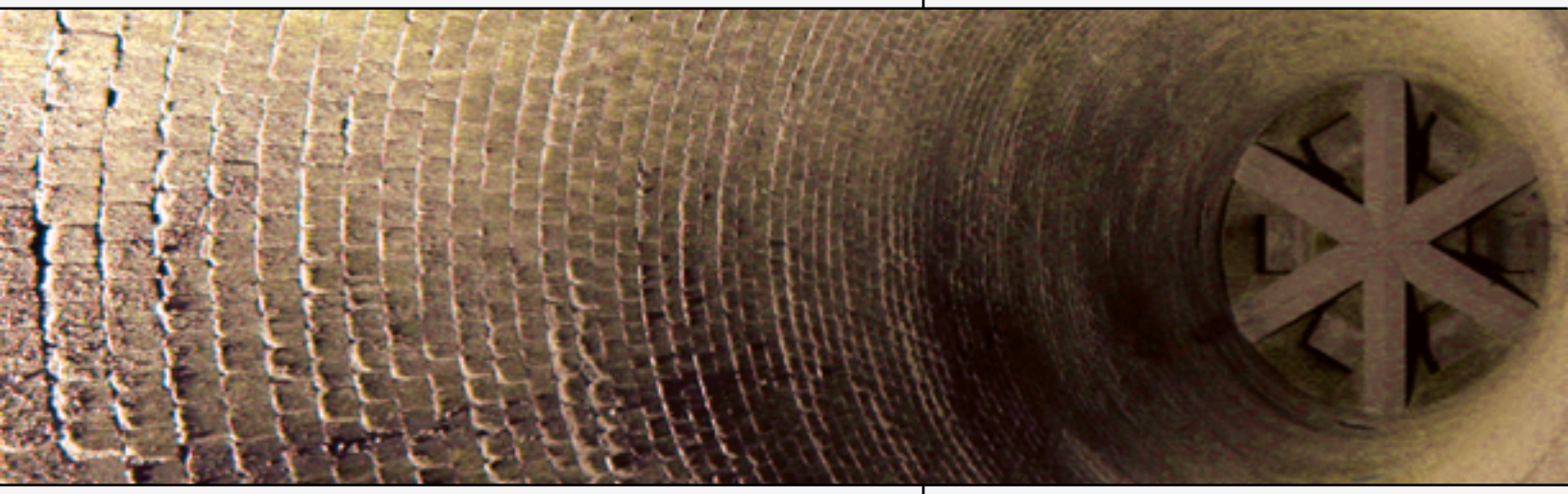
PORTA ONE®

PORTA Billing 100®

Maintenance Release 9



Templates Guide

www.portaone.com

Copyright notice & disclaimers

PortaBilling100 Templates Guide
V.1.9.1, August 2004

Please address your comments and suggestions to Sales Department, PortaOne, Inc., Suite 400, 2963 Glen Drive, Coquitlam, BC, V3B 2P7, Canada

Table of contents

Preface

The PortaBilling100 User Guide gives a brief explanation of templates. This manual describes in detail how to optimally import and export data from PortaBilling.

Where to get the latest version of this guide

The hard copy of this guide is updated at major releases only and does not always contain the latest material for enhancements occurring between minor releases. The online copy of this guide is always up-to-date and integrates the latest changes to the product. You can access the latest copy of this guide at **www.portaone.com / solutions / billing / docs**.

Conventions

This publication uses the following conventions:

- Commands and keywords are in **boldface**
- Terminal sessions, console screens, system file names are displayed in `fixed width font`

Caution means 'reader be careful'. You are capable of doing something that might result in program malfunction or loss of data.

NOTE: Means *reader take note*. Notes contain helpful suggestions or references to materials not contained in this manual

Timesaver means the described action saves time. You can save time by performing the action described in the paragraph.

Tips Means the following information might help you solve a problem

1. Templates Concepts

Real-life situation

Before we discuss technical details about the PortaBilling and templates, let's take a look at a "real life" example which will help us in our study.

