

OTEP — Open Tracking Event Protocol

Fordítás a kényelem érdekében — az angol változat a hiteles.

Állapot: Tervezet v0.1 · Nyílt, szállítófüggetlen specifikáció · Licenc: nyílt (lásd §10)

Az OTEP egy nyílt, szállítófüggetlen protokoll bármely követhető alany életciklusához. Egyetlen eseménymodellt és egyetlen állapotszótárt határoz meg, hogy a követési események bármilyen forrásból — saját flottás kézbesítés, harmadik feles futárok, fuvarozói címkék és azon túl — cserélhetők, érthetők és nemzetközi szabványokra vetíthetők legyenek anélkül, hogy minden féllal újra kellene integrálni.

Ez a dokumentum a specifikáció: a struktúra, a mezők, az állapottáblák, az állapotgép, a külső szabványokra való leképezések és a megfelelési szabályok egy megfelelő implementáció megépítéséhez. Implementációfüggetlen — a protokollt írja le, nem valamely konkrét szállító belső működését. Az RFC-2119 kulcsszavak (MUST / SHOULD / MAY) normatívak.

1. Mit old meg az OTEP

A követés széttöredezett: minden fuvarozó másként nevezi el a mezőit és állapotkódjait, minden kézbesítési csatorna a saját formájában jelent, és a globális szabványokhoz való csatlakozás azt jelenti, hogy újra és újra integrálni kell. Az OTEP egyetlen közös nyelvet ad a producereknek és a fogyasztóknak: egy producer egyszer bocsát ki OTEP eseményeket, és minden OTEP fogyasztó megérti őket, és arra a szabványra vetítheti, amelyikre szüksége van.

2. Tervezési alapelvek

1. Eseményalapú (event-sourced). Az eseménysorrend az igazság forrása; a „jelenlegi állapot” mindig egy vetület — a legutóbbi esemény állapota.
2. Mit / Mikor / Hol / Miért. Minden esemény e négy dimenzió köré épül — ugyanazok a dimenziók, amelyeket a GS1 EPCIS, az IATA ONE Record és a UN/CEFACT is megoszt —, így ezek a szabványok egy OTEP esemény kimeneti vetületei, nem pedig párhuzamos modellek.
3. A lista nélküli források nem akadályoznak. Egy olyan forrás, amely csak a jelenlegi állapotot teszi közzé (történet nélkül), úgy kezelhető, hogy minden megfigyelt változáshoz egy eseményt szintetizálunk (§5).
4. Additív. Az OTEP bármely meglévő követési API mellett elérhető; a bevezetése soha nem igényel törő változtatást abban, amit a fogyasztók már használnak.

3. Az OTEP idővonal

Az idővonal egy boríték, amely a követett alanyt és egy rendezett eseménylistát hordoz.

```
{
  "otep_version": "0.1",
  "profile": "parcel", // domain profile (§8)
  "subject": {
    "tracking_number": "SR123...", // ?1 identifier required
    "order_id": 12345, // optional
    "package_id": 67890, // optional
    "external_tracking_number": "1Z...", // optional
    "gs1_sccc": "00...", // optional — enables EPCIS epcList
    "piece_id": "...", // optional — enables ONE Record linkage
  },
  "current_status": "delivered", // projection of the latest event
  "delivered": true,
  "events": [ /* OTEP events, §4 */ ]
}
```

Az OTEP esemény

```
{
  // WHAT – carried once at the timeline level (subject above)

  // WHEN
  "occurred_at": "2026-06-10T09:30:00-04:00", // event instant, ISO-8601 with offset
  "recorded_at": "2026-06-10T09:45:23-04:00", // ingestion instant (optional)
  "time_type": "actual", // actual | estimated | scheduled

  // WHERE
  "location": {
    "name": "Toronto Hub",
    "code": "YYZ-2",
    "gln": "0614141000005", // GS1 GLN ? EPCIS SGLN
    "lat": 43.6777, "lng": -79.6248,
    "country": "CA" // ISO 3166-1 alpha-2
  },

  // WHY / WHAT HAPPENED
  "status_code": "out_for_delivery", // an OTEP status code (§4)
  "phase": "out_for_delivery", // derived from status_code
  "incident_reason": null, // an OTEP incident reason (§4) when exception

  // WHO
  "actor": { "type": "driver", "name": "Jane D.", "phone": "+1..." },

  // PROVENANCE
  "source": {
    "type": "self_delivery", // self_delivery | third_party_delivery | carrier_label
    "provider_id": null,
    "carrier_code": null,
    "external_event_code": null, // your raw status code (preserve it)
    "raw": { /* original payload */ }
  },

  // PROOF
  "pod": {
    "photos": [ { "url": "...", "content_base64": null } ],
    "signature": [ { "url": "...", "content_base64": null } ],
    "recipient": "John Smith"
  }
}
```

