



1
2

3 **Linux based 3G Specification**

4 **Multimedia Mobile Phone API**

5

6 **Packet Switched Communication Service**

7

8 Document: CELF_MPP_PS_FR1_20060301

9

10

11

12

13

14

15

16

17

18

19

20

WARNING : This is a working draft for review only, it is NOT a published specification of the CE Linux Forum. It is likely that further substantial changes will be made in the course of review and issue resolution. Send comments on this version to: MppApiComments@tree.celinuxforum.org

21 **Revision History**

Revision	Comment	Reviewer	Editor	Date
1.0	Initial		NEC/Panasonic	05/09/28
1.0.1	Editorial changes		John Mehaffey	05/09/28
1.0.2	Editorial changes		Christophe Guinet	05/10/24
1.0.3	Editorial changes		Christophe Guinet	05/11/03
2.2.1	Version change and new logo		Scott Preece	05/11/05
2.2.2	Updated with comments received during November face-to-face meeting in San Francisco	CELF MPPWG	Christophe Guinet	05/11/16
2.2.3	Updated with comments received during review	CELF MPPWG	Christophe Guinet	05/12/23
FR1	Version with line numbers for Formal Review		Scott Preece	2006.3.1

22

23

24

25	0. INTRODUCTION	6
26	0.1 REFERENCES.....	6
27	0.1.1 Normative	6
28	0.1.2 Informative	6
29	1. PRIMITIVES	7
30	1.1 CONSTANTS	7
31	1.1.1 CELF_MP_PS_HOST_MAX	7
32	1.1.2 CELF_MP_PS_APN_MAX	7
33	1.1.3 CELF_MP_PS_TITLE_MAX	7
34	1.1.4 CELF_MP_PS_DNS_RECORD_MAX	7
35	1.2 ENUMS.....	7
36	1.2.1 CelfMpPsEvent.....	7
37	1.2.2 CelfMpPsApnId	7
38	1.2.3 CelfMpPsApnInit	8
39	1.2.4 CelfMpPsConn	8
40	1.2.5 CelfMpPsApnMode	8
41	1.2.6 CelfMpPsExtEventSet.....	8
42	1.2.7 CelfMpPsPdpType.....	8
43	1.2.8 CelfMpPsEventSet.....	8
44	1.2.9 CelfMpPsUPlane.....	8
45	1.2.10 CelfMpPsStatus	8
46	1.3 DATA TYPES AND STRUCTURES	9
47	1.3.1 CelfMpPsHost.....	9
48	1.3.2 CelfMpPsApn.....	9
49	1.3.3 CelfMpPsDnsRec.....	9
50	1.3.4 CelfMpPsTitle.....	9
51	1.3.5 CelfMpPsApnHostData.....	9
52	1.4 EVENTS TYPE.....	9
53	1.4.1 Packet switched call state notification event.....	9
54	1.4.2 Packet switched service status notification event.....	10
55	1.4.3 Packet switched call activity notification event.....	10
56	1.4.4 External equipment status notification event.....	11
57	2. START PACKET SWITCHED COMMUNICATION NOTIFICATION.....	12
58	2.1 SYMBOL: CELF_MP_PS_NOTIFICATION_START	12
59	2.1.1 Syntax.....	12
60	2.1.2 Argument	12
61	2.1.3 Return Value	12
62	2.1.4 Include File.....	12
63	2.1.5 Functional Description	12
64	3. STOP PACKET SWITCHED COMMUNICATION NOTIFICATION.....	14
65	3.1 SYMBOL: CELF_MP_PS_NOTIFICATION_STOP	14
66	3.1.1 Syntax.....	14
67	3.1.2 Argument	14
68	3.1.3 Return Value	14
69	3.1.4 Include File.....	14
70	3.1.5 Functional Description	14
71	4. GET PACKET SWITCHED COMMUNICATION STATUS	15
72	4.1 SYMBOL: CELF_MP_PS_COM_STATUS_GET	15
73	4.1.1 Syntax.....	15
74	4.1.2 Argument	15
75	4.1.3 Return Value	15
76	4.1.4 Include File.....	15
77	4.1.5 Functional Description	15
78	5. REQUEST PACKET SWITCHED COMMUNICATION CONNECTION	16
79	5.1 SYMBOL: CELF_MP_PS_CONNECT	16
80	5.1.1 Syntax.....	16

Classification: Packet Switched Communication Service

81	5.1.2	Argument	16
82	5.1.3	Return Value	16
83	5.1.4	Include File	16
84	5.1.5	Functional Description	16
85	6.	RESPONSE TO A NETWORK REQUEST FOR PACKET SWITCHED CONNECTION	17
86	6.1	SYMBOL: CELF_MP_PS_CONNECT_RSP	17
87	6.1.1	Syntax	17
88	6.1.2	Argument	17
89	6.1.3	Return Value	17
90	6.1.4	Include File	17
91	6.1.5	Functional Description	17
92	7.	REQUEST THE DEACTIVATION OF PACKET SWITCHED COMMUNICATION	18
93	7.1	SYMBOL: CELF_MP_PS_CALL_DISCONNECT	18
94	7.1.1	Syntax	18
95	7.1.2	Argument	18
96	7.1.3	Return Value	18
97	7.1.4	Include File	18
98	7.1.5	Functional Description	18
99	8.	SPECIFY APN SETTINGS	19
100	8.1	SYMBOL: CELF_MP_PS_APN_SET	19
101	8.1.1	Syntax	19
102	8.1.2	Argument	19
103	8.1.3	Return Value	19
104	8.1.4	Include File	19
105	8.1.5	Functional Description	20
106	9.	GET APN SETTINGS	21
107	9.1	SYMBOL: CELF_MP_PS_APN_GET	21
108	9.1.1	Syntax	21
109	9.1.2	Argument	21
110	9.1.3	Return Value	21
111	9.1.4	Include File	21
112	9.1.5	Functional Description	21
113	10.	SELECT APN FOR PACKET SWITCHED COMMUNICATION	22
114	10.1	SYMBOL: CELF_MP_PS_APN_SELECT	22
115	10.1.1	Syntax	22
116	10.1.2	Argument	22
117	10.1.3	Return Value	22
118	10.1.4	Include File	22
119	10.1.5	Functional Description	22
120	11.	GET APN USED FOR PACKET SWITCHED COMMUNICATION	23
121	11.1	SYMBOL: CELF_MP_PS_APN_ID_GET	23
122	11.1.1	Syntax	23
123	11.1.2	Argument	23
124	11.1.3	Return Value	23
125	11.1.4	Include File	23
126	11.1.5	Functional Description	23
127	12.	FORCE DEACTIVATION OF A PACKET COMMUNICATION	24
128	12.1	SYMBOL: CELF_MP_PS_FORCE_DISCONNECT	24
129	12.1.1	Syntax	24
130	12.1.2	Argument	24
131	12.1.3	Return Value	24
132	12.1.4	Include File	24
133	12.1.5	Functional Description	24
134	13.	GET USER PLANE CONNECTION STATUS	25
135	13.1	SYMBOL: CELF_MP_PS_UPLANE_STATUS_GET	25
136	13.1.1	Syntax	25
137	13.1.2	Argument	25
138	13.1.3	Return Value	25

