**Ref. No. IT-3.10**

**NP, CPS**

Inter-Operator Interface for   
NP & CPS

**Inter-Operator interface for NP & CPS**

**Contents**

1 INTRODUCTION 1-1

1.1 Background 1-1

1.2 Document history 1-1

1.3 References 1-2

1.4 Definitions 1-3

2 INTERFACE ARCHITECTURE 2-1

2.1 General requirements on the interface 2-1

2.2 Message structure 2-1

2.3 Meshed architecture (NP) or star architecture (CPS) 2-2

2.4 File exchange over the Internet 2-3

2.5 File exchange using HTTP-s 2-4

2.6 Security 2-6

2.6.1 Security requirements 2-6

2.6.2 Security risks 2-7

2.6.3 Security implementation 2-7

2.6.4 On SSL 2-7

2.6.5 Procedures 2-7

2.6.6 Additional procedure 2-8

2.7 Interface Protocol Stack 2-8

2.7.1 Industry standard 2-8

2.7.2 XML 2-9

2.7.3 Flat file 2-9

2.7.4 SOAP and HTTP-s 2-13

2.7.5 SSL 2-13

2.7.6 TCP/IP 2-13

2.7.7 Fault handling 2-13

3 XML structure specification (order messages specification) 3-1

3.1 NP messages specification 3-1

3.1.1 npProvide 3-1

3.1.2 npAccept 3-3

3.1.3 npReject 3-4

3.1.4 npChange 3-4

3.1.5 npCancel 3-5

3.1.6 npAbort 3-6

3.1.7 npPortComplete 3-6

3.1.8 npReturnNumber 3-7

3.1.9 npSubsequentPort 3-8

3.1.10 npIDcheck 3-8

3.1.11 npIDcheckResp 3-9

3.1.12 recipientOperator, losingOperator, donorOperator 3-9

3.1.13 DDI range 3-11

3.1.14 NP Message attributes specification 3-12

3.2 CPS (PRE) messages specification 3-1

3.2.1 cpsProvide 3-1

3.2.2 preProvide 3-2

3.2.3 cpsReject 3-4

3.2.4 cpsAccept 3-4

3.2.5 cpsActivated 3-5

3.2.6 preReject 3-5

3.2.7 preActivated 3-6

3.2.8 cpsOperator 3-6

3.2.9 CPS Message attributes specification 3-6

3.3 Common messages specification 3-10

3.3.1 header 3-10

3.3.2 fileReject 3-12

3.3.3 fileAccept 3-12

3.3.4 serviceContractOwner 3-13

3.3.5 installationAddress 3-14

3.3.6 Operator ID 3-17

3.3.7 Customer information 3-17

3.3.8 Date and time 3-18

3.3.9 CompleteTime and CompleteDate 3-19

3.3.10 Tariff 3-19

3.3.11 ProductList 3-20

3.3.12 Rejection 3-21

3.3.13 Message attributes specification 3-22

4 Error handling 4-1

4.1 Error handling principles 4-1

4.1.1 Rejection categories 4-3

4.1.2 Rejection codes 4-3

5 APPENDIX A: OUTAGE SITUATIONS (NP AND CPS) 5-1

6 APPENDIX B: INSTALLATION ADDRESS REFERENCES (NP AND CPS) 6-1

7 APPENDIX C: DDI numbering 7-1

7.1 Detailed Number rules for Ordering CPS over DDI services 7-1

7.2 Sub-Ranges and The Customer’s Active DDI Range 7-2

8 APPENDIX E: DTD DISTRIBUTION AND CHANGE MANAGEMENT (NP AND CPS) 8-1

9 APPENDIX F: HANDLING FRAUDULENT ORDERS (NP AND CPS) 9-1

10 APPENDIX G: BRIEF EXPLANATION ON DTD NOTATION (NP AND CPS) 10-1

11 APPENDIX H: DTD 11-1

12 APPENDIX I: XML FILE EXAMPLES 12-1

12.1 ORDR 12-1

12.2 RSLT 12-3

12.3 VLDT 12-4

12.4 FLAC 12-4

12.5 FILE SAMPLES 12-5

13 APPENDIX J: SOAP Message 13-1

13.1 Request 13-1

13.2 Response 13-1

14 Appendix K: WSDL 14-1

15 APPENDIX L: General description of communication between OLOs 15-1

# INTRODUCTION

The purpose of the following document is to provide a set of specifications for Number Portability Carrier Pre-Selection, WholesaleLineRental (NP&CPS).

This document regards the technical details of the interface between operators to support NP&CPS order processing.

This document addresses the following subjects related to this technical interface:

* Format of the messages exchanged between operators
* The mechanisms which enables the exchange of the messages between operators

## Background

There will be between 15 and 30 operators offering Carrier Selection (CS), Carrier Pre-Selection (CPS) and Number Portability (CP) services in the Czech Republic.

For CS, the service establishment is primarily a responsibility of the Access Provider (AP) operator with Significant Market Power (SMP), at the present time CETIN For NP, Service establishment process will affect all operators in accordance with the Telecommunications Law.

## Document history

|  |  |  |
| --- | --- | --- |
| **Date** | **Version** | **Change** |
| 29-05-02 | 1.0 | Former Czech Telecom proposal of IT document changed into APVTS document |
| 22-10-02 | 1.3 | The final SOAP specification included. Document finalized. |
| 11-02-03 | 1.4 | Working version. Incorporated changes agreed during implementation |
| 06-03-03 | 1.4-1 Draft | Working version. Incorporated comments of Project team NP&CS of the Technical Committee of the APVTS |
| 18-03-03 | 1.4-2 Draft | Working version. References to the DIOP document were substituted by proper references to appropriate service standard |
| 20-03-03 | 1.5 Draft | Some references to CETIN removed. Minor changes in the wording |
| 31-03-03 | 1.5 | Operator IDs changed in examples to reflect real IDs |
| 31-05-03 | 1.6 | * Document restructured, common part for both NP and CPS messages created * Description of fault handling related to SOAP added * New rules for sequenceNr validation added |
| 22-03-06 | 2.0 | * The NP PROVIDE ORDER shall include a new parameter called a Loop reuse flag, which will be set true if this order is in conjunction with a local loop unbundling order ( HasLLU= true, false ) flag. * New HassLLU optional flag was added into DTD – NP Provide definiton * General requirement for the interface availability The interface of each individual operator shall be 95% available and reliable 24 hours / day * VLDT, FLAC – processing time frame was changed  Between 20.00 and 8.59 for the file types 'Validate' and 'File acknowledge' CETIN will send VLDT, FLAC between 20.00-8.00 * In the terms of exceptional situation (OHS upgrade, OHS drop out ) FLAC can be sent immediately when problem is resolved, ie. FLAC time sent frame was changed to 24 hours anytime. |
| 02-09-06 | 2.1 | * OHS specification for the new PRE service (Preuctovani pausalu) added * CPS PROVIDE ORDER shall include a new parameter – (pre= true, false) * NEW ORDER SPECIFICATION for PRE orders added – preProvide message ( PRE service further ORDER for already processed CPS) * NEW RESPONSE SPECIFIACTION for PRE service added – preActivated, preRejected * New PRE service XML message structure added into DTD |
|  |  |  |

## References

|  |  |
| --- | --- |
| **Ref** | **Title** |
| [VPNPC] | * Výběr provozovatele nastavením předvolby čísel (APVTS a ČTÜ – listopad 2002). The Acronym VPNPC is used only in the scope of this document. * Referenční nabídka propojení společnosti CETIN |
| [PC] | * Přenositelnost čísla (APVTS a ČTÜ – prosinec 2002) * OOP10- opatření opecné povahy popisující process přenesení čísla v platném znění * Referenční nabídka propojení společnosti CETIN |
| [XML] | XML Specificatoin <http://www.w3c.org/xml> |
| [SOAP] | SOAP 1.1 Specification <http://www.w3.org/TR/SOAP> |
| [HTTP] | HTTP 1.1 Specification <ftp://ftp.isi.edu/in-notes/rfc2616.txt> |
| [SSL] | SSL Specification <http://wp.netscape.com/security/techbriefs/ssl.html> |

## Definitions

The term “order” and “service order” are used interchangeably throughout this document.

The following definitions will be used for CS and CPS:

CPS operator - The operator to which calls are routed per CPS subscription

CS operator - The operator to which calls are routed when a customer dials a CS access code

Access Provider - the operator providing access capabilities to the CPS/CS operator

Start of Business (SOB) – 9 a.m. on business days

Close of Business (COB) – 5 p.m. on business days

The following definitions will be used in regards to NP:

losing operator/network: the operator from which the number is being ported

recipient operator/network: the operator to whom the number is being or has been ported

donor operator/network: the operator originally assigned the ported number (i.e., the number range holder from which the number originally came).

# INTERFACE ARCHITECTURE

## General requirements on the interface

**The interface of each individual operator shall be 95% available and reliable 24 hours / day!!!**

1. All orders shall be processed in the order in which they were received. (e.g., First in first out).
2. The data exchanged between two operators will be encrypted
3. Before two operators exchange data they will check authentication (username/password)
4. Each individual operator will take security measures, for example the usage of firewalls.
5. Each individual operator will log the events on the interface

## Message structure

The messages exchanged between the operators are in fact flat files containing multiple messages.

A flat file consists of a valid XML document.



The files will be sent on fixed moments during the day. A flat file will contain all the orders related messages of that day.

## Meshed architecture (NP) or star architecture (CPS)

The different operators will be connected by a meshed architecture; every operator will have a connection to every other operator:



In the case of Carrier Preselect only a CPS Access Provider will offer CPS services to their customers: each CPS operator is connected to that Access Provider.



## File exchange over the Internet

To enable the exchange of the files over the Internet every operator will have a HTTP-s server functionality (supporting SOAP 1.1) available for all other operators.



## File exchange using HTTP-s

When an operator has a file to send to another operator he uses an HTTP request (POST) to the HTTP-s server of that other operator.

This operator will store the file internally, and acknowledge receiving the request.



Principles:

* It is not required to send a file on a given day:  
     
  If an operator 1 does not have any orders on a given day to send to operator 2, he will not send a file.
* Each received file will be acknowledged by a SOAP Response   
  After an error occurred the initiator can send a new file (but before the deadline for this file type)
* The initiator (operator sending a file) must send a file (if applicable for that day) to the server of the responder (operator receiving a file) before a specified time on business days:  
  - Before COB (= 18.00) for the file types 'ORDR' and 'RSLT'  
  - Before SOB (= 9.00) for the file types 'VLDT' and 'FLAC'

- In the terms of exceptional situation (OHS upgrade, OHS drop out ) FLAC can be sent   
 immediately when problem is resolved, ie. FLAC can be sent before COB and SOB ie. 24   
 hours a day .  
  
Processing will start after that time. Only one file of each type a day will be processed.

* The responder (operator receiving a file) must have the server for receiving files available on business days:  
  - Between SOB (= 9.00) and COB (=18.00) for the file type 'Orders' and ‘Result’  
   After outage, in some cases described in appendix A, it must be possible to extend the COB from 18.00 to 19.00.  
  - Between 18.00 and 8.59 for the file types 'Validate' and 'File acknowledge'  
  CETIN will send VLDT, FLAC between 19.00-9.00
* If a file cannot be delivered, because of outage of the server of the responder, the initiator will make 3 additional attempts to transfer the file. If also this retry fails a manual outage procedure will be used.  
  The file will be delivered after the deadline (9.00 or 18.00).
* It is not possible to process more then one file of a file type from one initiator.  
  If, more then one file of the same type is received from an initiator, only the last one will be processed.  
  The other files are deleted: no fileAccept/ fileReject message will be send back!
* If there are multiple files received, but not yet processed, from an initiator with different dates (as a result of outage) the files will be accepted in the order they were received.

**Example 1:**

Operator 1 sends to operator 2 a new file every hour:

* file 1 at 12.00 with 30 orders
* file 2 at 13.00 with 40 orders (30 from file 1 + 10 new)
* file 3 at 14.00 with 50 orders (40 from file 2 + 10 new)
* file 4 at 15.00 with 60 orders (50 from file 3 + 10 new)
* file 5 at 16.00 with 70 orders (60 from file 4 + 10 new)

Each file transfer is successful (HTTP OK returned).

The file of 16.00 is processed.

**Example 2:**

Operator 1 sends to operator 2 a file of the type ‘orders’ at 15.00 and receives an HTTP acknowledgement (HTTP OK). At 16.00 a second file of the same type is sent, the transfer fails (HTTP error).

After 16.00 no new file is sent.

The first file, sent at 15.00, is processed.

**Example 3**

Operator 1 sends to operator 2 a file with 100 orders at 12.00.

At 14.00 he discovers that 20 orders are wrong, and 10 there are 10 new customer orders.

At 14.30 he sends a new file with 90 orders (80 excisting ones, 10 new).

## Security



### Security requirements

**Access channel protection**

The portal of each operator shall be protected by to limit the access to authorised operators.

**Authentication**

Before operators exchange order related messages their legitimacy should be established.

**Authorization**

An authenticated operator can only access the services that he is allowed to.

**Confidentiality**

Since the information contained in an order is confidential, it must be protected from unauthorized access by third parties.

**Data integrity**

Communications must be protected from undetectable alteration by third parties in transmission on the Internet.

**Non repudiation**

It should not be possible for an operator to reasonably claim that he or she did not send or receive an order. There must be sufficient evidence available of the transaction.

**Audit trail**

All communication between operators shall be registered in audit trail.

### Security risks

1. **Sabotage by a hacker**
2. **Sabotage by an operator**
3. **Denial of a business transaction**
4. **Slamming with denial**

**Damage:**

Financial damage: loss of customer revenue.