A `subject`, `occurred_at`, `status_code` és `source` kivételével minden mező opcionális — a részleges források azt töltik ki, amijük van. A csomópont/hub szkennelések a `location` mezőt a szkennelő létesítményre állítják. A nagyfrekvenciás GPS-telemetria NEM OTEP esemény; egy esemény állapotváltozáskor vagy csomópont-szkenneléskor keletkezik.

4. Szókészlet

4.1 Állapotkódok

20 kanonikus életciklus-kód. A `phase` mindig levezethető a kódból.

kód	phase	terminális	POD	jelentés
information_submitted	pre_shipment			rendelési információ beérkezett
booking_confirmed	pre_shipment			fuvarozó/foglalás visszaigazolta
awaiting_pickup	pre_shipment			gyűjtésre kész
out_for_pickup	pickup			úton a felvételhez
picked_up	pickup		✓	feladótól átvéve
pickup_failed	exception			felvételi kísérlet sikertelen
pickup_rescheduled	exception			felvétel újrapróbálkozik
received	inbound			létesítményben átvéve
arrival_scan	inbound			beérkezési szkennelés a csomópontonál
in_transit	transit			mozgásban
package_outbound	transit			létesítményből elindult
removed_from_route	exception			útvonalról levéve

kód	phase	terminális	POD	jelentés
route_cancelled	exception			útvonal törölve
out_for_delivery	out_for_delivery			a járművön
delivered	delivered	✓	✓	címzettnek kézbesítve
delivery_failed	exception			kézbesítési kísérlet sikertelen
delivery_rescheduled	exception			újrapróbálkozik / újrakézbesít
return_to_sender	return	✓		visszatérés a feladóhoz
rejected_by_recipient	return	✓		címzett visszautasította
cancelled	return	✓		rendelés törölve

4.2 Fázisok

pre_shipment · preparing · pickup · inbound · in_custody · transit · out_for_delivery · delivered · exception · return . (A preparing és az in_custody a nem-csomag profilok számára van fenntartva — §8.)

4.3 Forrástípusok és időtípusok

source.type : self_delivery · third_party_delivery · carrier_label . time_type : actual · estimated · scheduled (alapértelmezett actual).

4.4 Incidens-okok

Amikor egy esemény az exception fázisban van, SHOULD hordoznia egy incident_reason értéket ebből a normalizált szókészletből:

- Fuvarozó: carrier_damaged_parcel , carrier_sorting_error , carrier_address_not_found , carrier_parcel_lost , carrier_not_enough_time , carrier_vehicle_issue , carrier_capacity_exceeded , carrier_mechanical_delay
- Kiskereskedő/feladó: retailer_cancelled , retailer_incorrect_data , retailer_not_ready , retailer_incorrect_parcel , retailer_incorrect_dimensions , retailer_packaging_issue
- Címzett: consignee_refused , consignee_business_closed , consignee_not_available , consignee_not_home , consignee_cancelled , consignee_verification_failed , consignee_incorrect_address , consignee_access_restricted , consignee_safe_place_unavailable
- Vám: customs_delay , customs_documentation , customs_duties_unpaid , customs_prohibited , customs_inspection
- Vis maior: weather_delay , natural_disaster , force_majeure
- Egyéb: parcel_being_researched , security_issue , regulatory_hold , unknown

5. Állapotgép és csak állapotot közlő források

A fázisok előre haladnak; a kivételek megszakítják, majd visszaoldódnak. A terminális állapotok (delivered , return_to_sender , rejected_by_recipient , cancelled) lezárják az alanyt.

```
pre_shipment ? preparing ? pickup ? inbound ? in_custody ? transit ? out_for_delivery ? delivered ?
                ?????????????? exception ??????????????
                ?????????????? return / cancelled ?
```

Szabályok:

- A fogyasztóknak MUST occurred_at szerint rendezniük az eseményeket, és MUST tolerálniuk a sorrenden kívüli érkezést.
- A terminális állapotba való átmenetek idempotensek; az ismétlések deduplikálódnak.