Classification: Packet Switched Communication Service

139	13.1.4	Include File.....	25
140	13.1.5	Functional Description	25
141	14.	GET DNS RECORD NAME.....	26
142	14.1	SYMBOL: CELF_MP_PS_DNS_RECORD_GET	26
143	14.1.1	Syntax.....	26
144	14.1.2	Argument	26
145	14.1.3	Return Value.....	26
146	14.1.4	Include File.....	26
147	14.1.5	Functional Description	26
148	15.	GET DEFAULT APN MODE.....	27
149	15.1	SYMBOL: CELF_MP_PS_APN_MODE_GET.....	27
150	15.1.1	Syntax.....	27
151	15.1.2	Argument	27
152	15.1.3	Return Value.....	27
153	15.1.4	Include File.....	27
154	15.1.5	Functional Description	27
155	16.	FORCE DEACTIVATION OF ALL PACKET SWITCHED COMMUNICATIONS	28
156	16.1	SYMBOL: CELF_MP_PS_SERVICE_SHUTDOWN	28
157	16.1.1	Syntax.....	28
158	16.1.2	Argument	28
159	16.1.3	Return Value.....	28
160	16.1.4	Include File.....	28
161	16.1.5	Functional Description	28
162	17.	REQUEST APN INITIALIZATION	29
163	17.1	SYMBOL: CELF_MP_PS_APN_INITIALIZE.....	29
164	17.1.1	Syntax.....	29
165	17.1.2	Argument	29
166	17.1.3	Return Value.....	29
167	17.1.4	Include File.....	29
168	17.1.5	Functional Description	29
169	18.	START MONITORING EXTERNAL EQUIPMENT FOR PACKET SWITCHED	
170	COMMUNICATION	30	
171	18.1	SYMBOL: CELF_MP_PS_EXT_DEVICE_NOTIFICATION_START.....	30
172	18.1.1	Syntax.....	30
173	18.1.2	Argument	30
174	18.1.3	Return Value.....	30
175	18.1.4	Include File.....	30
176	18.1.5	Functional Description	30
177	19.	STOP MONITORING EXTERNAL EQUIPMENT FOR PACKET SWITCHED	
178	COMMUNICATION	31	
179	19.1	SYMBOL: CELF_MP_PS_EXT_DEVICE_NOTIFICATION_STOP	31
180	19.1.1	Syntax.....	31
181	19.1.2	Argument	31
182	19.1.3	Return Value.....	31
183	19.1.4	Include File.....	31
184	19.1.5	Functional Description	31
185			

186

187

0. Introduction

188

Packet Switched Communication Service (PS Service) has the function of the packet call control and the packet data sending and receiving.

189

190

Packet Switched Communication Service includes PPP dial-up communication service and IP connection data transfer service.

191

192

193

0.1 References

194

0.1.1 Normative

195

RFC 2119: "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997, URL: <http://www.ietf.org/rfc/rfc2119.txt>

196

197

198

RFC 2234: "Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell. November 1997, URL: <http://www.ietf.org/rfc/rfc2234.txt>

199

200

201

0.1.2 Informative

202

3GPP TS 23.060 General Packet Radio Service (GPRS); Service Description; Stage 2
URL: http://www.3gpp.org/ftp/Specs/2005-09/Rel-5/23_series/23060-5b0.zip

203

204

205

1. Primitives

206

1.1 Constants

207

1.1.1 CELF_MP_PS_HOST_MAX

208

Description: Maximum length of the character string for the ISP (Internet Service Provider)

209

URL

210

211

1.1.2 CELF_MP_PS_APN_MAX

212

Description: Maximum length of the character string for the APN

213

214

1.1.3 CELF_MP_PS_TITLE_MAX

215

Description: Maximum length of the character string for the APN title string

216

217

1.1.4 CELF_MP_PS_DNS_RECORD_MAX

218

Description: Maximum length of the character string for the DNS record

219

220

1.2 Enums

221

1.2.1 CelfMpPsEvent

222

Description: APN number

223

Definition:

224

CELF_MP_PS_RECEPTION_START_IND

225

CELF_MP_PS_EMISSION_START_IND

226

CELF_MP_PS_RECEPTION_REJ_IND

227

CELF_MP_PS_EMISSION_REJ_IND

228

CELF_MP_PS_CONNECT_START_IND

229

CELF_MP_PS_DISCONNECT_REQ

230

CELF_MP_PS_DEACTIVATE_CNF

231

CELF_MP_PS_DEACTIVATE_IND

232

CELF_MP_PS_EMISSION_SETUP_RSP

233

CELF_MP_PS_CALLED_SETUP_RSP

234

CELF_MP_PS_CALLED_REJ_REQ

235

CELF_MP_PS_CALLED_END_IND

236

CELF_MP_PS_SERVICE_RING

237

CELF_MP_PS_SERVICE_ERROR

238

CELF_MP_PS_SERVICE_OK

239

CELF_MP_PS_CONTROL_DISCONNECT

240

CELF_MP_PS_DATA_SEND_IND

241

CELF_MP_PS_DATA_RECV_IND

242

CELF_MP_PS_APN_INIT_END_IND

243

244

1.2.2 CelfMpPsApnId

245

Description: APN number

246

Definition:

247

CELF_MP_PS_APN_DEFAULT : Default setting

248

CELF_MP_PS_APN_USERSET1 : User-specified 1

249

CELF_MP_PS_APN_USERSET2 : User-specified 2

250

CELF_MP_PS_APN_USERSET3 : User-specified 3

251

CELF_MP_PS_APN_USERSET4 : User-specified 4

252

CELF_MP_PS_APN_USERSET5 : User-specified 5

253

CELF_MP_PS_APN_USERSET6 : User-specified 6

254

CELF_MP_PS_APN_USERSET7 : User-specified 7

255

CELF_MP_PS_APN_USERSET8 : User-specified 8

256

CELF_MP_PS_APN_USERSET9 : User-specified 9

257

CELF_MP_PS_APN_USERSET10 : User-specified 10

258

259
260
261
262
263
264
265

266
267
268
269
270
271

272
273
274
275
276
277

278
279
280
281
282
283

284
285
286
287
288
289
290
291
292

293
294
295
296
297
298
299
300
301
302

303
304
305
306
307
308

309
310
311
312
313
314

1.2.3 CelfMpPsApnInit

Description: result of the APN initialization

Definition:

CELF_MP_PS_UIM_UPDATED : Successful, UIM updated
 CELF_MP_PS_UIM_NOT_UPDATED : Successful, UIM not updated
 CELF_MP_PS_AUTH_ERR : Failure returned from the authentication service

1.2.4 CelfMpPsConn

Description: accept or reject an incoming PS connection

Definition:

CELF_MP_PS_COM_CONN_ACCEPT : Accept incoming PS connection
 CELF_MP_PS_COM_CONN_REJECT : Reject incoming PS connection