Damage to customers: temporarily loss of the phone service

Image damage: hacking or large scale slamming damages the image of the industry

### Security implementation

|  |  |  |
| --- | --- | --- |
| **Requirement** | **Implementation** | **Security level** |
| Access channel protection | Firewall | Sufficient |
| Authentication | User id/password, exchange encrypted using SSL  + Secure server ID issued by Certification Authority | Sufficient |
| Authorization | HTTP Post only | Sufficient |
| Confidentiality | SSL encrypts transfer | Sufficient |
| Data integrity | SSL encrypts transfer | Sufficient |
| Non repudiation | Authentication, event logging, acknowledgement of file and order | Limited |
| Audit trail | event logging | Sufficient |

### On SSL

The Secure Socket Layer is a widely used standard from Netscape that secures web-based communication.

SSL is built in to virtually all web servers.

An independent Certification Authority must certify servers that support SSL.

To make mutual authentication possible also the clients must be authenticated.

### Procedures

Next to the already described technical security measures, both “Výběr provozovatele nastavením předvolby čísel” and “Přenositelnost čísla” documents ([VPNPC] and [PC])) describe functional and procedural measures, which improve the security:

1. In the case of an NP order the customer must send a CAF to the loosing operator.
2. In the case of a CPS order the operator must afterwards be able to provide a CAF to Access Provider
3. Orders exchanged will be acknowledged (NP: by day 2, CPS: after service activation)

### Additional procedure

To improve the Interface proposal on the 'Non repudiation' requirement an additional procedure (see appendix) is added on the handling of fraudulent orders

## Interface Protocol Stack



### Industry standard

**The Referenc offer CETIN agrees on a set of standardised messages and final process. This document defining only detail of IT communication**.

Message types have been specified in the [VPNPC] and [PC] documents, these documents specify the structure of the following message types.

Message types defined:

* npProvide
* npAccept
* npReject
* npChange
* npCancel
* npAbort
* npPortComplete
* npReturnNumber
* npSubsequentPort
* npIDcheck
* npIDcheckResp
* cpsProvide
* cpsReject
* cpsActivated
* cpsAccept
* preProvide
* preActivated
* preRejected

### XML

This layer of the stack formats the defined message structures into a machine-readable structure: XML

**Character set**

In order to exchange special (Czech) characters correctly Unicode (UTF 8) will be used.

**DTD's**

The structure of the XML messages will be registered in DTD's.

DTD's of messages (the xxx.dtd files) will be stored locally, there is no centralised DTD server.

More information on XML can be found at: [XML]

### Flat file

The individual XML messages are gathered into flat files, and sent at agreed time.

**Flat file content**

Each flat file is an XML message, consisting of:

- A message header

- one or more CPS/NP order messages (as described in the XML message layer)

**Flat file message header specification**

|  |  |
| --- | --- |
| **Item** | **Contents** |
| To Operator | Code of the operator receiving this message |
| From Operator | Code of the operator originating this message |
| Flat file type | Code for flat file type:  ORDR– provide  VLDT - validate  RSLT – result  FLAC - File acknowledge |
| Message count | The number of messages contained in this file |
| TimeSent | Time this file was sent by originating operator |
| DateSent | Date this file was sent by originating operator |

Chapter 3.2 contains the exact XML specification of the header, including the DTD.

**Flat file types**

Since specific message types cannot be combined in one flat file, four different batch file types have been defined:

|  |  |  |  |
| --- | --- | --- | --- |
| **File type** | **Message types** | **Initiated by** | **Received before** |
| Orders | npProvide  npCancel  npChange  npSubsequenPort  npPortComplete  npReturnNumber  npIDcheck  (cpsProvide)  (preProvide) | Recipient operator  (CPS Operator) | COB (18.00), each business day |
| Validate | npReject  npIDcheckResp  (cpsReject) (preReject) | Losing operator  (Access Provider) | SOB (9.00), each business day |
| Result | npReject  npIDcheckResp  npAccept  npAbort  (cpsAccept)  (cpsReject)  (cpsActivated) | Losing operator  (Access Provider) | COB (18.00), each business day |
| File acknowledget | fileReject  fileAccept | All | SOB (9.00), each business day |

Note: the flat files can contain NP - CPS - message types mixed.

The Validate file sent at 9.00 will contain quota rejections and rejections due to validation errors, however only those where order validation is completed before the submission of the file. Rejected orders processed after the submission of this file will be sent either by 17:00 on Day 1 or at the latest by 17:00 on Day 2.

Rejections as a result of technical investigation (Customer CAF, DDI, directory nr's) will be reported by proces specification descripted on Referenc ofer CETIN

The following diagram illustrates the exchange of NP messages between two operators.



The following diagram illustrates the exchange of CPS messages between two operators.



**Flat file reject/accept**

If the whole flat file cannot be processed because of file errors (e.g. XML not well formed or XML not inline with DTD), a file of type FLAC will be sent back, including fileReject message for this file with the attribute refType containing the type of the file being refferenced (ORDR, RSLT or VLDT).

If the flat file does not contain any file errors and can be processed, a file of type FLAC will be sent back, including fileAccept message for this file with the attribute refType containing the type of the file being refferenced (ORDR, RSLT or VLDT).

### SOAP and HTTP-s

Each operator will provide a basic SOAP/HTTP server (shall be HTTP 1.1 and SOAP 1.1 compatible) to which other operators can connect.

An operator that wants to send a file to another operator uses a SOAP request sent by HTTP POST, calling a specific web service on the other operator’s server.

An operator that receives an HTTP post will check if the received file is complete (covered by SSL) and is stored.

The SOAP Response message is returned to confirm the transfer results.

The combination of HTTP and SSL makes the HTTP-s standard.

More information on HTTP 1.1 can be found at: [HTTP].

More information on SOAP 1.1 can be found at: [SOAP].

### SSL

Secure Sockets Layer (SSL) is a protocol providing privacy and reliability between two communicating applications.

The session key length shall be 128 bits (much more secure then using 40-bit encrypted communication).

SSL consists of three properties:

* Private connection by using encryption
* Authentication of identity by using public, cryptography
* Reliable connection by using message integrity check

More information on SSL can be found at: [SSL]

### TCP/IP

The Transmission Control Protocol (TCP) is intended for use as a highly reliable host-to-host protocol between hosts in packet-switched computer communication networks, and in interconnected systems of such networks.

### Fault handling

In the file exchange process there are two basic types of faults that might occur:

* Error in the file delivery
  + These error types may be further decomposed into more categories of which the most important in this process are:
    - SOAP faults (see chapter 4.4 in [SOAP] for more details)
    - HTTP faults (see chapter 6.1.1 in [HTTP] for more details)
    - SSL errors – problems with certificates (see [SSL] for more details)
  + If this sort of an error occurs, the request has to be repeated.
* Errors in the batch file being passed to the other site
  + Also these errors can be divided into two main categories:
    - File errors
    - Order/Message errors
  + See Chapter 4 Error handling of this document for the complete specification

# XML structure specification (order messages specification)

This chapter describes in detail the XML structure of the exchanged orders and other messages among operators within the ordering process. It’s divided into 3 parts. First two of them provides definitions of the messages involved in NP ordering, CPS description and the third and last one describes the constructions used within both NP and CPS ordering processes.

## NP messages specification

### npProvide

|  |  |
| --- | --- |
| **<npProvide>** | |
|  | |
| **Description** | Sent by the recipient operator to the losing operator to convey all initial details about an order. |
| **Parent** | Root |
| **Syntax** | <npProvide >  <recipientOperator>  <losingOperator>  <serviceContractOwner>  <installationAddress>  <customerReferenceNumber>  <icoNumber>  <directoryNumber>  <ddiRange>  **<npIDinfo>**  **<npIDattachment>**  <portActivationDate>  <portActivationTime>  </npProvide > |
| **Attributes** | OrderNr  SequenceNr  NpServiceType  ComplexOrder HasLLU |
| **DTD Source** | <!ELEMENT npProvide (recipientOperator, losingOperator,  serviceContractOwner, installationAddress,  customerReferenceNumber, icoNumber?,  (directoryNumber+|ddiRange+),  npIDinfo?, npIDattachment?,  portActivationDate, portActivationTime)>  <!ATTLIST npProvide  orderNr CDATA #REQUIRED  sequenceNr CDATA #REQUIRED  npServiceType (geog | non\_geog) #REQUIRED  complexOrder (yes | no) #REQUIRED  hasLLU (true | false) #IMPLIED > |
|  |  |

|  |  |
| --- | --- |
| <npIDinfo> | |
|  | |
| **Description** | Info about npID:   * NPID – číslo výpovědi u opouštěného operátora, předpokládáme strukurovanou textovou hodnotu * SPID\_OUT – číslo poskytovatele služby u opouštěného operátora (předpokládáme trojmístné číslo) * SPID\_IN – číslo poskytovatele služby u přijímajícího operátora (předpokládáme trojmístné číslo) |
| **Parent** | npProvide |
| **Syntax** | < npIDinfo >  <npID>  <spIDout>  <spIDin>  </npIDinfo > |
| **DTD Source** | <!ELEMENT npIDinfo (npID, spIDout?, spIDin? )>  <!—Informace o NPID--> |

|  |  |
| --- | --- |
| <npIDattachment> | |
|  | |
| **Description** | * Příloha – předpokládáme PDF nebo jiný podobný soubor, který bude pro potřeby XML zprávy zkonvertován v bytove podobe, v ramci konverze nebude ulozen nazev souboru, ten si pouzije kazdy operator podle sve vlastni potreby, zkonverovan bude pouze content. |
| **Syntax** | < npIDattachment >  <attachment>  </ npIDattachment > |
| **DTD Source** | <!ELEMENT npIDattachment (attachment)>  <!--attachment--> |

|  |  |
| --- | --- |
| <attachment> | |
|  | |
| **Description** | V tomto elementu bude primo dekodovany obsah pdf souboru zakodovana metodou base64 |
| **Syntax** | < attachment >  base64Binary datatype  </ attachment > |
| **DTD Source** | <!ELEMENT attachment (#PCDATA) > |

|  |  |  |
| --- | --- | --- |
| **<npID>** | | |
|  | | |
| **Description** | | OLO Code of losing operator |
| **Parent** | npIDinfo | |
| **Content** | | String datatype, alphanumeric, length = 20 |
| **Syntax** | | < npIDnumber >  string datatype  </ npIDnumber > |
| **Validation** | | The npID string (formatted value) must be provided |
| **DTD Source** | | <!ELEMENT npIDnumber (#PCDATA) #REQUIRED > |

|  |  |  |
| --- | --- | --- |
| **<**spIDout**>** | | |
|  | | |
| **Description** | | OLO Code of new operator |
| **Parent** | npIDinfo | |
| **Content** | | String datatype, numeric, length = 3 |
| **Syntax** | | < spIDout >  string datatype  </ spIDout > |
| **Validation** | | The npID string (formatted value) must be provided |
| **DTD Source** | | <!ELEMENT spIDin (#PCDATA) > |

|  |  |  |
| --- | --- | --- |
| **<**spIDin**>** | | |
|  | | |
| **Description** | | Description of validated NPID |
| **Parent** | npIDinfo | |
| **Content** | | String datatype, numeric, length = 3 |
| **Syntax** | | < spIDin >  string datatype  </ spIDin > |
| **Validation** | | The npID string (formatted value) must be provided |
| **DTD Source** | | <!ELEMENT spIDin (#PCDATA) > |

|  |  |
| --- | --- |
| <portActivationTime> | |
|  | |
| **Description** | Time when the NP port should be activated |
| **Syntax** | <portActivationTime>  <time>  </portActivationTime> |
| **DTD Source** | <!ELEMENT portActivationTime (time)>  <!--Cas aktivace portace--> |

|  |  |
| --- | --- |
| **<portActivationDate>** | |
|  | |
| **Description** | Date when the NP port should be activated |
| **Syntax** | <portActivationDate>  <date>  </portActivationDate> |
| **DTD Source** | <!ELEMENT portActivationDate (date)>  <!--Datum aktivace portace--> |

|  |  |
| --- | --- |
| **<directoryNumber>** | |
|  | |
| **Description** | The telephone number to be ported |
| **Content** | String datatype, alphanumeric, length = 9 |
| **Syntax** | <directoryNumber>  string datatype  </directoryNumber> |
| **Validation** | The complete (9 digit) line number must be provided |
| **DTD Source** | <!ELEMENT directoryNumber (#PCDATA)> |

### npAccept

|  |  |
| --- | --- |
| **<npAccept>** | |
|  | |
| **Description** | Sent by the losing operator to indicate positive validation and acceptance of the previous message |
| **Parent** | Root |
| **Syntax** | <npAccept >  <recipientOperator>  <losingOperator>  </npAccept > |
| **Attributes** | orderNr  sequenceNr  refSequenceNr |
| **DTD Source** | <!ELEMENT npAccept (recipientOperator, losingOperator)>  <!ATTLIST npAccept orderNr ID #REQUIRED  sequenceNr CDATA #REQUIRED  refSequenceNr CDATA #REQUIRED>  <!--Akceptace NP--> |

### npReject

|  |  |
| --- | --- |
| **<npReject>** | |
|  | |
| **Description** | Sent by the operator to indicate rejection of the referenced message |
| **Parent** | root |
| **Syntax** | <npReject>  <recipientOperator>  <losingOperator>  <rejectionCode>  <rejectionDescription>  <rejectionParameter>  </npReject > |
| **Attributes** | OrderNr  SequenceNr  RefSequenceNr |
| **DTD Source** | <!ELEMENT npReject (recipientOperator , losingOperator,  rejectionCode, rejectionDescription? ,  rejectionParameter\*)>  <!ATTLIST npReject orderNr ID #REQUIRED  sequenceNr CDATA #REQUIRED  refSequenceNr CDATA #REQUIRED>  <!--Odmitnuti NP --> |

