nicht gefunden werden, werden in der Standardsprache wiedergegeben.

Beispiel:

```
"data":{
  "translation": {
    "French": {
      "hello world": "Bonjour le monde",
      "hello world {0}": "Bonjour le monde ({0})"
    },
    "German": {
      "hello world": "Hallo Welt",
      "hello world {0}": "Hallo Welt ({0})"
    }
  }
},
...
{"set_message": {"text":"hello world"}},
{"set_objective_title": "hello world"},
```


Translation Test Programm

```
{
  "title": "Translation Test Program",
  "data": {
    "translation": {
      "French": {
        "hello world": "Bonjour le monde",
        "hello world {0}": "Bonjour le monde ({0})"
      },
      "German": {
        "hello world": "Hallo Welt",
        "hello world {0}": "Hallo Welt ({0})"
      }
    }
  },
  "briefing": [
    {"text": "Language selection:"},
    {"buttonbar": [
      {
        "title": "English (default)",
        "commands": [ {"set": {"local": "$MISSION_LANGUAGE"}, "value": null} ],
        "select_condition": {"require": {"local": "$MISSION_LANGUAGE"}, "eq": null}
      },
      {
        "title": "French",
        "commands": [ {"set": {"local": "$MISSION_LANGUAGE"}, "value": "French"} ],
        "select_condition": {"require": {"local": "$MISSION_LANGUAGE"}, "eq": "French"}
      },
      {
        "title": "German",
        "commands": [ {"set": {"local": "$MISSION_LANGUAGE"}, "value": "German"} ],
        "select_condition": {"require": {"local": "$MISSION_LANGUAGE"}, "eq": "German"}
      }
    ]},
    {"text": "hello world"},
    {"text": {"text": "hello world {0}", "params": [ 99 ]}}
  ],
  "objectives": [
    {
      "title": "Done",
      "commands": [
        {
          "#comment": "adding a loop here only because it won't re-evaluate ever otherwise",
          "while": 1, "eq": 1, "do": [
            {"set_message": {"text": "hello world"}},
            {"set_objective_title": "hello world"},
            {"sleep": 1}
          ]
        },
        {"sleep": "forever"}
      ]
    }
  ]
}
```

Server (Remote) Missionen

Die Missionen können stattdessen auf dem Server ausgeführt werden und Befehle an den Client senden.

Wenn der Benutzer die Mission auswählt, wird Ihr Server kontaktiert und zu diesem Zeitpunkt können Sie das Missionssystem auf unbestimmte Zeit verwalten, bis der Benutzer manuell eine andere Mission auswählt.

```
{
  "title": "Connect to Server",
  "aircraft": ["H145"],
  "api_version": 0.1,
  "url": "localhost:9998"
}
```

Ein Missionsserver kann dynamisch Missionsbeschreibungen erstellen und anwenden sowie andere Befehle senden und den Status beobachten. Der Server ist im Grunde nur ein Websocket-Server, der darauf wartet, dass der Simulator eine Verbindung herstellt und dann ein JSON-RPC-Protokoll verwendet.

Ein sehr einfaches Missionsserver-Beispiel in node.js ist im Tools-Ordner enthalten.

Vom Luftfahrzeug an den Server gesendete Befehle

| Befehl | Beschreibung |
|---|--|
| <code>{"control_msg": "hello"}</code> | Nach dem Herstellen der Verbindung meldet der H145, dass er bereit ist, eine Mission zu senden. |
| <code>{"control_msg": "canceled_by_user"}</code> | Der H145 macht Sie darauf aufmerksam, dass der Benutzer eine andere Mission ausgewählt hat und Sie nicht mehr aktiv sind. Die Verbindung wird nach dieser Meldung getrennt |
| <code>{"remote_notify": "my_message_name", "params": [0, 99, 3]}</code> | Bei Verwendung des Befehls <code>remote_notify</code> werden Ereignisse ausgelöst. |

Vom Server an das Luftfahrzeug gesendete Befehle

| Befehl | Beschreibung |
|---|---|
| <code>{"load_mission": MISSION_DESCRIPTOR}</code> | Nach dem Herstellen der Verbindung meldet der H145, dass er bereit ist, eine Mission zu senden. |
| <code>{"exec_commands": COMMANDLIST}</code> | Der H145 macht Sie darauf aufmerksam, dass der Benutzer eine andere Mission ausgewählt hat und Sie nicht mehr aktiv sind. Die Verbindung wird nach dieser Meldung getrennt. |

`MISSION_DESCRIPTOR` ist einfach das (bereits geparste) JSON, das normalerweise eine flache Dateimission sein würde.

`exec_commands` ermöglicht es Ihnen als Missionsserver, jederzeit mit Logik einzugreifen.

API Reference - COMMAND

Alle Befehle sind im Folgenden aufgeführt.

#comment

`#comment` wird verwendet, um menschenlesbare Informationen in Befehlslisten einzufügen. Er hat keine Auswirkungen und wird sofort ausgeführt.

Beispiel:

```
{"#comment": "This section of code is very delicate"},
{"#comment": [
  "This section of code is very delicate",
  "This section of code is very delicate",
  "This section of code is very delicate"
]},
```

sleep

`sleep` wird verwendet, um zu warten oder die Ausführung für einige Zeit zu verzögern.

Beispiel:

```
{"sleep": QUERY},
{"sleep": 0.25},
{"sleep": 1},
{"sleep": {"rand": [0, 60]}},
{"sleep": {"var": ["L:MY_SLEEP_TIME", "number"]}},
{"sleep": "forever"},
```

wait_for

`wait_for` geht erst dann zum nächsten Befehl über, wenn der Vergleich zwischen QUERY_1 und QUERY_2 zufriedenstellend ist. Format:

```
{"wait_for": QUERY_1, "eq": QUERY_2},
```

Dabei ist eq der Operator, der eine der folgenden Eigenschaften haben kann

| Operator | Funktion |
|----------|---------------------|
| eq | gleich |
| ne | Nicht gleich |
| lt | Kleiner als |
| lte | Kleiner oder gleich |
| gt | größer |
| gte | Größer oder gleich |

Beispiel:

```
{"wait_for": {"var":["L:MY_TEST_VAR", "number"]}, "eq": 1},
```

if

`if` ermöglicht es, eine Bedingung (einmalig) zu prüfen und dann den `then`-Zweig von Befehlen oder optional den `else`-Zweig von Befehlen abzuarbeiten. Sobald der ausgewählte Zweig ausgeführt wurde, wird die Verarbeitung mit dem nächsten Befehl nach `if` fortgesetzt.

Format:

```
{"if": QUERY_1, "eq": QUERY_2, "then": COMMANDLIST},  
{"if": QUERY_1, "eq": QUERY_2, "then": COMMANDLIST, "else": COMMANDLIST},
```

Dabei ist `eq` ein Operator und verwendet dieselbe Liste wie `wait_for`.

Beispiel:

```
{"if": 1, "eq": 1, "then": [  
  {"set_message":{"text":"1 is always equal to 1}}  
], "else": [  
  {"set_message":{"text":"this never executes, since 1 always equals 1"}}  
]},
```

while

`while` ermöglicht die Ausführung einer `do` COMMANDLIST, bis eine Bedingung erfüllt ist.

Format:

```
{"while": QUERY, "do": COMMANDLIST}
```

Beispiel:

```
{"while": {"var":["L:MY_TEST_VAR","number"]}, "gt": 1, "do": [  
  {"set_message":{"text":"this message runs over and over while L:TEST_VAR is greater than 1"}}  
]},  
{"set_message":{"text":"this message runs once, after L:MY_TEST_VAR becomes less than one"}},
```

for_each

`for_each` wird verwendet, um eine `do` COMMANDLIST für jedes Element in einem Array aufzurufen. Die Parameter `$index` und `$item` werden für jede Iteration definiert.

Format:

```
{"for_each": QUERY, "do": COMMANDLIST},
```

Beispiel:

```
{"for_each": {"create_array": 4}, "do": [
  {"set_message": {"text": "my array item: idx={0} item={1}", "params": [{"param": "$index"}, {"param": "$item"}]}},
  {"sleep": 5}
]},
-----
{"set": {"param": "my_array"}, "value": [1, 2, 3, 4]},
{"set": {"param": "my_result_array"}, "value": []},
{"for_each": {"param": "my_array"}, "do": [
  {"if": {"param": "$index", "eq": 1, "then": [
    {"continue": 1}
  ]},
  {"if": {"param": "$index", "eq": 3, "then": [
    {"break": 1}
  ]},
  {"modify_array": {"param": "my_result_array"}, "append": {"param": "$index"}}
]},
{"set_message": {"text": "ret={0}", "params": [
  {"json: stringify": {"param": "my_result_array"}}
]}},
]}
```

try

`try` und `catch` können verwendet werden, um einen Fehler abzufangen, der sonst zu einer Meldung an den Benutzer führen würde. `$ERROR` wird mit dem Fehlerergebnis definiert.

Beispiel:

```
{"try": [
  {"set": {"object": ""}},
], "catch": [
  {"set_message": "oops! {$ERROR}"}
]}
```

switch

`switch` wird verwendet, um aus einer Reihe bekannter Ergebnisse auszuwählen (jeweils eine `COMMANDLIST`).

Beispiel:

```
{"switch": 2, "case": {
  "0": [ {"set_message": "You selected 0"} ],
  "1": [ {"set_message": "You selected 1"} ],
  "2": [ {"set_message": "You selected 2"} ],
  "default": [ {"set_message": "You selected another number"} ]
}}
```

`set` ermöglicht das Setzen von Variablen im MSFS und im Missionssystem und auf Missionsobjekten. Sie können den MSFS-Ereignissen ein `K:` voranstellen, die [Liste finden Sie hier](#)

Beispiel:

```
{"set": {"object": "my_object", "var": "MODE"}, "value": QUERY}
{"set": {"var": ["L:TEST", "number"]}, "value": QUERY}
{"set": {"table": "my_table", "key": {"text": "blah{0}", "params": [99]}}, "value":
QUERY}
{"set": {"local": "my_local"}, "value": QUERY}
{"set": {"param": "my_param"}, "value": QUERY}
{"set": {"global": "my_global"}, "value": QUERY}
```

trigger

`trigger` ist eine Abkürzung, die dazu dient, `H:` und `K:` Ereignisse an die Simulation zu senden. Sie können alle HPG-SDK-Ereignisse an das Flugzeug und alle anwendbaren [Sim-Ereignisse](#) senden.

Beispiel:

```
{"trigger": "H:MY_EVENT"}
{"trigger": ["H:EVENT_1", "H:EVENT_2"]}
```

call_macro

`call_macro` ruft synchron ein Makro nach `Namen` auf. Makros können in der Mission definiert werden oder sind als "Systemmakros" in das Produkt integriert.

Makros, die den `Return`-Befehl verwenden, haben ihr Ergebnis über den `$RET`-Parameter verfügbar, nachdem der Befehl abgeschlossen ist. Sie können `$RET` in einen anderen Parameternamen ändern, indem Sie `result` verwenden.

Beispiel:

```
{"call_macro": "do_it_now"}
{"call_macro": "my_calc", "params": {"num1": 2, "num2": 4}}
{"call_macro": "my_calc", "params": {"num1": 2, "num2": 4}, "result": "my_result"}
```

return

`return` wird verwendet, um `$RET` auf den aufrufenden Kontext zu setzen, wenn die Funktion zurückkehrt.

`return` beendet auch die Verarbeitung weiterer Befehle des Makros (mit Ausnahme von Threads, die weiterlaufen).

Beispiel:

```
{"return": QUERY}
{"return": {"param": "my_ret"}}
{"return": "ERROR"}
```

break

`break` wird verwendet, um aus einer Schleife auszubrechen (siehe `for_each`). Nach `break` werden keine weiteren Iterationen der Schleife ausgeführt.

continue

`continue` wird verwendet, um aus einer einzelnen Schleifen-Iteration (siehe `for_each`) auszusteigen, jedoch mit der nächsten Iteration fortzufahren.

private_macros

`private_macros` ermöglicht es Ihnen, eine Liste von Makros zu erstellen, die nur innerhalb dieses Bereichs sichtbar ist.

Beispiel:

```
{"private_macros":{
  "my_macro_name": [
    {"#comment": "macro commands here"}
  ]
}}
```

create_thread

`create_thread` ermöglicht die asynchrone Ausführung von Code (einer `COMMANDLIST`).

- `commands`: wird benötigt.
- `interval`: optional. Default sind 100ms

Beispiel:

```
{"create_thread": {"commands": [
  {"sleep": 100},
  {"set_message": {"text": "this runs 100 seconds later!}}
]}}
{"set_message": {"text": "this runs instantly and the next command continues}}
```

create_event_handler

`create_event_handler` ermöglicht es Ihnen, auf MSFS H:Events zu reagieren.

Beispiel:

```
{"create_event_handler": "BAMBI_BUCKET_DUMPED", "commands": [
  {"set_message":{"text":"bambi dumped!"}}
]}
```

throw_error

`throw_error` ermöglicht es Ihnen, einen eigenen Fehler zu erstellen.

Beispiel:

```
{"try":[
  {"throw_error":"my custom error message"}
], "catch":[
  {"set_message":"oops! {$ERROR}"}
]},
```

modify_array

`modify_array` ermöglicht einige gängige Array-Operationen, wie das Voranstellen oder Anhängen von Elementen oder das Entfernen eines Elements an einem Index.

Beispiel:

```
{"modify_array":{"local": "my_array"}, "append": QUERY}
{"modify_array":{"local": "my_array"}, "prepend": QUERY}
{"modify_array":{"local": "my_array"}, "removeIndex": QUERY}
```

reload_mission

`reload_mission` ermöglicht es, die Mission zurückzusetzen, ohne die lokalen Variablen zu löschen.

Beispiel:

```
{"reload_mission": 1}
```


load_mission

`load_mission` ermöglicht den Aufruf einer andern Mission (die aktuelle Mission wird beendet). Lokale werden nicht geräumt.

Beispiel:

```
{"load_mission":"other_mission_id"}
```

create_object

`create_object` instanziiert ein neues KI-Objekt in der Welt. Es wird durch seinen Namen bezeichnet. Objektnamen müssen eindeutig sein, und nachfolgende Aufrufe von `create_object` mit demselben Namen werden fehlschlagen. Verwende `destroy_object`, um ein Objekt zu entfernen, wenn du mit ihm fertig bist.

- `name`: Dies ist der Name, der verwendet wird, um das Objekt in nachfolgenden Aufrufen wie, `set`, `drive_object` and `destroy_object`.
- `title`: Dies ist der Titel aus der `aircraft.cfg/sim.cfg` im MSFS, der das Objekt eindeutig identifiziert.
- `fallback_title`: Sollte `title` nicht vorhanden sein, wird automatisch `fallback_title` verwendet.
- `location`: dies ist die `LOCATIONREF`, an der das Objekt erstellt werden soll.
- `is_flight_object`: 1 oder 0, je nachdem, ob es sich um ein Objekt handelt, das fliegen soll oder nicht.
- `is_ground_object`: 1 oder 0, je nachdem, ob es sich um ein `aircraft.cfg`-Objekt oder ein `sim.cfg`-Objekt handelt.
- `is_static_object`: 1 oder 0, je nachdem, ob es sich um ein statisches `simobject` handelt

Beispiel:

```
{"create_object":{  
  "name": "my_object",  
  "title": "HPG Airbus H145 Ambulance",  
  "location": "$USER"  
}}
```

destroy_object

`destroy_object` deallokiert ein Objekt und wartet darauf, dass es zerstört wird. Danach kann der Objektname wieder verwendet werden (`create_object` mit demselben Namen).

Beispiel:

```
{"destroy_object": QUERY}
{"destroy_object": "my_object"}
```

track_object

`track_object` fügt der Karte ein Symbol hinzu, das dem spezifischen Objekt folgt. `track_object` verwendet einen Thread, um seine Arbeit zu erledigen, und kehrt sofort zurück.

`icon` kann sein:

1. data-uri für ein 44x44 PNG-Bild
2. eine Zeichenkette, die auf die `Icon`-Tabelle in der Mission verweist, die (1) enthält
3. eine Zeichenfolge, die auf ein bekanntes Symbol verweist (siehe Tabelle unten)

Bekannte Icon:

| Icon | Description |
|------------------|----------------------|
| ki_waypoint_blue | Waypoint (blue) |
| ki_target | Zielsymbol |
| ki_helipad | Helipad Symbol |
| ki_medic | Medizinisches Symbol |

Beispiel:

```
{"track_obejct": {"object": QUERY, "icon": QUERY}}
{"track_object": {"object": "my_object", "icons": "ki_medic"}}
```

drive_object

`drive_object` sendet ein Objekt entlang der Wegpunktnavigation und kehrt erst zurück, wenn das Objekt fertig ist.

- `name`: Der Name des zu steuernden Objekts.
- `speed`: Geschwindigkeit, die während der Fahrt verwendet werden soll, in Meter pro Sekunde.
- `to`: ARRAY aus `LOCATIONREF` oder eine `ROUTE`
- `data`: das ist das `set_drive_data` Datenobjekt.
- `VAR1`: Wert zur Einstellung von VAR1 auf dem Missionsobjekt während der Fahrt.

Beispiel:

```
{"drive_object": {"name": "soldier_1", "to": ["pax_right_door"], "VAR1": 2, "speed": 10}}, {"drive_object": {"name": "tanker1", "to": [[34.921710973784805, -117.88296989234365, 2200, 100], [34.91159609892966, -117.90097049623692, 2500, 100], [34.894605381452905, -117.90550330903535, 2600, 100], [34.90274380665833, -117.86989409383754, 2700, 100], [34.91631769396497, -117.86277032013513, 2800, 100]], "speed": 100, "data": {"use_safety_height": true, "safety_height": 100, "max_vertical_speed": 50, "max_vertical_speed_heightdelta": 100}}}],
```

move_object

`move_object` teleportiert ein Objekt an einen neuen Ort.

Beispiel:

```
{"move_object": QUERY, "to": LOCATIONREF} {"move_object": "my_object", "to": "$USER"}
```

point_object

`point_object` ermöglicht es, ein Objekt so auszurichten, dass es auf ein anderes Objekt zeigt.

Beispiel:

```
{"point_object": QUERY, "to": LOCATIONREF} {"point_object": "my_object", "to": "$USER"}
```

set_drive_data

`set_drive_data` ermöglicht es Ihnen, das Verhalten von `drive_object` nach dem Aufruf von `drive_object` zu konfigurieren (mitten im Laufwerk).

- `use_safety_height`: Legt fest, ob ein Flugobjekt auf die Sicherheitshöhe (Boden) beschränkt ist.
- `safety_height`: Sicherheitshöhe (Mindestfunkhöhe). In Feet.

- `max_vertical_speed`: Bestimmt die maximale vertikale Steig-/Sinkgeschwindigkeit des Flugobjekts
- `max_vertical_speed_heightdelta`: Legt fest, bei welcher Höhe das Delta zur Höchstgeschwindigkeit führt. Werte, die über diesen Punkt hinausgehen, werden begrenzt.

Beispiel:

```
{set_drive_data: {
  "use_safety_height": true|false,
  "safety_height": 0
  "max_vertical_speed": 0
  "max_vertical_speed_heightdelta": 0
}}
```

set_df

`set_df` kann verwendet werden, um den Ort des aktiven Peilsignals einzustellen. (Peilquelle auf dem MFD).

Beispiel:

```
{"set_df": {"location": LOCATIONREF, "freq": QUERY}}
{"set_df": {"location": "my_boat", "freq": 255.0}}
```

set_carls_radio

`set_carls_radio` die Anzeigen von `CARLS TACTICAL RADIO` im Cockpit einstellen.

Beispiel:

```
{"set_carls_radio": {
  "LSK": ["PG1", "", ""],
  "RSK": ["", "", "INOP"],
  "Items": [
    ["Group 1", "misc contacts"],
    ["Group 2", "important"],
    ["Group 3", "other"]
  ]
}}

{"set_carls_radio": {
  "LSK": ["PG1", "", ""],
  "RSK": ["", "", "INOP"],
  "Items": [
    ["Group 1", "misc contacts"],
    {"item": ["Group 2", "important", "show_condition": ...}
    [{"text": {"Group 3 {0}=99, {1}=88"}, "params": [99, 88]}, "other"]
  ]
}}
```

Ein vollständiges Beispielprogramm finden Sie unter [Beispiele](#).

set_tfm_radio

set_tfm_radio funktioniert ähnlich wie set_carls_radio.

Beispiel:

```
{ "set_tfm_radio": {
  "main": [
    [ "DISPATCH", "168.9000" ],
    [ "BKP DISP", "169.0000" ],
  ],
  "guard": [
    [ "GUARD 1 NAME", "164.350" ],
    [ "GUARD 1 NAME", "168.350" ]
  ]
} },

{ "set_tfm_radio": {
  "main": [
    [ "DISPATCH", "168.9000" ],
    [ "BKP DISP", "169.0000" ]
  ],
  "guard": [
    { "item": [ "G1 NAME", "165.0000" ], "show_condition":
{ "require": 2, "eq": 2 } },
    [ { "text": "G{0} NAME", "params": [ 99 ], "167.0000" },
    [ "G 3 NAME", "164.350" ],
    [ "G 4 NAME", "168.350" ]
  ]
} },
```

set_rescuetrack

set_rescuetrack konfiguriert die DMAP RescueTrack-Oberfläche.

Beispiel:

```
{ "set_rescuetrack": null },

{ "set_rescuetrack": {
  "statusVar": "L:MY_DISPATCH_STATUS",
  "statusMessages": { "static": "statusMessages" },
  "dispatcherMessages": { "local": "Messages" },
  "activate_waypoint_commands": [
    { "#comment": "param - $index - in dispatcherMessages" },
    { "#comment": "param - $command - DIRECTTO" },
    { "set_message": "{ $index } { $command }" },
    { "set_route": { "struct": { "struct": { "local": "Messages", "index":
{ "param": "$index" }, "path": "waypoint" },
    { "#comment": "" }
  ]
} }

{ "set_rescuetrack": {
  "statusVar": "L:MISSION_RESCUETRACK_STATUS",
  "statusMessages": [
    "1. Unavailable for dispatch",
```

```

    "2. Ready for dispatch",
    "3. Dispatch accepted, en route to scene"
  ],
  "dispatcherMessages": [
    {
      "from": "My Dispatcher",
      "time": "00:16:00",
      "text": "the accident site is now clear, proceed to the destination",
      "waypoint": [0, 0]
    },
    {
      "from": "My Dispatcher",
      "time": "00:07:00",
      "text": "the accident site is blocked, enter a hold",
      "waypoint": [0, 0]
    }
  ]
}],
}

```

open_door

`open_door` öffnet die Flugzeugtür, wenn sie noch nicht geöffnet ist. Wenn die Tür geöffnet wird, wird auch gewartet, bis sie fertig geöffnet ist.

Beispiel:

```

{"open_door": "cockpit_left"}
{"open_door": "pax_left"}
{"open_door": "cargo_left"}

```

close_door

`close_door` schließt die Flugzeugtür, wenn sie noch nicht geschlossen ist. Wenn die Tür geschlossen wird, wird auch gewartet, bis sie fertig geschlossen ist.

Beispiel:

```

{"close_door": "cockpit_right"}
{"close_door": "pax_right"}
{"close_door": "cargo_right"}

```

create_fire

`create_fire` wird eine Reihe von Bränden erzeugen.

- `size`: Anzahl der zu entfachenden Brände
- `title`: Name des Brandobjekts, z. B. `Airbus h145 Fire`.

- `showIcon`: Optional. Standardwert `true`. ob Symbole für die Brände angezeigt werden sollen oder nicht. `L:DEBUG_CREATE_FIRE_DIST_MULT`: (mit Standardeinstellung)
`L:DEBUG_CREATE_FIRE_SIZE_MULT`: (mit Standardeinstellung)

Beispiel:

```
{ "create_fire": "fire_spawn_area", "size": { "var": ["L:MISSION_FIRE_SIZE", "number"] }, "title": "Airbus H145 Fire" },
```

launch_missile

`launch_missile` wird ein Projektil von einem Objekt auf ein anderes abschießen.

Beispiel:

```
{ "launch_missile": {
  "from": "my_ai_fighter_jet",
  "to": "$USER"
}}
```

designate_target

`designate_target` ermöglicht die Einstellung eines Ziels auf dem Zielcomputer.

Beispiel:

```
{ "designate_target": "my_target_object" }
{ "designate_target": { "location": LOCATIONREF, "alt": QUERY } }
{ "designate_target": { "location": "my_target_ground_location", "alt": 1500 } }
```

set_route

`set_route` kann verwendet werden, um einen Direct-to-Flugplan auf der Karte einzustellen

Beispiel:

```
{ "set_route": LOCATIONREF }
{ "set_route": "my_location" }
```

set_map

`set_map` kann verwendet werden, um:

1. Hinzufügen, Entfernen oder Aktualisieren von Punkten auf der Karte. Punkte können ein Symbol und/oder Text haben.
2. Hinzufügen oder Entfernen von Linien auf der Karte

Beispiel:

```
{ "set_map": { "add": { "point": { "location": "$USER", "icon": "ki_helipad", "text": "waypoint text" } } }  
  
{ "copy_location": { "bearing": 330, "dist": 500, "to": "P1" },  
{ "copy_location": { "bearing": 30, "dist": 500, "to": "P2" },  
{ "copy_location": { "bearing": 120, "dist": 500, "to": "P3" },  
{ "copy_location": { "bearing": 240, "dist": 500, "to": "P4" },  
{ "set_map": { "add": { "line": { "points": [ "P1", "P2", "P3", "P4", "P1", "stroke": { "color": "#4287f5",  
"width": 4 } } } } },  
{ "set_map": { "add": { "point": { "location": "P1", "text": "waypoint text" } } } },  
{ "set_map": { "add": { "point": { "location": "P4", "icon": "ki_helipad" } } } },
```

wait_modal

`wait_modal` kann dem Benutzer einen modalen Dialog (Singleton) anzeigen. Der Benutzer kann eine Auswahl treffen, um fortzufahren.

Beispiel:

```
{ "wait_modal": {  
  "title": "Mission Parameters",  
  "text": "Select a sling activity",  
  "options": [  
    { "text": "Utility", "style": "primary", "commands": [  
      { "#command": "use a sleep 0 here to make sure button with empty list still executes"  
        "sleep": 0  
      }  
    ] },  
    { "text": "Logging", "style": "", "commands": [  
      { "set": { "object": "cargo", "var": "VAR 1", "value": 8 },  
        { "set": { "object": "cargo2", "var": "VAR 1", "value": 8 },  
        { "set": { "object": "cargo3", "var": "VAR 1", "value": 8 },  
        { "set": { "object": "cargo4", "var": "VAR 1", "value": 8 }  
      ] }  
  ] }  
}
```

set_modal

`set_modal` funktioniert genau so wie `wait_modal`, wartet aber nicht auf die Fortsetzung der Ausführung.

set_message

`set_message` zeigt eine Meldung am unteren Rand der Missions-App an.

