FTLSupporter használata

Verzió:4.0.0

Tartalom:

Tartalomjegyzék

[FTLSupport – Járművek szállítási feladatokhoz rendelése minősítéssel 1](#_Toc489037931)

[Működés lépései: 3](#_Toc489037932)

[AETR szabályok 3](#_Toc489037933)

[Paraméterek: 4](#_Toc489037934)

[Visszatérési érték: 5](#_Toc489037935)

[Hibaüzenetek: 6](#_Toc489037936)

[FTLSupportX – Szállítási feladatokhoz egy teljesítő jármű meghatározása 7](#_Toc489037937)

[Paraméterek: 7](#_Toc489037938)

[Visszatérési érték: 7](#_Toc489037939)

[Mezőnév-változások 8](#_Toc489037940)

[3.8.0 8](#_Toc489037941)

[4.0.0 8](#_Toc489037942)

[Egy verzió használata előtt végrehajtandó adatbázis-műveletek 8](#_Toc489037943)

[3.7.7 8](#_Toc489037944)

[3.8.0 9](#_Toc489037945)

[3.8.0 10](#_Toc489037946)

## FTLSupport – Járművek szállítási feladatokhoz rendelése minősítéssel

Az FTLInterface.FTLSupport célja, a teljesítendő szállítási feladatok beosztásának felhasználói támogatása. Ehhez a program minden egyes átadott szállítási feladathoz a megkapott járműadatok alapján egy-egy teljesítés-listát ad vissza. A teljesítés-listákban a költség alapján egy sorrend van meghatározva, evvel is segítve a megfelelő választást.

Az átadott járműveknek három féle típusa lehetséges:

* Egy időpillanatban egy koordinátán elérhető (Available): A jármű pl. áll egy telephelyen és a megadott időpontban felhasználható
* Tervezett túrája van (Planned): Ez esetben a járműnek már meghatároztak egy feladatot, ami teljesítése után osztható be a kapott szállítási feladat. A program a tervezett túra pontjainak alapján meghatározza az érkezés/indulás időpontokat és költésgeket (km és útdíj)
* Futó túrája van (Running): A jármű teljesít egy túrát, a befejezés után osztható be a kapott szállítási feladat. A nem teljesített pontok érkezés/indulás időit a program kiszámolja.

A fentieken felül megadható, hogy jármű irányos túrát teljesít-e. Irányos túra esetén a beosztott szállítási feladat teljesítése után nem tér vissza a raktárba.

A program a túrapontok teljesítési sorrendjén nem változtat.

A beosztandó szállítási feladatra meghatározható hogy mely járműtípusok rendelhetőek (TruckTypes) hozzá, illetve az egyes járművek mely árutípustokat szállíthatják (CargoTypes). Ezen paraméterek beállításával megadható például hogy húsárut (CargoType) hűtős (TruckType) jármű szállíthat. Egy jármű többféle árutípust is szállíthat, illetve egy szállítási feladat teljesítése is megoldható többféle járműtípussal. Amennyiben a CargoType és TruckType paraméterek nincsenek kitöltve, nincs korlátozás.

A fentieken felül megadható az egyes túrapontokra, hogy mely járműtulajdonságok (TruckTypes) engedélyezettek ill. tiltottak a teljesítésben (InclTruckProps ill ExclTruckProps).

Egy beosztandó szállítási feladathoz az alábbi esetben rendelhető jármű (zöld háttérben a vonatkozó üzenet):