- Terminális állapot után a producer MUST NOT bocsásson ki további eseményeket, kivéve egy dokumentált RMA / újranyitási folyamatot.
- Egy csak a jelenlegi állapotot közlő forrás MUST minden megfigyelt változáshoz egy eseményt szintetizáljon (stabil deduplikációs kulccsal), ahelyett, hogy kihagyná a történetet. Idővel a pillanatfelvételek idővonallá halmozódnak.

6. Külső szabványokra való leképezések

Az OTEP négy dimenziója mezőről mezőre illeszkedik a fő szabványokhoz, így mindegyik egy OTEP esemény kimeneti vetülete.

OTEP	GS1 EPCIS 2.0	IATA ONE Record	UN/CEFACT
<code>occurred_at (+offset)</code>	<code>eventTime + eventTimeZoneOffset</code>	<code>eventDate + eventTimeType=Actual</code>	Event Date/Time
<code>recorded_at</code>	<code>recordTime</code>	<code>recordedAt</code>	—
<code>subject (sscc/piece)</code>	<code>epcList (SSCC URN)</code>	<code>linkedObject</code> → Piece/Shipment	Consignment
<code>location (gln/lat/lng)</code>	<code>readPoint / bizLocation (SGLN)</code>	<code>recordedAtLocation</code> → Location	Location
<code>status_code</code>	<code>bizStep + disposition</code>	<code>eventCode</code>	Transport status code
<code>actor</code>	<code>sourceList / extension</code>	Actor / Party	Party
<code>incident_reason</code>	<code>disposition / ErrorDeclaration</code>	<code>event remark</code>	Status reason code

A kódonkénti értékek (EPCIS CBV `bizStep / disposition`, ONE Record `eventCode`, UN/CEFACT kód) a gépi olvasású kódkönyvben kerülnek közzétételre. E nemzetközi szabványokon túl egy OTEP idővonal OpenTelemetry trace-ekre, OGC SensorThings megfigyelésekre és gyakori kereskedelmi platformokra (AfterShip, Shopify, Amazon, Walmart, BigCommerce, Magento, WooCommerce, Etsy) is vetíthető.

Bizonyosság: az EPCIS CBV értékek stabil szabványos URN-ek. A ONE Record és UN/CEFACT kódértékek a végső kézbesítésre vett legjobb illeszkedések, és külső használat előtt ellenőrizni kell őket a hivatalos kódlistákkal szemben. A „címzettnek kézbesítve” esetnek nincs pontos CBV `bizStep`-je — a legközelebbi illeszkedés (`receiving` + `received`) használatos, vagy egy felhasználói szókészlet-kiterjesztés URN.

7. Az OTEP API

Az OTEP egy kis, csak olvasható HTTP felületen keresztül fogyasztható; minden végpont nyilvános.

Ige	Útvonal	Visszaad
GET	<code>/api/v1/otep/trackings/{tracking_number}</code>	egy követési szám idővonalát
GET	<code>/api/v1/otep/trackings/{tracking_number}/events</code>	csak az eseményeket
POST	<code>/api/v1/otep/trackings/batch</code>	sok követési szám egy hívásban
POST	<code>/api/v1/otep/validate</code>	megfelelőségi ellenőrzés egy beküldött idővonalra (§9)

Egy GraphQL lekérdezés, amely ugyanazt az idővonalat teszi elérhetővé, szintén rendelkezésre áll.

Tartalomjegyzetetés

Ugyanaz az idővonal abba a reprezentációba szerializálódik, amelyiket kérsz, egy `?format=` lekérdezési paraméteren vagy egy `Accept` profilon keresztül:

Kérés	Reprezentáció
<code>?format=otep</code> (alapértelmezett)	natív OTEP idővonal
<code>?format=epcis</code>	GS1 EPCIS 2.0 (JSON-LD <code>ObjectEvent</code> -ek)
<code>?format=onerecord</code>	IATA ONE Record (<code>LogisticsEvent</code> -ek)

Kérés	Reprezentáció
?format=uncefact	UN/CEFACT szállítási állapot
?format=otlp	OpenTelemetry trace-ek
?format=sensorthings	OGC SensorThings megfigyelések
?format=aftership shopify amazon walmart bigcommerce magento woocommerce etsy	a platform követési/teljesítési formája

Azok az események, amelyekhez egy vetületben nem rendelhető kód, kihagyásra és megszámlálásra kerülnek (soha nem dobódnak el csendben).