- `align` kann sein `left`, `center`, oder `right` (links, mitte, rechts).
- `size` kann sein `small`, `medium`, `large`, oder `extralarge` (klein, mittel, groß, oder extragroß).
- `color` kann sein `blue`, `red`, `green`, `orange`, `purple`, `hotpink`, `brown`, `cyan`, or `yellow` (Blau, Rot, Grün, Orange, Lila, Pink, Braun, Cyan oder Gelb).

Beispiel:

```
{ "set_message": { "text": "hello" } }  
{ "set_message": { "text": "hello {0}", "params": [ "dave" ] } }  
{ "set_message": { "text": "hello {0}", "params": [ { "local": "my_local" } ] } }
```


set_progressbar

`set_progressbar` wird die Anzeige eines Fortschrittsbalkens am unteren Rand der Missions-App aktivieren.

Beispiel:

```
{"set_progressbar":{"min":0,"max":100,"var":["L:TEST","number"],  
"color":"green"}}  
{"set_progressbar":null}
```

set_dispatch

`set_dispatch` erlaubt es, den Inhalt des Einsatzdialogs einzustellen. Dieser ist ähnlich wie das Briefing, kann aber während der Mission geändert werden. Es stehen alle Widgets des Briefings zur Verfügung.

Beispiel:

```
{"set_dispatch": [  
  {"text":"hello world"}  
]}
```

set_briefing_dialog

`set_briefing_dialog` öffnet oder schließt den Briefing-Dialog.

Beispiel:

```
{"set_briefing_dialog": QUERY}  
{"set_briefing_dialog": 1}  
{"set_briefing_dialog": 0}
```

set_dispatch_dialog

`set_dispatch_dialog` öffnet oder schließt den Einsatzdialog.

Beispiel:

```
{"set_dispatch_dialog": QUERY}  
{"set_dispatch_dialog": 1}  
{"set_dispatch_dialog": 0}
```

scroll_to_briefing_item

`scroll_to_briefing_item` blättert auf der Informationsseite zu dem genannten Abschnitt.

Beispiel:

```
{"scroll_to_briefing_item": "header1"}
```

scroll_to_dispatch_item

`scroll_to_dispatch_item` blättert auf der Einsatzseite zu dem genannten Abschnitt.

Beispiel:

```
{"scroll_to_dispatch_item": "header1"}
```

set_objective_title

`set_objective_title` ermöglicht das Ändern des Zieltitels (Text am unteren Rand der Missions-App) zu einem anderen Zeitpunkt als dem, zu dem die Zielliste selbst zum nächsten Ziel wechselt.

- `color` kann sein `blue`, `red`, `green`, `orange`, `purple`, `hotpink`, `brown`, `cyan`, or `yellow` (Blau, Rot, Grün, Orange, Lila, Pink, Braun, Cyan oder Gelb).

Beispiel:

```
{"set_objective_title": QUERY}  
{"set_objective_title": "Fly to the target"}
```

set_hover_display

`set_hover_display` ermöglicht es Ihnen, ein Fadenkreuz auf der Missionskarte anzuzeigen.

range: Meter

Beispiel:

```
{"set_hover_display": {"target": LOCATIONREF, "range": QUERY}},  
{"set_hover_display": {"target": "load1_dest", "range": 0.02}},
```

create_user_action

Eine Benutzeraktion ist ein Befehl, den der Benutzer anklicken kann und der oben auf der Missionskarte angezeigt wird.

`create_user_action` wird die benannte Benutzeraktion erstellen. `click_commands` ist eine COMMANDLIST, die ausgeführt wird, wenn der Benutzer die Schaltfläche anklickt oder den Hotkey betätigt.

Beispiel:

```
{ "create_user_action": {
  "id": "accept_dispatch",
  "title": "Accept Dispatch",
  "click_commands": [
    { "destroy_user_action": "accept_dispatch" }
  ]
}},

{ "create_user_action": { "id": "change_accident_location", "title": "Change Location",
"click_commands": [
  { "set_message": { "text": "" } },
  { "call_macro": "user_pick_accident_location" },
  { "set_route": "accident_location" },
  { "set_message": { "text": "Accident Location: {0:LOCATION}", "params": [ "accident_location" ] } }
] } },
```

destroy_user_action

`destroy_user_action` entfernt eine bestehende Benutzeraktion.

Beispiel:

```
{ "destroy_user_action": "my_action" }
```

trigger_user_action

`trigger_user_action` löst manuell eine Benutzeraktion aus, als ob der Benutzer darauf geklickt hätte.

Beispiel:

```
{ "trigger_user_action": "my_action" }
```

set_user_poi

`set_user_poi` ermöglicht das Anklicken der Karte im Namen des Benutzers.

Beispiel:

```
{ "clear_user_poi": 1 }
```

create_route

`create_route` nutzt einen Online-Dienst zur Berechnung von Anweisungen für den Transit über das Straßennetz von einem Ort zum anderen. Nach dem Aufruf von `create_route` steht der Name als Referenz für andere APIs zur Verfügung.

- `type`: Optional. Standardmäßig auf `car` (Auto).

Beispiel:

```
{"create_route": {name: "route-name-here", "query": {
  "location_from": LOCATIONREF,
  "location_to": LOCATIONREF,
  "type": "car|foot|bike"
}}}

{"create_route": {name: "my_route_name", "query": {
  "location_from": "$USER",
  "location_to": {"bearing":0, "dist": 1000},
  "type": "car"
}}}
```

draw_route

`draw_route` zeichnet Linien auf der Karte für die angegebene Route.

`Linie`: Optional. Standardwert ist `{width: 8, color: '#FF33FF' }`

Beispiel:

```
{"draw_route": "route_name", "id":"my_route_id"}
```

copy_stringtoken

`copy_stringtoken` kopiert ein String-Token nach Namen in einen anderen Namen.

Beispiel:

```
{"copy_stringtoken": "token1", "to": "token2"}
```

open_url

`open_url` öffnet ein Webbrowser-Fenster auf dem PC des Benutzers.

Beispiel:

```
{"open_url": QUERY}
{"open_url":"https://hypeperformancegroup.com/"}
```

copy_location

`copy_location` nimmt eine `LOCATIONREF`, löst sie sofort auf und speichert sie dann unter einem neuen Namen.

Beispiel:

```
{"copy_location":LOCATIONREF, "to": "my_new_location_name"}
{"copy_location":"my_location_name", "to": "my_new_location_name"}
```

open_location

`open_location` öffnet Google Maps zu einem bestimmten `LOCATIONREF`

Beispiel:

```
{"open_location": LOCATIONREF}
{"open_location": "object1"}
{"open_location": [34.1, -122.9]}
```

create_location

`create_location` erstellt einen Ortsnamen, indem es aus den angebotenen Zonen auswählt und den Ort aus den Informationen der Zone erstellt. Die Zonen werden nach dem Zufallsprinzip aus der Liste ausgewählt, Sie können aber auch einfach eine Zone angeben, wenn Sie möchten.

Format:

```
{"create_location": "location_name", zones: [ZONE]}
{"create_location": "location_name", zones: [ZONE1, ZONE2, ZONE3, ...]}
{"create_location": "location_name", zones: [ZONE], no_results_commands: COMMANDLIST}
```

`no_results_commands`: Optional. Standardmäßig wird ein modaler Dialog erstellt, wenn die Datenabfrage nicht erfolgreich ist.

Eine `ZONE` hat folgende Eigenschaften:

- `location`: `LOCATIONREF` die das Zentrum der Zone ist.
- `radius`: Radius der Zone in Metern.
- `minRadius`: Optional. Der Standardwert ist 0. Meter.
- `commands`: `COMMANDLIST`, die ausgeführt werden soll. Der Parameter `$LOCATION:NAME` enthält den standortbezogenen Namen.
- `zone_type`: Wählen Sie aus der nachfolgenden Liste.
- `query`: `DATAQUERY` (nur wenn ein Datenabfragebereich)

Zone Typen:

| Zone | Beschreibung |
|----------------------|---|
| random_point | Wähle eine zufällige Position innerhalb dieses Ortes. |
| query_list_result | Data Query: Führt die Abfrage aus und präsentiert dann eine Liste von Ergebnissen, aus denen der Benutzer wählen kann. |
| query_random_result | Data Query: Führt die Abfrage aus und wählt dann ein zufälliges Ergebnis |
| query_closest_result | Data Query: Führt die Abfrage aus und wählt dann das nächstgelegene Ziel aus. Wenn die Abfrage fehlschlägt, vergrößert sich der Bereich und versucht es erneut, bis es ein Ergebnis gibt. |

Wenn eine Datenabfrage ausgewählt wird, werden die folgenden Parameter nach dem Aufruf ausgefüllt:

- `$LOCATION:NAME` der Name-Tag auf dem Ergebnis.
- `$LOCATION:ID` die ID des Ergebnisses.

Beispiel:

```
{
  "create_location": "$LOCATION",
  "zones": [
    {
      "zone": {
        "zone_type": "query_closest_result",
        "query": "[out:json];way({bbox})[highway~\"^(motorway|trunk|primary|secondary|tertiary|motorway|trunk|primary|secondary)_link\"]->.major;way({bbox})[highway~\"^(unclassified|residential|living_street|service)\"]->.minor;node(w.major)(w.minor);out;",
        "location": "city_center",
        "radius": 25000,
        "minRadius": 0,
        "commands": []
      }
    }
  ]
}
```

query_data

`query_data` können Sie OSM-Daten abfragen und einen Callback für die Ergebnisse erhalten. Beachten Sie, dass dies eine alte API ist, die vor `for_each` aufgerufen wird.

- `location`: LOCATIONREF
- `query`: DATAQUERY
- `radius`: Meter
- `minRadius`: Optional. Default ist 0
- `commands`: ARRAY OF COMMANDLIST zur Rückgabe der Ergebnisse.
- `no_results_commands`: COMMANDLIST die auszuführen, wenn es nicht genügend Ergebnisse gibt (Länge von)
- `$LOCATION` (verwendbarer temporärer Standortname) wird bei jedem Rückruf von `commands` anders definiert.
- `$LOCATION` (Parameter) wird bei jedem Rückruf von `commands` anders definiert.
- `$LOCATION:NAME` (Parameter) trägt den ortsbezogenen Namen.

Verwenden Sie `bypass_commands` und `$ITEMS`, um die gesamte Liste zu verarbeiten..

Beispiel:

```
{
  "query_data": {
    "query": "[out:json]; ( node({bbox})[power=substation]; area({bbox})[power=substation]; );
    out center;";
    "location": "city_center",
    "radius": 25000,
    "minRadius": 0,
    "commands": [
      [{"set": {"var": ["L:MISSION_LOC_POWER_0", "number"], "value": {"location": "$LOCATION"}},
{"set": {"var": ["L:MISSION_SCORE_POWER_0", "number"], "value": 0}},
      [{"set": {"var": ["L:MISSION_LOC_POWER_1", "number"], "value": {"location": "$LOCATION"}},
{"set": {"var": ["L:MISSION_SCORE_POWER_1", "number"], "value": 0}},
      [{"set": {"var": ["L:MISSION_LOC_POWER_2", "number"], "value": {"location": "$LOCATION"}},
{"set": {"var": ["L:MISSION_SCORE_POWER_2", "number"], "value": 0}},
      [{"set": {"var": ["L:MISSION_LOC_POWER_3", "number"], "value": {"location": "$LOCATION"}},
{"set": {"var": ["L:MISSION_SCORE_POWER_3", "number"], "value": 0}},
      [{"set": {"var": ["L:MISSION_LOC_POWER_4", "number"], "value": {"location": "$LOCATION"}},
{"set": {"var": ["L:MISSION_SCORE_POWER_4", "number"], "value": 0}},
      [{"set": {"var": ["L:MISSION_LOC_POWER_5", "number"], "value": {"location": "$LOCATION"}},
{"set": {"var": ["L:MISSION_SCORE_POWER_5", "number"], "value": 0}},
      [{"set": {"var": ["L:MISSION_LOC_POWER_6", "number"], "value": {"location": "$LOCATION"}},
{"set": {"var": ["L:MISSION_SCORE_POWER_6", "number"], "value": 0}},
      [{"set": {"var": ["L:MISSION_LOC_POWER_7", "number"], "value": {"location": "$LOCATION"}},
{"set": {"var": ["L:MISSION_SCORE_POWER_7", "number"], "value": 0}},
      [{"set": {"var": ["L:MISSION_LOC_POWER_8", "number"], "value": {"location": "$LOCATION"}},
{"set": {"var": ["L:MISSION_SCORE_POWER_8", "number"], "value": 0}},
      [{"set": {"var": ["L:MISSION_LOC_POWER_9", "number"], "value": {"location": "$LOCATION"}},
{"set": {"var": ["L:MISSION_SCORE_POWER_9", "number"], "value": 0}}
    ]
  }
}
```

query_country

`query_country` ermittelt den Ländernamen (String) für einen Ort.

- `None` ist ein besonderer Ländername, der sich auf den offenen Ozean bezieht.
- `$COUNTRY` (param) wird nach der Rückkehr des Aufrufs definiert.
- `$COUNTRY` (stringToken) wird nach der Rückkehr des Aufrufs definiert.

Beispiel:

```
{
  "query_country": {
    "United States of America": [ {"set_message": {"text": "USA country $COUNTRY" }} ],
    "France": [ {"set_message": {"text": "FR country" }} ],
    "Germany": [ {"set_message": {"text": "DE country" }} ],
    "Other": [ {"set_message": {"text": "Other country: $COUNTRY" }} ],
    "None": [ {"set_message": {"text": "You are over open water. ($COUNTRY)" }} ]
  },
  "location": [65.34528194493097, -12.372530650689942]
}
```

osm_query_data

Verwenden Sie `osm_query_data`, um OSM nach Daten innerhalb eines bestimmten Gebiets abzufragen. Operationen auf den Daten nach `osm_query_data` verwenden das Netzwerk nicht.

Beispiel:

```
{ "#comment": "Query a block of road network data and save it into my_data",
  "osm_query_data":
    "[out:json];way({{bbox}})[highway~\\\"^(motorway|trunk|primary|secondary|unclassified|residential|living_street|service|tertiary|(motorway|trunk|primary|secondary|tertiary|)_link)$\\"];
    (._;>);out;";
    "location": "LOC",
    "size": 600,
    "result": "my_data"
  },
}
```

osm_get_parent_ways

Anhand einer `NodeId` liefert `osm_get_parent_ways` ein Array mit den Wegen, die diese Id enthalten. Verwenden Sie dies, um herauszufinden, zu welchen Wegen ein bestimmter Knoten gehört.

Beispiel:

```
{ "osm_get_parent_ways": { "struct": { "param": "$item", "path": "id" }, "data":
  { "param": "my_data", "result": "parents" },
}
```

osm_get_connected_nodes

Verwenden Sie `osm_get_connected_nodes`, um Knoten zu finden, die an die angegebene `nodeId` angrenzen. Dies ist gut für die Suche nach den Schenkeln einer Kreuzung oder den Auf-/Abwärtsknoten entlang einer Straße.

Beispiel:

```
{ "osm_get_connected_nodes": { "struct": { "param": "closest_node", "path": "id" },
  "data": { "param": "my_data", "result": "my_nodes_connected_to_nearest_node" },
}
```

osm_get_nodes

Verwenden Sie `osm_get_nodes`, um eine geordnete Liste aller Knoten innerhalb einer bestimmten `wayId` zu erhalten. Verwenden Sie dies, um eine Liste von Koordinaten entlang einer Straße zu erhalten.

Beispiel:

```
{ "osm_get_nodes": { "struct": { "param": "$item", "path": "id" }, "data":
  { "param": "my_data", "result": "my_nodes_on_way" },
}
```


osm_get_all_ways

Verwenden Sie `osm_get_all_ways`, um eine Liste aller Wege innerhalb des Datensatzes zu erhalten.

Beispiel:

```
{"osm_get_all_ways": {"param":"my_data"}, "result":"my_ways"},
```

osm_get_all_nodes

Verwenden Sie `osm_get_all_nodes`, um eine Liste aller Knoten innerhalb des Datensatzes zu erhalten.

Beispiel:

```
{"osm_get_all_nodes": {"param":"my_data"}, "result":"my_nodes"},
```

osm_get_closest_nodes

Verwenden Sie `osm_get_closest_nodes`, um eine geordnete Liste von Knoten zu erstellen, geordnet nach der Entfernung zum angegebenen `LOCATIONREF`.

Beispiel:

```
{"osm_get_closest_nodes": "LOC", "data": {"param":"my_data"}, "result":"my_closest_nodes"},
```

osm_is_point_within_way

Verwenden Sie `osm_is_point_within_way`, um festzustellen, ob ein gegebener `LOCATIONREF` innerhalb des geschlossenen Weges liegt.

Beispiel:

```
{"osm_is_point_within_way": {"struct":{"param":"way"}, "path":"id"}, "location":{"bearing": {"param":"brg"}, "dist":{"param":"dist"}}, "data":{"param":"my_data"}, "result":"is_in"},
```

osm_get_area_of_area

Verwenden Sie `osm_get_area_of_area`, um die Fläche des geschlossenen Weges in Quadratmetern zu berechnen.

Beispiel:

```
{"osm_get_area_of_area":{"struct":{"param":"way"}, "path":"id"}, "data": {"param":"my_data"}, "result":"way_area"},
```

open_table

`open_table` öffnet eine bestehende Datentabelle oder erstellt eine neue. Sobald die Tabelle geöffnet ist, sind Tabellenbefehle gültig zu verwenden.

Beispiel:

```
{"open_table": QUERY}
{"open_table": "my_table"}
```

save_table

`save_table` wird die Tabelle sofort auf der Festplatte speichern. Änderungen an Tabellen, bei denen `save_table` nicht abschließend (vor Verlassen der Mission) aufgerufen wird, gehen verloren.

Beispiel:

```
{"save_table": QUERY}
{"save_table": "my_table"}
```

clear_table

`clear_table` entfernt alle Schlüssel aus der Tabelle.

Beispiel:

```
{"clear_table": QUERY}
{"clear_table": "my_table"}
```

play_audio

`play_audio` aktiviert die Wiedergabe von eingebauten Audiosignalen. `play_audio` wird erst fortgesetzt, wenn das Signal zu Ende gespielt wurde.

Beispiel:

```
{"play_audio": "hold_position"}
{"play_audio": "4"}
```

Sound Liste:

```
0 1 2 3 4 5 6 7 8 9 10
we_are_not_in_range
we_are_too_high
```

```
we_are_too_low
hold_position
the_cabin_is_secure
forward
backward
left
right
ready_for_you_to_approach_and_hoist
ready_for_you_to_approach_and_land
tablet_alarm1
```

play_guidance_message

`play_guidance_message` kann verwendet werden, um den Zielort akustisch anzuleiten.

- `target`: Das entfernte Ziel zur Anleitung von
- `self`: Dies ist die Position an Ihrem Flugzeug, die mit dem Mittelpunkt des Zielobjekts übereinstimmen sollte, z. B. `$USER:HOIST` für die Position der Hebezeugvorrichtung.

Beispiel:

```
{"play_guidance_message": {"target": LOCATIONREF, "self": LOCATIONREF}},
{"create_thread": {"name": "main_crash_guidance_thread", "commands": [{"while": {"var":
["L:MISSION_GUIDANCE_ENABLED", "number"], "eq": 1, "do": [
{"wait_for": {"location": "main_crash", "var": "distance"}, "lt": 0.03},
{"play_guidance_message": {"target": "main_crash", "self": "$USER:HOIST"}},
{"sleep": 2}
]}]}}
```

connect_voice_server

`connect_voice_server` versucht, sich mit dem definierten Sprachdienst zu verbinden. `on_connected`-Befehle werden bei Erfolg ausgeführt, `on_disconnected` wird beim Trennen der Verbindung ausgeführt, auch wenn dies viel später geschieht.

Beispiel:

```
{"connect_voice_server": {
  "on_connected": [
    {"speak": "Speech activated."}
  ],
  "on_disconnected": [
    {"set_message": {"text": "No voice server available"}}
  ]
}}
```

speak

`speak` sendet einen Befehl an den Sprachserver zur Wiedergabe von Text-to-Speech oder einer Audiodatei.

`interrupt`: 1 oder 0. 1 bricht die Warteschlange ab und spielt sofort ab. `is_audio_file`: 1 oder 0. Bei 1 wird davon ausgegangen, dass es sich bei dem Text um die Datei `filename.wav` in dem für den Server verfügbaren Audioverzeichnis handelt.

Beispiel:

```
{"speak":"hello"}
{"speak":{"text":"hello {0}", "params":["dave"]}}
{"speak":"hello.wav", "is_audio_file":1}
```

Debugger & Fernsteuerungsbefehle

cancel_debugger

`cancel_debugger` sollte von Nicht-Debug-Remote-Missionen verwendet werden, dann wird die zusätzliche Debugger-Aktivität unterdrückt.

- Dieser Befehl gilt für die Verwendung eines entfernten Kontexts (eine Mission, die über einen Websocket verbunden ist) oder für die Verwendung des Debuggers (ein Tool, das auf demselben rpc basiert).

remote_notify

`remote_notify` meldet Daten an den entfernten Server, falls verfügbar.

- Dieser Befehl gilt für die Verwendung eines entfernten Kontexts (eine Mission, die über einen Websocket verbunden ist) oder für die Verwendung des Debuggers (ein Tool, das auf demselben rpc basiert).

Beispiel:

```
{"remote_notify":"my_connected_event"}
{"remote_notify":"hello_event", "params":[
  {"var":["A:PLANE ALTITUDE", "feet"]},
  {"var":["A:PLANE BANK DEGREES", "bank"]}
]}
```

teleport_to

`teleport_to` setzt den Breiten- und Längengrad des Spielerflugzeugs und teleportiert es sofort. Beachten Sie, dass dies einige Arbeit erfordert, um den Schwenkmodus für den Benutzer zu aktivieren und die Höhe anzupassen.

fetch

`fetch` ermöglicht die Interaktion mit entfernten Webdiensten unter Verwendung der Javascript-Fetch-API.

Beispiel:

```
{ "fetch": {
  "url": "http://127.0.0.1:3000/report?key=hello",
  "method": "POST",
  "headers": {
    "Accept": "application/json",
    "Content-Type": "application/json"
  },
  "body": { "param": "msg" }
},
},
```

set_shared_data

`set_shared_data` ändert den Zustand der gemeinsam genutzten Multiplayer-Daten. Es wird implizit die zuletzt erstellte Multiplayer-Verbindung verwendet.

Beispiel:

```
{ "set_shared_data": "update", "path": "connectedAircraft.{service_auth}.isHost", "value": true },
```

debug_write

`debug_write` sendet eine Zeichenkette an `console.log`.

hoist_control

`hoist_control` ermöglicht das Auf- und Abwickeln des Hebezeugs. Weitere Informationen finden Sie im Thema Hebezeug.

Beispiel:

```
{ "hoist_control": "reel_down", "speed": 1 }
{ "hoist_control": "reel_up", "speed": 1 }
```

API Reference - QUERY

Alle **QUERY** sind unten aufgeführt. Zusätzlich sind **Zahlen**, **Strings**, **Null** und **Arrays** (pass-through) als **Abfrage** gültig.

text

text kann zusammen mit **params** zum Aufbau einer beliebigen Zeichenfolge verwendet werden. **{N}** wird als Ersetzungstoken in der Zeichenkette verwendet. N beginnt bei 0 und wird hochgezählt und entspricht den Elementen in **params**.

Beispiel:

```
{"text": "object_name_{0}", "params": [ 99 ]}  
{"text": "object_name_{0}", "params": [ {"var": ["L:TEST", "number"]} ]}
```

Das Ergebnis ist eine Zeichenkette, die an eine andere API gesendet werden kann, mit einem Wert wie **object_name_99**.

Formatbezeichner:

```
{0}  
{0:TIME}  
{0:DMS}  
{0:LOCATION}
```

var

var ist die wichtigste Methode, um eine L:Var oder A:var aus dem Simulator zu lesen. Eine Liste verschiedener Variablen ist [hier verfügbar](#).

Alle **L:Vars** verwenden die **Unit**-Nummer.

Beispiel:

```
{"var": ["L:MY_SIM_VAR_HERE", "number"]}  
{"var": ["A:PLANE ALTITUDE", "feet"]}
```

object/var

Lesen einer Eigenschaft eines Missionsobjekts.

| Eigenschaft | Funktion |
|-------------|--|
| \$INDEX | Verknüpft einen Missionsobjektnamen mit L:MISSION OBJECT ... vars |
| VAR 1 | Allgemeiner Datenbereich 1 |

| Eigenschaft | Funktion |
|--------------------|---|
| VAR 2 | Allgemeiner Datenbereich 2 |
| VAR 3 | Allgemeiner Datenbereich 3 |
| MODE | Objektmodus |
| HEIGHT | Funkhöhe in Fuss (readonly) |
| ALT | Höhe in Fuss |
| AALT | Actual altitude in feet (readonly) |
| AHDG | Aktueller Kurs in Grad (readonly) |
| CREATED | 1: erstellt, 0: nicht erstellt, -1: fehlgeschlagene Erstellung (schreibgeschützt) |
| COUPLED | Spezial-Objekt-Modus. Siehe Tabelle unten |
| VELOCITY X | Geschwindigkeit X (Meter pro Sekunde) |
| VELOCITY Y | Geschwindigkeit Y (Meter pro Sekunde) |
| VELOCITY Z | Geschwindigkeit Z (Meter pro Sekunde) |
| WP INDEX | Wegpunktnavigation 0: inaktiv, >0: stellt einen aktiven Wegpunktindex dar. |
| distance | Entfernung zu einem Objekt berechnen |
| distance:ft | Entfernung zu einem Objekt berechnen (Einheiten umrechnen) |

Generische Daten-Slots haben nur für das jeweilige Objekt eine Bedeutung (z. B. das Setzen eines Animationsstatus, eine übliche Verwendung für **VAR 1**).

Diese Werte gelten für die **MODE** var:

| Objekt Mode | Funktion |
|--------------------|--|
| 0 | Standardmodus. Sie können VELOCITY Z einstellen |
| 1 | Neupositionierungsmodus. Sie können LAT , LON und HDG einstellen. |
| 2 | 3-Achsen-Geschwindigkeit |
| 3 | Standard-MSFS-Physik |
| 4 | Wie 1, aber um 180 Grad gedrehte Neigung |
| 5 | Stopp bei einer Funkhöhe von unter 10 Fuß |
| 6 | Flugobjekt neu positionieren. ALT und HDG einstellen |

Diese Werte gelten für die Variable **COUPLED**:

| Objekt Mode | Funktion |
|--------------------|---|
| 0 | Standardmodus. |
| 1 | Gekoppelt an die Hebezeugstation (hoist station) |
| 2 | Gekoppelt an die Hebevorrichtung (sling station) |
| 3 | Available to be coupled to sling station |
| 4 | Ziel der Brandbekämpfung VAR 1: Menge des Feuers |
| 5 | Löschwasserquelle VAR 1: Radius in Metern, VAR 2: Höhe in Fuß |
| distance:ft | (Einheitenumrechner) wird ebenfalls unterstützt |

Beispiel:

```
{"object": "my_object", "var": "VELOCITY Z"}
{"object": "my_object", "var": "$INDEX"}
{"object": "my_object", "var": "distance"}
```

location/var

Aus einem `LOCATIONREF` können Sie 3 Dinge ablesen: Entfernung in Seemeilen, Breitengrad und Längengrad.