### npChange

|  |  |
| --- | --- |
| **<npChange>** | |
|  | |
| **Description** | Sent by the recipient operator to change the porting date and/or time of an order |
| **Parent** | Root |
| **Syntax** | <npChange>  <recipientOperator>  <losingOperator>  <newPortDate>  <newPortTime>  </npChange> |
| **Attributes** | OrderNr  SequenceNr |
| **DTD Source** | <!ELEMENT npChange (recipientOperator , losingOperator ,  newPortDate , newPortTime)>  <!ATTLIST npChange orderNr ID #REQUIRED  sequenceNr CDATA #REQUIRED>  <!-- Zmena NP --> |

|  |  |
| --- | --- |
| **<newPortTime>** | |
|  | |
| **Description** | Request for new port time that is different from the original accepted port time |
| **Syntax** | <newPortTime>  <time>  </newPortTime> |
| **DTD Source** | <!ELEMENT newPortTime (time)>  <!—Novy cas portace--> |

|  |  |
| --- | --- |
| **<newPortDate>** | |
|  | |
| **Description** | Request for new port date, can be the same as original accepted port date if only the time changes.. |
| **Syntax** | <newPortDate>  <date>  </newPortDate> |
| **DTD Source** | <!ELEMENT newPortDate (date)>  <!--Nove datum portace--> |

### npCancel

|  |  |
| --- | --- |
| **<npCancel>** | |
|  | |
| **Description** | Sent by the recipient operator to cancel an order (e.g., PROVIDE) |
| **Parent** | root |
| **Syntax** | <npCancel>  <recipientOperator>  <losingOperator>  </npCancel > |
| **Attributes** | OrderNr  SequenceNr |
| **DTD Source** | <!ELEMENT npCancel (recipientOperator, losingOperator)>  <!ATTLIST npCancel orderNr ID #REQUIRED  sequenceNr CDATA #REQUIRED>  <!-- Zruseni NP--> |

### npAbort

|  |  |
| --- | --- |
| **<npAbort>** | |
|  | |
| **Description** | Sent by the losing operator under certain exception processing conditions |
| **Parent** | root |
| **Syntax** | <npAbort>  <recipientOperator>  <losingOperator>  </npAbort > |
| **Attributes** | orderNr  sequenceNr |
| **DTD Source** | <!ELEMENT npAbort (recipientOperator, losingOperator)>  <!ATTLIST npAbort orderNr CDATA #REQUIRED  sequenceNr CDATA #REQUIRED>  <!-- Storno NP --> |

### npPortComplete

|  |  |
| --- | --- |
| **<npPortComplete>** | |
|  | |
| **Description** | Message send from recipient operator indicating the successful completion of the number port. |
| **Parent** | root |
| **Syntax** | <npPortComplete>  <recipientOperator>  <losingOperator>  </npPortComplete > |
| **Attributes** | OrderNr  SequenceNr |
| **DTD Source** | <!ELEMENT npPortComplete (recipientOperator, losingOperator)>  <!ATTLIST npPortComplete orderNr ID #REQUIRED  sequenceNr CDATA #REQUIRED>  <!--Portace cisla NP dokoncena--> |

### npReturnNumber

|  |  |
| --- | --- |
| **<npReturnNumber>** | |
|  | |
| **Description** | Sent by the recipient operator to return a number to the donor operator following cessation of a number by a customer  Note: An npAccept or an npReject message to the npReturnNumber shall contain the donorOperator information within the losingOperator field. |
| **Parent** | root |
| **Syntax** | <npReturnNumber>  <recipientOperator>  <donorOperator>  <directoryNumber>  <ddiRange>  <serviceDisconnectionDate>  <serviceDisconnectionTime>  </npReturnNumber > |
| **Attributes** | OrderNr  SequenceNr |
| **DTD Source** | <!ELEMENT npReturnNumber (recipientOperator , donorOperator ,  (directoryNumber | ddiRange ),  serviceDisconnectionDate ,  serviceDisconnectionTime)>  <!ATTLIST npReturnNumber orderNr ID #REQUIRED  sequenceNr CDATA #REQUIRED>  <!--Vraceni cisla--> |

|  |  |
| --- | --- |
| **<serviceDisconnectionTime>** | |
|  | |
| **Description** | Time when directory number was taken out of service. |
| **Syntax** | <serviceDisconnectionTime>  <time>  </serviceDisconnectionTime> |
| **DTD Source** | <!ELEMENT serviceDisconnectionTime (time)>  <!--Cas vypojeni sluzby--> |

|  |  |
| --- | --- |
| **<serviceDisconnectionDate>** | |
|  | |
| **Description** | Date when directory number was taken out of service |
| **Syntax** | <serviceDisconnectionDate>  <date>  </serviceDisconnectionDate> |
| **DTD Source** | <!ELEMENT serviceDisconnectionDate (date)>  <!--Datum vypojeni sluzby--> |

### npSubsequentPort

|  |  |
| --- | --- |
| **<npSubsequentPort>** | |
|  | |
| **Description** | Sent by the recipient operator to the donor operator to indicate a change in number ownership  Note: An npAccept or an npReject message to the npSubsequentPort shall contain the donorOperator information within the losingOperator field. |
| **Parent** | root |
| **Syntax** | <npSubsequentPort>  <recipientOperator>  <donorOperator>  <losingOperator>  <directoryNumber>  <ddiRange>  </npSubsequentPort > |
| **Attributes** | OrderNr  SequenceNr |
| **DTD Source** | <!ELEMENT npSubsequentPort (recipientOperator, donorOperator ,  losingOperator ,  (directoryNumber | ddiRange))>  <!ATTLIST npSubsequentPort orderNr CDATA #REQUIRED  sequenceNr CDATA #REQUIRED>  <!--Nasledna portace NP--> |

### npIDcheck

|  |  |
| --- | --- |
| **<npIDcheck>** | |
|  | |
| **Description** | Sent by the recipient operator to the donor operator to validate combination of npID and phone number is correct |
| **Parent** | root |
| **Syntax** | <npIDcheck>  <recipientOperator>  <losingOperator>  <npIDnumber>  <directoryNumber>  </ npIDcheck > |
| **Attributes** | OrderNr  SequenceNr |
| **DTD Source** | <!ELEMENT npIDcheck (recipientOperator, losingOperator , npIDnumber, directoryNumber)>  <!ATTLIST npIDcheck orderNr CDATA #REQUIRED  sequenceNr CDATA #REQUIRED>  <!—kontrola spravnosti npID--> |

|  |  |  |
| --- | --- | --- |
| **<npIDnumber>** | | |
|  | | |
| **Description** | | Description of validated NPID – also knowns as “ČVOP – číslo výpovědi opouštěného poskytovatele” |
| **Parent** | npIDcheck | |
| **Content** | | String datatype, alphanumeric, length = 20 |
| **Syntax** | | < npIDnumber >  string datatype  </ npIDnumber > |
| **Validation** | | The npID string (formatted value) must be provided |
| **DTD Source** | | <!ELEMENT npIDnumber (#PCDATA) #REQUIRED > |

### npIDcheckResp

|  |  |
| --- | --- |
| **<npIDcheckResp>** | |
|  | |
| **Description** | Sent by the donor operator to the recipient operator as response to npIDcheck  Info from npIDcheck (npIDnumber, directoryNumber) are repetaed and key info about validation is in tag npIDcorrect |
| **Parent** | root |
| **Syntax** | <npIDcheckResp>  <recipientOperator>  <losingOperator>  < npIDnumber >  <directoryNumber>  <npIDcorrect>  </ npIDcheckResp > |
| **Attributes** | OrderNr  SequenceNr |
| **DTD Source** | <!ELEMENT npIDcheckResp (recipientOperator, losingOperator , npIDnumber, directoryNumber, npIDcorrect)>  <!ATTLIST npIDcheckResp orderNr CDATA #REQUIRED  sequenceNr CDATA #REQUIRED>  <!—odpoved, zda je komibnace npID a telefonu korektni--> |

|  |  |  |
| --- | --- | --- |
| **<npIDcorect>** | | |
|  | | |
| **Description** | | npID – result of validation |
| **Parent** | npIDcheckResp | |
| **Content** | | String datatype, alphanumeric, length = 1 |
| **Syntax** | | < npIDcorrect >  string datatype  </ npIDcorrect > |
| **Validation** | | The npIDcorrect string(Y = correct, N = incorrect) must be provided |
| **DTD Source** | | <!ELEMENT npIDcorrect (#PCDATA) >  <!ATTLIST npIDcorrect resp (Y|N) #REQUIRED> |

### recipientOperator, losingOperator, donorOperator

|  |  |
| --- | --- |
| **<recipientOperator>** | |
|  | |
| **Description** | Unique identification code specific to the recipient Operator |
| **Parent** | <npProvide>  <npAccept>  <npReject>  <npChange>  <npCancel>  <npPortComplete>  <npReturnNumber>  <npSubsequentPort> |
| **Syntax** | <recipientOperator>  <operatorID>  </recipientOperator > |
| **DTD Source** | <!ELEMENT recipientOperator ( operatorID)>  <!—Prejimajici operator--> |

|  |  |
| --- | --- |
| **<losingOperator>** | |
|  | |
| **Description** | Unique identification code specific to the losing Operator |
| **Parent** | <npProvide>  <npAccept>  <npReject>  <npChange>  <npCancel>  <npPortComplete> |
| **Syntax** | <losingOperator>  <operatorID>  </losingOperator> |
| **Field requirements** | Operator receiving this message is the losing operator |
| **DTD Source** | <!ELEMENT losingOperator ( operatorID)>  <!--Predavajici operator --> |

|  |  |
| --- | --- |
| **<donorOperator>** | |
|  | |
| **Description** | Unique identification code specific to the donor Operator |
| **Content** | Element only |
| **Parent** | <npReturnNumber>  <npSubsequentPort> |
| **Syntax** | <donorOperator>  <operatorID>  </donorOperator> |
| **DTD Source** | <!ELEMENT donorOperator ( operatorID)>  <!--Puvodni operator --> |

### DDI range

|  |  |
| --- | --- |
| **<ddiRange>** | |
|  | |
| **Description** | Number block range  The block size of a DDI can be either 10, 100, 1000, 10000 or 100000.  Appendix C describes the principles of DDI numbering. |
| **Parent** | <npProvide> |
| **Syntax** | <ddiRange >  <ddiRangeStart>  <ddiRangeEnd>  </ddiRange > |
| **Attributes** | None |
| **Message constraints** | <ddiRangeStart> is smaller then <ddiRangeEnd> |
| **DTD Source** | <!ELEMENT ddiRange ( ddiRangeStart,  ddiRangeEnd )>  <!--Rozsah provolby--> |

|  |  |
| --- | --- |
| **<ddiRangeStart>** | |
|  | |
| **Description** | Start of a number block range. |
| **Content** | String datatype, alphanumeric, length = 9 |
| **Parent** | <ddiRange> |
| **Syntax** | <ddiRangStart>  string datatype  </ddiRangeStart> |
| **Field requirements** | The <ddiRangeStart> always ends with 1 or more zero's  Example: if block size is 100 the <ddiRangeStart> is xxxxxxx00 |
| **DTD Source** | <!ELEMENT ddiRangeStart(#PCDATA)>  <!--Rozsah provolby – zacatek--> |

|  |  |
| --- | --- |
| **<ddiRangeEnd>** | |
|  | |
| **Description** | End of a number block range. |
| **Content** | String datatype, alphanumeric, length = = 9 |
| **Parent** | <ddiRange> |
| **Syntax** | <ddiRangEnd>  string datatype  </ddiRangeEnd> |
| **Field requirements** | The <ddiRangeEnds> always ends with 1 or more nine's  Example: if block size is 100 the <ddiRangeEnd> is xxxxxxx99 |
| **DTD Source** | <!ELEMENT ddiRangeEnd(#PCDATA)>  <!--Rozsah provolby – konec --> |

### NP Message attributes specification

|  |  |
| --- | --- |
| **npServiceType** | |
|  | |
| **Description** | Indicates the type of Number portability: Geographical or Non-Geographical  Typ sluzby np |
| **Content** | Possible values:   * geog * non\_geog |
| **Attribute spec** | npServiceType (geog | non\_geog ) #REQUIRED |

## CPS messages specification

### cpsProvide

|  |  |
| --- | --- |
| **<cpsProvide>** | |
|  | |
| **Description** | Sent by the CPS operator to the Access Provider to convey all initial details about an order |
| **Parent** | root |
| **Syntax** | <cpsProvide>  <cpsOperator>  <cscCode>  <serviceContractOwner>  <installationAddress>  <customerReferenceNumber>  <icoNumber>  <callingLineID>  </cpsProvide > |
| **Attributes** | OrderNr  SequenceNr  CpsServiceType  ComplexOrder pre |
| **DTD Source** | <!ELEMENT cpsProvide (cpsOperator , cscCode ,  serviceContractOwner ,  installationAddress ,  customerReferenceNumber,  icoNumber?, callingLineID)>  <!ATTLIST cpsProvide orderNr ID #REQUIRED  sequenceNr CDATA #REQUIRED  cpsServiceType (nat | intl | both )   pre (true | false) #IMPLIED  #REQUIRED  complexOrder (yes | no) #REQUIRED> |

|  |  |
| --- | --- |
| **<callingLineID>** | |
|  | |
| **Description** | CLI for which CPS will be used. |
| **Content** | String datatype, alphanumeric, min length = 4, max length = 9 |
| **Syntax** | <callingLineID>  string datatype  </callingLineID> |
| **Validation** | This field contains either the complete (9 digit) line number or the DDI Prefix (4 to 8 digits).  See appendix C for a description of the use of DDI prefix. |
| **DTD Source** | <!ELEMENT callingLineID (#PCDATA)>  <!—Identifikator volajici linky--> |

|  |  |
| --- | --- |
| **<cscCode>** | |
|  | |
| **Description** | Csc Code is the prefix to be used for CPS for this customer |
| **Content** | String datatype, numeric, min. length = 4, max. length = 5 |
| **Syntax** | <cscCode>  string datatype  </cscCode> |
| **Validation** | The CSC code is a 4 or 5 digit code. |
| **DTD Source** | <!ELEMENT cscCode (#PCDATA)>  <!--Identifikacni kod operatora--> |