* A jármű típusa (TruckType) nem megfelelő:Járműtípus miatt nem teljesítheti a túrát!
* Árutípus (CargoType) nem megfelelő: Árutípus miatt nem teljesítheti a túrát!
* A jármű kapacitása (Capacity) nagyobb, vagy egyenlő a beosztandó feladat súlyával (Weight): Kapacitás miatt nem teljesítheti a túrát!
* A jármű mind a futó- mind a beosztandó túrák összes pontjára a zárás előtt érkezik (nyitás előtti érkezés esetén a jármű várakozik):
* Túrapont már zárva a számítás időpontjában (Running típusú jármű esetén)
* Túrapont zárva az érkezés időpontjában
* A túra minden szakaszára találunk útvonalat. (Ha egy túrapont a jármű számára tiltott övezetben van, nincs oda vezető útvonal):
* Aktuális túra teljesítésénél hiányzó szakasz!
* Átállásnál hiányzó szakasz!
* Beosztandó túra teljesítésénél hiányzó szakasz!
* Visszatérés teljesítésénél hiányzó szakasz!
* A jármű max. teljesíthető távolságát nem lépjük túl : Teljesítés max. idő túllépés!
* A jármű max. teljesítési idejét nem lépjük túl: Teljesítés max. KM túllépés!
* Az AETR szabályok alapján a jármű max. vezetési idejét nem lépjük túl:
  + Vezetési idő túllépés I. túra teljesítésnél!
  + Vezetési idő túllépés átállásnál!
  + Vezetési idő túllépés IT. túra teljesítésnél!
  + Vezetési idő túllépés visszatérésnél!
* A túrapont engedélyezett járműtípusai kitöltöttek (InclTruckProps) és a jármű tulajdonságai (TruckProps) között szerepel a tulajdonság:A járműnek nincs olyan tulajdonsága ami a túrapont teljesítéséhez szükésges! (TruckProps)
* A túrapont tiltott járműtípusai kitöltöttek (InclTruckProps) és a jármű tulajdonságai (TruckProps) között nem szerepel a tulajdonság: A jármű tulajdonsága alapján nem teljesítheti a túrapontot (TruckProps)

### Működés lépései:

* Inicializálás, koordináta illesztés térképre, adatkitöltés validálások
* Az egyes szállítási feladatokhoz a szóba jöhető járművek meghatározása
  + Ha ki van töltve, mely típusú járművek szállíthatják, a megfelelő típusú járművek leválogatása
  + Szállíthatja-e jármű az árutípust?
  + Amennyiben a túrapontra megadtuk. az engedélyező feltételben szerepel-e a járműtulajdonság
  + Amennyiben a túrapontra megadtuk. az tiltó feltételben NEM szerepel-e a járműtulajdonság
  + Járműkapacitás megfelelő?
  + A jármű pillanatnyi időpontja az összes túrapont zárása előtti-e
* Kiszámolandó útvonalak összegyűjtése
* Útvonalszámítások
* Eredmény összeállítása
  + Útvonalak összeállítása
  + Vezetési idők, pihenőidők beállítása AETR szabályok alapján
  + Hibák beállítása
  + Költségszámítások elvégzése

### AETR szabályok

#### Kapott paraméterek

|  |  |
| --- | --- |
| maradék vezetési idő | RemainingDriveTime |
| maradék pihenőidő | RemainingRestTime |
| a pihenőig hátralévő munkaidő | RemainingTimeToStartDailyRest |
| maradék napi vezetési idő | RemainingDailyDriveTime |
| maradék napi pihenőidő | RemainingDailyRestTime |
| sofőr hátralevő heti vezető ideje | RemainingWeeklyDriveTime |
| sofőr hátralevő heti pihenő ideje | RemainingWeeklyRestTime |
| sofőr hátralevő kétheti vezetési ideje | RemainingTwoWeeklyDriveTime |
| sofőr hátralevő kétheti pihenőideje | RemainingTwoWeeklyRestTime |
| kompenzációként adandó pihenőidő  (a soron következő pihenőidőbe számoljuk be) | RemainingRestTimeToCompensate |

 Az időértékek másodpercben vannak megadva.

#### Számítás

Az input adatok alapján két vezetés-pihenés ciklussal tudunk számolni. A

**1.ciklus:**

**vezetési idő:** MIN(RemainingDriveTime, RemainingTimeToStartDailyRes,RemainingWeeklyDriveTime,RemainingTwoWeeklyDriveTime)

**pihenőidő :**MIN(  RemainingRestTime, RemainingDailyRestTime, RemainingWeeklyRestTime, RemainingTwoWeeklyRestTime)+RemainingRestTimeToCompensate

**2.ciklus**

**vezetési idő**: MIN( (RemainingDailyDriveTime- 1.ciklus vezetési idő), RemainingWeeklyDriveTime, RemainingTwoWeeklyDriveTime, RemainingTimeToStartDailyRes)