1.2.5 CelfMpPsApnMode

Description: default APN settings

Definition:

CELF_MP_PS_DEFAULT_APN_MS : APN settings stored in the mobile phone
 CELF_MP_PS_DEFAULT_APN_UIM : APN settings stored in the UIM

1.2.6 CelfMpPsExtEventSet

Description: External Equipment Event Notification class

Definition:

CELF_MP_PS_EXT_EVENT_APN_INIT_END : Notification of APN initialization completed
 CELF_MP_PS_EXT_EVENT_CLASS_ALL : All

1.2.7 CelfMpPsPdpType

Description: PDP type

Definition:

CELF_MP_PS_PDP_TYPE_X25 : X.25 connection
 CELF_MP_PS_PDP_TYPE_IP : IP connection
 CELF_MP_PS_PDP_TYPE_OSPIH : OSPIH connection
 CELF_MP_PS_PDP_TYPE_PPP : PPP (IP based PPP)
 CELF_MP_PS_PDP_TYPE_IPv6 : IPv6 connection

1.2.8 CelfMpPsEventSet

Description: Event notification class

Definition:

CELF_MP_PS_CLASS_CALL_STATE_MSG : PS call status message notification (except for data emission / reception)
 CELF_MP_PS_CLASS_SERVICE_STATE_MSG : PS service status message notification
 CELF_MP_PS_CLASS_CALL_DATA_MSG : PS call activity (data emission / reception) message notification
 CELF_MP_PS_CLASS_ALL : All

1.2.9 CelfMpPsUPlane

Description: User Plane connection status

Definition:

CELF_MP_PS_UPLANE_ON : U-Plane connected
 CELF_MP_PS_UPLANE_OFF : U-Plane not connected

1.2.10 CelfMpPsStatus

Description: status of the current PS communication

Definition:

CELF_MP_PS_STATUS_EMPTY : Communication ended (idle)
 CELF_MP_PS_STATUS_ACCEP_CONN : Connected for incoming call
 CELF_MP_PS_STATUS_CONNECT : Connected for outgoing call

Classification: Packet Switched Communication Service

315 CELF_MP_PS_STATUS_ACTIVE : Communication active (C-Plane active)
 316 CELF_MP_PS_STATUS_DISCONNECT : Disconnected
 317

318 **1.3 Data Types and Structures**

319 **1.3.1 CelfMpPsHost**

320 **Description:** It contains the ISP (Internet Service Provider) URL string terminated by a NULL
 321 character '\0'. The allowable size is 1 to CELF_MP_PS_HOST_MAX.
 322 **Definition:** unsigned char[CELF_MP_PS_HOST_MAX + 1]
 323

324 **1.3.2 CelfMpPsApn**

325 **Description:** It contains the Access Point Name string terminated by a NULL character '\0'. The
 326 allowable size is 1 to CELF_MP_PS_APN_MAX.
 327 **Definition:** unsigned char[CELF_MP_PS_APN_MAX + 1]
 328

329 **1.3.3 CelfMpPsDnsRec**

330 **Description:** It contains the DNS record name string terminated by a NULL character '\0'. The
 331 allowable size is 1 to CELF_MP_PS_DNS_RECORD_MAX.
 332 **Definition:** unsigned char [CELF_MP_PS_DNS_RECORD_MAX + 1]
 333

334 **1.3.4 CelfMpPsTitle**

335 **Description:** It contains the title string for the APN terminated by a NULL character '\0'. The
 336 allowable size is 1 to CELF_MP_PS_TITLE_MAX.
 337 **Definition:** unsigned char[CELF_MP_PS_TITLE_MAX + 1]
 338

339 **1.3.5 CelfMpPsApnHostData**

340 **Description:** APN/host data structure

341 **Definition:**
 342 APN/host data structure
 343 typedef struct {
 344 CelfMpPsApnId apn_id ;
 345 CelfMpPsApn Apn ;
 346 CelfMpPsHost Host ;
 347 CelfMpPsTitle Title ;
 348 } CelfMpPsApnHostData;
 349

350 **1.4 Events type**

351 **1.4.1 Packet switched call state notification event**

352 In this sub-section, the associated data structure is CELF_MP_EVENT with the following values:

353
 354 category = PacketNotify;
 355 subtype = PacketNotify_Call_State;
 356

357 The value of the field 'info' is the event from enum CelfMpPsEvent.
 358

359 **1.4.1.1 CELF_MP_PS_RECEPTION_START_IND**

360 **Description:** It notifies the start of packet switched reception
 361

362 **1.4.1.2 CELF_MP_PS_EMISSION_START_IND**

363 **Description:** It notifies the start of packet switched emission
 364

365 **1.4.1.3 CELF_MP_PS_RECEPTION_REJ_IND**

366 **Description:** It notifies the rejection of packet switched reception
 367

Classification: Packet Switched Communication Service

368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422

1.4.1.4 CELF_MP_PS_EMISSION_REJ_IND

Description: It notifies the rejection of packet switched emission

1.4.1.5 CELF_MP_PS_CONNECT_START_IND

Description: It notifies the start of packet switched connection

1.4.1.6 CELF_MP_PS_DISCONNECT_REQ

Description: Notification of packet disconnecting (on mobile station) indication started

1.4.1.7 CELF_MP_PS_DEACTIVATE_CNF

Description: Notification of packet disconnecting indication ended

1.4.1.8 CELF_MP_PS_DEACTIVATE_IND

Description: Notification of packet disconnecting (on network) completed

1.4.1.9 CELF_MP_PS_EMISSION_SETUP_RSP

Description: Notification of transmission record registered

1.4.1.10 CELF_MP_PS_CALLED_SETUP_RSP

Description: Notification of termination record registered

1.4.1.11 CELF_MP_PS_CALLED_REJ_REQ

Description: Notification of missed termination record registered

1.4.1.12 CELF_MP_PS_CALLED_END_IND

Description: Notification of packet termination indication ended

1.4.2 Packet switched service status notification event

In this sub-section, the associated data structure is CELF_MP_EVENT with the following parameters:

category = PacketNotify;
subtype = PacketNotify_Service_State;

The value of the field 'info' is the event from enum CelfMpPsEvent.

