

Project CARS - Dedicated Server Values and Types

The dedicated server exposes data of various types in its config, and through the HTTP API and Lua API. This document lists what kind of attributes and events the server defines, and all relevant data types.

These lists are useful for addon developers, but also to users configuring their dedicated servers. The writable session attributes are used in the server config and the sms_rotate addon. The track lists, vehicle lists and so on can then be used to determine the valid values for individual setup attributes.

If you have the HTTP API enabled, you can also query a running server for the most up to date attribute, enum and type lists, at the HTTP endpoint /api/lists. So assuming the server is configured with the default bind address and port ("httpApiInterface" and "httpApiPort" in the config file), you can view these lists in JSON format from the computer where the server is running at <http://127.0.0.1:9000/api/list>. You can use any web browser to view the list, or program like wget or curl to download the list.

[Project CARS - Dedicated Server Values and Types](#)

[Attributes](#)

[Session Attributes](#)

[Member Attributes](#)

[Participant Attributes](#)

[Events](#)

[Session Events](#)

[SessionSetup](#)

[StateChanged](#)

[StageChanged](#)

[SessionCreated](#)

[SessionDestroyed](#)

[ServerChat](#)

[Member Events](#)

[PlayerJoined](#)

[Authenticated](#)

[PlayerLeft](#)

[PlayerChat](#)

[Participant Events](#)

[ParticipantCreated](#)

[ParticipantDestroyed](#)

[Sector](#)

- [Lap](#)
- [State](#)
- [Results](#)
- [Impact](#)
- [CutTrackStart](#)
- [CutTrackEnd](#)

- [Track List](#)

- [Vehicle List](#)

- [Vehicle Classes](#)

- [Enums](#)

- [Damage](#)

- [TireWear](#)

- [FuelUsage](#)

- [Penalties](#)

- [AllowedView](#)

- [Weather](#)

- [Flags](#)

- [SessionFlags](#)

- [PlayerFlags](#)

Attributes

Attributes contain setup and status of the session, its members (players) and participants (vehicles). The attributes are read-only unless specified otherwise in their description. Note that boolean-style attributes are represented as single-byte integers, with zero meaning false and non-zero meaning true.

Session Attributes

These are global attributes associated with the session itself.

int8 ServerControlsSetup

Does the server control the game's setup? If yes, the host's lobby settings will be disabled. Set automatically from the server's settings, can't be modified by HTTP or Lua APIs. (ReadWrite)

int8 ServerControlsTrack

Does the server control the track selection? If yes, everyone's track selection will be disabled. (ReadWrite)

int8 ServerControlsVehicleClass

Does the server control the vehicle class selection? If yes, everyone's vehicle selection will be restricted to vehicles from class VehicleClassId, any attempts to change to another class via the garage will not work. Should be always combined with the FORCE_SAME_VEHICLE_CLASS session flag. (ReadWrite)

int8 ServerControlsVehicle

Does the server control the vehicle selection? If yes, everyone's vehicle selection will be restricted to vehicle VehicleModelId. Should be always combined with the FORCE_IDENTICAL_VEHICLE session flag. (ReadWrite)

int8 GridSize

Total maximum grid size, including AI participants. Can be modified only for the 'next session' and can never be larger than the maximum session size set in the server config. (ReadWrite)

int8 MaxPlayers

Maximum number of players. Can't be higher than GridSize, but can be lower and then the extra slots will be reserved for AI vehicles. Can be modified only for the 'next session'. (ReadWrite)

int8 OpponentDifficulty

Difficulty of AI participants. Valid values in range 0-100. (ReadWrite)

int32 Flags

Session setup flags. The flag values are available also via HTTP API at `/api/list/flags/session` or in Lua global tables `lists.flags.session`, `SessionFlags` and `value_to_session_flag`. (ReadWrite)

int16 Practice1Length

Length of first practice session in minutes. (ReadWrite)

int16 Practice2Length

Length of second practice session in minutes. (ReadWrite)

int16 QualifyLength

Length of qualification session in minutes. (ReadWrite)

int16 WarmupLength

Length of warmup session in minutes. (ReadWrite)

int16 Race1Length

Race 1 length in lap count. (ReadWrite)

int16 Race2Length

Race 2 length in lap count. Currently not used in multiplayer.

int8 Privacy

Session privacy, defined when the session is created. Usually server-hosted sessions will be public and use password-protection or blacklists/whitelists to control who can join them. The possible values are 0 to 2, with 0 meaning public, 1 for friends only and 2 for private.

