

EKAER Management Service

Tartalomjegyzék

1	Bevezetés	4
1.1	Célja.....	4
1.2	XML feltöltése az EKAER WEB-es felületen.....	4
2	Bejelentések struktúrája, felépítése és XML struktúrában való leképezése	4
2.1	Az XSD-ben definiált alapvető üzenettípusok.....	4
2.2	Az XML üzenetek általános felépítése	5
2.2.1	Header XML rész	5
2.2.2	User XML rész.....	6
2.2.3	A requestSignature generálása	8
2.3	ManageTradeCardsRequest, bejelentések kezelése (create, modify, delete)	9
2.3.1	TradeCardOperations.....	9
2.3.1.1	Create operation, bejelentés rögzítése.....	10
2.3.1.2	Modify operation, bejelentés módosítása.....	10
2.3.1.3	Delete operation, bejelentés törlése	11
2.3.1.4	Finalize operation, bejelentés véglegesítése	11
2.3.2	TradeCard element felépítése	11
2.3.2.1	TradeCard adatok.....	11
2.3.2.2	A bejelentésben megadott adatok ellenőrzése	14
2.3.2.3	A lerakodási és felrakodási címadat elem felépítése, mezői	15
2.3.2.4	A címadatok ellenőrzése	16
2.3.2.5	Országok listája	17
2.3.2.6	Items lista felépítése (tradeCardItem).....	17
2.3.2.7	Tételekkel kapcsolatos ellenőrzések.....	19
2.3.2.8	Fuvar oka (tradeReason).....	20
2.4	ManageTradeCardsResponse, a válasz felépítése	20
2.4.1	OperationResult felépítése	21
2.4.1.1	Result felépítése (OperationResultType).....	23
2.4.1.2	tradeCardInfo element felépítése.....	23

2.4.1.3	Bejelentés státuszai (status)	26
2.5	Biztosíték számítás folyamata, lépései	27
2.6	queryTradeCardsRequest felépítése.....	28
2.6.1	EKAER szám alapján (tcn) történő lekérdezés	29
2.6.2	queryParams ban megadható feltételek	30
2.7	queryTradeCardsResponse felépítése, a lekérdezésre adott válasz struktúra	31
3	Szolgáltatás technikai leírás	33
3.1	Általános technikai adatok.....	33
3.2	Operations	33
3.3	HTTP Headers.....	33
3.4	HTTP Status codes.....	33
3.5	Result element a válaszüzenetben	33
3.5.1	ReasonCode enumerált típusok.....	34
4	Melléklet	35
4.1	Példa XML-ek	36
4.2	Interface verziók	36
4.2.1	„1.0-ás verzió”	36
4.2.2	„1.6-os verzió”	36
4.3	Teszt rendszer elérhetősége	37
4.4	Éles rendszer elérhetősége	37

Ábrajegyzék

1. ábra	Header element felépítése	6
2. ábra	userelement felépítése	8
3. ábra	tradeCardOperation element felépítése	10
4. ábra	manageTradeCardsResponse element felépítése	21
5. ábra	tradeCardOperationsResults felépítése.....	22
6. ábra	queryTradeCardsRequest feltétel choice felépítése.....	29
7. ábra	queryTradeCardsResponse felépítése	32

Verzió

Név	Dátum	Verzió	Változás röviden
B. G.	2014.12.10	1.0	initial

K. B.	2014.12.12	1.1	lektorált
B. G.	2014.12.13	1.2	validáció kiegészítés, reasonCodes, ország lista, Teszt rendszer elérhetőség
B. G.	2014.12.23	1.3	hibakódok, + unloadReporter mező. carrier/carrierText choice megszűnt! save(Un)LoadLocation mező. Új tcnValidityStart és End mezők a válaszban!
B.G.	2015.01.10	1.4	TradeReason leírás, validáció! Item-ben a value és weight max 9 jegyű egész lehet! Technikai leírás pontosítás.
B.G	2015.01.12	1.5	Query operation kifejtés. StreetNumber nező 10 hossza rövidült! Ábrajegyzék, XML struktúra ábrák!
B.G		1.6	<ul style="list-style-type: none"> - Címadat leírás javítás. - StreetType nem kötelező. - Interface verzió és környezetek leírásának bővítése. - itemExternalId bevezetése 1.6-os verziótól a bejelentés tételeken. - factoryItemNumber, importerItemNumber 200 hosszú lett - ADRNumber hossz és pattern módosult - isIntermodal flag a bejelentéseken - vehicle3 kikerült a bejelentés adatokból! - telefonszám mező formátum leírás bővítése

1 BEVEZETÉS

Elindul az Elektronikus Kereskedelmi és Áruforgalom Ellenőrző Rendszer (továbbiakban: EKAER), amelyben a kereskedelmi tevékenységek során a törvény által előírt esetekben és módon regisztrálni kell a kereskedelmi tevékenységeket, fuvarokat, árumozgásokat (továbbiakban: bejelentés). Az EKAER-be regisztrált kereskedelmi tevékenységekről rögzített bejelentéseket a következő módokon lehet kezelni:

- WEB-es felület GUI-ján (Grafikus felületén) keresztül
- WEB-en XML file feltöltéssel
- Gép-gép kommunikációt támogató szolgáltatáson keresztül

A specifikáció, az elkészítésének pillanatában ismert feltételeknek és törvényi előírásoknak megfelelően készült! Ha változnak a jogszabályi, törvényi elvárások, változni fog a specifikáció is!

A WEB-es felület és az XML alapú bejelentések közötti különbség:

A web-en a bejelentés létrehozásakor "Tervezés alatt" státusszal jön létre, és nem kap automatikusan EKAER számot, míg **XML alapú kommunikáció esetén egyből aktív státuszba lép a bejelentés és EKAER számot is kap**, valamint a szükséges biztosítékszámítás is megtörténik!

1.1 CÉLJA

Jelen dokumentum célja az XML file feltöltés és gép-gép kommunikációt támogató szolgáltatás által használt XML struktúra, valamint a gép-gép kommunikációt leíró szolgáltatás használatának ismertetése.

A WEB-en feltöltött és válaszban visszakapott XML file struktúra és a szolgáltatás által használt XML struktúra megegyezik! Tehát ugyanolyan fájl kell a weben feltölteni, mint amilyen a szolgáltatás megszólításához szükséges XML struktúra. Tehát ugyan olyan file-t kell feltölteni a WEB-en, mint amilyen XML struktúrával kell megszólítani a szolgáltatást.

1.2 XML FELTÖLTÉSE AZ EKAER WEB-ES FELÜLETEN

Jelen dokumentumban részletezett XML struktúrát és műveleteket közvetlen gép-gép kommunikáció mellett a WEB-es felületen is feltölthetik a felhasználók, bejelentkezés után!

Az EKAER WEB-es felületen külön funkció van az xml file feltöltésére, aminek hatására egy XML válasz file letöltése indul be! A letöltött file-ban a dokumentációban definiált válasz XML lesz.

2 BEJELENTÉSEK STRUKTÚRÁJA, FELÉPÍTÉSE ÉS XML STRUKTÚRÁBAN VALÓ LEKÉPEZÉSE

Ebben a fejezetben bemutatjuk az XML és a bejelentések felépítését, a bejelentésekkel kapcsolatos belső logikai összefüggéseket és adattartalmakat.

2.1 AZ XSD-BEN DEFINIÁLT ALAPVETŐ ÜZENETTÍPUSOK

A mellékelt XSD-ben a következő üzenettípusok (element) vannak definiálva:

- **manageTradeCardsRequest:** Ez az üzenet a bejelentések módosítására, létrehozására, törlésére szolgál. Ebben a struktúrában lista formájában vannak átadva a bejelentésekkel kapcsolatos műveletek. Ennek megfelelő XML –t kell feltölteni a WEB-en, vagy átadni a szolgáltatásnak, melynek hatására az EKAER rendszer elvégzi az üzenetben kért műveleteket.
- **manageTradeCardsResponse:** A **manageTradeCardsRequest** üzenet feldolgozása során keletkezett válaszüzenetet írja le. Az EKAER rendszer egy ilyen felépítésű XML-t ad válaszul a **manageTradeCardsRequest**-re.
- **queryTradeCardsRequest:** A korábban rögzített bejelentések lekérdezésére használható XML felépítését írja le. Az üzenetben a lekérdezés paraméterei vannak.
- **queryTradeCardsResponse:** a **queryTradeCardsRequest** –re adott válasz XML struktúráját definiálja. A lekérdezés eredményét tartalmazza. A lekérdezésnek megfelelő bejelentéseket tartalmazza listaszerűen.

2.2 AZ XML ÜZENETEK ÁLTALÁNOS FELÉPÍTÉSE

Minden üzenetnek van **Header** és **User** része. Ezek általánosan minden üzenetben megtalálhatók. Üzenetváltásokkal kapcsolatos technikai és azonosításra szolgáló mezőket tartalmaznak.

2.2.1 HEADER XML RÉSZ

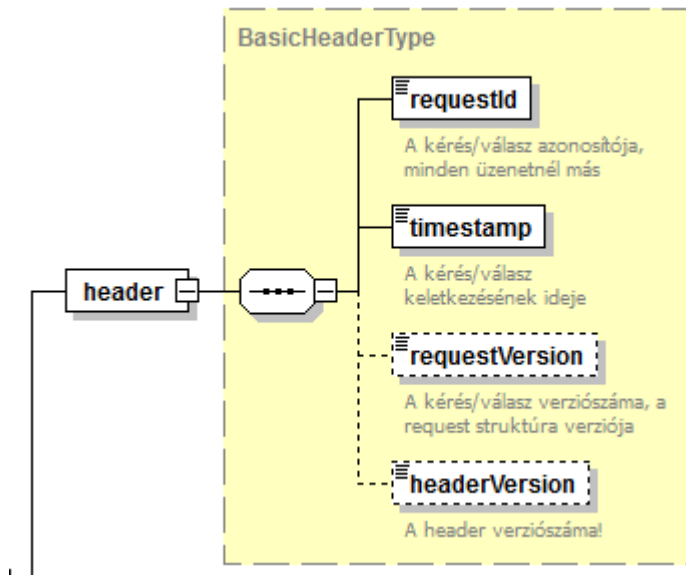
A Header-ben az üzenetváltással kapcsolatos általános technikai adatok vannak. Ezek segítségével lehet az egyes kéréseket beazonosítani, a kérés/válaszokat összepárosítani, valamint tartalmaznak általánosan megkövetelhető technikai mezőket.