`distance:ft` (Einheitenumrechner) wird ebenfalls unterstützt

Beispiel:

```
{"location": LOCATIONREF, "var": "distance"}
{"location": LOCATIONREF, "var": "lat"}
{"location": LOCATIONREF, "var": "lon"}
```

bearing

Peilung (`bearing`) berechnet den wahren Kurs zwischen zwei `LOCATIONREF`.

Beispiel:

```
{"bearing": {"to":LOCATIONREF, "from":LOCATIONREF}}
```

has_location

`has_location` gibt 1 oder 0 zurück, je nachdem, ob der Ortsname bereits existiert.

Beispiel:

```
{"has_location": QUERY}
{"has_location": "my_location_name"}
```

resolve_location

`resolve_location` gibt `[lat, lon]` aus einer `LOCATIONREF` zurück.

Beispiel:

```
{"has_location": QUERY}
{"has_location": "my_location_name"}
```


has_object

`has_object` gibt 1 oder 0 zurück, je nachdem, ob der Objektname bereits existiert.

Beispiel:

```
{"has_object": QUERY}
{"has_object": "my_object_name"}
```

has_user_action

`has_user_action` gibt 1 oder 0 zurück, je nachdem, ob die `user_action` gerade aktiv ist.

Beispiel:

```
{"has_user_action": QUERY}
{"has_user_action": "my_user_action_name"}
```

has_mission

`has_mission` gibt 1 oder 0 zurück, je nachdem, ob die Missions-ID im Index vorhanden ist.

Beispiel:

```
{"has_mission": QUERY}
{"has_mission": "my_mission_id_"}
```

has_macro

`has_macro` gibt einen booleschen Wert zurück, der angibt, ob der Makroname existiert oder nicht.

Beispiel:

```
{"has_macro": QUERY}
{"has_macro": "my_macro"}
```

no_resolve

`no_resolve` gibt nur die nicht ausgewerteten Daten zurück.

Beispiel:

```
{"no_resolve": {"arbitrary_data_here": "my_data"}}
```

resolve_icon

`resolve_icon` sucht einen Eintrag in der Icon-Tabelle.

Beispiel:

```
{"resolve_icon": "my_icon_name"}
```

static

`static` ermöglicht das Abrufen von Schlüsseln unter dem Datenabschnitt (`data section`) der Mission (statische Daten).

Beispiel:

```
"data": {  
  "my_static_key": 99  
}  
  
{"static": "my_static_key"}
```

has_static

`has_static` gibt einen booleschen Wert zurück, der angibt, ob der Schlüssel im Missionsdatenabschnitt (`data section`) existiert.

Beispiel:

```
{"has_static": "my_static_key"}
```

has_global

`has_mission` gibt 1 oder 0 zurück, je nachdem, ob der globale Name definiert ist.

Beispiel:

```
{"has_global": QUERY}  
{"has_global": "my_global_name"}
```

global

`global` ermöglicht die Abfrage einer globalen Variable nach Namen.

Beispiel:

```
{"global": QUERY}  
{"global": "my_global_name"}
```

has_route

`has_route` gibt einen booleschen Wert zurück, der angibt, ob die angegebene Route existiert.

Beispiel:

```
{"has_route": QUERY}
{"has_route": "my_route_name"}
```

route

`route` gibt die Routeninformationen für einen bestimmten Routennamen zurück.

Beispiel:

```
{"route": QUERY}
{"route": "my_route_name"}
```

create_array

`create_array` erstellt ein neues Array mit der angegebenen Größe. Arrays wachsen automatisch, daher ist 0 in Ordnung.

Beispiel:

```
{"create_array": QUERY}
{"create_array": 10}
```

create_struct

`create_struct` erstellt ein komplexes Objekt und jeder Schlüssel wird als QUERY ausgewertet

Beispiel:

```
{"create_struct": {
  "key1": QUERY,
  "key2": QUERY
}}
```

struct

`struct` wird für den Zugriff auf ein komplexes Objekt verwendet.

- `path`: Zugriff auf eine Eigenschaft.
- `has_path`: 1 oder 0, je nachdem, ob die Eigenschaft vorhanden ist.
- `function`: Funktion aufrufen
- `index`: Zugriff auf ein Arrayelement

Beispiel:

```
{"struct": ..., "path": "length"}
{"struct": {"js:get":"JSON"}, "function": "stringify", "params": [ {"local": "my_local"}, null, 2]}
{"struct": ..., "index": 0}
```

js:get

`js:get` holt ein Objekt aus dem `window`. Beispiele sind `Math` oder `JSON`.

Beispiel:

```
{"js:get": "Math"}
```

js:create_async_function

`js:create_async_function` erstellt eine asynchrone JS-Funktion, die eine `COMMANDLIST` mit `$args` als Parameter aufruft.

Beispiel:

```
{"js:create_async_function": [
  {"set_message":{"text":"js called:{0}", "params": [
    {"struct": {"param": "$args"}, "index": 0}
  ]}}
]}
```

js:function

`js:function` erstellt eine JS-Funktion, die eine `QUERY` mit `$args` als Param aufruft. Da es sich um eine `QUERY` handelt, können Sie auch synchron einen Wert zurückgeben.

Beispiel:

```
{"js:create_callback": [
  {"set_message":{"text":"js called:{0}", "params": [
    {"struct": {"param": "$args"}, "index": 0}
  ]}}
]}
```

js:new

`js:new` ruft den Konstruktor für ein Objekt auf und gibt `Parameter` an, wenn er definiert ist.

Beispiel:

```
{"js:new": "my_window_object", "params": [QUERY, QUERY, QUERY]}
```

json:stringify

`json:stringify` wandelt ein Objekt in eine JSON-Zeichenkette um.

Beispiel:

```
{"json:stringify": {"param":"$RET"}}
```

json:parse

`json:parse` wandelt eine JSON-Zeichenkette in ein Objekt um.

Beispiel:

```
{"json:parse": {"param":"$RET"}}
```

json:copy

`json:copy` wird eine Kopie des Objekts erstellen. Änderungen an dem neuen Objekt haben keine Auswirkungen auf das Eingabeobjekt.

Beispiel:

```
{"json:copy": {"param":"$RET"}}
```

object:keys

`object:keys` gibt ein Array mit den Schlüsselnamen des Zielobjekts zurück.

Beispiel:

```
{"object:keys": {"param":"$RET"}}
```

string:split

`string:split` erstellt ein Array aus den Teilen der Zeichenkette, die durch den Begrenzer angegeben sind.

`index`: Optional. Dies gibt nur einen Index im Array zurück, anstatt alle Teile des Arrays. Dies ist praktisch, wenn Sie ohnehin nur einen Teil der geteilten Zeichenfolge benötigen.

Beispiel:

```
{"string:split": {"struct": {"js:new": "Date"}, "function": "toISOString"}, "delimiter": "T", "index": 1}
```

string:join

`string:join` erstellt eine Zeichenkette, indem es jedes Element im Eingabe-Array zusammen mit einem Begrenzungszeichen anhängt.

Beispiel:

```
{"string:join": ["one", "two", "three"], "delimiter": "_"}
```

create_number

`create_number` verwendet die js `Number()`, um eine Zeichenkette in einen Zahlenwert umzuwandeln.

Beispiel:

```
{"create_number": QUERY}  
{"create_number": "99.5"}
```

has_local

`has_local` gibt 1 oder 0 zurück, je nachdem, ob der Schlüssel in den Locals vorhanden ist.

Beispiel:

```
{"has_local": "my_local_name"}
```

local

`local` liefert eine lokale Variable nach Namen.

Beispiel:

```
{"local": "my_local_name"}  
{"local": "my_local_name", "path": "key"}
```

gamevar

`gamevar` funktioniert wie `var`, ermöglicht aber die Abfrage von `SimVar.GetGameVarValue` im MSFS.

Beispiel:

```
{"gamevar": ["my_game_var", "my_unit"]}
```

table

`table` können Sie einen Schlüssel aus einer benannten Tabelle lesen. (Die Tabelle muss zuerst geöffnet werden)

Beispiel:

```
{"table": "my_table", "key": "my_key"}
```

param

`param` lässt Sie einen Parameter aus der `params`-Sammlung lesen. Es gibt eine `params`-Sammlung für jedes Makro und eine für den Haupt-Thread. `create_threads` nehmen die gleichen `params` wie der aufrufende Kontext.

Beispiel:

```
{"param": "my_param"}  
{"param": "my_param", "path": "my_key"}
```

has_param

`has_param` liefert die Angabe 1 oder 0 für das Vorhandensein oder Nichtvorhandensein des Parameterschlüssels.

Beispiel:

```
{"has_param": "my_param"}
```

rand

`rand` erzeugt einen zufälligen Dezimalwert zwischen den Grenzen von `QUERY` (Minimum) und `QUERY2` (Maximum).

Beispiel:

```
{"rand": [QUERY1, QUERY2]}  
{"rand": [0, 60]}
```

add

`add` erzeugt eine Liste von Abfragen (2 oder mehr).

Beispiel:

```
{"add": [QUERY1, QUERY2, ...]}  
{"add": [2, 2]}  
{"add": [{"var": ["L:TEST", "number"]}, 1]}
```

add360

`add360` ist wie `add`, aber das Endergebnis ist zwischen 0 und 360 normalisiert.

Beispiel:

```
{"add360":[QUERY1, QUERY2, ...]}
{"add360":[{"var":["L:TEST","number"]}, 90]}
```

compare360

`compare360` liefert den absoluten Wert zwischen zwei Werten 0-360.

Beispiel:

```
{"compare360":[1, 359]} // -> 2
```

subtract

`subtract` Subtrahiert `QUERY1` - `QUERY2`.

Beispiel:

```
{"subtract":[QUERY1, QUERY2]}
```

multiply

`multiply` Multipliziert `QUERY1` * `QUERY2`.

Beispiel:

```
{"multiply":[QUERY1, QUERY2]}
```

divide

`divide` Dividiert `QUERY1` / `QUERY2`. wenn `QUERY2` gleich Null ist, lautet das Ergebnis `0`.

Beispiel:

```
{"divide":[QUERY1, QUERY2]}
```

right_shift

`right_shift` ist der Bitverschiebeoperator nach rechts `>>`.

`QUERY >> QUERY2`

Beispiel:

```
{"right_shift": [QUERY1, QUERY2]}
{"right_shift": [0xFFFF, 2]}
```


left_shift

`left_shift` ist der Bitverschiebeoperator nach links \ll .

`QUERY` \ll `QUERY2`

Beispiel:

```
{"left_shift": [QUERY1, QUERY2]}  
{"left_shift": [0xFFFF, 2]}
```

xor

`xor` ist der Exklusiv-Oder-Operator.

Beispiel:

```
{"xor": [QUERY1, QUERY2]}
```

remainder

`remainder` ist der Mod- oder Restoperator.

Beispiel:

```
{"remainder": [QUERY1, QUERY2]}
```

exponent

`exponent` ist die Exponentialfunktion

Beispiel:

```
{"exponent": [QUERY1, QUERY2]}
```

round

`round` rundet eine Zahl auf den nächstliegenden ganzzahligen Wert.

Beispiel:

```
{"round": QUERY}  
{"round": 3.5}
```

toFixed

`toFixed` ist wie `round`, aber Sie können die Anzahl der Stellen festlegen, auf die gerundet werden soll. `digits=2` würde Zahlen wie `0.00` ergeben.

Beispiel:

```
{"toFixed":QUERY, "digits": QUERY}
{"toFixed":3.141592, "digits": 2}
```

floor

`floor` gibt die nächstliegende vorherige ganze Zahl zurück.

Beispiel:

```
{"floor":QUERY}
{"floor":2.5}
```

ceil

`ceil` (Höchstwert) gibt die nächstliegende ganze Zahl zurück.

Beispiel:

```
{"ceil":QUERY}
{"ceil":2.5}
```

abs

`abs` gibt den absoluten Wert einer Zahl zurück (ohne negatives Vorzeichen).

Beispiel:

```
{"abs":QUERY}
{"abs": -300}
```

Math. . . . Funktionen

Diese `Math`-Funktionen sind ebenfalls verfügbar:

```
Math.sign
Math.log
Math.log2
Math.log10
Math.sin
Math.sinh
Math.asinh
Math.cos
Math.cosh
Math.acosh
```

```
Math.atan
Math.atanh
Math.atan2
```

Beispiel:

```
{"Math.sign": -100}
{"Math.atan2": [QUERY, QUERY]}
```

clamp

`clamp` gibt eine Zahl zurück, die zwischen dem Bereich von `QUERY_MIN` und `QUERY_MAX` liegt. Liegt `QUERY_VAL` zwischen Min und Max, wird er direkt zurückgegeben.

Beispiel:

```
{"clamp": [QUERY_VAL, QUERY_MIN, QUERY_MAX]}
{"clamp": [5.5, 0, 100]}
```

scale

`scale` transformiert einen Wert aus Bereich A in den Bereich B.

Beispiel:

```
{"scale": [QUERY_A_VAL, QUERY_A_MIN, QUERY_A_MAX, QUERY_B_MIN, QUERY_B_MAX]}
{"scale": [0.05, 0, 1, 0, 100]}
```

require

`require` gibt 1 oder 0 zurück, je nachdem, ob die Bedingung `QUERY1` oder `QUERY2` wahr oder falsch ist.

Beispiel:

```
{"require": QUERY1, "eq": QUERY2}
{"require": {"var": ["L:TEST", "number"]}, "eq", 0}
```

and

`and` ist der logische UND-Operator.

`and` gibt 1 zurück, wenn jede der Abfragen 1 ist. Wenn eine Abfrage nicht 1 ist, ist das Gesamtergebnis 0.

Beispiel:

```
{"and": [
  {"require": QUERY, "eq": QUERY},
  ...
]}
```

or

`or` ist der logische ODER-Operator.

`or` gibt 1 zurück, wenn eine der Abfragen 1 zurückgibt. Es handelt sich auch um einen Abkürzungsprozess, da weitere Abfragen nicht geprüft werden, wenn eine der nachfolgenden Abfragen 1 ergibt.

Beispiel:

```
{"or": [
  {"require": QUERY, "eq": QUERY},
  ...
]}
```

not

`not` ist der logische NICHT-Operator.

`not` invertiert (1 zu 0 und 0 zu 1) jeden Query.

Beispiel:

```
{"not": QUERY}
```

typeof

`typeof` gibt eine Zeichenkette zurück, die die Art der Eingabe beschreibt, die sie erhalten hat.

| Typ | Typ Name |
|-----------|-------------|
| structs | "object" |
| null | "object" |
| arrays | "array" |
| strings | "string" |
| numbers | "number" |
| undefined | "undefined" |

isNaN

`isNaN` zeigt an, ob eine Zahl NaN ist oder nicht.

Beispiel:

```
{"isNaN": QUERY}
```

parseInt

`parseInt` wandelt eine Zeichenkette in eine ganze Zahl um

Beispiel:

```
{"parseInt": QUERY}
```

parseFloat

`parseFloat` wandelt eine Zeichenkette in einen Dezimalwert um.

Beispiel:

```
{"parseFloat": QUERY}
```

if

`if` funktioniert als QUERY oder als COMMAND. Die Syntax ist dieselbe, außer dass COMMANDLIST eine QUERY ist.

Beispiel:

```
{"if": QUERY, "then": QUERY, "else": QUERY}  
{"if": QUERY, "then": QUERY}
```

switch

`switch` funktioniert sowohl als QUERY als auch als COMMAND. COMMANDLISTs sind stattdessen eine QUERY in dieser Form.

Beispiel:

```
{"switch": QUERY, "case":{  
  "0": QUERY,  
  "1": QUERY,  
  "2": QUERY,  
  "3": QUERY,  
  "default": QUERY  
}}
```

convert

`convert` führt die Einheitenumrechnung durch.

Beispiel:

```
{"convert": QUERY, "from":"from_unit", "to":"to_unit"}
{"convert": QUERY, "from":"miles", "to":"meters"}
{"convert": QUERY, "from":"kg", "to":"lb"}
```

Sie können zwischen diesen Einheiten umrechnen:

| Gewicht | Einheiten |
|----------|--------------------|
| Kilogram | kilogram, kg, kilo |
| Pound | pound, lb |

| Längen | Einheiten |
|--------|----------------|
| Feet | feet, foot, ft |
| Meter | meter, m |
| Mile | mile, mi |

fn.HOIST_SEND_TO_GROUND

`HOIST_SEND_TO_GROUND` fährt die Winde abhängig von der Nähe zum Ziel aus und verstaut sie.

Beispiel:

```
{"HOIST_SEND_TO_GROUND", "params": {
  "target": LOCATIONREF,
  "after_deploy_commands": [
    ... commands after deploying the hoist
  ],
  "before_stow_commands": [
    ... commands before stow
  ]
}}
```

fn.HOIST_REEL_UP_AND_STOW

`HOIST_REEL_UP_AND_STOW` wartet darauf, dass die Winde aufgerollt wird, und führt die Befehle einmal aus, wenn dies geschieht:

`before_stow_commands` läuft einmal, wenn das Hebezeug verstaut ist.

Beispiel:

```
"fn":"HOIST_REEL_UP_AND_STOW", "params": {
  "before_stow_commands": [
    .. commands here to run before stow (to swap the objects)
  ]
}}
```

fn.HOIST_REEL_UP

`HOIST_REEL_UP` gibt true zurück, wenn der Aufzug hochgefahren und verstaut ist, und false, bis das passiert.

Beispiel:

```
{"fn":"HOIST_REEL_UP"}
```

fn.hoist_get_reel_distance:ft

Ermittelt die Entfernung, um die das Kabel ausgefahren wird/ist.

Beispiel:

```
{"fn":"hoist_get_reel_distance:ft"}
{"fn":"hoist_get_reel_distance:m"}
```

fn.hoist_get_distance_from_ground:ft

Ermittelt den Abstand zwischen dem Hebezeugobjekt und dem Boden.

Beispiel:

```
{"fn":"hoist_get_distance_from_ground:ft"}
{"fn":"hoist_get_distance_from_ground:m"}
```

fn.score_bambi_dump

`score_bambi_dump` gibt einen Score-Wert zurück, der aus allen Objekten berechnet wird, die mit `COUPLED=4` erstellt wurden. Die Gesamtpunktzahl ist die Summe von `VAR 2` aus jedem dieser Objekte.

Beispiel:

```
{"fn":"score_bambi_dump"}
```

fn.all_fires_extinguished

`all_fires_extinguished` zeigt an, ob es Objekte mit `COUPLED=4` gibt, die aktiv sind. Gibt 1 zurück, wenn ein Feuer aktiv ist.

Beispiel:

```
{"fn": "all_fires_extinguished"}
```

fn.has_remote_notify

Gibt 1 oder 0 zurück, je nachdem, ob die Mission von einem Server aus läuft.

Beispiel:

```
{"fn": "has_remote_notify"}
```

fn.is_voice_server_connected

`fn:is_voice_server_connected` zeigt an, ob der Sprachserver derzeit verbunden ist oder nicht.

Beispiel:

```
{"fn": "is_voice_server_connected"}
```

fn.create_guid

`fn:create_guid` erzeugt eine weltweit eindeutige Identifizierungszeichenfolge (GUID).

Beispiel:

```
{"fn": "create_guid"}
```

fn.create_date

`create_date` erzeugt ein JS-"Datums"-Objekt.

Beispiel:

```
{"set_message":{"text":"{0}", "params":[ {"fn":"create_date"} ]}},
```

fn.get_time_string

`get_time_string` liefert einen 24-Stunden-UTC-Zeitstempel wie `07:05:57`

Beispiel:

```
{"set_message":{"text":"{0}", "params":[  
  {"fn":"get_time_string"}  
]}}
```


fn.get_mission_objects

`get_mission_objects` liefert die aktiven Missionsobjekte (zur Aufzählung).

Beispiel:

```
{ "set_message": { "text": "{0}", "params": [
  { "json": stringify: { "fn": "get_mission_objects" } }
] },
```

fn.get_aircraft_moniker

`fn.get_aircraft_moniker` liefert einen String-Identifikator für das Luftfahrzeug, z. B. H145 oder H160.

Beispiel:

```
{ "#comment": "copy either H145 or H160 into a string local named HXX"},
{ "set": { "local": "HXX", "value": { "fn": "get_aircraft_moniker" } },
{ "#comment": "fire an event where the name is either H145 or H160 but otherwise is the same..."},
{ "trigger": "H:{local:HXX}_SDK_DO_RANDOM_THING"},
{ "#comment": "just show a message, but note that you can now use {local:HXX} in any string across the mission"},
{ "set_message": "hello from {local:HXX}" },
```

fn.is_any_sling_object_coupled

`fn.is_any_sling_object_coupled` gibt einen booleschen Wert zurück, der angibt, ob ein Objekt gerade gekoppelt ist.

fn.get_sling_object_type

`fn.get_sling_object_type` gibt einen Wert von 1 bis 10 zurück, der die Art des derzeit gekoppelten Hebegurtoobjekts angibt.

fn.get_mission_icons

`fn.get_mission_icons` erhält die gesamte Tabelle der Icons.

fn.create_multiplayer_connection

`fn.create_multiplayer_connection` erstellt eine `MPCClient`-Multiplayer-Verbindung.

API Referenz - LOCATION

Standorte sind ein grundlegendes Konzept innerhalb der Missionsplattform.

LOCATIONREF

Ein **LOCATIONREF** ist eines der folgenden Elemente:

1. Eine Zeichenfolge, die auf ein Element in der Tabelle **locations** oder ein Objekt in der Tabelle **objects** verweist.
2. Ein Array wie **[34.29, -122.4]** oder **[34.29, -122.4, 90]**. Letztere 90 ist eine Überschrift, falls vorhanden.
3. Eine spezielle Ortsangabe wie **\$USER**.
4. Ein Peilungs-/Distanz-Befehl
5. ein **nächstgelegener** Punkt

Beispiel:

```
"my_location"  
"object_name"  
"$USER"  
[34.29, -122.4]  
[34.29, -122.4, 90]  
{"bearing": 100, "dist": 100}  
{"bearing2": 100, "dist": 100}  
{"location_alter": ...}  
{"closest": ...}
```

bearing

bearing berechnet eine Peilung auf der Grundlage einer Vorgabe.

- **dist**: Meter
- **heading**: Optional, Default ist 0

Beispiel:

```
{  
  "bearing": 100,  
  "object": "$USER",  
  "heading": 0,  
  "dist": 100  
}
```

bearing2

`bearing2` berechnet eine Peilung, ohne den Kurs des Flugzeugs zu berücksichtigen.

- `dist`: Meter
- `heading`: Optional, Default ist 0

Beispiel:

```
{
  "bearing2": 100,
  "object": "$USER",
  "heading": 0,
  "dist": 100
}
```

location_alter

`location_alter` erstellt einen Ortsbezug mit einer geänderten Bezeichnung.

Beispiel:

```
{"location_alter": "$USER", "hdg": 0}
```

closest

`closest` wählt den nächstgelegenen Ort zu `to` aus. `to` ist standardmäßig `$USER`, wenn nicht angegeben.

Beispiel:

```
{"closest": [LOCATIONREF, LOCATIONREF, ...]}
{"closest": [LOCATIONREF, LOCATIONREF, ...], "to": LOCATIONREF}
```

Besondere Standorte

- `$USER`: entspricht dem `[lat,lon]` des Benutzerflugzeugs
- `$USER:HOIST`: entspricht `[lat,lon]` der Hubposition des Benutzerflugzeugs.
- `$MISSION_START_LOCATION`: ergibt den `[lat,lon]`, der der Startpunkt auf der Karte war.
- `$MISSION_SELECTED_POI_LOCATION`: ergibt sich aus dem `[lat,lon]`, den der Benutzer gerade auf der Missionskarte ausgewählt hat.

