



جمهوری اسلامی ایران
Islamic Republic of Iran
سازمان ملی استاندارد ایران

Iranian National Standards Organization



استاندارد ملی ایران

۲-۲۱۷۵۵

چاپ اول

۱۳۹۶

INSO
21755-2
1st.Edition

2017

Identical with
ISO 17575-2: 2016

جمع آوری هزینه به طور الکترونیکی -
تعریف واسط برنامه کاربردی برای
سامانه‌های مستقل - قسمت ۲: ارتباط و
اتصال به لایه‌های پایین‌تر

**Electronic fee collection — Application
interface definition for autonomous
systems — Part 2: Communication and
connection to the lower layers**

ICS: 35.240.60; 03.220.20

استاندارد ملی ایران شماره ۲-۲۱۷۵۵: سال ۱۳۹۶

سازمان ملی استاندارد ایران

تهران، ضلع جنوب غربی میدان ونک، خیابان ولیعصر، پلاک ۲۵۹۲

صندوق پستی: ۱۴۱۵۵-۶۱۳۹ تهران- ایران

تلفن: ۵-۸۸۸۷۹۴۶۱

دورنگار: ۸۸۸۸۷۰۸۰ و ۸۸۸۸۷۱۰۳

کرج، شهر صنعتی، میدان استاندارد

صندوق پستی: ۳۱۵۸۵-۱۶۳ کرج - ایران

تلفن: ۸-۳۲۸۰۶۰۳۱ (۰۲۶)

دورنگار: ۳۲۸۰۸۱۱۴ (۰۲۶)

رایانامه: standard@isiri.gov.ir

وبگاه: <http://www.isiri.gov.ir>

Iranian National Standardization Organization (INSO)

No. 2592 Valiasr Ave., South western corner of Vanak Sq., Tehran, Iran

P. O. Box: 14155-6139, Tehran, Iran

Tel: + 98 (21) 88879461-5

Fax: + 98 (21) 88887080, 88887103

Standard Square, Karaj, Iran

P.O. Box: 31585-163, Karaj, Iran

Tel: + 98 (26) 32806031-8

Fax: + 98 (26) 32808114

Email: standard@isiri.gov.ir

Website: <http://www.isiri.gov.ir>

به نام خدا

آشنایی با سازمان ملی استاندارد ایران

سازمان ملی استاندارد ایران به موجب بند یک ماده ۳ قانون اصلاح قوانین و مقررات مؤسسه استاندارد و تحقیقات صنعتی ایران، مصوب بهمن ماه ۱۳۷۱ تنها مرجع رسمی کشور است که وظیفه تعیین، تدوین و نشر استانداردهای ملی (رسمی) ایران را به عهده دارد.

تدوین استاندارد در حوزه‌های مختلف در کمیسیون‌های فنی مرکب از کارشناسان سازمان، صاحب‌نظران مراکز و مؤسسات علمی، پژوهشی، تولیدی و اقتصادی آگاه و مرتبط انجام می‌شود و کوششی همگام با مصالح ملی و با توجه به شرایط تولیدی، فناوری و تجاری است که از مشارکت آگاهانه و منصفانه صاحبان حق و نفع، شامل تولیدکنندگان، مصرف‌کنندگان، صادرکنندگان و واردکنندگان، مراکز علمی و تخصصی، نهادها، سازمان‌های دولتی و غیردولتی حاصل می‌شود. پیش‌نویس استانداردهای ملی ایران برای نظرخواهی به مراجع ذی‌نفع و اعضای کمیسیون‌های مربوط ارسال می‌شود و پس از دریافت نظرها و پیشنهادهای در کمیته ملی مرتبط با آن رشته طرح و در صورت تصویب، به عنوان استاندارد ملی (رسمی) ایران چاپ و منتشر می‌شود.

پیش‌نویس استانداردهایی که مؤسسات و سازمان‌های علاقه‌مند و ذی‌صلاح نیز با رعایت ضوابط تعیین شده تهیه می‌کنند در کمیته ملی طرح، بررسی و در صورت تصویب، به عنوان استاندارد ملی ایران چاپ و منتشر می‌شود. بدین ترتیب، استانداردهایی ملی تلقی می‌شود که بر اساس مقررات استاندارد ملی ایران شماره ۵ تدوین و در کمیته ملی استاندارد مربوط که در سازمان ملی استاندارد ایران تشکیل می‌شود به تصویب رسیده باشد.

سازمان ملی استاندارد ایران از اعضای اصلی سازمان بین‌المللی استاندارد (ISO)^۱، کمیسیون بین‌المللی الکتروتکنیک (IEC)^۲ و سازمان بین‌المللی اندازه‌شناسی قانونی (OIML)^۳ است و به عنوان تنها رابط^۴ کمیسیون کدکس غذایی (CAC)^۵ در کشور فعالیت می‌کند. در تدوین استانداردهای ملی ایران ضمن توجه به شرایط کلی و نیازمندی‌های خاص کشور، از آخرین پیشرفت‌های علمی، فنی و صنعتی جهان و استانداردهای بین‌المللی بهره‌گیری می‌شود.

سازمان ملی استاندارد ایران می‌تواند با رعایت موازین پیش‌بینی شده در قانون، برای حمایت از مصرف‌کنندگان، حفظ سلامت و ایمنی فردی و عمومی، حصول اطمینان از کیفیت محصولات و ملاحظات زیست‌محیطی و اقتصادی، اجرای بعضی از استانداردهای ملی ایران را برای محصولات تولیدی داخل کشور و/یا اقلام وارداتی، با تصویب شورای عالی استاندارد، اجباری کند. سازمان می‌تواند به منظور حفظ بازارهای بین‌المللی برای محصولات کشور، اجرای استاندارد کالاهای صادراتی و درجه‌بندی آن را اجباری کند. همچنین برای اطمینان بخشیدن به استفاده‌کنندگان از خدمات سازمان‌ها و مؤسسات فعال در زمینه مشاوره، آموزش، بازرسی، ممیزی و صدور گواهی سیستم‌های مدیریت کیفیت و مدیریت زیست‌محیطی، آزمایشگاه‌ها و مراکز واسنجی (کالیبراسیون) وسایل سنجش، سازمان ملی استاندارد این‌گونه سازمان‌ها و مؤسسات را بر اساس ضوابط نظام تأیید صلاحیت ایران ارزیابی می‌کند و در صورت احراز شرایط لازم، گواهینامه تأیید صلاحیت به آن‌ها اعطا و بر عملکرد آن‌ها نظارت می‌کند. ترویج دستگاه بین‌المللی یکاها، واسنجی وسایل سنجش، تعیین عیار فلزات گرانبها و انجام تحقیقات کاربردی برای ارتقای سطح استانداردهای ملی ایران از دیگر وظایف این سازمان است.

1- International Organization for Standardization

2- International Electrotechnical Commission

3- International Organization for Legal Metrology (Organisation Internationale de Metrologie Legals)

4- Contact point

5- Codex Alimentarius Commission

کمیسیون فنی تدوین استاندارد
«جمع آوری هزینه به طور الکترونیکی - تعریف واسط برنامه کاربردی برای سامانه‌های
مستقل - قسمت ۲: ارتباط و اتصال به لایه‌های پایین تر»

رئیس:

یزدیان ورجانی، علی
(دکتری، برق)

سمت و/ یا محل اشتغال:

عضو هیات علمی دانشگاه تربیت مدرس و عضو هیات مدیره
سازمان کنترل ترافیک تهران

دبیر:

قسمتی، سیمین
(فوق لیسانس مهندسی فناوری اطلاعات، ارتباطات)

مشاور مرکز آپا دانشگاه تربیت مدرس

اعضا: (اسامی به ترتیب حروف الفبا)

اسدی پویا، سمیرا
(فوق لیسانس مهندسی فناوری اطلاعات)

مدیر عامل شرکت مهندسی پویا دانش و کیفیت آوا

ترابی، مهنوش
(فوق لیسانس مهندسی فناوری اطلاعات، تجارت الکترونیک)

کارشناس استاندارد

شیخ‌الاسلامی، محمد کاظم
(دکتری، برق)

عضو هیات علمی دانشگاه تربیت مدرس

صالحی، فاطمه
(لیسانس مهندسی کامپیوتر، نرم‌افزار)

کارشناس مسئول پرداخت الکترونیک شرکت فناوری اطلاعات و
ارتباطات پاسارگاد (فناپ)

قندهاری، آزاده
(فوق لیسانس مهندسی کامپیوتر، نرم‌افزار)

عضو هیات علمی دانشگاه آزاد اسلامی واحد ساوه و کارشناس
مرکز تحقیقات مخابرات ایران

کماسی، مهدی
(لیسانس مهندسی کامپیوتر، نرم‌افزار)

کارشناس شرکت گسترش سرمایه‌گذاری ایران خودرو

محمدیان، مصطفی
(دکتری، برق)

عضو هیات علمی و معاون پژوهشی دانشکده برق و کامپیوتر
دانشگاه تربیت مدرس

معروف، سینا

کارشناس سازمان فناوری اطلاعات ایران

(لیسانس، مهندسی کامپیوتر، سخت افزار)

ویراستار:

فرهاد شیخ احمد، لیلا

کارشناس استاندارد

(فوق لیسانس مهندسی کامپیوتر، نرم افزار)



فهرست مندرجات

صفحه	عنوان
ز	پیش‌گفتار
ح	مقدمه
۱	۱ هدف و دامنه کاربرد
۲	۲ اصطلاحات و تعاریف
۶	۳ اصطلاحات کوتاه‌نوشت
۷	۴ ساختار ارتباطی انتهایی جلویی EFC
۷	۱-۴ کلیات
۸	۲-۴ رابطه با معماری کلی EFC
۸	۵ خدمات ارتباطی EFC (توابع)
۸	۱-۵ مفهوم کلی
۱۰	۲-۵ مرحله مقداردهی اولیه
۱۱	۳-۵ خدمت اولیه ارتباط نقطه به نقطه
۱۲	۴-۵ پایان نشست
۱۳	۵-۵ شکست نشست
۱۳	۶-۵ ملاحظات امنیتی
۱۳	۷-۵ گزینه‌های انتخاب رسانه
۱۳	۶ استفاده از یک پشته ارتباط
۱۳	۱-۶ کلیات
۱۴	۲-۶ الزامات برای فناوری ارتباط اساسی
۱۴	۳-۶ فراخوانی‌های پایان یافته سیار
۱۵	پیوست الف (الزامی) تعریف API انتزاعی
۲۳	پیوست ب (الزامی) پروفرم بیانیه انطباق پیاده‌سازی پروتکل (PICS)
۲۹	پیوست پ (آگاهی‌دهنده) الزامات API
۳۱	پیوست ت (آگاهی‌دهنده) مثال‌هایی از تعاریف برای زبان‌های مناسب
۳۶	پیوست ث (آگاهی‌دهنده) استفاده از این استاندارد برای EETS

پیش‌گفتار

استاندارد «جمع‌آوری هزینه به طور الکترونیکی - تعریف واسط برنامه کاربردی برای سامانه‌های مستقل - قسمت ۲: ارتباط و اتصال به لایه‌های پایین‌تر» که پیش‌نویس آن در کمیسیون‌های مربوط بر مبنای پذیرش استانداردهای بین‌المللی به عنوان استاندارد ملی ایران به روش اشاره شده در مورد الف، بند ۷، استاندارد ملی ایران شماره ۵ تهیه و تدوین شده، در سی و هفتمین اجلاس کمیته ملی استاندارد حمل و نقل مورخ ۱۳۹۶/۳/۲۲ تصویب شد. اینک این استاندارد به استناد بند یک ماده ۳ قانون اصلاح قوانین و مقررات مؤسسه استاندارد و تحقیقات صنعتی ایران، مصوب بهمن ماه ۱۳۷۱، به عنوان استاندارد ملی ایران منتشر می‌شود.

استانداردهای ملی ایران بر اساس استاندارد ملی ایران شماره ۵ (استانداردهای ملی ایران - ساختار و شیوه نگارش) تدوین می‌شوند. برای حفظ همگامی و هماهنگی با تحولات و پیشرفت‌های ملی و جهانی در زمینه صنایع، علوم و خدمات، استانداردهای ملی ایران در صورت لزوم تجدیدنظر خواهند شد و هر پیشنهادی که برای اصلاح یا تکمیل این استانداردها ارائه شود، در هنگام تجدیدنظر در کمیسیون فنی مربوط، مورد توجه قرار خواهد گرفت. بنابراین، باید همواره از آخرین تجدیدنظر استانداردهای ملی ایران استفاده کرد.

این استاندارد ملی بر مبنای پذیرش استاندارد بین‌المللی زیر به روش «معادل یکسان» تهیه و تدوین شده و شامل ترجمه تخصصی کامل متن آن به زبان فارسی می‌باشد و معادل یکسان استاندارد بین‌المللی مزبور است:

ISO 17575-2:2016, Electronic fee collection — Application interface definition for autonomous systems — Part 2: Communication and connection to the lower layers

این استاندارد یک قسمت از مجموعه استانداردهای ملی ایران شماره ۲۱۷۵۵ است.