1.4.2.1 CELF_MP_PS_SERVICE_RING

Description: packet switched service establishment

1.4.2.2 CELF_MP_PS_SERVICE_ERROR

Description: packet switched service ERROR

1.4.2.3 CELF_MP_PS_SERVICE_OK

Description: packet switched service OK (successful completion of primitive)

1.4.2.4 CELF_MP_PS_CONTROL_DISCONNECT

Description: packet switched service DISCONNECT

1.4.3 Packet switched call activity notification event

In this sub-section, the associated data structure is CELF_MP_EVENT with the following parameters:

category = PacketNotify;
subtype = PacketNotify_Call_Data;

Classification: Packet Switched Communication Service

423 The value of the field 'info' is the event from enum CelfMpPsEvent.
424

1.4.3.1 CELF_MP_PS_DATA_SEND_IND

425
426 **Description:** Data emission notification
427

1.4.3.2 CELF_MP_PS_DATA_RECV_IND

428
429 **Description:** Data reception notification
430

1.4.4 External equipment status notification event

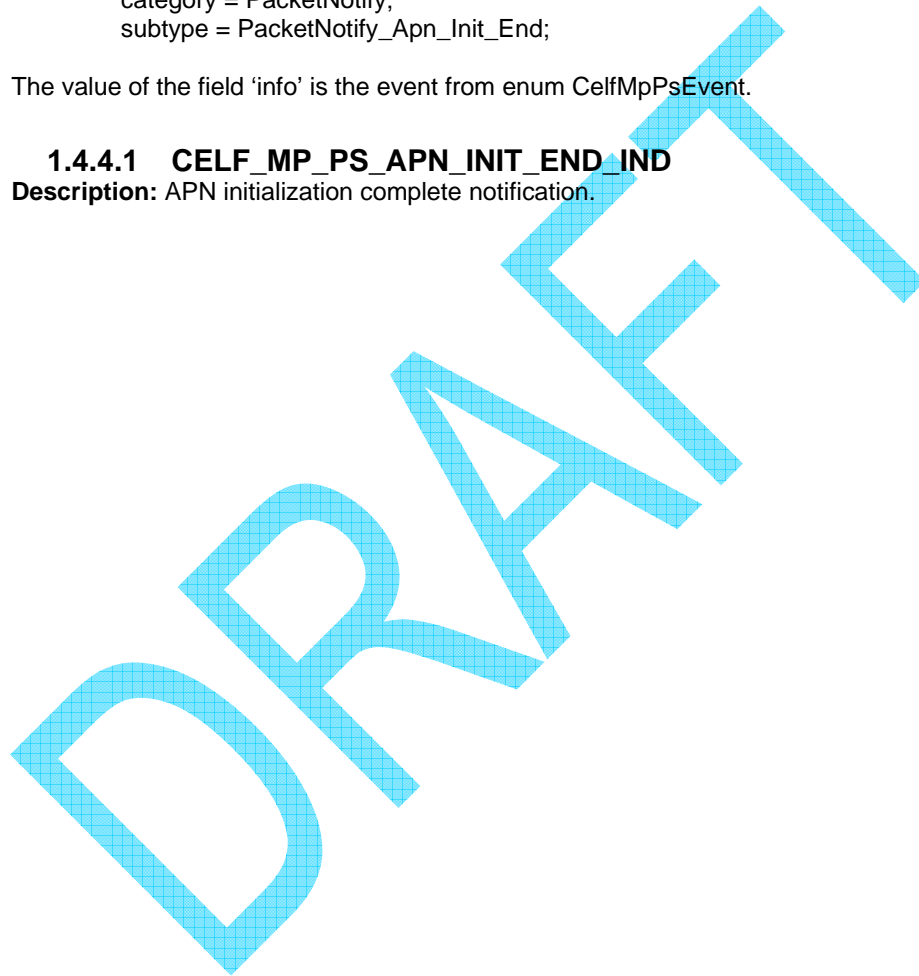
431 In this sub-section, the associated data structure is CELF_MP_EVENT with the following
432 parameters:
433

434 category = PacketNotify;
435 subtype = PacketNotify_Apn_Init_End;
436

437
438 The value of the field 'info' is the event from enum CelfMpPsEvent.
439

1.4.4.1 CELF_MP_PS_APN_INIT_END_IND

440
441 **Description:** APN initialization complete notification.
442



443

444

2. Start packet switched communication notification

445

2.1 Symbol: `celf_mp_ps_notification_start`

446

2.1.1 Syntax

447

```
CelfMpStatus celf_mp_ps_notification_start (
```

448

```
    CelfMpAppId      app_id,
```

449

```
    CelfMpPsEventSet event_set,
```

450

```
    CelfMpCallback   callback_func );
```

451

452

2.1.2 Argument

453

Name: `app_id`

454

Type: `CelfMpAppId`

455

I/O: |

456

Description: Application identifier.

457

458

Name: `event_set`

459

Type: `CelfMpPsEventSet`

460

I/O: |

461

Description: Set of notification events. Events belonging to one of the `CelfMpPsEventSet` classes **may** be registered to have a callback function called when the event occurs for the application identified by `app_id`. Classes of events are selected by setting the corresponding bit in `event_set`.

462

463

464

465

The following notification event classes **shall** be supported:

466

`CELf_MP_PS_CLASS_CALL_STATE_MSG`

467

`CELf_MP_PS_CLASS_SERVICE_STATE_MSG`

468

`CELf_MP_PS_CLASS_CALL_DATA_MSG`

469

470

A callback may be registered for all classes of events using the special event class `CELf_MP_PS_CLASS_ALL`, however to reduce overhead it is recommended that only the needed event classes should be registered.

471

472

473

474

475

Name: `callback_func`

476

Type: `CelfMpCallback`

477

I/O: |

478

Description: The callback function that shall be called when an event occurs from one of the classes registered in `event_set`.

479

480

481

2.1.3 Return Value

482

Type: `CelfMpStatus`

483

Description: `celf_mp_ps_notification_start ()` **shall** return one of the following values :

484

`CELf_MP_STATUS_OK`: Successful completion

485

`CELf_MP_STATUS_APP_ID_ERR`: Invalid Application ID

486

`CELf_MP_STATUS_EVENT_SET_ERR`: The set of event is invalid

487

`CELf_MP_STATUS_ERR`: Other unsuccessful completion

488

489

2.1.4 Include File

490

`/usr/include/celf/mp_ps.h`

491

492

2.1.5 Functional Description

493

This function is used to start notification for events related to packet switched communication.

494

Events from a registered class **shall** cause the registered callback function to be called when

495

the event occurs for the application identified by `app_id`. If a class of events does not have a

496

registered callback function, no callback shall occur for those events.

497

Classification: Packet Switched Communication Service

498 Different callback functions **may** be registered for different classes of events. Only the most
499 recently registered callback for an event class **shall** be called when an event of that class
500 occurs, and no notification shall be given when a callback for an event class is superseded.
501 Callbacks **may** be canceled using the `celf_mp_ps_notification_stop()` function.
502

The following notification event classes **shall** be supported:

503 CELF_MP_PS_CLASS_CALL_STATE_MSG
504 CELF_MP_PS_CLASS_SERVICE_STATE_MSG
505 CELF_MP_PS_CLASS_CALL_DATA_MSG
506

DRAFT

507 3. Stop packet switched communication notification

508 3.1 Symbol: `celf_mp_ps_notification_stop`

509 3.1.1 Syntax

```
510 CelfMpStatus celf_mp_ps_notification_stop (  
511 CelfMpAppId app_id,  
512 CelfMpPsEventSet event_set );  
513
```

514 3.1.2 Argument

515 **Name:** `app_id`
516 **Type:** `CelfMpAppId`
517 **I/O:** |
518 **Description:** Application identifier.