Mezőnév	Típus	Kötelező	Leírás	Minta
requestId	50 hosszú szöveg.	Igen	Az üzenet egyedi azonosítója. Minden üzenetnek egyedi azonosítót kell adni!	1EM9C109707208L
timestamp	xsd szabvány szerinti dateTime	Igen	A kérés létrehozásának időpontja. Gép-gép kommunikációnál a kérés időpontjának felel meg!	2014-12-05T17:10:00+01:00
requestVersion	Max 6 hosszú szöveg. Alapértelmezett: 1.0 értékkel. Maszk: ##.###. ponttal elválasztott egész	Nem, alapértelmezés 1.0	A kérés verziószámát tartalmazza. A kérés üzleti struktúra	1.0

	számok		változásánál lehet használni a későbbiekben.	
headerVersion	Max 6 hosszú szöveg. Alapértelmezett: 1.0 értékkel. Maszk: ##.###. ponttal elválasztott egész számok	Nem, alapértelmezés 1.0	A kérés verziószámát tartalmazza. A header struktúra változásánál lehet használni a későbbiekben.	1.0

Az egyes mezőkkel kapcsolatos megkötések:

- **A requestId user-enként egyedi kell, hogy legyen!** A rendszer ugyanattól a User-től nem fogad el két ugyanolyan requestId-val kérést!
- **A szerver nem fogad el 24 óránál régebbi timestamp értékkel érkező kéréseket és jövőbeni időpontot. Szerveridőhöz képest 5 perc tűrés van.**



1. ábra Header element felépítése

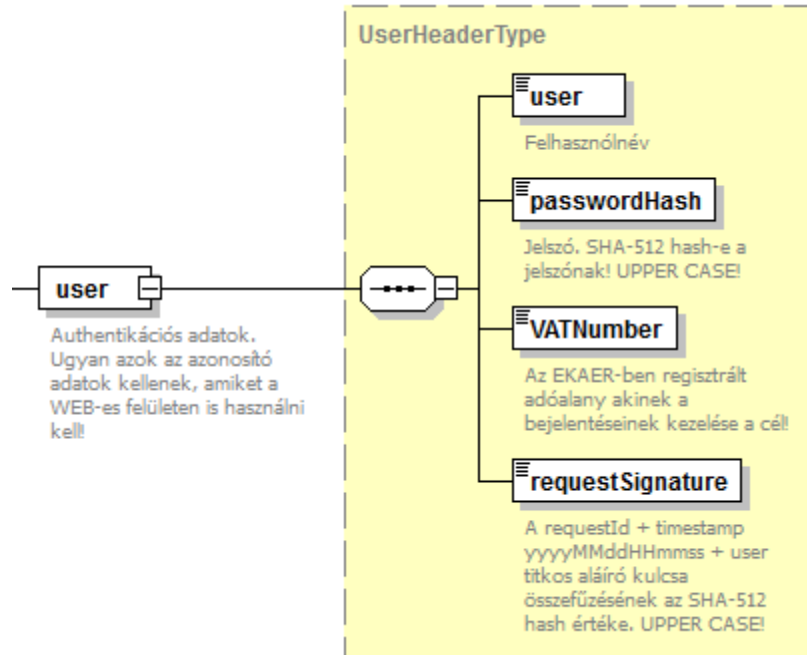
2.2.2 USER XML RÉSZ

A User a beküldőt, a változtatást kérő felhasználót azonosítja. Ebben a részben vannak az azonosításhoz és az üzenet valódiságának ellenőrzéséhez szükséges adatok.

FONTOS: WEB-en keresztüli XML feltöltésnél nem az aktuálisan bejelentkezett, az XML file feltöltését végző személy nevében történnek a bejelentésmódosítások, hanem az XML-ben a User részen megadott adatok alapján beazonosított felhasználó nevében!

Mezőnév	Típus	Kötelező	Leírás	Minta
User	30 hosszú szöveg.	Igen	A módosítást kérő user bejelentkezési neve. Login név	testelek
passwordHash	128 hosszú szöveg	Igen	A módosítást kérő user jelszó SHA-512 hash értéke! NEM A KÓDOLATLAN JELSZÓ!	BA3253876AED6BC2 2D4A6FF53D8406C6 AD864195ED144AB5 C87621B6C233B548 BAEAE6956DF346EC 8C17F5EA10F35EE3 CBC514797ED7DDD 3145464E2A0BAB41 3
VATNumber	8 hosszú adószám	Igen	Annak az adóalanynak az adószáma, akinek a bejelentéseit a felhasználó kezelni akarja. A teljes adószám első 8 számjegye.	32165498
requestSignature	128 hosszú szöveg	Igen	Az üzenet aláírása, amivel ellenőrzi a szerver, hogy valóban a user küldte be az XML-t. Egy generált SHA-512 hash érték az üzenetben szereplő adatok és a user titkos (az üzenetben nem szereplő, de a rendszer számára ismert) adatainak értéke alapján.	CE3687D87EDEFD4E AE471BEF11C28562 57B2B0CE879DCCB1 A38049D1ABB335CB DA49174EA4F8C8E9 5AAA8D7683E07349 94EFA72528E2C7EF 24CC9F3B80C02F97

A felhasználónév, jelszó és adószám ugyan azok az adatok, amelyekkel a felhasználók a web-es felületen is bejelentkeznek.



2. ábra userelement felépítése

2.2.3 A REQUESTSIGNATURE GENERÁLÁSA

A requestSignature mező azt a célt szolgálja, hogy illetéktelenek ne tudjanak bejelentéseket tenni. A hash értéket a szerver oldal minden kérésnél ellenőrzi, és csak akkor hajtja végre a műveletet, ha az ténylegesen legenerálható a kapott kérés alapján. A kérések requestId-jának egyediségét a rendszer ellenőrzi (adott user egy requestId-t csak egyszer használhat), amely az aláíró hash érték alapja, így nem lehet a kérés fejlécét lemásolva újabb kérést létrehozni, mert az ellenőrző requestSignature hash értéke nem lesz megfelelő.

A mezőben átadott értékek a következő szöveges értékek összefűzéséből kapott szöveg SHA-512 hash értéke:

- requestId
- timestamp mező a következő formában (UTC-ben!): yyyyMMddHHmmss. pl.: 2014.10.05 12:58:08 formája: 20141005125808. NAGYON FONTOS, hogy az aláírás hash generálásnál a Timestamp-ben küldött idő UTC megfelelőjét kell használni!
- A user titkos aláíró kulcsa. Ezt a jelszószerű adatot a WEB-en minden felhasználó magának tudja beállítani. Legalább 8 hosszú titkos jelszó, aminek tartalmaznia kell kis és nagybetűt, valamint számot! pl.: titkos7Password98. Akinek nincs beállítva az aláíró kulcsa, az nem tudja használni az XML-es interfészeket.

Példa:

A példában használt testelek user titkos aláíró kulcsa (amit ő maga állított be a WEB-es felületen):
Elek65Titkos

A példa request adatai:

- requestId = TSTKFT1222564
- timestamp = 2015.01.15T13:25:45+01 ebből a hash-hez használt érték: 20150115122545

XML-ben a timestamp element-ben mindegy milyen időzónában van megadva az idő, a hash gyártásnál viszont mindig ennek az időnek az UTC-ben vett megfelelőjét kell használni!

XML-ben a timestamp mező xs:dateTime típusú, aminek az egyik sajátossága, hogy ha nincs Időzóna a szöveges formában utazó időn (pl: 2015.01.15T13:25:45), akkor azt a server a saját időzónájában értelmezett helyi időnek tekinti!

Célszerű minden esetben megadni az időzónát, mert előfordulhat, hogy a server időzónája más mint a küldő rendszeré, és ebben az esetben az aláíró hash-hez használt utc idő nem fog egyezni, ebből kifolyólag az aláírást érvénytelennek tekintheti a szerver!

A szöveges érték, amelyből a hash készül, így épül fel:

TSTKFT1222564 + 20150115122545 + Elek65Titkos= TSTKFT122256420150115122545Elek65Titkos

Az így előállt („TSTKFT122256420150115122545Elek65Titkos”) szövegnek az SHA-512 hash értéke ez lenne:

AF84DC456B82234E67550C80169E517FBDAB4403607293985DECB09F534D9F73FADAABEFEE932554FA
BBC49F6E8F74A5DD54EA359D6B7644D95CFF3530AFB889

Ezen az oldalon lehet ellenőrzéseket végezni:

<http://www.convertstring.com/hu/Hash/SHA512>

2.3 MANAGETRADECARDSREQUEST, BEJELENTÉSEK KEZELÉSE (CREATE, MODIFY, DELETE)

Az üzenet általános részét (Header és User) a [2.2](#) pont részletezi.

Az XML struktúrában az üzleti adatok a tradeCardOperations listában vannak.

2.3.1 TRADECARDOPERATIONS

A tradeCardOperations elem tradeCardOperation listát tartalmaz, amelyben az elvégzendő műveletek vannak. Bejelentések rögzítése, meglévő bejelentések módosítása, törlése. Az elvégzendő műveletet a tradeCardOperation elem írja le.

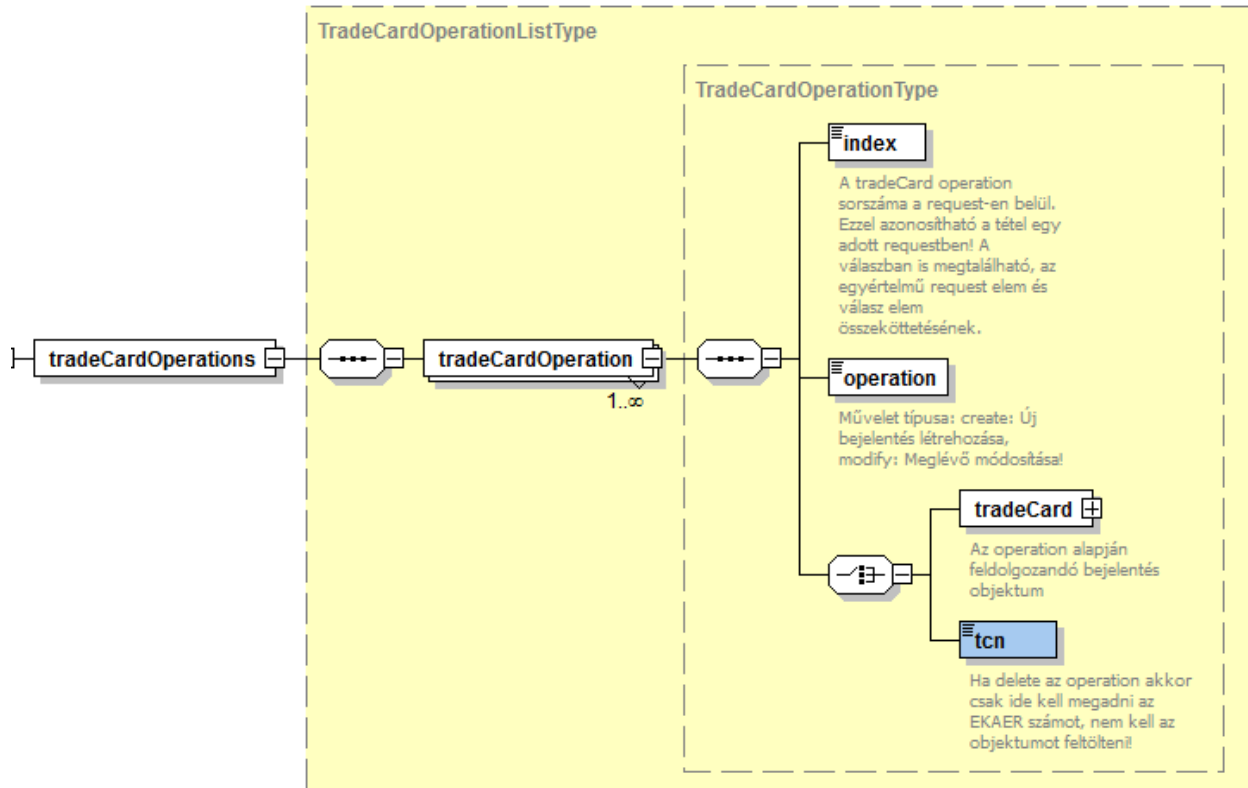
A tradeCardOperation felépítése:

Mezőnév	Típus	Kötelező	Leírás	Minta
index	xsd integer	Igen	A listában való elhelyezkedés szerinti sorszám. A kéresem belül az egyes módosítási műveletet azonosítja	1
operation	enumerált: create, modify,	Igen	Az módosítás módját jelöli. Az adott módosítási feladat	create

	delete, finalize		típusát.	
tradeCard / tcn	választó: vagy tradeCard element vagy tcn element	Igen	operation=create és operation=modify esetén tradeCard element kell! operation=delete esetén tcn	

Az operation mezőtől függ, hogy tradeCard vagy tcn element van az adott tradeCardOperation-ben.