8. Domain profilok

Az OTEP egy általános protokoll, nem csomagprotokoll. A protokollréteg (eseményboríték, fázisgerinc, állapotgép) univerzális; a konkrét állapotkódok egy profilhoz tartoznak, amelyet az idővonalon a `profile` deklarál. A §4 kódjai a **parcel** profil. Más domaineik — költöztetés, ételszállítás, tárolás és azon túl — saját kódkészleteket adnak a profiljuk alatt, `otep:<profile>:<code>` névtérrel, mindegyik ugyanahhoz a fázisgerinchez illeszkedve. Egy profil hozzáadása kiterjesztés, nem protokollváltozás.

9. Megfelelőségi specifikáció (normatív)

Egy producer vagy fogyasztó akkor OTEP-megfelelő, ha minden esemény, amelyet kibocsát vagy elfogad, kielégíti ezeket a táblákat és szabályokat.

9.1 Idővonal-boríték

Mező	Típus	Köt.	Megszorítások
otep_version	string	MUST	semver, pl. 0.1
profile	string	MUST	regisztrált profil
subject	object	MUST	§9.2
current_status	string null	SHOULD	egy állapotkód (§4)
current_phase	string null	SHOULD	MUST egyezzen a <code>current_status</code> fáziséval, ha mindkettő jelen van
delivered	boolean	SHOULD	<code>true</code> akkor és csak akkor, ha <code>current_status = delivered</code>
events	array	MUST	eseményobjektumok (§9.3), <code>occurred_at</code> szerint rendezhetők

9.2 subject

A `tracking_number` / `order_id` / `package_id` közül legalább EGY MUST jelen legyen.

Mező	Típus	Megszorítások
tracking_number	string	nem üres
order_id / package_id	integer null	
external_tracking_number	string null	
gs1_sssc	string null	18 számjegy
piece_id	string null	

9.3 Eseményobjektum

#	Mező	Típus	Köt.	Megszorítások
1	occurred_at	string	MUST	ISO-8601 eltolással
2	recorded_at	string null	SHOULD	ISO-8601

#	Mező	Típus	Köt.	Megszorítások
3	time_type	string	MAY (alapértelmezett actual)	actual estimated scheduled
4	status_code	string null	MUST ¹	egy kód a §4.1-ben
5	phase	string null	SHOULD	MUST egyezzen a status_code fáziséval
6	incident_reason	string null	SHOULD ²	egy ok a §4.4-ben
7	description	string null	MAY	ember által olvasható
8	location	object null	MAY	§9.4
9	actor	object null	MAY	{ type, name?, phone? }; type ∈ {driver, operator, carrier, system}
10	source	object	MUST	§9.5
11	pod	object null	MAY ³	{ photos[], signature[], recipient? }

¹ Egy kódolt esemény MUST hordozzon egy állapotkódot a §4.1-ből. Egy nyers szkennelés, amelyet még nem tudsz besorolni, MAY status_code = null értékre állítsa, de MUST megőrizze a natív kódot a source.external_event_code mezőben, és MUST megszámolódjon, soha ne dobódjon el. ² Egy exception-fázisú esemény SHOULD hordozzon egy incident_reason értéket. ³ Azok az események, ahol status_code ∈ {delivered, picked_up}, SHOULD hordozzanak egy pod értéket.

9.4 location

name (string) · code (string) · gln (GS1 GLN, 13 számjegy) · lat / lng (WGS-84) · country (ISO 3166-1 alpha-2). Mind opcionális.

9.5 source

Mező	Típus	Köt.	Megszorítások
type	string	MUST	self_delivery third_party_delivery carrier_label
provider_id	integer null	MAY	
carrier_code	string null	MAY	
external_event_code	string null	SHOULD ⁴	a nyers állapotkódod
raw	object null	MAY	eredeti payload

⁴ MUST jelen legyen, amikor a status_code null, így a natív kód soha nem vész el.