519
520 **Name:** `event_set`
521 **Type:** `CelfMpPsEventSet`
522 **I/O:** |
523 **Description:** Set of notification events (see 2. Start packet switched communication
524 notification).
525

526 3.1.3 Return Value

527 **Type:** `CelfMpStatus`
528 **Description:** `celf_mp_ps_notification_stop ()` **shall** return one of the following values :
529 `CELF_MP_STATUS_OK`: Successful completion
530 `CELF_MP_STATUS_APP_ID_ERR`: Invalid Application ID
531 `CELF_MP_STATUS_EVENT_SET_ERR`: The set of event is invalid
532 `CELF_MP_STATUS_ERR`: Other unsuccessful completion
533

534 3.1.4 Include File

535 `/usr/include/celf/mp_ps.h`
536

537 3.1.5 Functional Description

538 This function stops notification callbacks for the selected event classes related to packet
539 switched communication.
540 Events from the classes identified by `event_set` **shall** have their callbacks cancelled for the
541 application identified by `app_id`. If no callback is registered for a selected class, no error **shall**
542 occur.

543 To restart callbacks for these event classes, the application **must** call
544 `celf_mp_ps_notification_start()` again.

545
546 For more information about packet communication classes, (see 2. Start packet switched
547 communication notification).
548

549 4. Get packet switched communication status

550 4.1 Symbol: `celf_mp_ps_com_status_get`

551 4.1.1 Syntax

```
552 CelfMpStatus celf_mp_ps_com_status_get (
553     CelfMpAppId app_id,
554     CelfMpPsPdpType pdp_type,
555     CelfMpPsStatus * status);
556
```

557 4.1.2 Argument

558 **Name:** `app_id`
 559 **Type:** `CelfMpAppId`
 560 **I/O:** |
 561 **Description:** Application identifier.

562 **Name:** `pdp_type`
 563 **Type:** `CelfMpPsPdpType`
 564 **I/O:** |
 565 **Description:** This parameter **shall** have one of the following values :
 566 `CELf_MP_PS_PDP_TYPE_X25`
 567 `CELf_MP_PS_PDP_TYPE_IP`
 568 `CELf_MP_PS_PDP_TYPE_OSPIH`
 569 `CELf_MP_PS_PDP_TYPE_PPP`
 570 `CELf_MP_PS_PDP_TYPE_IPv6`

571 **Name:** `status`
 572 **Type:** `CelfMpPsStatus`
 573 **I/O:** O
 574 **Description:** `celf_mp_ps_com_status_get ()` **shall** return one of the following values :
 575 `CELf_MP_PS_EMPTY`: Communication ended (idle)
 576 `CELf_MP_PS_ACCEP_CONN`: Connected for incoming call
 577 `CELf_MP_PS_CONNECT`: Connected for outgoing call
 578 `CELf_MP_PS_ACTIVE`: Communication active (C-Plane active)
 579 `CELf_MP_PS_DISCONNECT`: Disconnected

583 4.1.3 Return Value

584 **Type:** `CelfMpStatus`
 585 **Description:** `celf_mp_ps_com_status_get ()` **shall** return one of the following values :
 586 `CELf_MP_STATUS_OK`: Successful completion
 587 `CELf_MP_STATUS_APP_ID_ERR`: Invalid Application ID
 588 `CELf_MP_STATUS_PS_PDP_TYPE_ERR`: unsupported PDP type
 589 `CELf_MP_STATUS_ERR`: Other unsuccessful completion

591 4.1.4 Include File

592 `/usr/include/celf/mp_ps.h`
 593

594 4.1.5 Functional Description

595 This function **shall** return the current packet communication status of the specified PDP type for
 596 the application identified by `app_id`. It is not necessary to call
 597 `celf_mp_ps_notification_start ()` to use `celf_mp_ps_com_status_get()`.
 598
 599 `pdp_type` **must** be an individual enumerator (`pdp_type` does not allow the OR operator)

600

601

5. Request packet switched communication connection

602

603

5.1 Symbol: `celf_mp_ps_connect`

604

5.1.1 Syntax

605

```
CelfMpStatus   celf_mp_ps_connect (
606     CelfMpAppId      app_id,
607     CelfMpPsPdpType  pdp_type);
608
```

609

5.1.2 Argument

610

Name: `app_id`

611

Type: `CelfMpAppId`

612

I/O: `I`

613

Description: Application identifier.

614

615

Name: `pdp_type`

616

Type: `CelfMpPsPdpType`

617

I/O: `I`

618

Description: This parameter **shall** have one of the following values :

619

`CELf_MP_PS_PDP_TYPE_X25`

620

`CELf_MP_PS_PDP_TYPE_IP`

621

`CELf_MP_PS_PDP_TYPE_OSPIH`

622

`CELf_MP_PS_PDP_TYPE_PPP`

623

`CELf_MP_PS_PDP_TYPE_IPv6`

624

625

5.1.3 Return Value

626

Type: `CelfMpStatus`

627

Description: `celf_mp_ps_connect ()` **shall** return one of the following values :

628

`CELf_MP_STATUS_OK:` Successful completion

629

`CELf_MP_STATUS_PS_PDP_TYPE_ERR:` unsupported PDP type

630

`CELf_MP_STATUS_PS_DENIED:` Request rejected by network due to

631

no subscription to packet communication service

632

`CELf_MP_STATUS_ERR:` Other error

633

634

5.1.4 Include File

635

`/usr/include/celf/mp_ps.h`

636

637

5.1.5 Functional Description

638

This function **shall** open a packet communication channel of the specified type for the application specified by `app_id`.

639

640

641

When the processing of this function is started by MPP, it **shall** cause the `CELf_MP_PS_EMISSION_START_IND` event to occur. Applications **should** wait for this message before starting another MPP packet switched service related function call.

642

643

644

645

If the channel is successfully opened, it **shall** cause the `CELf_MP_PS_SERVICE_OK` event to occur.

646

647

648

If the channel is not successfully opened, it **shall** cause the `CELf_MP_PS_CONTROL_ERR` event to occur.

649

650

651

6. Response to a network request for packet switched connection

652

653

6.1 Symbol: celf_mp_ps_connect_rsp

654

6.1.1 Syntax

```

655 CelfMpStatus celf_mp_ps_connect_rsp (
656     CelfMpAppId app_id,
657     CelfMpPsConn type,
658     CelfMpCallRef call_ref );
659

```

660

6.1.2 Argument

Name: app_id
Type: CelfMpAppId
I/O: I
Description: Application identifier.

665

666

Name: type
Type: CelfMpPsConn
I/O: I
Description: CELF_MP_PS_COM_CONN_ACCEPT: Accept incoming
 CELF_MP_PS_COM_CONN_REJECT: Reject incoming

667

668

669

670

671

Name: call_ref
Type: CelfMpCallRef
I/O: I
Description: Call Reference for incoming call

672

673

674

675

676

677

6.1.3 Return Value

678

Type: CelfMpStatus

679

Description: celf_mp_ps_connect_rsp () **shall** return one of the following values :

680

CELF_MP_STATUS_OK: Successful completion

681

CELF_MP_STATUS_APP_ID_ERR: Invalid Application ID

682

CELF_MP_STATUS_CALL_REF_ERR: Invalid Call reference

683

CELF_MP_STATUS_ERR: Other unsuccessful completion

684

685

6.1.4 Include File

686

/usr/include/celf/mp_ps.h

687

688

6.1.5 Functional Description

689

This function accepts or rejects the packet communication depending on the Parameter "type".