Operation alapján dől el hogy milyen műveletet hajt végre a szerver!



3. ábra tradeCardOperation element felépítése

2.3.1.1 CREATE OPERATION, BEJELENTÉS RÖGZÍTÉSE

“Create” operation esetén a tradeCard element-et kell tartalmaznia tradeCardOperation –nek.

A tradeCard element-ben a bejelentés adatai szerepelnek, melynek alapján a szerver létrehozza a bejelentést.

Létrehozás esetén a tradeCard –on belül a tcn element-et el kell hagyni.

A tradeCardItem element-en belül az id attribute-umot el kell hagyni.

2.3.1.2 MODIFY OPERATION, BEJELENTÉS MÓDOSÍTÁSA

“modify” operation esetén a tradeCard element-et kell tartalmaznia tradeCardOperation –nek.

A tradeCard element-ben a bejelentés adatai szerepelnek, ami alapján a bejelentést módosítja a szerver.

A módosítás logikája:

A bejelentés fej részében található adatok mentésre kerülnek.

A tétel adatok feldolgozásának módja a következő:

- A tétel id attribútuma alapján a szerver kikeresi a tételt és módosítja a kapott adatok alapján. Ha nem találja meg, akkor az egész bejelentés módosítása sikertelen lesz, nem hajtja végre.
- Ha a kérésben nem szerepel egy létező tétel, akkor az adott tételt törli a szerver oldal. Tehát a módosítási kérésben a bejelentéshez tartozó tételek közül törlik a nem szereplő tételeket!
- Ha a kérésben id nélkül érkezik egy tétel, akkor azt új tételként értelmezi a server oldal, és felveszi a bejelentéshez!

2.3.1.3 DELETE OPERATION, BEJELENTÉS TÖRLÉSE

Delete esetén csak a tcn (EKAER szám) számot kell megadni és nem kell a tradeCARD teljes objektumot felépíteni. A szerver a tcn-ben átadott EKAER szám alapján megkeresi a bejelentést és törli. A törlés csak akkor hajtható végre, ha még "aktív" a bejelentés.

2.3.1.4 FINALIZE OPERATION, BEJELENTÉS VÉGLEGESÍTÉSE

Finalize esetén csak a tcn (EKAER szám) számot kell megadni és nem kell a teljes tradeCard objektumot felépíteni. A szerver a tcn-ben átadott EKAER szám alapján megkeresi a bejelentést és véglegesíti azt. A véglegesítésnél ellenőrzéseket is végez. Ezekről bővebben a következő fejezetben olvashatunk: [A bejelentésben megadott adatok ellenőrzése](#)

FONTOS: Mielőtt a Finalize-zal véglegesítjük a bejelentést, előtte a Modify operation-nel minden szükséges értéket be kell állítani, mert a véglegesítés után a rendszer nem engedi az adatok módosítását! Például a lerakodás időpontja adat megadását a véglegesítés előtt szükséges lehet módosítani.

2.3.2 TRADECARD ELEMENT FELÉPÍTÉSE

A tradeCard element-ben a bejelentéssel kapcsolatos adatok vannak tárolva. Két fő részre oszthatjuk: fej rész és item lista. A fejrészben a bejelentéssel kapcsolatos adatok vannak, míg az item listában a bejelentéshez tartozó termékenkénti tételes adatok.

2.3.2.1 TRADECARD ADATOK

A tradeCard-ben szereplő adatok írják le a bejelentés részleteit.

Mező név	Típus	Kötelező	Leírás	Minta
tcn	20 hosszú szöveg	modify operation esetén kötelező, egy	A bejelentés EKAER száma. Ez azonosítja a bejelentést.	12312312331

		ébként elhagyható		
orderNumber	50 hosszú szöveg	nem	A bejelentő saját rendszerében azonosítja a bejelentést/megrendelést	ASDF234fFfas3
tradeType	Enumerált: E, I, D	igen	Ez határozza meg, hogy az árumozgás milyen viszonylatban történik. Közösségből belföldre (I), Belföldről közösségbe (E), Belföldről belföldre (D)	I
modByCarrier Enabled	boolean	igen	A szállító módosíthatja-e a bejelentést vagy sem. Igen esetén módosíthatja, nem esetén nem.	true
carrier	30hosszú szöveg	nem	Nem kötelező megadni. A szállítmányozó EKAER-ben lévő azonosítója! (Regisztrál szállítmányozó)	
carrierText	200 hosszú szöveg	nem	Szállítmányozó szöveges megnevezése, neve	Trans2015 Kft.
isIntermodal	Logikai. xs:boolean	nem	Intermodális szállítmány esetén ezt igen-re kell állítani. Ha ez az érték igaz, akkor a felrakodás és lerakodás országa nincs validálva! 1.6-os interface verziótól létezik!	true
sellerName	200 hosszú szöveg	nem (tradeType E és D esetén igen)	A feladó/eladó cég neve, akitől az árumozgás indul.	„Első Kereskedő Kft.”
sellerVatNumber	15 hosszú szöveg	nem (tradeType E és D esetén igen)	Magyar feladó esetén magyar adószám első 8 számjegye. Külföldi esetén a közösségi adószám.	32165478
sellerCountry	2 hosszú szöveg	nem (tradeType E és D esetén)	A feladó/eladó országkódja	HU

		igen)		
sellerAddress	200 hosszú szöveg	nem (tradeType E és D esetén igen)	A feladó/eladó címe	Budapest Kisdobos tér 2.
destinationName	200 hosszú szöveg	nem (tradeType I esetén igen)	A átvevő/vevő cég neve, akitől az árumozgás indul.	„Első Kereskedő Kft.”
destinationVatNumber	15 hosszú szöveg	nem (tradeType I és D esetén igen)	átvevő/vevő adószáma. Magyar cél esetén magyar adószám első 8 számjegye. Külföldi esetén a közösségi adószám.	32165478
destinationCountry	2 hosszú szöveg	nem (tradeType I és D esetén igen)	A átvevő/vevő országkódja	HU
destinationAddress	200 hosszú szöveg	nem (tradeType I esetén igen)	A átvevő/vevő címe	Budapest Kisdobos tér 1.
unloadReporter	Enumerált: S, D	nem, default az S	CSAK Belföldi fuvar esetén (tradeType=D) van figyelembe véve! Azt jelöli, hogy ki jelentheti le a lerakodást. S: csak a bejelentő. D: Bejelentő és a címzett is! D esetén a destinationVatNumber alapján a címzettnek létező regisztrációjának kell lennie az EKAER-ben.	S
loadLocation	element	nem (TradeType E és D esetén igen)	A felrakodás címe	Budapest Ipartelep u 1.
saveLoadLocation	xs:boolean	nem default: false	Igen esetén a felrakodási címet elmenti a kedvenc címekhez, ha még nem létezik!	true
unloadLocation	element	nem	A lerakodás címe	Budapest Közraktár

n		(TradeType I és D esetén igen)		utca 1.
saveUnloadLocation	xs:boolean	nem, default: false	Igen esetén a felrakodási címet elmenti a kedvenc címekhez, ha még nem létezik	false
vehicle/plateNumber	element (jármű adatok) rendszám	nem (de a bejelentés véglegesítése előtt ki kell tölteni)	A vontató jármű rendszáma	ABC321
vehicle/country	3 hosszú szöveg	nem	A rendszámhoz tartozó felségjel. A-Z –ig elfogadott!	H
vehicle2/plateNumber	element (jármű adatok)	nem	Az első vontatmány	FFF397
vehicle2/country	3 hosszú szöveg	nem	A rendszámhoz tartozó felségjel. A-Z –ig elfogadott!	H
loadDate	xsd dateTime	nem	Felrakodás ideje	2014-12-04T08:45:00+01
arrivalDate	xsd dateTime	nem (A bejelentés véglegesítése előtt ki kell tölteni)	Lerakodás időpontja	2014-12-05T21:15:00+01
items	Element lista (tradeCardItem)	igen	A bejelentés tételei. Legalább egy elemű lista.	

2.3.2.2 A BEJELENTÉSBEN MEGADOTT ADATOK ELLENŐRZÉSE

Legalább egy tételnek (items) kell lennie az items listán!

tradeType = I esetén (közösségből belföldre irány): A seller* mezők (feladó/eladó adatai) kitöltése opcionális, a destination mezők (átvevő/vevő adatai) kötelezőek (destinationCountry=HU), és a destinationVatNumber kötelezően magyar adószámot (8 hosszú), vagy magyar adóazonosítót (10 hosszú) kell, hogy tartalmazzon.

tradeType = E esetén (belföldről közösségbe irány): A destination* mezők (átvevő/vevő adatai) kitöltése opcionális, a seller* mezők (feladó/eladó adatai) kötelezőek (sellerCountry=HU), és a sellerVatNumber kötelezően magyar adószámot (8 hosszú), vagy magyar adóazonosítót (10 hosszú) kell, hogy tartalmazzon.

tradeType = D esetén (belföld -> belföld irány): A destination* és seller* mezők is kötelezőek és magyarországinak kell lenniük. Magyar adószámnak vagy adóazonosítónak kell szerepelnie az adószám mezőkben!

A bejelentés véglegesítése/lezárása előtt a következő adatokat meg kell adni:

- a vehicle element-nek valós jármű adatokat kell tartalmaznia.
- Az arrivalDate-nek a lerakodás idejét tartalmaznia kell.

A címadatok ellenőrzéséről a [2.3.2.4](#) pontban olvashatunk.

