

Контракты статистик данных. СПИК

- 1 Контракт данных SpicAddonStatisticsResult
- 2 Контракт данных SpicConcreteStatisticsResult
- 3 Контракт данных SpicStatisticsChunkInfo
- 4 Контракт данных SpicStatisticsSession
- 5 Контракт данных SpicStatisticsSessionRequest
- 6 Контракт данных SpicStatisticsSessionResponse
- 7 Контракт данных SpicStatisticsOperationResult
- 8 Контракт данных SpicDiscreteSensorsStatistics
- 9 Контракт данных SpicDiscreteSensorsStatisticsResult
- 10 Контракт данных SpicDiscreteSensorStatistics
- 11 Контракт данных SpicDiscreteSensorValuePoint
- 12 Контракт данных SpicAnalogSensorValuePoint
- 13 Контракт данных SpicFuelFlowStatistics
- 14 Контракт данных SpicFuelFlowStatisticsResult
- 15 Контракт данных SpicFuelFlowStatisticsSensor
- 16 Контракт данных SpicFdStatisticsFuelEventType
- 17 Контракт данных SpicFuelingDefuelingStatistics
- 18 Контракт данных SpicFuelingDefuelingStatisticsEvent
- 19 Контракт данных SpicFuelingDefuelingStatisticsResult
- 20 Контракт данных SpicMotorModesStatistics
- 21 Контракт данных SpicMotorModesStatisticsResult
- 22 Контракт данных SpicMotorPeriod
- 23 Контракт данных SpicMotorPeriodType
- 24 Контракт данных SpicNavigationValidationStatistics
- 25 Контракт данных SpicNavigationValidationStatisticsResult
- 26 Контракт данных NavigationFiltrationStatisticsRequest
- 27 Контракт данных SpicLocation
- 28 Контракт данных SpicNavigation
- 29 Контракт данных SpicNavigationFiltrationStatistics
- 30 Контракт данных SpicNavigationFiltrationStatisticsResult
- 31 Контракт данных SpicNavigationFiltrationStatisticsSettings
- 32 Контракт данных SpicNavigationPoint
- 33 Контракт данных SpicNavigationValidationFilter
- 34 Контракт данных SpicTrackPeriodsFilter
- 35 Контракт данных SpicTrackPeriod
- 36 Контракт данных SpicTrackPeriodsStatistics
- 37 Контракт данных SpicTrackPeriodStatisticsResult

- 38 Контракт данных SpicTrackPeriodType
- 39 Контракт данных SpicTrackRecoil
- 40 Контракт данных SpicTrackPeriodMileage
- 41 Контракт данных SpicTrackPeriodsMileageStatistics
- 42 Контракт данных SpicTrackPeriodsMileageStatisticsResult
- 43 Контракт данных SpicOdometerStatisticsResult

Контракт данных SpicAddonStatisticsResult

```
// ?????????? ??????????? ??????????? ? ?????? ??????????? ???????????.
SpicAddonStatisticsResult
{
    // ?????? ??????????.
    SpicStatisticsOperationResult Status

    // ?????????? ?? ??????.
    string ErrorText
}
```

Контракт данных SpicConcreteStatisticsResult

```
// ?????????? ??????????? ??????????? ????????????,
// ??????? ?????? ??????????? ?? ?????????? ??????????? ???????????.
// ?????????? ?????? ??????????.
SpicConcreteStatisticsResult
{
    // ??????????? ? ???? ??????????? ??????????? ???????????.
    SpicStatisticsChunkInfo ChunkInfo

    // ??????????? ???????????. ???? ?? ??????????? ?????????? - null.
    IStatistics Statistics
}
```

Контракт данных SpicStatisticsChunkInfo

```
// ??????????? ? ??????????? ??????????? ???????????.
SpicStatisticsChunkInfo
{
    // ?????? ?????? ?????? ???????????.
    SpicDateTimeRange Period

    // ??????????? ??????.
    int ChunkNumber

    // ?????????? ?? ?????? ?????? ??????????? ???????????.
    bool IsFinalChunk

    // ??????????? ??????.
    SpicStatisticsOperationResult Status

    // ?????????? ??????.
    string ErrorText
}
```