Beispiele

Konvertierte Funktion aus JS Beispiel

Bei dieser in JS geschriebenen Funktion:

```
function polarToCartesian(radius, angleInDegrees) {
  let angleInRadians = (angleInDegrees-90) * Math.PI / 180.0;
  return {
    x: radius * Math.cos(angleInRadians),
    y: radius * Math.sin(angleInRadians)
  };
}
```

Die gleiche Funktion, geschrieben als Makro, lautet wie folgt:

```
"polarToCartesian": [
  {"#comment": [
    "param - radius",
    "param - angleInDegrees"
  ]},
  {"set": {"param": "angleInDegrees", "value": {"multiply": [ {"subtract":
[ {"param": "angleInDegrees"}, -90 ]}, {"divide": [3.14159 / 180]} ]}},
  {"return": {"create_struct": {
    "x": {"multiply": [ {"param": "radius"}, {"Math.cos": {"param": "angleInDegrees"}} ]}},
    "y": {"multiply": [ {"param": "radius"}, {"Math.sin": {"param": "angleInDegrees"}} ]}},
  }}}
]
```

Beispiel für die Erkennung von Szenerien

Verwenden Sie `fetch`, um das VFS nach Inhalten abzufragen, die spezifisch für das Paket sind, das Sie erkennen möchten.

```
{
  "title": "Scenery detection Sample",
  "api_version": 0.1,
  "aircraft": ["H145"],
  "briefing": [
    {"image": "/VFS/ContentInfo/revelstoke-logging1-scenery/Thumbnail.jpg", "show_condition":
{"require": {"local": "SCENERY_INSTALLED_1"}, "eq": 1}},
    {"text": "Installed: {local:SCENERY_INSTALLED_1}"},
    {"image": "/VFS/ContentInfo/revelstoke-mill1-scenery/Thumbnail.jpg", "show_condition":
{"require": {"local": "SCENERY_INSTALLED_2"}, "eq": 1}},
    {"text": "Installed: {local:SCENERY_INSTALLED_2}"},
    {"image": "/VFS/ContentInfo/revelstoke-lakeview-scenery/Thumbnail.jpg", "show_condition":
{"require": {"local": "SCENERY_INSTALLED_3"}, "eq": 1}},
    {"text": "Installed: {local:SCENERY_INSTALLED_3}"},
  ],
  "objectives": [
    {
      "title": "Done",
      "commands": [
        {"fetch": {"url": "/VFS/ContentInfo/revelstoke-logging1-scenery/Thumbnail.jpg"}},
        {"set": {"local": "SCENERY_INSTALLED_1", "value": {"if":
{"param": "$FETCH_STATUS", "eq": 200, "then": 1, "else": 0}}},

```

```

        {"fetch":{"url":"/VFS/ContentInfo/revelstoke-mill1-scenery/Thumbnail.jpg"}},
        {"set":{"local":"SCENERY_INSTALLED_2"},"value":{"if":
{"param":"$FETCH_STATUS"},"eq":200,"then":1,"else":0}},
        {"fetch":{"url":"/VFS/ContentInfo/revelstoke-lakeview-scenery/Thumbnail.jpg"}},
        {"set":{"local":"SCENERY_INSTALLED_3"},"value":{"if":
{"param":"$FETCH_STATUS"},"eq":200,"then":1,"else":0}},

        {"sleep": "forever"}
    ]
}
]
}

```

Zufälliges Element aus der statischen Liste holen

Diese statischen Daten sind gegeben:

```

"data": {
  "cars": [
    "Car Title 1",
    "Car Title 2",
    "Car Title 3",
    "Car Title 4"
  ]
},

```

Jeder Aufruf wird ein zufälliges Auto liefern:

```

{"struct": {"static": "cars"}, "index": {"floor": {"rand": [0, {"struct":
{"static":"cars"},"path":"length"} ]}}}

```

Beispiel:

```

{
  "title": "test",
  "api_version": 0.1,
  "aircraft": ["H145"],
  "data": {
    "cars": [
      "Car Title 1",
      "Car Title 2",
      "Car Title 3",
      "Car Title 4"
    ]
  },
  "briefing": [
    {"text": "Your selected car: {0}", "params": [
      {"local": "selectedCar"}
    ]},
    {"buttonbar": [
      {"title": "Pick a new car", "commands": [
        {"set": {"local": "selectedCar"}, "value": {"struct": {"static": "cars"}, "index": {"floor":
{"rand": [0, {"struct": {"static": "cars"}, "path": "length"} ]}}}
      ]}
    ]}
  ],
  "objectives": [
    {

```

```

    "title": "Done",
    "commands": [
      {"sleep": "forever"}
    ]
  }
}
}
}

```

CARLS Radio Test Program

Dieses Programm sendet Informationen an das Radio und verarbeitet auch die Ereignisse für das Anklicken der Schaltflächen.

```

{
  "title": "Radio test program",
  "api_version": 0.1,
  "aircraft": ["H145"],
  "macros": {
    "render": [
      {"if": {"var": ["L:MY_PAGE", "number"]}, "eq":0, "then": [
        {"set_carls_radio": {
          "LSK": ["PG1", "", ""],
          "RSK": ["", "", "INOP"],
          "Items": [
            ["Group 1", "misc contacts"],
            ["Group 2", "important"],
            ["Group 3", "other"]
          ]
        }
      ]
    }
  ]},
  {"if": {"var": ["L:MY_PAGE", "number"]}, "eq":1, "then": [
    {"set_carls_radio": {
      "LSK": ["PG2", "", ""],
      "RSK": ["", "", "INOP"],
      "Items": [
        ["Contact 1", "000-5555-1234"],
        ["Contact 2", ""],
        ["Contact 3", ""]
      ]
    }
  ]
}
]
},
"objectives": [
  {
    "title": "Initializing...",
    "commands": [
      {"#comment": "select keys"},
      {"create_event_handler": "MISSION_RADIO_CARLS_L1", "commands": [{"set_message": {"text":
"LSK1" }}}},
      {"create_event_handler": "MISSION_RADIO_CARLS_L2", "commands": [{"set_message": {"text":
"LSK2" }}}},
      {"create_event_handler": "MISSION_RADIO_CARLS_L3", "commands": [{"set_message": {"text":
"LSK3" }}}},
      {"create_event_handler": "MISSION_RADIO_CARLS_R1", "commands": [{"set_message": {"text":
"RSK1" }}}},
      {"create_event_handler": "MISSION_RADIO_CARLS_R2", "commands": [{"set_message": {"text":
"RSK2" }}}},
      {"create_event_handler": "MISSION_RADIO_CARLS_R3", "commands": [{"set_message": {"text":
"RSK3" }}}},
      {"#comment": "dial pad"},
      {"create_event_handler": "MISSION_RADIO_CARLS_0", "commands": [{"set_message": {"text":
"Num 0" }}}},
      {"create_event_handler": "MISSION_RADIO_CARLS_1", "commands": [{"set_message": {"text":
"Num 1" }}}},

```

```

"Num 2"  {"create_event_handler": "MISSION_RADIO_CARLS_2", "commands": [{"set_message": {"text":
}}]},
{"create_event_handler": "MISSION_RADIO_CARLS_3", "commands": [{"set_message": {"text":
"Num 3" }}]},
{"create_event_handler": "MISSION_RADIO_CARLS_4", "commands": [{"set_message": {"text":
"Num 4" }}]},
{"create_event_handler": "MISSION_RADIO_CARLS_5", "commands": [{"set_message": {"text":
"Num 5" }}]},
{"create_event_handler": "MISSION_RADIO_CARLS_6", "commands": [{"set_message": {"text":
"Num 6" }}]},
{"create_event_handler": "MISSION_RADIO_CARLS_7", "commands": [{"set_message": {"text":
"Num 7" }}]},
{"create_event_handler": "MISSION_RADIO_CARLS_8", "commands": [{"set_message": {"text":
"Num 8" }}]},
{"create_event_handler": "MISSION_RADIO_CARLS_9", "commands": [{"set_message": {"text":
"Num 9" }}]},
{"create_event_handler": "MISSION_RADIO_CARLS_STAR", "commands": [{"set_message": {"text":
"Num *"}]}]},
{"create_event_handler": "MISSION_RADIO_CARLS_SHARP", "commands": [{"set_message": {"text":
"Num #"}]}]},
{"#comment": "phone keys"},
{"create_event_handler": "MISSION_RADIO_CARLS_PICK", "commands": [{"set_message": {"text":
"PICK"}]}]},
{"create_event_handler": "MISSION_RADIO_CARLS_HANG", "commands": [{"set_message": {"text":
"HANG"}]}]},
{"create_event_handler": "MISSION_RADIO_CARLS_WARNING", "commands": [{"set_message":
{"text": "WARNING"}]}]},

{"#comment": "change page when using <- and -> arrows"},
{"create_event_handler": "MISSION_RADIO_CARLS_LEFT", "commands": [
{"set": {"var": ["L:MY_PAGE", "number"], "value": 0},
{"sleep": 0.2},
{"call_macro": "render"}
]},
{"create_event_handler": "MISSION_RADIO_CARLS_RIGHT", "commands": [
{"set": {"var": ["L:MY_PAGE", "number"], "value": 1},
{"sleep": 0.2},
{"call_macro": "render"}
]},

{"#comment": "Use L:CARLS_LIST_SELECTED_INDEX to get the highlighted list item !!!!"},

{"call_macro": "render"}
]
},
{
"title": "Done",
"commands": [
{"sleep": "forever"}
]
}
]
}

```

Testprogramm für Remote Dispatcher

HEMS- Dispatcher (nur Multiplayer)

1. Jeder Benutzer stellt eine Verbindung zum Server her und lädt eine Liste "gültiger Missionsoptionen" hoch.
2. Der Operator kann Flugzeuge abfertigen und die Flugzeuge können die Abfertigung annehmen.

```
{
"title": "Multiplayer Dispatch Test Program V2",
```



```

    "4": {"button": "View", "commands": [ {"set_map_center": {"param": "location"}, "zoom":
16} ]}
    },
    "row1": {
      "1": {"text": "{0}", "params": [ {"struct": {"static": "statusMessages"}, "index":
{"param": "Status"}} ]}
    }
  }
},
"incomingMessageList": {
  "type": "list",
  "source": {"static": "messagesToDispatcher"},
  "title": "Message Inbox",
  "emptyText": "No messages right now",
  "rows": {
    "row0": {
      "1": {"text": "{UserName}"},
      "2": {"button": "Delete", "commands": [ {"set_shared_data": "delete",
"path": "messagesToDispatcher.{id}"} ]}
    },
    "row1": {
      "1": {"text": "{Text}"}
    }
  }
},
"outgoingMessageList": {
  "type": "list",
  "source": {"static": "messagesToAircraft"},
  "title": "Recent Dispatches",
  "emptyText": "No messages right now",
  "rows": {
    "row0": {
      "1": {"text": "{from} {to} "},
      "2": {"text": "{mission}"},
      "3": {"button": "Delete", "commands": [ {"set_shared_data": "delete",
"path": "messagesToAircraft.{id}"} ]},
      "4": {"button": "View", "commands": [ {"set_map_center": {"param": "location"}, "zoom":
16} ]}
    },
    "row1": {
      "1": {"text": "{message}"}
    }
  }
}
},
"statusMessages": [
  "0. Dispatch accepted",
  "1. On the way to the scene",
  "2. At the scene",
  "3. On the way to the hospital",
  "4. At the hospital",
  "5. On the way back to base (Available)",
  "6. At Home base (Available)",
  "7. Unavailable for dispatch"
],
"missionList1": {
  "0. Road Accident": 0,
  "1. Motorcycle Crash": 1,
  "2. Tipped over tractor": 2
},
"missionList2": {
  "0. Hospital": 0,
  "1. Meet Ambulance": 1
}
},
"briefing": [
  {"#comment": [
    "MP_MODE ... 0: not set, 1: offline, 2: online"
  ]},
  {"title": "Mission Initial Setup", "show_condition": {"require": {"local": "MP_MODE"}, "eq": 0}},
  {"buttonbar": [
    {"title": "Offline (Single player)", "commands": [ {"set": {"local": "MP_MODE"}, "value": 1} ],

```

```

"disabled_condition":{"require":1,"eq":1}},
  {"title":"Online (Multiplayer)", "commands": [ {"call_macro":"mp_open_login_dialog"} ]
  }, "show_condition": {"require":{"local":"MP_MODE"},"eq": 0}},

  {"title":"Multiplayer (Online)", "show_condition": {"require":{"local":"MP_MODE"},"eq": 2}},
  {"buttonbar":[
    {"title":"View Multiplayer Status", "commands": [ {"call_macro":"mp_open_login_dialog"} ]
    }, "show_condition": {"require":{"local":"MP_MODE"},"eq": 2}},

    {"title":"My Status", "show_condition": {"require":{"local":"MP_MODE"},"ne":0}},
    {"text":"My status: {0}", "params": [ {"var":["L:MY_DISPATCH_STATUS","number"]} ],
    "show_condition": {"require":{"local":"MP_MODE"},"ne":0}},
    {"text":"{0}", "params": [
      {"struct":{"static":"statusMessages"},"index":{"var":["L:MY_DISPATCH_STATUS","number"]}}
    ], "show_condition": {"require":{"local":"MP_MODE"},"ne":0}},
    {"text":"Change my status:", "show_condition": {"require":{"local":"MP_MODE"},"ne":0}},
    {"buttonbar":[
      { "title":"1", "commands":[ {"set":{"var":["L:MY_DISPATCH_STATUS","number"],"value":1}} ] },
      { "title":"2", "commands":[ {"set":{"var":["L:MY_DISPATCH_STATUS","number"],"value":2}} ] },
      { "title":"3", "commands":[ {"set":{"var":["L:MY_DISPATCH_STATUS","number"],"value":3}} ] },
      { "title":"4", "commands":[ {"set":{"var":["L:MY_DISPATCH_STATUS","number"],"value":4}} ] },
      { "title":"5", "commands":[ {"set":{"var":["L:MY_DISPATCH_STATUS","number"],"value":5}} ] },
      { "title":"6", "commands":[ {"set":{"var":["L:MY_DISPATCH_STATUS","number"],"value":6}} ] }
    ],
    "show_condition": {"require":{"local":"MP_MODE"},"ne":0}},

    {"buttonbar":[
      { "title":"Send message to dispatcher", "commands":
      [ {"call_macro":"open_dispatcher_msg_dialog"} ]
      },
      "show_condition": {"require":{"local":"MP_MODE"},"ne":0}},
      {"text":"Change my available missions for dispatch:", "show_condition": {"require":
{"local":"MP_MODE"},"ne":0}},
      {"buttonbar":[
        {
          "title":"Set Mission Set 1 (rescue)",
          "commands":[
            {"set_shared_data":"update",
              "path":"connectedAircraft.{local:service_auth}.MissionList",
              "value": {"static": "missionList1"}
            },
            {"set":{"local":"ACTIVE_MISSION_SET"},"value":1}
          ],
          "select_condition": {"require":{"local":"ACTIVE_MISSION_SET"},"eq":1}
        },
        {
          "title":"Set Mission Set 2 (hospital etc.)",
          "commands":[
            {"set_shared_data":"update",
              "path":"connectedAircraft.{local:service_auth}.MissionList",
              "value": {"static": "missionList2"}
            },
            {"set":{"local":"ACTIVE_MISSION_SET"},"value":2}
          ],
          "select_condition": {"require":{"local":"ACTIVE_MISSION_SET"},"eq":2}
        }
      ],
      "show_condition": {"require":{"local":"MP_MODE"},"ne":0}},

      {"title":"Incoming Dispatch"},

      {"text":"Dispatcher Name: {local:DISPATCH_FROM}"},
      {"text":"Selected Mission: {local:DISPATCH_MISSION}"},
      {"text":"Location: {local:DISPATCH_LOCATION}"},
      {"text":"Text Message: {local:DISPATCH_MESSAGE}"},
      {"text":"Patient Life Score: {local:DISPATCH_LIFESCORE}"
    ],
    "events": {
      "ON_MISSION_ABORTING": {
        "commands": [ {"call_macro":"mp_aborting_mission"} ]
      }
    },
    "macros":{
      "open_dispatcher_msg_dialog": [

```

```

{"set_dispatch":[
  {"title":"Send message"},
  {"textbox":"mp_dispatcher_msg"},
  {"buttonbar":[
    {"title":"Send Message to dispatcher", "commands": [
      {"set":{"param":"id"}, "value": {"fn": "create_guid"}},
      {"set_shared_data":"update",
        "path":"messagesToDispatcher.{id}",
        "value": {"create_struct":{"
          "Text":{"local":"mp_dispatcher_msg"},
          "UserName": {"local":"mp_userName"}
        }}}},
      {"set_briefing_dialog":1}
    ]}
  ]}
]},
{"set_dispatch_dialog":1}
],
"mp_open_login_dialog":[
  {"#comment": "Show the login dialog dispatch (or multiplayer status)"},
  {"set_dispatch":[
    {"buttonbar":[ {"title":"<- Back to briefing", "commands":
[{"set_briefing_dialog":1} ]} ]}],
    {"title":"Log in", "show_condition": {"require":{"local":"MP_MODE"}, "eq": 0}},
    {"text":"You are playing offline.", "show_condition": {"require":{"local":"MP_MODE"}, "eq":
1}},
    {"text":{"text":"User Id: {0}", "params":{"local":"service_auth"}}, "show_condition":
{"require":{"local":"MP_MODE"}, "eq": 0}},
    {"text":"User Name:", "show_condition": {"require":{"local":"MP_MODE"}, "eq": 0}},
    {"textbox":"mp_userName", "show_condition": {"require":{"local":"MP_MODE"}, "eq": 0}},
    {"text":"Room:", "show_condition": {"require":{"local":"MP_MODE"}, "eq": 0}},
    {"textbox":"mp_room", "show_condition": {"require":{"local":"MP_MODE"}, "eq": 0}},
    {"text":"Password:", "show_condition": {"require":{"local":"MP_MODE"}, "eq": 0}},
    {"textbox":"mp_password", "show_condition": {"require":{"local":"MP_MODE"}, "eq": 0}},
    {"buttonbar":[
      {"title":"Create Room (Opens on PC)", "commands":
[ {"open_url":{"static:create_room_url}?room={local:mp_room}"} ]},
      {"title":"Log In", "commands": [ {"call_macro":"mp_login"} ]}
    ]},
    {"disabled_condition":{"require":{"struct":{"local":"MP_CONN"},
"path":"Status"},"eq":"Connected"},
"show_condition": {"require":{"local":"MP_MODE"}, "eq": 0}},
    {"text":{"text":"MP Connection Status: {0}", "params":[
      {"struct":{"local":"MP_CONN"}, "path":"Status"}
    ]}, "show_condition": {"require":{"local":"MP_MODE"}, "ne": 1}},
    {"text":{"text":"MP Server Last Error: {local:MP_LAST_ERROR}"}, "show_condition":
{"require":{"local":"MP_MODE"}, "ne": 1}},
    {"title":"Debug Info"},
    {"text":{"text":"Multiplayer Mode: {0}", "params":{
      "switch":{"local":"MP_MODE"}, "case":{
        "0": "Undecided",
        "1": "Offline, Singleplayer",
        "2": "Multiplayer"
      }
    }}}
  ]}],
  {"#comment":{"text":"Debug MP Message: {local:MP_MSG}"}, "show_condition": {"require":
{"local":"MP_MODE"}, "ne": 1}
]},
{"set_dispatch_dialog":1}
],
"mp_login":[
  {"#comment":"try to make the actual connection to the server"},
  {"set":{"param":"service_auth"}, "value":{"local":"service_auth"}},
  {"set":{"local":"MP_LAST_ERROR"}, "value":""},
  {"set":{"local":"MP_CONN"}, "value": {"fn": "create_multiplayer_connection"}},
  {"set":{"local":"MP_CONN", "path":"OnError"}, "value":{"js:create_async_function":[
    {"set":{"local":"MP_LAST_ERROR"}, "value":{"struct": {"param":"$args"}, "index": 0}

```

```

    ]}},
    {"set":{"local":"MP_CONN", "path":"OnMessage"}, "value":{"js:create_async_function":[
      {"set":{"param":"arg0"},"value":{"struct":{"param":"$args"}, "index": 0}},
      {"call_macro":"mp_on_message", "params":{"msg":{"param":"arg0"}}}
    ]}},
    {"set":{"param":"unused"},"value":{"struct":{"local":"MP_CONN"}, "function":"Connect",
"params":[
  {"static":"server_url"}, {"param":"service_auth"}, {"local":"mp_room"},
{"local":"mp_password"}
]}}},
    {"create_thread":{"commands":[
      {"wait_for":{"struct":{"local":"MP_CONN"}, "path":"Status"},"eq":"Connected"},
      {"#comment":"once we log in once, we're committed to multiplayer"},
      {"set":{"local":"MP_MODE"}, "value": 2},
      {"set_briefing_dialog":1},
      {"#comment":"First create terminationCommands with no_overwrite, then add an entry for us,
and then populate with commands to clear us from connectedAircraft and terminationCommands when we
become stale on the server"},
      {"set_shared_data":"update",
        "path":"terminationCommands",
        "policy":"no_overwrite",
        "value":{"create_struct":{}}
      },
      {"set_shared_data":"update",
        "path":"terminationCommands.{service_auth}",
        "value":{"create_struct":{"removeFromConnectedAircraft":{"create_struct":{"type":"delete",
"path":"connectedAircraft.{service_auth}"
}},
        "removeFromFlightPlans":{"create_struct":{"type":"delete",
"path":"flightPlans.{service_auth}"
}},
        "removeFromTerminationCommands":{"create_struct":{"type":"delete",
"path":"terminationCommands.{service_auth}"
}}
      }}
    ]}},
    {"#comment":"make sure we have connectedAircraft table. all players must use no_overwrite
when ensuring the table exists to prevent anybody from destroying the table."},
    {"set_shared_data":"update", "path":"connectedAircraft", "policy":"no_overwrite", "value":
{"create_struct":{}} },
    {"set_shared_data":"update", "path":"messagesToDispatcher", "value":{"create_struct":
{}} },
    {"set_shared_data":"update", "path":"messagesToAircraft", "value":{"create_struct":{}} },
    {"set":{"param":"unused"},"value":{"struct":{"local":"MP_CONN"}, "function":"Subscribe",
"params":["messagesToAircraft"]}},
    {"set_shared_data":"update", "path":"icons", "policy":"no_overwrite", "value":
{"fn":"get_mission_icons"} },
    {"set_shared_data":"update", "path":"flightPlans", "policy":"no_overwrite", "value":
{"create_struct":{}} },
    {"set_shared_data":"update", "path":"webConfig", "policy":"no_overwrite", "value":
{"static":"webConfig"} },
    {"set_shared_data":"update", "path":"statusMessages", "policy":"no_overwrite", "value":
{"static":"statusMessages"} },
    {"set_shared_data":"update",
      "path":"connectedAircraft.{service_auth}",
      "value":{"create_struct":{"location":{"resolve_location":"$USER"},
        "UserName":{"local":"mp_userName"},
        "Status": 0,
        "MissionList":{"static":"missionList1"}
      }}
    },
    {"set":{"local":"ACTIVE_MISSION_SET"}, "value":1},

```

```

        {"#comment":"update our location, score and flightplan (if changed) forever"},
        {"while":1,"eq":1,"do":[
            {"sleep":5},
            {"set_shared_data":"update", "path":"connectedAircraft.{service_auth}.location", "value":
{"resolve_location":"$USER"} },
            {"set_shared_data":"update", "path":"connectedAircraft.{service_auth}.Status", "value":
{"var":["L:MY_DISPATCH_STATUS","number"]} }},

            {"if":{"json:stringify": {"local":"$FLIGHTPLAN"}}, "ne": {"param":"FPL"},"then":[
                {"set":{"param":"FPL"},"value":{"json:stringify": {"local":"$FLIGHTPLAN"}},
                {"set_shared_data":"update", "path":"flightPlans.{service_auth}", "value":
{"create_struct":{"
                    "points":{"local":"$FLIGHTPLAN"}
                }}}
            ]}
        ]}
    ]},
    "mp_initialize":[
        {"#comment":"set up for multiplayer operations later"},
        {"set":{"local":"MP_LAST_ERROR"},"value":""},
        {"set":{"local":"MP_MODE"},"value":0},
        {"#comment":"MP_MODE 0: undecided, 1: offline, 2:online"},

        {"#comment":"these are for debugging only"},
        {"set":{"local":"MP_MSG"},"value":""},
        {"set":{"local":"mp_room"},"value":""},
        {"set":{"local":"mp_password"},"value":""},
        {"set":{"local":"mp_userName"},"value":{"var":["ATC AIRLINE","string"]}},
        {"#comment": "Create or access a unique ID to identify you on the server irrespective of
callsign"},
        {"set":{"local":"service_auth"}, "value":{"fn":"create_guid"}},

        {"create_thread":{"commands":[
            {"wait_for":{"local":"MP_MODE"},"ne":0},
            {"call_macro":"mp_begin"}
        ]}}
    ],
    "mp_on_message":[
        {"#comment":"param - msg"},

        {"#comment":"handle READ, UPDATE and DELETE operations below"},
        {"set":{"param":"json"},"value":{"json:stringify":{"param":"msg"}}},
        {"switch":{"struct": {"param":"msg"}, "path": "type"}, "case": {
            "read":[
                {"set":{"local":"MP_MSG"},"value": "we got an read: {json}"
            },
            "update": [
                {"set":{"local":"MP_MSG"},"value": "we got an update: {json}"},

                {"#comment":"split the path into parts based on ."},
                {"set":{"param":"parts"}, "value":{"string:split":{"struct": {"param":"msg"}, "path":
"path"}, "delimiter":"."}},

                {"#comment":"messagesToAircraft NEW MESSAGE"},
                {"if": {"and": [
                    {"require": {"struct": {"param":"parts"}, "path":"length"}, "eq": 2},
                    {"require": {"struct": {"param":"parts"}, "index":"0"}, "eq": "messagesToAircraft"},
                    {"require": {"struct": {"param":"parts"}, "index":"1"}, "eq": "{local:service_auth}"
                ]}, "eq":1, "then":[
                    {"call_macro":"on_got_dispatch","params":{"
                        "dispatchInfo": {"struct": {"param":"msg"}, "path": "value"}
                    }}
                ]}
            ],
            "delete": [
                {"set":{"local":"MP_MSG"},"value": "we got an delete: {json}"
            }
        ]}
    ],
    "mp_begin":[

```

```

    {"#comment":"called once we decided if we are single or multiplayer. MP_MODE 1:offline,
2:online"},
    {"#comment":"offline case, manually run the logic and complete logic"},

    {"call_macro":"Update_RescueTrack"},
    {"set_objective_title":"Ready to play the game!"}
],
"mp_aborting_mission":[
    {"#comment":"we want to clean up our multiplayer connection if it was created"},
    {"if":{"local":"MP_CONN"},"ne":null, "then":[
        {"set":{"param":"unused"},"value":{"struct":{"local":"MP_CONN"}, "function":"Close",
"params":[]}}
    ]}
],
"on_got_dispatch":[
    {"#comment":"param - dispatchInfo"},
    {"#comment":"this macro is called when the dispatch is received from the web"},

    {"set":{"local":"DISPATCH_FROM"}, "value":{"struct":{"param":"dispatchInfo"}, "path":
"from"}},
    {"set":{"local":"DISPATCH_MISSION"}, "value":{"struct":{"param":"dispatchInfo"}, "path":
"mission"}},
    {"set":{"local":"DISPATCH_LOCATION"}, "value":{"struct":{"param":"dispatchInfo"}, "path":
"location"}},
    {"set":{"local":"DISPATCH_MESSAGE"}, "value":{"struct":{"param":"dispatchInfo"}, "path":
"message"}},
    {"set":{"local":"DISPATCH_LIFESCORE"}, "value":{"struct":{"param":"dispatchInfo"}, "path":
"lifeScore"}},

    {"modify_array":{"local":"Messages"},"append":{"create_struct":{"
    "from":{"local":"DISPATCH_FROM"},
    "time":{"fn":"get_time_string"},
    "text":{"local":"DISPATCH_MESSAGE"},
    "mission":{"local":"DISPATCH_MISSION"},
    "waypoint":{"local":"DISPATCH_LOCATION"}
    }}}},
    {"call_macro":"Update_RescueTrack"},

    {"#comment":""}
],
"Update_RescueTrack":[
    {"set_rescuetrack":{"
        "statusVar": "L:MY_DISPATCH_STATUS",
        "statusMessages":{"static":"statusMessages"},
        "dispatcherMessages":{"local":"Messages"},
        "activate_waypoint_commands":[
            {"#comment":"param - $index - in dispatcherMessages"},
            {"#comment":"param - $command - DIRECT-TO"},

            {"#comment":"below we set a nav line to the location, and we can select the type of
mission scene to spawn there"},
            {"set_route":{"struct":{"struct":{"local":"Messages"}, "index":{"param":"$index"}},
"path":"waypoint"}},
            {"switch":{"struct":{"struct":{"local":"Messages"}, "index":{"param":"$index"}},
"path":"mission"}, "case":{"
                "0. Road Accident": [ {"#comment":"TODO: Set up for 0. Road Accident"} ],
                "1. Motorcycle Crash": [ {"#comment":"TODO: Set up for 1. Motorcycle Crash"} ],
                "2. Tipped over tractor": [ {"#comment":"TODO: Set up for 2. Tipped over tractor"} ],
                "0. Hospital": [ {"#comment":"TODO: Set up for 0. Hospital"} ],
                "1. Meet Ambulance": [ {"#comment":"TODO: Set up for 1. Meet Ambulance"} ],
                "default": [ {"#comment":"TODO: Set up for unknown mission"} ]
            }
        }
    ]}
]}
],
"objectives": [
    {
        "title": "Setup required",
        "commands": [
            {"set":{"local":"Messages"},"value":[]},
            {"call_macro":"mp_initialize"},
            {"sleep":"forever"}
        ]
    }
]