۱-۰ سامانه‌های خودمختار

استانداردهای ISO 17575 مجموعه‌ای از استانداردها هستند که تبادل اطلاعات بین انتهای جلویی^۱ و انتهای پستی^۲ را در جمع‌آوری هزینه به طور الکترونیکی (EFC)^۳ بر اساس تجهیزات روی برد خودمختار (OBE)^۴ تعریف می‌کند. سامانه‌های EFC به طور خودکار داده‌های شارژ را برای استفاده از زیرساخت جاده جمع‌آوری می‌کند از جمله این موارد عبارتند از: عوارض^۵ بزرگراه‌ها، هزینه‌های مبتنی بر منطقه^۶ در نواحی شهری^۷، عوارض برای زیرساخت‌های خاص مانند پل‌ها و تونل‌ها، شارژ بر اساس مسافت و هزینه‌های پارکینگ.

تجهیزات روی برد خودمختار OBE بدون تکیه بر زیرساخت اختصاصی کنار جاده با به کارگیری فناوری‌های گسترده مانند سامانه‌های ماهواره‌ای ناوبری جهانی (GNSS)^۸ و شبکه‌های سلولی (CN)^۹ عمل می‌کند. این سامانه‌های EFC با نام‌های مختلفی اشاره می‌شود. علاوه بر اصطلاحات سامانه‌های خودمختار و سامانه‌های GNSS/CN، اصطلاحات سامانه موقعیت‌یابی جهانی (GPS)^{۱۰} / سامانه جهانی برای ارتباطات سیار (GSM)^{۱۱} و سامانه‌های شارژ گسترده^{۱۲} نیز استفاده می‌شود.

سامانه‌های خودمختار که از موقعیت‌یابی ماهواره‌ای استفاده می‌کند، اغلب با فناوری‌های دارای حسگر مانند مانند ژيروسکوپ، کیلومترشمار و شتاب‌سنج ترکیب می‌شود، تا وسیله نقلیه را محل‌یابی کنند و موقعیت آن را روی نقشه پیدا کنند که شامل اشیاء^{۱۳} جغرافیایی دارای شارژ، مانند جاده‌های دارای شارژ یا نواحی دارای شارژ است. از اشیاء دارای شارژ مشخصه‌های وسیله نقلیه، زمان و سایر داده‌هایی که مربوط به توصیف استفاده از جاده‌ها، تعرفه و در نهایت هزینه استفاده از جاده است، تعیین می‌شود.

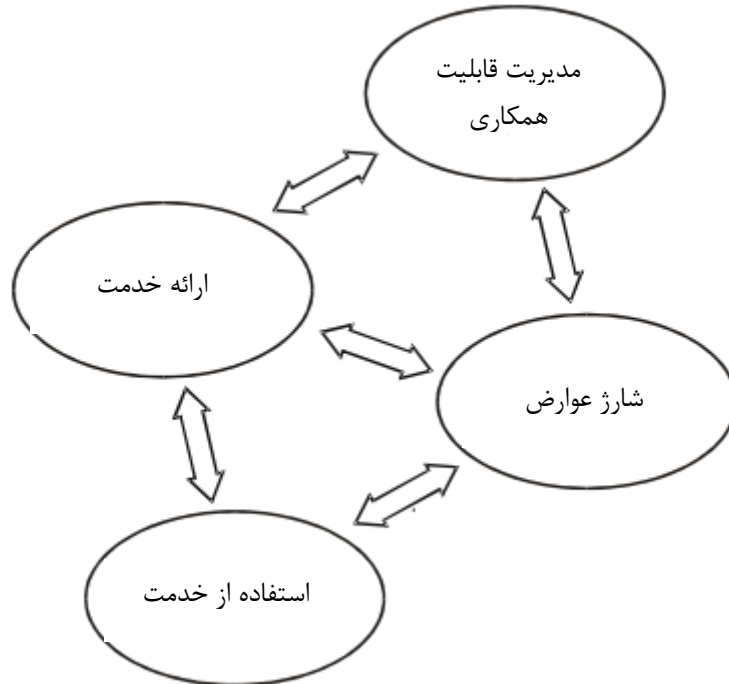
دو نقطه قوت رویکرد خودمختار در جمع‌آوری هزینه به طور الکترونیکی، انعطاف پذیری آن یعنی اجازه پیاده‌سازی تقریباً تمام اصول شارژ قابل تصور و استقلال آن از زیرساخت محلی است که در نتیجه این

-
- 1 - Front End
 - 2 - Back End
 - 3 - Electronic fee collection
 - 4 - Autonomous on-board equipment
 - 5 - Toll
 - 6 - Zone-Based
 - 7 - Urban areas
 - 8 - Global Navigation Satellite Systems
 - 9 - Cellular Networks
 - 10 - Global positioning system
 - 11 - Global system for mobile communications
 - 12 - Wide-area charging systems
 - 13 - Objects

فناوری را به قابلیت همکاری در سراسر سامانه‌های شارژ و کشورها مستعد می‌کند. قابلیت همکاری تنها می‌تواند با واسط به وضوح تعریف شده به دست آید که هدف و توجیه استاندارد ISO 17575 است.

۲-۰ معماری کسب و کار

این استاندارد، مطابق با معماری کسب و کار تعریف شده در استاندارد ISO 17573 است. با توجه به این معماری، شارژکننده عوارض، ارائه‌دهنده زیرساخت جاده است و از این رو دریافت کننده شارژهای استفاده از جاده است. شارژکننده عوارض، اقدام کننده مرتبط با نقش شارژکننده عوارض است (شکل ۱).

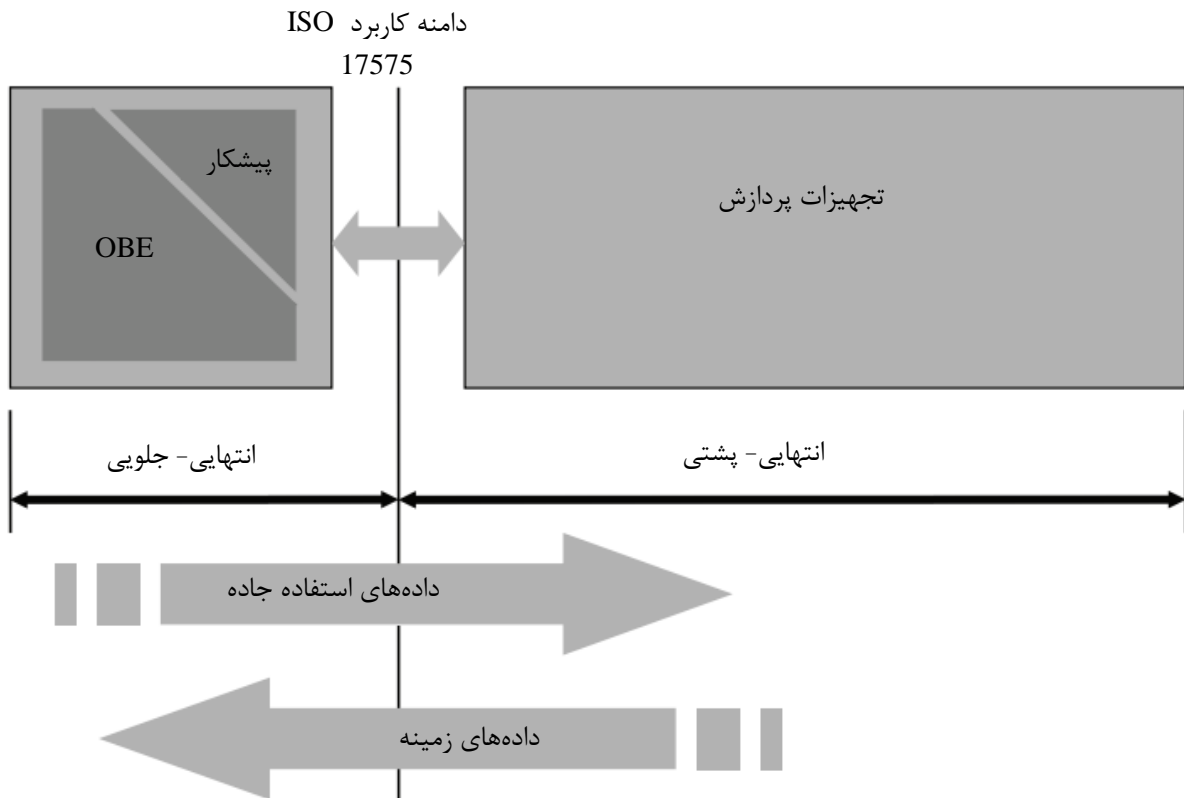


شکل ۱ - مدل مبتنی بر نقش بر اساس استاندارد ISO 17575

ارائه‌دهندگان خدمت، OBE را برای کاربران زیرساخت جاده صادر می‌کنند. ارائه‌دهندگان خدمت، مسئول عملیاتی کردن OBE هستند که میزان استفاده از جاده را در تمام سامانه‌های شارژ عوارض که وسیله نقلیه از آنها عبور کرده، ضبط می‌کنند و برای ارائه داده‌های شارژ به چندین شارژکننده عوارض هستند. به طور کلی، هر ارائه‌دهنده خدمت، داده شارژ را به چند شارژکننده عوارض ارائه می‌دهد و به طور کلی، هر شارژکننده عوارض، داده‌های شارژ را از بیش از یک ارائه دهنده خدمت دریافت می‌کند. مدیریت ارتباطات متقابل، همان طور که در شکل نشان داده شده، شامل تمام ویژگی‌ها و فعالیت‌هایی است که مجموعه ای از قواعد حاکم بر محیط کلی شارژ عوارض را تعریف و حفظ می‌کند.

۳-۰ معماری فنی

معماری فنی شکل ۲ مستقل از هرگونه تحقق عملی خاص است. این شکل نشان دهنده این واقعیت است که برخی قابلیت‌های کارکردی پردازش هم می‌تواند به OBE یا به یک مولفه (پیشکار^۱) خارج از برد مرتبط اختصاص داده شود. مثالی از قابلیت کارکردی پردازش این است که می‌تواند متوجه شود که مولفه روی برد یا خارج از برد مطابق با نقشه است یا خیر، که در آن مکان‌های وسیله نقلیه از نظر مختصات اندازه‌گیری شده از GNSS به اشیای جغرافیایی روی نقشه مربوط است که روی برد یا خارج از برد قرار گرفته اند. همچنین، محاسبه تعرفه‌ها می‌تواند با استفاده از جداول و پردازش OBE تعرفه یا با یک مولفه خارج از برد انجام شود.



شکل ۲ - معماری فنی و واسط‌های مفروض

قابلیت کارکردی ترکیبی OBE و پیشکار به عنوان انتهایی جلویی نشان داده می‌شود. پیاده‌سازی انتهایی جلویی که در آن پردازش به طور عمده در سمت OBE است به عنوان یک کارخواه زیرک (یا کارخواه هوشمند، کارخواه fat) یا لبه سنگین شناخته می‌شود. انتهایی جلویی که در آن پردازش به طور عمده خارج از برد انجام می‌شود به عنوان کارخواه thin یا معماری لبه سبک نشان داده می‌شود. بسیاری از پیاده‌سازی‌های بین انتهایی "thin" و "thick" امکان‌پذیر است، همان‌طور که در انتقال تدریجی در تکه‌های شکل ۲ نشان داده شده است. هر دو انتهای انتخاب وابسته به معماری، شایستگی‌های خود را دارد و به یک

1 - Proxy

معنی است، که در آن سازندگان با تخصیص‌های انفرادی قابلیت‌های کارکردی بین منابع روی برد و منابع مرکزی به رقابت می‌پردازند.

به خصوص برای OBE کارخواه thin، سازندگان ممکن است طیف گسترده‌ای از بهینه‌سازی‌ها را در انتقال داده‌های محلی‌سازی بین OBE و مولفه‌های خارج از برد در نظر گیرند که در آن الگوریتم‌های اختصاصی برای کاهش داده و فشرده‌سازی داده استفاده می‌شود. استانداردسازی این انتقال نه کاملاً امکان‌پذیر است و نه سودمند.

۴-۰ محل واسط و ویژگی

به منظور انتزاعی و مستقل بودن از این انتخاب‌های پیاده‌سازی وابسته به معماری، دامنه کاربرد اصلی ISO17575 تبادل داده بین انتهایی-جلویی و انتهایی-پشتی است (به خط عمودی مرتبط آن در شکل ۲ مراجعه شود). برای هر نظام عوارض، انتهایی-پشتی داده‌های زمینه را ارسال می‌کند، به این معنی که نظام عوارض را از نظر اشیاء، دارای شارژ، قواعد شارژ و در صورت لزوم، طرح تعرفه به انتهایی-جلویی توصیف می‌کند و داده‌های استفاده را از انتهایی-جلویی دریافت خواهد کرد.

لازم به ذکر است که توزیع وظایف و مسئولیت‌های بین ارائه‌دهنده خدمت و شارژکننده عوارض به صورت جداگانه متفاوت خواهد بود. بسته به وضعیت حقوقی محلی، شارژکنندگان عوارض نیازمند داده‌های «سبک‌تر» یا «سنگین‌تر» خواهند بود و ممکن است وظایف پردازش داده خاصی را به ارائه‌دهندگان خدمت واگذار کنند یا نکنند. از این رو، تعاریف داده در استاندارد ISO 17575 ممکن است در چندین واسط مفید باشد.

استاندارد ISO ۱۷۵۷۵ همچنین برای خدمات ارتباطی مستقل از رسانه پایه است که ممکن است برای ارتباط بین انتهایی-جلویی و انتهایی-پشتی استفاده شود که ممکن است مبتنی بر خط یا پیوند هوایی باشد و همچنین می‌تواند برای پیوند هوایی بین OBE و کارساز ارتباط مرکزی استفاده شود.