int8 DamageType

Damage type. The enum values are available also via HTTP API at `/api/list/enums/damage` or in Lua global tables `lists.enums.damage`, `Damage` and `value_to_damage`. (ReadWrite)

int8 TireWearType

Tire wear type. The enum values are available also via HTTP API at `/api/list/enums/tire_wear` or in Lua global tables `lists.enums.tire_wear`, `TireWear` and `value_to_tire_wear`. (ReadWrite)

int8 FuelUsageType

Fuel usage type. The enum values are available also via HTTP API at `/api/list/enums/fuel_usage` or in Lua global tables `lists.enums.fuel_usage`, `FuelUsage` and `value_to_fuel_usage`. (ReadWrite)

int8 PenaltiesType

Penalties type. The enum values are available also via HTTP API at </api/list/enums/penalties> or in Lua global tables `lists.enums.penalties`, `Penalties` and `value_to_penalties`. (ReadWrite)

int8 AllowedViews

Allowed cameras. Only Any (0) and CockpitHelmet (2) are supported by the game, writes of other values will be ignored. The enum values are available also via HTTP API at /api/list/enums/allowed_view or in Lua global tables `lists.enums.allowed_view`, `AllowedView` and `value_to_allowed_view`. (ReadWrite)

int32 TrackId

Track id. Players will be able to change this unless `ServerControlsTrack` is set to non-zero. The list of all valid tracks is available also via HTTP API at </api/list/tracks> or in Lua global tables `lists.tracks`, `id_to_track` and `name_to_track`. (ReadWrite)

int32 VehicleClassId

Vehicle class id, used when forcing vehicle class (session flag `FORCE_SAME_VEHICLE_CLASS` is set). If controlled from the server also set attribute `ServerControlsVehicleClass`. Latest class list is available also via HTTP API at /api/list/vehicle_classes and in Lua global tables `lists.vehicle_classes`, `id_to_vehicle_class` and `name_to_vehicle_class`. (ReadWrite)

int32 VehicleModelId

Vehicle model id, used when forcing identical vehicles (session flag `FORCE_IDENTICAL_VEHICLES` is set). If controlled from the server also set attribute `ServerControlsVehicle`. Latest vehicle list is available also via HTTP API at </api/list/vehicles> and in Lua global tables `lists.vehicles`, `id_to_vehicle` and `name_to_vehicle`. (ReadWrite)

int16 DateYear

Starting race date - year. (ReadWrite)

int8 DateMonth

Starting race date - month. (ReadWrite)

int8 DateDay

Starting race date - day of month. (ReadWrite)

int8 DateHour

Starting race date - hour of day. (ReadWrite)

int8 DateMinute

Starting race date - minutes. (ReadWrite)

int8 DateProgression

Race date/time progression multiplier. Only some values between 1 and 60 are valid, the game might change the requested value to closest valid one. (ReadWrite)

int8 ForecastProgression

Time-based forecast progression multiplier. Only some values between 1 and 30 are valid, the game might change the requested value to closest valid one. (ReadWrite)

int8 WeatherSlots

Number of weather slots in use. Set to 0 to use real weather, and to a number between 1 and 4 to interpolate between predefined templates in WeatherSlotN. (ReadWrite)

int32 WeatherSlot1

Fixed weather, slot 1. The enum values are available also via HTTP API at </api/list/enums/weather> or in Lua global tables `lists.enums.weather`, `Weather` and `value_to_weather`. (ReadWrite)

int32 WeatherSlot2

Fixed weather, slot 2. The enum values are available also via HTTP API at </api/list/enums/weather> or in Lua global tables `lists.enums.weather`, `Weather` and `value_to_weather`. (ReadWrite)

int32 WeatherSlot3

Fixed weather, slot 3. The enum values are available also via HTTP API at </api/list/enums/weather> or in Lua global tables `lists.enums.weather`, `Weather` and `value_to_weather`. (ReadWrite)

int32 WeatherSlot4

Fixed weather, slot 4. See The enum values are available also via HTTP API at </api/list/enums/weather> or in Lua global tables `lists.enums.weather`, `Weather` and `value_to_weather`. (ReadWrite)

int32 GameMode

Game mode hash. Multiplayer sessions always use -1958878043 or 'MP_RaceWeekend'.

int32 Latitude

Track's latitude, in degrees multiplied by 1000 (so 180000 corresponds to 180 degrees).

int32 Longitude

Track's longitude, in degrees multiplied by 1000.

int32 Altitude

Track's base altitude, in millimeters.

string SessionState

Multiplayer session state. Valid values are None, Lobby, Loading, Race, PostRace, Returning.

string SessionStage

Multiplayer session flow/stage. Valid values are Practice1, Practice2, Qualifying, Warmup, FormationLap, Race1. Race2 might be used if two-race format is implemented in multiplayer.

string SessionPhase

Race manager phase. This is relatively internal value used during race startup synchronization. Valid values are PreCountDownSync, PreRace, CountdownWait, Countdown, Green, Invalid.

int32 SessionTimeElapsed

Time elapsed since the start of the session (this stage), in seconds. Note that this value might currently start counting during loading and other transitions, and then reset back to zero when the race really starts.

int32 SessionTimeDuration

Time elapsed since the start of the session, in seconds.