### cpsReject

|  |  |
| --- | --- |
| **<cpsReject>** | |
|  | |
| **Description** | Sent by the Access Provider used to indicate rejection of the referenced message |
| **Parent** | Root |
| **Syntax** | <cpsReject>  <cpsOperator>  <rejectionCode>  <rejectionDescription>  <rejectionParameter>  </cpsReject > |
| **Attributes** | OrderNumber  SequenceNumber  RefSequenceNumber |
| **DTD Source** | <!ELEMENT cpsReject (cpsOperator, rejectionCode,  rejectionDescription?,  rejectionParameter\* )>  <!ATTLIST cpsReject orderNr CDATA #REQUIRED  sequenceNr CDATA #REQUIRED  refSequenceNr CDATA #REQUIRED>  <!--Odmitnuti CPS--> |

### cpsAccept

|  |  |
| --- | --- |
| **<cpsAccept>** | |
|  | |
| **Description** | Sent by the losing operator to indicate positive validation and acceptance of the previous message |
| **Parent** | Root |
| **Syntax** | <cpsAccept >  <cpsOperator>  </cpsAccept > |
| **Attributes** | orderNr  sequenceNr  refSequenceNr |
| **DTD Source** | <!ELEMENT cpsAccept (cpsOperator)>  <!ATTLIST cpsAccept orderNr ID #REQUIRED  sequenceNr CDATA #REQUIRED  refSequenceNr CDATA #REQUIRED>  <!--Akceptace CPS--> |

### cpsActivated

|  |  |
| --- | --- |
| **<cpsActivated>** | |
|  | |
| **Description** | Sent by the Access Provider to indicate order has been activated |
| **Parent** | Root |
| **Syntax** | <cpsActivated>  <cpsOperator>  <callingLineID>  </cpsActivated> |
| **Attributes** | OrderNr  SequenceNr |
| **DTD Source** | <!ELEMENT cpsActivated (cpsOperator, callingLineID)>  <!ATTLIST cpsActivated orderNr CDATA #REQUIRED  sequenceNr CDATA #REQUIRED> |

### cpsOperator

|  |  |
| --- | --- |
| **<cpsOperator>** | |
|  | |
| **Description** | Unique identification code specific to the CPS Operator |
| **Content** | Element only |
| **Parent** | <cpsProvide>  <cpsReject>  <cpsActivated> |
| **Syntax** | <cpsOperator>  <operatorID>  </cpsOperator> |
| **DTD Source** | <!ELEMENT cpsOperator ( operatorID)>  <!—Operátor CPS --> |

### CPS Message attributes specification

|  |  |
| --- | --- |
| **cpsServiceType** | |
|  | |
| **Description** | Indicates the type of Carrier Preselect: National, International or both  Typ sluzby CPS |
| **Content** | Possible values:   * nat * intl * both |

## Common messages specification

### header

|  |  |
| --- | --- |
| **<header>** | |
|  | |
| **Description** | Provides information on the routing, handling and processing of this flat file. |
| **Parent** | Root |
| **Syntax** | <header>  <toOperator>  <fromOperator>  <fileType>  <messagecount>  <timeSent>  <dateSent>  </header> |
| **Attributes** | None |
| **Message constraints** | None |
| **DTD Source** | <!ELEMENT header (toOperator, fromOperator, fileType, messagecount  timeSent, dateSent)> |

|  |  |
| --- | --- |
| **<toOperator>** | |
|  | |
| **Description** | Unique identification code of the operator receiving this flat file |
| **Content** | Element only |
| **Syntax** | <toOperator>  <operatorID>  </toOperator> |
| **Attributes** | None |
| **DTD Source** | <!ELEMENT toOperator (operatorID)> |

|  |  |
| --- | --- |
| **<fromOperator>** | |
|  | |
| **Description** | Unique identification code of the operator originating this flat file |
| **Content** | Element only |
| **Syntax** | <fromOperator>  <operatorID>  <fromOperator> |
| **Attributes** | None |
| **DTD Source** | <!ELEMENT fromOperator ( operatorID)> |

|  |  |
| --- | --- |
| **<fileType>** | |
|  | |
| **Description** | Code for flat file type |
| **Content** | string datatype, alphanumeric, length = 4  possible values:  ORDR– provide  VLDT - validate  RSLT – result  FLAC - File acknowledge |
| **Syntax** | <fileType>  string datatype  <fileType> |
| **Attributes** | None |
| **DTD Source** | <!ELEMENT fileType (#PCDATA)> |

|  |  |
| --- | --- |
| **<messageCount>** | |
|  | |
| **Description** | Shows the number of messages in this file |
| **Content** | string datatype, numeric, length = 4, leading zero’s |
| **Syntax** | <messageCount>  string datatype  </messageCount> |
| **Attributes** | None |
| **DTD Source** | <!ELEMENT messageCount(#PCDATA)> |

|  |  |
| --- | --- |
| **<timeSent>** | |
|  | |
| **Description** | Time this flat file was sent by originating operator  If an operator sends a second file during the day this MUST have a different <timeSent> !! |
| **Content** | Element only |
| **Syntax** | <timeSent>  <time>  </timeSent> |
| **DTD Source** | <!ELEMENT timeSent (time)> |

|  |  |
| --- | --- |
| **<dateSent>** | |
|  | |
| **Description** | Date this flat file was sent by originating operator |
| **Content** | Element only |
| **Syntax** | <dateSent>  <date>  </dateSent> |
| **DTD Source** | <!ELEMENT dateSent (date)> |

### fileReject

|  |  |
| --- | --- |
| **<fileReject>** | |
|  | |
| **Description** | Sent by an operator to indicate rejection of the referenced file |
| **Parent** | root |
| **Syntax** | <fileReject>  <refTimeSent>  <refDateSent>  <rejectionCode>  <rejectionDescription>  </fileReject > |
| **Attributes** | None |
| **DTD Source** | <!ELEMENT fileReject (refTimeSent, refDateSent, rejectionCode,  rejectionDescription)> |

### fileAccept

|  |  |
| --- | --- |
| **<fileAccept>** | |
|  | |
| **Description** | Sent by an operator to indicate acceptation of the referenced file |
| **Parent** | root |
| **Syntax** | <fileAccept>  <refTimeSent>  <refDateSent>  </fileAccept > |
| **Attributes** | None |
| **DTD Source** | <!ELEMENT fileAccept (refTimeSent, refDateSent)> |

|  |  |
| --- | --- |
| **<refTimeSent>** | |
|  | |
| **Description** | Refers to <timeSent> of the rejected/accepted flat file |
| **Content** | Element only |
| **Syntax** | <refTimeSent>  <time>  </refTimeSent> |
| **DTD Source** | <!ELEMENT refTimeSent (time)> |

|  |  |
| --- | --- |
| **<refDateSent>** | |
|  | |
| **Description** | Refers to <dateSent> of the rejected/accepted flat file |
| **Content** | Element only |
| **Syntax** | <refDateSent>  <date>  </refDateSent> |
| **DTD Source** | <!ELEMENT refDateSent (date)> |

### serviceContractOwner

|  |  |
| --- | --- |
| **<serviceContractOwner>** | |
|  | |
| **Description** | Contains the basic information on the owner of this contract |
| **Parent** | <npProvide>, <cpsProvide>, |
| **Syntax** | <serviceContractOwner>  <ownerSurname>  <ownerFirstName>  <ownerCompanyName>  </serviceContractOwner > |
| **Attributes** | None |
| **DTD Source** | <!ELEMENT serviceContractOwner ((ownerSurname, ownerFirstName) |  ownerCompanyName)>  <!--Vlastnik kontraktu na sluzbu--> |

|  |  |
| --- | --- |
| **<ownerSurname>** | |
|  | |
| **Description** | Surname of Contract Owner or name on CAF |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 110 |
| **Parent** | <serviceContractOwner> |
| **Syntax** | <ownerSurname>  string datatype  </ownerSurname > |
| **Attributes** | None |
| **DTD Source** | <!ELEMENT ownerSurname (#PCDATA)>  <!--Prijmeni vlastnika--> |

|  |  |
| --- | --- |
| **<ownerFirstname>** | |
|  | |
| **Description** | Firstname of Contract Owner or name on CAF |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 32 |
| **Parent** | <serviceContractOwner> |
| **Syntax** | <ownerFirstname>  string datatype  </ownerFirstname > |
| **Attributes** | None |
| **DTD Source** | <!ELEMENT ownerFirstName (#PCDATA)>  <!--Krestni jmeno vlastnika--> |

|  |  |
| --- | --- |
| **<ownerCompanyname>** | |
|  | |
| **Description** | Company name of Contract Owner or name on CAF |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 32 |
| **Parent** | <serviceContractOwner> |
| **Syntax** | <ownerCompanyname>  string datatype  </ownerCompanyname > |
| **Attributes** | None |
| **DTD Source** | <!ELEMENT ownerCompanyName(#PCDATA)>  <!--Nazev organizace--> |

### installationAddress

|  |  |
| --- | --- |
| **<installationAddress>** | |
|  | |
| **Description** | Contains the basic information on the installation address  See appendix for references on addressing standards |
| **Parent** | <npProvide>, <cpsProvide>, |
| **Syntax** | <installationAddress>  <houseNumberA>  <houseNumberB>  <streetName>  <city>  <ciySection>  <district>  <postCode>    </installationAddress > |
| **Attributes** | None |
| **Constraints** | Either the A or B house number is required.  Both are allowed.  postcode is manadatory for NP only |
| **DTD Source** | <!ELEMENT installationAddress (((housenumberA, housenumberB) |  housenumberA | housenumberB),  streetName, city, citySection?,  district, postCode?)>  <!--Adresa instalace-->  Note: The construction ((housenumberA, housenumberB) | housenumberA | housenumberB) is not a deterministic model. |

|  |  |
| --- | --- |
| **<housenumberA>** | |
|  | |
| **Description** | House/Residence number (e.g., 5)  This is the number of the house within the street, the blue housenumber. |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 5 |
| **Syntax** | <housenumberA>  string datatype  </housenumberA> |
| **DTD Source** | <!ELEMENT housenumberA (#PCDATA)>  <!-- Cislo popisne mista instalace --> |

|  |  |
| --- | --- |
| **<housenumberB>** | |
|  | |
| **Description** | This is the location orientation number, the number that appears on the red plate.  Cislo orientacni adresy umisteni pripojeni. |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 8 |
| **Syntax** | <housenumberB>  string datatype  </housenumberB> |
| **DTD Source** | <!ELEMENT housenumberB (#PCDATA)>  <!--Cislo orientacni mista instalace--> |

|  |  |
| --- | --- |
| **<streetName>** | |
|  | |
| **Description** | Name of the street the address is situated in (e.g, Francouzska) |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 64 |
| **Syntax** | <streetName>  string datatype  </streetName> |
| **DTD Source** | <!ELEMENT streetName (#PCDATA)>  <!-- Nazev ulice mista instalace --> |

|  |  |
| --- | --- |
| **<city>** | |
|  | |
| **Description** | Name of the city where the address is situated in (e.g, Praha) |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 64 |
| **Syntax** | <city>  string datatype  </city> |
| **DTD Source** | <!ELEMENT city (#PCDATA)>  <!-- Obec instalace --> |

|  |  |
| --- | --- |
| **<citySection>** | |
|  | |
| **Description** | Name of the part of the city where the address is situated in (e.g, Praha 2) |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 64 |
| **Syntax** | <citySection>  string datatype  </citySection> |
| **DTD Source** | <!ELEMENT citySection(#PCDATA)>  < !-- Obec instalace --> |

|  |  |
| --- | --- |
| **<district>** | |
|  | |
| **Description** | Name of the location district where the address is situated |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 64 |
| **Syntax** | <district>  string datatype  </district> |
| **DTD Source** | <!ELEMENT district (#PCDATA)>  <!--Uctovaci oblast--> |

|  |  |
| --- | --- |
| **<postCode>** | |
|  | |
| **Description** | Installation address postal area (e.g. 736 01) |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 8 |
| **Syntax** | <postCode>  string datatype  </postCode> |
| **DTD Source** | <!ELEMENT postCode (#PCDATA)>  <!--PSC dodaci posty--> |

### Operator ID

|  |  |
| --- | --- |
| **<operatorID>** | |
|  | |
| **Description** | Unique identification code specific to an operator or service provider.  The numbering plan of OperatorID will be in responsibility of CTU.  CTU will assign to each and every operator with already assigned range of DDI (based on public network numbering plan) one code from range from 200 to 999. |
| **Content** | String datatype, numeric,length = 3,  Examples:  212 = Alliatel  202 = O2 Czech Republic  234 = GTS  224 = Ceske RadioKomunikace  …etc |
| **Parent** | <recipientOperator,>  <losingOperator>  <donorOperator>  <cpsOperator>  <to>  <from> |
| **Syntax** | <operatorID>  string datatype  </operatorID > |
| **DTD Source** | <!ELEMENT operatorID (#PCDATA)>  <!-- Identifikátor operátora--> |