۵-۰ قسمت‌های ISO 17575

قسمت ۱: شارژکردن، صفات برای انتقال داده‌های استفاده را از انتهایی-جلویی تا انتهایی-پشتی تعریف می‌کند. محتویات گزارش‌های شارژ ممکن است بین نظام‌های پرداخت عوارض متفاوت باشد، از این رو، صفات برای تمام الزامات ارائه می‌شود، اعم از صفات برای داده‌های محلی‌سازی اولیه، برای اشیاء جغرافیایی مطابق با نقشه^۱ و برای تراکنش‌های عوارض با قیمت کامل. یک نظام پرداخت عوارض شامل مجموعه‌ای از قواعد^۲ برای شارژ، از جمله شبکه شارژ، اصول شارژ وسایل نقلیه مسئول و تعریفی از محتویات مورد نیاز گزارش شارژ است.

1 - Map-matched
2 - Rules

قسمت ۲: ارتباط و اتصال با لایه‌های پایین‌تر، خدمات ارتباطی پایه را برای انتقال داده در پیوند هوایی OBE یا بین انتهایی جلویی و انتهایی پشتی تعریف می‌کند. داده‌های تعریف شده در این استاندارد و استاندارد ISO 17575-3 می‌تواند با استفاده از پشته ارتباطی همان طور که در استاندارد ISO 17575-2 تعریف شده، تبادل شود، اما نیازی به این کار ندارد.

قسمت ۳: داده‌های زمینه، داده‌هایی را که باید برای توصیف سامانه‌های شارژ منفرد از نظر اشیاء جغرافیایی دارای شارژ و شارژینگ و قواعد گزارش‌دهی استفاده شود، تعریف می‌کند. برای هر سامانه شارژ عوارض، صفات همان طور که در استاندارد ISO 17575-3 تعریف شده برای انتقال داده‌ها به انتهایی جلویی استفاده می‌شود تا راهنمایی کند کدام داده را جمع‌آوری و گزارش کند.

۶-۰ کاربردهایی که نیاز است توسط ISO 17575 تحت پوشش قرار گیرد

مجموعه استانداردهای ISO 17575

- با معماری تعریف شده در استاندارد ISO 17573:2010 سازگار است،
- شارژهای استفاده از بخش‌های جاده (از جمله پل‌ها، تونل‌ها، معابر و غیره)، محدوده عبور (ورود / خروج) را پشتیبانی و از زیرساخت در یک ناحیه (بسته به مسافت، زمان)، استفاده می‌کند،
- از جمع‌آوری هزینه بر اساس واحدهای مسافت یا زمان و بر اساس وقوع حوادث پشتیبانی می‌کند،
- از مدولاسیون هزینه‌ها بر اساس دسته وسیله نقلیه، دسته جاده، زمان استفاده و نوع قرارداد (به طور مثال وسایل نقلیه معاف وسایل نقلیه با تعرفه خاص و غیره) پشتیبانی می‌کند،
- از محدود کردن هزینه‌ها بر اساس بیشینه تعریف شده در هر دوره از استفاده پشتیبانی می‌کند،
- از هزینه‌ها با وضعیت قانونی^۱ مختلف (به عنوان مثال مالیات عمومی، عوارض خصوصی) پشتیبانی می‌کند،
- از الزامات متفاوت شارژکننده‌های مختلف عوارض، به ویژه در موارد زیر پشتیبانی می‌کند
 - حوزه^۲ جغرافیایی و توصیف‌های زمینه،
 - محتویات و تواتر گزارشات عوارض،
 - بازخورد به راننده (برای مثال «چراغ سبز» یا «چراغ قرمز»)، و
 - ارائه اطلاعات دقیق بیشتر در صورت درخواست، به عنوان مثال، برای حل و فصل اختلافات،
- از هم‌پوشانی حوزه‌های عوارض جغرافیایی پشتیبانی می‌کند،
- از سازگاری‌ها با تغییرات در موارد زیر پشتیبانی می‌کند،

1 - Legal
2 - Domain

- زیرساخت‌های دارای عوارض،

- تعرفه‌ها، و

- مشارکت در طرح‌های عوارض، و

- از ارائه ضمانت اعتماد توسط ارائه‌دهنده خدمت عوارض به شارژکننده عوارض برای منشا داده از انتهایِ جلویی پشتیبانی می‌کند،

این مجموعه استاندارد شامل قسمت‌های زیر است:

قسمت ۱: شارژکردن

قسمت ۲: ارتباط و اتصال با لایه‌های پایین‌تر

قسمت ۳: داده‌های زمینه



جمع‌آوری هزینه به طور الکترونیکی - تعریف واسط برنامه کاربردی برای سامانه‌های مستقل - قسمت ۲: ارتباط و اتصال به لایه‌های پایین‌تر

۱ هدف و دامنه کاربرد

هدف از تدوین این استاندارد، تعریف نحوه انتقال تمام یا قسمتی از ساختار عنصر داده تعریف‌شده در سایر قسمت‌های این مجموعه استاندارد در هر پشته ارتباطی و رسانه مناسب برای این برنامه کاربردی است. این استاندارد تنها در پیوندهای ارتباطی سیار کاربردپذیر است (گرچه پیوند از طریق سیم، به عنوان مثال اتصالات دفتر پستی می‌تواند از روشگان یکسان استفاده کند).

برای ایجاد پیوند، دنباله‌ای از فراخوانی‌های خدمات که کانال ارتباطی را راه اندازی اولیه می‌کند، پرداختن به دریافت پیام و انتقال پایه‌بار^۱ مورد نیاز است. تعریف ارائه‌شده در این استاندارد شامل خدمات مستقل رسانه ارتباطی مورد نیاز است که توسط واسط برنامه‌نویسی برنامه کاربردی (API)^۲ انتزاعی ارائه شده است.

واسط ارتباطی به عنوان یک API در محیط برنامه‌نویسی انتخابی سامانه انتهایی جلویی (FE)^۳ پیاده‌سازی می‌شود. ویژگی API انتهایی پستی (BE)^۴ خارج از دامنه کاربرد این استاندارد است.

تعریف این API از نظر واقعی خارج از دامنه کاربرد این استاندارد است. این استاندارد یک API انتزاعی را مشخص می‌کند که معناسازی‌های API واقعی را همان‌طور که در شکل ۳ و پروفورم بیانیه انطباق پیاده‌سازی پروتکل آن (PICS)^۵ (به پیوست ب مراجعه شود) نشان داده شده تعریف می‌کند. مثالی از یک API واقعی در پیوست پ ارائه شده است که در آن هیچ تمایزی بین انتزاع و ارتباطات واقعی API ها، ایجاد نمی‌شود، اصطلاح «API ارتباطاتی» یا فقط "API" می‌تواند استفاده می‌شود.

-
- 1 - Payload
 - 2 - Application Program Interface
 - 3 - Front End
 - 4 - Back End
 - 5 - Protocol implementation conformance statement



شکل ۳ - دامنه کاربرد این استاندارد

این استاندارد همچنین ویژگی دقیق ساختار بیانیه‌های API مرتبط را ارائه می‌کند، به عنوان مثال در مورد نحوه پیاده‌سازی آن و نقش آن در یک خوشه عوارض پیچیده مانند خدمت عواض الکترونیکی اروپا (EETS)^۱ (به پیوست الف تا ث مراجعه شود).

خط‌مشی‌های انتخاب رسانه، رسیدگی به گواهینامه و سازوکارهای رمزگذاری خارج از دامنه کاربرد این استاندارد است.

۲ اصطلاحات و تعاریف

در این استاندارد، اصطلاحات و تعاریف زیر به کار می‌رود:

۱-۲

صفت

attribute

بسته قابل نشانی‌دهی متشکل از یک عنصر داده واحد یا دنباله ساختاریافته از عناصر داده است.

[منبع: زیربند 2.3 استاندارد ISO 17575-1: 2016]

1 - European Electronic Toll Service

۲-۲

اصالت سنج

authenticator

داده، احتمالاً رمزگذاری شده که برای اصالت‌سنجی استفاده می‌شود.

[منبع: زیربند 3.3 استاندارد EN 15509: 2014]

۳-۲

انتهایی پشتی

BE

Back End

قسمتی از یک سامانه اداری پشتی که واسط یک یا چند انتهایی جلویی (۳-۱۲) است.

[منبع: زیربند 3.4 استاندارد ISO 17575-1: 2016]

۴-۲

عنصر داده

data element

اطلاعات گذشته که ممکن است خودش از ساختارهای اطلاعاتی سطح پایین‌تر تشکیل شده باشد.

[منبع: زیربند 3.9 استاندارد ISO 17575-1: 2016]

۵-۲

یکپارچگی داده

data integrity

خصوصیتی که داده به شیوه غیر مجاز تغییر یا تخریب نشده است.

۶-۲

انتهایی جلویی

FE

Front End

قسمتی از سامانه عوارض که شامل تجهیزات روی برد (OBE) (۲-۹) و احتمالاً یک پیشکار (۲-۱۰) است که در آن اطلاعات عوارض جاده و داده‌های استفاده، جمع‌آوری و برای تحویل به انتهایی پشتی (۲-۳) پردازش می‌شود.

[منبع: زیربند ۳-۱۷ استاندارد ISO / TS 19299: 2015]

یادآوری ۱- انتهایی جلویی شامل تجهیزات روی برد (۲-۹) و یک پیشکار/اختیاری (۲-۱۰) است.

۷-۲

برنامه کاربردی انتهایی جلویی

Front End application

قسمتی از انتهایی جلویی بالای API است.

۸-۲

قابلیت همکاری

interoperability

توانایی سامانه برای تبادل اطلاعات و استفاده متقابل از اطلاعاتی است که تبادل شده است.

[منبع: زیربند ۳-۲-۱ استاندارد ISO / IEC / TR 10000-1: 1998، تغییر یافته است.]

۹-۲

تجهیزات روی برد

OBE

on-board equipment

تمام تجهیزات مورد نیاز روی برد یک وسیله نقلیه برای انجام توابع EFC و خدمات ارتباطی مورد نیاز است.

1 - On-board equipment

یادآوری ۱- زیر واحدهای دیگر باید اختیاری در نظر گرفته شود.

۱۰-۲

پیشکار

proxy

قسمت اختیاری/انتهاایی جلویی (۲-۶) که با تجهیزات خارجی ارتباط برقرار می کند و داده های دریافت شده را در قالب مورد توافق تحویل به انتهاایی پشتی (۲-۳) پردازش می کند.

[منبع: زیربند 13.3 استاندارد ISO 17575-1: 2016]

۱۱-۲

خدمت اولیه

service primitive

خدمات ارتباطی ابتدایی که توسط فرآیندهای برنامه کاربردی پروتکل لایه کاربرد ارائه می شود.

یادآوری- فراخوانی خدمت اولیه توسط فرآیند برنامه کاربردی به طور ضمنی فراخوانی می شود و خدمات ارائه شده توسط لایه های پروتکل پایین تر را استفاده می کند.

[منبع: زیربند 3.18 استاندارد ISO 14906: 2011، اصلاح شده - موضوع حذف شده است]

۱۲-۲

ارائه دهنده خدمت عوارض

toll service provider

هستار ارائه دهنده خدمات عوارض در یک یا چند دامنه عوارض است.

[منبع: زیربند ۳-۲۳ استاندارد ISO 17573: 2010، اصلاح شده - تعریف مختصر شده است]

۱۳-۲

عوارض

toll

شارژ، مالیات یا عوارض گمرکی مرتبط با استفاده از یک وسیله نقلیه در دامنه عوارض است.

[منبع: زیربند ۳-۴۲ استاندارد ISO / TS 19299: 2015، اصلاح شده - «هر» از قبل «شارژ» حذف شده است]

یادآوری- این تعریف به تعریف قدیمی عوارض به عنوان شارژ، مالیات یا عوارض گمرکی برای کسب اجازه عبور از مانع یا ادامه در امتداد جاده، پل و غیره گسترش یافته است. این تعریف همچنین شامل هزینه‌های در نظر گرفته شده به عنوان تعهد (اجرایی) برای مثال، مالیات یا عوارض گمرکی است.

۱۴-۲

شارژکننده عوارض

toll charger

هستاری که عوارض برای استفاده از وسایل نقلیه در یک دامنه عوارض را وضع می‌کند.

[منبع: زیربند ۳-۱۶ استاندارد ISO 17573: 2010، اصلاح شده - «قانونی» از قبل «هستار» حذف و «استفاده» اضافه شده است.]