**pihenőidő** : MIN(RemainingDailyRestTime, RemainingWeeklyRestTime,RemainingTwoWeeklyRestTime)

(A RemainingRestTime nem számít bele a RemainingDailyRestTime időbe)

https://ssl.gstatic.com/ui/v1/icons/mail/images/cleardot.gif

**3.maradék vezetési idő**

**vezetési idő:** MIN( RemainingWeeklyDriveTime, RemainingTwoWeeklyDriveTime)-2.ciklus vezetési idő

**pihenőidő :** nem számolható

### Paraméterek:

* List<FTLTask> p\_TaskList : Beosztandó túrafeladatok listája. Egy számítási menetben több feladat beosztása elvégezhető. Az FTLTask.TPoints listában kell a túrapontokat átadni. Az elemek sorrendje meghatározza a kötelező teljesítési sorrendet.
* List<FTLTruck> p\_TruckList: Hozzárendelhető járművek listája és azok teljesítési információi.Az FTLTruck.CurrTPoints listában kell a jármű aktuális túrájának pontjait megadni. Az elemek sorrendje meghatározza a teljesítési sorrendet is.
* p\_iniPath: PMap.ini paraméterfájl könyvtára
* p\_dbConf: A PMap.ini-ben lévő CTIniFile paraméterállományból mely adatbáziskacsolatot használjuk?
* p\_cacheRoutes: True esetén a kiszámolt útvonalakat letárolja a program, amik egy későbbi futás alkalmával újra felhasználhatóak. Amennyiben valószínűsíthető, hogy a futások többször használnak egyes útvonalakat, érdemes bekapcsolni ezt a lehetőséget.

### Visszatérési érték:

Egy FTLResult objektumokból álló lista. A Status mező alapján háromféle visszatést különböztethetünk meg:

* **EXCEPTION**: A program futása során nem kezelt kivétel történt. A visszatérési érték listája egy elemű. Mezők töltése:
  + ObjectName, ItemID: nincs kitöltve.
  + Data property egy FTLResMsg objektumot tartalmaz, amelynek az alábbi mezői töltöttek:
    - Message: A kivétel szöveges üzenete
    - CallStack:Hivási verem
* **VALIDATIONERROR**: Az átadott FTLTask és/vagy FTLTruck objektumok valamilyen adathibát tartalmaznak. A visszaadott listában minden hibás objektumra egy-egy FTLResult objektumot kapunk. Ha a validálás során hibás adatot talál a program, nincs további feldolgozás! Mezők töltése:
  + ObjectName: Mely típusú objektumban van a hiba.
    - FTLPoint:Túrapont
    - FTLTask:Szállítási feladat
    - FTLTruck:Járműteljesítés
  + ItemID: Az objektum egyedi azonosítója
  + Data: FTLResMsg listát tartalmaz, amelynek az alábbi mezői töltöttek:
    - Field: A hibás property neve
    - Message: A hibaüzenet.
* **RESULT**: A visszaadott lista egy elemű. A Data propertyben lévő FTLCalcTask listábol olvasható ki az eredmény. A tömb annyi elemű ahány eleme van az átadott p\_TaskList listának. Egy FTLCalcTask objektum propertyei:
  + Task:A beosztandó szállítási feladat
  + CalcTours:Túraajánlatok listája:
    - Status: Értéke ERR vagy OK. 'ERR' esetén az Msg listában szöveges üzenet van a problémával kapcsolatban
    - Msg: Hibaüzenetek listája (Status == ERR esetén)
    - Rank:Költségek alapján meghatározott sorrend (a kisebb költség van előbb). A nem teljesíthető túra rank-ja:9999
    - Truck: Erre a járműre vonatkozik a túraajánlat

Az alábbi mezők csak Status == OK esetén értelmezettek:

* + - Futó túra részletezője (T1 kezdetű mezők)
    - Átállás részletezője (Rel kezdetű mezők)
    - Beosztandó túra részletezője (T2 kezdetű mezők)
    - Visszatérés részletezője (Ret kezdetű mezők). Csak körös túra esetén értelmezett
    - Összesítő mezők