int32 NumParticipantsValid

Number of 'valid' race participants, or positions.

int32 NumParticipantsDisqualified

Number of disqualified participants.

int32 NumParticipantsRetired

Number of retired participants.

int32 NumParticipantsDNF

Number of DNF participants.

int32 NumParticipantsFinished

Number of finished participants.

int16 CurrentYear

Current race date - year.

int8 CurrentMonth

Current race date - month.

int8 CurrentDay

Current race date - day of month.

int8 CurrentHour

Current race date - hour of day.

int8 CurrentMinute

Current race date - minutes.

int16 RainDensity

Rain density, this implies how "visually" rainy the weather is. Values between 0 and 1000 are valid.

int16 WetnessOnPath

Average wetness on path. Values between 0 and 1000 are valid.

int16 WetnessOffPath

Average wetness off path. Values between 0 and 1000 are valid.

int16 WetnessAverage

Overall average wetness on track. Values between 0 and 1000 are valid.

int16 WetnessPredictedMax

Predicted maximum wetness on track, if available.

int16 WetnessMaxLevel

Maximum water level, this implies how "physically" rainy the weather is. Values between 0 and 1000 are valid.

int32 TemperatureAmbient

Ambient temperature, in deg Celsius multiplied by 1000 (so 37000 is 37 degrees Celsius).

int32 TemperatureTrack

Track temperature, in deg Celsius multiplied by 1000.

int32 AirPressure

Air pressure.

Member Attributes

Setup and status of each session member (player).

int32 VehicleId

Player's vehicle id. Latest vehicle list is available also via HTTP API at `/api/list/vehicles` and in Lua global tables `lists.vehicles`, `id_to_vehicle` and `name_to_vehicle`.

int32 LiveryId

Livery id used on the player's vehicle. Latest vehicle+liveries list is available via HTTP API at `/api/list/liveries` and in Lua global tables `lists.vehicles`, `id_to_vehicle` and `name_to_vehicle`. It is not included in the Values and Types documentation.

string LoadState

Player's loading state. This is an internal variable with many possible states. There are two sets of states, one for the admin (host) player and another for clients, the states range from ready-for-sync to syncing to loading to ready.

int32 RaceStatFlags

Player's race stat flags. The flag values are available also via HTTP API at `/api/list/flags/player` or in Lua global tables `lists.flags.player`, `PlayerFlags` and `value_to_player_flag`.

int16 Ping

Player's ping to the dedicated server. This is the round trip time, so "there and back".

Participant Attributes

Status of each participant (vehicle).

int16 RefId

RefId of the session member owning this participant.

string Name

Participant's name. Same as the name of the owning player if this is a player participant, otherwise a generated AI name.

int8 IsPlayer

1 if this participant is controlled by the owning player, 0 if this is an AI participant.

int8 GridPosition

Starting grid position of the participant.

int32 VehicleId

Participant's vehicle id. Latest vehicle list is available also via HTTP API at `/api/list/vehicles` and in Lua global tables `lists.vehicles`, `id_to_vehicle` and `name_to_vehicle`.

int32 LiveryId

Livery id used on the participant's vehicle. Latest vehicle+liveries list is available via HTTP API at `/api/list/liveries` and in Lua global tables `lists.vehicles`, `id_to_vehicle` and `name_to_vehicle`. It is not included in the Values and Types documentation.

int8 RacePosition

Current race position of the participant.

int16 CurrentLap

Current lap. Zero if not in a lap yet.

int8 CurrentSector

Current sector's index (1 to 3 for valid circuit sectors). Sector time corresponding to this sector is the current sector time. Other sector times are the times last set in those sectors, if available.

int32 Sector1Time

Last time set in sector 1, or current sector 1 time if that sector is the CurrentSector, in milliseconds.

int32 Sector2Time

Last time set in sector 2, or current sector 2 time if that sector is the CurrentSector, in milliseconds.

int32 Sector3Time

Last time set in sector 3, or current sector 3 time if that sector is the CurrentSector, in milliseconds.

int32 LastLapTime

Last lap time, in milliseconds.

int32 FastestLapTime

Fastest lap time, in milliseconds.

string State

Racing state of the participant. Valid values are Racing, Finished, DNF, Disqualified, Retired, InGarage, EnteringPits, InPits, ExitingPits.

int8 HeadlightsOn

Flag set to non-zero if the participant's headlights are turned on.

int8 WipersOn

Flag set to non-zero if the participant's wipers are turned on.

int16 Speed

The participant's vehicle speed, in kph.

int8 Gear

Current gear.

int16 RPM

Current RPM.

int32 PositionX

Position on map, X coordinate, in millimeters.

int32 PositionY

Position on map, Y coordinate, in millimeters.

int32 PositionZ

Position on map, Z coordinate, in millimeters.

int16 Orientation

Vehicle orientation (heading) on the map, in degrees from 0 to 360.