Situation

You are the owner of a VoIP billing system. You receive a list of rates from your vendors and need to import them into the system.

Solution 1: Manual entry

In order to do this, you decide to hire a temporary worker who will type in all the data. That person does not know much about your business, so you will have to give her detailed instructions how to proceed. So your instructions might sound similar to following:

1. Here is an Excel file from our vendor
2. After you open it, on a second worksheet called "Rates" on the fist row you will see a title and list of rates under it.
3. The value in column "C" (titled "Prefix") goes into the field "Destination" on the web interface, but do not enter **011** on the beginning of the number. The value in column "F" (titled "Price per minute") goes into the fields "Price 1" and "Price N" on the web. There is no "Effective From" data in the file, so just put current date into that field on the web.

Does not sound too complicated? Now let's take a look on what you have just done:

- You have specified in which file format data are ("**an Excel file**")
- You have specified where within that file are the requied data ("**second worksheet called 'Rates' on the fist row**")
- And finally for the each data element which needs to be imported into the system (Destination, Price) you have specified:
    - In which column data are ("**column 'C' - titled 'Prefix'**")
    - Format of the data (if they need to be transformed somehow prior to be entered into the system) ("**do not enter 011 on the beginning**")

Now let's take a look on the alternative to this, which will let you do the same but more easily, faster and with fewer errors.

Solution 2: Template engine

You can use the PortaBilling template engine to handle import or export of the data from PortaBilling. When using templates in PortaBilling, you are in a situation similar to the one described above. Since the template processor knows nothing about the format of your data, you will have to provide the template engine with detailed instructions similar to the directions you gave to the human person in our example. You will have to tell template processor about file format of the input file, where the useful

data is inside of that file and formatting information of the data. Of course, you will enter your instructions as a template definition, so:

- They stay stored in the system and you can always reuse them later.
- Everything is specified explicitly, no communication errors.
- The template engine processing is fast and automatic, so you do not need to wait for the human data-entry (which might take days.)

Template engine

The template engine allows you to define data conversion rules (templates) and execute them. There are two ways of template-based data processing, and two types of templates respectively.

Import data into PortaBilling (upload)

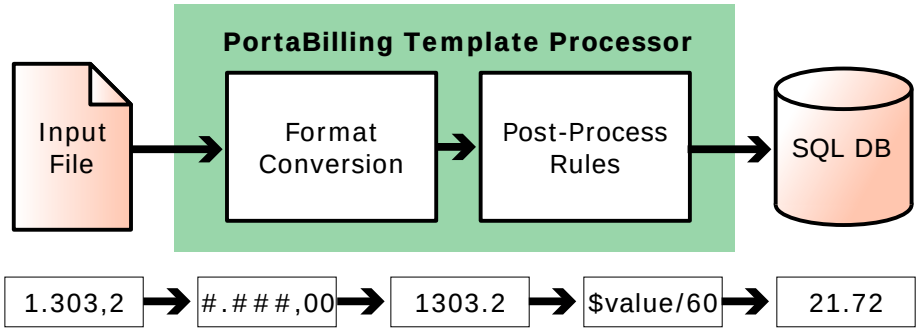


Figure 1-1

Such a template (called an upload template) defines:

- The input file type (for example, "CSV" or "Excel")
- The location of the input file
- The format of the file
- Transformations to apply to the data before inserting into the database

When data processing occurs, the template engine parses the input file according to it's type, locates the data inside of the file, converts the data into an internal representation according to the format specification, and applies post processing rules and writes data to the database.