690

691

The start of processing by MPP **shall** be notified by the following event:

692

CELF_MP_PS_RECEPTION_START_IND

693

694

The completion of this function **shall** be notified by the following event:

695

CELF_MP_PS_SERVICE_OK

696

697

Unsuccessful processing of this function **shall** be notified by the following event:

698

CELF_MP_PS_SERVICE_ERROR

699

700 7. Request the deactivation of packet switched 701 communication

702 7.1 Symbol: celf_mp_ps_call_disconnect

703 7.1.1 Syntax

```
704 CelfMpStatus celf_mp_ps_disconnect (  
705 CelfMpAppId app_id,  
706 CelfMpCallRef call_ref );  
707
```

708 7.1.2 Argument

709 **Name:** app_id
710 **Type:** CelfMpAppId
711 **I/O:** I
712 **Description:** Application identifier.

713
714 **Name:** call_ref
715 **Type:** CelfMpCallRef
716 **I/O:** I
717 **Description:** Call Reference of the packet switched call to be disconnected
718

719 7.1.3 Return Value

720 **Type:** CelfMpStatus
721 **Description:** celf_mp_ps_disconnect () shall return one of the following values :
722 CELF_MP_STATUS_OK: Successful completion
723 CELF_MP_STATUS_APP_ID_ERR: Invalid Application ID
724 CELF_MP_STATUS_CALL_REF_ERR: Invalid Call reference
725 CELF_MP_STATUS_ERR: Other unsuccessful completion
726

727 7.1.4 Include File

```
728 /usr/include/celf/mp_ps.h  
729
```

730 7.1.5 Functional Description

731 This function disconnects a packet switched communication.

732
733 The start of processing by MPP shall be notified by the following event:
734 CELF_MP_PS_DISCONNECT_REQ
735

736 The completion of this function shall be notified by the following event:
737 CELF_MP_PS_SERVICE_OK
738

739 Unsuccessful processing of this function shall be notified by the following event:
740 CELF_MP_PS_SERVICE_ERROR
741

742

8. Specify APN settings

743

8.1 Symbol: celf_mp_ps_apn_set

744

8.1.1 Syntax

```

745 CelfMpStatus celf_mp_ps_apn_set (
746     CelfMpAppId      app_id,
747     CelfMpPsApnId   apn_id,
748     CelfMpPsApn *   Apn,
749     CelfMpPsHost *  Host,
750     CelfMpPsTitle * Title );
751

```

752

8.1.2 Argument

753 **Name:** app_id
 754 **Type:** CelfMpAppId
 755 **I/O:** I
 756 **Description:** Application identifier.

758 **Name:** apn_id
 759 **Type:** CelfMpPsApnId
 760 **I/O:** I
 761 **Description:** User APN to set:

762	CELLF_MP_PS_APN_USERSET1:	User-specified 1
763	CELLF_MP_PS_APN_USERSET2:	User-specified 2
764	CELLF_MP_PS_APN_USERSET3:	User-specified 3
765	CELLF_MP_PS_APN_USERSET4:	User-specified 4
766	CELLF_MP_PS_APN_USERSET5:	User-specified 5
767	CELLF_MP_PS_APN_USERSET6:	User-specified 6
768	CELLF_MP_PS_APN_USERSET7:	User-specified 7
769	CELLF_MP_PS_APN_USERSET8:	User-specified 8
770	CELLF_MP_PS_APN_USERSET9:	User-specified 9
771	CELLF_MP_PS_APN_USERSET10:	User-specified 10

773 **Name:** Apn
 774 **Type:** CelfMpPsApn *
 775 **I/O:** I
 776 **Description:** APN data.

778 **Name:** Host
 779 **Type:** CelfMpPsHost *
 780 **I/O:** I
 781 **Description:** Host data.

783 **Name:** Title
 784 **Type:** CelfMpPsTitle *
 785 **I/O:** I
 786 **Description:** Title data.

788

8.1.3 Return Value

789 **Type:** CelfMpStatus
 790 **Description:** celf_mp_ps_apn_set () shall return one of the following values :
 791 CELLF_MP_STATUS_OK: Successful completion
 792 CELLF_MP_STATUS_APP_ID_ERR: Invalid Application ID
 793 CELLF_MP_STATUS_ERR: Other unsuccessful completion
 794

795

8.1.4 Include File

796 /usr/include/celf/mp_ps.h
 797

798
799
800

8.1.5 Functional Description

This function allows the modification of the APN, Host and Title data of one of the 10 user-specified APNs.

DRAFT

801

802 9. Get APN settings

803 9.1 Symbol: `celf_mp_ps_apn_get`

804 9.1.1 Syntax

```
805 CelfMpStatus celf_mp_ps_apn_get (  
806     CelfMpAppId app_id,  
807     CelfMpPsApnHostData * apn_data );  
808
```

809 9.1.2 Argument

810 **Name:** `app_id`
811 **Type:** `CelfMpAppId`
812 **I/O:** I
813 **Description:** Application identifier.
814
815 **Name:** `apn_data`
816 **Type:** `CelfMpPsApnHostData`
817 **I/O:** O
818 **Description:** APN/host data structure
819

820 9.1.3 Return Value

821 **Type:** `CelfMpStatus`
822 **Description:** `celf_mp_ps_apn_get ()` shall return one of the following values :
823 `CELf_MP_STATUS_OK`: Successful completion
824 `CELf_MP_STATUS_APP_ID_ERR`: Invalid Application ID
825 `CELf_MP_STATUS_ERR`: Other unsuccessful completion
826

827 9.1.4 Include File

```
828 /usr/include/celf/mp_ps.h  
829
```

830 9.1.5 Functional Description

831 This function returns the current APN data, host data and title data.

832

833

10. Select APN for packet switched communication

834

10.1 Symbol: celf_mp_ps_apn_select

835

10.1.1 Syntax

836

```
CelfMpStatus celf_mp_ps_apn_select (
```

837

```
    CelfMpAppId app_id,
```

838

```
    CelfMpPsApnId apn_id );
```

839

840

10.1.2 Argument

841

Name: app_id

842

Type: CelfMpAppId

843

I/O: |

844

Description: Application identifier.