Events

Events are generated by the server for various important... events. An event can be associated globally with the session, or with its member or participant. Each event has a name and includes attributes specific to that event.

Session Events

These are global events associated with the session itself.

SessionSetup

This event is generated when the game starts loading. It includes the basic information about the session's setup. More can be queried from the detailed setup attributes, they will no longer change after this event happens.

int8 GridSize

Total maximum grid size, including AI participants. Can be modified only for the 'next session' and can never be larger than the maximum session size set in the server config.

int8 MaxPlayers

Maximum number of players. Can't be higher than GridSize, but can be lower and then the extra slots will be reserved for AI vehicles. Can be modified only for the 'next session'.

int16 Practice1Length

Length of first practice session in minutes.

int16 Practice2Length

Length of second practice session in minutes.

int16 QualifyLength

Length of qualification session in minutes.

int16 WarmupLength

Length of warmup session in minutes.

int16 Race1Length

Race 1 length in lap count.

int16 Race2Length

Race 2 length in lap count. Currently not used in multiplayer.

int32 Flags

Session setup flags. The flag values are available also via HTTP API at `/api/list/flags/session` or in Lua global tables `lists.flags.session`, `SessionFlags` and `value_to_session_flag`.

int32 TrackId

Track id. Players will be able to change this unless `ServerControlsTrack` is set to non-zero. The list of all valid tracks is available also via HTTP API at `/api/list/tracks` or in Lua global tables `lists.tracks`, `id_to_track` and `name_to_track`.

int32 GameMode

Game mode hash. Multiplayer sessions always use `-1958878043` or `'MP_RaceWeekend'`.

StateChanged

This event is generated whenever the multiplayer session's state changes.

string PreviousState

Previous session state.

string NewState

New session state. Valid values are `None`, `Lobby`, `Loading`, `Race`, `PostRace`, `Returning`.

StageChanged

This event is generated whenever the multiplayer session's stage (phase) changes.

string PreviousStage

Previous session stage.

string NewStage

New session stage. Valid values are `Practice1`, `Practice2`, `Qualifying`, `Warmup`, `FormationLap`, `Race1`, `Race2`.

int16 Length

Length of the new stage. In minutes or laps, depending on the stage. Matches value sent in the setup event.

SessionCreated

This event is generated when someone joins an empty server and a new multiplayer session is created.

SessionDestroyed

This event is generated when all players leave the server and the multiplayer session is destroyed.

ServerChat

This event contains a chat message sent by the server.

int16 RefId

RefId of the recipient of the message, or zero if this message is sent to all players.

string Message

The message.

Member Events

Events related to session members (players).

PlayerJoined

This event is generated when a new player joins the session. The player is not authenticated with Steam at this point and will not be reported to the game yet.

string Name

Player's name.

string SteamId

Player's Steam id.

Authenticated

This event is generated when Steam authenticates a player.

PlayerLeft

This event is generated when a player leaves the session.

string Reason

Reason for the leave.

int8 GameReasonId

If leaving via game call (on their own or kicked), the reason id used by the game. Otherwise this attribute won't be set at all. Usual game-specific values are: 1 when the user leaves via the UI, 2 when kicked, 3 or 5 when disconnected. But the value can't be always fully trusted.

PlayerChat

Event with a chat message sent by a player to everyone else in the session.

string Message

The message.

Participant Events

Events related to session participants (vehicles).

ParticipantCreated

This event is generated when a new participant is created.

string Name

Participant's name. Same as the player owning the participant if this is a player's vehicle, otherwise a generated AI name.

int8 IsPlayer

1 if this participant is controlled by the owning player, 0 if this is an AI participant.

int32 VehicleId

Participant's vehicle id. Latest vehicle list is available also via HTTP API at `/api/list/vehicles` and in Lua global tables `lists.vehicles`, `id_to_vehicle` and `name_to_vehicle`.

int32 LiveryId

Livery id used on the participant's vehicle. Latest vehicle+liveries list is available via HTTP API at `/api/list/liveries` and in Lua global tables `lists.vehicles`, `id_to_vehicle` and `name_to_vehicle`. It is not included in the Values and Types documentation.

ParticipantDestroyed

This event is generated when an existing participant is destroyed.

Sector

This event is generated when a participant crosses a sector boundary.

int16 Lap

Lap number. When crossing the finish sector this is the finished lap's number, the lap counter is advanced after this event happens.

int8 Sector

Index of the sector boundary crossed by the participant. 0=start, 1,2=sectors, 3=finish.

int32 SectorTime

Final time set in the finished sector, in milliseconds.

int32 TotalTime

Total race time, in milliseconds.

int8 CountThisLap

One if this lap counts, zero if the lap is not considered valid.

int8 CountThisLapTimes

One if this lap's time counts, zero if the time has been invalidated (usually by going off-track).