2.3.2.3 A LERAKODÁSI ÉS FELRAKODÁSI CÍMADAT ELEM FELÉPÍTÉSE, MEZŐI

Mezőnév	Típus	Kötelező	Leírás	Minta
name	150 hosszú szöveg	nem	A címhez tartozó cég neve. Raktár üzemeltetője, tulajdonosa.	Raktározó kft.
VATNumber	15 hosszú szöveg	nem, (country=HU esetén kötelező!)	Magyar cég esetén magyar adószám első 8 számjegye. Külföldi esetén közösségi adószám.	24653422
phone	15 hosszú szöveg	nem	A raktár, telephely telefonos elérhetősége. 00-val vagy + jellel vagy 06 -al kezdődhet. A 00 vagy + jel után legalább 8 maximum 14 szám karakter következhet. 06 után 1-2 szám karakteren a körzetszám következhet, ami után 6-7 szám karakteren a telefonszámnak kell következnie!	+36221321654
email	128 hosszú szöveg	nem	A raktár, telephely elektronikus elérhetősége	info@raktar.hu
country	2 hosszú szöveg	nem	Országkód	HU

zipCode	7 hosszú szöveg	nem	Irányítószám	1111
city	50 hosszú szöveg	nem	Város	Budapest
street	150 hosszú szöveg	nem	Közterület neve	Fő
streetType	50 hosszú szöveg	nem (street mezőben is átadható)	Közterület jellege	utca
streetNumber	10 hosszú szöveg	Nem	házsám	1
lotNumber	15 hosszú szöveg	Nem	helyrajzi szám. Ha nem ismert a házsám, vagy nincs kiosztva...stb	11231/A.

A felrakodási és lerakodási címeknél ha nem ismert a házsám, vagy nincs, akkor a helyrajzi számot kell megadni a lotNumber mezőben.

2.3.2.4 A CÍMADATOK ELLENŐRZÉSE

- **tradeType=I esetén (közösségből belföldre irány):** A felrakodási cím opcionális, a lerakodási cím kötelező és kötelezően magyar címnek kell lennie, a lerakodási címnél megadott adószámnak (8 hosszú) létező magyar adószámnak kell lennie.
- **tradeType=E esetén (belföldről közösségbe irány):** A felrakodási cím kötelező és a címnél megadott adószámnak létező magyar adószámnak (8 hosszú) kell lennie, magyar címmel. A lerakodási cím opcionális.
- **tradeType=D esetén (belföld -> belföld irány):** A felrakodási és lerakodási cím is kötelező. A címekben megadott címeknek magyarnak kell lenniük és az adószámoknak (8 hosszú) is létező magyar adószámnak kell lenniük.

Közösségből belföldre való fuvarozás esetén, ha a bejelentésben olyan minősített termék szerepel, amiket csak NÉBIH által kiadott FELÍR számmal rendelkező cégek hozhatnak be az országba, akkor a lerakodási hely csak olyan cím lehet, ami szerepel a NÉBIH által nyilvántartott címtárba!

A címadatoknál xsd szinten minden opcionális, viszont az üzleti adatok logikája szerint validálva vannak! A megadandó címen belül a következő szabály érvényes:

name: kötelező

VATNumber: kötelező (tipikusan magyar címek megadása a kötelező)

country: kötelező

zipCode: kötelező

city: kötelező

street: kötelező ha nincs lotNumber megadva

streetType: nem kötelező

streetNumber: kötelező ha nincs lotNumber megadva

lotNumber: opcionális, de ha nincs megadva akkor az steet, és streetNumber kötelező.

A nem kötelező címadatok nincsenek validálva!

2.3.2.5 ORSZÁGOK LISTÁJA

A címadatoknál és országot jelölő mezőknél csak a következő országekódot szerepelhetnek!

- AT Ausztria
- BE Belgium
- BG Bulgária
- CY Ciprus
- CZ Cseh Köztársaság
- DK Dánia
- GB Egyesült Királyság
- EE Észtország
- FI Finnország
- FR Franciaország
- GR Görögország
- NL Hollandia
- HR Horvátország
- IE Írország
- PL Lengyelország
- LV Lettország
- LT Litvánia
- LU Luxemburg
- HU Magyarország
- MT Málta
- DE Németország
- IT Olaszország
- PT Portugália
- RO Románia
- ES Spanyolország
- SE Svédország
- SK Szlovákia
- SI Szlovénia

2.3.2.6 ITEMS LISTA FELÉPÍTÉSE (TRADECARDITEM)

Az items listában tradeCardItem-ek vannak, amelyek a bejelentéssel kapcsolatos tételeket írják le! A tétel tartalmazza a fuvarral kapcsolatos terméke(ke)t, azok súlyát értékét és egyéb információit.

Egy tétellel a következő adatok vannak kapcsolatban:

Mezőnév	Típus	Kötelező	Leírás	Minta
id	Attribútum, 30 hosszú azonosító	nem	A bejelentés rögzítésénél a szerver generálja. Create operation-nél nem kell kitölteni. Módosításnál kötelező. Ez alapján azonosítja a szerver, melyik tételt módosítsa.	12ASDF356DFG
itemExternalld	50 hosszú szöveg.	nem	1.6-os requestVersion-től! Bejelentő tetszőleges azonosítóval, sorszámmal láthatja el a tétel, amivel a saját rendszerében meg tudja feleltetni azt!	1
tradeReason	Enumerált: S: Értékesítés/B eszerzés W: Bérmunka O: Egyéb	igen	A tétel fuvarozásának oka. Az A: Saját tulajdonú termék kivezetésre került, de a korábbi bejelentések miatt az interface-be benne marad. Új bejelentésnél már nem adható meg!	S
productVtsz	4-8 hosszú szöveg. Csak számot tartalmazhat	igen	A tételhez tartozó termék VTSZ száma.	03034921
productName	200 hosszú szöveg	igen	A termék szöveges neve, amit a bejelentő használ rá. (nem a VTSZ szám megfelelője)	Kékúszójú tonhal filé
adrNumber	Max 200 hosszú szöveg	Csak veszélyes áru esetén kötelező (pl.: üzemanyag)	UN (veszélyes áru) kód, veszélyes áru szállításnál a besorolási érték. Ha többet szállít, akkor vesszővel elválasztva fel lehet sorolni! <u>UN prefix nélkül!</u>	0336,1263
transportLicense	30 hosszú szöveg	Nem	Veszélyes áru szállítása esetén az engedély száma. Katasztrófavédelem állítja ki.	

weight	xs:decimal	igen	Tétel tömege: bruttó súlykg-ban. Max 9 jegyű egész szám!	425
value	xs:decimal	igen	A tétel beszerzési értéke HUF-ban. Ha devizában van a pénzügyi teljesítés, akkor az aktuálisan ismert árfolyamon számolva. Max 9 jegyű egész szám!	12500000 *
factoryItemNumber	200 hosszú szöveg	nem	Gyári szám, ha a tétel mögött egy konkrét termék érkezik csak.	7622210240200
importerItemNumber	200 hosszú szöveg	nem	A tétel bejelentő által használt cikkszáma. Ha a tétel mögött egy konkrét termék van.	TS7622/11
expirationDate	xs:date	Nem	Ha a tétel élelmiszer, akkor a lejárati dátuma.	2015-07-20
batchNumber	30 hosszú	Nem	Sarzsorszám. Gyártási azonosító.	234

- Value meghatározása:
Amennyiben a termék közúti fuvarozásának indoka termékbeszerzés vagy termékértékesítés, az egyes terméknevezésekhez (tételhez) tartozó adó nélküli ellenérték, egyéb célú közúti fuvarozás esetén az egyes terméknevezésekhez (tételhez) tartozó adó nélküli beszerzési ár, vagy hasonló termék adó nélküli beszerzési ára, ilyen ár hiányában pedig az adó nélküli előállítási érték.

2.3.2.7 TÉTELEKKEL KAPCSOLATOS ELLENŐRZÉSEK

A tételek rögzítésénél a rendszer a következők alapján végez ellenőrzéseket:

- VTSZ szám ellenőrzése (létezik-e).
- VTSZ alapján kockázatos-e a termék, ha igen akkor biztosítékot számol utána. A tétel rögzítése csak akkor lehetséges, ha van elegendő biztosítékfedezet.
- VTSZ alapján FELIR szám köteles-e. Megtörténik a bejelentő FELIR szám ellenőrzése (NÉBIH adatok alapján). Ezeket a termékeket csak a NEBIH által kezelt címlistában (Első beraktározási hely) szereplő helyeken lehet lerakodni!
- VTSZ alapján veszélyes-e az áru.
- A veszélyes és kockázatos áruk VTSZ számát 8 hosszán kell megadni.
- Első beraktározási hely ellenőrzése
- Élelmiszer-e az adott termék.

Tételszinten az Aktív bejelentések esetén, csak a következő mezőket lehet módosítani:

- Mennyiség, súly (Kg)
- Érték (HUF)
- Rendszám, jármű adatok (Rendszám, felségjel)
- Lerakodási hely adatok (Lerakodási hely címadata)

2.3.2.8 FUVAR OKA (TRADEREASON)

A bejelentés tételeinél meg kell adni, hogy az adott tétel rögzítésének mi az oka. Ez befolyásolja a biztosítékszámítást is! A tételeknél a tradeReason mezőben kell megadni, hogy az adott tétel fuvarozásának mi az indoka.

S: Termékértékesítés. Van biztosítékszámítás!

A: Saját tulajdonú termék! Van biztosíték számítás! (Kivezetésre került 2015.03.01 után nem használható!)

W: Bér munka. Nincs biztosítékszámítás!

O: Egyéb cél. Nincs biztosítékszámítás!

A fuvar okokat a bejelentés fej részben, a tradeType-ban megadott fuvar viszonylatnak megfelelően lehet csak beállítani!

Belföldről közösségbe irány (E):

- Bér munka (W)
- Termék értékesítés (S)
- Egyéb (O)

Közösségből belföldre irány (I):

- Termék beszerzés (S):
- Bér munka (W)
- Egyéb (O)

Belföld-belföld irány (D):