```

```
}  
],  
"icons":{
```

```
"wp_blue": "data:image/png;base64,iVBORw0KGGoAAAANSUHEUGAAFAAAAABQCAYAAAH5F5I7AAAAAXNSR0IArs4c6QAAAA  
RnQU1BAACxjwv8YQUAAAAJcEhZcwAADsIAAA7CARUoSoAAAA6GSURBVGhd7ZsJWBRHFOcr55IBhvtStg3gGjWJGrJqv0IaY7Kaj  
Uk0EjUE6EYNJqioWRUUVQxBEFFS8QP2UYzRkKeigIEVEEQDPH7ZmYmnoU3u6emuEadEbaPxc//  
+9r+r2qpuf1q+t1VTVQR1BE9GIovoQ0DIOSCGY8d2G/  
sCAEiq+TwxFXJxE2nYpJmQ6TUUCV8PS40ptfE6vvLKZC8SX892zqjF0N2g7F14Dbh+SLNChrQxRgIgoOodICQ2T8gVATDCCIFpXen  
VAXntmLrMnj8t5wCuPR8qHfS4cJm04kmrhZL6M6HfW2JV22S6c/TYXCPAbIUaFUDrTsu/  
9SD1PqN8XA0eu8sj+wUGWlrX1onJf1Qe6e3YuYJzJkT+d7Lqwi55aTJs9Z+195cvd3ux60KiH/  
ma7AbVPwBf5ASTGZ1uxqj1DsdBLsdPIDk0hbtL5/AI0Kmw8GpN/Dz67MpkU+sqGDRuYmlnHVTe0NF3VyhRBPkMjy1dugRqVIA/  
7qqUinSoUQDhVzSpRj7UsNoQM0H5xWh0Qx2bnj1pb7z0hnPmzAG74xMmAkBRrdkewxBPnL9Kq64kE/pMmpg07cCB/  
XreAYHDYTRNCY56FAhymLQ9CnEYiZoVDZkgV9g1j1PVij2LSp0JcyGdLliSVJLj6uqq8Q1ie3+InWFDubzjHpkEjt61N8jUtqa+  
vfMPwax0Py7BPFNgby9ZsliLaqfKcF6Y0Kd+uzbU3f0HnPNWF8LicjTZ3NB04xMoILjsZnfd3n0voI8g+0LBmjKVExoCKWx03k  
pcVug8GI0vuHesqJDU0w7xg0odSwsDa73K0sorQ2UkyYna1f477e79RevDrWRL/Qe21oIJJNMMkLn3K/  
Q96j1841kgcba2F0kkFcZkGr3IQvHiFWNHws/YEJ708vLiwRyagBvLkVx8Wym7pgtLSZkOhJy+NEVZ/0jy4UBLF/  
RRoWJQhhyOfri53wycOXMMoHLhyjif+OdReqqANwyPeXBjnMtFzFASBYSBxJHfhdLU8dv2Xc/  
w2ELtbQzNYbsCpbfq8S0H8QeLq0m6nUvDVuvAcxpkVRRNS0PKUmK20fn54/CHI2YPXs2cF2Z2LS6sf0u/W/  
pTgSt1cNTWYrM7jBbgLlDu9z/XqGbbq9auXIFa8wn33yebj10y3EH3ihx0RdPVRNns1qByWCxfFM6/9+M4kt/tcuWzSectXh4/  
KjrMuuf824aWg88acLxltUQ91XTRbKawEAbuXRjzVvX/  
ap032WL5pfAnFdn7ty5IPziZFEURqJhF3HVzxtddyT5K1badDdvFUpaiHrGyToZsmZznq/  
ExwcpEsY9P33yxVBbUYz3kDa6wMgCW7g/  
LS6IqXscackkZTpArOkjuxWdkT88Q9aFa+SQ5cTFHkGcWtF9BpFukAYhh+2NSIxT0kzLIVbChRTLD9FNHYiBboRHJXpT0bVFKLZ  
xIUg7LSy9e0QC4PHawkUTjhrCL7/  
vknV0CNhuAGIvmNlBYxZY0k+HQkuTaFiagdeiKUGQ52h0K9ISHoh1TXRRcMqFR3AEIYCr658CDow+TAGVQ2FGzGzVdGxkTyhw  
nbIGTaAVetIGxT8KI5mAT9nkECUEdAK5QL5cYHkJvwIqAfWlXApLhgJCo9Juc5y6tXg/  
SGFDoerJE81sNkARLMOX3oMiJVBi4DkznSHK1YphZY/IBapFixZS03PWJ/AIhLks6ChwvIjDlt/  
809T6BEWVGQGLq9ueQoUkw2qoNxt7BGwtbXxd4XYokhyz446BYv/  
iuyvAvEerfdCct2QZS1ZkZzS3WJ8McpjMxH3IfnEpGXP8ddMIJIp0iWYTjUf80gpTwqPjJ305Zdfan1w1P4D/  
gKOTJj13FR7pk47jjrofCAu18dTemw+DRAd8KwWlnjrr1hCrh7967JzBHv1QsL2AjAeLmg0m0G1ubtkg0ZVYebcmJ3b/  
Zcp37C83TU/  
bkWfL4UPCaKBXqj+5EuxohoJsch4vLZKzdGEVNW2kLM0YRcz9jmUCVo67yXoexhxiLKLbsXu0T4v8QxZpmz6c0G1IL+yGTF4zz  
9Ib35tVrKfPA7cDNzY0x/mu3WUn69rt08nj0QNIz0PXPpDZWt7jxy3YQ6zTg7He8bu1RVslrp/  
IDpjpmlWFTdnje9uc437drRiMDAPS/cvrQv/  
NQ77dbjNu10Mz8jK0YwgJSYfuu0HLAHNoM15a2PR1Vm785Luno+KChYQuS8tNIuXLIa8eGseVMLbFy2HHTUmdxaQvXAd4/  
jheDCUN1rfdLzmADu4E+g2G63EdiYiywbHtYeIaUS93i4r9R8Bf5V2Lhxg+6PUekJZN3pBKtcmYl+6reYmFpmzdW65VKkt/  
d2BmGM986dHfPPRB+Y09w//  
aBacAPDbhbbugBq29QTKJuf6Tu4Qq7VaBVUSH3P9UcHoNynKBqLW0FXLRhASEFBe4leHadZMbuImy0wFzQphjYcvDP2UspUIKkH  
f35QwFFcMvSdiUUKYESAxmFtwKgCCL1WQ0hSDPwYj0SGT0TGESzXt33u74qLAWsBA8HjcSjaAQsqBMXwqQ+Q6mBYXbc0cT5b4K  
ynWQCvBvBnILcBSKOUWwFybp4/TQp0YjAR5xHdc4VdDAFLhtyXEMHRUqhzMDm3IQghUSjIbgHePFyazu9wFMAPA2YCFx/  
zq46ApX/b8JiszzJuBLv+I0VP6BP4PG/gpkAhe/+GEZuLqEPDcRajhxILyV/jLTZvtzr17yS0CAPVRpBS0dkLb+wANgd/  
uBwDDkZ/+BBVa0b8v1L3wdXpddSRmZglVEnvbRnGJLSH5UKYnaFCD16/3ZETqsr04j6CknsnZtm0rpZsdqIB2DnQe/  
9LXoNmILJnmDYA9UEjKofUPDBn/xQKFQh9o58D7Vs5eUARLsRPFpW25EAVJBnb/  
wxF2kC7PpBdzsckVeThhIC80mVeu1TMXmEzhXw7ZF00Apm9IS37bVqWl/zkMEXwjMd/  
eYptNytXrMB2q04d0bi0PjImkCrJozYT2vNMlRX0ZuoDt06LDMWRRLXAILLEbTcJsGTbDgN0mVNe1Yfa7uTAUHmdk7lek8tIF  
WfYw0tQYG+vr/q1j5HFGEKqXUJZ5VrQXgXSmBACgWdtBcHctHz4+mm7f9yNGYLMVXHfhhVS81w47YyYhqV01X23s9U/  
EF0g2gT2MM7VyhYlYfLzFioqNIBf4c12P0+/  
Qoo9ju0c0DatWuQSD2mLUKtgNbs6B7TV4Z1GXiaYq6xrJ7N8fLaTAvb+zWmIhzhM7ewFQ+f7L1z4LjFRJq0bT1oszJXaxdSyMcw  
gSjG3M5/cgbLvuYtF6GfV/  
zi0+ksrjZQ5sBVq1bq0I4YpCtsLReXkjlnxLNT89FXnVm0ogYGQ8ZnASDv5d0lbqxvK44LmKhmewP0ojr8k8yB2nwWQDuT0YbY  
BhLgKGTPT5FASH05tdk2bGkf/CLMtIKsx88Dw7e17Gc3QfUPtGmTf/1DX7nr06Igd3kWiBvo8dmJu/+4YzYtRxEkAdwbGwa/  
Sxd45IGPrBxLBJhwJNLZheKcEY2NtzlNZYmYsDg1NTmxLjz8hFa7XBcvXsQa/  
8kMc7a+zdhGHYUPc3hGE644cd4WtLkZkjrcNnkziZ2EpVYAhAjMdAxaA02Ek7L5yVo6LBUGAdERXefp01/+iD1ria29qbicghy  
LOLKGDenGelARjhFE/AFYrYBpqqEm0kjspkFzQXvos1Juq1vt9Lntc13bkXXhIaG9RhF6Yc7+w/sVM/  
G9jAmpzWqmszIUOfMynamecQAGTbCUyYrZMQ6zcpMA/  
7cT7yPz5ruBD1zXrE6wGr40d0sAsrZAhANXFr9DCoYgYICZSoGcuxT7PR8pGCAVZlrLmZNBcmpz/  
MCXff4fP691N1AvEd6P0I8cPbd01/  
CjXwHTadtDl1J0u16baADAGoidUSShaS3F64RuC9C1YAJLqhb9TW6tT3X8SR9t6jJw8PhJ12XKDNsmLunHrVzLsf47ImxTvo2  
wnoJA2vENs7XgVdqAt5iue2AbYLXZLZM+mWhNGuksCKiKSchwNPTU+2gc0HCBWalwFA9+cYWE38bwhzJLzBZEGHuhFai5mgxKDP  
aAWLQAozM0fLpBU2VH6CSVG5LZRq3qSouNz0LMD+/  
oOny5cvw4hejLQnfhXr1jIdh71jA0ztJon1bcY850LmjnbiofLbmEXJL7fzmqK2dzz8ALBXDMbFYghUu2BV3yCpLjLs9HUVmvi  
0AIapGJgaMGSf5jcXjUbFydzmjsyflFyXlZGMT5Y0LI7oZSAvUFwXm5b4i0DKVS/  
d2X79m0DwEPL7l0Z5hsSMhJmvYGAXJGhr2pHL0Z/BpNVHLz654/  
KfE6FUK2T+wPavMq55TWkQBE8GPQ2+eFEZ+5Y0ayFIlh2vYaiV00bRw4rDxrUYIWA+DYNN57pNCJ82Zs1brweP7zVdSd2jJwI  
cJ1+IRY8UCKqTCABw4eVbVxx08Ezkb4ys2fWejFKRgn72uUN7Qhw/  
vV5co+7mDsXmXYDLfYV9yRzk+tBxtgslv6E5oVlQ70Le8tE4182wk6vh2NTQum1Yrc7RpwgTysVqjgK2oYGi+AeLj729xK0y4aU  
M+SxGvMltBQfl5UFKMcA89kNYW69pGcUUJj/wi5ZAwdxc+rMhWGF/  
LX4Ru5MyXNxia0W53Aq3Yf6PwgLK50LSj7fbVgnal7hebdwFerhto1YQJ5AW3/  
ACQknJXDYNTgh+kAuSA9SyRsv3hVEG7JkygUyPA2kph2AJhDRICqbuJ7eyLXQ0kwF6ApkJRxdKytiwo0gpa0nBE5WPVW4mScd0+  
nqYLtHRg4u9n4xBjEdRweI0g8VKYaLnzDRwzf3KMuXo61TR2AKTAQdtY6oJDDR4gbU1b0ZLxR2nqyemJCYmwhw6AcB/  
AID93zd+9+L6AAAAAEFLTksUqMCC",  
"h160_icon":  
"data:image/png;base64,iVBORw0KGGoAAAANSUHEUGAAADgAAAA4CAYAAAHfgQuIAAAAAAXNSR0IArs4c6QAAAAARnQU1BAACx
```

```

jwv8YQUAAAAJcEhZcwAADSMAAA7DAcdvqGQAAAd+SURBVFH7ZlpbFVFFMdPdyhLbUG2KDbIYlMJJSCLoEYU06AYIdHIZhUriIU
iglfjTEz8AEhViBlgZHBObKNGPwDQdAYRYmgAdGQKsomINAFuvfV/
39m7ut99713371viyb8kpNZ7r0zc2b0zJyZK25kmVACa6QjJ0Mu7T4u/
ZH8Vefa6MALFJOUTBOS0SYkXU0YRp4J04Jp6Y06FdpaivMm7Hxo6XeyXFvHhN7B5BIxQZxFriJ1SYMoaVSAgi66VQUTLswMGxd
LR1tEP2kkwThmFXkM9+HtKj6cnyQfmxEAi4+XHGQZC6xoKWQKZpVLRcHaLnag9Z31kwr6Me8atxqQx0YROBseqPQeIexMvWi8
Hngz/
yNk5tNTXIaw2gbsjc00UcXTQDjmi04prTBhCt0FoMCGpN2EiZib0gF3WtsGc87JND0AV3YTg053qJMN0wqM6qWGeoydZyfT2CB0
VAnsy1hBEY9qhxds4r70AZmmo+GEjQ9xNjHSfHRjXHFPEcHEfcFainQ0PvqYMCLEXQAMj0Ce0NEQhvpv+W6GTGpMX3AviJZtGtn
aifGjSQ1mwJUhzZRNjiHcx7oafUS1ABfU1TSK9upgcG7AQLoP/
6FR0fJkRyMI610b8yGtLJNY0im00sLbTRgTq0LuTwn0VHIhSzK+XFUR5xj14wNw1IROPoeMzM6UtdbYnq5UW3AF5DYI/
a04GWAWSkF6rs52Jd+8GzcPsYBIPkGqGGJafL90esev0dDXmAKX9i3aENPjVCqFwPv5fTrPhb8awlJf2Xb0K3iJN8Tv8hc/
xjDoz3li6kB5jt/
UrIjPycg0FXbXyU444c2znjPHMc0SBAmDx2aZgi2XgVa5jHmUJSXyMdJqDTbvjWI8UaaawpSryTl0liClh9kSE4853l7m4TFY72
aiwPOI52WEF8LF4BDeGEY4K3XF68Tfm7leKpoqBft9GFt/vyhtCH/TSXd8mfylCgl0yw3/
BtrF6fBkmb6WtoJX5BYTDYLKXkLgeRr40pAH6ZysiBp6LsePhnmwCF8NjIRbhXw2/KmJ8r6ZAK1ln8ke/
SgUc4pcDBkMcw2U9ZAb61+QFkgxCqim+Z9vkPaiLeq4dhDck+cMVBxSav2zdPR8vc3RKY2r5JMu2brMfSflx0TtsgZRHjh/
gag7FqvCYSHImXUbzqQ5L0sJonzJSQ7eY60CYPyGIXB6cwUQLLFY0V7Knt4q9+I9LhABp/pMu07eBrh/
XXM6uw0FsetjTfgFEF6j/aBSPHltxrSjd56sM3me8TUPDcGuvTgs75qoz+KpkDcPFp70E3biqdB+Tww/
ufBEXBwerFeLNYllAnBzZdzko7vZjPZGWcefR9P0KeF6WwzF0mHglmbgfmbly95QmbT+OnwYq4WpXV+ZVnj84irgrbedTURL6
DTgGzTz+WvsqukCr6nChXxu0DkFQdE7mF8jRRDBkDudrKpYX5nPrfPiDViHs5vrBhoyG8Qr4b4rYY9ji4UBpZPm/
wds1Vvshk5gunaWKC7fK/DEInJhL5ECqzju/aBxkzzH07BRTsmS/nzPNyCK/F4sb3wuuaDzu0bWo3XcrT2CMHVqHczT/
VU13+20ZK0ZAbxbDggc1FsB1yGUJLkIdyVRfghZukUKkayFk6iCFIz/
x2PH2odNJA0NfhhpNroRwmpLlwgjCqGAt+qc11R6nszm9SRuD7Z5dCDKljfiIKS0ELmtM9UGHbBwsc4PMLBsLS96aLYv043B
ohzjmljhbhgMB/
QVBuHrI+glgeU0IkW0GLvgcWyNEB+ofcp8QEyvGv4H6dSg6pUpCw7KmYn19HuPEIobFV0vKr1N3rcZ2TPFLnQooUtVTKV7GUI+b
UN1ankksqFeQk68dCwq40k0gYFKTbzZ90EyCLz2tXttbVcirHn88xMLfNgnte0789DBHhPehvKkoaVgpm2BawQ7iwXT/
KwmatF39M0GlnWnIzVhReY/qSgP89m5V8iw65cVSW5dzJa5rLkCv12QZkrxy4f56AcLtx9Mqa1ewX2uL/
J2VJRnVF6T1um3Kx6wZf5XcuX+hrNKvi0BZY9BtytWiEmcgrCwahRjNs0X57paC1ZN/AL/
QqSD8ht4Rf+YNgVxfXyGbcnIjL2W2G25H0yF0GhSRRpB/
DzLKLXJpLochVBzhXDn0cw7HMMw9Jh4G0xMdRo+LXuFEBT2g0ajnvcutEiiokoFis/
N9mwi80LQXovJt0tLJG7orZ6M6KfyhcXKGI87B0QE5ywxX00kSnCs7T9WF3FgpoNxMBMLsjlyC/
AhRypF0KEhgB22V0SauqGLS5v6zTqW0dCLiAdZBJrwjzyNgj6eUdCp4prLNLXaK6hrLc6b8YiedCtbuOCLfmzhRq1yqSdUq0gan
/Z/8Fr74U9n09mF1FnTbW32RKgW5D3Fztv+v6dovX5aeWa4cCG7uv0awLzJ093g9T48l5D/P/
4VR1h00pUuWrEqe98iUks45GFxyKACddmUctKp4HFs9sFVs7pW3dMk5Vriv0T33fPkGEKeSq6Q0CL/Ap3jTWX/
GRgjAAAAELFTkSuQmCC"
}
}

```

RescueTrack Test Program

Dieses Programm zeigt die grundlegende Verwendung der RescueTrack-Benutzeroberfläche.

```

{
  "title": "Rescue Track Test Program",
  "data": {
    "statusMessages": [
      "0. Dispatch accepted",
      "1. On the way to the scene",
      "2. At the scene",
      "3. On the way to the hospital",
      "4. At the hospital",
      "5. On the way back to base (Available)",
      "6. At Home base (Available)",
      "7. Unavailable for dispatch"
    ]
  },
  "briefing": [
    { "title": "Rescue Track Test" },
    { "text": "Enable Rescue Track (visible on DMAP STATUS):" },
    { "buttonbar": [
      {
        "title": "Enabled",
        "commands": [
          { "set": { "local": "IS_RESCUE_TRACK_ENABLED" }, "value": 1 },
          { "call_macro": "Update_RescueTrack" }
        ]
      }
    ]
  },
}

```



```

    "select_condition":{"require":{"local":"IS_RESCUE_TRACK_ENABLED"}, "eq":1}
  },
  {
    "title":"Disabled",
    "commands":[
      {"set":{"local":"IS_RESCUE_TRACK_ENABLED"}, "value": 0},
      {"call_macro":"Update_RescueTrack"}
    ],
    "select_condition":{"require":{"local":"IS_RESCUE_TRACK_ENABLED"}, "eq":0}
  }
]},
{"text":"My Status:"},
{"buttonbar":[
  {
    "title":{"struct":{"static":"statusMessages"}, "index": 0},
    "commands":[
      {"set":{"var":["L:MISSION_RESCUETRACK_STATUS", "number"]}, "value": 0},
      {"call_macro":"Update_RescueTrack"}
    ],
    "select_condition":{"require":{"var":["L:MISSION_RESCUETRACK_STATUS", "number"]}, "eq":0}
  }
]},
{"buttonbar":[
  {
    "title":{"struct":{"static":"statusMessages"}, "index": 1},
    "commands":[
      {"set":{"var":["L:MISSION_RESCUETRACK_STATUS", "number"]}, "value": 1},
      {"call_macro":"Update_RescueTrack"}
    ],
    "select_condition":{"require":{"var":["L:MISSION_RESCUETRACK_STATUS", "number"]}, "eq":1}
  }
]},
{"buttonbar":[
  {
    "title":{"struct":{"static":"statusMessages"}, "index": 2},
    "commands":[
      {"set":{"var":["L:MISSION_RESCUETRACK_STATUS", "number"]}, "value": 2},
      {"call_macro":"Update_RescueTrack"}
    ],
    "select_condition":{"require":{"var":["L:MISSION_RESCUETRACK_STATUS", "number"]}, "eq":2}
  }
]},
{"buttonbar":[
  {
    "title":"Other status",
    "commands":[
      {"set":{"var":["L:MISSION_RESCUETRACK_STATUS", "number"]}, "value": 3},
      {"call_macro":"Update_RescueTrack"}
    ],
    "select_condition":{"require":{"var":["L:MISSION_RESCUETRACK_STATUS", "number"]}, "gt":2}
  }
]},
{"title":"Dispatcher messages"},
{"text":"Location:"},
{
  "text":"You must select a location on the map prior to sending a dispatch. Go back to the map
and click a location, then come back to send your message.",
  "color":"red",
  "show_condition":{"require":{"has_location":"$MISSION_SELECTED_POI_LOCATION"}, "eq": 0}
},
{
  "text":{"text":"{0:LOCATION}", "params": ["$MISSION_SELECTED_POI_LOCATION"]},
  "show_condition":{"require":{"has_location":"$MISSION_SELECTED_POI_LOCATION"}, "eq": 1}
},
{
  "text":"Name: (required field)",
  "color":"red",
  "show_condition":{"require":{"local":"dispatcher_textbox_name"}, "eq":""}
},
{

```

```

    "text":"Name:",
    "show_condition": {"require":{"local": "dispatcher_textbox_name"}, "ne":""}
  },

  {"textbox":"dispatcher_textbox_name"},

  {
    "text":"Message: (required field)",
    "color":"red",
    "show_condition": {"require":{"local": "dispatcher_textbox_msg"}, "eq":""}
  },
  {
    "text":"Message:",
    "show_condition": {"require":{"local": "dispatcher_textbox_msg"}, "ne":""}
  },
  {"textbox":"dispatcher_textbox_msg"},
  {"buttonbar":[
    {
      "title":"Send Message",
      "commands":[

        {"modify_array": {"local": "Dispatcher_Messages"}, "prepend": {"create_struct": {
          "from": {"local": "dispatcher_textbox_name"},
          "time": {"fn":"get_time_string"},
          "text":{"local": "dispatcher_textbox_msg"},
          "waypoint": {"location": "$MISSION_SELECTED_POI_LOCATION"}
        }}}},

        {"set": {"local":"dispatcher_textbox_msg"}, "value": ""},

        {"call_macro":"Update_RescueTrack"}
      ],
      "select_condition":{"and":[
        {"require":{"local": "dispatcher_textbox_msg"}, "ne":0},
        {"require": {"has_location": "$MISSION_SELECTED_POI_LOCATION"}, "eq": 1}
      ]}
    }
  ]}],

  {"text":"outgoing messages (Dispatcher_Messages):"},
  {"text": {"json:stringify":{"local":"Dispatcher_Messages"}}},

  {"#comment":"you may add comments as needed"}
],
"macros": {
  "Update_RescueTrack":[
    {"if":{"local":"IS_RESCUE_TRACK_ENABLED"}, "eq":1,"then":[
      {"set_rescuetrack":{"
        "statusVar": "L:MISSION_RESCUETRACK_STATUS",
        "statusMessages": {"static": "statusMessages"},
        "dispatcherMessages": {"local": "Dispatcher_Messages"}
      }}, {"activate_waypoint_commands":[
        {"set":{"param":"ACTIVE_MSG"}, "value": {"struct": {"local": "Dispatcher_Messages"},
"index": {"param":"$index"}}},

        {"copy_query_to_location": {"struct": {"param":"ACTIVE_MSG"}, "path": "waypoint"},
"to":"temp"},
        {"set_route":"temp"},

        {"set_message":{"text": "Go direct: {0}<br />\n{1:LOCATION}", "params": [
          {"json:stringify":{"struct": {"param":"ACTIVE_MSG"}, "path": "waypoint"}},
          "temp"
        ]}}
      ]}],
    ], "else":[
      {"set_rescuetrack": null}
    ]
  ]
}
],
"objectives": [
  {

```

```
"title": "Done",
"commands": [
  {"set":{"local":"IS_RESCUE_TRACK_ENABLED"}, "value":1},
  {"set":{"local": "Dispatcher_Messages"}, "value": []},
  {"set": {"local":"dispatcher_textbox_name"}, "value": ""},
  {"set": {"local":"dispatcher_textbox_msg"}, "value": ""},

  {"call_macro":"Update_RescueTrack"},
  {"sleep": "forever"}
]
}
]
```

SDK H:Events

Home Cockpit SDK

Unter hpg-airbus-h145\html_ui\HPGH145-System\H145_Keys.txt finden Sie eine vollständige Auflistung der Ereignisse für Ihren Build.