### Customer information

|  |  |
| --- | --- |
| **<customerReferenceNumber>** | |
|  | |
| **Description** | Number that appears on the customer bill  ( Referencni cislo platce ) |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 15 |
| **Parent** | <npProvide>  <cpsProvide> |
| **Syntax** | <customerReferenceNumber>  string datatype  </customerReferenceNumber |
| **DTD Source** | <!ELEMENT customerReferenceNumber(#PCDATA)>  <!--Referencni cislo platce --> |

|  |  |
| --- | --- |
| **<icoNumber>** | |
|  | |
| **Description** | This is the unique identification of the business customer, that appears on the customer bill that was send out by CETIN  In the case of a business customer this field is required.  Note: if there is no ico number available for a business customer this is not a reason to reject the order |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 15 |
| **Parent** | <npProvide>  <cpsProvide> |
| **Syntax** | <icoNumber>  string datatype  </icoNumber |
| **DTD Source** | <!ELEMENT icoNumber (#PCDATA)>  <!—-ICO--> |

### Date and time

|  |  |
| --- | --- |
| **<date>** | |
|  | |
| **Description** | Standard element to describe date.  Definition=CCYY-MM-DD where “CC“ stands for century, “YY” for year, "MM" for month and "DD" for day-->  Example = to indicate 31 May 2002, you type <date>2002-05-31</date> |
| **Content** | sting datatype, alphanumeric, length = 10 |
| **Parent** | <newPortDate>  <portActivationDate>  <dateSent> |
| **Syntax** | <date>  string datatype  </date> |
| **Field requirements** | Date must be valid calendar date, so not 2002-13-32 |
| **DTD Source** | <!ELEMENT date (#PCDATA)>  <!—datum--> |

|  |  |
| --- | --- |
| **<time>** | |
|  | |
| **Description** | Standard element to describe time  Definition= hh:mm:ss "hh", "mm", "ss" represent hours, minutes and seconds, respectively.    Example=to indicate 1.20 p.m. enter <time>13:20:00</time>  Time is always Central European Time, so the time in Czech republic.  It is not possible to define the time zone. |
| **Content** | sting datatype, alphanumeric, length = 8 |
| **Parent** | <newPortTime>  <portActivationTime>  <timeSent> |
| **Syntax** | <time>  string datatype  </time> |
| **Field requirements** | Time must be a valid time, so not 25:61 |
| **DTD Source** | <!ELEMENT time (#PCDATA)>  <!—Cas--> |

### Rejection

|  |  |
| --- | --- |
| **<rejectionCode>** | |
|  | |
| **Description** | Code indicating the reason why a message or file was rejected.  See the chapter on error handling for a list of codes and their description |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 4 |
| **Parent** | <npReject>, <cpsReject>, <fileReject>, |
| **Syntax** | <rejectionCode>  string datatype  </rejectionCode> |
| **DTD Source** | <!ELEMENT rejectionCode(#PCDATA)>  <!--Kod odmitnuti--> |

|  |  |
| --- | --- |
| **<rejectionDescription>** | |
|  | |
| **Description** | Description of the reason why a message was rejected.  See the chapter on error handling for a list of codes ant there description |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 50 |
| **Parent** | <npReject>, <cpsReject>, <fileReject>, |
| **Syntax** | <rejectionDescription>  string datatype  </rejectionDescription> |
| **DTD Source** | <!ELEMENT rejectionDescription (#PCDATA)> |

|  |  |
| --- | --- |
| **<rejectionParameter>** | |
|  | |
| **Description** | Additional parameter to indicate the reason why a message was rejected. |
| **Content** | String datatype, alphanumeric, min length = 1, max length = 200 |
| **Parent** | <npReject>, <cpsReject> |
| **Syntax** | <rejectionParameter>  string datatype  </rejectionParameter> |
| **DTD Source** | <!ELEMENT rejectionParameter(#PCDATA)> |

### Message attributes specification

Attributes provide a means of associating simple properties with elements.

|  |  |
| --- | --- |
| **orderNr** | |
|  | |
| **Description** | Unique identification for each individual order of an operator  Each individual NP PROVIDE, CPS PROVIDE or RETURN NUMBER message has a new order number. All other messages refer to existing  ordernumbers. |
| **Content** | NP,CPS Orders:  String datatype  Position 1 O  Positon 2 to 9: numeric, pending zero’s  Example: O00000012  Note: the O is needed for correct usage of the XML id field  Note: the O is needed for correct usage of the XML id field  IF NP,CPS,orders are sent in one ORDR file, Orders with O sequence first.  Example how mixed NP,CPS,orders are sent in ORDR file:  O00000012  O00000013  O00000014 |
| **Field constrains** | The orderNr must have an ascending sequence, except if a maximum is reached, and a rollover is required.  Gaps are allowed.  It is possible that two operators use the same ordernumber, only the combination with recipientOperatorID or cpsOperator makes it unique! |
| **Attribute spec** | orderNr ID #REQUIRED  Note:  The ID type makes the orderNr unique throughout the XML document (so the whole file).  In the case of Subsequent Port CDATA is used, to enable multiple of these message types on one  OrderNr within an XML document. In the case of npAbort CDATA is used, to enable an npReject and npAbort on the same order in one file. |

|  |  |
| --- | --- |
| **sequenceNr** | |
|  | |
| **Description** | Each message for a single order has the same order number but has unique sequence number.  Poradove cislo zpravy |
| **Content** | Sring datatype, numeric, min length = 1, max length = 2 |
| **Attribute requirements** | The sequence number must be in ascending order with the prior message with the same order number sent by the same operator.  Gaps are allowed.  Each operator assigns its own sequence numbers independently of other operator(s) involved in the ordering process i.e. each operator is using its own number sequence. |
| **Attribute spec** | sequenceNr CDATA #REQUIRED |

|  |  |
| --- | --- |
| **refSequenceNr** | |
|  | |
| **Description** | Refers to the SequenceNumber of the message that is rejected or accepted  Referencni poradové cislo |
| **Content** | String datatype, numeric |
| **Attribute spec** | refSequenceNr CDATA #REQUIRED |

|  |  |
| --- | --- |
| **complexOrder** | |
|  | |
| **Description** | Indicates if this specific order is a complex or a simple order  Slozita objednavka |
| **Content** | Possible values: yes, no |
| **Attribute spec** | ComplexOrder (yes | no) #REQUIRED |

|  |  |
| --- | --- |
| **refType** | |
|  | |
| **Description** | Contains the reference to the file type being referenced by respective fileAccept/fileReject message. |
| **Content** | Possible values: ORDR, RSLT, VLDT |
| **Attribute spec** | refType (ORDR | RSLT | VLDT) #REQUIRED |

# Error handling

## Error handling principles

1. If the received file can not be processed due to file errors, a file of the type 'File acvknowledge/reject' will be sent back by start of business day 1 containing a fileReject message
2. Service orders over the access provider’s or losing operator’s daily order processing capacity (Quota rejection) will be rejected with a cpsReject or npReject message by   
   Start of Business Day 1.
3. When there is an error in the order, the operator shall be sent a cpsReject or npReject message.  
   Error orders shall be detected and rejected by Day 2 (two business days from when service order request is sent).
4. The validation file sent at 09.00 will contain orders rejected due to validation errors, however only those where order validation is completed before the submission of the file. Rejected orders processed after the submission of this file will be sent either by 17:00 on Day 1 or at the latest by 18:00 on Day 2.
5. Errors which are a result of the technical investigation of an NP order (errors in the directory number/DDI range) will be reported by Day from Referenc offer CETIN..
6. If the contract cancellation CAF for an NP order is not received by the losing operator within business days descripted on Referenc offer CETIN the respective NP order is rejected.
7. If multiple mandatory parameters are missing then all missing mandatory parameters will be identified in the npReject or cpsReject message. If no mandatory parameters are missing then all syntax related errors will be identified in the npReject or cpsReject message.   
   If there are no syntax errors in the NP order then all DNs will be verified and identified in the npReject message in case of any problems.  
   Otherwise, the npReject or cpsReject message shall only identify the first parameter/field that contains an error



### Rejection categories

The rejections are grouped in the following categories:

* File format general problems with the received file
* Quota quota is exceeded
* Field constraints error(s) in individual fields
* Operator operator not valid
* Logic of order orders not in line with agreed principles
* Customer customer data in message incorrect
* CAF Customer cancellation form
* Directory number/DDI number data in message incorrect

### Rejection codes

The following pages contain a list of rejections possible,

The list contains the following columns:

* Constraints/requirements the constraint this rejection code can be a result of
* Message the message(s) this constraint applies to
* Rejection code code indicating rejection reason
* Rejection description description of the rejection reason
* Rejection parameter parameter(s) which were rejected

The rejection code’s first letter indicates the category.   
  
NP Rejection codes table

| **Constraints/requirements** | **Message** | **Rejection** | | |
| --- | --- | --- | --- | --- |
| **Code** | **Description** | **Parameter** |
| **File format** | | | | |
| The file must be a well-formed XML document, and in line with the DTD. | flat file | X001 | XML not well formed or DTD error | none |
| The content of the <header> must be correct. | flat file | X002 | Header error | none |
| The systems date of the file on the HTTP server of the operator receiving a file must be equal to the date in the header/filename. | flat file | X003 | Wrong date | none |
| The flat file can only contain the message-types specified. | flat file | X004 | Unexpected message | none |
|  | | | | |
| **Quota** | | | | |
| An individual service order message cannot exceed the access providers’ order processing capacity.  *Note: the order quotas principles apply, as described in the PC document.* | cpsProvide  npProvide  npChange | Q001 | Order quota exceeded  (for each individual order) | none |
|  | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field constraints** | | | | |
| Mandatory fields  *(If multiple mandatory fields are missing, the parameter field will contain all)* | All | F001 | Mandatory Field missing | Message tag of field(s) that is missing, |
| Multiplicity constraints  *(If multiple fields have a constraint error, the parameter field will contain all fields)* | All | F002 | Cardinality error | Message tag of field(s) |
| Field syntax constraints  *(If multiple fields have a syntax error, the parameter field will contain all fields)*  *(Note: Field syntax constraints are specified in the individual field specifications)* | All | F003 | Incorrect, invalid or incomplete data | Message tag or field(s) |
|  | | | | |
| **Operator** | | | | |
| The operator must be a registered operator (recipient, losing, donor, CPS operator,) | All | O001 | Operator unknown | OperatorId |
| A recipient operator must have a NP agreement with the donor operator of the number involved or agreement on the service. | cpsProvide  npProvide | O002 | No service establishment | OperatorID |
| The CSC code provided in the message must be a registered CSC code | cpsProvide | O003 | CSC code invalid | The CSC code provided in the message must be a registered CSC code |
|  | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logic of order messages** | | | | |
| The <order nr> in a provide message must be in ascending order with previous order (but gaps are allowed). | cpsProvide  npProvide | L001 | Out of sequence ordernumber | none |
| The <sequence number> in a message must be in ascending order with the previous message | All | L002 | Sequence number out of sequence | none |
| Reference is only possible to an existing order (so no accept before provide) | All but provide | L003 | Inappropriate message | none |
| The <portActivationTime> in a <npProvide> message must be between 6 am and 10 pm | npProvide | L004 | Port time invalid | none |
| The <portActivationDate> in a <npProvide> message must be on day 15 (15 business days from receipt of the npProvide).  Exception: if the <portActivationDate> is a Saturday or Sunday, Day 15 must be the last business day of the week (in most cases Friday) | npProvide | L005 | Port date invalid | none |
| The <donorOperator> receiving a <subsequentPort> must correct. | npSubsequentPort | L006 | Sent to wrong operator | none |
| An <npChange> and <npCancel> message must be received at least four business days before current port date | npChange | L007 | npChange/npCancel received too late | none |
| The <newPortDate> in a <npChange> must be exactly 15 business days after the date this message was sent (<dateSend> in header).  Exception: if the <portActvationDate> is a Saturday or Sunday, Day 15 must be the last business day of the week (in most cases ftiday) | npChange | L008 | new Port date invalid | none |
| The <newPorttime> in a NP Change must be later then current port time and be between 6 am and 10 pm | npChange | L009 | New Port time older then current port time | none |
| The order shall be rejected if the request is for a GNP port of a non-geographic number and vice versa | npProvide | L010 | Service type incorrect | none |
| The <complex order> indicator in a provide must be in line with the actual content of the order | cpsProvide  npProvide | L011 | Order complexity incorrect | none |
| The <donorOperator> receiving a <npReturnNumber> must be correct. | npReturnNumber | L012 | Sent to wrong operator | none |
|  | | | | |
| **Customer** | | | | |
| The customer must belong to the losing operator or access provider receiving a message. | cpsProvide  npProvide | C001 | Customer not known | none |
| The customer must be allowed to port (as defined in PC) | npProvide | C002 | Customer not subjected to NP | none |
| The <callinglineID> in a message must be applicable to CPS (as defined in [VPNPC]) | cpsProvide | C003 | Number not subjected to CPS | none |
| The customer must be allowed to subscribe to CPS (as defined in [VPNPC]) | cpsProvide | C004 | Customer not subjected to CPS | none |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | | | | |
| **Customer CAF** | | | | | |
| The customer contract cancellation CAF for a NP Provide must be received by day 10 (10 business days from receipt of the message) | npProvide | | C005 | Missing customer cancellation CAF | none |
| The customer contract cancellation CAF content must be consistent with <npProvide> message. | npProvide | | C006 | Inconsistent information between PROVIDE and contract cancellation CAF | none |
| The customer contract cancellation is incorect. | npProvide | | C007 | Incorrect CAF or CVOP | none |
| **Directory Number/DDI** | | | | | |
| The <directoryNumber> or <ddiRange> in a provide must belong to the losing operator | npProvide | N001 | | Sent to wrong operator | none |
| The <directoryNumber> or <ddiRange> in a npProvide message may not be already exported | npProvide | N002 | | Number has already been ported | directory number/DDI range |
| The <directoryNumber> or <ddiRange> in a npProvide message must be applicable to NP | npProvide | N003 | | Number not subjected to NP | directory number/DDI range |
| An individual <directoryNumber> of a customer can not be ported out of an existing DDI range. | npProvide | N004 | | Cannot port individual number out of DDI range | directory number |
| The <directoryNumber> or <ddiRange> in a npProvide must be correct | npProvide | N005 | | directoryNumber or DDI range incorrect | directory number/DDI range |
| Chybí MSN číslo | npProvide | N006 | |  | None |
| Chybné MSN číslo | npProvide | N007 | |  | None |
| Duplicitní objednávka | npProvide | N008 | |  | None |
| Chybí LLU Provide | npProvide | N009 | |  | None |

# APPENDIX A: OUTAGE SITUATIONS (NP AND CPS)

The following business rules shall apply when an operator experiences an outage of their HTTPS server:

If the outage is resolved prior to 3:00 PM, or lasts less than 30 minutes, it shall be the responsibility of other operators to re-attempt transmission of their batch files at a later time during the day.