Контракт данных SpicStatisticsSession

```
// ?????????????? ?????? ?????????????? ?????????????? ? ?????.
SpicStatisticsSession
{
    // ?????? ???????.
    Guid StatisticsSessionId
}
```

Контракт данных SpicStatisticsSessionRequest

```
// ??????? ?? ?????????? ??????? ??? ?????????????? ??????????????.
SpicStatisticsSessionRequest
{
    // ??????? ?????????????? ??????????????.
    SpicDateTimeRange Period

    // ???????, ?? ?????????? ?????? ?????????? ??????????????.
    SpicObjectIdentity TargetObject
}
```

Контракт данных SpicStatisticsSessionResponse

```
// ??????????? ??????????? ??????? ??? ?????????????? ??????????????.
SpicStatisticsSessionResponse
{
    // ???????.
    SpicStatisticsSession Session

    // ??????????? ???????.
    SpicStatisticsOperationResult OperationResult
}
```

?????????? ??????? SpicStatisticsOperationResult

```
// Состояние выполнения операции при построении статистик
SpicStatisticsOperationResult
{
    string Value
}
```

?????????? **Value** ?????? ?????????????? ?????????????? ??????????????:

- **Ok** - ???????
- **Processing** – ??? ???????????
- **Error** – ???????
- **None** – ??????????????.

Контракт данных SpicDiscreteSensorsStatistics

```
// ?????? ?????????? ?? ?????????? ??????????  
SpicDiscreteSensorsStatistics  
{  
    // ??????????  
    SpicDiscreteSensorStatistics[] Sensors  
}
```

Контракт данных SpicDiscreteSensorsStatisticsResult

```
// ?????????? ?????????? ??????????  
SpicDiscreteSensorsStatisticsResult  
{  
    // ?????????? ? ?????????? ?????????? ?????? ??????????  
    SpicStatisticsChunkInfo ChunkInfo  
  
    // ??????????. ??? ? ?????????? ?????????? - null.  
    SpicDiscreteSensorsStatistics Statistics  
}
```

Контракт данных SpicDiscreteSensorStatistics

```
// ?????????? ?? ?????? ??????????. ?????? Points ?????????????? ??????????????  
????????????? ?????? ?? ?????????? ??????????. ?????????, ??? ? ?????????? ?????????? ??????  
????????????? ?????????? ?????? ?????????????, ? ? ?????? ? ???? ? ????  
????????????????????? ?????? ?????? Points.  
SpicDiscreteSensorStatistics  
{  
    // ?????? ??????  
    SpicDiscreteSensorValuePoint[] Points  
}
```

Контракт данных SpicDiscreteSensorValuePoint

```
// ?????? ?????????????? ?? ?????????????? ??????????  
SpicDiscreteSensorValuePoint  
{  
    // ?????? ?????????? ??????????  
    DateTime Timestamp  
  
    // ??????????  
    bool Value  
}
```

Контракт данных SpicAnalogSensorValuePoint

```
// ?????? ?????????????? ?? ?????????????? ??????????  
SpicAnalogSensorValuePoint  
{  
    // ?????? ?????????? ??????????  
    DateTime Timestamp  
  
    // ??????????  
    double Value  
}
```

Контракт данных SpicFuelFlowStatistics

```
// ?????????? ?? ?????? ???????
SpicFuelFlowStatistics
{
    // ????? ?????????? ?? ????????? ??????? ???????
    SpicFuelFlowStatisticsSensor[] Sensors

    // ?????????? ???????????
    SpicFuelFlowStatisticsSensor Summary
}
```

Контракт данных SpicFuelFlowStatisticsResult

```
// ?????????? ??????? ??????????
SpicFuelFlowStatisticsResult
{
    // ?????????? ? ?????????? ?????????? ?????? ??????????
    SpicStatisticsChunkInfo ChunkInfo

    // ???????????. ??? ? ?????????? ?????????? - null.
    SpicFuelFlowStatistics Statistics
}
```