845

846

Name: apn_id

847

Type: CelfMpPsApnId

848

I/O: |

849

Description: Destination number to be used

850

CELF_MP_PS_APN_DEFAULT : Default setting

851

CELF_MP_PS_APN_USERSET1: User-specified 1

852

CELF_MP_PS_APN_USERSET2: User-specified 2

853

CELF_MP_PS_APN_USERSET3: User-specified 3

854

CELF_MP_PS_APN_USERSET4: User-specified 4

855

CELF_MP_PS_APN_USERSET5: User-specified 5

856

CELF_MP_PS_APN_USERSET6: User-specified 6

857

CELF_MP_PS_APN_USERSET7: User-specified 7

858

CELF_MP_PS_APN_USERSET8: User-specified 8

859

CELF_MP_PS_APN_USERSET9: User-specified 9

860

CELF_MP_PS_APN_USERSET10: User-specified 10

861

862

10.1.3 Return Value

863

Type: CelfMpStatus

864

Description: celf_mp_ps_apn_select () shall return one of the following values :

865

CELF_MP_STATUS_OK: Successful completion

866

CELF_MP_STATUS_APP_ID_ERR: Invalid Application ID

867

CELF_MP_STATUS_ERR: Other unsuccessful completion

868

869

10.1.4 Include File

870

/usr/include/celf/mp_ps.h

871

872

10.1.5 Functional Description

873

This function sets the APN to be used for packet switched communication.

874

875 **11. Get APN used for packet switched**
876 **communication**

877 **11.1 Symbol: celf_mp_ps_apn_id_get**

878 **11.1.1 Syntax**

879 CelfMpPsApnId celf_mp_ps_apn_id_get (
880 CelfMpAppId app_id);
881

882 **11.1.2 Argument**

883 **Name:** app_id
884 **Type:** CelfMpAppId
885 **I/O:** I
886 **Description:** Application identifier.
887

888 **11.1.3 Return Value**

889 **Type:** CelfMpPsApnId
890 **Description:** celf_mp_ps_apn_id_get () shall return one of the following values :
891
892 CELF_MP_PS_APN_DEFAULT : Default setting
893 CELF_MP_PS_APN_USERSET1: User-specified 1
894 CELF_MP_PS_APN_USERSET2: User-specified 2
895 CELF_MP_PS_APN_USERSET3: User-specified 3
896 CELF_MP_PS_APN_USERSET4: User-specified 4
897 CELF_MP_PS_APN_USERSET5: User-specified 5
898 CELF_MP_PS_APN_USERSET6: User-specified 6
899 CELF_MP_PS_APN_USERSET7: User-specified 7
900 CELF_MP_PS_APN_USERSET8: User-specified 8
901 CELF_MP_PS_APN_USERSET9: User-specified 9
902 CELF_MP_PS_APN_USERSET10: User-specified 10
903

904 **11.1.4 Include File**

905 /usr/include/celf/mp_ps.h
906

907 **11.1.5 Functional Description**

908 This function returns the APN being used for packet switched communication.
909

910

911

12. Force deactivation of a packet communication

912

12.1 Symbol: `celf_mp_ps_force_disconnect`

913

12.1.1 Syntax

914

```
CelfMpStatus celf_mp_ps_force_disconnect (
```

915

```
    CelfMpAppId  app_id,
```

916

```
    CelfMpCallRef call_ref );
```

917

918

12.1.2 Argument

919

Name: `app_id`

920

Type: `CelfMpAppId`

921

I/O: `I`

922

Description: Application identifier.

923

924

Name: `call_ref`

925

Type: `CelfMpCallRef`

926

I/O: `I`

927

Description: Packet switched call reference to be disconnected forcibly.

928

929

12.1.3 Return Value

930

Type: `CelfMpStatus`

931

Description: `celf_mp_ps_force_disconnect ()` shall return one of the following values :

932

`CELF_MP_STATUS_OK`: Successful completion

933

`CELF_MP_STATUS_APP_ID_ERR`: Invalid Application ID

934

`CELF_MP_STATUS_ERR`: Other unsuccessful completion

935

936

12.1.4 Include File

937

`/usr/include/celf/mp_ps.h`

938

939

12.1.5 Functional Description

940

This function disconnects a packet communication forcibly.

941

942

The completion of this function shall be notified by the following event:

943

`CELF_MP_PS_DEACTIVATE_IND`

944

945

Unsuccessful processing of this function shall be notified by the following event:

946

`CELF_MP_PS_SERVICE_ERROR`

947

948 **13. Get User Plane connection status**

949 **13.1 Symbol: celf_mp_ps_uplane_status_get**

950 **13.1.1 Syntax**

951 CelfMpPsUPlane celf_mp_ps_uplane_status_get (
952 CelfMpAppId app_id);
953

954 **13.1.2 Argument**

955 **Name:** app_id
956 **Type:** CelfMpAppId
957 **I/O:** I
958 **Description:** Application identifier.
959

960 **13.1.3 Return Value**

961 **Type:** CelfMpPsUPlane
962 **Description:** celf_mp_ps_uplane_status_get () **shall** return one of the following values :
963 CELF_MP_PS_UPLANE_ON : U-Plane connected
964 CELF_MP_PS_UPLANE_OFF: U-Plane not connected
965

966 **13.1.4 Include File**

967 /usr/include/celf/mp_ps.h
968

969 **13.1.5 Functional Description**

970 This function gets the current U-Plane connection status.

971

972 14. Get DNS record name

973 14.1 Symbol: celf_mp_ps_dns_record_get

974 14.1.1 Syntax

```
975 CelfMpStatus celf_mp_ps_dns_record_get (  
976 CelfMpAppId app_id,  
977 CelfMpPsDnsRec * DNS_Record );  
978
```

979 14.1.2 Argument

980 **Name:** app_id
981 **Type:** CelfMpAppId
982 **I/O:** I
983 **Description:** Application identifier.

984
985 **Name:** DNS_Record
986 **Type:** CelfMpPsDnsRec *
987 **I/O:** O
988 **Description:** DNS record name.
989

990 14.1.3 Return Value

991 **Type:** CelfMpStatus
992 **Description:** celf_mp_ps_dns_record_get () **shall** return one of the following values :
993 CELF_MP_STATUS_OK: Successful completion
994 CELF_MP_STATUS_APP_ID_ERR: Invalid Application ID
995 CELF_MP_STATUS_ERR: Other unsuccessful completion
996

997 14.1.4 Include File

998 /usr/include/celf/mp_ps.h
999

1000 14.1.5 Functional Description

1001 This function retrieves the name of the DNS for the destination currently being used.
1002

1003

1004 15. Get default APN mode

1005 15.1 Symbol: `celf_mp_ps_apn_mode_get`

1006 15.1.1 Syntax

1007 `CelfMpPsApnMode celf_mp_ps_apn_mode_get (`
1008 `CelfMpAppId app_id);`
1009

1010 15.1.2 Argument

1011 **Name:** `app_id`
1012 **Type:** `CelfMpAppId`
1013 **I/O:** `I`
1014 **Description:** Application identifier.
1015
1016

1017 15.1.3 Return Value

1018 **Type:** `CelfMpPsApnMode`
1019 **Description:** `celf_mp_ps_apn_mode_get ()` shall return one of the following values :
1020 `CELf_MP_PS_DEFAULT_APN_MS`: Settings in the mobile phone
1021 `CELf_MP_PS_DEFAULT_APN_UIM`: Settings in the UIM
1022

1023 15.1.4 Include File

1024 `/usr/include/celf/mp_ps.h`
1025

1026 15.1.5 Functional Description

1027 This function gets the type of default APN settings. The default APN could be set to UIM or to
1028 Mobile phone.
1029

1030

1031

16. Force deactivation of all packet switched communications

1032

1033

16.1 Symbol: `celf_mp_ps_service_shutdown`

1034

16.1.1 Syntax

1035

```
CelfMpStatus celf_mp_ps_service_shutdown (
```

1036

```
    CelfMpAppId app_id );
```

1037

1038

16.1.2 Argument

1039

Name: `app_id`

1040

Type: `CelfMpAppId`

1041

I/O: `I`

1042

Description: Application identifier.