If the outage occurs after, or extends beyond, 3:00 PM and lasts for more than 30 minutes, the operator shall extend the operating window of its server for two hours beyond COB. Other operators shall then be responsible for re-attempting their batch file transmissions during this extended period.

If the operator is unable to restore the server's operation and comply with the two-hour extension requirement, the operator shall adjust its industry quota limit to assist in clearing the order backlog. The following business rules shall apply for this process:  
- The operator shall increase its overall industry quota limit by five percent for the next 20 business days.  
- If the outage spans multiple days, the five percent industry quota increase shall extend an additional 20 business days for each additional day of outage.  
- It shall be the responsibility of other operators to effectively use this increased quota capacity to satisfy their business needs.

The operator will announce the extended operating window or adjustment of the quota by a FAX message.

# APPENDIX B: INSTALLATION ADDRESS REFERENCES (NP AND CPS)

The following two references describe standardisation of addresses in the Czech Republic:

**A Číselníky České pošty na: http://www.ceskaposta.cz/**

1. Přehled adresních míst

Jedná se o soustavu níže uvedených číselníků jednoznačně identifikujících dodací místo pro adresování zásilek. Patří sem číselníky:

* Okresy s údaji: kód okresu a název okresu,
* Seznam obcí a jejich částí s údaji: kód okresu, kód obce, název obce, kód části obce, název části obce, využitelné PSČ v obci, resp. v části obce,
* Ulice obsahující přehled všech ulic v obcích a jejich částech v ČR s údaji: kód obce, kód části obce, kód ulice, název ulice, využitelné PSČ v ulici (u ulic, které mají více využitelných PSČ než jedno je položka PSČ nevyplněna), původní název ulice, městský obvod,
* Adresy s údaji: kód obce, kód části obce, kód ulice, kód adresy, číslo popisné/evidenční, číslo orientační, PSČ adresy, číselník. Přehled pošt v ČR s údaji: název pošty, PSČ, kód a název okresu.

2. PSČ organizací

Soubor obsahuje tyto údaje: PSČ přidělená vybraným organizacím, název organizace, číslo přihrádky nebo adresa, název dodávací pošty, kód okresu.

3. PSČ v ulici I

Soubor obsahuje seznam všech PSČ ulic, v nichž je evidována alespoň jedna adresa. Struktura číselníku: kód okresu, kód obce, název obce, kód části obce, název části obce, kód ulice, název ulice, využitelné PSČ. (vyplněno pouze u ulic, které mají pouze jedno využitelné PSČ)

4. PSČ v ulici II

Soubor obsahuje seznam adres v ulicích, u nichž je užito více než jedno PSČ.

Struktura číselníku: kód okresu, kód obce, název obce, kód části obce, název části obce, kód ulice, název ulice, kód adresy, číslo popisné/evidenční, číslo orientační, využitelné PSČ v ulici.

5. PSČ v ulici III

Soubor obsahuje výběr ze seznamu ulic a k nim příslušejících PSČ, kdy v jedné obci existuje více PSČ, ale pro celou ulici platí jediné PSČ. Struktura číselníku: kód okresu, kód obce, název obce, kód části obce, název části obce, kód ulice, název ulice, využitelné PSČ v ulici.

**B, MINISTERSTVO PRO MÍSTNÍ ROZVOJ: http://www.mmr.cz/:**

Územně identifikační registr základních sídelních jednotek

Vyhledávání dle názvu obce a dle okresu.

# APPENDIX C: DDI numbering

## Detailed Number rules for Ordering CPS over DDI services

The new numbering plan for the Czech Republic consists of 9 numbers since 22nd September 2002)

2

7

1

4

2

5

4

4

3

NDC

Prefix

For the purposes of representing a DDI Prefix in a CPS order the number has the following structure (from left to Right):

1. National Destination Code (NDC) is the first 1 or 2 digits
2. Prefix is the next 2 to 7 Digits

The remaining digit(s) should not be presented in the DDI Prefix in a CPS order.

CETIN provides DDI ranges in block sizes of 10, 100, 1000, 10000 and 100000 numbers. Therefor a DDI prefix in a CPS order must be between 4 and 8 digits long.

Example 1

A customer in Prague has purchased a DDI range of 100 numbers, the whole range is active on the network. A valid dialling sequence to the customer would consist of:

NDC = 2

Prefix = 234567

Remaining Digits 00 to 99

The DDI Prefix in a CPS order should be seven digits long as follows: 2234567

Example 2

A customer in Prague has purchased a DDI range of 10000 numbers, the whole range is active on the network. A valid dialling sequence to the customer would consist of:

NDC = 2

Prefix = 4567

Remaining Digits 0000 to 9999

The DDI Prefix in a CPS order should be five digits long as follows: 24567

## Sub-Ranges and The Customer’s Active DDI Range

Customer’s may ask for the whole range or for sub-ranges to be activated. If only a sub-range is active then a DDI block Id is used to identify the active blocks. A DDI block id a single digit ‘0’ through ‘9’. A service could have several sub-ranges active.

The DDI prefix in a CPS order should be the NDC, Prefix and DDI Block Id of one of the active DDI sub-ranges.

Example 3

A customer in Prague has purchased a DDI range of 1000 numbers but only 3 sub-ranges are active. The table below shows the active ranges.

| NDC | Prefix | DDI Block Id | Remaining Digits |
| --- | --- | --- | --- |
| 2 | 56789 | 0 | Not Active |
| 2 | 56789 | 1 | 00 to 99 |
| 2 | 56789 | 3 | Not Active |
| 2 | 56789 | 4 | 00 to 99 |
| 2 | 56789 | 5 | Not Active |
| 2 | 56789 | 6 | 00 to 99 |
| 2 | 56789 | 7 | Not Active |
| 2 | 56789 | 8 | Not Active |
| 2 | 56789 | 9 | Not Active |

A valid dialling sequence to the customer would consist of:

NDC = 2

Prefix = 56789

DDI block Id = 1 or 4 or 6

Remaining Digits 00 to 99

The DDI Prefix in a CPS order should be seven digits long as follows: 2567891 or 2567894 or 2567896. Any one of the DDI prefixes shown will result in the whole of the service having CPS applied. An order showing any other number as the DDI prefix would be rejected as not a valid number.

# APPENDIX E: DTD DISTRIBUTION AND CHANGE MANAGEMENT (NP AND CPS)

The DTD will be stored on a secured WEB server, which can only be accessed (read only) by licenced operators.

One of the menbers of APVTS will control the access rights.

**Change management principles**

1. The inter-operator interface document and the DTD file will be maintained by CETIN.
2. Operators can issue change requests on the document/DTD to CETIN via APVTS meetings
3. CETIN will analyze the impact of the change request on the interface specification.
4. The received change requests + analysis will be distributed among operators. All operators will analyze the impact on their specific interface.  
   A regular APVTS meeting (frequency to be decided) will decide if the change request is accepted, and in which version of the document/DTD the change will be implemented
5. It is proposed to have a new version of the document/DTD no more then two times a year
6. Anouncements of new versions of the DTD will be sent via e-mail to all licensed operators with access to the DTD file on the APVTS web server.
7. The DTD will get a version number in the header (first version: 1.0)
8. The implementation/and testing of the new interface document/DTD will be specifically planned for each new version.
9. According to the plan described in 7) a new version of the DTD will be installed on the APVTS web server
10. Both the document and the DTD will hold a change log

# APPENDIX F: HANDLING FRAUDULENT ORDERS (NP AND CPS)

The following business rules shall apply for detecting and responding to fraudulent CPS orders:

The batch file acknowledgment sent by the Access Provider shall be used by the CPS operator to detect fraudulent transmissions of batch files. If a CPS operator receives a batch file acknowledgment that it was not expecting, it shall be the responsibility of the CPS operator to detect these fraudulent transmissions and immediately notify the Access Provider.

The ACTIVATE message sent by the Access Provider shall be used by the CPS operator to detect fraudulent insertion of individual CPS orders into a batch file. If a CPS operator receives an ACTIVATE message that cannot be matched to a corresponding PROVIDE message, it shall be the responsibility of the CPS operator to detect these fraudulent orders and immediately notify the Access Provider.

In both of the above cases, the Access Provider shall attempt to reverse the fraudulent orders.

# APPENDIX G: BRIEF EXPLANATION ON DTD NOTATION (NP AND CPS)

A Document Type Definition (DTD) describes the grammar of the language defined, and ensures that a document can provide a parser with meta-information about the content of the document. Meta-information defines the permitted sequence and nesting of tags, attribute values with their types and default values, the names of external files which may be referred to, formats of external (non-XML) data that can be used, and entities.

Products of the W3C XML Activity can be found on www.w3.org/xml.

A DTD can be explicitly written into the header of an XML document or can consist of a reference. A combination of reference and inclusion in the header is also possible.

**Element declarations.**

These indicate the names and content type of the elements. For example:

<!ELEMENT book (title, subtitle?, author+)>

This example indicates that the element ‘book’ consists of the content: title, subtitle (where relevant) and one or more authors. The elements named in the content of an element must also be specifically declared as elements (in this case: title, subtitle and author).

Instead of the ', 'separator between elements also a ' | ' can be used, to indicate a choice.

**Entity declarations**.

These can be used to associate a name with another fragment in the document; they are actually macros, therefore. Entity declaration can involve a piece of regular text (abbreviation), part of the document type declaration (import), or a reference to an external file that contains text or binary data. A number of examples:

<!ENTITY ATI “ArborText, Inc.”>

<!ENTITY boilerplate SYSTEM “/standard/legalnotice.xml”>

<!ENTITY ATIlogo SYSTEM “/standard/logo.gif” NDATA GIF\*&A>

**Notation declarations**.

These identify specific types of external binary data. The information is passed on to the processing application, which can use it in whatever way it requires. For example:

<!NOTATION GIF87A SYSTEM “GIF”>

Optionality in a DTD

Optionality of an element "element" is indicated as:

Element occurrence of element is 1

Element? occurrence of element is 0 or 1

Element+ occurrence of element is 1 or more

Element\* occurrence of element is 0 or more

**Naming conventions**

Names consist of a maximum of 4 words joined to each other without spaces, the first word beginning with a low letter and the following words beginning with a capital letter. This is called "lowerCamelCase". Use “lowerCamelCase” for properties, references, etc.

For example: <dateDue>.

# APPENDIX H: DTD



<!ELEMENT installatioFirstName ( #PCDATA ) >

<!ELEMENT installatioSurname ( #PCDATA ) >

<!ELEMENT installationCompanyName ( #PCDATA ) >

<!ELEMENT installationdescription ( #PCDATA ) >

<!ELEMENT minspeed ( #PCDATA ) >

<!ELEMENT newActivationDate ( date ) >

<!ELEMENT newActivationTime ( time ) >

<!ELEMENT postCode ( #PCDATA ) >

<!ELEMENT serviceInstallationOwner ( (installatioSurname, installatioFirstName) | (installationCompanyName, icoNumber) ) >

<!ELEMENT speedAccept ( #PCDATA ) >

<!ELEMENT streetName ( #PCDATA ) >

<!ELEMENT time ( #PCDATA ) >

<!ELEMENT uircode ( #PCDATA ) >

<!ELEMENT billingaccount ( #PCDATA ) >

<!ELEMENT city ( #PCDATA ) >

<!ELEMENT citySection ( #PCDATA ) >

<!ELEMENT contactName ( #PCDATA ) >

<!ELEMENT contactPerson ( #PCDATA ) >

<!ELEMENT contactemail ( #PCDATA ) >

<!ELEMENT district ( #PCDATA ) >

<!ELEMENT speedRequired ( #PCDATA ) >

<!ELEMENT expectedActivationDate ( date ) >

<!ELEMENT expectedActivationTime ( time ) >

<!ELEMENT newPortTime (time)>

<!ELEMENT serviceContractOwner ((ownerSurname, ownerFirstName) | ownerCompanyName)>

<!ELEMENT serviceDisconnectionTime (time)>

<!ELEMENT installationAddress ( uircode?,((housenumberA, housenumberB) | housenumberA | housenumberB), streetName, city, citySection?, district, postCode, contactName?, contactPerson?, contactemail?, installationdescription?, speedRequired? )>

<!ELEMENT newPortDate (date)>

<!ELEMENT operatorID (#PCDATA)>

<!ELEMENT preActivated (cpsOperator, callingLineID)>

<!ATTLIST preActivated

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT header (toOperator, fromOperator, fileType, messageCount, timeSent, dateSent)>

<!ELEMENT cpsActivated (cpsOperator, callingLineID)>

<!ATTLIST cpsActivated

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT serviceDisconnectionDate (date)>

<!ELEMENT housenumberB (#PCDATA)>

<!ELEMENT housenumberA (#PCDATA)>

<!ELEMENT fileType (#PCDATA)>

<!ELEMENT cpsReject (cpsOperator, rejectionCode, rejectionDescription?, rejectionParameter\*)>