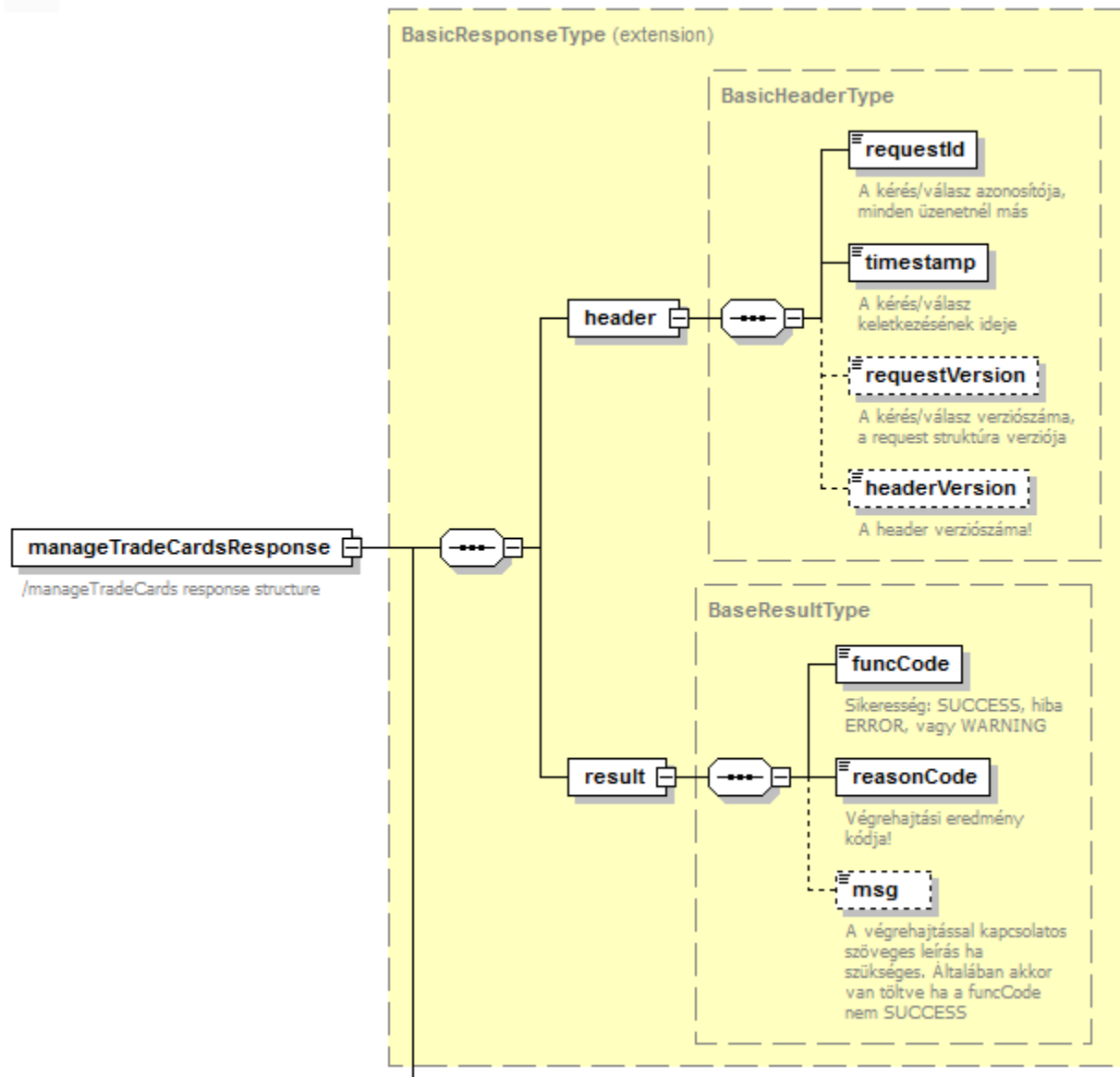
- Termék értékesítés (S):
- Bér munka (W)
- Egyéb (O)

2.4 MANAGETRADECARDSRESPONSE, A VÁLASZ FELÉPÍTÉSE

A kérésként beküldött manageTradeCardsRequest XML-re a rendszer egy válasz XML-t szolgáltat, melyet a manageTradeCardsResponse element ír le az XSD-ben. Ebben a válasz XML-ben van a feldolgozás eredménye.

A válasz XML-ek ugyanolyan header és user fejléce van, mint a kérésnek.

Az üzleti válasz a tradeCardOperationsResults element-en belül van, amely egy operationResult lista. A lista annyi elemű amennyi a kérésben volt. Minden, a kérésben érkezett művelethez ez a lista adja vissza az eredményt.



4. ábra manageTradeCardsResponse element felépítése

2.4.1 OPERATIONRESULT FELÉPÍTÉSE

A válaszban visszakapott listában operationResult element-ek vannak. Egy element egy a kérésben kapott művelet eredményét tartalmazza.

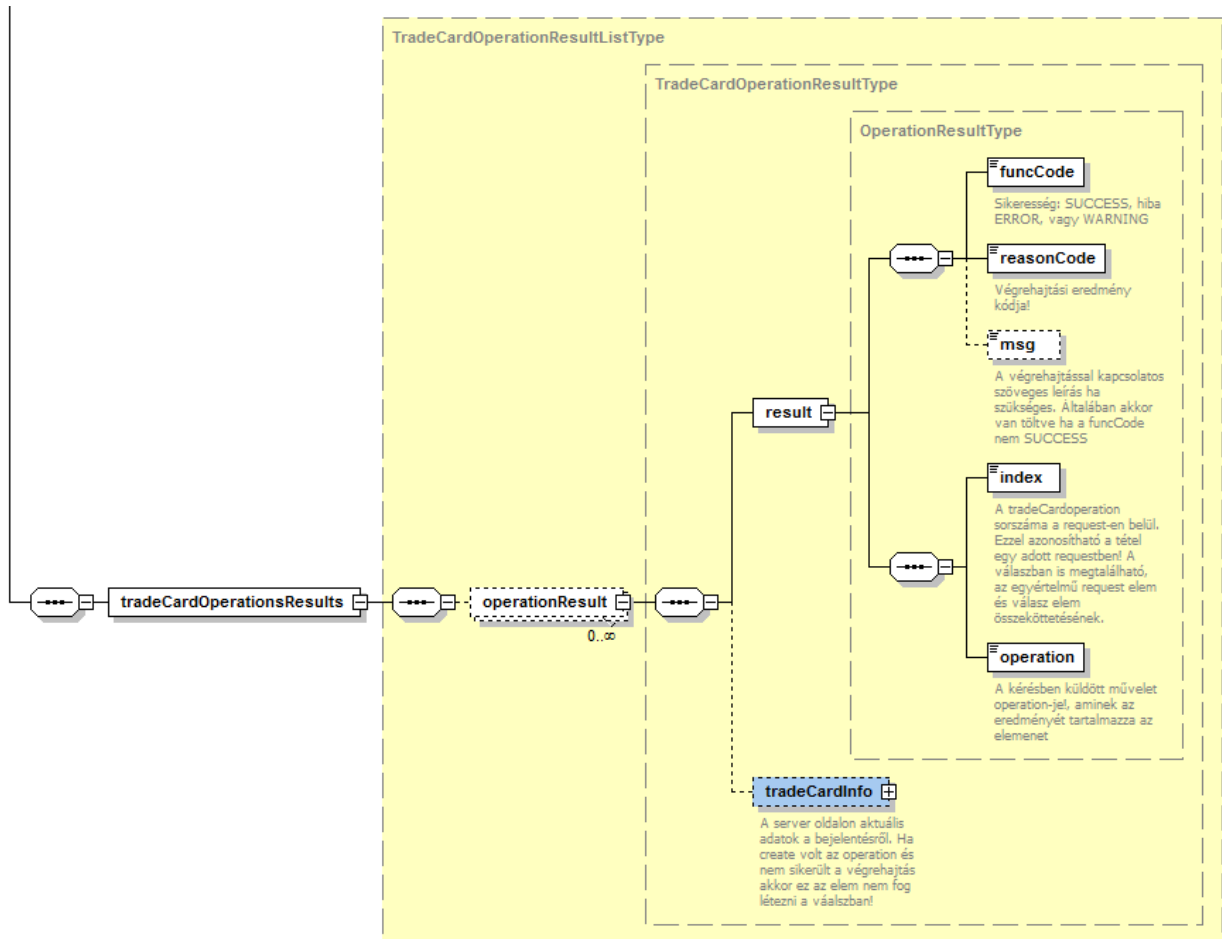
Mezők:

Mezőnév	Típus	Kötelező	Leírás	Minta
---------	-------	----------	--------	-------

result	OperationResultType xsd típus	igen	A művelet eredményét tartalmazza.	
tradeCardInfo	TradeCardBasicInfoType	igen	A bejelentés alap adatai, amivel kapcsolatban a művelet végre lett hajtva.	

A result tartalmazza a művelettel kapcsolatos adatokat és eredményességet.

A tradeCardInfo tartalmazza az információt a bejelentéssel kapcsolatban.



5. ábra tradeCardOperationsResults felépítése

2.4.1.1 RESULT FELÉPÍTÉSE (OPERATIONRESULTTYPE)

A result element mezői:

Mezőnév	Típus	Kötelező	Leírás	Minta
funcCode	Enumerált, OK, WARNING, ERROR	igen	A művelet sikerét jelöli. OK: Minden sikeres, WARNING: részben sikeres (jellemzően ez nem lesz használatban) ERROR: Hiba történt, sikertelen a művelet végrehajtása	
reasonCode	Enumerált típus	igen	A végrehajtás eredményének pontos hibakódja. SUCCESS a siker. A többi hibára utal. Pl.:	
msg	200 hosszú szöveg!	nem	Hiba esetén msg-ben van a hiba pontosabb szöveges leírása.	
index	xs:integer xsd egész szám típus	igen	A művelet sorszáma (a kérdésben), aminek az eredményét tartalmazza az operationResult	
operation	enumerált:create, modify, delete, finazlie	igen	Az módosítás módját jelöli. Az adott módosítási feladat típusát.	create

Az index és az operation a kérdésben kapott műveletből vannak kimásolva. Ez alapján látszik, hogy a kérdésben melyik művelethez tartozik az adott válasz.

A végrehajtás eredménye a funcCode és reasonCode –ból derül ki, míg ha volt hiba, a szöveges leírását az msg mező tartalmazza.

2.4.1.2 TRADECARDINFO ELEMENT FELÉPÍTÉSE

A válasz XML-ben ez az element tartalmazza az üzleti adatokat a bejelentéssel kapcsolatban (a művelet végrehajtása utáni aktuális állapotáról). Ennek nagy része a kérdésben is érkezett.

Mező név	Típus	Kötelező	Leírás	Minta
tcn	20 hosszú szöveg	Igen	A bejelentés EKAER száma. Ez azonosítja a bejelentést.	12312312331
orderNumber	50 hosszú szöveg	Nem	A bejelentő saját rendszerében azonosítja a bejelentést/megrendelést	ASDF234fFfas3
tradeType	Enumerált: E, I, D	Igen	Ez határozza meg, hogy az árumozgás milyen viszonylatban történik.	I

			Közösségből belföldre (I), Belföldről közösségbe (E), Belföldről belföldre (D)	
modByCarrierEnabled	boolean	igen	A szállító módosíthatja-e a bejelentést vagy sem. Igen esetén módosíthatja, nem esetén nem.	true
carrier	30	nem	Nem kötelező megadni. Ha megadjuk, akkor létező szállító azonosítój.	
carrierText	200 hosszú szöveg	nem	Nem kötelező megadni, szállító szöveges megnevezése!	Pelda Trans Kft.
sellerName	200 hosszú szöveg	nem (tradeType E és D esetén kötelező)	A feladó/eladó cég neve, akitől az árumozgás indul.	„Első Kereskedő Kft.”
sellerVatNumber	15 hosszú szöveg	nem (tradeType E és D esetén kötelező)	Magyar feladó esetén magyar adószám első 8 számjegye. Külföldi esetén a közösségi adószám.	32165478
sellerCountry	2 hosszú szöveg	igen	A feladó/eladó országkódja	HU
sellerAddress	200 hosszú szöveg	nem (tradeType E és D esetén kötelező)	A feladó/eladó címe	Budapest Kisdobos tér 2.
destinationName	200 hosszú szöveg	nem (tradeType I esetén kötelező)	A átvevő/vevő cég neve, akitől az árumozgás indul.	„Első Kereskedő Kft.”
destinationVatNumber	15 hosszú szöveg	nem (tradeType I esetén igen)	átvevő/vevő adószáma. Magyar cél esetén magyar adószám első 8 számjegye! Külföldi esetén a közösségi adószám	32165478
destinationCountry	2 hosszú szöveg	nem (tradeType I esetén igen)	A átvevő/vevő országkódja	HU