Overhead Panel

| Name | Event |
|------------------------|---|
| Bus Tie 1 ON | H:H145_SDK_OH_BUSTIE_1_ON |
| Bus Tie 1 OFF | H:H145_SDK_OH_BUSTIE_1_OFF |
| Bus Tie 2 ON | H:H145_SDK_OH_BUSTIE_2_ON |
| Bus Tie 2 OFF | H:H145_SDK_OH_BUSTIE_2_OFF |
| Generator 1 ON | H:H145_SDK_OH_GEN_1_ON |
| Generator 1 OFF | H:H145_SDK_OH_GEN_1_OFF |
| Generator 2 ON | H:H145_SDK_OH_GEN_2_ON |
| Generator 2 OFF | H:H145_SDK_OH_GEN_2_OFF Emergency Shed Bus |
| Generator 2 ON | H:H145_SDK_OH_EMER_SHED_BUS_ON |
| Emergency Shed Bus OFF | H:H145_SDK_OH_EMER_SHED_BUS_OFF |
| Master Battery OFF | H:H145_SDK_OH_BAT_MASTER_OFF |
| Master Battery ON | H:H145_SDK_OH_BAT_MASTER_ON |
| Master Battery ENGAGE | H:H145_SDK_OH_BAT_MASTER_ENGAGE |
| Master Battery UP | H:H145_SDK_OH_BAT_MASTER_UP |
| Master Battery DOWN | H:H145_SDK_OH_BAT_MASTER_DOWN |
| HYD System 1 TEST | H:H145_SDK_OH_HYD_TEST_SYS1 |
| HYD Test OFF | H:H145_SDK_OH_HYD_TEST_OFF |
| HYD System 2 TEST | H:H145_SDK_OH_HYD_TEST_SYS2 |
| Engine 1 Fire Test OFF | H:H145_SDK_OH_FIRE_ENG1_TEST_OFF |
| Engine 1 Fire Test EXT | H:H145_SDK_OH_FIRE_ENG1_TEST_EXT Engine 1 Fire Test |
| EXT+WARN | H:H145_SDK_OH_FIRE_ENG1_TEST_EXT_WARN |
| Engine 2 Fire Test OFF | H:H145_SDK_OH_FIRE_ENG2_TEST_OFF |
| Engine 2 Fire Test EXT | H:H145_SDK_OH_FIRE_ENG2_TEST_EXT Engine 2 Fire Test |
| EXT+WARN | H:H145_SDK_OH_FIRE_ENG2_TEST_EXT_WARN |
| TEST PREFLIGHT | H:H145_SDK_OH_LAMP_TEST_PREFLIGHT |
| TEST OFF | H:H145_SDK_OH_LAMP_TEST_OFF |
| TEST LAMP | H:H145_SDK_OH_LAMP_TEST_LAMP |
| DC Receptacles OFF | H:H145_SDK_OH_DC_RECEPT_OFF |
| DC Receptacles ON | H:H145_SDK_OH_DC_RECEPT_ON |
| Standby Battery ON | H:H145_SDK_OH_STANDBY_BATTERY_ON |
| Standby Battery OFF | H:H145_SDK_OH_STANDBY_BATTERY_OFF Standby Battery |
| TOGGLE | H:H145_SDK_OH_STANDBY_BATTERY_TOGGLE |
| Avionics Bus 1 ON | H:H145_SDK_OH_AVIONICS_1_ON |
| Avionics Bus 1 OFF | H:H145_SDK_OH_AVIONICS_1_OFF |
| Avionics Bus 1 TOGGLE | H:H145_SDK_OH_AVIONICS_1_TOGGLE |
| Avionics Bus 2 ON | H:H145_SDK_OH_AVIONICS_2_ON |
| Avionics Bus 2 OFF | H:H145_SDK_OH_AVIONICS_2_OFF |
| Avionics Bus 2 TOGGLE | H:H145_SDK_OH_AVIONICS_2_TOGGLE |
| Emergency Floats OFF | H:H145_SDK_OH_EMER_FLOATS_OFF |
| Emergency Floats ARM | H:H145_SDK_OH_EMER_FLOATS_ARM |
| Emergency Floats TEST | H:H145_SDK_OH_EMER_FLOATS_TEST |
| Fuzz Burner OFF | H:H145_SDK_OH_FUZZ_CHIP_BURNER_OFF |
| Fuzz Burner ON | H:H145_SDK_OH_FUZZ_CHIP_BURNER_ON |
| LAVCS OFF | H:H145_SDK_OH_LAVCS_OFF |
| LAVCS PIL | H:H145_SDK_OH_LAVCS_PIL |
| LAVCS PAX | H:H145_SDK_OH_LAVCS_PAX |
| Windshield Wiper OFF | H:H145_SDK_OH_WINDSHIELD_WIPER_OFF |
| Windshield Wiper SLOW | H:H145_SDK_OH_WINDSHIELD_WIPER_SLOW |

| | |
|---------------------------|---|
| Windshield Wiper FAST | H:H145_SDK_OH_WINDSHIELD_WIPER_FAST |
| Air Conditioning OFF | H:H145_SDK_OH_AIR_CONDITIONING_OFF |
| Air Conditioning ON | H:H145_SDK_OH_AIR_CONDITIONING_ON |
| Cockpit Vent OFF | H:H145_SDK_OH_COCKPIT_VENT_OFF |
| Cockpit Vent ON | H:H145_SDK_OH_COCKPIT_VENT_ON |
| Pitot Heater 1 ON | H:H145_SDK_OH_PITOT_1_ON |
| Pitot Heater 1 OFF | H:H145_SDK_OH_PITOT_1_OFF |
| Pitot Heater 1 TOGGLE | H:H145_SDK_OH_PITOT_1_TOGGLE |
| Pitot Heater 2 ON | H:H145_SDK_OH_PITOT_2_ON |
| Pitot Heater 2 OFF | H:H145_SDK_OH_PITOT_2_OFF |
| Pitot Heater 2 TOGGLE | H:H145_SDK_OH_PITOT_2_TOGGLE |
| IBF 1 CLOSED | H:H145_SDK_OH_IBF_1_CLOSED |
| IBF 1 OPEN | H:H145_SDK_OH_IBF_1_OPEN |
| IBF 1 NORMAL | H:H145_SDK_OH_IBF_1_NORMAL |
| IBF 2 CLOSED | H:H145_SDK_OH_IBF_2_CLOSED |
| IBF 2 OPEN | H:H145_SDK_OH_IBF_2_OPEN |
| IBF 2 NORMAL | H:H145_SDK_OH_IBF_2_NORMAL |
| IBF RECALL OFF | H:H145_SDK_OH_IBF_RECALL_OFF |
| IBF RECALL ON | H:H145_SDK_OH_IBF_RECALL_ON |
| Fuel Engine 1 Prime OFF | H:H145_SDK_OH_FUEL_ENG1_PRIME_OFF |
| Fuel Engine 1 Prime ON | H:H145_SDK_OH_FUEL_ENG1_PRIME_ON |
| Fuel Engine 2 Prime OFF | H:H145_SDK_OH_FUEL_ENG2_PRIME_OFF |
| Fuel Engine 2 Prime ON | H:H145_SDK_OH_FUEL_ENG2_PRIME_ON |
| Fuel Transfer Forward OFF | H:H145_SDK_OH_FUEL_TRANSFER_FWD_OFF |
| Fuel Transfer Forward ON | H:H145_SDK_OH_FUEL_TRANSFER_FWD_ON |
| Fuel Transfer Aft OFF | H:H145_SDK_OH_FUEL_TRANSFER_AFT_OFF |
| Fuel Transfer Aft ON | H:H145_SDK_OH_FUEL_TRANSFER_AFT_ON |
| ACAS MUTE | H:H145_SDK_OH_AUDIO_ACAS_MUTE |
| ACAS NORMAL | H:H145_SDK_OH_AUDIO_ACAS_NORMAL |
| ACAS TEST | H:H145_SDK_OH_AUDIO_ACAS_TEST |
| HTAWS MUTE | H:H145_SDK_OH_AUDIO HTAWS_MUTE |
| HTAWS MUTE-FOR-5-MINS | H:H145_SDK_OH_AUDIO HTAWS_MUTE_5MIN |
| HTAWS NORMAL | H:H145_SDK_OH_AUDIO HTAWS_NORMAL |
| HTAWS STANDBY | H:H145_SDK_OH_AUDIO HTAWS_STANDBY |
| Int Lights Cargo/Pax OFF | H:H145_SDK_OH_INT_LIGHT_CARGO_PAX_OFF |
| Int Lights Cargo/Pax PAX | H:H145_SDK_OH_INT_LIGHT_CARGO_PAX_PAX |
| Int Lights Cargo/Pax BOTH | H:H145_SDK_OH_INT_LIGHT_CARGO_PAX_ON |
| Int Emergency Exits OFF | H:H145_SDK_OH_INT_LIGHT_EMERGENCY_EXITS_OFF |
| Int Emergency Exits ARM | H:H145_SDK_OH_INT_LIGHT_EMERGENCY_EXITS_ARM |
| Int Emergency Exits ON | H:H145_SDK_OH_INT_LIGHT_EMERGENCY_EXITS_ON |
| Int Panel Lights DAY | H:H145_SDK_OH_INT_LIGHT_INSTRUMENT_PANEL_DAY |
| Int Panel Lights NIGHT | H:H145_SDK_OH_INT_LIGHT_INSTRUMENT_PANEL_NIGHT |
| Int Panel Lights NVG | H:H145_SDK_OH_INT_LIGHT_INSTRUMENT_PANEL_NVG |
| Ext Lights HISL ON | H:H145_SDK_OH_EXT_LIGHT_HISL_ON |
| Ext Lights HISL OFF | H:H145_SDK_OH_EXT_LIGHT_HISL_OFF |
| Ext Lights HISL TOGGLE | H:H145_SDK_OH_EXT_LIGHT_HISL_TOGGLE |
| Cockpit Vent INCREASE | H:H145_SDK_OH_COCKPIT_VENT_POT_INC |
| Cockpit Vent DECREASE | H:H145_SDK_OH_COCKPIT_VENT_POT_DEC |
| Bleed Heading INCREASE | H:H145_SDK_OH_BLEED_HEATING_POT_INC |
| Bleed Heading DECREASE | H:H145_SDK_OH_BLEED_HEATING_POT_DEC |
| Panel Lights INCREASE | H:H145_SDK_OH_INT_LIGHT_INSTRUMENT_PANEL_KNOB_INC |
| Panel Lights DECREASE | H:H145_SDK_OH_INT_LIGHT_INSTRUMENT_PANEL_KNOB_DEC |
| Front Light TOGGLE | H:H145_SDK_OH_COCKPIT_LIGHT_FRONT_TOGGLE |
| Front Light ON | H:H145_SDK_OH_COCKPIT_LIGHT_FRONT_ON |
| Front Light OFF | H:H145_SDK_OH_COCKPIT_LIGHT_FRONT_OFF |
| Rear Light TOGGLE | H:H145_SDK_OH_COCKPIT_LIGHT_REAR_TOGGLE |
| Rear Light ON | H:H145_SDK_OH_COCKPIT_LIGHT_REAR_ON |
| Rear Light OFF | H:H145_SDK_OH_COCKPIT_LIGHT_REAR_OFF |

Engine Control Panel (ECP)

| Name | Event |
|---------------------------------|---------------------------------------|
| Toggle both engines FLIGHT/IDLE | H:H145_SDK_ECP_FADEC_DUAL_TOGGLE |
| Main 1 FLIGHT | H:H145_SDK_ECP_MAIN_1_FLIGHT |
| Main 1 IDLE | H:H145_SDK_ECP_MAIN_1_IDLE |
| Main 1 OFF | H:H145_SDK_ECP_MAIN_1_OFF |
| Main 1 UP | H:H145_SDK_ECP_MAIN_1_UP |
| Main 1 DOWN | H:H145_SDK_ECP_MAIN_1_DOWN |
| Main 2 FLIGHT | H:H145_SDK_ECP_MAIN_2_FLIGHT |
| Main 2 IDLE | H:H145_SDK_ECP_MAIN_2_IDLE |
| Main 2 OFF | H:H145_SDK_ECP_MAIN_2_OFF |
| Main 2 DOWN | H:H145_SDK_ECP_MAIN_2_DOWN |
| Main 2 UP | H:H145_SDK_ECP_MAIN_2_UP |
| Main 1 Latch OFF | H:H145_SDK_ECP_MAIN_LATCH_1_OFF |
| Main 1 Latch ON | H:H145_SDK_ECP_MAIN_LATCH_1_ON |
| Main 2 Latch OFF | H:H145_SDK_ECP_MAIN_LATCH_2_OFF |
| Main 2 Latch ON | H:H145_SDK_ECP_MAIN_LATCH_2_ON |
| FADEC EMER 1 OFF | H:H145_SDK_ECP_FADEC_EMER_1_OFF |
| FADEC EMER 1 ON | H:H145_SDK_ECP_FADEC_EMER_1_ON |
| FADEC EMER 1 Latch CLOSE | H:H145_SDK_ECP_FADEC_EMER_1_CAP_CLOSE |
| FADEC EMER 1 Latch OPEN | H:H145_SDK_ECP_FADEC_EMER_1_CAP_OPEN |
| FADEC EMER 2 OFF | H:H145_SDK_ECP_FADEC_EMER_2_OFF |
| FADEC EMER 2 ON | H:H145_SDK_ECP_FADEC_EMER_2_ON |
| FADEC EMER 2 Latch CLOSE | H:H145_SDK_ECP_FADEC_EMER_2_CAP_CLOSE |
| FADEC EMER 2 Latch OPEN | H:H145_SDK_ECP_FADEC_EMER_2_CAP_OPEN |

Autopilot Control Panel (APCP)

| Name | Event |
|--------------------------|------------------------------------|
| A.TRIM TOGGLE | H:H145_SDK_APCP_ATRIM_TOGGLE |
| A.TRIM ON | H:H145_SDK_APCP_ATRIM_ON |
| A.TRIM OFF | H:H145_SDK_APCP_ATRIM_OFF |
| AP1 TOGGLE | H:H145_SDK_APCP_AP1_TOGGLE |
| AP1 ON | H:H145_SDK_APCP_AP1_ON |
| AP1 OFF | H:H145_SDK_APCP_AP1_OFF |
| AP2 TOGGLE | H:H145_SDK_APCP_AP2_TOGGLE |
| AP2 ON | H:H145_SDK_APCP_AP2_ON |
| AP2 OFF | H:H145_SDK_APCP_AP2_OFF |
| BKUP TOGGLE | H:H145_SDK_APCP_BKUP_TOGGLE |
| BKUP ON | H:H145_SDK_APCP_BKUP_ON |
| BKUP OFF | H:H145_SDK_APCP_BKUP_OFF |
| ALT TOGGLE | H:H145_SDK_APCP_ALT_TOGGLE |
| ALT ON | H:H145_SDK_APCP_ALT_ON |
| ALT OFF | H:H145_SDK_APCP_ALT_OFF |
| (VS/FPA HDG/TRK) TOGGLE | H:H145_SDK_APCP_GPSMODE_TOGGLE |
| (VS/FPA HDG/TRK) VS/HDG | H:H145_SDK_APCP_GPSMODE_TRAD |
| (VS/FPA HDG/TRK) TRK/FPA | H:H145_SDK_APCP_GPSMODE_GPS |
| ALT.A TOGGLE | H:H145_SDK_APCP_ALTA_TOGGLE |
| ALT.A ON | H:H145_SDK_APCP_ALTA_ON |
| ALT.A OFF | H:H145_SDK_APCP_ALTA_OFF |
| ALT.A Clockwise | H:H145_SDK_APCP_ALTA_Clockwise |
| ALT.A AntiClockwise | H:H145_SDK_APCP_ALTA_AntiClockwise |
| CR.HT TOGGLE | H:H145_SDK_APCP_CRHT_TOGGLE |
| CR.HT ON | H:H145_SDK_APCP_CRHT_ON |
| CR.HT OFF | H:H145_SDK_APCP_CRHT_OFF |
| CR.HT Clockwise | H:H145_SDK_APCP_CRHT_Clockwise |
| CR.HT AntiClockwise | H:H145_SDK_APCP_CRHT_AntiClockwise |

| | |
|-------------------|-----------------------------------|
| VS TOGGLE | H:H145_SDK_APCP_VS_TOGGLE |
| VS ON | H:H145_SDK_APCP_VS_ON |
| VS OFF | H:H145_SDK_APCP_VS_OFF |
| VS Clockwise | H:H145_SDK_APCP_VS_Clockwise |
| VS AntiClockwise | H:H145_SDK_APCP_VS_AntiClockwise |
| HDG TOGGLE | H:H145_SDK_APCP_HDG_TOGGLE |
| HDG ON | H:H145_SDK_APCP_HDG_ON |
| HDG OFF | H:H145_SDK_APCP_HDG_OFF |
| HDG Clockwise | H:H145_SDK_APCP_HDG_Clockwise |
| HDG AntiClockwise | H:H145_SDK_APCP_HDG_AntiClockwise |
| IAS TOGGLE | H:H145_SDK_APCP_IAS_TOGGLE |
| IAS ON | H:H145_SDK_APCP_IAS_ON |
| IAS OFF | H:H145_SDK_APCP_IAS_OFF |
| IAS Clockwise | H:H145_SDK_APCP_IAS_Clockwise |
| IAS AntiClockwise | H:H145_SDK_APCP_IAS_AntiClockwise |
| Easy AFCS Toggle | H:H145_SDK_AP_AFCS_EASY_TOGGLE |
| Easy AFCS On | H:H145_SDK_AP_AFCS_EASY_ON |
| Easy AFCS Off | H:H145_SDK_AP_AFCS_EASY_OFF |

Cyclic Control

| Name | Event |
|---------------------------------------|--|
| AP/BKUP ON | H:H145_SDK_AP_APBKUPON_UP |
| AP/BKUP ON (AP1 Only) | H:H145_SDK_AP_APBKUPON_LEFT |
| AP/BKUP ON (AP2 Only) | H:H145_SDK_AP_APBKUPON_RIGHT |
| AP/BKUP CUT | H:H145_SDK_AP_APBKUPCUT |
| AP/UM OFF | H:H145_SDK_AP_UM_OFF |
| AP/GTC | H:H145_SDK_AP_GTCGTCH |
| AP/GTC (Direct to GTC.H)(Advanced) | H:H145_SDK_AP_GTCGTCH_HOVER |
| Cyclic Beep Trim RIGHT | H:H145_SDK_CYCLIC_BEEP_RIGHT |
| Cyclic Beep Trim LEFT | H:H145_SDK_CYCLIC_BEEP_LEFT |
| Cyclic Beep Trim UP | H:H145_SDK_CYCLIC_BEEP_UP |
| Cyclic Beep Trim DOWN | H:H145_SDK_CYCLIC_BEEP_DOWN |
| Cyclic Beep Trim RESET/Zero(Uncommon) | H:H145_SDK_CYCLIC_BEEP_RESET |
| Set New Cyclic Center | H:H145_SDK_CYCLIC_FORCE_TRIM_SET_NEW_CENTER |
| Displace Cyclic Center (Force Trim) | H:H145_SDK_CYCLIC_FORCE_TRIM_DISPLACE_CENTER |
| Trim Release (HOLD) | H:H145_SDK_CYCLIC_TRIM_RELEASE_HOLD |
| Trim Release (Latch: Open) | H:H145_SDK_CYCLIC_TRIM_RELEASE_LATCH_PUSH |
| Trim Release (Latch: Closed) | H:H145_SDK_CYCLIC_TRIM_RELEASE_LATCH_RELEASE |
| Trim Release (Latch: Toggle) | H:H145_SDK_CYCLIC_TRIM_RELEASE_LATCH_TOGGLE |
| Message List RESET | H:H145_SDK_MESSAGELIST_RESET |

Collective Control

| Name | Event |
|---|--|
| Collective Beep Trim RIGHT | H:H145_SDK_COLLECTIVE_BEEP_RIGHT |
| Collective Beep Trim LEFT | H:H145_SDK_COLLECTIVE_BEEP_LEFT |
| Collective Beep Trim UP | H:H145_SDK_COLLECTIVE_BEEP_UP |
| Collective Beep Trim DOWN | H:H145_SDK_COLLECTIVE_BEEP_DOWN |
| Collective Beep Trim ATT YAW AUTORESET | H:H145_SDK_COLLECTIVE_YAW_TRIM_AUTO_RESET |
| Collective Trim Release (HOLD) | H:H145_SDK_COLLECTIVE_TRIM_RELEASE_HOLD |
| Collective Trim Release (Latch: Open) | H:H145_SDK_COLLECTIVE_TRIM_RELEASE_LATCH_PUSH |
| Collective Trim Release (Latch: Closed) | H:H145_SDK_COLLECTIVE_TRIM_RELEASE_LATCH_RELEASE |
| Collective Trim Release (Latch: Toggle) | H:H145_SDK_COLLECTIVE_TRIM_RELEASE_LATCH_TOGGLE |
| OEI HI/LO (Low) | H:H145_SDK_COLLECTIVE_OEI_HILO_LO |
| OEI HI/LO (High) | H:H145_SDK_COLLECTIVE_OEI_HILO_HI |
| OEI HI/LO (Toggle) | H:H145_SDK_COLLECTIVE_OEI_HILO_TOGGLE |
| Fill Floats | H:H145_SDK_FILL_FLOATS |
| GA (Go Around) | H:H145_SDK_COLLECTIVE_GA |

H145M Weapons

| Name | Event |
|--------------------------|--|
| Fire (Primary) | H:H145_SDK_PRIMARY_ACTION_COMMAND |
| Fire (Secondary) | H:H145_SDK_SECONDARY_ACTION_COMMAND |
| Installed (Toggle) | H:H145_SDK_EQUIP_WEAPONS_TOGGLE |
| Installed (On) | H:H145_SDK_EQUIP_WEAPONS_ON |
| Installed (Off) | H:H145_SDK_EQUIP_WEAPONS_OFF |
| Pod Left Type (Toggle) | H:H145_SDK_EQUIP_WEAPONS_POD_LEFT_TOGGLE |
| Pod Left Type (Gun) | H:H145_SDK_EQUIP_WEAPONS_POD_LEFT_GUN |
| Pod Left Type (Rockets) | H:H145_SDK_EQUIP_WEAPONS_POD_LEFT_ROCKETS |
| Pod Right Type (Toggle) | H:H145_SDK_EQUIP_WEAPONS_POD_RIGHT_TOGGLE |
| Pod Right Type (Gun) | H:H145_SDK_EQUIP_WEAPONS_POD_RIGHT_GUN |
| Pod Right Type (Rockets) | H:H145_SDK_EQUIP_WEAPONS_POD_RIGHT_ROCKETS |
| Reload Rockets | H:H145_SDK_WEAPON_RELOAD |
| Cleanup All Rockets | H:H145_SDK_WEAPON_CLEANUP |
| Master Arm TOGGLE | H:H145_SDK_EQUIP_WEAPONS_MASTER_ARM_TOGGLE |
| Master Arm OFF (SAFE) | H:H145_SDK_EQUIP_WEAPONS_MASTER_ARM_OFF |
| Master Arm ON (ARMED) | H:H145_SDK_EQUIP_WEAPONS_MASTER_ARM_ON |

Cabin

| Name | Event |
|--|-------------------------------------|
| Cockpit Door Left TOGGLE | H:H145_SDK_DOOR_COCKPIT_L_TOGGLE |
| Cockpit Door Left OPEN | H:H145_SDK_DOOR_COCKPIT_L_OPEN |
| Cockpit Door Left CLOSE | H:H145_SDK_DOOR_COCKPIT_L_CLOSE |
| Cockpit Door Right TOGGLE | H:H145_SDK_DOOR_COCKPIT_R_TOGGLE |
| Cockpit Door Right OPEN | H:H145_SDK_DOOR_COCKPIT_R_OPEN |
| Cockpit Door Right CLOSE | H:H145_SDK_DOOR_COCKPIT_R_CLOSE |
| Pax Door Left TOGGLE | H:H145_SDK_DOOR_PAX_L_TOGGLE |
| Pax Door Left OPEN | H:H145_SDK_DOOR_PAX_L_OPEN |
| Pax Door Left CLOSE | H:H145_SDK_DOOR_PAX_L_CLOSE |
| Pax Door Right TOGGLE | H:H145_SDK_DOOR_PAX_R_TOGGLE |
| Pax Door Right OPEN | H:H145_SDK_DOOR_PAX_R_OPEN |
| Pax Door Right CLOSE | H:H145_SDK_DOOR_PAX_R_CLOSE |
| Cargo Door Left TOGGLE | H:H145_SDK_DOOR_CARGO_L_TOGGLE |
| Cargo Door Left OPEN | H:H145_SDK_DOOR_CARGO_L_OPEN |
| Cargo Door Left CLOSE | H:H145_SDK_DOOR_CARGO_L_CLOSE |
| Cargo Door Right TOGGLE | H:H145_SDK_DOOR_CARGO_R_TOGGLE |
| Cargo Door Right OPEN | H:H145_SDK_DOOR_CARGO_R_OPEN |
| Cargo Door Right CLOSE | H:H145_SDK_DOOR_CARGO_R_CLOSE |
| Cockpit And Pax Doors TOGGLE | H:H145_SDK_DOORS_TOGGLE |
| Cockpit And Pax Doors INSTALL ALL | H:H145_SDK_DOORS_INSTALLED |
| Cockpit And Pax Doors REMOVE ALL | H:H145_SDK_DOORS_REMOVED |
| Pilot TOGGLE | H:H145_SDK_PILOT_CAPT_TOGGLE |
| Pilot ON | H:H145_SDK_PILOT_CAPT_ON |
| Pilot OFF | H:H145_SDK_PILOT_CAPT_OFF |
| Copilot TOGGLE | H:H145_SDK_PILOT_FO_TOGGLE |
| Copilot ON | H:H145_SDK_PILOT_FO_ON |
| Copilot OFF | H:H145_SDK_PILOT_FO_OFF |
| HEMS Stretcher Toggle | H:H145_SDK_HEMS_STRETCHER_TOGGLE |
| HEMS Stretcher Eject | H:H145_SDK_HEMS_STRETCHER_EJECT |
| HEMS Stretcher Retract | H:H145_SDK_HEMS_STRETCHER_RETRACT |
| HEMS Stretcher Removed | H:H145_SDK_HEMS_STRETCHER_REMOVED |
| HEMS Stretcher Present without patient | H:H145_SDK_HEMS_STRETCHER_NOPATIENT |
| HEMS Stretcher Present with patient | H:H145_SDK_HEMS_STRETCHER_PATIENT |
| Pax 1 Toggle | H:H145_SDK_PAX_1_TOGGLE |

| | |
|--------------|-------------------------|
| Pax 1 On | H:H145_SDK_PAX_1_ON |
| Pax 1 Off | H:H145_SDK_PAX_1_OFF |
| Pax 2 Toggle | H:H145_SDK_PAX_2_TOGGLE |
| Pax 2 On | H:H145_SDK_PAX_2_ON |
| Pax 2 Off | H:H145_SDK_PAX_2_OFF |
| Pax 3 Toggle | H:H145_SDK_PAX_3_TOGGLE |
| Pax 3 On | H:H145_SDK_PAX_3_ON |
| Pax 3 Off | H:H145_SDK_PAX_3_OFF |
| Pax 4 Toggle | H:H145_SDK_PAX_4_TOGGLE |
| Pax 4 On | H:H145_SDK_PAX_4_ON |
| Pax 4 Off | H:H145_SDK_PAX_4_OFF |
| Pax 5 Toggle | H:H145_SDK_PAX_5_TOGGLE |
| Pax 5 On | H:H145_SDK_PAX_5_ON |
| Pax 5 Off | H:H145_SDK_PAX_5_OFF |
| Pax 6 Toggle | H:H145_SDK_PAX_6_TOGGLE |
| Pax 6 On | H:H145_SDK_PAX_6_ON |
| Pax 6 Off | H:H145_SDK_PAX_6_OFF |
| Pax 7 Toggle | H:H145_SDK_PAX_7_TOGGLE |
| Pax 7 On | H:H145_SDK_PAX_7_ON |
| Pax 7 Off | H:H145_SDK_PAX_7_OFF |
| Pax 8 Toggle | H:H145_SDK_PAX_8_TOGGLE |
| Pax 8 On | H:H145_SDK_PAX_8_ON |
| Pax 8 Off | H:H145_SDK_PAX_8_OFF |