Lap

This event is generated whenever a participant finishes a lap.

int16 Lap

Lap number of the finished lap.

int32 LapTime

Lap time, in milliseconds.

int32 Sector1Time

Sector 1 time, in milliseconds.

int32 Sector2Time

Sector 2 time, in milliseconds.

int32 Sector3Time

Sector 3 time, in milliseconds.

int8 RacePosition

Race position.

int32 DistanceTravelled

Distance travelled in this lap, in meters.

int8 CountThisLapTimes

One if this lap's time counts, zero if the time has been invalidated (usually by going off-track).

State

This event is generated when a participant's state changes.

string PreviousState

Previous state.

string NewState

New state. Valid values are Racing, Finished, DNF, Disqualified, Retired, InGarage, EnteringPits, InPits, ExitingPits

Results

This event is generated just before the current stage finishes, and contains the results for this participant.

int8 RacePosition

Participant's final race position. Relevant only in qualification and race stages.

int16 Lap

Current lap number.

int32 VehicleId

Id of vehicle used.

string State

Participant's state. Valid values are Racing, Finished, DNF, Disqualified, Retired, InGarage, EnteringPits, InPits, ExitingPits.

int32 TotalTime

Participant's total race time.

int32 FastestLapTime

Participant's fastest lap time.

Impact

This event is generated when a participant collides with something. This event is reported by one of the participants in the collision, and can contain information about the other participant (unless this is a collision with the environment). Which participant reports this event is not defined, it is not the one who "caused" the collision. Because of the assymetric nature of physical collisions in multiplayer, the other participant might report the same event too, with slightly different values.

int32 OtherParticipantId

Other participant's id, or -1 if not collided into a participant.

int32 CollisionMagnitude

Magnitude of the impact. Value reported by physics multiplied by 1000, with 0 meaning no impact (never reported) and 1000 being the highest value.

CutTrackStart

This event is reported when a participant goes off-track.

int16 Lap

Current lap.

int8 RacePosition

Current race position.

int8 IsMainBranch

Is this the main branch of the track?

int32 LapTime

Current time into lap, in milliseconds.

CutTrackEnd

This event is reported when a participant returns back to track. This matches the previous CutTrackStart event reported for the participant.

int32 ElapsedTime

Time elapsed while off-track.

int32 SkippedTime

Estimated time skipped by cutting the track.

int8 PlaceGain

Number of places gained or lost while off-track.

int32 PenaltyValue

Penalty value computed from various cut-track variables.

int32 PenaltyThreshold

Minimum penalty value for the cut to be considered for actual penalty.

Track List

All tracks supported by the game. The track identifiers are used for example by the TrackId session attribute.

name	id	gridsize
Autodromo Nazionale Monza GP	-52972612	54
Autodromo Nazionale Monza Short	368740158	40
Azure Circuit	832629329	36
Azure Coast	560711985	32
Azure Coast Stage 1	550129415	32
Azure Coast Stage 2	-780879576	32
Azure Coast Stage 3	-1737261125	36
Azure Coast Westbound	-1936790504	25
Bannochbrae Road Circuit	-602684269	20
Bathurst	921120824	36
Brands Hatch GP	1988984740	30
Brands Hatch Indy	1300627020	30
Brno	-907901266	40
Cadwell Club Circuit	328972919	20
Cadwell GP	1876749797	12
Cadwell Woodland	-1408189041	12
California Highway Full	-1593944167	36
California Highway Reverse	928006536	32
California Highway Stage 1	1676943041	40
California Highway Stage 2	940391868	36
California Highway Stage 3	-331502851	36
Chesterfield	-1735912413	24
Circuit de Barcelona-Catalunya Club	-1042928898	40

Circuit de Barcelona-Catalunya GP	521933422	36
Circuit de Barcelona-Catalunya National	-998191994	36
Circuit de Spa-Francorchamps	904625875	36
Circuit des 24 Heures du Mans	1740968730	56
Donington Park GP	354022214	38
Donington Park National	-1194019375	38
Dubai Autodrome Club	1735854797	36
Dubai Autodrome GP	-661887517	36
Dubai Autodrome International	-710712693	36
Dubai Autodrome National	-31727447	36
Dubai Kartdrome	-232513374	22
Glencairn	-1066742780	16
Glencairn East	766599953	16
Glencairn East Reverse	-446794969	16
Glencairn Reverse	-1520844580	16
Glencairn West	-1408779593	16
Glencairn West Reverse	-913625358	16
Greenwood Karting Circuit	-1160443077	22
Hockenheim Classic	1552853772	36
Hockenheim GP	1695182971	30
Hockenheim National	-1977142985	30
Hockenheim Short	1768660198	30
Imola	920145926	40
Le Circuit Bugatti	-1027934689	40
Mazda Raceway Laguna Seca	-1612023328	26