<!ATTLIST cpsReject

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

>

<!ELEMENT icoNumber (#PCDATA)>

<!ELEMENT ddiRange (ddiRangeStart, ddiRangeEnd)>

<!ELEMENT npAbort (recipientOperator, losingOperator)>

<!ATTLIST npAbort

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT timeSent (time)>

<!ELEMENT fileAccept (refTimeSent, refDateSent)>

<!ATTLIST fileAccept

refType (ORDR | RSLT | FLAC | VLDT | LLUORDR | LLURSLT | LLUFLAC | LLUVLDT) #REQUIRED

>

<!ELEMENT acknFile (header, (fileAccept | fileReject)\*)>

<!ELEMENT rejectionCode (#PCDATA)>

<!ELEMENT ownerSurname (#PCDATA)>

<!ELEMENT messageCount (#PCDATA)>

<!ELEMENT callingLineID (#PCDATA)>

<!ELEMENT resultFile (header, (npReject | cpsReject | cpsAccept | npAccept | npAbort | npIDcheckResp | cpsActivated)\*)>

<!ELEMENT npPortComplete (recipientOperator, losingOperator)>

<!ATTLIST npPortComplete

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT portActivationTime (time)>

<!ELEMENT refDateSent (date)>

<!ELEMENT portActivationDate (date)>

<!ELEMENT directoryNumber (#PCDATA)>

<!ELEMENT npChange (recipientOperator, losingOperator, newPortDate, newPortTime)>

<!ATTLIST npChange

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT ddiRangeStart (#PCDATA)>

<!ELEMENT ownerCompanyName (#PCDATA)>

<!ELEMENT cscCode (#PCDATA)>

<!ELEMENT losingOperator (operatorID)>

<!ELEMENT preAccept (cpsOperator, billingperiod)>

<!ATTLIST preAccept

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

>

<!ELEMENT ownerFirstName (#PCDATA)>

<!ELEMENT cpsProvide (cpsOperator, cscCode, serviceContractOwner, installationAddress, customerReferenceNumber, icoNumber?, callingLineID)>

<!ATTLIST cpsProvide

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

cpsServiceType (nat | intl | both) #REQUIRED

complexOrder (yes | no | Yes | No) #REQUIRED

pre (true | false) #IMPLIED

invoice (true | false) #IMPLIED

>

<!ELEMENT npAccept (recipientOperator, losingOperator)>

<!ATTLIST npAccept

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

>

<!ELEMENT npCancel (recipientOperator, losingOperator)>

<!ATTLIST npCancel

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT customerReferenceNumber (#PCDATA)>

<!ELEMENT dateSent (date)>

<!ELEMENT fileReject (refTimeSent, refDateSent, rejectionCode, rejectionDescription)>

<!ATTLIST fileReject

refType (ORDR | RSLT | FLAC | VLDT | LLUORDR | LLURSLT | LLUFLAC | LLUVLDT) #REQUIRED

>

<!ELEMENT fromOperator (operatorID)>

<!ELEMENT preTerminate (cpsOperator, cscCode, serviceContractOwner, installationAddress, customerReferenceNumber, icoNumber?, callingLineID)>

<!ATTLIST preTerminate

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

cpsServiceType (nat | intl | both) #REQUIRED

complexOrder (yes | no | Yes | No) #REQUIRED

>

<!ELEMENT npReturnNumber (recipientOperator, donorOperator, (directoryNumber | ddiRange), serviceDisconnectionDate, serviceDisconnectionTime)>

<!ATTLIST npReturnNumber

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT rejectionParameter (#PCDATA)>

<!ELEMENT orderFile (header, (cpsProvide | npProvide | npCancel | npChange | npPortComplete | npSubsequentPort | npReturnNumber | npIDcheck )\*)>

<!ELEMENT validateFile (header, (npReject | npIDcheckResp | cpsReject | preReject)\*)>

<!ELEMENT rejectionDescription (#PCDATA)>

<!ELEMENT toOperator (operatorID)>

<!ELEMENT cpsOperator (operatorID)>

<!ELEMENT npReject (recipientOperator, losingOperator, rejectionCode, rejectionDescription?, rejectionParameter\*)>

<!ATTLIST npReject

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

>

<!ELEMENT npProvide (recipientOperator, losingOperator, serviceContractOwner, installationAddress, customerReferenceNumber, icoNumber?, (directoryNumber+ | ddiRange+), npIDinfo?, npIDattachment?, portActivationDate, portActivationTime)>

<!ATTLIST npProvide

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

npServiceType (geog | non\_geog) #REQUIRED

complexOrder (yes | no | Yes | No) #REQUIRED

hasLLU (true | false) #IMPLIED

>

<!ELEMENT recipientOperator (operatorID)>

<!ELEMENT preReject (cpsOperator, rejectionCode, rejectionDescription?, rejectionParameter\*)>

<!ATTLIST preReject

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

>

<!ELEMENT npIDinfo (npID, spIDout?, spIDin? ) >

<!ELEMENT npID (#PCDATA) >

<!ELEMENT spIDout (#PCDATA) >

<!ELEMENT spIDin (#PCDATA) >

<!ELEMENT npIDattachment (attachment) >

<!ELEMENT attachment (#PCDATA)> <!-- base64Binary -->

<!ELEMENT npIDcheck (recipientOperator, losingOperator , npIDnumber, directoryNumber) >

<!ATTLIST npIDcheck orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT npIDcheckResp (recipientOperator, losingOperator , npIDnumber, directoryNumber, npIDcorrect) >

<!ATTLIST npIDcheckResp orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT npIDnumber (#PCDATA) >

<!ELEMENT npIDcorrect (#PCDATA) >

<!ATTLIST npIDcorrect resp (Y|N) #REQUIRED>

<!ELEMENT cpsAccept (cpsOperator)>

<!ATTLIST cpsAccept

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

>

<!ELEMENT npSubsequentPort (recipientOperator, donorOperator, losingOperator, (directoryNumber | ddiRange))>

<!ATTLIST npSubsequentPort

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT date (#PCDATA)>

<!ELEMENT donorOperator (operatorID)>

<!ELEMENT ddiRangeEnd (#PCDATA)>

<!ELEMENT refTimeSent (time)>

<!ELEMENT billingperiod (#PCDATA)>

<!ELEMENT preProvide (cpsOperator, cscCode, serviceContractOwner, installationAddress, customerReferenceNumber, icoNumber?, callingLineID)>

<!ATTLIST preProvide

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

cpsServiceType (nat | intl | both) #REQUIRED

complexOrder (yes | no | Yes | No) #REQUIRED

invoice (true | false) #IMPLIED

>

<!ELEMENT CeaseForReturnLine (lluOperator, loopID)>

<!ATTLIST CeaseForReturnLine

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT PSDMask (#PCDATA)>

<!ELEMENT lluLQM (lluOperator, callingLineID, refOrderNr, submitDate, submitTime)>

<!ATTLIST lluLQM

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

complexOrder (yes | no | Yes | No) #REQUIRED

lluCOS (A | B | C | D | E | F | G) #REQUIRED

loops CDATA #IMPLIED

>

<!ELEMENT LLUvalidateFile (header, (lluReject)\*)>

<!ELEMENT MDFID (#PCDATA)>

<!ELEMENT lluLQI (lluOperator, customerReferenceNumber, callingLineID, submitDate, submitTime)>

<!ATTLIST lluLQI

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

complexOrder (yes | no | Yes | No) #REQUIRED

lluCOS (A | B | C | D | E | F | G) #REQUIRED

loops CDATA #IMPLIED

>

<!ELEMENT lluChangeCOS (lluOperator, loopID)>

<!ATTLIST lluChangeCOS

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

currentCOS (A | B | C | D | E | F | G) #REQUIRED

newCOS (A | B | C | D | E | F | G) #REQUIRED

>

<!ELEMENT lluReturnLine (lluOperator, loopID)>

<!ATTLIST lluReturnLine

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT lluAbort (lluOperator)>

<!ATTLIST lluAbort

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

>

<!ELEMENT lluConvert (lluOperator, callingLineID, loopID, convertDate, convertTime)>

<!ATTLIST lluConvert

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

lluServiceType (PPV | PPVNP) #REQUIRED

lluCOS (A | B | C | D | E | F | G) #REQUIRED

>

<!ELEMENT lluChange (lluOperator, orderActivationDate, orderActivationTime)>

<!ATTLIST lluChange

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT lluSlaChange (lluOperator, SLALevel, loopID)>

<!ATTLIST lluSlaChange

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT correlation (callingLineID, MDFID?, rejectionCode?, rejectionDescription?)>

<!ELEMENT lluComplete (lluOperator, loopID, loopID2?, loopID3?)>

<!ATTLIST lluComplete

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT correlationsList (correlation+)>

<!ELEMENT lluCancel (lluOperator)>

<!ATTLIST lluCancel

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT lluAccept (lluOperator, PSDMask?, MDFID?)>

<!ATTLIST lluAccept

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

>

<!ELEMENT loops (#PCDATA)>

<!ELEMENT convertTime (time)>

<!ELEMENT loopID (#PCDATA)>

<!ELEMENT loopID2 (#PCDATA)>

<!ELEMENT loopID3 (#PCDATA)>

<!ELEMENT LLUacknFile (header, (fileAccept | fileReject)\*)>

<!ELEMENT lluMDFQuery (lluOperator, CLIList)>

<!ATTLIST lluMDFQuery

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT lluOrderFile (header, (lluProvide | lluProvideB | lluconfirm | lluTransfer | lluConvert | lluChangeCOS | lluReturnLine | lluLQI | lluLQM | lluMDFQuery | ceaseForTransfer | CeaseForReturnLine |

lluSlaChange | lluChange | lluCancel | lluComplete)\*)>

<!ELEMENT lluTransfer (recipientOperator, customerReferenceNumber, callingLineID, loopID, HDFPairNumber, MDFID, orderActivationDate, orderActivationTime)>

<!ATTLIST lluTransfer

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

lluServiceType (PPV | PPVM | SPV | SPVM | PPVNP | PPVNPM) #REQUIRED

lluCOS (A | B | C | D | E | F | G) #REQUIRED

complexOrder (yes | no | Yes | No) #REQUIRED

loops CDATA #IMPLIED

>

<!ELEMENT convertDate (date)>

<!ELEMENT lluInstallationComplete (lluOperator, loopID, loopID2?, loopID3?, HDFPairNumber, HDFPairNumber2?, HDFPairNumber3?, MDFID, orderActivationDate, orderActivationTime)>

<!ATTLIST lluInstallationComplete

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

RefSequenceNr CDATA #REQUIRED

>

<!ELEMENT LLUresultFile (header, (lluAccept | lluReject | lluAbort | lluMDFResponse | lluInstallationComplete | lluinformation | lludatetime | lluSlaActivated)\*)>

<!ELEMENT HDFPairNumber (#PCDATA)>

<!ELEMENT HDFPairNumber2 (#PCDATA)>

<!ELEMENT HDFPairNumber3 (#PCDATA)>

<!ELEMENT CLI (callingLineID, customerReferenceNumber)>

<!ELEMENT CLIList (CLI+)>

<!ELEMENT lluMDFResponse (lluOperator, correlationsList)>

<!ATTLIST lluMDFResponse

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

>

<!ELEMENT lluReject (lluOperator, MDFID, rejectionCode, rejectionDescription, rejectionParameter)>

<!ATTLIST lluReject

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

refSequenceNr CDATA #REQUIRED

>

<!ELEMENT orderActivationTime (time)>

<!ELEMENT lluOperator (operatorID)>

<!ELEMENT submitTime (time)>

<!ELEMENT lluProvide (lluOperator, customerReferenceNumber, callingLineID+, HDFPairNumber, HDFPairNumber2?, HDFPairNumber3?, MDFID, SLALevel?, orderActivationDate, orderActivationTime)>

<!ATTLIST lluProvide

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

complexOrder (yes | no | Yes | No) #REQUIRED

lluServiceType (PPV | PPVM | SPV | PPVNPM | PPVNP) #REQUIRED

lluCOS (A | B | C | D | E | F | G) #REQUIRED

lluLQMOrderNr CDATA #IMPLIED

loops CDATA #IMPLIED

>

<!ELEMENT lluProvideB ( lluOperator, billingaccount?, minspeed?, SLALevel?, serviceInstallationOwner, installationAddress,

expectedActivationDate, expectedActivationTime ) >

<!ATTLIST lluProvideB lluCOS (A | B | C | D | E | F | G) #REQUIRED >

<!ATTLIST lluProvideB lluServiceType (PPV | PPVM | SPV | SPVM | PPVNP | PPVNPM | PPVB) #REQUIRED >

<!ATTLIST lluProvideB orderNr CDATA #REQUIRED >

<!ATTLIST lluProvideB sequenceNr CDATA #REQUIRED >

<!ELEMENT lluconfirm ( lluOperator, HDFPairNumber, MDFID ) >

<!ATTLIST lluconfirm orderNr NMTOKEN #REQUIRED >

<!ATTLIST lluconfirm refSequenceNr NMTOKEN #REQUIRED >

<!ATTLIST lluconfirm sequenceNr NMTOKEN #REQUIRED >

<!ELEMENT lludatetime ( lluOperator, newActivationDate, newActivationTime ) >

<!ATTLIST lludatetime orderNr CDATA #REQUIRED >

<!ATTLIST lludatetime refSequenceNr CDATA #REQUIRED >

<!ATTLIST lludatetime sequenceNr CDATA #REQUIRED >

<!ELEMENT lluinformation ( lluOperator, MDFID, speedAccept ) >

<!ATTLIST lluinformation orderNr CDATA #REQUIRED >

<!ATTLIST lluinformation refSequenceNr CDATA #REQUIRED >

<!ATTLIST lluinformation sequenceNr CDATA #REQUIRED >

<!ELEMENT lluSlaActivated ( lluOperator, SLALevel, loopID) >

<!ATTLIST lluSlaActivated orderNr CDATA #REQUIRED >

<!ATTLIST lluSlaActivated refSequenceNr CDATA #REQUIRED >

<!ATTLIST lluSlaActivated sequenceNr CDATA #REQUIRED >

<!ELEMENT SLALevel (#PCDATA)>

<!ELEMENT orderActivationDate (date)>

<!ELEMENT submitDate (date)>

<!ELEMENT ceaseForTransfer (losingOperator, loopID, MDFID, PSDMask)>

<!ATTLIST ceaseForTransfer

orderNr CDATA #REQUIRED

sequenceNr CDATA #REQUIRED

>

<!ELEMENT refOrderNr (#PCDATA)>

# APPENDIX I: XML FILE EXAMPLES

## ORDR

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE orderFile PUBLIC "" "np\_cps.dtd">