۳ کوتاه‌نوشت‌ها

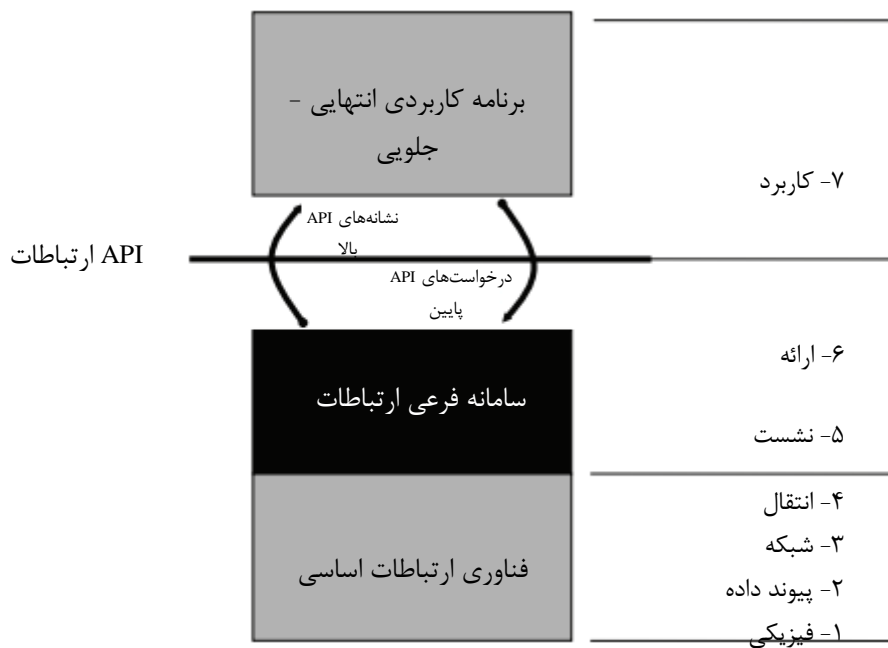
در این استاندارد، کوتاه‌نوشت‌های زیر به کار می‌رود، مگر این که مورد دیگری مشخص شده باشد.

ADU	Application data unit (ISO 14906)	واحد داده برنامه کاربردی (ISO 14906)
APDU	Application protocol data unit (ISO 14906)	واحد داده پروتکل برنامه کاربردی (ISO 14906)
AP	Application process (ISO 14906)	فرآیند برنامه کاربردی (ISO 14906)
API	Application programming interface	واسط برنامه‌نویسی برنامه کاربردی
ASN.1	Abstract Syntax Notation One (ISO/IEC 8824-1)	نشانه گذاری نحو انتزاعی ۱ (ISO/IEC 8824-1)
BE	Back End	انتهایی پشتی
CN	Cellular network	شبکه سلولی
EID	Element identifier (ISO 14906)	شناسانه عنصر (ISO 14906)
FE	Front End	انتهایی جلویی
GNSS	Global Navigation Satellite System	سامانه ماهواره‌ای ناوبری جهانی
OBE	On-board equipment	تجهیزات روی برد
VAT	Value added tax	مالیات بر ارزش افزوده

۴ ساختار ارتباطی انتهایی جلویی EFC

۱-۴ کلیات

سامانه فرعی ارتباط برای ایجاد و برقراری ارتباط بین برنامه کاربردی انتهایی جلویی (FE) و انتهایی پشتی (BE) مورد نیاز است. این سامانه انتقال داده‌ها را برای برنامه کاربردی FE عوارض از طریق نشست ارتباطات که در خط نشان داده شده در شکل ۴ رخ می‌دهد، ارائه می‌کند. در مواردی که پیشکار در سامانه FE وجود دارد، سامانه فرعی ارتباطات، ارتباطات بین BE و پیشکار را تعریف می‌کند. پیوند بین پیشکار و تجهیزات روی برد (OBE) خارج از دامنه کاربرد این استاندارد است. در مواردی که پیشکار وجود ندارد («کارخواه هوشمند»)، سامانه فرعی ارتباطات، ارتباطات بین OBE و BE را تعریف می‌کند.



شکل ۴ - رابطه بین برنامه کاربردی و پشتی پروتکل

سامانه فرعی ارتباطات به دو مولفه مجزا تقسیم می‌شود. API ارتباطات، قابلیت‌های کارکردی ارتباط را به برنامه کاربردی FE ارائه می‌دهد. در زیر آن فناوری ارتباطات اساسی وجود دارد که قابلیت‌های کارکردی خلاصه API را ارائه می‌کند. اگرچه API مستقل از فناوری اساسی است اما به صورت مستقیم تعدادی از تقاضاهای کارکردی را جای می‌دهد. به همین دلیل، الزامات کارکردی در فناوری ارتباطات اساسی در زیربند ۲-۶ بیان شده است.

برخی از فناوری‌های اساسی بیش از سایر فناوری‌ها دارای قابلیت هستند. در مواردی که فناوری دارای قابلیت، در حال استفاده است، کد واسط API به فناوری اساسی، کارکردی کمی بیشتر از یک عبور ساده را به کار می‌برد. برای فناوری‌های لایه انتقال ساده‌تر، سامانه فرعی ارتباطات باید قابل توجه‌تر باشد.

انتظار می‌رود که این API‌ها در BE «منعکس» شود، FE ها و BE ها می‌تواند در زیرساخت‌های حامل دلخواه ارتباط برقرار کند. جزئیات تعاریف API انتزاعی در پیوست الف مشخص شده است. ویژگی BE API خارج از دامنه کاربرد این استاندارد است.

۲-۴ رابطه با معماری کلی EFC

API ارتباطات لایه‌های پایین‌تر واسط نشان داده شده در شکل ۵ را ارائه می‌کند. API، دانش معناسازی واحدهای داده برنامه کاربردی (ADU ها) که حمل می‌کند را ندارد. API بین ADU های «خاص استاندارد» و «دلخواه» تمایز می‌گذارد اما هیچ دانش معناسازی در مورد معنی آن ندارد و به سادگی آنها را به عنوان جریان هشتمایی شفاف بالای حامل ارتباط دلخواه که در زمان اجرا انتخاب شده، حمل می‌کند.

۵ خدمات ارتباطی EFC (توابع)

۱-۵ مفهوم کلی

API دو «نوع» پیام (ADU) را حمل می‌کند: عناصر ساختار که به طور مستقیم به تعاریف سایر قسمت‌های ISO 17575 مربوط است و عناصر بدون ساختار که در خارج از دامنه کاربرد این استاندارد است و در این استاندارد مورد توجه بیشتر قرار نمی‌گیرند.

شناسایی عناصر داده برای انتقال و پایه بار مرتبط خارج از دامنه کاربرد این استاندارد است.

یادآوری ۱- protocolVersion (قسمتی از ChargeReport) همان طور که در استاندارد ISO 17575-1:2016 تعریف می‌شود و protocolVersion و tollContext (هر دو قسمتی از aduHeader) همان طور که در استاندارد ISO 17575-3:2016 تعریف می‌شود هنگام پیاده‌سازی تراکنش(های) خاص می‌تواند مفید باشد.

API انتزاعی برای خدمات ارتباطات می‌تواند در هر محیط برنامه‌نویسی که مفهوم تحویل رویداد را تعریف می‌کند، پیاده‌سازی شود و اجازه گزارش اطلاعات یا تحویل نتایج عملیات به برنامه کاربردی FE را به API دهد. موارد زیر دنباله کلی رویدادها است:

- مقداردهی اولیه و پارامتری کردن واسط ارتباطات،

- ایجاد یک نشست ارتباط،

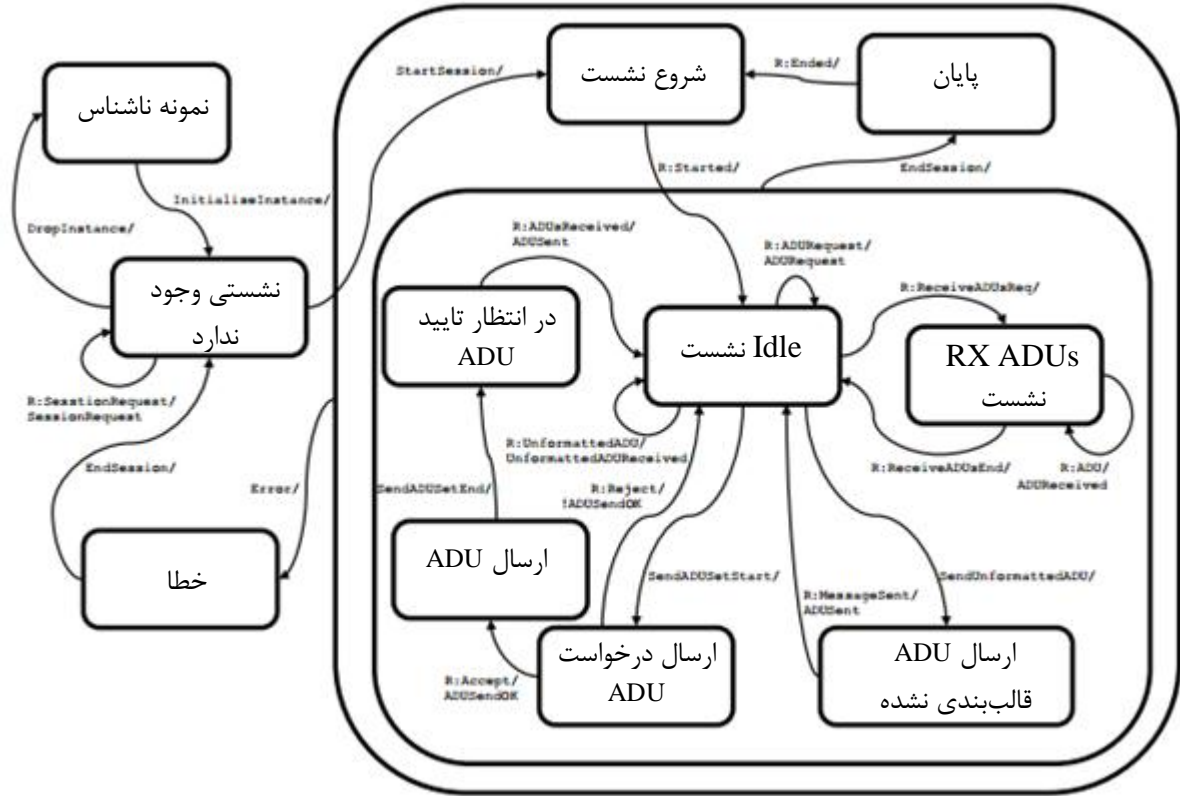
- انتقال داده در زمینه نشست،

- پایان دادن به نشست ارتباط و

- انتشار^۱ واسط ارتباطات.

1 - De-initializing

در مورد عادی، برنامه کاربردی FE (تعدادی از) واسط‌های ارتباطات را هنگامی که در ابتدا شروع می‌شود مقداردهی اولیه خواهد کرد. سپس یک نشست فعال به عنوان اقدام مستقیم توسط برنامه کاربردی FE یا در پاسخ به درخواست‌های ورودی برای یک نشست از BE ایجاد می‌شود. جریان وقایع از طریق طول عمر در بندهای ۵-۲ تا ۷-۵ نشان داده شده و تعاریف مربوط در پیوست الف ارائه شده است.



شکل ۵ - نمودار وضعیت نشست

پشته ارتباطات را به دو رده پایین فراخوانی می‌کند: همزمان و ناهمزمان. فراخوانی‌های همزمان که نتایج را به سرعت ارائه می‌کند به API هایی که می‌تواند به سرعت نتیجه را برگرداند، محدود می‌شود (InitialiseInstance و GetParameter). سایر فراخوانی‌های API ناهمزمان هستند و نتایج خود را از طریق سازوکار رویداد که در طول InitialiseInstance مقداردهی اولیه می‌شود، برمی‌گرداند.

یادآوری ۲- این موضوع مانع از قفل شدن موضوعات می‌شود، در حالی که منتظر برگرداندن شدن اطلاعات است و در نتیجه الزامات برنامه‌نویسی چند رشته‌ای را کاهش می‌دهد، این الزامات اغلب برای برنامه‌های کاربردی تعبیه شده نظیر مواردی که به طور معمول در حوزه برنامه کاربردی دیده می‌شود، نامناسب است.

شکل ۶ مثالی از رابطه و تعاملات بین وضعیت‌های مختلف را که در API قابل مشاهده است، نشان می‌دهد.

مثال- نمایش جریان پیام برای مواردی است که برنامه کاربردی FE می‌خواهد یک نشست ایجاد کند، برخی اطلاعات را با BE تبادل کند و سپس نشست را به پایان می‌رساند. مراجع رویدادهای API به تعریف پیوست الف اشاره دارد.



شکل ۶ - جریان پیام مثال و نمودار وضعیت نشست

۲-۵ مرحله مقداردهی اولیه

۱-۲-۵ کلیات

برنامه کاربردی FE باید واسط‌های ارتباطات که می‌خواهد از آنها استفاده کند را با استفاده از فراخوانی `InitialiseInstance` مقداردهی اولیه کند. برای هر نمونه ایجادشده، برنامه کاربردی FE تعریف می‌کند کدام پشته ارتباطات اساسی استفاده شود. بیش از یک واسط ممکن است در یک زمان استفاده شود و انتخاب بین واسط‌ها، تصمیم برنامه کاربردی FE است. برنامه کاربردی FE همچنین مجموعه‌ای از قابلیت‌های کارکردی پذیرش رویداد را ارائه می‌کند که توسط API برای اطلاع تغییر وضعیت به برنامه کاربردی FE استفاده می‌شود.

هر پارامتر اضافی از نمونه که مورد نیاز است توسط فراخوانی‌های مکرر `SetParameter` انجام می‌شود. مجموعه‌ای از پارامترها توسط خود API ارتباطات به رسمیت شناخته می‌شود و مواردی که به رسمیت شناخته نمی‌شود از طریق شفاف به پشته اساسی عبور می‌کنند. پرس‌وجوها برای وضعیت موجود پارامترها ممکن است از طریق فراخوانی `GetParameter` ایجاد شود.