Export data from PortaBilling (download)

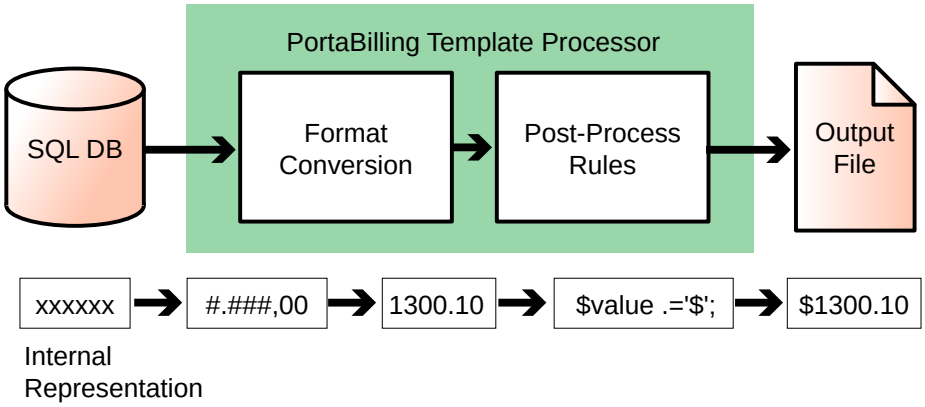


Figure 1-2

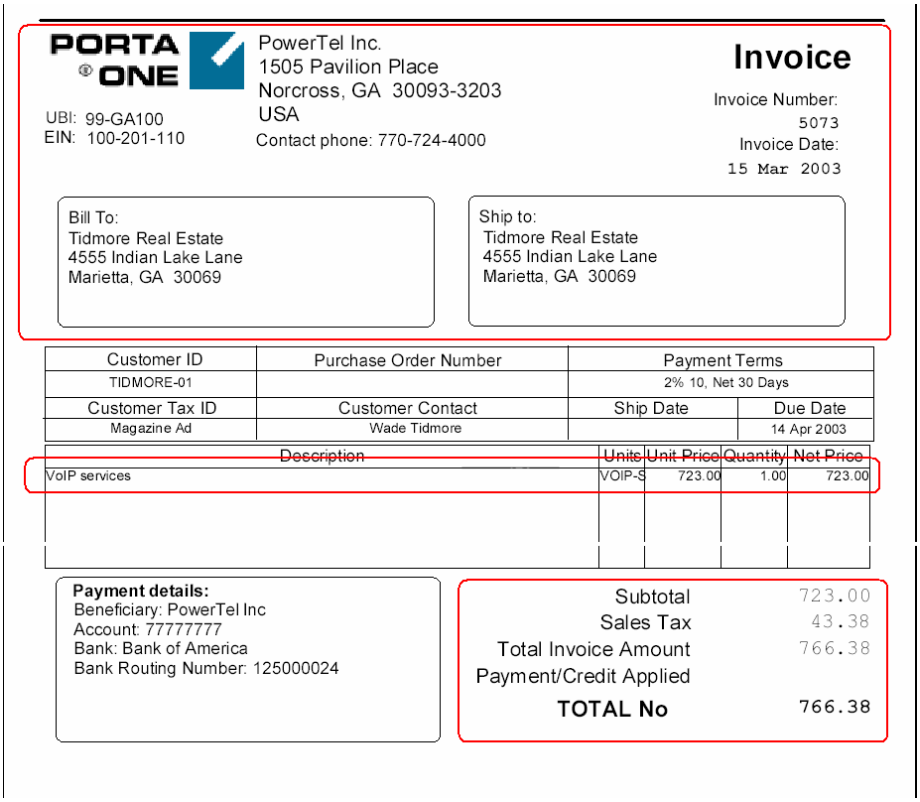
Such a template (called a download template) defines:

- In which format the data should be represented
- How it should be processed before being inserted into the database
- How the data should be arranged in the output file
- The output data forma

When data processing occurs, the template engine extracts data from the database, converts it into the specified format, applies post processing rules and then creates a file of the specified type, placing the data inside the file where it should be.

Template components

Data for an invoice or tariff may have a very complex structure. To deal with these cases more easily, data within the template is split into different groups called "template components", so they can be managed separately. See an example invoice on the picture below.