9.6 Szabályok

- Idő. Az occurred_at MUST ISO-8601 formátumként értelmezhető legyen. A numerikus epoch értékeket és a .NET /Date(ms)/ formákat normalizáld a beolvasáskor; ne ezeket a formákat bocsásd ki.
- Rendezés / deduplikáció. A fogyasztóknak MUST occurred_at szerint rendezniük, tolerálniuk a sorrenden kívüli érkezést, és deduplikálniuk a (subject, status_code, occurred_at) alapján.
- Állapotgép. Terminális állapot után ne bocsáss ki további eseményeket, kivéve egy dokumentált RMA / újranyitást.
- Nincs csendes veszteség. A nem kódolható eseményeket MUST megszámolni, soha nem eldobni.
- Levezetés. A phase, current_status, current_phase, delivered vetületek — ha jelen vannak, MUST konzisztensek legyenek az eseményidővonallal.

9.7 Megfelelőségi szintek és validáció

- 1. szint — kibocsátja a borítékot (§9.1), a kötelező mezőkkel rendelkező eseményeket (§9.3), érvényes állapotkódokat (§4.1), érvényes fázisokat (§4.2), és betartja az állapotgépet (§5).
- 2. szint — emellett kibocsát legalább egy külső szabványra való vetületet (§6), és ahol alkalmazható, profilspecifikus kódokat (§8).

Ellenőrizd a kimenetedet egy idővonal beküldésével a `POST /api/v1/otep/validate` végpontra. Minden `errors` értéket kezelj blokkolóként; foglalkozz a `warnings` értékekkel. Egy gépi olvasású JSON Schema és a teljes kódkönyv (minden állapotkód, fázis és külső leképezés) offline validációhoz közzétételre kerül.

10. Nyíltság és irányítás

Az OTEP egy nyílt specifikáció, amelyet bármely fél szabadon implementálhat.

- Normatív vs. informatív. Normatív: az eseményboríték, a fázisgerinc, az állapotszókészlet, az állapotgép és a külső szabványra való mezőleképezések. Hogy egy implementáló hogyan köti az OTEP-et a saját belső rendszereihez, az a saját ügye, és itt nem tárgyal.
- Stabil azonosítók. A kódok `otep:<profile>:<code>`, a fázisok `otep:phase:<name>` formában címezhetők. Ha egyszer egy kiadott verzióban közzétételre kerül, egy azonosító jelentése megváltoztathatatlan.
- Verziózás. Szemantikus verziózás. Kódok/profilok hozzáadása MINOR (visszafelé kompatibilis) változás; egy meglévő kód jelentésének megváltoztatása MAJOR változás, és SHOULD elkerülni. A protokollverzió minden idővonallal utazik (`otep_version`).
- Kiterjesztés. Az új profilokat és kódokat e specifikáció ellenében javasolják, nem forkolják, így a független implementálók konvergálnak. A kísérleti kódok MAY `x-` előtagot használjanak (`otep:parcel:x-my_code`), amíg regisztrálva nincsenek.
- Licenc. A specifikációt nyílt licenc alatt szándékozzuk kiadni — TBD, jóváhagyásra várva.

11. Szállítók közötti interoperabilitás — hozd a saját szabványod

Az OTEP üdvözl, hogy más szállítók behozzák a saját követésiesemény-szabványukat, hogy az OTEP együttműködhessen vele, mindkét irányban:

- OTEP → a te szabványod vetítése. Definiálj egy leképezést egy OTEP idővonalról a saját formátumodra, újrahasználva az OTEP állapotszókészletet. Ez egy tiszta transzformáció — idővonal be, a te struktúrád ki —, így a leképezés egyszer íródik meg, és minden OTEP producer kibocsáthatja a formátumodat.
- A te szabványod → OTEP leképezése. Adj egy megfeleltetést a saját állapotszókészletedről az OTEP kódokra (§4), plusz, ha szükséges, egy profilra (§8). A nyers kódjaid megőrződnek a `source.external_event_code` mezőben; a nem leképezett kódok megszámolódnak, soha nem dobódnak el.

Még ha nem is tudod közvetlenül átvenni az OTEP-et, szívesen látott, hogy egyetlen normalizált `otep_status` értéket adj a saját API-válaszaidhoz, és megoszd a követésiesemény-kódjaidat megfeleltetési leképezéshez. A leképezéseket e specifikáció ellenében javasold (nem forkolva), hogy az implementálók konvergáljanak; az új formátumok ugyanabba a `?format=` tartalomjegyeztetésbe csatlakoznak, és soha nem törnek meg egy meglévő fogyasztót.