هنگامی که این فرایند تکمیل شد، برنامه کاربردی FE به (مجموعه‌ای از) نمونه‌های ارتباطات دسترسی دارد. تغییرات وضعیت برای رویدادهایی که در هر یک از این واسط‌ها رخ می‌دهد با استفاده از اطلاع‌رسانی رویداد InstanceStateChange تحویل می‌شود.

نموداری که رابطه و تعاملات بین هر یک از این وضعیت‌ها را نشان می‌دهد در مثال ارائه‌شده در شکل ۶ نشان داده شده است.

۲-۲-۵ درخواست نشست (برنامه کاربردی BE به FE) ورودی

ممکن است BE مایل به ایجاد ارتباطات با FE باشد. در این مورد، BE هرگونه اقدام مناسب برای فناوری ارتباطات خاص در نظر گرفته شده را انجام می‌دهد. این کار سبب می‌شود برنامه کاربردی FE از درخواست در برخی نقاط در همان لحظه با استفاده از رویداد SessionRequested با داده SessionHandle مطلع شود که شناسانه‌ای که توصیه می‌شود برنامه کاربردی FE برای نشست استفاده کند را نشان می‌دهد. از اینجا، فرآیند مشابه ایجاد نشست خروجی است، همان طور که در زیربند ۲-۲-۵ نشان داده شده است.

یادآوری- برنامه کاربردی FE ممکن است یک درخواست نشست را برای مدت زمان دلخواه با توجه به محدودیت‌های عملیاتی ارائه کند.

۳-۲-۵ ایجاد نشست خروجی (برنامه کاربردی FE)

برنامه کاربردی FE به صورت غیر همزمان یا از طریق فرآیند ۲-۲-۵، نشستی در زمینه ارتباط انتخاب‌شده با استفاده از StartSession را که با اطلاعات در مورد نشستی که ایجاد می‌شود پارامترشده است، درخواست می‌کند. این موضوع بلافاصله برمی‌گردد در عین حال فرایند ایجاد نشست را در لایه‌های فناوری ارتباطات پایان شروع می‌کند.

هنگامی که نشست ایجاد شد، رویداد InstanceStateChange برنامه کاربردی FE را مطلع می‌کند. نشست در حال حاضر فعال است و ارتباط بین برنامه کاربردی FE و BE می‌تواند در زمینه آن ایجاد شود.

۳-۵ خدمت اولیه ارتباط نقطه به نقطه

۱-۳-۵ کلیات

هنگامی که نشست فعال است، ADU ها می‌توانند به BE توسط برنامه کاربردی FE ارسال شود. قابلیت‌های کارکردی ارتباط ساختاریافته (به عنوان مثال آگاه از زمینه) و بدون ساختار (بدون آگاهی از زمینه) ارائه شده است. تمام ADU ها (ساختاریافته و بدون ساختار) برای لایه‌های ارتباطات مبهم است و تمایز تنها برای جلوگیری از سربار demultiplexing لایه بالاتر است.

در زمینه نشست ارتباط، برنامه کاربردی FE، اصلی در نظر گرفته شده و کنترل نشست را عهده دارد.

۲-۳-۵ پیام‌های بدون ساختار (ADU ها)

تسهیلی برای انتقال ADU های قالب‌بندی نشده در زیرساخت‌های ارتباطی ارائه می‌شود. این تسهیل معمولاً به منظور ارتقاء نرم‌افزار و سایر تبادلات اطلاعات تولیدکننده خاص استفاده می‌شود.

ADU ها از طریق فراخوانی `SendUnformattedADU` ارسال می‌شود. بیشینه اندازه ADU که می‌تواند از طریق این سازوکار ارسال شود، از طریق پارامتر `API` در دسترس است. هنگامی که پیام منتقل شد، نشانه‌ای باید (`ADUSent`) ارائه شود که پیام با موفقیت در انتهای از دور^۱ دریافت شده و تراکنش‌های بیشتر امکان‌پذیر است.

۳-۳-۵ پیام‌های ساختاریافته (ADU ها)

ADU های ساختاریافته باید در مجموعه‌ها ارسال شود. مجموعه‌ای باید برای انتقال از طریق فراخوانی `SendADUSetStart` ایجاد شود. این کار، مجوز ورود به وضعیت `ADU-Sending` ساختاریافته از انتهای دور^۲ را درخواست می‌کند. ADU ها به مجموعه‌ای که از طریق فراخوانی `SendADU` ارسال می‌شود، اضافه می‌شود که باید مقدار بازگشت فضای باقی مانده در بافر انتقال را بازگرداند. مجموعه باید با فراخوانی `SendADUSetEnd` بسته شود. هنگامی که انتقال کامل شد، برنامه کاربردی FE باید از طریق رویداد `ADUSent` آگاه شود.

این گزینه بهینه‌سازی پیاده‌سازی است چرا که ADU در حالی که مجموعه هنوز در حال هم‌گذاری^۳ است، منتقل می‌شود. این بدان معنی است که فضای بافر در دسترس که با بازگشت از `SendADU` نشان داده شده است باید بیش از محاسبات محلی در برنامه کاربردی FE مورد اعتماد قرار گیرد، چرا که زمانی که عناصر منتقل می‌شود، فضای بیشتری می‌تواند در دسترس قرار گیرد.

۴-۵ پایان نشست

در زمان پایان نشست، برنامه کاربردی FE باید `EndSession` را فراخوانی کند و کد دلیل مناسب را ارائه کند. این کار باید فرآیند پایان دادن به نشست در فناوری ارتباطات اساسی را شروع کند. پایان‌دهی نشست ممکن است فوری نباشد و تراکنش‌های در حال پردازش قبل از بسته شدن تکمیل شود. پیاده‌سازی کنندگان برنامه کاربردی FE باید انتظار رسیدگی به فعالیت‌ها را تا زمانی که `InstanceStateChange` به وضعیت `STNoSession` دریافت شود از نشست باز داشته باشند.

1 - Remote end

2 - Far end

3 - Assemble

۵-۵ شکست نشست

نشست ممکن است از طریق بدون اشکال بودن BE یا برنامه کاربردی FE به وسیله شکست زیرساخت ارتباطات مداخله‌ای، پایان یابد. در این مثال، برنامه کاربردی FE باید از حقیقت ماجرا با استفاده از InstanceStateChange به STErrorred اطلاع داشته باشد. هرگونه ADU ارسال شده توسط برنامه کاربردی FE که تایید ADUSent را دریافت نکرده باید شکست خورده فرض شود.

فناوری ارتباطات نباید سعی در ایجاد دوباره نشست به صورت خودکار کند؛ این کار مسئولیت برنامه کاربردی FE است. اگر نشست ارتباطات، نتیجه درخواست BE بود، نشست باید دوباره در اولین فرصت مناسب ایجاد شود. اگر نشست نتیجه رویداد برنامه کاربردی FE بود، تصمیم به ایجاد دوباره به برنامه کاربردی FE واگذار می‌شود.

توجه داشته باشید که برخی از فناوری‌های ارتباطات، بر اساس ماهیت خود، به طور متناوب در دسترس است. برای پشتیبانی، فراخوانی StackAvail ارائه می‌شود تا به برنامه کاربردی FE اجازه دهد انتخاب هوشمندانه‌ای بین زیرساخت‌های موجود داشته باشد. اگر رسانه در طول نشست شکست بخورد، این موضوع باید با توجه به فرآیندهای اشاره شده در بالا نشان داده شود.

۶-۵ ملاحظات امنیتی

توصیه می‌شود تمام ارتباطات رمزگذاری شود. رسیدگی به گواهینامه و سازوکار رمزگذاری خارج از دامنه کاربرد این استاندارد است.

۷-۵ گزینه‌های انتخاب رسانه

رسانه ممکن است بین استفاده از داشتن چندین نمونه که با فناوری‌های ارتباطی مستقل نمونه‌شده، انتخاب شود. قابلیت‌های هر رسانه می‌تواند از طریق فراخوانی‌های GetParameter تعیین شود و در دسترس بودن محلی هر رسانه خاص در یک نمونه فعال می‌تواند از طریق فراخوانی StackAvail تعیین شود.

۶ استفاده از یک پشته ارتباط

۱-۶ کلیات

این استاندارد، اجازه استفاده همزمان فناوری‌های ارتباط متعدد و برنامه کاربردی FE که قادر به انتخاب یکی از مناسب‌ترین آنها برای ایجاد ارتباط خاص است را می‌دهد. پیاده‌سازی باید این قابلیت را برای OBE‌های «چند وضعیتی» داشته باشد تا اجازه استفاده از فناوری‌های مختلف بسته به ضرورت ارتباطات، قابلیت و میزان زیرساخت‌های موجود را بدهد.

با این حال، نیاز به ارائه چند قابلیت اساسی از هر فناوری ارتباط (یا پشته‌ای که در بالای فناوری قرار می‌گیرد) وجود دارد.

۲-۶ الزامات برای فناوری ارتباط اساسی

این استاندارد، اجازه طیف گسترده‌ای از فناوری‌های ارتباط اساسی را می‌دهد. با این حال، فهرست خواص زیر برای این پشته‌ها باید ارائه شود:

- قابلیت پشتیبانی یا تقلید «نشست»؛
- قابلیت اطمینان از انتقال ADU ها به صورت قابل اعتماد، به ترتیب، دو جهته در سراسر پیوند؛
- قابلیت گزارش‌دهی دریافت امن ADU ها در انتهای از دور پیوند؛
- قابلیت انتقال عناصر داده به طول دلخواه بین FE و BE؛
- قابلیت ایجاد یک نشست از FE به BE؛
- برخی وسایل نشانک‌دهی^۱ یک تقاضا از BE به برنامه کاربردی FE که BE بخواند نشست را ایجاد کند؛
- قابلیت اطمینان از تشخیص قابل اعتماد از دست دادن یک نشست و تحویل آن اطلاعات به API ارتباطات؛
- تحویل ADU ها در سراسر پیوند به صورت به موقع؛
- قابلیت حمل مقدار مناسب داده.

۳-۶ فراخوانی‌های پایان یافته سیار

رویکرد اتخاذ شده در این استاندارد هرگز نیاز به باز بودن پورت ارتباط داخل باند ورودی ندارد. اگر BE بخواند که اتصال به یک برنامه کاربردی FE به هر منظوری ایجاد کند، می‌تواند از نشانک‌دهی خارج از باند استفاده کند؛ این سازوکارها خارج از دامنه کاربرد این استاندارد است. طیف وسیعی از گزینه‌های نشانک‌دهی خارج از باند قابل توجه است (به عنوان مثال پیام‌های SMS، دکمه فشاری^۲ در واحد، سیگنال پخش و غیره). پیامد این رویکرد، استفاده از نشانک‌دهی خارج از باند است که هرگز نیاز به نقطه پایان قابل نشانی‌دهی IP برای فراخوانی پایان یافته سیار وجود ندارد. این کار از موضوعات امنیتی جلوگیری کرده و همچنین مسائل مربوط به ترجمه نشانی شبکه و زیرشبکه خصوصی را ساده می‌کند، چرا که برنامه کاربردی FE تنها نیاز به ایجاد اتصال خروجی دارد.

یادآوری - پیامد این رویکرد این است که هیچ گونه الزامی برای نقطه پایان قابل نشانی‌دهی IP وجود ندارد. این موضوع از موضوعات امنیتی مورد بحث بالا جلوگیری می‌کند و مسائل مربوط به ترجمه نشانی شبکه و زیرشبکه خصوصی را ساده می‌کند، چرا که برنامه کاربردی FE تنها نیاز به ایجاد اتصال خروجی دارد.

1 - Signaling
2 - Push button

پیوست الف

(الزامی)

تعریف API انتزاعی

الف-۱ کلیات

این پیوست خصوصیات دستورات و جریان داده‌ها را در API بین نرم‌افزار خاص EFC تعریف می‌کند - برنامه کاربردی انتهایی جلویی (FE) و لایه واسط این زیر سامانه ارتباطات سکو برای اجرای برنامه کاربردی EFC استفاده می‌شود، همچنین شکل ۴ مراجعه شود. مثال‌هایی از این که چگونه این واسط می‌تواند تشخیص داده شود در پیوست پ و ت ارائه شده است.

API ارتباطات متشکل از یک API «پایین» از برنامه کاربردی FE به پشته ارتباطات و API «بالا» از پشته ارتباطات به برنامه کاربردی FE است. هر یک از آنان به نوبه خود در نظر گرفته خواهد شد.

الف-۲ API پایین (برنامه کاربردی FE برای پشته ارتباطات)

الف-۲-۱ InitialiseInstance

هدف: مقداردهی اولیه ارتباطات API برای استفاده. واسط در شرایط STNoSession مقداردهی اولیه می‌شود.

می‌گیرد: StackID پشته ارتباطات اساسی که به کار گرفته می‌شود

Callbacks به گرداننده رویداد که برای این مثال استفاده می‌شود، اشاره دارد. به زیربند ۵-۲ مراجعه شود

پیش شرط: واسط نباید قبلاً مقداردهی اولیه شده باشد

باز می‌گرداند: رسیدگی نمونه API. این رسیدگی برای تمام ارتباطات استفاده می‌شود

خطاها: اگر ایجاد نمونه ممکن نباشد، یک نمونه نامعتبر را باز می‌گرداند

الف-۲-۲ SetParameter

هدف: تعیین پارامتر برای این نمونه API. تمام پارامترها و مقادیر به عنوان رشته بیان می‌شود.