### Hibaüzenetek:

#### Validációs hibák:

* *Kötelező mező:xxxxxx!* A mező kitöltését megköveteli a program
* *Helytelen teljesített túrapont érték!* Futó jármű (FTLTruck) objektumban a teljesített túrapont (TPointCompleted) mező értékének nulla és a túrapontok száma -1 (azaz a legutolsó túrapont nincs teljesítve, mert akkor befejezett lenne a túra)
* *Helytelen koordináta!* A kapott koordinátához nem található közeli térképi pont. (adathiba). További részletek a hibaüzenet objektum Field mezőjében

##### Eredmény hibák

* *Járműtípus miatt nem teljesítheti a túrát!* A beosztandó szállítási feladat TruckTypes (teljesítő járműtípusok) felsorolásáben nem szerepel a jármű típusa
* *Árutípus miatt nem teljesítheti a túrát!* A jármű kiszolgálható rakománytípuai (CargoTypes) között nem szerepel a beosztandó feladat rakománytípusa
* *Kapacitás miatt nem teljesítheti a túrát!* A jármű kapacitása (Capacity) nagyobb, vagy egyenlő a beosztandó feladat súlyával (Weight)
* *Túrapont már zárva a számítás időpontjában:xxxxxxxxxxxxx* A program az aktuális követési adatok alapján valósidejű ajánlatokat ad. Ezért, ha a beosztandó szállítási feladat egy pontja a számítás időpontjában már nem elérhető, a feladatra ajánlat nem adható
* *Aktuális túra teljesítésénél/Átállásnál/Beosztandó túra teljesítésénél/Visszatérésnél hiányzó szakasz!* A program két túrapont közötti útvonal meghatározásánál figyelembe veszi az engedélyezett (lásd a FTLTruck.RZones mező leírását!) behajtási zónákat. Amennyiben egy túrapont a jármű számára nem engedélyezett övezetben van, nem kapunk arra a túraszakaszra útvonalat. A másik tipikus probléma lehet, hogy egy nagyobb kiterjedésű lerakó GPS koordinátája valamelyik behajtási övezethez van közel, és a program ebben lévő túrapontot rendeli a hozzá. Például az M5-ös mellett lévő TESCO az autópálya és a Dél-Pest 7,5 T övezet között terül el. Ha ehhez a lerakóhoz megadott a koordináta a Dél-Pest 7,5 T behajtási övezethez van közelebb, akkor az alapján számol a program, emiatt a nehezebb járművek ki lesznek zárva a teljesítésből. **A fentieken felül a program figyelembe veszi a útszakaszokra előírt súly- (GWR), magasság- és szélesség korlátozásokat is***.*
* *Teljesítés max. idő túllépés!*
* Teljesítés max. KM túllépés!
* Vezetési idő túllépés I. túra teljesítésnél!
* Vezetési idő túllépés átállásnál!
* Vezetési idő túllépés IT. túra teljesítésnél!
* Vezetési idő túllépés visszatérésnél!
* *Túrapont zárva az érkezés időpontjában:xxxxxxxxxxxx*  Ha a teljes túrában zárva tartás miatt egy pont is teljesíthetetlen, nincs eredmény
* A járműnek nincs olyan tulajdonsága, ami a túrapont teljesítéséhez szükésges! (TruckProps): A járműnek nincs olyan tulajdonsága, amelyik alapján a túrapont teljesíthető lenne. (azaz a szállítási feladat InclTruckPops mezője nem tartalmaz olyan elemet, amely a jármű TruckProps mezőjében megtalálható)
* A jármű tulajdonsága alapján nem teljesítheti a túrapontot (TruckProps): A járműnek olyan tulajdonsága van, amelyik alapján a túrapont teljesítése nem engedélyezett. (A szállítási feladat ExclTruckPops mezője tartalmaz olyan elemet, amely a jármű TruckProps mezőjében is megtalálható)

## FTLSupportX – Szállítási feladatokhoz egy teljesítő jármű meghatározása

Az FTLInterface.FTLSupportX segítségével az egyes szállítási feladatokhoz egy-egy járművet rendelünk az alábbiak szerint:

1. Vesszük a legjobb helyezésű járművet.
2. Megvizsgáljuk, hogy van-e **másik** olyan szállítási feladat, amelyben az adott jármű a legjobb helyezésű és kisebb átállás+visszatérés költséggel teljesít (azaz kevesebb a veszteség).
3. Amennyiben van ilyen, a következő helyezésű járművet vesszük és ugrunk a 2. pontra.
4. Keletkezhetnek olyan szállítási feladatok, amelyekhez egy menetben nem tudunk járművet rendelni. (pl. azért, mert a járművek minden más túrát kevesebb költséggel teljesítenek). Ebben az esetben a letervezett szállítási feladatok alapján ’Available’ típusú járműteljesítéseket készítünk, amelyben rendelkezésre állás kezdőidőpontja és pozíciója az jármű előző szállítási feladatának utolsó túrapontja.
5. A beosztatlan szállítási feladatokhoz az előző pontban létrehozott járműteljesítések hozzárendelését elvégezzük. Amennyiben ebben a fázisban is marad beosztatlan szállítás, ugrunk a 4-es pontra. Ez a ciklikus feldolgozás addig tart, amíg valamilyen eredményt kapunk vissza.

Megjegyzés: Amennyiben egy járművet szállítási feladathoz rendeltünk ebben a menetben és a járműnek vissza kell térni az indulási helyre, az előző túrateljesítés visszatérés útvonalát és költségét töröljük, és az aktuális szállítási feladat teljesítésében szerepel a visszatérés.

### Paraméterek:

Az FTLSupport-tal megegyezik

### Visszatérési érték:

Az FTLSupport-tal megegyezik, a különbségek:

* A FTLCalcTask. CalcTours túraajánlat lista csak egy elemet tartalmazhat, amelyben Status == OK
* ERR státuszt kapnak azon túraajánlatok is, amelyek vagy más szállítási feladatot teljesítenek, vagy az adott szállítási feladathoz egy másik jármű került hozzárendelésre. Ebben az esetben – az egyéb hibákkal ellentétben- a Rank mező értéke nem lesz 999999. Az Msg mező alapján állapítható meg a kimaradás oka:
* Más szállítási feladatot teljesít!: A jármű egy másik szállítási feladatot kisebb átállás költséggel teljesít.
* Nem teljesíti a szállítási feladatot!:Kiválasztásra került egy túraajánlat, a többi érvényes ajánlat ERR státuszt és ezt az üzenetet kapja
* Hiba a kimaradt szállítási feladatok tervezésében ! (lásd a többi FTLResult tételt): Hiba történt az első menetben kimaradt szállítási feladatok számítási menetében. A részletek a többi FTLResult tételben.

## Mezőnév-változások

### 3.8.0

* FTLCalcTour.T1Duration 🡪 FTLCalcTour.T1FullDuration
* FTLCalcTour.RelDuration 🡪 FTLCalcTour.RelFullDuration
* FTLCalcTour.T2Duration 🡪 FTLCalcTour.T2FullDuration
* FTLCalcTour.RetDuration 🡪 FTLCalcTour.RetFullDuration
* FTLCalcRoute.RouteDuration 🡪 FTLCalcRoute.DrivingDuration

### 4.0.0

Új mezők:

* FTLTruck. Width
* FTLTruck. Height

## Egy verzió használata előtt végrehajtandó adatbázis-műveletek

### 3.7.7

CREATE FUNCTION [dbo].[fnDistanceBetweenLineAndPoint]( @ln1X FLOAT, @ln1Y FLOAT, @ln2X FLOAT, @ln2Y FLOAT, @ptX FLOAT, @ptY FLOAT)

RETURNS FLOAT AS

BEGIN

return abs( (@ln2X-@ln1X)\*(@ln1Y-@ptY) - (@ln1X-@ptX)\* (@ln2Y-@ln1Y)) / sqrt( power(@ln2X-@ln1X,2) + power(@ln2Y-@ln1Y,2))

END

GO

CREATE FUNCTION [dbo].[fnDistanceBetweenSegmentAndPoint]( @ln1X FLOAT, @ln1Y FLOAT, @ln2X FLOAT, @ln2Y FLOAT, @ptX FLOAT, @ptY FLOAT)