Контракт данных SpicFuelFlowStatisticsSensor

```
// ?????????? ?? ??????? ????????? ?????????
SpicFuelFlowStatisticsSensor
{
    // ?????? ?????????
    int SensorNumber

    // ??? ?????????
    string SensorTitle

    // ?????????????? ?? ?????? ??????????, ? ??????
    double? MovementConsumptionL

    // ?????????????? ?????????? ?? ?????? ?????????? ??????? ?????????????, ? ??????
    double? EngineActiveWorkConsumptionL

    // ?????????????? ?????????? ?? ?????? ?????????? ??????? ?????????????, ? ??????
    double? EngineIdleConsumptionL

    // ?????? ?????????????? ?????????, ? ??????
    double? TotalConsumptionVolumeL

    // ?????? ?????????????? ?????????, ? ??????????????
    double? TotalConsumptionWeightKg

    // ?????? ??????????????
    SpicAnalogSensorValuePoint[] Points
}
```

Контракт данных SpicFdStatisticsFuelEventType

```
// ??? ?????????? ??????????/???????? ?????????
```

```
SpicFdStatisticsFuelEventType
{
    string Value
}
```

Параметр **Value** может принимать следующие значения:

- **Fueling** - Заправка
- **Defueling** - Слив
- **None** – НЕИЗВЕСТНО.

Контракт данных SpicFuelingDefuelingStatistics

```
SpicFuelingDefuelingStatistics
{
    // ?????????? ?????? ? ???????.
    double? BeginFuelVolumeL

    // ?????????? ?????? ? ???????.
    double? EndFuelVolumeL

    // ?????????????? ?????? ? ???????.
    double? MinFuelVolumeL

    // ?????????????????? ?????? ? ???????.
    double? MaxFuelVolumeL

    // ?????? ?????????? ? ???????.
    double? FuelingTotalVolumeL

    // ?????????? ?????????? ? ???????.
    double? DefuelingTotalVolumeL

    // ?????? ?????????? ?????????? ? ???????.
    double? TotalFuelConsumptionL

    // ?????????????? ???????????.

    int FuelingCount

    // ?????????????? ??????????.
    int DefuelingCount

    // ?????????????? ??????????.
    SpicFuelingDefuelingStatisticsEvent[] Events
}
```

Контракт данных SpicFuelingDefuelingStatisticsEvent

```
SpicFuelingDefuelingStatisticsEvent
{
    // ??????? ?????????????????????? ??????????.
    SpicDateTimeRange Period

    // ??????? ?????????????????????? ?????????????? ?????????? (??? ?????????????????? ??????????).
    SpicDateTimeRange OriginalPeriod
}
```

```

// ?????? ?????????????? ??????????.
    DateTime Timestamp

// ??? ??????????.
    SpicFdStatisticsFuelEventType EventType

// ?????????????? ?????????? ?????????? ? ??????????.
    double BeginFuelVolumeL

// ?????????????? ?????????? ?????????? ? ??????????.
    double EndFuelVolumeL

// ?????????????? ?????????? ?????????? ? ?????????? ??? ?????????????? ?????????? (??? ?????????????????
?????????).
    double? OriginalBeginFuelVolumeL

// ?????????????? ?????????? ?????????? ? ?????????? ??? ?????????????? ?????????? (??? ?????????????????
?????????).
    double? OriginalEndFuelVolumeL

// ?????? ?????????????????????? ??????????.
    SpicLocation Location
}

```

Контракт данных SpicFuelingDefuelingStatisticsResult

```

// ?????????????? ?????????? ?????????????? ?? ?????????????? ? ?????????? ??????????
SpicFuelingDefuelingStatisticsResult
{
    // ?????????????? ? ?????????????? ?????????????? ?????????? ??????????????
    SpicStatisticsChunkInfo ChunkInfo

    // ??????????????. ??? ? ? ?????????????? ?????????????? - null.
    SpicFuelingDefuelingStatistics Statistics
}

```

Контракт данных SpicMotorModesStatistics

```

// ?????????????? ?? ?????????? ?????????? ??????????????
SpicMotorModesStatistics
{
    // ?????????? ?????????? ??????????
    SpicMotorPeriod[] Periods

// ?????? ?????????? ??????????????
    TimeSpan EngineOnHours

// ?????????????? ?????????? ? ?????????????????? ??????????????????
    TimeSpan EngineOffHours

// ?????? ?????????????? ?????????? ??????????????????
    TimeSpan EngineActiveWorkHours

// ?????? ?????????? ?? ?????????????? ??????
    TimeSpan EngineIdleHours
}