destinationAddress	200 hosszú szöveg	nem (tradeType I esetén igen)	A átvevő/vevő címe	Budapest Kisdobos tér 1.
loadLocation	element	nem (TradeType E és D esetén igen)	A felrakodás címe	Budapest Ipartelep u 1.
unloadLocation	element	nem (TradeType I és D esetén igen)	A lerakodás címe	Budapest Közraktár utca 1.
vehicle/plateNumber	element (jármű adatok) rendszám	nem? (A bejelentés véglegesítése előtt ki kell tölteni)	A vonó jármű rendszáma	ABC321
vehicle/country	3 hosszú szöveg	nem	A rendszámhoz tartozó felségjel. A-Z –ig elfogadott.	H
vehicle2/plateNumber	element (jármű adatok)	nem	Az első vontatmány	FFF397
vehicle2/country	3 hosszú szöveg	nem	A rendszámhoz tartozó felségjel. A-Z –ig elfogadott.	H
loadDate	xsd dateTime	nem	Felrakodás ideje	2014-12-04T08:45:00+01
arrivalDate	xsd dateTime	nem (A bejelentés véglegesítése előtt ki kell tölteni)	Lerakodás időpontja	2014-12-05T21:15:00+01
items	element	igen	A bejelentés tételei. Legalább egy elemű lista.	2.3.2.5 fejezet írja le a felépítését
VATNumber	8 hosszú szöveg	Nem, csak ha a bejelentőnek van adószáma	A bejelentést tevő adószáma. Szerver oldal automatikusan kezeli, tölti!	32165498

taxIdentifier	10 hosszú szöveg	Nem, csak ha a bejelentőnek van adóazonosítója	A bejelentést tevő adóazonosítója. Szerver oldal automatikusan kezeli, tölti.	321654879
status	Enumerált: P: tervezés alatt S: Aktív EKAER, számot kapott F: Véglegesített, befejezett I: Inaktív D: Törölt	igen	A bejelentés aktuális státusza. XML alapú bejelentésnél egyből S státuszba kerül. Csak WEB-en való rögzítésnél jön létre P státuszban a bejelentés	S
totalWeight	xs:decimal	igen	A bejelentésben rögzített tételek össz súlya kg-ban.	1500
totalValue	xs:decimal	igen	A bejelentésben rögzített tételek összértéke HUF-ban.	1250000
totalAssuranceLocked	xs:decimal	igen	A bejelentéshez tartozó biztosítékfoglalás mértéke HUF-ban.	187500
finalizationTime	xs:dateTime	nem	Véglegesítés bejelentésének időpontja. Lerakódás után.	2015-01-15T17:35:00+01
insDate	xs:dateTime	igen	A bejelentés rögzítésének időpontja.	2015-01-14T10:25:15+01
tcnValidityStart	xs:date	nem	EKAER számmal rendelkező bejelentéseknél az EKAER szám érvényesség kezdete	2015-01-14+01
tcnValidityEnd	xs:date	nem	EKAER számmal rendelkező bejelentéseknél az EKAER szám érvényesség vége. (Kezdete + 15 nap)	2015-01-30+01

2.4.1.3 BEJELENTÉS STÁTUSZAI (STATUS)

A bejelentéseknek van egy technikai életciklusuk, amit a status mező kezel. Adott, hogy melyik státuszról melyikbe lehet lépni, illetve státuszváltásnál milyen megfeleléségi vizsgálatokat végez a

rendszer. Ha a megfelelőségi vizsgálat során hiányosságok vannak, akkor a státuszmódosítás nem lehetséges.

A státuszok kódjai:

- **P:** Tervezés alatt. Ebbe a státuszba csak WEB-es felületen való létrehozáskor kerül a bejelentés! Addig marad ebben a státuszban, amíg a felhasználó nem kér EKAER számot a bejelentéshez, ezzel jelezve, hogy vége a tervezésnek!
- **S:** Aktív, EKAER számmal rendelkező bejelentés. A lerakódás bejelentése még nem történt meg, vagy még a 15 napon belül van. A biztosítékszámítás megtörtént. Az XML kommunikációs interfészen keresztül létrehozott bejelentések automatikusan ebben a státuszban jönnek létre, tehát egyből automatikusan EKAER számot kapnak, és a biztosítékkalkuláció is megtörténik.
- **F:** Véglegesített bejelentés, aminek vagy lejárt a 15 napos élelciklusa, vagy megtörtént a lerakódás tényének és idejének bejelentése.
- **I:** Inaktív bejelentés. Egy bejelentés törlés hatására kerülhet S (Aktív) státuszba. Ilyenkor inaktívulódik a bejelentés, a biztosítékszámítás lefut és ennek a hatására felszabadul a bejelentés által lefoglalt keret!
- **D:** Törölt bejelentés. Egy bejelentés törlés hatására kerülhet P (Tervezés alatt) státuszba. P státuszba csak WEB-es felületen történt rögzítés hatására kerülhet.

2.5 BIZTOSÍTÉK SZÁMÍTÁS FOLYAMATA, LÉPÉSEI

A rendszer a biztosítékokat 60 napos „csúszó” ablakban kezeli. A bejelentések mögötti biztosítékokat az EKAER szám kiadásától számítva 60 napig visszamenőleg számítja a bejelentésen szereplő kockázatos termékek értéke alapján.

Biztosítékot csak a következő fuvarviszonylatokba számít a rendszer:

- Közösségből belföldre történő fuvarozás, nemzetközi
- Belföldről belföldre történő fuvarozás, hazai

Minden (a törvény által) kockázatosnak minősített termék bejelentési értéke alapján kockázati biztosítékot számol a rendszer.

A biztosíték számítása az EKAER szám kiosztásával egy időben történik meg. Ez gyakorlatban azt jelenti, hogy az XML kommunikációval létrejött új bejelentés esetén egyből megtörténik (mert S státuszba jön létre a bejelentés), WEB-en történő bejelentés szerkesztésnél az „EKAER szám kérése” funkció hatására (amikor P státuszba lép) számítja a rendszer a biztosítékot (rendelkezésre álló biztosítékszámítás és foglalás)!

Az élő, S státuszban levő bejelentések tételeinek módosítása során, ha az adott tétel értékét módosítják, a rendszer a szükséges biztosítékot automatikusan újra kalkulálja. Ha a változás értéknövekedéssel jár, és a megnövekedett érték hatására megnövekedett kockázati biztosítékra nincs elegendő biztosítékkeret, akkor a rendszer a módosítást nem engedi elvégezni. Ha a változás csökkenés, akkor a bejelentés mögötti kockázati biztosíték összege is csökken.

Tétel törlése esetén a bejelentés mögötti kockázati biztosíték összege is felszabadul.

Amikor egy bejelentés inaktív vagy törölt státuszba kerül, akkor a mögötte levő kockázatos árukkal kapcsolatos biztosítékok kikerülnek a biztosítékszámításból!

2.6 QUERYTRADECARDSREQUEST FELÉPÍTÉSE

Az ügyfél saját bejelentései lekérdezéséhez ilyen XML üzenetet kell beküldeni a server-nek! Az XML-ben azonosítva van a hívó és a megadott paraméterek alapján a szerver vissza adja a lekérdezési feltételeknek megfelelő bejelentések adatait.

Az XML-ben a header és user szekciót a [Az XML üzenetek általános felépítése](#) pontban írtak szerint kell felépíteni.

Alapvetően kétféle képpen lehet bejelentés adatot lekérdezni:

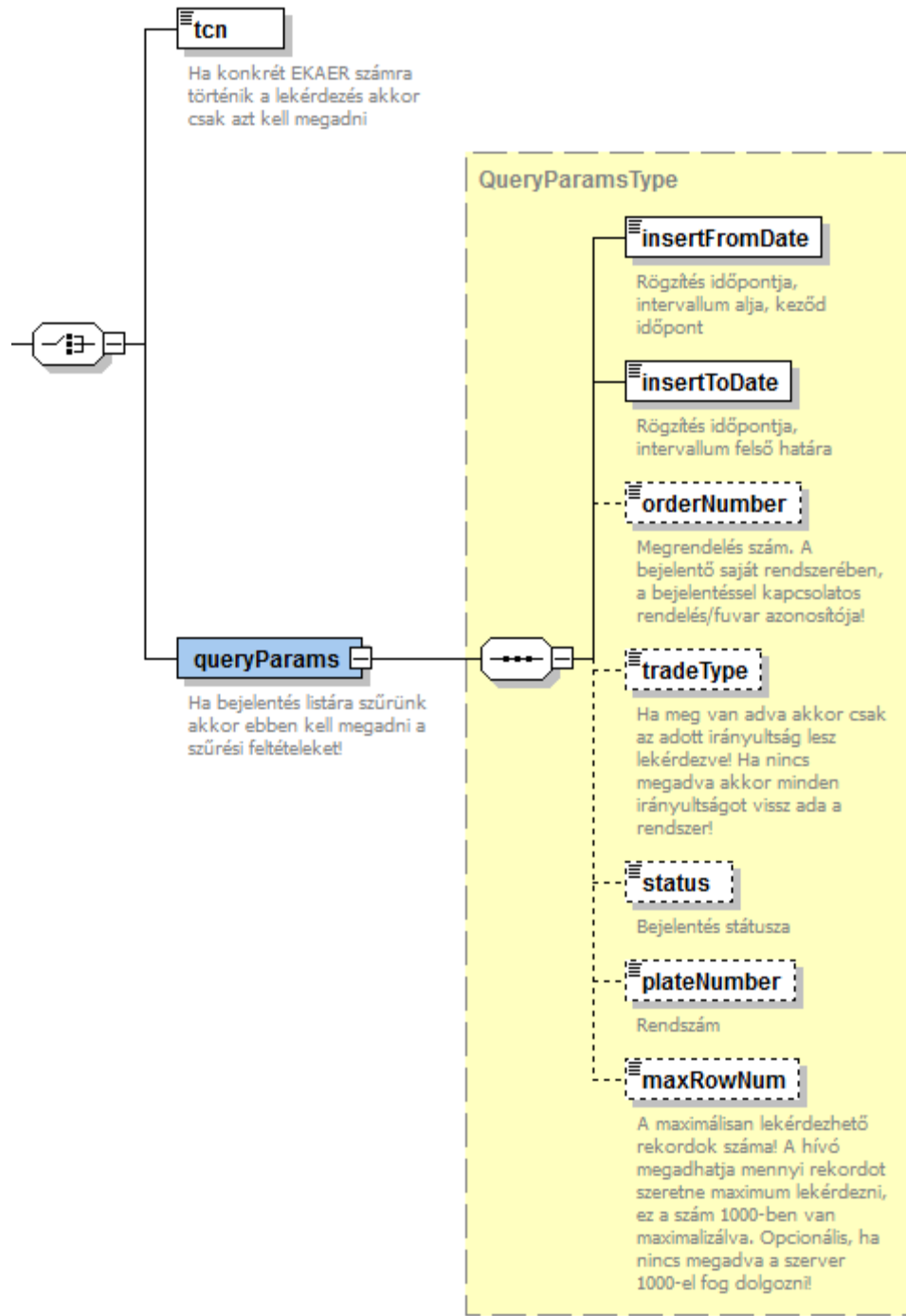
- EKAER szám alapján (tcn)
- queryParams elementben átadott lekérdezési feltételek mentén.

