

# EN 12834 - Vyhrazené spojení krátkého dosahu (DSRC) - Aplikační vrstva

**Aplikační oblast:** [Vyhrazené spojení krátkého dosahu \(DSRC\)](#)  
**Rok vydání normy a počet stran:** Vydána 2004, 44 stran  
**Zavedení normy do ČSN:** vyhlášením  
**Rok zpracování extraktu:** 2008  
**Skupina témat:** Vyhrazené spojení krátkého dosahu (DSRC)  
**Téma normy:** Definování aplikační vrstvy v jednotkách OBU a RSU  
**Charakteristika tématu:** Definuje požadavky na aplikační vrstvu obou jednotek v rámci DSRC komunikace

|                                                               |
|---------------------------------------------------------------|
| Úvod, vysvětlení východisek                                   |
| Obecný popis DSRC                                             |
| Popis architektury, hierarchie, rolí a vztahů objektů         |
| Popis procesu / funkce / způsobu použití                      |
| Definuje popis komunikace mezi jednotkami, sekvence, kódování |
| Popis rozhraní / API / struktury systému                      |
| Definice protokolu / algoritmu / výpočtu                      |
| Popisuje protokol pro obousměrnou komunikaci mezi jednotkami  |
| Definice reprezentace dat / fyzikálního významu               |
| Definice konstant / rozsahů / omezení                         |

## Úvod

Cílem této normy je specifikace linkové vrstvy pro DSRC na frekvenci 5,8 GHz pro aplikace v oblasti dopravní telematiky.

Jedná se o druh komunikace, který má v prostředí ČR, ale i v zahraničí, velkou budoucnost. Základní aplikací, pro kterou byla navržena, je elektronický výběr poplatků (EFC). Trendem však je, aby jediné zařízení bylo možné použít i pro další aplikace, například inteligentní značky (dopravní značka zasílá informaci o maximální povolené rychlosti, nebezpečí, či další pomocí DSRC přímo do vozidla), řízení vjezdu na parkoviště a mnohé další.

Poznámka: Extrakt uvádí vybrané kapitoly popisovaného dokumentu a přejímá původní číslování kapitol.

## Užití

Pro zajištění interoperability mezi zařízeními je skutečně nezbytné, aby byly všechny tyto normy implementovány. Proto se tato norma týká především dodavatelů technologií. Tato konkrétní norma potom definuje parametry a vlastnosti aplikační vrstvy. **Dodavatelé technologie** musí zajistit, že jejich zařízení odpovídá definicím uvedeným v této skupině norem. Jen tak je možné zajistit budoucí interoperabilitu zařízení (pokud odpovídají i ostatní vrstvy modelu ISO/OSI). Velmi detailní znalost této skupiny norem je proto nezbytná. **Orgány státní správy** musí pro každou veřejnou zakázku v oblasti dopravní telematiky zahrnující komunikaci mezi RSU a OBU vyžadovat, aby dodávaná technologie odpovídala těmto normám.

## 1. Předmět normy

Tato norma popisuje aplikační vrstvu pevných zařízení ([RSU](#)) i mobilních jednotek ([OBU](#)). Ta je klíčovou pro zajištění interoperability mezi zařízeními. Pokud není fyzická vrstva řešena jednotně, nejedná se o problém, který se dá vyřešit přehráním software. Z tohoto důvodu je tato norma určena především dodavatelům technologie.

Tato norma se věnuje mimo jiné následujícím oblastem:

- struktura aplikační vrstvy;
- služby povolující datové přenosy a vzdálený provoz;
- základní chování aplikační vrstvy a parametry ([fragmentace](#), multiplexování, kódování a další);
- procedury pro inicializaci a release;
- služby v oblasti vysílání (broadcast);
- a další.

## 2. Související normy

Tato norma je součástí souboru norem, které definují rámec pro vyhrazenou komunikaci krátkého dosahu v oblasti dopravní telematiky. Těmito souvisejícími normami jsou:

EN [12795](#) Dopravní telematika – DSRC – Linková vrstva: přístupové prostředky a logická kontrola spojení

### 3. Termíny a definice

Kapitola uvádí 17 termínů a definic, v tomto extraktu jsou uvedeny pouze ty nutné pro jeho pochopení.