```

Контракт данных SpicMotorModesStatisticsResult

```
SpicMotorModesStatisticsResult
{
    // ?????????? ? ?????????? ?????????? ?????? ??????????
    SpicStatisticsChunkInfo ChunkInfo

    // ??????????. ??? ? ?????????? ?????????? - null.
    SpicMotorModesStatistics Statistics
}
```

Контракт данных SpicMotorPeriod

```
// ?????? ?????? ??????????
SpicMotorPeriod
{
    // ??? ???????
    // ?????????, ????????? ??????, ?????????? ?? ??????
    SpicMotorPeriodType TypeId

    // ?????????? ?????? ???????
    SpicDateTimeRange Period

    // ?????????? ?????????? true - ?????????? ?????????, false - ??????
    bool IsIgnitionOn

    // ?????????? ?????????? ?????????? ???????, true - ?????? ?????????, false - ??????
    bool IsActiveWork
}
```

Контракт данных SpicMotorPeriodType

```
// ??? ?????????? ??????.
SpicMotorPeriodType
{
    string Value
}
```

Параметр **Value** может принимать следующие значения:

- **Unknown** – Не удалось определить
- **Movement** - Движение
- **TrailMovement** – Движение на трале
- **ActiveWorkParking** – Активная работа на стоянке
- **ActiveWorkMovement** – Активная работа в движении
- **Parking** - Стоянка
- **IdleParking** - Стоянка на холостом ходу.
- **Break** - Разрыв
- **NoData** – Нет данных

Контракт данных SpicNavigationValidationStatistics

```
// ?????????????? ??????? ?? ?????????????? ?????????????? ?????????????????????? ??????????.
SpicNavigationValidationStatistics
```



```
{
    // ?????????? ?????????? ??????
    int ValidPointsCount

    // ?????????? ?????????? ??????
    int InvalidPointsCount

    // ?????? ??????????
    SpicNavigationPoint[] Points
}
```

Контракт данных SpicNavigationValidationStatisticsResult

```
// ?????????? ?????????? ?????????? ?? ?????????? ?????????????????? ??????
SpicNavigationValidationStatisticsResult
{
    // ?????????? ? ?????????? ?????????? ?????? ??????????
    SpicStatisticsChunkInfo ChunkInfo

    // ??????????. ?????? ?? ?????????? ?????????? - null.
    SpicNavigationValidationStatistics Statistics
}
```

Контракт данных NavigationFiltrationStatisticsRequest

```
// ?????? ?????????? ?????????? ?? ?????????? ?????????????????? ??????
NavigationFiltrationStatisticsRequest
{
    // ?????? ??????????
    SpicStatisticsSession Session

    // ?????????? ?????????? ??????????
    SpicNavigationFiltrationStatisticsSettings Settings
}
```

Контракт данных SpicLocation

```
// ??????????
SpicLocation
{
    // ??????
    double Latitude

    // ??????
    double Longitude
}
```

Контракт данных SpicNavigation

```
//?????????
SpicNavigation
{
    // ??????????.
    SpicLocation Location

    // ?????? ? ??????
    double? AltitudeMeters
}
```

```

// ???? (0 == North, 90 == East)
short? Angle

// ?????????? ??????????
byte? SatellitesCount

// ?????????? (??/? )
double? Speed

// ??? ?????????????? ?????????????? ?????????? (???????)
string NavigationSystemType

// ?????????????? ?????????????? ??????? (GPS Fix ?? ??????????)
bool? HardwareValidation
}

```

Параметр **NavigationSystemType** может принимать следующие значения:

- **Unknown**
- **Gps**
- **Glonass**
- **GpsGlonass**
- **Galileo**
- **Lbs**

Контракт данных SpicNavigationFiltrationStatistics

```

// ?????????? ?? ?????????? ?????????????? ???????
SpicNavigationFiltrationStatistics
{
    // ?????? ??????????
    SpicNavigationPoint[] Points
}