PORTA ONE

PowerTel Inc.
1505 Pavilion Place
Norcross, GA 30093-3203
USA
Contact phone: 770-724-4000

UBI: 99-GA100
EIN: 100-201-110

Invoice

Invoice Number: 5073
Invoice Date: 15 Mar 2003

Bill To:
Tidmore Real Estate
4555 Indian Lake Lane
Marietta, GA 30069

Ship to:
Tidmore Real Estate
4555 Indian Lake Lane
Marietta, GA 30069

Customer ID	Purchase Order Number	Payment Terms	
TIDMORE-01		2% 10, Net 30 Days	
Customer Tax ID	Customer Contact	Ship Date	Due Date
Magazine Ad	Wade Tidmore		14 Apr 2003

Description	Units	Unit Price	Quantity	Net Price
VoIP services	VOIP-S	723.00	1.00	723.00

Payment details:
Beneficiary: PowerTel Inc
Account: 77777777
Bank: Bank of America
Bank Routing Number: 125000024

Subtotal	723.00
Sales Tax	43.38
Total Invoice Amount	766.38
Payment/Credit Applied	
TOTAL No	766.38

Figure 1-3

We can clearly identify a few independent groups of data here:

1. **Invoice header** - is present only once in the file and contains global information
2. **Invoice line(s)** – groups of data which repeat multiple times in a file, each time with different values
3. **Invoice footer** – is present only once in the file and contains global information

Here is another example, this time for the tariff.