**3.1 aplikace** (*application*) množina procesů včetně relevantních funkcí a strukturovaných dat, které využívají služby [DSRC](#) komunikace

**3.4 BST** (*Beacon Service Table*) datová struktura vysílaná [RSU](#) a indikující dostupné služby

**3.13 multiplexování** (*multiplexing*) [funkce](#) transportního jádra umožňující současnou podporu více než jedné aplikace v jedné [OBU](#)

Další termíny a zkratky z oboru ITS jsou obsaženy ve [slovníku ITS terminology](#).

### 4. Symboly a zkratky

Tato kapitola obsahuje 28 zkratk, z nichž podstatné jsou uvedeny níže.

**4.1 APDU-** (*Application protocol [data](#) unit*) protokol datové jednotky aplikační

**4.2 ASDU-** (*Application service [data](#) unit*) služby datové jednotky aplikační

**4.6 DSRC-**(*Dedicated Short Range Communication*) dedikovaná komunikace krátkého dosahu

**4.11 T-APDU-** (*Transfer application protocol [data](#) unit*) přenosový protokol datové jednotky aplikační

**4.19 OBU-** (*On-Board Unit*) [palubní jednotka](#) – mobilní [zařízení](#) instalované ve vozidle

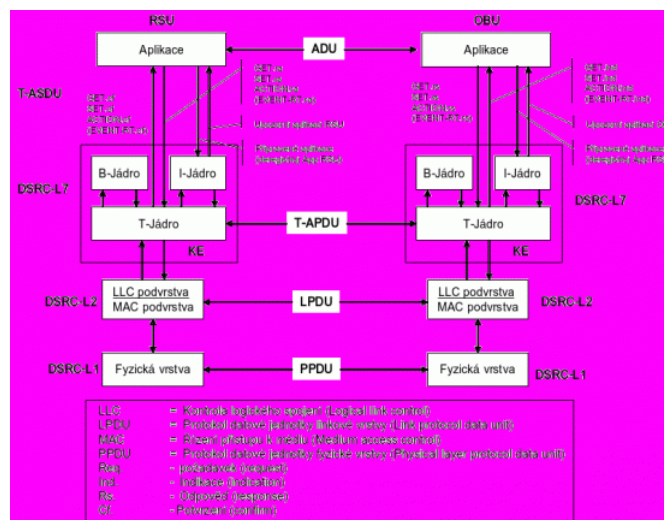
**4.22 RSU-** (*Road Side Unit*) [jednotka na straně infrastruktury](#) – pevné [zařízení](#) instalované v rámci infrastruktury

**4.26 VST-** (*Vehicle Service Table*) servisní tabulka [vozidla](#)

Další termíny a zkratky z oboru ITS jsou obsaženy ve slovníku ITS terminology ( [www.ITsterminology.org](http://www.ITsterminology.org) ).

## Kapitola 5 Struktura jádra aplikační vrstvy

V této kapitole je popsána struktura jádra aplikační vrstvy. Ta se sestává z transferového (přenosového) jádra T (T-Kernel) a buď z inicializačního jádra (I-Kernel), broadcastového (vysílacího) jádra (B-Kernel), či z obou. Obrázek 2 zobrazuje tato jádra a jejich vztah k externím blokům.



Obrázek 3 – Kontext a struktura jádra aplikační vrstvy

## Kapitola 6 Transferové jádro

T-Kernel přenáší informace mezi oběma jádry (I a B) nebo aplikacemi a za úkol má abstrahovat vlastní realizaci přenosu.

Článek 6.2 shrnuje základní funkce, které tato vrstva nabízí pro zajištění přenosu. Základní nabízené služby jsou GET, SET, ACTION, EVENT-REPORT a INITIALISATION, pro které jsou definovány funkce jako v následujícím příkladu pro službu GET:

- GET.request

- GET.indication
- GET.response
- GET.confirm

Tento článek definuje jejich formát (6.2.3) i parametry (6.2.4).