<orderFile>

<header>

<toOperator>

<operatorID>202</operatorID>

</toOperator>

<fromOperator>

<operatorID>212</operatorID>

</fromOperator>

<fileType>ORDR</fileType>

<messageCount>0006</messageCount>

<timeSent>

<time>16:30:01</time>

</timeSent>

<dateSent>

<date>2006-09-01</date>

</dateSent>

</header>

<npChange orderNr="O00000003" sequenceNr="2">

<recipientOperator>

<operatorID>212</operatorID>

</recipientOperator>

<losingOperator>

<operatorID>202</operatorID>

</losingOperator>

<newPortDate>

<date>2003-05-09</date>

</newPortDate>

<newPortTime>

<time>11:30:00</time>

</newPortTime>

</npChange>

<cpsProvide orderNr="O00000012" sequenceNr="1" cpsServiceType="both" complexOrder="no" pre="true">

<cpsOperator>

<operatorID>212</operatorID>

</cpsOperator>

<cscCode>1012</cscCode>

<serviceContractOwner>

<ownerSurname>Peter</ownerSurname>

<ownerFirstName>Bondra</ownerFirstName>

</serviceContractOwner>

<installationAddress>

<housenumberA>12</housenumberA>

<housenumberB/>

<streetName>Keplerova</streetName>

<city>Praha</city>

<citySection/>

<district>Praha 6</district>

<postCode>118 00</postCode>

</installationAddress>

<customerReferenceNumber>0123456789</customerReferenceNumber>

<callingLineID>123123123</callingLineID>

</cpsProvide>

<npProvide orderNr="O00000013" sequenceNr="1" npServiceType="geog" complexOrder="yes" hasLLU="true">

<recipientOperator>

<operatorID>212</operatorID>

</recipientOperator>

<losingOperator>

<operatorID>202</operatorID>

</losingOperator>

<serviceContractOwner>

<ownerCompanyName>ABC Consult</ownerCompanyName>

</serviceContractOwner>

<installationAddress>

<housenumberA/>

<housenumberB>3</housenumberB>

<streetName>Sokolovská</streetName>

<city>Praha</city>

<district>Praha 2</district>

</installationAddress>

<customerReferenceNumber>0121212122</customerReferenceNumber>

<directoryNumber>123123124</directoryNumber>

<directoryNumber>123123125</directoryNumber>

<directoryNumber>123123126</directoryNumber>

<portActivationDate>

<date>2003-05-09</date>

</portActivationDate>

<portActivationTime>

<time>10:00:00</time>

</portActivationTime>

</npProvide>

<cpsProvide orderNr="O00000014" sequenceNr="1" cpsServiceType="both" complexOrder="yes">

<cpsOperator>

<operatorID>212</operatorID>

</cpsOperator>

<cscCode>1012</cscCode>

<serviceContractOwner>

<ownerSurname>Peter</ownerSurname>

<ownerFirstName>Bondra</ownerFirstName>

</serviceContractOwner>

<installationAddress>

<housenumberA>12</housenumberA>

<housenumberB>100</housenumberB>

<streetName>Keplerova</streetName>

<city>Praha</city>

<citySection>Pohořelec</citySection>

<district>Praha 6</district>

<postCode>118 00</postCode>

</installationAddress>

<customerReferenceNumber>0123456789</customerReferenceNumber>

<callingLineID>1231233</callingLineID>

</cpsProvide>

<npReturnNumber orderNr="O00000019" sequenceNr="1">

<recipientOperator>

<operatorID>212</operatorID>

</recipientOperator>

<donorOperator>

<operatorID>202</operatorID>

</donorOperator>

<directoryNumber>222111222</directoryNumber>

<serviceDisconnectionDate>

<date>2003-04-17</date>

</serviceDisconnectionDate>

<serviceDisconnectionTime>

<time>10:00:00</time>

</serviceDisconnectionTime>

</npReturnNumber>

<preProvide orderNr="O00000015" sequenceNr="1" cpsServiceType="both" complexOrder="yes">

<cpsOperator>

<operatorID>212</operatorID>

</cpsOperator>

<cscCode>1012</cscCode>

<serviceContractOwner>

<ownerSurname>Peter</ownerSurname>

<ownerFirstName>Bondra</ownerFirstName>

</serviceContractOwner>

<installationAddress>

<housenumberA>12</housenumberA>

<housenumberB>100</housenumberB>

<streetName>Keplerova</streetName>

<city>Praha</city>

<citySection>Pohořelec</citySection>

<district>Praha 6</district>

<postCode>118 00</postCode>

</installationAddress>

<customerReferenceNumber>0123456789</customerReferenceNumber>

<callingLineID>1231233</callingLineID>

</preProvide>

</orderFile>

## RSLT

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE resultFile PUBLIC "" "np\_cps.dtd">

<resultFile>

<header>

<toOperator>

<operatorID>212</operatorID>

</toOperator>

<fromOperator>

<operatorID>202</operatorID>

</fromOperator>

<fileType>RSLT</fileType>

<messageCount>0007</messageCount>

<timeSent>

<time>16:45:01</time>

</timeSent>

<dateSent>

<date>2006-09-05</date>

</dateSent>

</header>

<npReject orderNr="O00000003" sequenceNr="2" refSequenceNr="2">

<recipientOperator>

<operatorID>212</operatorID>

</recipientOperator>

<losingOperator>

<operatorID>202</operatorID>

</losingOperator>

<rejectionCode>L007</rejectionCode>

<rejectionDescription>npChange received too late</rejectionDescription>

</npReject>

<npAbort orderNr="O00000003" sequenceNr="3">

<recipientOperator>

<operatorID>212</operatorID>

</recipientOperator>

<losingOperator>

<operatorID>202</operatorID>

</losingOperator>

</npAbort>

<cpsAccept orderNr="O00000012" sequenceNr="1" refSequenceNr="1">

<cpsOperator>

<operatorID>212</operatorID>

</cpsOperator>

</cpsAccept>

<cpsReject orderNr="O00000013" sequenceNr="1" refSequenceNr="1">

<cpsOperator>

<operatorID>212</operatorID>

</cpsOperator>

<rejectionCode>C003</rejectionCode>

<rejectionDescription>Number not subjected to CPS</rejectionDescription>

</cpsReject>

<preReject orderNr="O00000015" sequenceNr="1" refSequenceNr="1">

<cpsOperator>

<operatorID>212</operatorID>

</cpsOperator>

<rejectionCode>C003</rejectionCode>

<rejectionDescription>Number not subjected to CPS</rejectionDescription>

</preReject>

<cpsActivated orderNr="O00000016" sequenceNr="2">

<cpsOperator>

<operatorID>212</operatorID>

</cpsOperator>

<callingLineID>582396118</callingLineID>

</cpsActivated>

<preActivated orderNr="O00000017" sequenceNr="1">

<cpsOperator>

<operatorID>212</operatorID>

</cpsOperator>

<callingLineID>582396118</callingLineID>

</preActivated>

</resultFile>

## VLDT

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE validateFile SYSTEM "np\_cps.dtd">

<validateFile>

<header>

<toOperator>

<operatorID>202</operatorID>

</toOperator>

<fromOperator>

<operatorID>255</operatorID>

</fromOperator>

<fileType>VLDT</fileType>

<messageCount>0002</messageCount>

<timeSent>

<time>08:47:11</time>

</timeSent>

<dateSent>

<date>2003-04-18</date>

</dateSent>

</header>

<npReject orderNr="O00000007" sequenceNr="1" refSequenceNr="1">

<recipientOperator>

<operatorID>202</operatorID>

</recipientOperator>

<losingOperator>

<operatorID>255</operatorID>

</losingOperator>

<rejectionCode>F001</rejectionCode>

<rejectionDescription>Mandatory field missing</rejectionDescription>

<rejectionParameter>npProvide/directoryNumber</rejectionParameter>

</npReject>

<preReject orderNr="O00000015" sequenceNr="1" refSequenceNr="1">

<cpsOperator>

<operatorID>212</operatorID>

</cpsOperator>

<rejectionCode>C003</rejectionCode>

<rejectionDescription>Number not subjected to CPS</rejectionDescription>

</preReject>

</validateFile>

## FLAC

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE acknFile SYSTEM "np\_cps.dtd">

<acknFile>

<header>

<toOperator>

<operatorID>255</operatorID>

</toOperator>

<fromOperator>

<operatorID>202</operatorID>

</fromOperator>

<fileType>FLAC</fileType>

<messageCount>0002</messageCount>

<timeSent>

<time>08:45:00</time>

</timeSent>

<dateSent>

<date>2003-04-18</date>

</dateSent>

</header>

<fileAccept refType="ORDR">

<refTimeSent>

<time>16:31:34</time>

</refTimeSent>

<refDateSent>

<date>2003-04-17</date>

</refDateSent>

</fileAccept>

<fileReject refType="RSLT">

<refTimeSent>

<time>16:31:39</time>

</refTimeSent>

<refDateSent>

<date>2003-04-17</date>

</refDateSent>

<rejectionCode>X001</rejectionCode>

<rejectionDescription>XML not well formed or DTD error</rejectionDescription>

</fileReject>

</acknFile>

## FILE SAMPLES



# APPENDIX J: SOAP Message

## Request

<?xml version='1.0' encoding='UTF-8'?>

<SOAP-ENV:Envelope

xmlns:SOAP-ENV='http://schemas.xmlsoap.org/soap/envelope/'

xmlns:SOAP-ENC='http://schemas.xmlsoap.org/soap/encoding/' xmlns:xsi='http://www.w3.org/1999/XMLSchema-instance' xmlns:xsd='http://www.w3.org/1999/XMLSchema'>

<SOAP-ENV:Body>

<ns0:send

xmlns:ns0='urn:ReceiveMessaging'

SOAP-ENV:encodingStyle='http://schemas.xmlsoap.org/soap/encoding/'>

<ns0:arg0 xsi:type='xsd:string'>

[XML file content]

</ns0:arg0>

<ns0:arg1 xsi:type='xsd:string'>ORDR</ns0:arg1>

</ns0:send>

</SOAP-ENV:Body>

</SOAP-ENV:Envelope>

## Response

<?xml version='1.0' encoding='UTF-8'?>

<SOAP-ENV:Envelope

xmlns:SOAP-ENV='http://schemas.xmlsoap.org/soap/envelope/'

xmlns:SOAP-ENC='http://schemas.xmlsoap.org/soap/encoding/'

xmlns:xsi='http://www.w3.org/1999/XMLSchema-instance' xmlns:xsd='http://www.w3.org/1999/XMLSchema'>

<SOAP-ENV:Body>

<ns0:sendResponse

xmlns:ns0='urn:local'

SOAP-ENV:encodingStyle='http://schemas.xmlsoap.org/soap/encoding/'>

<ns0:return xsi:type='xsd:string'>0</ns0:return>

</ns0:sendResponse>

</SOAP-ENV:Body>

</SOAP-ENV:Envelope>

# Appendix K: WSDL

<definitions

targetNamespace="[target namespace URI]"

xmlns="http://schemas.xmlsoap.org/wsdl/"

xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"

xmlns:tns="[tns namespace URI]"

xmlns:xsd="http://www.w3.org/1999/XMLSchema"

xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">

<types>

<schema

targetNamespace='[target namespace URI]'

xmlns='http://www.w3.org/1999/XMLSchema'>

</schema>

</types>

<message name="sendRequest">

<part name="arg0" type="xsd:string" />

<part name="arg1" type="xsd:string" />

</message>

<message name="sendResponse">

<part name="return" type="xsd:string" />

</message>

<portType name="ReceiveMessagingPortType">

<operation name="send">

<input message="tns:sendRequest"/>

<output message="tns:sendResponse"/>

</operation>

</portType>

<binding name="ReceiveMessagingBinding" type="tns:ReceiveMessagingPortType">

<soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>

<operation name="send">

<soap:operation soapAction="urn:send"/>

<input>

<soap:body

use="encoded"

namespace='urn:ReceiveMessaging'

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"/>

</input>

<output>

<soap:body

use="encoded"

namespace='urn:ReceiveMessaging'

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"/>

</output>

</operation>

</binding>

<service name="ReceiveMessaging">

<documentation>todo</documentation>

<port name="ReceiveMessagingPort" binding="tns:ReceiveMessagingBinding">

<soap:address location="[Web service URL]"/>

</port>

</service>

</definitions>

# APPENDIX L: General description of communication between OLOs

**Terms and definitions**

*Web Service* is a web application page where client POSTs the request.

Client connects to server using *http over SSL* - all data are encrypted during transfer.

General description of communication

Client sends to the server username, password in http header. Authentication is performed on http level and authentication type is basic. These authentication data are sent together with SOAP request in one POST request. Because of security reasons the client must know authentication realm name on server side.

If authentication is successful, the SOAP Request is handed over and processed. SOAP Response is created by server and returned to the client. The response contains Response code (string). The values of this response are as follows:

0 - OK

1 - Out of business hours (see deadline rules)

2 - Internal Server Error

Deadline rules:

* files that transfer will be **started** till defined deadline (17.00) will be processed in business day of receiving file. „Master“ time is time on the server side not on the client.
* files that transfer will be **started** after defined deadline (17.00) will be rejected