می‌گیرد: Instance نمونه‌ای است که این پارامتر به آن مربوط است

Parameter پارامتر تعیین می‌شود

Value مقداری که به پارامتر داده می‌شود

پیش شرط: نمونه معتبر مورد نیاز است

باز می‌گرداند: کد خطا

خطاها: ERNotSet ,ERNoError

الف-۲-۳ GetParameter

هدف: مقدار پارامتر را برای این نمونه API می‌گیرد

می‌گیرد: Instance نمونه‌ای که این پارامتر به آن مربوط است

Parameter پارامتر تعیین می‌شود

پیش شرط: یک نمونه معتبر مورد نیاز است

باز می‌گرداند: مقدار پارامتر مشخص شده به عنوان رشته

خطاها: اگر پارامتر شناخته شده نباشد، رشته نامعتبر است

الف-۲-۴ DeleteParameter

هدف: پارامتر از این نمونه API حذف می‌شود

می‌گیرد: Instance نمونه‌ای که این پارامتر به آن مربوط است

Parameter پارامتر تعیین می‌شود

پیش شرط: یک نمونه معتبر مورد نیاز است

باز می‌گرداند: کد خطا

خطاها: ERNotSet ,ERNoError

الف-۲-۵ StackAvail

هدف: نشان می‌دهد که پشته ارتباطات در حال حاضر موجود است

می‌گیرد: Instance نمونه‌ای که این واریسی به آن مربوط است

پیش شرط: نمونه معتبر مورد نیاز است

باز می‌گرداند: بولین نشان می‌دهد که پشته در حال حاضر موجود است

خطاها: در صورت خطا FALSE باز می‌گرداند

الف-۲-۶ DropInstance

هدف:	واسط API را حذف می کند
می گیرد:	Instance نمونه‌ای که این درخواست افت به آن مربوط است
	Severity سرعتی که با آن، این واسط باید حذف شود (SEUrgent, SENormal, SEUnconditional)
پیش شرط:	یک نمونه معتبر در وضعیت STNoSession (برای SENormal و SEUrgent) مورد نیاز است
باز می گرداند:	کد خطا
خطاها:	ERNoError, ERBadState, ERUnknownInstance

الف-۲-۷ StartSession

هدف:	فرآیند ایجاد نشست در زمینه این واسط را شروع می کند (به عنوان مثال با استفاده از کانال ارتباطات و پارامترهای ایجاد شده). ایجاد نشست توسط یک رویداد تعریف می شود که نشان دهنده تغییری در STSessionIdle است، زمانی که نشست وجود دارد. پارامتری کردن نشست از طریق روش‌های SetParameter / GetParameter است.
می گیرد:	Instance نمونه‌ای که در آن، این درخواست نشست باید برآورده شود
	Reason دلیل ایجاد این نشست
	SessionHandle هرگونه نشست که به اطلاعات مورد نیاز رسیدگی می کند
پیش شرط:	نمونه فعال و هیچ نشست فعالی در حال حاضر وجود نداشته باشد. پارامتری سازی درست وجود داشته باشد.
باز می گرداند:	کد خطا
خطاها:	ERInSession, ERUnknownEndpoint, ERNoInstance, ERInSession, ERNoError, ERSessionFailed

الف-۲-۸ EndSession

هدف:	شروع فرآیند برای پایان دادن به یک نشست فعال در زمینه این نمونه. پایان نشست از طریق رویداد تغییر وضعیت به وضعیت STNoSession تایید می شود. در شرایط عادی هر گونه تراکنش‌های برجسته در زمینه نشست قبل از بسته شدن تکمیل
------	--

خواهد شد.

می‌گیرد: Instance نمونه‌ای که در آن این نشست پایان یافته است

Reason دلیل پایان نشست

پیش شرط: وجود یک نشست فعال

باز می‌گرداند: کد خطا

خطاها: ERInSession, ERNoInstance, ERNoError

الف-۲-۹ SendUnformattedADU

هدف: ارسال ADU قالب‌بندی نشده به انتهای دیگر پیوندهای ارتباطی. در زمینه این مستند، ADU قالب‌بندی نشده، هیچ معنی خاص برای API ارتباطات ندارد و توسط کد خاص فروشنده پردازش می‌شود. تایید دریافت در انتهای دور^۱ با استفاده از نشانه رویداد است.

می‌گیرد: Instance نمونه‌ای که در آن، این پیام ارسال می‌شود

MessageLen طول پیام که به صورت هشتایی ارسال می‌شود

Message خود پیام

پیش شرط: وجود یک نشست فعال بدون تراکنش در حال انجام

باز می‌گرداند: کد خطا

خطاها: ERBadState, ERNoInstance, ERSessionFailed, ERNoError

الف-۲-۱۰ SendADUSetStart

هدف: ارسال درخواست برای شروع به ارسال مجموعه‌ای از ADU های ساختاریافته در پایان راه دور پیوندهای ارتباطی. تایید پذیرش انتقال با استفاده از رویداد

می‌گیرد: Instance نمونه‌ای که در آن، این پیام ارسال می‌شود

پیش شرط: وجود یک نشست فعال بدون تراکنش در حال انجام

باز می‌گرداند: کد خطا

1 - Far end

خطاها: ERNoError ,ERSessionFailed ,ERBadState ,ERNoInstance

الف-۲-۱۱ SendADU

هدف: برای اضافه کردن ADU به مجموعه ارسالی به پایان راه دور پیوند ارتباطی

می‌گیرد: Instance نمونه‌ای که در آن، این پیام ارسال می‌شود

ElementLen طول عنصر که به صورت هشتایی ارسال می‌شود

Element خود عنصر

پیش شرط: یک نشست فعال بدون تراکنش در حال انجام وجود دارد و مجموعه عنصر برای ساخت باز است (به الف-۲-۱۰ مراجعه شود)

باز می‌گرداند: تعداد هشتایی باقی مانده در بافر انتقال

خطاها: در صورت خطا 0 باز می‌گرداند

الف-۲-۱۲ SendADUSetEnd

هدف: برای نتیجه‌گیری از مجموعه ADU های ساختاریافته (همان طور که در سایر قسمت‌های این استاندارد تعریف شده است) که به پایان دور پیوند ارتباطی ارسال می‌شود. دریافت انتهای دور عناصر با استفاده از رویداد ADUSent تایید می‌شود.

می‌گیرد: Instance نمونه‌ای که در آن، این پیام ارسال می‌شود

پیش شرط: یک نشست فعال بدون تراکنش در حال انجام وجود دارد و مجموعه عنصر برای ساخت باز است (به الف-۲-۱۰ مراجعه شود)

باز می‌گرداند: کد خطا

خطاها: ERNoError ,ERSessionFailed ,ERBadState

الف-۲-۱۳ CommsQuery

هدف: نظر سنجی در مورد وضعیت کنونی نمونه ارتباطات

می‌گیرد: Instance نمونه‌ای که این درخواست نظرسنجی مربوط به آن است

پیش شرط: یک نمونه معتبر مورد نیاز است

باز می‌گرداند: وضعیت فعلی نشست:

STUnknownInstance نمونه نامعتبر است

STNoSession هیچ نشستگی در حال انجام نیست

STStarting شروع نشست

STSessionIdle نشست باز و آماده به کار است

STSendingADU ارسال ADU ها

STSendingADURequest درخواست مجوز برای ارسال ADU ها

STSendingUnformattedADU ارسال ADU قالب‌بندی نشده

STAwaitingADUConfirm در انتظار تایید دریافت ADU

STSessionRxADU دریافت ADU ها از انتهای دور

STErrored در وضعیت خطا

STEnding در فرایند پایان

STUnknownInstance

خطاها:

یادآوری- خطای STUnknownInstance به عنوان وضعیت گزارش می‌شود.

الف-۳ API بالا (پشته ارتباطات به برنامه کاربردی FE)

الف-۳-۱ InstanceStateChange

هدف: به برنامه کاربردی FE نشان می‌دهد که تغییری در وضعیت نمونه ارتباط رخ داده است

دریافت می‌کند: Instance نمونه‌ای که این نشانه مربوط به آن است

OldState وضعیت قدیمی مثال (مانند الف-۲-۱۳)

NewState وضعیت جدید مثال (مانند الف-۲-۱۳)

الف-۳-۲ UnformattedADUReceived

هدف: به برنامه کاربردی FE نشان می‌دهد که ADU قالب‌بندی نشده دریافت شده است. بعد از دریافت این رویداد، انتهایی-پشتی (BE) به طور خودکار مطلع خواهد شود که برنامه کاربردی FE، به طور رضایت بخشی ADU را دریافت کرده است.

دریافت می‌کند: Instance نمونه‌ای که این نشانه مربوط به آن است

UnformattedMessageLen طول پیام‌های دریافتی

UnformattedMessage خود پیام

الف-۳-۳ ADURequest

هدف: به برنامه کاربردی FE نشان می‌دهد که BE درخواست عنصر(های) خاص را از آن دارد.

دریافت می‌کند: Instance نمونه‌ای که این نشانه مربوط به آن است
Elements عناصری که به BE بازمی‌گردد

الف-۳-۴ ADUReceived

هدف: تحویل عنصر به برنامه کاربردی FE از BE که مطابق با سایر قسمت‌های ISO 17575 ساختاریافته است. پس از دریافت این رویداد به BE به طور خودکار اطلاع داده می‌شود که برنامه کاربردی FE، به طور رضایت بخشی ADU را دریافت کرده است.

دریافت می‌کند: Instance نمونه‌ای که این نشانه مربوط به آن است
Element عنصر دریافت شده

یادآوری- اگر چند عنصر در یک پیام لایه ارتباطات واحد ترکیب شود، نشانه دریافت موفق تنها زمانی که عنصر نهایی تحویل داده شده، ارسال خواهد شد. این موضوع می‌تواند هنگامی که ارتباطات از دست می‌رود، منجر به انتقال دوباره برخی عناصر شود.

الف-۳-۵ ADUSent

هدف: تایید به برنامه کاربردی FE که ADU یا مجموعه ADU توسط BE با موفقیت دریافت شده است

دریافت می‌کند: Instance نمونه‌ای که این نشانه مربوط به آن است

الف-۳-۶ ADUSendOK

هدف: تایید به برنامه کاربردی FE که در حال حاضر ممکن است مجموعه‌ای از عناصر را ارسال کند

دریافت می‌کند: Instance نمونه‌ای که این نشانه مربوط به آن است
CanSend پرچمی که نشان می‌دهد درخواست ارسال پذیرفته شده

الف-۳-۷ SessionRequest

هدف: هشدار به برنامه کاربردی FE که این نمونه، یک درخواست نشست را تشخیص داده است. پارامترهای نشست ممکن است از طریق GetParameter ادعا شود.

دریافت می‌کند: Instance نمونه‌ای که این درخواست را تولید می‌کند
Handle رسیدگی که برای ایجاد نشست استفاده می‌شود

پیوست ب

(الزامی)

پروفرم بیانیه انطباق پیاده‌سازی پروتکل (PICS)

ب-۱ کلیات

این پیوست پروفرم بیانیه‌های انطباق پیاده‌سازی پروتکل (PICS) را شامل می‌شود که در پیاده‌سازی تجهیزات روی برد (OBE) خدمات ارتباطات استفاده می‌شود، همان طور که در بند ۴ و ۵ و ۶ تعریف شده است.

یادآوری- خطمشی‌های یادداشت رسانه خارج از دامنه کاربرد این استاندارد است.

ب-۲ پشتیبانی از تراکنش‌ها

ب-۲-۱ کلیات

این زیربند در پیاده‌سازی‌ها برای API‌های ارتباطات انتهایی جلویی به کار می‌رود، به جداول ب-۱ و ب-۲ مراجعه شود. خدمات ارتباطی مناسب برای انتهایی پشتی نهایی باید بیانیه‌های مشابه با دید انتهایی پشتی را استفاده کند.