Článek 6.3 popisuje chování transférového jádra. Přenos se skládá z následujících kroků:

- překlad SDU na PDU;
- zakódování PDU;
- **fragmentace**;
- rozdělení do oktetů (octet alignment);
- multiplexování, zřetězení (concatenation), a přístup k LLC;
- demultiplexování;
- defragmentace;
- dekódování PDU, deřetězení a odebrání vložených bitů;
- překlad PDU na SDU a distribuce adresátovi.

Všechny tyto kroky a jejich chování jsou popsány detailně v jednotlivých podkapitolách.

## Kapitola 7 Inicializační jádro

Podobně jako v předchozí kapitole jsou zde definovány jednotlivé služby a základní funkce tohoto jádra. Jedná se především o:

- RegisterApplicationRSU (registruj aplikaci RSU);
- RegisterApplicationOBU (registruj aplikaci OBU);
- DeregisterApplication (odhlaš aplikaci);
- NotifyApplicationOBU (upozorni aplikaci OBU na přítomnost potenciálního partnera pro komunikaci);
- NotifyApplicationRSU (upozorni aplikaci RSU na přítomnost potenciálního partnera pro komunikaci);
- EndApplication (ukonči aplikaci).

Tento článek definuje jejich formát (7.2.3) i parametry (7.2.4).

Článek 7.3 popisuje chování těchto základních funkcí (popis včetně logických toků):

- opakované vysílání BST;
- příjem BST a vyslání **VST**;
- odpověď na **VST**;
- registrace aplikace RSU;
- registrace aplikace OBU;
- deregistrace aplikace na OBU;
- deregistrace aplikace na RSU;
- RSU: Vydání (release) aplikace;
- OBU: Příjem nového releasu.

## Kapitola 8 Broadcastové jádro

V této kapitole jsou podobně popsány funkce a služby broadcastového jádra. Jedná se o:

- BroadcastData (vysílej data);
- GetBroadcastData (přijmi vysílaná data).

Více podrobností je uvedeno přímo v normě.

**Příloha A (normativní) Datové struktury**

Tato příloha popisuje datové struktury v notaci ASN.1. Příklad (výběr) je uveden níže:

```
DSRCDData {iso(1) standard(0) iso$$($$)} DEFINITIONS::= BEGIN
IMPORTS
Container.J.y FROM Application.J -- this line shall be given for each application
-- which defines data of type container, J and y
-- shall be replaced by an unambiguous suffix;
Record.J.y FROM Application.J -- this line shall be given for each application
-- which defines data of type record, J and y
-- shall be replaced by an unambiguous suffix;

-- EXPORTS everything;
Action-Request::= SEQUENCE{
mode BOOLEAN,
eid Dsrc-EID,
actionType ActionType,
accessCredentials OCTET STRING (SIZE (0..127,...)) OPTIONAL,
actionParameter Container OPTIONAL,
iid Dsrc-EID OPTIONAL
}
Action-Response::= SEQUENCE{
fill BIT STRING (SIZE(1)),
eid Dsrc-EID,
iid Dsrc-EID OPTIONAL,
responseParameter Container OPTIONAL,
ret ReturnStatus OPTIONAL
}
```

**Příloha B (normativní) Pojmenovávání a registrace**

Tato příloha se věnuje pojmenovávání a registraci komponent.

**Příloha C (informativní) Příklad**

Tato příloha uvádí příklad obsahu aplikační vrstvy pro aplikaci elektronického mýta (EFC).

**Příloha D (informativní) Odchylky typu A**

Tato příloha obsahuje národní odchylku typu A, kterou do normy prosadila Itálie.

**Související termíny**

- [aplikační vrstva](#)
- [spojování](#)
- [soubor vysílání](#)
- [slučování](#)
- [počátek fronty](#)
- [operace](#)
- [multiplex](#)
- [management](#)
- [identifikátor prvku](#)
- [identifikátor atributu](#)
- [fragmentace](#)
- [čas](#)
- [tabulka služeb vozidla](#)