Az XML-ben a tcn és a queryParams choice-ban szerep, tehát vagy az egyik van megadva vagy a másik, ahogy azt a 6.ábra mutatja!

A lekérdező operation csak a következő státuszú bejelentéseket adja vissza:

- **S:** Aktív, EKAER számmal rendelkező bejelentés. A lerakodás bejelentése még nem történt meg, vagy még a 15 napon belül van. A biztosítékszámítás megtörtént. Az XML kommunikációs interfészen keresztül létrehozott bejelentések automatikusan ebben a státuszban jönnek létre, tehát egyből automatikusan EKAER számot kapnak, és a biztosítékkalkuláció is megtörténik.
- **F:** Véglegesített bejelentés, aminek vagy lejárt a 15 napos életciklusa, vagy megtörtént a lerakodás tényének és idejének bejelentése.
- **D:** Törölt bejelentés. Egy bejelentés törlés hatására kerülhet P (Tervezés alatt) státuszról ebbe a státuszba. P státuszba csak WEB-es felületen történt rögzítés hatására kerülhet.

A státuszokról a [Bejelentés státuszai \(status\)](#) pontban olvasható bővebben!



6. ábra queryTradeCardsRequest feltétel choice felépítése

2.6.1 EKAER SZÁM ALAPJÁN (TCN) TÖRTÉNŐ LEKÉRDEZÉS

Az XML-ban a tcn element-ben meg kell adni a lekérdezni kívánt bejelentés tcn számát! A queryParams element-nek nem szabad szerepelnie a kérésben! EKAER szám alapján történő lekérdezésnek maximum 1 találat van!

2.6.2 QUERYPARAMS BAN MEGADHATÓ FELTÉTELEK

Ha nem EKAER szám alapján egyetlen bejelentést kell lekérdezni, hanem intervallum (és egyéb szűrési feltételek) alapján több bejelentést, akkor azt a queryParams element-ben megadható feltételek mentén lehet megtenni.

A lekérdezés feltételeit a következő mezőkkel lehet megadni:

Mező név	Típus	Kötelező	Leírás	Minta
insertFromDate	xs:dateTime	Igen	A bejelentés rögzítésének időpontja. Amikor ez EKAER rendszerbe rögzítésre került! Intervallum alsó határa.	2015-01-04T10:25:15+01:00
insertToDate	xs:dateTime	Igen	A bejelentés rögzítésének időpontja. Amikor ez EKAER rendszerbe rögzítésre került! Intervallum felső határa.	2015-01-14T10:25:15+01:00
orderNumber	50 hosszú szöveg	Nem	A bejelentésben megadott „importóri rendelés” száma/azonosítója. Ha nincs megadva akkor nem kerül bele a szűrő feltételbe.	2015SDF234DFG
tradeType	Enumerált: E, D, I	Nem	A fuvar irányultsága! Belföldről közösségbe, hazai fuvar, közösségből belföldre. Ha nincs megadva akkor ez alapján nem szűr, minden fajtát vissza ad.	I
status	Enumerált: S, F, I	Nem	Bejelentés státusza! A lekérdezés csak S, F, I státuszú bejelentéseket adhat vissza. Ha nincs megadva akkor mindegyiket vissza adja.	S
plateNumber	4-15 hosszú rendszám	Nem	Rendszámra is lehet szűrni, ha nincs megadva nem kerül bele a szűrő feltételbe!	
maxRowNum	xs:integer 1-1000-ig egész	Nem, default	Maximum átadandó rekordok száma.	500

	szám.	1000	Opcionális, ha nincs megadva 1000-nek van értelmezve! Azt lehet állítani vele, hogy maximum hány bejelentést adjon vissza a szerver!	
--	-------	------	--	--

A lekérdezéseknél a következő szabályokat kell betartani:

- A insertFromDate és az insertToDate maximum 30 napos intervallumot ölelhet fel!
- A maxRowNum –al lehet szabályozni hogy mennyi adatot akarunk lekérdezni. 1000-ben van ez maximalizálva. Ha egy intervallumra 1000 találatot ad a szerver, akkor érdemes az intervallumon szűkíteni, hogy biztosak legyünk abban, hogy minden bejelentést megkaptunk!

2.7 QUERYTRADECARDSRESPONSE FELÉPÍTÉSE, A LEKÉRDEZÉSRE ADOTT VÁLASZ STRUKTÚRA

A bejelentések lekérdezésére queryTradeCardsResponse element-nek megfelelő választ szolgáltat a service.

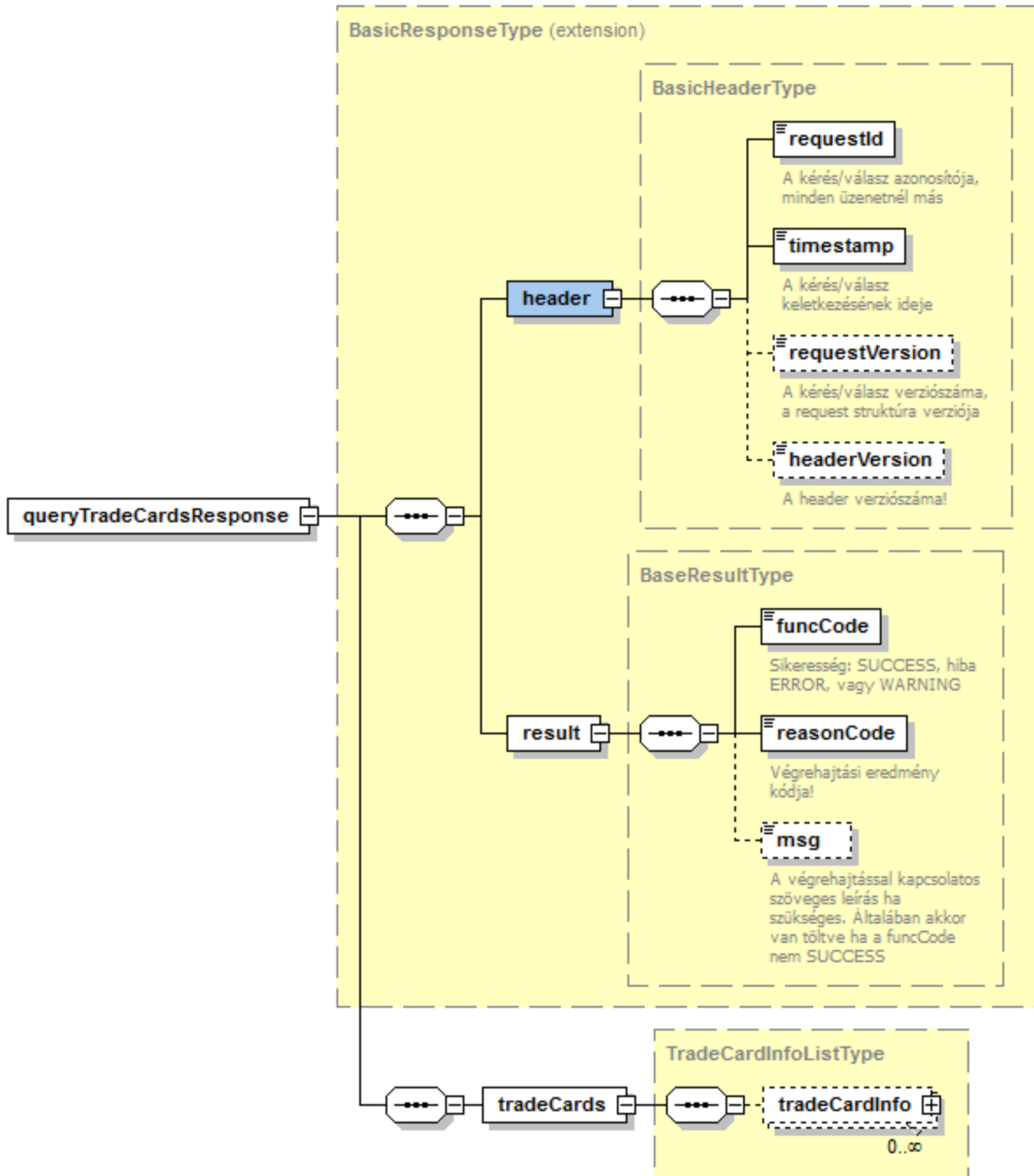
A válasz XML a bejelentések kezelésére adott válaszban megszokott módon kezdődik, header és result element-el. A header element a kérésben megadott módon szerepel, a result pedig a feldolgozás eredményét jelöli.

A result felépítését a [3.5 Result element a válaszüzenetben](#) fejezetben részletezzük!

A kérésben megadott feltételek mentén a service vissza adja a megfelelő bejelentésadatokat a válasz tradeCards element-ben.

A tradeCards element egy tradeCardInfo listát tartalmaz. A bejelentések kezelésénél is ilyen struktúrában adja vissza a szerver a bejelentések állapotát!

Bővebben tradeCardInfo struktúráról a [2.7 tradeCardInfo element felépítése](#) fejezetben olvashatunk!



7. ábra queryTradeCardsResponse felépítése

3 SZOLGÁLTATÁS TECHNIKAI LEÍRÁS

3.1 ÁLTALÁNOS TECHNIKAI ADATOK

A szolgáltatásnak http POST metódussal kell a megfelelő XML-t elküldeni, aminek hatására a válasz body-ban XML-t ad vissza. A kérésben az elvégzendő műveletet definiálja a hívó, míg a válaszban a művelet elvégzéséről ad eredményt a szerver.

Context root:

/EkaerManagementService

XSD:

hu\gov\nav\schemas\EKAER\1.0\ekaermanagement.xsd

Az xsd által leírt XML üzeneteket kell POST metódussal elküldeni a servernek.

A kommunikációhoz használt entitások element-ként vannak definiálva XSD-ben.

Az egyes elemek használata és értelmezése az XSD-ben is dokumentált.

3.2 OPERATIONS

- **/customer/manageTradeCards:** Bejelentések kezelése.
- **/customer/queryTradeCards:** Bejelentések lekérdezése.

3.3 HTTP HEADERS

A kérésben a következő http header-eket kötelező megadni:

content-type=text/xml

accept=text/xml

3.4 HTTP STATUS CODES

A következő HTTP státusz kódok fognak működni:

- **200 OK:**
A szolgáltatás az adott operation-nek megfelelő üzleti választ adja.

Az XSD-ben definiált választ adja a server. A result részben a végrehajtás eredményének megfelelő reasonCode-al!

3.5 RESULT ELEMENT A VÁLASZÜZENETBEN

A result element minden válaszüzenetben szerepel. Ez mindig az üzleti válasz egységes eredményességét tükrözi.

- **funcCode:** OK, WARNING, ERROR értékeket vehet fel. Egyszerűen azt mutatja, hogy az üzleti végrehajtás sikerült, hibára futott, vagy „warning” esetén részben sikerült (ahol ennek van létjogosultsága).

- **reasonCode:** A végrehajtás eredménykódja. **Az xsd definiálja az itt használható értékeket, enumerált típus.**
- **msg:** A reasonCode által definiált eredmény szöveges leírása. Hiba pontosabb leírása. Sikeres végrehajtás esetén nem kell kitölteni, elhagyható.