```

Контракт данных SpicNavigationFiltrationStatisticsResult

```

// ?????????? ?????????????? ?????????????? ?? ?????????????? ?????????????????? ???????
SpicNavigationFiltrationStatisticsResult
{
    // ?????????????? ? ?????????????? ?????????????? ??????? ??????????????
    SpicStatisticsChunkInfo ChunkInfo

    // ??????????????. ????? ?? ?????????????? ?????????????? - null.
    SpicNavigationFiltrationStatistics Statistics
}

```

Контракт данных SpicNavigationFiltrationStatisticsSettings

```

// ?????????? ?????????????? ?????????????? ?? ?????????????? ?????????????????? ???????
SpicNavigationFiltrationStatisticsSettings
{
    // ?????????????? ?????????????? ??????
    SpicNavigationValidationFilter NavigationValidationFilter

    // ?????????????? ?????????????? ?????????? ??????
}

```

```
    SpicTrackPeriodsFilter TrackPeriodsFilter
}
```

Контракт данных SpicNavigationPoint

```
// ?????? ??????????
SpicNavigationPoint
{
    // ??????? ?????????? ???????
    DateTime Timestamp

    // ?????????????
    SpicNavigation Navigation

    // ?????????????? ?????????????????? ??????
    bool? IsNavigationValid
}
```

Контракт данных SpicNavigationValidationFilter

```
// ??????????? ?????????????? ?????????????????? ???????
SpicNavigationValidationFilter
{
    // ??????????? ??????????? ??????
    bool ExcludeValidPoints

    // ?????????????? ?????????????? ??????
    bool ExcludeInvalidPoints

    // ?????????????? ??????, ?????????? ?? ?????????????? ??????????????????????
    bool ExcludeNotValidatedPoints
}
```

Контракт данных SpicTrackPeriodsFilter

```
// ?????????????? ?????????????? ?????????????? ??????
SpicTrackPeriodsFilter
{
    // ?????????????? ??????????
    bool ExcludeRecoilPoints

    // ?????????????? ?????? ??????????????
    bool ExcludeNotMovePoints

    // ?????????????? ?????? ??????????
    bool IncludeParkingPoints
}
```

Контракт данных SpicTrackPeriod

```
//????????? ?????????????
SpicTrackPeriod
{
    // ??? ??????????.
    // ??????????, ?????????? ??? ??????????.
    SpicTrackPeriodType Type
}
```

```

// ?????????? ?????? ????????.
// ????????? "?????????" ??? ??????????, ?.?. ?????? ??????? ?????????? ??????????
?? ??????????
// ??????????? ?????????????? ??????????.
SpicDateTimeRange Period
}

```

Контракт данных SpicTrackPeriodsStatistics

```

// ??????????? ?? ??????
SpicTrackPeriodsStatistics
{
// ?????????? ??????????
SpicTrackPeriod[] Periods

// ??????????
SpicTrackRecoil[] Recoils
}

```

Контракт данных SpicTrackPeriodStatisticsResult

```

// ??????????? ?????????????? ?????????????? ?? ??????
SpicTrackPeriodStatisticsResult
{
// ?????????????? ? ?????????????? ?????????????? ??????? ??????????????
SpicStatisticsChunkInfo ChunkInfo

// ??????????????. ?????? ?? ?????????????? ?????????????? - null.
SpicTrackPeriodsStatistics Statistics
}

```

Контракт данных SpicTrackPeriodType

```

// ??? ?????????? ??????.
SpicTrackPeriodType
{
string Value
}

```

Параметр **Value** может принимать следующие значения:

- **Movement**
- **Parking**
- **Break**
- **NoData**
- **Unknown**

Контракт данных SpicTrackRecoil

```

// ???????
SpicTrackRecoil
{
// ?????????? ?????????? ??????????
SpicDateTimeRange Period
}

```

Контракт данных SpicTrackPeriodMileage

```
SpicTrackPeriodMileage
{
    // ??? ??????.
    // ????????, ????????.
    SpicTrackPeriodType Type

    // ?????????? ?????? ??????????.
    // ????????? "?????????" ??? ??????????, ?.?. ?????? ??????? ?????????? ??????????
    ?? ??????????
    // ?????????????? ?????????????? ??????????.
    SpicDateTimeRange Period

    // ??????? ? ??????? ??????????.
    // ?????????????? ?????????????? ?????????????? ?????????????? ? ??????? ?????????? ??????.
    double? MileageKm

    // ??????? ? ?????????? ??????, ??
    double? WorkTimeMileageKm

    // ????????? ??????????, ??/?
    double? AverageSpeedKmh

    // ?????????????? ??????????, ??/?
    double? MinSpeedKmh

    // ?????????????????? ??????????, ??/?
    double? MaxSpeedKmh
}
```