Misc

| Name | Event |
|-------------------------------|--|
| State Load READY FOR TAKEOFF | H:H145_SDK_MISC_CMD_READYFORTAKEOFF |
| State Load COLD AND DARK | H:H145_SDK_MISC_CMD_COLDANDDARK |
| Rotor Brake TOGGLE | H:H145_SDK_ROTOR_BRAKE_TOGGLE |
| Rotor Brake ON | H:H145_SDK_ROTOR_BRAKE_ON |
| Rotor Brake OFF | H:H145_SDK_ROTOR_BRAKE_OFF |
| FMS1 Source TOGGLE | H:H145_SDK_MISC_FMS1_TOGGLE |
| FMS1 Source ON | H:H145_SDK_MISC_FMS1_ON |
| FMS1 Source OFF | H:H145_SDK_MISC_FMS1_OFF |
| FMS2 Source TOGGLE | H:H145_SDK_MISC_FMS2_TOGGLE |
| FMS2 Source ON | H:H145_SDK_MISC_FMS2_ON |
| FMS2 Source OFF | H:H145_SDK_MISC_FMS2_OFF |
| Master Brightness Increase | H:H145_SDK_MASTERBRIGHTNESS_INC |
| Master Brightness Decrease | H:H145_SDK_MASTERBRIGHTNESS_DEC |
| Luxury Divider Wall TOGGLE | H:H145_SDK_LUX_DIVIDER_TOGGLE |
| Luxury Divider Wall UP | H:H145_SDK_LUX_DIVIDER_UP |
| Luxury Divider Wall DOWN | H:H145_SDK_LUX_DIVIDER_DOWN |
| TDSSim GTNXi Nav Source UNIT1 | H:H145_SDK_MISC_GTN750_TDSSIM_NAVSOURCE_UNIT_1 |
| TDSSim GTNXi Nav Source UNIT2 | H:H145_SDK_MISC_GTN750_TDSSIM_NAVSOURCE_UNIT_2 |
| TDSSim GTNXi Nav Source MSFS | H:H145_SDK_MISC_GTN750_TDSSIM_NAVSOURCE_MSFS |
| TDSSim GTNXi Nav Source NEXT | H:H145_SDK_MISC_GTN750_TDSSIM_NAVSOURCE_NEXT |

Hoist

| Name | Event |
|-------------------------------------|---|
| Hoist Mode AUTO | H:H145_SDK_HOIST_CONTROL_MODE_AUTO |
| Hoist Mode MANUAL | H:H145_SDK_HOIST_CONTROL_MODE_MANUAL |
| Hoist Manual Control UP | H:H145_SDK_HOIST_CONTROL_MOTOR_UP |
| Hoist Manual Control STOP | H:H145_SDK_HOIST_CONTROL_MOTOR_STOP |
| Hoist Manual Control DOWN | H:H145_SDK_HOIST_CONTROL_MOTOR_DOWN |
| Hoist Manual Control MOMENTARY_UP | H:H145_SDK_HOIST_CONTROL_MOTOR_MOMENTARY_UP |
| Hoist Manual Control MOMENTARY_DOWN | H:H145_SDK_HOIST_CONTROL_MOTOR_MOMENTARY_DOWN |
| Hoist Arm STOW | H:H145_SDK_HOIST_ARM_STOW |
| Hoist Arm DEPLOY | H:H145_SDK_HOIST_ARM_DEPLOY |

Center Console WXRCP

| Name | Event |
|-------------------------------|--------------------------|
| Weather Radar Power OFF | H:H145_SDK_WXR_OFF |
| Weather Radar Power STANDBY | H:H145_SDK_WXR_STBY |
| Weather Radar Power TEST | H:H145_SDK_WXR_TEST |
| Weather Radar Power ON | H:H145_SDK_WXR_ON |
| Weather Radar Power Knob UP | H:H145_SDK_WXR_UP |
| Weather Radar Power Knob NEXT | H:H145_SDK_WXR_UP_LOOP |
| Weather Radar Power Knob DOWN | H:H145_SDK_WXR_DOWN |
| Weather Radar Tilt Knob UP | H:H145_SDK_WXR_TILT_UP |
| Weather Radar Tilt Knob DOWN | H:H145_SDK_WXR_TILT_DOWN |

Search Light

| Name | Event |
|----------------|----------------------------|
| Light TOGGLE | H:H145_SDK_SL_LIGHT_TOGGLE |
| Light OFF | H:H145_SDK_SL_LIGHT_OFF |
| Light ON | H:H145_SDK_SL_LIGHT_ON |
| Steering UP | H:H145_SDK_SL_STEER_UP |
| Steering DOWN | H:H145_SDK_SL_STEER_DOWN |
| Steering LEFT | H:H145_SDK_SL_STEER_LEFT |
| Steering RIGHT | H:H145_SDK_SL_STEER_RIGHT |
| Steering HOME | H:H145_SDK_SL_STEER_HOME |

Landing Light

| Name | Event |
|--------------|-----------------------------|
| Light TOGGLE | H:H145_SDK_LDG_LIGHT_TOGGLE |
| Light OFF | H:H145_SDK_LDG_LIGHT_OFF |
| Light ON | H:H145_SDK_LDG_LIGHT_ON |

Center Console HISLCP

| Name | Event |
|----------------------------|-------------------------------------|
| HISL Deploy or Stow TOGGLE | H:H145_SDK_HISL_STOW_TOGGLE |
| HISL STOW | H:H145_SDK_HISL_STOW |
| HISL DEPLOY | H:H145_SDK_HISL_DEPLOY |
| HISL Dim TOGGLE | H:H145_SDK_HISL_DIM_TOGGLE |
| HISL Dim ON | H:H145_SDK_HISL_DIM_ON |
| HISL Dim OFF | H:H145_SDK_HISL_DIM_OFF |
| HISL Lamp TOGGLE | H:H145_SDK_HISL_LAMP_TOGGLE |
| HISL Lamp ON | H:H145_SDK_HISL_LAMP_ON |
| HISL Lamp OFF | H:H145_SDK_HISL_LAMP_OFF |
| Easy HISL TOGGLE | H:H145_SDK_HISL_EASYTOGGLE |
| Easy HISL OFF | H:H145_SDK_HISL_EASY_OFF |
| Easy HISL ON | H:H145_SDK_HISL_EASY_ON |
| Beam Zoom (Wide) | H:H145_SDK_HISL_ZOOM_WIDE |
| Beam Zoom (Narrow) | H:H145_SDK_HISL_ZOOM_NARROW |
| Filter ENTER | H:H145_SDK_HISL_FILTER_ENTER |
| Filter SELECT | H:H145_SDK_HISL_FILTER_SELECT |
| Directly Select Filter 1 | H:H145_SDK_HISL_FILTER_EASYSELECT_1 |
| Directly Select Filter 2 | H:H145_SDK_HISL_FILTER_EASYSELECT_2 |
| Directly Select Filter 3 | H:H145_SDK_HISL_FILTER_EASYSELECT_3 |
| Directly Select Filter 4 | H:H145_SDK_HISL_FILTER_EASYSELECT_4 |

Tablet

| Name | Event |
|---------------------------------------|--|
| Hinge Open/Close | H:H145_SDK_TABLET_OPENCLOSE |
| Home (Push) | H:H145_SDK_TABLET_HOME_PUSH |
| Home (Push Long) | H:H145_SDK_TABLET_HOME_PUSH_LONG |
| Open Action Center | H:H145_SDK_TABLET_OPEN_ACTIONCENTER |
| Launch Maps | H:H145_SDK_TABLET_OPENAPP_MAPS |
| Launch Missions | H:H145_SDK_TABLET_OPENAPP_MISSIONS |
| Launch Setup | H:H145_SDK_TABLET_OPENAPP_SETUP |
| Launch Documents | H:H145_SDK_TABLET_OPENAPP_DOCUMENTS |
| Launch EFBCConnect | H:H145_SDK_TABLET_OPENAPP_WEB_EFBCCONNECT |
| Launch Web Browser | H:H145_SDK_TABLET_OPENAPP_WEB |
| Launch METAR | H:H145_SDK_TABLET_OPENAPP_METAR |
| Launch LittleNavMap | H:H145_SDK_TABLET_OPENAPP_LITTLENAVMAP |
| Launch Navigraph Charts | H:H145_SDK_TABLET_OPENAPP_NAVIGRAPH |
| Launch Flappy Bird | H:H145_SDK_TABLET_OPENAPP_FLAPPYBIRD |
| Launch Alarms & Clock | H:H145_SDK_TABLET_OPENAPP_CLOCK |
| Launch Activity Log | H:H145_SDK_TABLET_OPENAPP_ACTIVITYLOG |
| Launch Direction Finder | H:H145_SDK_TABLET_OPENAPP_DF |
| Launch Hype Radio | H:H145_SDK_TABLET_OPENAPP_HYPERADIO |
| Launch Neopad | H:H145_SDK_TABLET_OPENAPP_NEOPAD |
| Map ZOOM IN | H:H145_SDK_TABLET_MAPSAPP_ZOOM_IN |
| Map ZOOM OUT | H:H145_SDK_TABLET_MAPSAPP_ZOOM_OUT |
| Map ZOOM Level 3 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_3 |
| Map ZOOM Level 4 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_4 |
| Map ZOOM Level 5 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_5 |
| Map ZOOM Level 6 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_6 |
| Map ZOOM Level 7 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_7 |
| Map ZOOM Level 8 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_8 |
| Map ZOOM Level 9 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_9 |
| Map ZOOM Level 10 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_10 |
| Map ZOOM Level 11 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_11 |
| Map ZOOM Level 12 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_12 |
| Map ZOOM Level 13 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_13 |
| Map ZOOM Level 14 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_14 |
| Map ZOOM Level 15 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_15 |
| Map ZOOM Level 16 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_16 |
| Map ZOOM Level 17 | H:H145_SDK_TABLET_MAPSAPP_ZOOM_SET_17 |
| Map FollowMe TOGGLE | H:H145_SDK_TABLET_MAPSAPP_FOLLOWME_TOGGLE |
| Map FollowMe ON | H:H145_SDK_TABLET_MAPSAPP_FOLLOWME_ON |
| Map FollowMe OFF | H:H145_SDK_TABLET_MAPSAPP_FOLLOWME_OFF |
| Map Orientation TOGGLE | H:H145_SDK_TABLET_MAPSAPP_ORIENTATION_TOGGLE |
| Map Orientation NorthUP | H:H145_SDK_TABLET_MAPSAPP_ORIENTATION_NORTHUP |
| Map Orientation HeadingUP | H:H145_SDK_TABLET_MAPSAPP_ORIENTATION_HEADINGUP |
| Map DB Layer Hospital Helipad ON | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_ON_H_HOSPITAL |
| Map DB Layer Civil Helipad ON | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_ON_H_CIVIL |
| Map DB Layer Airport Primary ON | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_ON_AIRPORT |
| Map DB Layer Airport Secondary ON | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_ON_AIRPORT-NOCODE |
| Map DB Layer Hospital Helipad OFF | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_OFF_H_HOSPITAL |
| Map DB Layer Civil Helipad OFF | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_OFF_H_CIVIL |
| Map DB Layer Airport Primary OFF | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_OFF_AIRPORT |
| Map DB Layer Airport Secondary OFF | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_OFF_AIRPORT-NOCODE |
| Map DB Layer Hospital Helipad TOGGLE | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_TOGGLE_H_HOSPITAL |
| Map DB Layer Civil Helipad TOGGLE | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_TOGGLE_H_CIVIL |
| Map DB Layer Airport Primary TOGGLE | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_TOGGLE_AIRPORT |
| Map DB Layer Airport Secondary TOGGLE | H:H145_SDK_TABLET_MAPSAPP_SET_DB_LAYER_TOGGLE_AIRPORT-NOCODE |
| Mission Command 1 PRESS | H:H145_SDK_MISSION_ACTION_COMMAND_1 |
| Mission Command 2 PRESS | H:H145_SDK_MISSION_ACTION_COMMAND_2 |
| Mission Command 3 PRESS | H:H145_SDK_MISSION_ACTION_COMMAND_3 |
| Mission Command 4 PRESS | H:H145_SDK_MISSION_ACTION_COMMAND_4 |
| Mission Command 5 PRESS | H:H145_SDK_MISSION_ACTION_COMMAND_5 |

Hype Radio App

| Name | Event |
|--------------------------------|-----------------------------------|
| Connect_Reconnect_SyncLocation | H:H145_SDK_HYPERADIO_CONNECT |
| Volume Down | H:H145_SDK_HYPERADIO_VOLUME_DOWN |
| Volume Up | H:H145_SDK_HYPERADIO_VOLUME_UP |
| Stop | H:H145_SDK_HYPERADIO_STOP |
| Select Previous Station | H:H145_SDK_HYPERADIO_STATION_PREV |
| Select Next Station | H:H145_SDK_HYPERADIO_STATION_NEXT |
| Select Station 1 | H:H145_SDK_HYPERADIO_STATION_1 |
| Select Station 2 | H:H145_SDK_HYPERADIO_STATION_2 |
| Select Station 3 | H:H145_SDK_HYPERADIO_STATION_3 |
| Select Station 4 | H:H145_SDK_HYPERADIO_STATION_4 |
| Select Station 5 | H:H145_SDK_HYPERADIO_STATION_5 |
| Select Station 6 | H:H145_SDK_HYPERADIO_STATION_6 |
| Select Station 7 | H:H145_SDK_HYPERADIO_STATION_7 |
| Select Station 8 | H:H145_SDK_HYPERADIO_STATION_8 |
| Select Station 9 | H:H145_SDK_HYPERADIO_STATION_9 |
| Select Station 10 | H:H145_SDK_HYPERADIO_STATION_10 |
| Select Station 11 | H:H145_SDK_HYPERADIO_STATION_11 |
| Select Station 12 | H:H145_SDK_HYPERADIO_STATION_12 |
| Select Station 13 | H:H145_SDK_HYPERADIO_STATION_13 |
| Select Station 14 | H:H145_SDK_HYPERADIO_STATION_14 |
| Select Station 15 | H:H145_SDK_HYPERADIO_STATION_15 |

Equipment Setup

| Name | Event |
|---------------------------------|--|
| Radome TOGGLE | H:H145_SDK_EQUIP_RADOME_TOGGLE |
| Radome 1 ON | H:H145_SDK_EQUIP_RADOME_ON |
| Radome 2 ON | H:H145_SDK_EQUIP_RADOME_2_ON |
| Radome OFF | H:H145_SDK_EQUIP_RADOME_OFF |
| WSPS Top TOGGLE | H:H145_SDK_EQUIP_WSPS_TOP_TOGGLE |
| WSPS Top ON | H:H145_SDK_EQUIP_WSPS_TOP_ON |
| WSPS Top OFF | H:H145_SDK_EQUIP_WSPS_TOP_OFF |
| WSPS Bottom TOGGLE | H:H145_SDK_EQUIP_WSPS_BOTTOM_TOGGLE |
| WSPS Bottom ON | H:H145_SDK_EQUIP_WSPS_BOTTOM_ON |
| WSPS Bottom OFF | H:H145_SDK_EQUIP_WSPS_BOTTOM_OFF |
| WSPS Skid TOGGLE | H:H145_SDK_EQUIP_WSPS_SKID_TOGGLE |
| WSPS Skid ON | H:H145_SDK_EQUIP_WSPS_SKID_ON |
| WSPS Skid OFF | H:H145_SDK_EQUIP_WSPS_SKID_OFF |
| Skid Settling Preventers TOGGLE | H:H145_SDK_EQUIP_SKID_SETTLING_PREVENTERS_TOGGLE |
| Skid Settling Preventers ON | H:H145_SDK_EQUIP_SKID_SETTLING_PREVENTERS_ON |
| Skid Settling Preventers OFF | H:H145_SDK_EQUIP_SKID_SETTLING_PREVENTERS_OFF |
| Air Conditioning TOGGLE | H:H145_SDK_EQUIP_AIRCONDITIONING_TOGGLE |
| Air Conditioning ON | H:H145_SDK_EQUIP_AIRCONDITIONING_ON |
| Air Conditioning OFF | H:H145_SDK_EQUIP_AIRCONDITIONING_OFF |
| Fuel Flow Sensor TOGGLE | H:H145_SDK_EQUIP_FUELFLOWSENSOR_TOGGLE |
| Fuel Flow Sensor ON | H:H145_SDK_EQUIP_FUELFLOWSENSOR_ON |
| Fuel Flow Sensor OFF | H:H145_SDK_EQUIP_FUELFLOWSENSOR_OFF |
| ACAS (Traffic) TOGGLE | H:H145_SDK_EQUIP_ACAS_TOGGLE |
| ACAS (Traffic) ON | H:H145_SDK_EQUIP_ACAS_ON |
| ACAS (Traffic) OFF | H:H145_SDK_EQUIP_ACAS_OFF |
| HTAWS (Terrain) TOGGLE | H:H145_SDK_EQUIP_HTAWS_TOGGLE |
| HTAWS (Terrain) ON | H:H145_SDK_EQUIP_HTAWS_ON |
| HTAWS (Terrain) OFF | H:H145_SDK_EQUIP_HTAWS_OFF |
| IBF (Filter) TOGGLE | H:H145_SDK_EQUIP_IBF_TOGGLE |

| | |
|------------------------------|--|
| IBF (Filter) ON | H:H145_SDK_EQUIP_IBF_ON |
| IBF (Filter) OFF | H:H145_SDK_EQUIP_IBF_OFF |
| Cockpit Weapon Sights TOGGLE | H:H145_SDK_EQUIP_WEAPONS_SIGHT_TOGGLE |
| Cockpit Weapon Sights OFF | H:H145_SDK_EQUIP_WEAPONS_SIGHT_OFF |
| Cockpit Weapon Sights ON | H:H145_SDK_EQUIP_WEAPONS_SIGHT_ON |
| Helmet Display ON | H:H145_SDK_EQUIP_HMD_ON |
| Helmet Display OFF | H:H145_SDK_EQUIP_HMD_OFF |
| Helmet Display TOGGLE | H:H145_SDK_EQUIP_HMD_TOGGLE |
| Bambi Bucket ON | H:H145_SDK_EQUIP_BAMBI_ON |
| Bambi Bucket OFF | H:H145_SDK_EQUIP_BAMBI_OFF |
| Bambi Bucket TOGGLE | H:H145_SDK_EQUIP_BAMBI_TOGGLE |
| Cargo Hook ON | H:H145_SDK_EQUIP_HOOK_ON |
| Cargo Hook OFF | H:H145_SDK_EQUIP_HOOK_OFF |
| Cargo Hook TOGGLE | H:H145_SDK_EQUIP_HOOK_TOGGLE |
| Fabric Glareshields ON | H:H145_SDK_EQUIP_FABRIC_FLARESHIELDS_ON |
| Fabric Glareshields OFF | H:H145_SDK_EQUIP_FABRIC_FLARESHIELDS_OFF |
| Fabric Glareshields TOGGLE | H:H145_SDK_EQUIP_FABRIC_FLARESHIELDS_TOGGLE |
| Sun Visors ON | H:H145_SDK_EQUIP_SUN_VISORS_ON |
| Sun Visors OFF | H:H145_SDK_EQUIP_SUN_VISORS_OFF |
| Sun Visors TOGGLE | H:H145_SDK_EQUIP_SUN_VISORS_TOGGLE |
| Safety Patches ON | H:H145_SDK_EQUIP_SAFETY_PATCHES_ON |
| Safety Patches OFF | H:H145_SDK_EQUIP_SAFETY_PATCHES_OFF |
| Safety Patches TOGGLE | H:H145_SDK_EQUIP_SAFETY_PATCHES_TOGGLE |
| ELT (ADELT) ON | H:H145_SDK_EQUIP_ADELT_ON |
| ELT (ADELT) OFF | H:H145_SDK_EQUIP_ADELT_OFF |
| ELT (ADELT) TOGGLE | H:H145_SDK_EQUIP_ADELT_TOGGLE |
| Hoist ON | H:H145_SDK_EQUIP_HOIST_ON |
| Hoist OFF | H:H145_SDK_EQUIP_HOIST_OFF |
| Hoist TOGGLE | H:H145_SDK_EQUIP_HOIST_TOGGLE |
| HISL ON | H:H145_SDK_EQUIP_HISL_ON |
| HISL OFF | H:H145_SDK_EQUIP_HISL_OFF |
| HISL TOGGLE | H:H145_SDK_EQUIP_HISL_TOGGLE |
| Snow Skis ON | H:H145_SDK_EQUIP_SKID_SKI_ON |
| Snow Skis OFF | H:H145_SDK_EQUIP_SKID_SKI_OFF |
| Snow Skis TOGGLE | H:H145_SDK_EQUIP_SKID_SKI_TOGGLE |
| Emergency Floats ON | H:H145_SDK_EQUIP_SKID_FLOATS_ON |
| Emergency Floats OFF | H:H145_SDK_EQUIP_SKID_FLOATS_OFF |
| Emergency Floats TOGGLE | H:H145_SDK_EQUIP_SKID_FLOATS_TOGGLE |
| Long Skids ON | H:H145_SDK_EQUIP_SKID_LONG_ON |
| Long Skids OFF | H:H145_SDK_EQUIP_SKID_LONG_OFF |
| Long Skids TOGGLE | H:H145_SDK_EQUIP_SKID_LONG_TOGGLE |
| Second Landing Light ON | H:H145_SDK_EQUIP_SECOND_LANDING_LIGHT_ON |
| Second Landing Light OFF | H:H145_SDK_EQUIP_SECOND_LANDING_LIGHT_OFF |
| Second Landing Light TOGGLE | H:H145_SDK_EQUIP_SECOND_LANDING_LIGHT_TOGGLE |
| Chin Window Plates ON | H:H145_SDK_EQUIP_CHIN_WINDOW_PLATES_ON |
| Chin Window Plates OFF | H:H145_SDK_EQUIP_CHIN_WINDOW_PLATES_OFF |
| Chin Window Plates TOGGLE | H:H145_SDK_EQUIP_CHIN_WINDOW_PLATES_TOGGLE |

MFDs

| Name | Event |
|---------------------|-------------------|
| MFD1 SoftKey Top 1 | H:MFD1_SoftKey_T1 |
| MFD1 SoftKey Top 2 | H:MFD1_SoftKey_T2 |
| MFD1 SoftKey Top 3 | H:MFD1_SoftKey_T3 |
| MFD1 SoftKey Top 4 | H:MFD1_SoftKey_T4 |
| MFD1 SoftKey Top 5 | H:MFD1_SoftKey_T5 |
| MFD1 SoftKey Top 6 | H:MFD1_SoftKey_T6 |
| MFD1 SoftKey Left 1 | H:MFD1_SoftKey_L1 |
| MFD1 SoftKey Left 2 | H:MFD1_SoftKey_L2 |
| MFD1 SoftKey Left 3 | H:MFD1_SoftKey_L3 |
| MFD1 SoftKey Left 4 | H:MFD1_SoftKey_L4 |

| | |
|------------------------------------|---------------------------------------|
| MFD1 SoftKey Left 5 | H:MFD1_SoftKey_L5 |
| MFD1 SoftKey Left 6 | H:MFD1_SoftKey_L6 |
| MFD1 SoftKey Right 1 | H:MFD1_SoftKey_R1 |
| MFD1 SoftKey Right 2 | H:MFD1_SoftKey_R2 |
| MFD1 SoftKey Right 3 | H:MFD1_SoftKey_R3 |
| MFD1 SoftKey Right 4 | H:MFD1_SoftKey_R4 |
| MFD1 SoftKey Right 5 | H:MFD1_SoftKey_R5 |
| MFD1 SoftKey Right 6 | H:MFD1_SoftKey_R6 |
| MFD1 SoftKey Bottom 1 | H:MFD1_SoftKey_B1 |
| MFD1 SoftKey Bottom 2 | H:MFD1_SoftKey_B2 |
| MFD1 SoftKey Bottom 3 | H:MFD1_SoftKey_B3 |
| MFD1 SoftKey Bottom 4 | H:MFD1_SoftKey_B4 |
| MFD1 SoftKey Bottom 5 | H:MFD1_SoftKey_B5 |
| MFD1 SoftKey Bottom 6 | H:MFD1_SoftKey_B6 |
| MFD1 Small Knob Clockwise | H:MFD1_SoftKey_KnobInnerClockwise |
| MFD1 Small Knob AntiClockwise | H:MFD1_SoftKey_KnobInnerAntiClockwise |
| MFD1 Small Knob Push | H:MFD1_SoftKey_KnobInnerPush |
| MFD1 Small Knob Push (Long) | H:MFD1_SoftKey_KnobInnerPushLong |
| MFD1 Large Knob Clockwise | H:MFD1_SoftKey_KnobOuterClockwise |
| MFD1 Large Knob AntiClockwise | H:MFD1_SoftKey_KnobOuterAntiClockwise |
| MFD1 LUM (oveall intensity) Up | H:MFD1_SoftKey_LUM_UP |
| MFD1 LUM (oveall intensity) Down | H:MFD1_SoftKey_LUM_DOWN |
| MFD1 BRT (underlay intensity) Up | H:MFD1_SoftKey_BRT_UP |
| MFD1 BRT (underlay intensity) Down | H:MFD1_SoftKey_BRT_DOWN |
| MFD1 CTRS (overlay intensity) Up | H:MFD1_SoftKey_CTRS_UP |
| MFD1 CTRS (overlay intensity) Down | H:MFD1_SoftKey_CTRSW_DOWN |
| MFD1 Power | H:MFD1_SoftKey_POWER |
| MFD2 SoftKey Top 1 | H:MFD2_SoftKey_T1 |
| MFD2 SoftKey Top 2 | H:MFD2_SoftKey_T2 |
| MFD2 SoftKey Top 3 | H:MFD2_SoftKey_T3 |
| MFD2 SoftKey Top 4 | H:MFD2_SoftKey_T4 |
| MFD2 SoftKey Top 5 | H:MFD2_SoftKey_T5 |
| MFD2 SoftKey Top 6 | H:MFD2_SoftKey_T6 |
| MFD2 SoftKey Left 1 | H:MFD2_SoftKey_L1 |
| MFD2 SoftKey Left 2 | H:MFD2_SoftKey_L2 |
| MFD2 SoftKey Left 3 | H:MFD2_SoftKey_L3 |
| MFD2 SoftKey Left 4 | H:MFD2_SoftKey_L4 |
| MFD2 SoftKey Left 5 | H:MFD2_SoftKey_L5 |
| MFD2 SoftKey Left 6 | H:MFD2_SoftKey_L6 |
| MFD2 SoftKey Right 1 | H:MFD2_SoftKey_R1 |
| MFD2 SoftKey Right 2 | H:MFD2_SoftKey_R2 |
| MFD2 SoftKey Right 3 | H:MFD2_SoftKey_R3 |
| MFD2 SoftKey Right 4 | H:MFD2_SoftKey_R4 |
| MFD2 SoftKey Right 5 | H:MFD2_SoftKey_R5 |
| MFD2 SoftKey Right 6 | H:MFD2_SoftKey_R6 |
| MFD2 SoftKey Bottom 1 | H:MFD2_SoftKey_B1 |
| MFD2 SoftKey Bottom 2 | H:MFD2_SoftKey_B2 |
| MFD2 SoftKey Bottom 3 | H:MFD2_SoftKey_B3 |
| MFD2 SoftKey Bottom 4 | H:MFD2_SoftKey_B4 |
| MFD2 SoftKey Bottom 5 | H:MFD2_SoftKey_B5 |
| MFD2 SoftKey Bottom 6 | H:MFD2_SoftKey_B6 |
| MFD2 Small Knob Clockwise | H:MFD2_SoftKey_KnobInnerClockwise |
| MFD2 Small Knob AntiClockwise | H:MFD2_SoftKey_KnobInnerAntiClockwise |
| MFD2 Small Knob Push | H:MFD2_SoftKey_KnobInnerPush |
| MFD2 Small Knob Push (Long) | H:MFD2_SoftKey_KnobInnerPushLong |
| MFD2 Large Knob Clockwise | H:MFD2_SoftKey_KnobOuterClockwise |
| MFD2 Large Knob AntiClockwise | H:MFD2_SoftKey_KnobOuterAntiClockwise |
| MFD2 LUM (oveall intensity) Up | H:MFD2_SoftKey_LUM_UP |
| MFD2 LUM (oveall intensity) Down | H:MFD2_SoftKey_LUM_DOWN |
| MFD2 BRT (underlay intensity) Up | H:MFD2_SoftKey_BRT_UP |
| MFD2 BRT (underlay intensity) Down | H:MFD2_SoftKey_BRT_DOWN |
| MFD2 CTRS (overlay intensity) Up | H:MFD2_SoftKey_CTRS_UP |
| MFD2 CTRS (overlay intensity) Down | H:MFD2_SoftKey_CTRSW_DOWN |