ب-۲-۲ پشتیبانی از API پایین

جدول ب-۱ - پشتیبانی از API پایین

وضعیت	توصیف
بله/خیر	API از InitialiseInstance پشتیبانی می‌کند
بله/خیر	از StackID پشتیبانی می‌شود
بله/خیر	Instance handle ارائه خواهد شد
بله/خیر	زمانی که ایجاد نمونه ممکن نیست، نمونه نامعتبر بر می‌گردد
بله/خیر	API از SetParameter پشتیبانی می‌کند
بله/خیر	Instance به کار می‌رود
بله/خیر	ERNoError در صورت موفقیت باز می‌گردد
بله/خیر	در صورت شکست برای تعیین پارامتر، ERNotSet باز می‌گردد
تعداد	بیشینه طول پارامتر که رسیدگی می‌شود
تعداد	بیشینه طول پارامتر رشته‌های مقدار
بله/خیر	پارامتر فراخوانی دنبال‌کننده را ذخیره می‌کند
بله/خیر	API از GetParameter پشتیبانی می‌کند

وضعیت	توصیف
بله/خیر	Instance به کار می‌رود
بله/خیر	Parameter به کار می‌رود
بله/خیر	هنگامی پارامترها معتبر است، Value به عنوان رشته برمی‌گردد
بله/خیر	هنگامی که پارامترها نامعتبر است، یک رشته نامعتبر برمی‌گردد
بله/خیر	API از DeleteParameter پشتیبانی می‌کند
بله/خیر	Instance به کار می‌رود
بله/خیر	Parameter به کار می‌رود
بله/خیر	هنگامی که پارامتر با موفقیت حذف می‌شود، ERNoError باز می‌گردد
بله/خیر	هنگامی که پارامتر با موفقیت حذف می‌شود، ERNotSet باز می‌گردد
بله/خیر	هنگامی که نمونه نامعتبر است، ERNoError باز می‌گردد
بله/خیر	هنگامی که نمونه یافت نمی‌شود/ نامعتبر است، ERNotSet باز می‌گردد
بله/خیر	API از DropIndex پشتیبانی می‌کند
بله/خیر	Instance به کار می‌رود
بله/خیر	Severity در نظر گرفته می‌شود
بله/خیر	هنگامی که نمونه ناشناس ارائه می‌شود، نمونه ERUnknown باز می‌گردد
بله/خیر	هنگامی که شدت SEUnconditional نیست و در وضعیت STNoSession نیست، ERBadState باز می‌گردد
بله/خیر	هنگامی که نمونه با موفقیت افت می‌کند، ERNoError باز می‌گردد
بله/خیر	API از StartSession پشتیبانی می‌کند
بله/خیر	Instance به کار می‌رود
بله/خیر	Reason به کار می‌رود
بله/خیر	SessionHandle به کار می‌رود
بله/خیر	اگر در حال حاضر در یک نشست باشد، ERInSession باز می‌گردد
بله/خیر	اگر نمونه نامعتبر باشد، ERNoInstance باز می‌گردد
بله/خیر	اگر نقطه پایانی پارامتر شده ناشناس باشد، ERUnknownEndpoint باز می‌گردد
بله/خیر	اگر ایجاد نشست به سرعت مردود شود، ERSessionFailed باز می‌گردد
بله/خیر	اگر ایجاد نشست به درستی شروع شود، ERNoError باز می‌گردد
بله/خیر	API از EndSession پشتیبانی می‌کند
بله/خیر	Instance به کار می‌رود
بله/خیر	Reason به کار می‌رود
بله/خیر	اگر نشست STSessionIdle نیست و Reason، RERemoteDrop، ERInSession باز می‌گردد

وضعیت	توصیف
بله/خیر	اگر نمونه نامعتبر باشد، ERNoInstance باز می‌گردد
بله/خیر	اگر EndSession به درستی شروع شود، ERNoError باز می‌گردد
بله/خیر	API از SendUnformattedADU پشتیبانی می‌کند
بله/خیر	Instance به کار می‌رود
بله/خیر	MessageLen به کار می‌رود
بله/خیر	Message در نظر گرفته می‌شود
بله/خیر	اگر نشست در وضعیت STSessionIdle نباشد، ERBadState باز می‌گردد
بله/خیر	اگر نمونه نامعتبر باشد، ERNoInstance باز می‌گردد
بله/خیر	اگر نشست مردود شود، ERSessionFailed باز می‌گردد
بله/خیر	اگر پیشرفت عادی باشد، ERNoError باز می‌گردد
بله/خیر	API از SendADUSetStart پشتیبانی می‌کند
بله/خیر	Instance پشتیبانی می‌شود
بله/خیر	اگر نمونه نامعتبر باشد، ERNoInstance باز می‌گردد
بله/خیر	اگر نشست در وضعیت STSessionIdle نباشد، ERBadState باز می‌گردد
بله/خیر	اگر نشست مردود شود، ERSessionFailed باز می‌گردد
بله/خیر	اگر پیشرفت عادی باشد، ERNoError باز می‌گردد
بله/خیر	API از SendADU پشتیبانی می‌کند
بله/خیر	Instance به کار می‌رود
بله/خیر	ElementLen در نظر گرفته می‌شود
بله/خیر	Element در نظر گرفته می‌شود
بله/خیر	در صورت خطا 0 باز می‌گردد
بله/خیر	در صورتی که خطا وجود نداشته باشد، فضای باقیمانده در بافر انتقال باز می‌گردد
بله/خیر	API از SendADUSetEnd پشتیبانی می‌کند
بله/خیر	Instance در نظر گرفته می‌شود
بله/خیر	اگر نشست در وضعیت STSendingElements نباشد، ERBadState باز می‌گردد
بله/خیر	اگر نشست مردود شود، ERSessionFailed باز می‌گردد
بله/خیر	اگر پیشرفت عادی باشد، ERNoError باز می‌گردد
بله/خیر	API از CommsQuery پشتیبانی می‌کند
بله/خیر	Instance در نظر گرفته می‌شود
بله/خیر	اگر نشست ناشناس باشد، STUnknownInstance باز می‌گردد
بله/خیر	وضعیت فعلی نمونه، به عنوان پیشرفت عادی باز می‌گردد.
بله/خیر	API از StackAvail پشتیبانی می‌کند

وضعیت	توصیف
بله/خیر	Instance در نظر گرفته می شود
بله/خیر	اگر نمونه نامعتبر باشد، FALSE باز می گردد
بله/خیر	اگر پشته در دسترس نباشد، FALSE باز می گردد
بله/خیر	اگر پشته در دسترس باشد، TRUE باز می گردد

ب-۲-۳ پشتیبانی از API بالا

جدول ب-۲- پشتیبانی از API بالا

وضعیت	توصیف
بله/خیر	API از InstanceStateChangeEvent پشتیبانی می کند
بله/خیر	Instance تحویل می شود
بله/خیر	OldState تحویل می شود
بله/خیر	NewState تحویل می شود
بله/خیر	رویداد زمانی که تغییر وضعیت قابل مشاهده خارجی تولید می شود، تولید می شود
بله/خیر	API از UnformattedADUReceivedevent پشتیبانی می کند
بله/خیر	Instance تحویل می شود
بله/خیر	UnformattedMessageLen تحویل می شود
بله/خیر	UnformattedMessage تحویل می شود
بله/خیر	رویداد زمانی که UnformattedMessage دریافت می شود، تولید می شود
بله/خیر	API از ADUReceivedevent پشتیبانی می کند
بله/خیر	Instance تحویل می شود
بله/خیر	Element تحویل می شود
بله/خیر	رویداد زمانی که عنصر دریافت می شود، تولید می شود
بله/خیر	API از ADUSentevent پشتیبانی می کند
بله/خیر	Instance تحویل می شود
بله/خیر	رویداد زمانی که پیام توسط انتهای دور تایید شد، تولید می شود
بله/خیر	API از ADUSendOKevent پشتیبانی می کند
بله/خیر	Instance تحویل می شود
بله/خیر	CanSend تحویل می شود
بله/خیر	رویداد زمانی که انتهای دور، توانایی خود برای دریافت عنصر را نشان می دهد، تولید می شود
بله/خیر	CanSend زمانی که انتهای دور قادر به دریافت عناصر است، TRUE و در غیر این صورت FALSE است.

توصیف	وضعیت
API از SessionRequestevent پشتیبانی می کند	بله/خیر
Instance تحویل می شود	بله/خیر
Handle تحویل می شود	بله/خیر
زمانی که محرک تشخیص داده می شود برای این که نشان دهد انتهای دور می خواهد یک نشست را ایجاد کند، رویداد تولید می شود	بله/خیر

ب-۳ استفاده از پشته های ارتباط

ب-۳-۱ کلیات

این زیربند در انتهای جلویی و انتهای-پشتی به کار می رود، به جدول ب-۳ مراجعه شود.

ب-۳-۲ پشته های ارتباط پشتیبانی شده

جدول ب-۳ - پشته های ارتباط پشتیبانی شده

توصیف	مقدار(ها)
ارتباطات استفاده از TCP / IP V4 را پشتیبانی می کند	بله / خیر
ارتباطات استفاده از TCP / IP V6 را پشتیبانی می کند	بله / خیر
ارتباطات از شماره پورت IP نشات می گیرد	هرگونه/مشخص شود
ارتباطات در شماره پورت IP پایان می یابد	مشخص شود
انتهای جلویی، نام دامنه کاملا واجد شرایط را به عنوان نقطه پایان می پذیرد	بله / خیر
انتهای جلویی، نشست منطقی را به صورت خودکار بعد از اتمام می بندد	به ثانیه مشخص شود

ب-۴ ظرفیت ذخیره سازی انتهای جلویی

ب-۴-۱ کلیات

این زیربند فقط در انتهای جلویی به کار می رود، به جداول ب-۴، ب-۵ و ب-۶ مراجعه شود.

ب-۴-۱ ظرفیت ذخیره‌سازی برای پودمان‌ها^۱ و اطلاعات تماس

جدول ب-۴ - ظرفیت ذخیره‌سازی برای پودمان‌ها و اطلاعات تماس

بیشینه مقدار یا بازه	توصیف
	بیشینه اندازه پیام واحد
	بیشینه تعداد نشست‌های ارتباطی نمونه به صورت موازی
	بیشینه ذخیره‌سازی در دسترس برای صف‌بندی پیام
	بیشینه تعداد گواهی‌نامه کلید / اعتماد در دسترس
	بیشینه تعداد عناصری که می‌تواند در یک تراکنش منتقل شود
	بیشینه اندازه پیام واحد

ب-۴-۲ مقادیر عمومی

جدول ب-۵ - مقادیر عمومی

بیشینه مقدار یا بازه	توصیف
	اعداد صحیح (کمینه و بیشینه مقدار)
	رشته (بیشینه اندازه)
	بیشینه طول پارامتر
	بیشینه طول مقدار پارامتر

ب-۴-۳ امنیت ارتباط

جدول ب-۶ - امنیت ارتباط

وضعیت	توصیف
بله/خیر	همه ارتباطات روی کانال رمزگذاری شده است
تعداد	طول کلید برای رمزگذاری را مشخص کنید
متن	فناوری رمزگذاری مورد استفاده را مشخص کنید
مقارن / نامتقارن	کلید مقارن یا نامتقارن است؟

پیوست پ

(آگاهی‌دهنده)

الزامات API

پ-۱ کلیات

این پیوست، مثالی از API دقیق را ارائه می‌دهد. این پیوست به تشریح الزامات تعریف API ارتباطات می‌پردازد.

پ-۲ الزامات غیر کارکردی

سامانه فرعی ارتباطات به شیوه‌ای استاندارد گرا مورد نیاز است تا به چگونگی ایجاد پیوند بپردازد، داده‌ها روی آن منتقل شده و در پایان تراکنش واضح است.

سامانه فرعی ارتباطات باید هنگامی که اتصال شبکه مناسب در دسترس است و هنگامی که درخواست برای یک نشست فوق العاده است، یک نشست غیر همزمان ایجاد کند.

زیر سامانه ارتباطات API باید مستقل از فناوری ارتباطات باشد. API نیاز به فناوری اساسی را همان طور که در زیربند ۶-۲ ارائه شده، جای می‌دهد.

پ-۳ الزامات عملیاتی

پیوند باید بر مبنای تقاضای برنامه کاربردی انتهایی - جلویی (FE) به انتهایی پشتی (BE) ایجاد شود. وسایلی باید ارائه شود که توسط آن BE بتواند اتصالی را به برنامه کاربردی FE ایجاد کند.

یادآوری- زمان ایجاد اتصال تعریف نشده است (به عنوان مثال FE ممکن است خاموش باشد).

پیوند باید در پارادایم «نشست» سازمان یافته باشد. مراحل مجزایی از فعالیت مربوط به راه اندازی پیوند، بهره برداری و پیاده کردن وجود خواهد داشت.

API باید نشانه‌ای مثبت به برنامه کاربردی FE در مورد وضعیت پیوند فعلی ارائه کند.

قطع موقت پیوند باید برای برنامه کاربردی FE شفاف باشد.

هنگامی که نشست ارتباط از دست می‌رود، نباید به طور خودکار توسط سامانه ارتباطی دوباره ایجاد شود.

API باید سازوکارهایی را ارائه دهد تا به BE اجازه دهد که عناصر اطلاعات خاص (همان طور که در سایر قسمت‌های ISO 17575 تعریف شده و با استفاده از تعاریف آن) از برنامه کاربردی FE درخواست کند.

API باید سازوکارهایی ارائه بدهد تا عناصر اطلاعات خاص (همان طور که در سایر قسمت‌های ISO 17575 تعریف شده و با استفاده از تعاریف آن) را به BE تحویل دهد.

API انتزاعی باید قادر به حمل اطلاعات سازنده خاص باشد که خارج از دامنه کاربرد این استاندارد است.
API باید نشانه‌ای مثبت برای برنامه کاربردی FE ارائه کند که ADU قطعا به BE تحویل شده است.
تمام ارتباطات به و از انتهای از دور باید رمزگذاری شود.

پیوست ت

(آگاهی‌دهنده)

مثال‌هایی از تعاریف برای زبان‌های مناسب

ت-۱ کلیات

API به عمد به زبان بی طرف طراحی شده است، چرا که در محیط‌های مختلف بسیاری استفاده خواهد شد. به طور کلی، توصیه می‌شود API به روشی سازگار با استفاده کلی و اصطلاحات زبان تحت بررسی، پیاده‌سازی شود، در حالی که نویسه کلی و عملیات API را حفظ می‌کند.

مثال زیر که به زبان نرم‌افزاری C نوشته شده نشان می‌دهد که چگونه API مشخص شده با جزئیات در پیوست الف می‌تواند پیاده‌سازی شود. برای هر یک از دستورات فهرست‌شده در پیوست الف، کد C مناسب تعریف می‌شود. همچنین رفتار مورد انتظار ارائه شده و در نمودار وضعیت نشست شکل ۵ نشان داده شده است.