3.5.1 ReasonCode enumerált típusok

Az XSD-ben is van leírás a következő enumerált típusokhoz. A következő típusokat és resultCode-okat minden operation-nél az adott üzleti folyamatnak megfelelően kell értelmeznie. Nem fog minden reasonCode értelmet nyerni minden operation esetén, ez egy általános lista.

- **SUCCESS:** sikeres végrehajtás
- **OPERATION_FAILED:** Végrehajtás sikertelen. Általános hiba, egyéb hibakód alá nem besorolható.
- **INVALID_INPUT:** A kapott request adattartalma nem megfelelő, vagy hiányos. Üzletileg vagy egyéb adatvalidációs szabálynak nem felel meg.
- **INVALID_REQUEST:** A kapott kérés nem értelmezhető. PL.: Kapott kérés felépítése nem well formed.
- **INVALID_USER_OR_PASSWORD:** Login sikertelen. Érvénytelen felhasználónév vagy jelszó.
- **ACCESS_DENIED:** A hívónak nincs joga az adott operation meghívására.
- **OBJECT_NOT_FOUND:** Üzleti objektum nem található. PL.: Query esetén, tranzakció esetén. Ha olyan tranzakcióval kapcsolatban akar a kliens műveletet végezni, ami nem létezik... stb.
- **REQUESTID_NOT_UNIQUE:** A context header-ben érkező requestId nem egyedi. A context header felépítésének leírása a 2.2 pontban található.
- **SUCCESS_WITH_WARNING:** Általános hibakód, ha lista alapú a hívási request, és a listából nem minden tételt sikerült végrehajtani/kezeln. Csak speciálisan olyan operation-nél van értelmezve ahol a request és response felépítése ezt indokoltá teszi.
- **TC_ITEM_NOT_FOUND:** tradeCard elemen-et kell tartalmaznia tradeCardOperation -nek
- **TC_CREATE_ELEMENT_FOUND:** Létrehozás esetén a tradeCard -on belül a tcn element-et el kell hagyni!
- **TCI_ID_FOUND:** A tradeCardItem element-en belül az id attribute-umot el kell hagyni!
- **TC_SELLER_NAME_EMPTY:** sellerName: tradeType E és D esetén kötelező
- **TC_SELLER_VAT_NUMBER_EMPTY:** sellerVatNumber: tradeType E és D esetén kötelező
- **TC_SELLER_VAT_NUMBER_ERROR:** sellerVatNumber: nem megfelelő
- **TC_SELLER_COUNTRY_EMPTY:** sellerCountry: tradeType E és D esetén kötelező
- **TC_SELLER_ADDRESS_EMPTY:** sellerAddress: tradeType E és D esetén kötelező
- **TC_DESTINATION_NAME_EMPTY:** destinationName: tradeType I és D esetén kötelező
- **TC_DESTINATION_VAT_NUMBER_EMPTY:** destinationVatNumber: tradeType I és D esetén kötelező
- **TC_DESTINATION_VAT_NUMBER_ERROR:** destinationVatNumber: értéke nem megfelelő
- **TC_DESTINATION_COUNTRY_EMPTY:** destinationCountry: tradeType I és D esetén kötelező
- **TC_DESTINATION_ADDRESS_EMPTY:** destinationAddress: tradeType I és D esetén kötelező
- **TC_LOAD_LOCATION_NOT_FOUND:** loadLocation: tradeType E és D esetén kötelező
- **TC_UNLOAD_LOCATION_NOT_FOUND:** unloadLocation: tradeType I és D esetén kötelező
- **TC_VEHICLE_NOT_FOUND:** vehicle: tradeType E és D esetén kötelező
- **TC_LOCATION_NOT_HUNGARY:** Magyar címnek kell lennie!
- **TC_LOCATION_NOT_COMPLETE:** A címadatoknál a name, vatNumber, country, zipCode, city, street mezők kötelezőek!
- **TC_DELETE_ONLY_ACTIVE:** A törlés csak akkor hajtható végre, ha még "aktív" a bejelentés!
- **TC_FINALIZE_VEHICLE_DATA_EMPTY:** vehicle/plateNumber: A bejelentés véglegesítése előtt ki kell tölteni

- **TC_FINALIZE_ARRIVAL_DATE_EMPTY:** arrivalDate: A bejelentés véglegesítése előtt ki kell tölteni
- **TC_MODIFY_BY_CARRIER_DISABLED:** A szállító nem módosíthatja a bejelentést!
- **TCI_DANG_PROD_ADRNUMBER_NOT_FOUND:** addrNumber: veszélyes tétel esetén kötelező!
- **TC_DESTINATION_MUST_BE_HUNGARY:** destinationCountry: tradeType I esetén csak 'HU' lehet
- **TC_SELLER_MUST_BE_HUNGARY:** destinationCountry: tradeType E esetén csak 'HU' lehet
- **TC_VTSZ_UNKNOWN:** Ismeretlen VTSZ
- **TC_FELIR_NEBIH_REG_NEEDED:** A bejelentéshez NÉBIH regisztráció, FELIR szám szükséges!
- **TC_UNLOAD_ADDR_MUST_BE_REG_IN_NEBIH:** Lerakodási hely csak NÉBIH által ismert hely lehet! Első beraktározási hely címlistán szerepelnie kell!
- **TC_VTSZ_TOO_SHORT:** VTSZ szám túl röviden lett megadva! 8 hosszán szükséges!
- **NO_TAX_DATA:** Az adófizető adatai az EKAER rendszerben nem megfelelőek. (nem az XML-ben szereplő adatok!)
- **TC_SELLER_CANT_BE_HUNGARY:** Feladó nem lehet Magyarországi (pl.: Import esetén)
- **INVALID_REASON_WITH_TRADE_TYPE:** A fuvar okok, a tételnél a fuvar irányultságtól függően meghatározottak lehetnek csak!

4 MELLÉKLET

Mellékletként megtalálható a szolgáltatást leíró XSD, valamint néhány példa XML!

XSD: ekaermanagement.xsd

A minta XML-ek teljes http request-eket és response-okat takarnak! Az XML-en kívül tartalmazzák, hogy milyen http header mezőket tartalmaztak a hívások és válaszok!

4.1 PÉLDA XML-EK

A példa XML-ek megtalálhatók az EKÁÉR FAQ oldalon.

validation_sample:

Validációra kérés minta. A create példát küldjük a validációra.

create_sample:

Bejelentés létrehozására példa. Kérés válasz egyaránt. Két tételt tartalmaz.

modify_sample:

A create kérésben létrehozott bejelentés módosítására példa. Fejet és tételeket is módosít, és egy új tételt is felvesz!

4.2 INTERFACE VERZIÓK

A kérések fejlécében levő header element-en belül található requestVersion element megfelelő töltésével tudja a hívó szabályozni, hogy melyik interface verziót használja. Ezzel biztosítja a visszafelé való kompatibilitást, illetve hogy a felhasználók az egyes verziók között megfelelően át tudjanak állni. A szolgáltatás a kérés verziójának megfelelően viselkedik. (pl.: Egy új verzióba bevezetett element-et nem ad vissza, ha korábbi a kérés verziója, mint amiben az adott element megjelent. Ugyan ez érvényes az enumerált típusokra, mint pl a reasonCode. Újonnan bevezett reasonCode-ot nem ad vissza korábbi kérés verzió esetén)

4.2.1 „1.0-ás verzió”

A dokumentáció 1.5-ös verziószámáig a header-ben levő requestVersion –ben 1.0 –át vártunk, vagy ha nem jött ez az opcionális element, akkor 1.0-ás verziót feltételezett a szerver. A dokumentáció 1.5-ös verziójáig így működött a szolgáltatás a dokumentációban foglaltak és a hozzá tartozó xsd-nek megfelelően. A rendszer eléréseknél ezt a szolgáltatást a „Régi 1.0 requestVersion” címszó alatt leírt URL-en érhető el.

Szolgáltatás címe:

TEST: <https://import-test-b.ekaer.nav.gov.hu/TradeCardService>

PROD: <https://import.ekaer.nav.gov.hu/TradeCardService>

4.2.2 „1.6-os verzió”

Az 1.6-os dokumentum verzióval együtt változott a szolgáltatás elérési urlje. A rendszer elérhetőségeknél a „Új 1.0 és 1.6 requestVersion, backward kompatibilis” címszó alatt jelölt URL-en érhető el az 1.6-os illetve később az az feletti verziószámmal jelölt kéréseknek megfelelően működő szolgáltatás. Az 1.6-os verzióval az elérésben a **TradeCardService változott TradeCardManagementService-re. Az új szolgáltatás beüzemelését követően a régi címen elérhető szolgáltatás továbbra is üzemel, viszont azon az újdonságok nem lesznek elérhetőek, továbbra is az 1.0 requestVersion-nek (, az 1.5 dokumentum változásig definiáltaknak) megfelelően fog működni.**

Szolgáltatás címe:

TEST: <https://import-test-b.ekaer.nav.gov.hu/TradeCardManagementService>

PROD: <https://import.ekaer.nav.gov.hu/TradeCardManagementService>

4.3 TESZT RENDSZER ELÉRHETŐSÉGE

(Régi 1.0 requestVersion)

URL: <https://import-test-b.ekaer.nav.gov.hu/TradeCardService/customer/manageTradeCards>

(Új 1.0 és 1.6 requestVersion, backward kompatibilis)

URL: <https://import-test-b.ekaer.nav.gov.hu/TradeCardManagementService/customer/manageTradeCards>

A teszt rendszer eléréséhez rendelkezni kell a megfelelő regisztrációval, valamint az XML-t előállító felhasználónak rendelkeznie kell a titkos aláíró kulccsal, ami a requestSignature hash előállításához szükséges!

A szolgáltatásnak van egy fejlesztést támogató művelete (operation), ami csak az XML validációját végzi el, de valós üzleti folyamatot nem generál. A request és response felépítése megegyezik a bejelentések tényleges kezelését végző műveletnél definiálttal! Tehát az xsd-ben definiált manageTradeCardsRequest üzenettípust (element) vár és manageTradeCardsResponse üzenettípust szolgáltat!

A validációs operation URL-je:

(Régi 1.0 requestVersion)

<https://import-test-b.ekaer.nav.gov.hu/TradeCardService/customer/validateTradeCardRequest>

(Új 1.0 és 1.6 requestVersion, backward kompatibilis)

<https://import-test-b.ekaer.nav.gov.hu/TradeCardManagementService/customer/validateTradeCardRequest>

4.4 ÉLES RENDSZER ELÉRHETŐSÉGE

(Régi 1.0 requestVersion)

<https://import.ekaer.nav.gov.hu/TradeCardService/customer/manageTradeCards>

(Új 1.0 és 1.6 requestVersion, backward kompatibilis)

<https://import.ekaer.nav.gov.hu/TradeCardManagementService/customer/manageTradeCards>

A validációs operation URL-je:

(Régi 1.0 requestVersion)

<https://import.ekaer.nav.gov.hu/TradeCardService/customer/validateTradeCardRequest>

(Új 1.0 és 1.6 requestVersion, backward kompatibilis)

<https://import.ekaer.nav.gov.hu/TradeCardManagementService/customer/validateTradeCardRequest>