Mojave Boa Ascent	850003838	47
Mojave Cougar Ridge	-688586697	47
Mojave Coyote Noose	-2125682335	47
Mojave Gila Crest	2089801285	47
Mojave Sidewinder	-1463443929	47
Nordschleife	697498609	32
Nordschleife Stage 1	1459212514	36
Nordschleife Stage 2	-300387291	36
Nordschleife Stage 3	1128950148	36
Nürburgring GP	-945967394	30
Nürburgring Müllenbach	-246966400	36
Nürburgring Sprint	-709737101	30
Nürburgring Sprint Short	-810715843	30
Oschersleben C Circuit	-1359299594	23
Oschersleben GP	-1194185720	30
Oschersleben National	816601966	30
Oulton Park Fosters	-2021024495	26
Oulton Park International	545979690	26
Oulton Park Island	-1877699523	26
Road America	-660300766	64
Rouen Les Essarts	-1031249929	20
Rouen Les Essarts Short	-1515473908	20
Ruapuna Park A Circuit	619694160	28
Ruapuna Park B Circuit	-2046633090	28
Ruapuna Park Club	1446378877	28
Ruapuna Park GP	1277693448	28
Ruapuna Park Outer Loop	1940584155	28
Sakitto GP	-1759743046	54
Sakitto International	-1474170192	26
Sakitto National	-1260826266	26

Sakitto Sprint	-879282119	54
Silverstone Classic	-1194290828	18
Silverstone GP	1641471184	32
Silverstone International	1101719627	32
Silverstone National	1952936927	32
Silverstone Stowe	1600840139	16
Snetterton 100	-867340010	26
Snetterton 200	1058872832	26
Snetterton 300	1508903068	26
Sonoma Raceway GP	-1454279631	43
Sonoma Raceway National	-995202729	33
Sonoma Raceway Short	1035110721	43
Summerton	-44748320	28
Summerton National	1408845203	28
Summerton Sprint	-1605913568	28
Watkins Glen GP	-1785781495	43
Watkins Glen Short	1590386668	43
Willow Springs Horse Thief Mile	-1849531562	24
Willow Springs International Raceway	-103312908	50
Zhuhai International Circuit	1836888499	42
Zolder	-360711057	36

Vehicle List

All vehicles supported by the game. The vehicle identifiers are used for example by the VehicleModelId session attribute. This document does not include ids of all individual vehicle liveries, please run /api/list/liveries to get those.

name	id	class
125cc Shifter Kart	844159614	Kart1
250cc Superkart	-937933849	Kart3
Alpine A450	2022787216	LMP2
Ariel Atom 300 Supercharged	-1770401008	Road B
Ariel Atom 500 V8	607772150	Road A
Ariel Atom Mugen	526708607	Road B
Aston Martin DBR1-2	738755746	LMP1
Aston Martin DBR1/300	-91815086	Vintage GT2
Aston Martin Rapide S Hydrogen Hybrid	-324240456	GT4
Aston Martin V12 Vantage GT3	1452261378	GT3
Aston Martin V8 Vantage GT4	2086246081	GT4
Aston Martin V8 Vantage GTE	1401532035	GT3
Aston Martin Vantage GT12	1268015922	Road B
Audi 90 quattro IMSA GTO	1470929381	Trans-Am
Audi A1 quattro	2082176226	Road C2
Audi R18 TDI	-1067902110	LMP1
Audi R18 e-tron quattro	1219511257	LMP1
Audi R8 (LMP900)	-1761671051	LMP900
Audi R8 LMS Ultra	1884411907	GT3
Audi R8 V10 plus	1469658023	Road B
BAC Mono	1400443574	Road B

BMW 1-Series M Coupe	-1226176940	Road C1
BMW 1-Series M Coupe StanceWorks Edition	-1411323812	Road B
BMW 2002 StanceWorks Edition	-1187748261	Historic Touring 2
BMW 2002 Turbo	143364290	Historic Touring 2
BMW 320 TC	9503224	TC2
BMW 320 Turbo Group 5	779111340	Group 5
BMW M1 Procar	1368036017	Group 4
BMW M3 E30 Group A	-934098507	Group A
BMW M3 GT	-1365918055	GT3
BMW M3 GT4	-1545450182	GT4
BMW V12 LMR	975104023	LMP900
BMW Z4 GT3	1161219858	GT3
Bentley Continental GT3	987814806	GT3
Bentley Speed 8	-494100071	LMP900
Cadillac ATS-V.R GT3	-2025231366	GT3
Caper Monterey Stockcar	578910088	Vintage Stockcar
Caterham SP/300.R	675194619	LMP3
Caterham Seven Classic	-886212684	Road D
Caterham Superlight R500	-162683612	Road B
Chevrolet Corvette C7.R	-384044277	GT3
Dallara DW12 Indycar (Road Course)	1901402958	Indycar
Ford Escort Mk1 RS1600	1639105598	Historic Touring 2
Ford Falcon FG V8 Supercar	1357515789	V8 Supercars
Ford Focus RS	-1738120892	Road C2
Ford Fusion Stockcar	851491257	Modern Stockcar
Ford MkIV	-1774335742	Vintage GT
Ford Mustang 2+2 Fastback	1397255601	Historic Touring 2
Ford Mustang Boss 302R1	1111049682	GT4