ت-۲ تعریف API در C

```
#ifndef _ISO17575_
#define _ISO17575_
#ifndef BOOL
#define BOOL char
#endif
#ifndef FALSE
#define FALSE 0
#define TRUE !FALSE
#endif
#ifndef BYTE
#define BYTE char
#endif
#ifndef WORD
#define WORD unsigned short int
#endif
// These macros allow us to reuse the Enums in string form for lookup
// Assumes that macros are indexed in order from 0
//
// For any enum AAA, these macros give you;
```

```

// typedef enum { x, y, z } AAAE
// const char *AAAS [] = {"x","y","z"}
// const char *toStringAAA(AAAE x)
#define MAKEstring(a) static const char *a ## S []={a}; const char *toString ##a(a ## E x)
{ return a ##S[x]; }
#define MAKEenum(a) typedef enum {a} a ## E; const char *toString ##a(a ## E x);
#define E(x) x
// Errors returned from the API
#define ISO175752Error \
    E(ERNoError), \
    E(ERNoInstance), \
    E(ERInterfacePreviouslyInitialised), \
    E(ERInsufficientPrivledge), \
    E(ERSessionFailed), \
    E(ERInProgress), \
    E(ERUnknownStack), \
E(ERNotActive), \
    E(ERInSession), \
    E(ERUnknownInstance), \
    E(ERUnknownEndpoint), \
    E(ERNoSession), \
    E(EREndinProgress), \
    E(ERHold), \
    E(ERDropInProgress), \
    E(ERMemoryError), \
    E(ERBadState), \
    E(ERNotSet), \
E(ERUnknownError)
MAKEenum(ISO175752Error);
// Reasons for a callback or drop
#define ISO175752Reason \
    E(REUnknown), \
    E(RENormal), \
    E(RECallback), \

```

```

    E(RERemoteDrop), \
E(RETimed)
MAKEenum(ISO175752Reason);
// Responses to a session state query, or via a state change notification
#define ISO175752State \
    E(STUnknownInstance), \
    E(STNoSession), \
    E(STStarting), \
    E(STSessionIdle), \
    E(STSendingElements), \
    E(STSendingElementsRequest), \
    E(STSendingUnformattedMessage), \
    E(STAwaitingElementsConfirm), \
    E(STSessionRxElements), \
    E(STErrored), \
    E(STEnding)
MAKEenum(ISO175752State);
// Severity of a drop request
#define ISO175752Severity \
    E(SENormal), \
    E(SEUrgent), \
    E(SEUnconditional)
MAKEenum(ISO175752Severity);
// This change in the definition of E is needed for the MAKEstring macro..which is used in
the C file
#undef MAKEenum
#undef E
#define E(x) #x
// Definitions to simplify creation of up API
#define ISO175752APICALLBACKInstanceStateChange(x) void (x)(ISO175752API
*instance,
ISO175752StateE oldState, ISO175752StateE newState)
#define ISO175752APICALLBACKUnformattedADUReceived(x) void (x)(ISO175752API
*instance, WORD
unformattedMessageLen, BYTE *unformattedMessage)

```

```

#define ISO175752APICALLBACKADURequest(x) void (x)(ISO175752API *instance, void
*elementReq)
#define ISO175752APICALLBACKADUReceived(x) void (x)(ISO175752API *instance,
void *element)
#define ISO175752APICALLBACKADUSent(x) void (x)(ISO175752API *instance, BOOL
sent)
#define ISO175752APICALLBACKADUSend(x) void (x)(ISO175752API *instance, BOOL
canSend)
#define ISO175752APICALLBACKSessionRequest(x) void (x)(char *handle);
typedef void ISO175752API;
ISO175752ErrorE GetLastError(ISO175752API *instance);
ISO175752API *InitialiseInstance(char *stackID,
ISO175752APICALLBACKInstanceStateChange(*InstanceStateChangeSet),
ISO175752APICALLBACKUnformattedADUReceived(*UnformattedMessageReceivedSet),
ISO175752APICALLBACKADURequest(*ElementRequestSet),
ISO175752APICALLBACKADUReceived(*ElementReceivedSet),
ISO175752APICALLBACKADUSent(*MessageSentSet),
ISO175752APICALLBACKADUSend(*ElementSendSet),
ISO175752APICALLBACKSessionRequest(*SessionRequestSet));
ISO175752ErrorE DropInstance(ISO175752API *instance,
ISO175752SeverityE severity);
ISO175752ErrorE SetParameter(ISO175752API *instance,
char *param,
char *val);
char *GetParameter(ISO175752API *instance,
char *param);
ISO175752ErrorE DeleteParameter(ISO175752API *instance,
char *param);
ISO175752ErrorE StartSession(ISO175752API *instance,
ISO175752ReasonE reason,
char *handle);
ISO175752ErrorE EndSession(ISO175752API *instance,
ISO175752ReasonE reason);
ISO175752ErrorE SendUnformattedADU(ISO175752API *instance,
WORD messageLen,

```

```
BYTE *message);  
ISO175752ErrorE SendADUSetStart(ISO175752API *instance);  
WORD SendADU(ISO175752API *instance,  
WORD messageLen,  
BYTE *message);  
ISO175752ErrorE SendADUSetEnd(ISO175752API *instance);  
ISO175752StateE CommsQuery(ISO175752API *instance);  
BOOL StackAvail(ISO175752API *instance);  
#endif
```


پیوست ث

(آگاهی‌دهنده)

استفاده از این استاندارد برای EETS

ث-۱ کلیات

ت-۱ کلیات

در سال ۲۰۰۴ بخشنامه اتحادیه اروپا به شماره 2004/52/EC شورای پارلمان اروپای «در مورد قابلیت همکاری سامانه‌های الکترونیکی عوارض جاده در جامعه» به تصویب رسید. این بخشنامه اروپا برای ایجاد خدمت عوارض الکترونیکی اروپا (EETS) فراخوانی شد.

در سال ۲۰۰۹ تصمیم 2009/750/EC EC «در تعریف خدمت عوارض الکترونیکی اروپا و عناصر فنی آن» به تصویب رسید. این تصمیم ویژگی‌های فنی لازم و الزامات برای این منظور و قواعد قراردادی مربوط به ارائه EETS را تعیین می‌کرد. این تصمیم بر پایه حقوق و تعهدات در مورد ارائه دهندگان EETS، شارژکنندگان عوارض و کاربران EETS است.

سایر الزامات و سایر بخشنامه‌های اتحادیه اروپا ممکن است برای محصول (های) خارج از دامنه کاربرد این استاندارد نیز کاربردپذیر باشد.

ث-۲ رابطه کلی بین استانداردسازی اروپا و EETS

بخشنامه اتحادیه اروپا 2004/52/EC همچنین باعث ایجاد ماموریت در استانداردسازی (M / 338)، «ماموریت استانداردسازی برای کمیته استانداردسازی اروپا (CEN)»، CENELEC و ETSI در پشتیبانی از ارتباطات متقابل سامانه عوارض جاده الکترونیکی در جامعه» شد که برای توسعه استانداردهای فنی در پشتیبانی از EETS خوانده می‌شود. فعالیت‌های تحت M/338 توسط «گروه خود هماهنگی ITS نظارت می‌شود» (ITS-CG پیش از این ICTSB / ITSSG بوده است)

M/338 به صراحت برای ارائه استانداردهای هماهنگ خوانده نمی‌شود (با توجه به بخشنامه 98/34/EC در رویکرد جدید برای هماهنگی فنی و استانداردها) که بدین معنی است که این احتمال برای استانداردهای اروپایی که در پشتیبانی از EETS تدوین شده اند، در دسترس نیست. در عوض این پیوست کوتاه، طرح این استاندارد که می‌تواند در زمینه EETS استفاده شود را ارائه می‌کند.

تصمیم‌های EC به استفاده از استانداردهای خاص، حتی اگر آنها به طور رسمی هماهنگ نباشد، اشاره دارد. این موضوع همچنین در تصمیم‌های 2009/750/EC EC برای تعداد کمی استاندارد انجام می‌شود (به عنوان

مثال استانداردهایی که در زمان تصویب آن در دسترس بوده است). در موردی که تصمیم‌های EC بیشتری وجود داشته باشد، استانداردهای اروپایی بیشتری می‌تواند اشاره شود.

در سال ۲۰۱۱ کمیسیون اروپا نیز «راهنمای کاربرد بخشنامه قابلیت همکاری سامانه‌های عوارض جاده الکترونیکی» (ISBN 978-92-79-18637-0) را منتشر کرد. این راهنما یک کتابچه راهنمای مرجع برای تمام طرف‌های مستقیم یا غیر مستقیم 2004/52/EC و تصمیم 2009/750/EC در نظر گرفته است. هدف آن کمکی برای پیاده‌سازی EETS، از جمله فهرست استانداردهایی است که می‌تواند استفاده شود. راهنما تنها آگاهی‌دهنده است (به عنوان مثال مستند نمی‌تواند مشخص کند که استانداردهای خاصی برای استفاده در EETS «اجباری» است) و این طور در نظر گرفته شده بود که به صورت منظم به روز شود.

ث-۳ فعالیت استانداردسازی اروپا که از EETS پشتیبانی می‌کند

بسیاری از استانداردهای تدوین‌شده توسط CEN / TC278 با مد نظر قرار دادن الزامات EETS پیش‌نویس شده است (از جمله استفاده از نتایج حاصل از پروژه‌های اروپایی مانند CARDME، PISTA، CESARE و RCI). نمایندگان CEN به عنوان ناظر در گروه‌های کاری و غیره شرکت کردند که این کار توسط EC برای EETS آغاز شد. از این رو، برخی کارها در همکاری نزدیک بین گروه‌های کاری CEN و EC انجام شده است.

لازم به ذکر است که هیچ یک از استانداردهای CEN / ISO راه حل «کلیدی» برای EETS نیست. این استانداردها برای استفاده به عنوان «بلوک‌های ساختاری» برای EETS، پشتیبانی از چارچوب قانونی^۱ EETS و توافقات بین طرف‌های در نظر گرفته شده EETS است. ویژگی EETS دقیق در دامنه کاربرد استانداردهای CEN / ISO نیست، اما وظیفه مالک (های) EETS باقی می‌ماند.

همچنین باید توجه داشت که CEN/ISO دامنه گسترده‌تری از EETS دارد که یک خدمت مکمل به خدمات ملی اعضا و اختیاری برای کاربران است، در حالی که توصیه می‌شود استانداردهای CEN/ISO برای تمام خدمات EFC در سراسر جهان کاربردپذیر شود.

ث-۴ تناظر بین این استاندارد و EETS

این استاندارد نحوه اتصال به تجهیزات روی برد (OBE) نرم‌افزار خاص عوارض و همچنین پیشکار یا انتهایی پستی به لایه‌های پایین‌تر از هر کانال ارتباطی سیار را تعریف می‌کند تا نرم‌افزار عوارض را از خصوصیات خاص کانال ارتباطی انتزاعی کند. با این استاندارد، نرم‌افزار خاص عوارض می‌تواند توسعه یافته و به طور مستقل از کانال ارتباطی در دسترس واقعی عمل کند.

این استاندارد از الزامات عمومی پشتیبانی می‌کند تا سامانه‌های عوارض GNSS / بر اساس CN را در تمام مناطقی که در آن کمینه یک نوع از شبکه ارتباطات سیار در دسترس است، مجاز بداند. این برنامه همچنین

1- Legal

از OBE پشتیبانی می‌کند، ارتباط ارائه‌دهنده خدمت عوارض از طریق شبکه‌های ارتباطی سیار مختلف در هنگام رومینگ در سراسر مناطق یا در چند نسل فناوری ارتباطات وجود دارد، در حالی که این موضوع برای نرم‌افزار خاص عوارض شفاف است.

با این حال، واسطه‌های ارتباطی تعریف‌شده در این استاندارد در مدل نقش EETS قابل مشاهده نیست و از این رو، هیچ پیوند مستقیمی به الزامات تصمیم EC/750/2009 EC وجود ندارد.

کتابنامه

- [1] ISO/IEC 9646-7, Information technology — Open Systems Interconnection — Conformance testing methodology and framework — Part 7: Implementation Conformance Statements
- [2] ISO/IEC 8824-1, Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation
- [3] ISO 14906:2011/Amd1:2015, Electronic fee collection — Application interface definition for dedicated short-range communication
- [4] ISO 17573:2010, Electronic fee collection — Systems architecture for vehicle-related tolling
- [5] ISO 17575-1:2016, Electronic fee collection — Application interface definition for autonomous systems — Part 1: Charging
- [6] ISO 17575-3:2016, Electronic fee collection — Application interface definition for autonomous systems — Part 3: Context data
- [7] Directive 2004/52/EC of the European Parliament and of the Council of 29 April 2004 on the interoperability of electronic road toll systems in the Community. OJ L. 2004, 166 pp. 124–143
- [8] 2009/750/EC, Commission Decision of 6 October 2009 on the definition of the European Electronic Toll Service and its technical elements (notified under document C(2009) 7547). OJ L. 2009, 268 pp. 11–29
- [9] EC – DG for mobility and transport. Guide for the application of the directive on the interoperability of electronic road toll systems, ISBN 978-92-18637-0, 2011