1043

1044

16.1.3 Return Value

1045

Type: `CelfMpStatus`

1046

Description: `celf_mp_ps_service_shutdown ()` shall return one of the following values :

1047

`CELF_MP_STATUS_OK`: Successful completion

1048

`CELF_MP_STATUS_APP_ID_ERR`: Invalid Application ID

1049

`CELF_MP_STATUS_ERR`: Other unsuccessful completion

1050

1051

16.1.4 Include File

1052

`/usr/include/celf/mp_ps.h`

1053

1054

16.1.5 Functional Description

1055

This function disconnects all packet switched communications currently being used by the application.

1056

1057

1058 17.Request APN initialization

1059 17.1 Symbol: celf_mp_ps_apn_initialize

1060 17.1.1 Syntax

```
1061 CelfMpPsApnInit celf_mp_ps_apn_initialize (  
1062 CelfMpAppId app_id );  
1063
```

1064 17.1.2 Argument

1065 **Name:** app_id
1066 **Type:** CelfMpAppId
1067 **I/O:** I
1068 **Description:** Application identifier.
1069

1070 17.1.3 Return Value

1071 **Type:** CelfMpPsApnInit
1072 **Description:** celf_mp_ps_apn_initialize () **shall** return one of the following values :
1073
1074 CELF_MP_PS_UIM_UPDATED: Successful, UIM updated
1075 CELF_MP_PS_UIM_NOT_UPDATED: Successful, UIM not updated
1076 CELF_MP_PS_AUTH_ERR : Failure returned from the authentication service
1077

1078 17.1.4 Include File

```
1079 /usr/include/celf/mp_ps.h  
1080
```

1081 17.1.5 Functional Description

1082 This function initializes APN, this initialized APN is called default APN.
1083 When the power is turned on, this function starts APN initialization according to a request from
1084 an authentication application which initializes APN according to the process below:
1085 (case1) When the APN information in the UIM is not the same as the APN information
1086 requested by the authentication application,
1087 the APN information requested by the authentication application is stored in the UIM or in the
1088 mobile phone.
1089 (case2) When the APN information in UIM is same as the APN information requested by
1090 authentication application,
1091 the information in the UIM is not changed.
1092 (case3) In case of an error such as failure of reading the UIM information,
1093 the APN information requested by the authentication application is set in the mobile phone.
1094 The APN initialization completion is notified as an event
1095 CELF_MP_PS_EXT_EVENT_APN_INIT_END
1096 Additional information to it is used to post the used destination number.
1097

1098 18.Start monitoring external equipment for packet 1099 switched communication

1100 18.1 Symbol: celf_mp_ps_ext_device_notification_start

1101 18.1.1 Syntax

```
1102 CelfMpStatus celf_mp_ps_ext_device_notification_start (  
1103 CelfMpAppId app_id,  
1104 CelfMpPsExtEventSet mask,  
1105 CelfMpCallback callback_func);  
1106
```

1107 18.1.2 Argument

1108 **Name:** app_id
1109 **Type:** CelfMpAppId
1110 **I/O:** I
1111 **Description:** Application identifier.

1112
1113 **Name:** mask
1114 **Type:** CelfMpPsExtEventSet
1115 **I/O:** I
1116 **Description:** Set of external notification events.
1117 Enabled with the bit turned ON
1118 CELF_MP_PS_EXT_EVENT_APN_INIT_END : Notification of APN initialization
1119 completed
1120 CELF_MP_PS_EXT_EVENT_CLASS_ALL : All

1121
1122 **Name:** callback_func
1123 **Type:** CelfMpCallback
1124 **I/O:** I
1125 **Description:** Callback function used to notify the event.
1126

1127 18.1.3 Return Value

1128 **Type:** CelfMpStatus
1129 **Description:** celf_mp_ps_ext_device_notification_start () shall return one of the
1130 following values :
1131 CELF_MP_STATUS_OK: Successful completion
1132 CELF_MP_STATUS_APP_ID_ERR: Invalid Application ID
1133 CELF_MP_STATUS_EVENT_SET_ERR: The set of event is invalid
1134 CELF_MP_STATUS_ERR: Other unsuccessful completion
1135

1136 18.1.4 Include File

1137 /usr/include/celf/mp_ps.h
1138

1139 18.1.5 Functional Description

1140 This function start sending event notification regarding the external equipment i.e. external TE
1141 for packet communication.

1142 Applications may be notified of the following events:

1143
1144 - APN initialization completed: Notifies the APN initialization completed and the destination
1145 number to be used.

1146

1147

19. Stop monitoring external equipment for packet switched communication

1148

1149

19.1 Symbol: `celf_mp_ps_ext_device_notification_stop`

1150

19.1.1 Syntax

1151

```
CelfMpStatus celf_mp_ps_ext_device_notification_stop (
```

1152

```
    CelfMpAppId app_id,
```

1153

```
    CelfMpPsExtEventSet mask );
```

1154

1155

19.1.2 Argument

1156

Name: `app_id`

1157

Type: `CelfMpAppId`

1158

I/O: `I`

1159

Description: Application identifier.

1160

1161

Name: `mask`

1162

Type: `CelfMpPsExtEventSet`

1163

I/O: `I`

1164

Description: Set of external notification events.

1165

Stops the notification when the bit is set

1166

`CELf_MP_PS_EXT_EVENT_APN_INIT_END` : Notification of APN initialization

1167

completed

1168

`CELf_MP_PS_EXT_EVENT_CLASS_ALL` : All

1169

1170

19.1.3 Return Value

1171

Type: `CelfMpStatus`

1172

Description: `celf_mp_ps_ext_device_notification_stop ()` shall return one of the following values :

1173

`CELf_MP_STATUS_OK`: Successful completion

1174

`CELf_MP_STATUS_APP_ID_ERR`: Invalid Application ID

1175

`CELf_MP_STATUS_EVENT_SET_ERR`: The set of event is invalid

1176

`CELf_MP_STATUS_ERR`: Other unsuccessful completion

1177

1178

1179

19.1.4 Include File

1180

`/usr/include/celf/mp_ps.h`

1181

1182

19.1.5 Functional Description

1183

This function stops sending event notification for the events specified in Symbol:

1184

`celf_mp_ps_ext_device_notification_start`.