Ford Mustang Cobra TransAm	-11335215	Trans-Am
Ford Mustang GT	1230061845	Road B
Ford Sierra RS500 Cosworth Group A	-1253474718	Group A
Ford Zakspeed Capri Group 5	1817703058	Group 5
Formula A	1909945073	FA
Formula B	-1318848040	FB
Formula C	-1041674971	FC
Formula Gulf 1000	-1706259671	F4
Formula Renault 3.5	1626504761	Formula Renault
Formula Rookie	-2075284877	F5
GUMPERT apollo S	-674255528	Road A
Ginetta G40 Junior	310900789	GT5
Ginetta G55 GT3	-1170674276	GT3
Ginetta G55 GT4	2091910841	GT4
KTM X-Bow R	761457895	Road B
Lotus Type 25 Climax	-713284494	Vintage F1 A1
Lotus Type 38 Ford	1162971218	Vintage Indy 65
Lotus Type 40 Ford	-1204688299	Vintage GT
Lotus Type 49 Cosworth	578969971	Vintage F1 A
Lotus Type 49C Cosworth	1061494025	Vintage F1 A
Lotus Type 51	-1435057179	Vintage F3 A
Lotus Type 72D Cosworth	-1320616846	Vintage F1 B
Lotus Type 78 Cosworth	-1835861548	Vintage F1 B
Lotus Type 98T Renault Turbo	1959097924	Vintage F1 C
Marek RP 219D LMP2	1355771595	LMP2
Marek RP 339H LMP1	-2030487367	LMP1
Mazda MX-5 Radbul	-1966060946	Road A

McLaren 12C	-444124510	Road A
McLaren 12C GT3	-1166911988	GT3
McLaren F1	307010432	Road B
McLaren F1 GTR Long Tail	-1001569309	GT1
McLaren P1™	-1748676965	Road A
Mercedes-Benz 190E 2.5-16 Evolution2 DTM	262982797	Group A
Mercedes-Benz 300SEL 6.8 AMG	-85660500	Historic Touring 2
Mercedes-Benz 300SL (W194)	1401308680	Vintage GT3
Mercedes-Benz A45 AMG	-1522922538	Road C1
Mercedes-Benz AMG C-Coupe DTM	-525060572	TC3
Mercedes-Benz CLK-LM	1979398129	GT1
Mercedes-Benz SLS AMG	-1729266457	Road B
Mercedes-Benz SLS AMG GT3	274862187	GT3
Mitsubishi Lancer Evolution IX FQ360	-149617068	Road C1
Mitsubishi Lancer Evolution VI TME	460478144	Road C2
Mitsubishi Lancer Evolution X FQ400	998947753	Road C1
Oreca 03 Nissan	-98064499	LMP2
Pagani Huayra	-2062679088	Road A
Pagani Zonda Cinque Roadster	-1617916111	Road A
Pagani Zonda R	-2066888471	GT1X
Palmer Jaguar JP-LM	-1921505310	LMP3
RWD P20 LMP2	146414985	LMP2
RWD P30 LMP1	1023089804	LMP1
Radical RXC Turbo	-1048050877	Road A

Radical SR3-RS	1231996358	LMP3
Radical SR8-RX	152867459	LMP3
Renault Alpine A442B	-699643670	Group 6
Renault Clio Cup	-648709823	TC1
Renault Megane R.S. 265	1694837381	Road C2
Renault Megane R.S. 275 Trophy-R	-956881226	Road C1
Renault Megane Trophy V6	-931590477	Megane Trophy
Renault Sport R.S. 01	-1856998124	RS01 Trophy
Ruf CTR	2017392050	Road B
Ruf CTR3	558051123	Road A
Ruf CTR3 SMS-R	-2046825312	GT1X
Ruf RGT-8	462547146	Road B
Ruf RGT-8 GT3	-2020758805	GT3
Ruf Rt 12 R	704998514	Road A
Sauber C9 Mercedes-Benz	65306143	Group C1
Scion FR-S	-356101373	Road D
Scion FR-S Rocket Bunny Edition	-1127314200	Road B
Toyota 86	-41807622	Road D
Toyota GT-86	-235751604	Road D
Toyota GT-86 Rocket Bunny GT Edition	1764851930	GT4
Toyota TS040 Hybrid	1810453820	LMP1
W Motors Lykan HyperSport	-180129877	Road A

Vehicle Classes

All vehicle classes supported by the game. The class identifiers are used for example by the VehicleClassId session attribute.