RETURNS FLOAT AS

BEGIN

declare @xu float

declare @yu float

declare @xv float

declare @yv float

set @xu = @ptX - @ln1X;

set @yu = @ptY - @ln1Y;

set @xv = @ln2X - @ln1X;

set @yv = @ln2Y - @ln1Y;

if (@xu \* @xv + @yu \* @yv < 0)

return Sqrt(Power(@ptX - @ln1X, 2) + Power(@ptY - @ln1Y, 2));

set @xu = @ptX - @ln2X;

set @yu = @ptY - @ln2Y;

set @xv = -@xv;

set @yv = -@yv;

if (@xu \* @xv + @yu \* @yv < 0)

return Sqrt(Power(@ptX - @ln2X, 2) + Power(@ptY - @ln2Y, 2));

return Abs((@ptX \* (@ln1Y - @ln2Y) + @ptY \* (@ln2X - @ln1X) + (@ln1X \* @ln2Y - @ln2X \* @ln1Y))

/ Sqrt(Power(@ln2X - @ln1X, 2) + Power(@ln2Y - @ln1Y, 2)));

END

### 3.8.0

ALTER TABLE EDG\_EDGE ADD XNOD\_NUM varchar(12)

ALTER TABLE EDG\_EDGE ADD XNOD\_NUM2 varchar(12)

go

update EDG\_EDGE set XNOD\_NUM = NOD\_NUM

update EDG\_EDGE set XNOD\_NUM2 = NOD\_NUM2

go

drop index INX\_EDG\_NOD\_NUM on EDG\_EDGE

ALTER TABLE EDG\_EDGE DROP COLUMN NOD\_NUM

drop index INX\_EDG\_NOD\_NUM2 on EDG\_EDGE

ALTER TABLE EDG\_EDGE DROP COLUMN NOD\_NUM2

ALTER TABLE EDG\_EDGE ADD NOD\_NUM varchar(12)

ALTER TABLE EDG\_EDGE ADD NOD\_NUM2 varchar(12)

update EDG\_EDGE set NOD\_NUM = XNOD\_NUM

update EDG\_EDGE set NOD\_NUM2 =XNOD\_NUM2

ALTER TABLE EDG\_EDGE DROP COLUMN XNOD\_NUM

ALTER TABLE EDG\_EDGE DROP COLUMN XNOD\_NUM2

CREATE NONCLUSTERED INDEX [INX\_EDG\_NOD\_NUM] ON [dbo].[EDG\_EDGE] ([NOD\_NUM])

INCLUDE ([ID],[EDG\_STRNUM1],[EDG\_STRNUM2],[EDG\_STRNUM3],[EDG\_STRNUM4],[EDG\_NAME\_ENC])

CREATE NONCLUSTERED INDEX [INX\_EDG\_NOD\_NUM2] ON [dbo].[EDG\_EDGE] ([NOD\_NUM2])

INCLUDE ([ID],[EDG\_STRNUM1],[EDG\_STRNUM2],[EDG\_STRNUM3],[EDG\_STRNUM4],[EDG\_NAME\_ENC])

go

ALTER TABLE NOD\_NODE ADD XNOD\_NUM varchar(12)

go

update NOD\_NODE set XNOD\_NUM = NOD\_NUM

drop index INX\_NOD\_ZIP\_NUM on NOD\_NODE

ALTER TABLE NOD\_NODE DROP COLUMN NOD\_NUM

ALTER TABLE NOD\_NODE ADD NOD\_NUM varchar(12)

update NOD\_NODE set NOD\_NUM = XNOD\_NUM

ALTER TABLE NOD\_NODE DROP COLUMN XNOD\_NUM

CREATE NONCLUSTERED INDEX [INX\_NOD\_ZIP\_NUM] ON [dbo].[NOD\_NODE] ( [ZIP\_NUM] Asc)

WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, SORT\_IN\_TEMPDB = OFF, IGNORE\_DUP\_KEY = OFF, DROP\_EXISTING = OFF, ONLINE = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

CREATE STATISTICS [statNOD\_NUM] ON [dbo].[NOD\_NODE]([NOD\_NUM])

### 4.0.0

Correct-Tour\_RB.4.0.0\_SWH lefuttatása. Csomópontok, élek, irányítószámok, útdíjjak és behajrási övezetek betöltése