| | |
|------------------------------------|---------------------------------------|
| MFD2 Power | H:MFD2_SoftKey_POWER |
| MFD4 SoftKey Top 1 | H:MFD4_SoftKey_T1 |
| MFD4 SoftKey Top 2 | H:MFD4_SoftKey_T2 |
| MFD4 SoftKey Top 3 | H:MFD4_SoftKey_T3 |
| MFD4 SoftKey Top 4 | H:MFD4_SoftKey_T4 |
| MFD4 SoftKey Top 5 | H:MFD4_SoftKey_T5 |
| MFD4 SoftKey Top 6 | H:MFD4_SoftKey_T6 |
| MFD4 SoftKey Left 1 | H:MFD4_SoftKey_L1 |
| MFD4 SoftKey Left 2 | H:MFD4_SoftKey_L2 |
| MFD4 SoftKey Left 3 | H:MFD4_SoftKey_L3 |
| MFD4 SoftKey Left 4 | H:MFD4_SoftKey_L4 |
| MFD4 SoftKey Left 5 | H:MFD4_SoftKey_L5 |
| MFD4 SoftKey Left 6 | H:MFD4_SoftKey_L6 |
| MFD4 SoftKey Right 1 | H:MFD4_SoftKey_R1 |
| MFD4 SoftKey Right 2 | H:MFD4_SoftKey_R2 |
| MFD4 SoftKey Right 3 | H:MFD4_SoftKey_R3 |
| MFD4 SoftKey Right 4 | H:MFD4_SoftKey_R4 |
| MFD4 SoftKey Right 5 | H:MFD4_SoftKey_R5 |
| MFD4 SoftKey Right 6 | H:MFD4_SoftKey_R6 |
| MFD4 SoftKey Bottom 1 | H:MFD4_SoftKey_B1 |
| MFD4 SoftKey Bottom 2 | H:MFD4_SoftKey_B2 |
| MFD4 SoftKey Bottom 3 | H:MFD4_SoftKey_B3 |
| MFD4 SoftKey Bottom 4 | H:MFD4_SoftKey_B4 |
| MFD4 SoftKey Bottom 5 | H:MFD4_SoftKey_B5 |
| MFD4 SoftKey Bottom 6 | H:MFD4_SoftKey_B6 |
| MFD4 Small Knob Clockwise | H:MFD4_SoftKey_KnobInnerClockwise |
| MFD4 Small Knob AntiClockwise | H:MFD4_SoftKey_KnobInnerAntiClockwise |
| MFD4 Small Knob Push | H:MFD4_SoftKey_KnobInnerPush |
| MFD4 Small Knob Push (Long) | H:MFD4_SoftKey_KnobInnerPushLong |
| MFD4 Large Knob Clockwise | H:MFD4_SoftKey_KnobOuterClockwise |
| MFD4 Large Knob AntiClockwise | H:MFD4_SoftKey_KnobOuterAntiClockwise |
| MFD4 LUM (oveall intensity) Up | H:MFD4_SoftKey_LUM_UP |
| MFD4 LUM (oveall intensity) Down | H:MFD4_SoftKey_LUM_DOWN |
| MFD4 BRT (underlay intensity) Up | H:MFD4_SoftKey_BRT_UP |
| MFD4 BRT (underlay intensity) Down | H:MFD4_SoftKey_BRT_DOWN |
| MFD4 CTRS (overlay intensity) Up | H:MFD4_SoftKey_CTRS_UP |
| MFD4 CTRS (overlay intensity) Down | H:MFD4_SoftKey_CTRSW_DOWN |
| MFD4 Power | H:MFD4_SoftKey_POWER |

IESI

| Name | Event |
|-------------------------|------------------------------------|
| Baro Knob Clockwise | H:H145_SDK_IESI_BARO_CLOCKWISE |
| Baro Knob AntiClockwise | H:H145_SDK_IESI_BARO_ANTICLOCKWISE |
| Baro STD | H:H145_SDK_IESI_BARO_STD |
| Cage | H:H145_SDK_IESI_CAGE |
| Brightness Up | H:H145_SDK_IESI_BRT_UP |
| Brightness Down | H:H145_SDK_IESI_BRT_DOWN |

Center Console Other

| Name | Event |
|---------------------------------|-----------------------------|
| ELTCP ELT ON | H:H145_SDK_ELT_SWITCH_ON |
| ELTCP ELT ARM | H:H145_SDK_ELT_SWITCH_ARM |
| ELTCP ELT RESET | H:H145_SDK_ELT_SWITCH_RESET |
| GPUCP Ground Power (LIGHTS) ON | H:H145_SDK_GPCP_PWR_ON |
| GPUCP Ground Power (LIGHTS) OFF | H:H145_SDK_GPCP_PWR_OFF |
| AIRCP DEFOG ON | H:H145_SDK_AIRCP_DEFOG_ON |
| AIRCP DEFOG OFF | H:H145_SDK_AIRCP_DEFOG_OFF |
| AIRCP AIR MIX ON | H:H145_SDK_AIRCP_AIRMIX_ON |
| AIRCP AIR MIX OFF | H:H145_SDK_AIRCP_AIRMIX_OFF |

Sensor Pod

| Name | Event |
|--------------|---|
| Power TOGGLE | H:H145_SDK_SENSORPOD_MONITOR_POWER_TOGGLE |
| Power ON | H:H145_SDK_SENSORPOD_MONITOR_POWER_ON |
| Power OFF | H:H145_SDK_SENSORPOD_MONITOR_POWER_OFF |
| Move RIGHT | H:H145_SDK_SENSORPOD_MOVE_RIGHT |
| Move LEFT | H:H145_SDK_SENSORPOD_MOVE_LEFT |
| Move FORWARD | H:H145_SDK_SENSORPOD_MOVE_FWD |
| Move AFT | H:H145_SDK_SENSORPOD_MOVE_AFT |

System Failures

Note that more failures are directly writable to their L:Var.

| Name | Event |
|-------------------------|-------------------------------------|
| Engine 1 Failure ON | H:H145_SDK_FAILURE_ENG1_FAIL_ON |
| Engine 1 Failure OFF | H:H145_SDK_FAILURE_ENG1_FAIL_OFF |
| Engine 1 Failure TOGGLE | H:H145_SDK_FAILURE_ENG1_FAIL_TOGGLE |
| Engine 2 Failure ON | H:H145_SDK_FAILURE_ENG2_FAIL_ON |
| Engine 2 Failure OFF | H:H145_SDK_FAILURE_ENG2_FAIL_OFF |
| Engine 2 Failure TOGGLE | H:H145_SDK_FAILURE_ENG2_FAIL_TOGGLE |
| Engine 1 FIRE ON | H:H145_SDK_FAILURE_ENG1_FIRE_ON |
| Engine 1 FIRE OFF | H:H145_SDK_FAILURE_ENG1_FIRE_OFF |
| Engine 1 FIRE TOGGLE | H:H145_SDK_FAILURE_ENG1_FIRE_TOGGLE |
| Engine 2 FIRE ON | H:H145_SDK_FAILURE_ENG2_FIRE_ON |
| Engine 2 FIRE OFF | H:H145_SDK_FAILURE_ENG2_FIRE_OFF |
| Engine 2 FIRE TOGGLE | H:H145_SDK_FAILURE_ENG2_FIRE_TOGGLE |
| Mast Moment Exceed OFF | H:H145_SDK_MASTMOMENT_EXCEED_OFF |
| Mast Moment Exceed ON | H:H145_SDK_MASTMOMENT_EXCEED_ON |
| Fire Bottle 1 EMPTY | H:H145_SDK_FIREBOTTLE1_EMPTY |
| Fire Bottle 1 CHARGED | H:H145_SDK_FIREBOTTLE1_FULL |
| Fire Bottle 2 EMPTY | H:H145_SDK_FIREBOTTLE2_EMPTY |
| Fire Bottle 2 CHARGED | H:H145_SDK_FIREBOTTLE2_FULL |

H145 Mission System Documentation

Diese Dokumentation ist vorläufig und kann sich ändern.

Last Update: 2022/6/23

Grundlegende Details zu der Mission

Eine Missions-Json-Datei wird als Missionsdeskriptor bezeichnet. Sie kann in H145 geladen werden und dann allein arbeiten, während der Benutzer die Mission durchführt.

| | |
|-------------|---|
| Titel | Titel, der verwendet wird, wenn Ihre Mission in einer Liste angezeigt wird |
| aircraft | Muss H145 sein (Reihe der unterstützten Flugzeuge) |
| applicable | Array von Varianten. Wird diese Angabe weggelassen, werden alle Varianten angewendet. Nicht zutreffende Missionen werden im Missionskatalog ausgeblendet. EMS FIREFIGHTER |
| api_version | Muß 0.1 sein |
| start_info | Der Startort oder die Startorte können angegeben werden. Dadurch wird verhindert, dass die Mission in der Bibliothek angezeigt wird, da sie einen natürlichen Startpunkt auf der Karte hat. Wenn Sie keine start_info angeben, verwenden Sie die Bibliothek, um Ihre Mission zu beginnen. location Geben Sie [lat, lon] für den festen Startpunkt an. icon_src Geben Sie einen HTTPS- oder Daten-URI an. Dieses Symbol wird auf der Karte angezeigt. Vorgeschlagene Größe 32x32px. query Eine Datenabfrage in demselben Format wie unten in Missionen verwendet |

Missionen von einem Server laden

Um Missionen von einem Server zu laden, geben Sie keine Orte/Objekte/Threads/Objekte an, sondern eine URL, die ein Websocket-Server ist. Wenn der Benutzer die Mission auswählt, wird Ihr Server kontaktiert, und ab diesem Zeitpunkt können Sie das Missionssystem auf unbestimmte Zeit verwalten, bis der Benutzer manuell eine andere Mission auswählt.

| | |
|-----|-------------------|
| Url | "localhost:40510" |
|-----|-------------------|

Erstellen von Missionspaketen

Missionen können zu jedem anderen Community-Paket hinzugefügt oder alleine erstellt werden. Das einzige, was Sie tun müssen, ist einen hpgmission-Ordner innerhalb Ihres Pakets zu erstellen und eine Ordnerhierarchie mit Ihren json-Dateien darunter zu platzieren. Alle Inhalte (Ordner und json-Dateien) unterhalb von hpgmission in allen Community-Paketen werden in die Katalogliste aufgenommen. Es steht Ihnen frei, eine Ordnerstruktur für Regionen zu erstellen oder anderweitig für Ordnung zu sorgen.

Missions Abschnitte

| | |
|-------------|--|
| locations | Tabelle der Orte, auf die in der Missionsdatei verwiesen wird. Diese sind Orte wie "Unfall_Ort" oder "Krankenhaus_Helipad", die die Koordinaten markieren. Sie können einen Ort aus Bing Maps oder Google Maps einfach kopieren/einfügen, indem Sie mit der rechten Maustaste klicken und die Koordinaten aus dem Menü auswählen. |
| objects | Tabelle der dynamischen Objekte, die beim Start der Mission erstellt werden. Die Objekte haben einen Titel, der sie in MSFS identifiziert (wie ein Flugzeug), und sie haben einen vorgegebenen Ort, an dem Sie sie platzieren können. |
| threads | Tabelle der Hintergrundprozesse, die unabhängig vom aktuellen Ziel ablaufen. Dies ermöglicht die parallele Verarbeitung der Logik. Sie können warten, bis eine bestimmte Variable den Wert "true" annimmt, eine bestimmte Verarbeitung durchführen und dann für immer aufhören oder den Prozess erneut starten. Dies kann dazu verwendet werden, Auslöser zu entwerfen und andere Logik zu Ihrer Mission hinzuzufügen, wie z. B. das Aktivieren einer Sequenz von Ereignissen nur dann, wenn der Benutzer einen Bereich betritt, unabhängig davon, wo er sich in der Liste der Missionsziele befindet. |
| objectives | Liste der aufeinanderfolgenden Aufgaben, die der Benutzer abarbeitet. Jede Mission hat mindestens ein Ziel, und wenn die Liste der Ziele vollständig ist, hat der Benutzer die Mission beendet. Jedes Ziel selbst ist eine Reihe von Befehlen, die nacheinander ausgeführt werden. Sie können den Benutzer zu einem Gebiet leiten und erst dann mit dem nächsten Ziel fortfahren, wenn er das betreffende Gebiet erreicht hat. |
| userActions | TODO – Noch nicht dokumentiert |

OBJECT

Die Objekte werden beim Start der Mission erstellt und während der gesamten Mission manipuliert. Die Variable VAR 1 wird üblicherweise zur Konfiguration des visuellen Zustands des Objekts verwendet.

| | | |
|----------|-------------|--|
| title | String | Titel aus einer aircraft.cfg, registriert im MSFS. Siehe den Abschnitt <u>benutzerdefinierten Objekten</u> |
| location | LOCATIONREF | Ort, an dem das Objekt erstellt werden soll. Optional: Objekte ohne Standort werden auf der Nullinsel [0, 0] erstellt und können später mit move_object verschoben werden. |

Spezielle Objektvariablen

Diese Variablen werden vom System auf eine besondere Weise interpretiert.

| Name | Function |
|----------------|---|
| VAR 1 VAR 2 | Auf Simulationsvariablen abgebildet, die für das Objekt eindeutig sind: VAR 1: (A:GENERAL ENG THROTTLE LEVER POSITION:1, Prozent) VAR 2: (A:SPOILERS LEFT POSITION, Prozent) Diese Variablen sind für jedes Objekt eindeutig und werden in der XML-Datei für das Modellverhalten verfügbar sein. Dadurch kann jedes Objekt unabhängige visuelle Zustände und Verhaltensweisen haben. |
| COUPLED | Objekt-Benutzer-Kopplungsmodus. Wenn ein Objekt gekoppelt ist, wird es automatisch auf der Grundlage des Kopplungsstatus geändert. 0: Keine Kopplung 1: Kopplung an Hubwerksposition - Das Objekt wird kontinuierlich an die Position unterhalb des Hebezeugs gekoppelt. 2: Kopplung an die Position der externen Ladung - Das Objekt wird kontinuierlich an der Position unterhalb des Lasthakens eingerastet. 3: Externe Ladungsposition Auto-Kopplung aktiviert - Das Objekt schaltet automatisch in den Kopplungsmodus 2, wenn es in Reichweite ist. 4: Feuerlöschziel (Feuer) - Der Benutzer kann den Bambi-Eimer verwenden, um VAR 1 (Feuermenge) für dieses Ziel zu reduzieren. VAR 2 wird auf die letzte Mengenreduzierung durch die Eimer Entladung gesetzt. 5. Feuerlöschpool - VAR 1: Radius des Pools (METER). VAR2: Tiefe des Beckens (FEET, negativ) |

| | |
|--|---|
| MODE | <p>Objektmodus. Der Modus wird verwendet, um die Physik und das Verhalten des Objekts zu steuern.</p> <p>0: Position am Boden halten</p> <p>1: Neupositionierungsmodus</p> <ul style="list-style-type: none"> - Verwenden Sie LAT/LON, um die nächste Position zu konfigurieren, und setzen Sie dann den Modus auf 0, um wieder zum Halten am Boden zu wechseln. <p>2: 3-Achsen-Geschwindigkeitssteuerung</p> <ul style="list-style-type: none"> - Verwenden Sie VELOCITY X, VELOCITY Y und VELOCITY Z, um die Objektphysik über die Zeit zu steuern. <p>3: MSFS-Standard-Physik</p> |
| WP INDEX | <p>Aktivierungs-Navigationsindex. Setzen Sie den Index 1, um das Wegpunktsystem zu aktivieren und das Objekt zu veranlassen, sich um die Gierachse zu drehen, um sich so zu orientieren, dass die Geschwindigkeit z das Objekt zum Wegpunkt führt.</p> <p>0: nicht aktiv</p> <p>1-5: Navigation zum Wegpunkt 1-5.</p> <p>Das Wegpunktsystem setzt den WP INDEX auf 0, wenn der nächste Wegpunkt (WP INDEX + 1) ein Wegpunkt an der Position 0,0 ist. Die Wegpunkt-Engine setzt zu diesem Zeitpunkt auch VELOCITY Z auf 0.</p> |
| VELOCITY X VELOCITY Y VELOCITY Z | <p>Objektgeschwindigkeiten. Nur anwendbar, wenn MODE=2. Diese Geschwindigkeiten werden direkt an MSFS gesendet, um die Objektbewegung anzuweisen.</p> |

THREAD

Threads sind Hintergrund-Befehlslisten, die unabhängig von dem gerade aktiven Ziel ausgeführt werden. Threads können zur Planung von Aktivitäten verwendet werden, unabhängig davon, wo sich der Benutzer in der Aufgabenliste befindet.

| | | |
|----------|--------------|---|
| interval | milliseconds | Update Interval (je höher, desto besser für die Leistung) |
| commands | COMMANDLIST | Liste der Befehle, die der Reihe nach ausgeführt werden. |

OBJECTIVE

Ihre Mission muss mindestens ein Ziel haben, sonst wird sie sofort nach dem Start beendet. Die Ziele haben jeweils eine Liste von Befehlen, und wenn ein Ziel abgeschlossen ist, wird der erste Befehl im nächsten Ziel gestartet. Wenn der letzte Befehl des letzten Ziels beendet ist, ist die

Mission abgeschlossen und endet.

| | | |
|----------|-------------|--|
| title | string | Text, der dem Benutzer für dieses Ziel angezeigt werden soll |
| commands | COMMANDLIST | Liste der Befehle, die der Reihe nach ausgeführt werden. |

Commands

Die Befehle werden nacheinander aus einer Befehlsliste ausgeführt, und jeder Befehl kann sofort ausgeführt werden oder eine gewisse Zeit dauern, bis er fertig ist. Siehe API-Referenz [COMMAND](#), [QUERY](#) and [LOCATION](#)

Dynamic Object Library

H145 Crew

Das H145 Crew-Objekt enthält die Besatzung, die Piloten und die Trage. Die folgenden visuellen Zustände können für die verschiedenen Zustände Stehen/Gehen/Winken konfiguriert werden.

| | |
|-------|--|
| title | \$TITLE Crew Airbus H145 ADAC Crew Airbus H145 DRF Crew Airbus H145 CMH Crew Airbus H145 HeliOtago Crew Airbus H145 Norsk Luftambulanse Crew Airbus H145 Bundeswehr Crew Airbus H145 CAL FIRE Crew Airbus H145 San Diego Gas Electric Crew |
|-------|--|

HINWEIS: Livery-Autoren sollten ihren Titel hinzufügen, damit `$TITLE Crew` funktionieren kann. Dieser wird automatisch auf der Grundlage des Livery-Namens ersetzt, und es wird geprüft, ob der Livery-Autor einen Ersatz für den Crew-Titel in seiner Livery-Json-Datei angegeben hat. Weitere Informationen hierzu finden Sie im Abschnitt Livery-Autoren im Hauptbenutzerhandbuch.

Sichtbare Zustände

| | |
|-------|--|
| VAR 1 | -1: Hidden 0: HEMS stehend 1: HEMS stehend mit (backpack) 2: HEMS gehend 3: HEMS gehend mit (backpack) 4: HEMS auf dem Boden kauern |
|-------|--|

| | |
|-------|---|
| | 5: HEMS auf dem Boden kauern mit (backpack) 6: HEMS auf dem Boden kauern mit (backpack auf dem Boden) 7: HEMS wartend 8: Krankentrage ohne Patient 9: Krankentrage mit Patient 10: Krankentrage gehend ohne Patient 11: Krankentrage gehend mit Patient 12: Krankentrage stehend1 ohne Patient 13: Krankentrage stehend1 mit Patient 14: Pilot stehend 15: Pilot winkend 16: Pilot laufend |
| VAR 2 | Gilt nur für VAR 1-Werte von 14-17. 0: Schwarzer Pilot mit Kopfhörer 1: Schwarzer Pilot mit Helm 2: Weißer Pilot mit Kopfhörer 3: Weißer Pilot mit Helm |

H145 verletzter Mensch

Das verletzte menschliche Objekt ist ein Mensch, der auf dem Boden liegt und auf medizinische Hilfe wartet.

| | |
|-------|---------------------------|
| title | Airbus H145 Injured Human |
|-------|---------------------------|

Visual states

| | |
|-------|--|
| VAR 1 | -1: unterdrückt 0: Verletzter Mensch mit Schmerzen 1: Verletzter Mensch auf einer hebbaren Trage |
|-------|--|

H145 Winkender Zivilist

Der winkende Zivilist ist ein Mensch, der winkend steht und versucht, Hilfe für seinen verletzten Freund zu holen.

| | |
|-------|-----------------------------|
| title | Airbus H145 Waving Civilian |
|-------|-----------------------------|

Visual states

| | | |
|------------------------------------|--|-------------------|
| Inhaltsverzeichnis | | Seite 134 von 137 |
|------------------------------------|--|-------------------|

| | |
|-------|---|
| VAR 1 | -1: versteckt 0: Zivilist winkend HINWEIS: Verwenden Sie L:WAVING_CIVILIAN_STOP auf 1, um das Winken zu beenden. |
|-------|---|

H145 Leuchtsignal

Dies ist eine Seenotfackel mit orangefarbenem Rauch.

| | |
|-------|-------------------|
| title | Airbus H145 Flare |
|-------|-------------------|

Visual states

| | |
|-------|--|
| VAR 1 | -1: versteckt 0: Rauchautomatik (EIN für hohe Sichtbarkeit, AUS für Realismus) 1: Rauch ein (ON für beide Einstellungen) |
|-------|--|

Erstellen von benutzerdefinierten dynamischen Objekten

Sie können Ihre eigenen dynamischen Missionsobjekte erstellen, die H145 erstellen kann. Sie können die gleichen COUPED- und MODE-Flags verwenden wie die eingebauten Objekte.

Entpacken Sie das **Mission Object Sample** aus **Tools**. Das Beispiel enthält ein Blender-Asset, das bereits für Sie in das MSFSPackage exportiert wurde. Dabei handelt es sich um ein SDK-Projekt, das Sie in MSFS laden, um das Asset zu kompilieren und ein Paket für den Weitervertrieb zu erstellen.

Die Vorgehensweise ist wie folgt:

- Bereiten Sie ein Asset vor. Nehmen Sie das `Blender-Asset\Ambulance.blend` als Beispiel.
- Exportieren Sie Ihr Asset in `MSFSPackage\PackageSources\SimObjects\Airplanes\sample-ambulance\model\H145_GenericVehicle`
- Aktivieren Sie in MSFS den Entwicklermodus und laden Sie das Projekt `MSFSPackage\MSFS_DynamicObjectSample.xml`
- Kopieren Sie das Ausgabepaket `hype-mission-dynamicobjectsample` aus `MSFSPackage\Packages` in Ihren Community-Ordner.

Jetzt ist das Objekt im Simulator registriert und kann erstellt werden. Verwenden Sie im Szenario-Editor die Symbolleiste "More Objects" und suchen Sie in der Liste "Sample Ambulance". Das Objekt kann nun platziert und in H145-Missionen verwendet werden.

Wenn Sie mehrere Objekte verpacken möchten, müssen Sie den Namen ändern. Um den Namen Ihres Objekts zu ändern, müssen Sie diese Orte unter `MSFSPackage\PackageSources` bearbeiten:

| Datei | Zu ändernder Text |
|---|---|
| <code>ExtraFiles\hpgmission\packageObjects.objmeta</code> | Airbus H145 Ambulance Sample |
| <code>SimObjects\Airplanes\sample-ambulance\aircraft.cfg</code> | Airbus H145 Ambulance Sample Tipp: <code>isUserSelectable=1</code> ermöglicht es Ihnen, das Objekt direkt zu sehen, und <code>isUserSelectable=0</code> stellt sicher, dass Ihr verteiltes Paket keine zusätzlichen Dinge enthält, die im Flugzeugauswahlmenü für den Endbenutzer angezeigt werden. |

Um mehrere Assets in einem Paket zu kombinieren, verwenden Sie `MSFSLayoutGenerator.exe`, um die Datei `thelayout.json` zu aktualisieren, nachdem Sie alle Ausgabeordner zusammengeführt haben.

Mission Server

Ein Missionsserver kann dynamisch Missionsbeschreibungen erstellen und anwenden sowie andere Befehle senden und den Status beobachten. Der Server ist im Wesentlichen nur ein WebSocket-Server, der darauf wartet, dass der Simulator eine Verbindung herstellt und dann über ein Protokoll vom Typ JSON RPC kommuniziert.

Ein sehr einfaches **Mission Server-Beispiel** in `node.js` ist im **Tools-Ordner** enthalten.

Vom H145 an den Missionsserver gesendete Befehle

| | |
|--|---|
| <code>{control_msg: "hello"}</code> | Nach dem Herstellen der Verbindung wird der H145 Sie darauf hinweisen, dass er bereit ist, eine Mission zu senden. |
| <code>{control_msg:"canceled_by_user"}</code> | Der H145 macht Sie darauf aufmerksam, dass der Benutzer eine andere Mission ausgewählt hat und Sie nicht mehr aktiv sind. Die Verbindung wird nach dieser Meldung getrennt |
| <code>{remote_notify: "tag_name", params:[QUERY1, QUERY2, ...]}</code> | Gesendet von der aktiven H145-Mission. Dies sind Daten, über die Sie informiert werden möchten. <code>Remote_notify</code> kann innerhalb von Zielen verwendet oder in einem Hintergrund-Thread konfiguriert werden, um Benachrichtigungen für bestimmte Bedingungen und Daten bereitzustellen. |

Vom Server an den H145 gesendete Befehle

| | |
|---|---|
| <code>{load_mission: MISSION_DESCRIPTOR}</code> | Fordert die H145 auf, die aktuelle Mission zu |
|---|---|

| | |
|--|--|
| | löschen und dann sofort die neue Mission zu laden. |
| {exec_commands: [COMMAND1, COMMAND2, ...]} | Veranlasst den H145, eine eigenständige Befehlsliste auszuführen. Diese Liste wird parallel zum aktuellen Ziel und allen Hintergrund-Threads ausgeführt. |