Контракт данных SpicTrackPeriodsMileageStatistics

```
// ?????????????? ?????? ? ?????????? ?? ?????????? ??????????.
SpicTrackPeriodsMileageStatistics
{
    // ?????????? ?? ?????????? ??????.
    SpicTrackPeriodMileage[] PeriodsMileage

    // ?????? ?????? ? ??????????.
    TimeSpan MovementDuration

    // Общее время на стоянках.
    TimeSpan ParkingDuration

    // ?????? ?????? ? ??????????.
    TimeSpan BreakDuration

    // ?????? ??????????????????????????.
    TimeSpan TotalDuration

    // ????????? ? ??????????, ???.
    double? MovementMileageKm

    // ????????? ? ??????????, ???.
    double? BreakMileageKm
}
```

```

// ?????? ???????, ??
double? TotalMileageKm

// ??????? ? ???????? ??????, ??
double? WorkTimeMileageKm

// ?????????????? ??????????, ??/?
double? MinSpeedKmh

// ?????????????? ??????????, ??/?
double? MaxSpeedKmh
}

```

Контракт данных SpicTrackPeriodsMileageStatisticsResult

```

// ?????????? ?????????????? ?????????????? ?? ?????? ? ?????????? ?? ?????????? ??????????
SpicTrackPeriodsMileageStatisticsResult
{
    // ?????????????? ? ?????????????? ?????????????? ??????? ??????????????
    SpicStatisticsChunkInfo ChunkInfo

    // ??????????????. ????? ?? ?????????????? ?????????????? - null.
    SpicTrackPeriodsMileageStatistics Statistics
}

```

Контракт данных SpicOdometerStatisticsResult

Внимание. В данный момент функционал контракта не доступен на сервере СКАУТ 365.

```

// ?????????????? ?????????????? ?????????????? ?? ?????????????? ??????????, ?????????????????? ?? ??????????????
SpicOdometerStatisticsResult
{
    // ???????, ??
    double? MileageKm
    // ?????????????? ?? ????????.
    string Error;
}

```

в json будет полученный ответ будет выглядеть так:

{ MileageKm: 235.76, Error = null } или так { MileageKm: null, Error = "NoData" }

В ПО выбор датчика определяется следующим образом:

1. Если есть физические одометры (логический датчик), то берётся первый из них;
2. Если нет одометров, то берём виртуальный одометр (логический датчик);
3. Если нет виртуальных одометров, то берём общий пробег по CAN (логический датчик TotalMileage);
4. Если не получилось выбрать ни одного датчика – возвращаем ошибку NoSensor;

5. Если получилось выбрать, то переходим к определению значения.

Значение датчика определяется так:

1. Если за заданный период нет ни одного значения – отдаём ошибку NoData;
2. Если за заданный период есть хотя бы одно значение – отдаём последнее из них.

Обратите внимание!

Из-за особенностей работы ПО (расчёта статистики с учётом нахлёста в 12 часов), запрашивать данные с использованием статистики SpicOdometerStatisticsResult можно только за сутки. В случае запроса статистики за более короткий промежуток, или в случае отсутствия данных, показания будут переданы с некорректным значением времени.

Например: При запросе с 09.03.2017 00:00 по 10.03.2017 00:00 данных не будет (терминал их ещё не выгрузил), но в базе найдётся значение например за 08.03.2017 17:00 равное 32000 км, оно и будет выдано, но со временем 09.03.2017 00:00.

Чтобы избежать потенциальных проблем и не получить такое значение, рекомендуем делать проверку, получая статистики по периодам движения и стоянок и формируя запрос по статистике Одометра только по уже выгруженным на сервер данным.

[<<Назад](#)
[оглавлению](#)

[К](#)

[Далее>>](#)