name	value
F4	529493584
F5	1361234018
FA	2111151040
FB	1189178377
FC	1646832695
Formula Renault	-1981841276
GT1	1323122160
GT1X	-1206906486
GT3	-112887377
GT4	1553262379
GT5	-1358908608
Group 4	-718294614
Group 5	-48135339
Group 6	-427741350
Group A	-1270088329
Group C1	-1503297100
Historic Touring 2	-1176205746
Indycar	61661019
Kart1	2028372520
Kart3	-1562493614
LMP1	-1289517523
LMP2	-564539194
LMP3	974854672
LMP900	1543160927
Megane Trophy	-824554870

Modern Stockcar	1992634946
RS01 Trophy	-4596059
Road A	767810805
Road B	-1300724070
Road C1	-101706458
Road C2	-1811847806
Road D	298754909
TC1	-1529501352
TC2	-1199540832
TC3	-1251471527
Trans-Am	99101703
V8 Supercars	1572999773
Vintage F1 A	-541941989
Vintage F1 A1	-434935163
Vintage F1 B	1153998110
Vintage F1 C	-1741070546
Vintage F3 A	-322104719
Vintage GT	-105273098
Vintage GT2	-453292945
Vintage GT3	-985885375
Vintage Indy 65	1554536027
Vintage Stockcar	-880731824

Enums

The attributes use several "enum" types listed below. In server.cfg configuration, only the numeric values can be used. From Lua addons, and therefore also in the sms_rotate addon's configuration, either the numeric value or the name string can be used.

Damage

The Damage enum type, used by the DamageType session attribute.

name	value
OFF	0
VISUAL_ONLY	1
PERFORMANCEIMPACTING	2
FULL	3

TireWear

The TireWear enum type, used by the TireWearType session attribute.

name	value
X7	0
X6	1
X5	2
X4	3
X3	4
X2	5
STANDARD	6
SLOW	7
OFF	8

FuelUsage

The FuelUsage enum type, used by the FuelUsageType session attribute.

name	value
STANDARD	0
SLOW	1
OFF	2

Penalties

The Penalties enum type, used by the PenaltiesType session attribute.

name	value
NONE	0
FULL	1

AllowedView

The AllowedView enum type, used by the AllowedViews session attribute. Note that the game supports only the "Any" and "CockpitHelmet" values.

name	value
Any	0
CockpitHelmet	2

Weather

The Weather enum type, used by the WeatherSlotN session attributes. To use "real" weather, set WeatherSlots to 0.

name	value
Clear	-934211870
FogWithRain	-358600329
Foggy	2067843977
Hazy	-1299791789
HeavyCloud	129238383
HeavyFog	-754279862
HeavyFogWithRain	-1604560069

LightCloud	296956818
LightRain	270338437
MediumCloud	888299130
Overcast	-1293634875
Rain	1461703858
Random	1275961519
Storm	-1592958063
ThunderStorm	-2112363295

Flags

In addition to "enums", several attributes have a "flags" type. While enums can be set to one selected value from their list, flags can combine any number of the values together by adding the values. The attributes use several "enum" types listed below. In server.cfg configuration, only the added values can be used. From Lua addons, and therefore also in the sms_rotate addon's configuration, either the numeric added value or a string format can be used. The string is created by separating several flag names (or event their numeric values) by commas, with no whitespace inbetween. So for example "ABS_ALLOWED,SC_ALLOWED" is the same session flags value as 96, which is the same as "SC_ALLOWED,32".

SessionFlags

The SessionFlags flags type, used by the Flags session attributes.

name	value
FORCE_IDENTICAL_VEHICLES	2
ALLOW_CUSTOM_VEHICLE_SETUP	8
FORCE_REALISTIC_DRIVING_AIDS	16
ABS_ALLOWED	32
SC_ALLOWED	64
TCS_ALLOWED	128
FORCE_MANUAL	256
ROLLING_STARTS	512
FORCE_SAME_VEHICLE_CLASS	1024
FILL_SESSION_WITH_AI	131072
MECHANICAL_FAILURES	262144
AUTO_START_ENGINE	524288
TIMED_RACE	1048576
GHOST_GRIEFERS	2097152
ENFORCED_PITSTOP	4194304

PlayerFlags

The PlayerFlags flags type, used by the RaceStatFlags player (session member) attributes. This

is a status attribute which can't be modified and is not used in session setups.

name	value
MODEL_NOVICE	0
SETUP_USED	1
CONTROLLER_GAMEPAD	2
CONTROLLER_WHEEL	4
CONTROLLER_MASK	6
AID_STEERING	8
AID_BRAKING	16
AID_ABS	32
AID_TRACTION	64
AID_STABILITY	128
AID_NO_DAMAGE	256
AID_AUTO_GEARs	512
AID_AUTO_CLUTCH	1024
MODEL_NORMAL	2048
MODEL_EXPERIENCED	4096
MODEL_PRO	6144
MODEL_ELITE	8192
MODEL_MASK	14336
AID_DRIVING_LINE	32768
VALID	1073741824