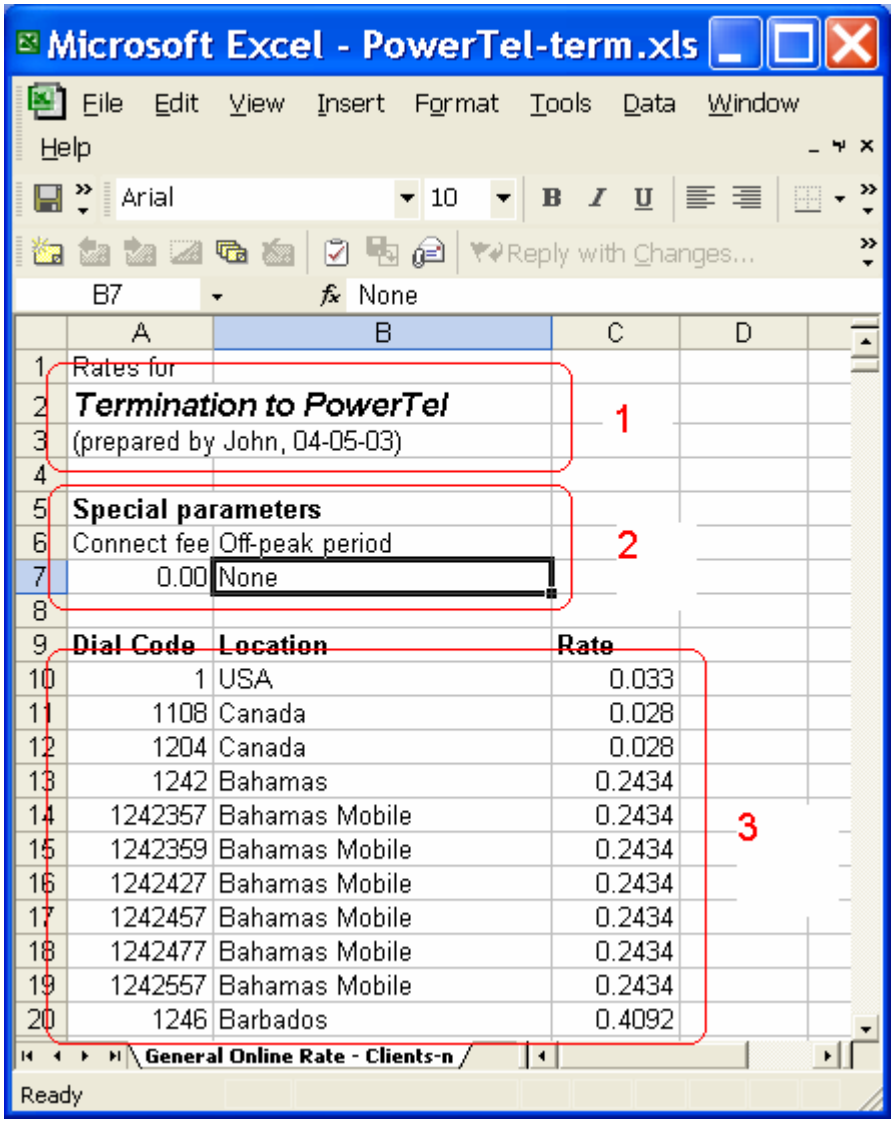


Figure 1-4

Again, we have different groups here:

1. **Header** - is present only once in the file and contains global information
2. **Tariff parameters** – is present only once in the file and also contains global information
3. **Rate data** – a grouping of data, which repeats multiple times in a file with different values

So there are 6 possible template components:

Header

Header contains data fields with the global data for the object. For example, on the invoice this would be an invoice number or the name of the company issuing the invoice. The tariff header contains tariff name, currency and description.

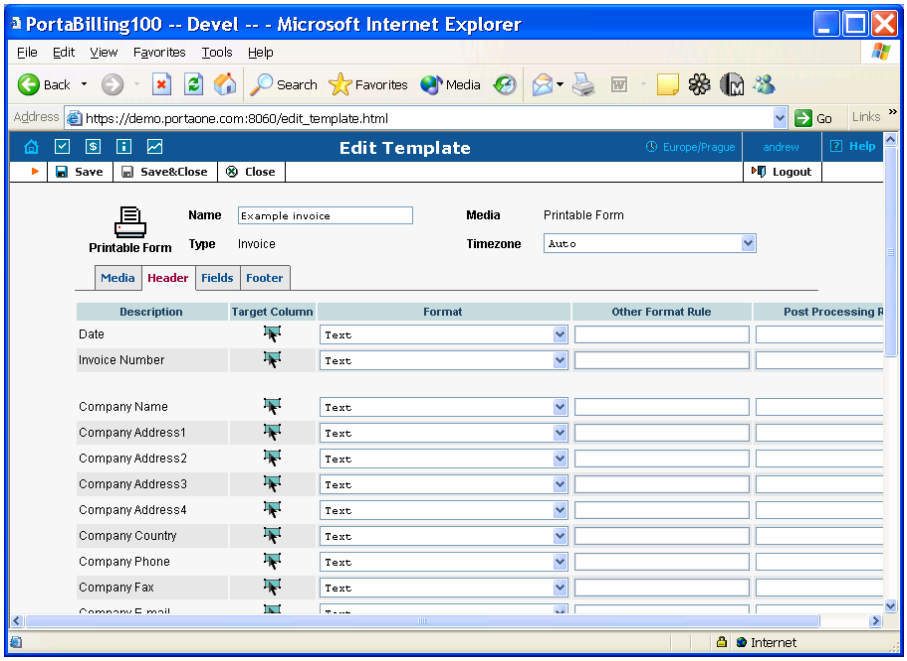


Figure 1-5

Page Header

This template component is only available only for printable forms in which the data is split across pages. (For CVS or Excel files the data is placed as one single set). Data fields, defined in this component will be placed at the beginning of each printed page.

NOTE: This component might not be available in your version of PortaBilling if it does not include multi-page invoice support.

Fields

Similar to the **Header**, this component contains global data for the object to be processed (tariff, invoice, etc.). Data fields are split between **Header** and **Fields** just for convenience, so the **Header** contains the main parameters and **auxiliary** parameters are **Fields**. For instance, templates for the tariff upload/download contain specialized **Fields:** off-peak period definition, free seconds, connection fee, etc.

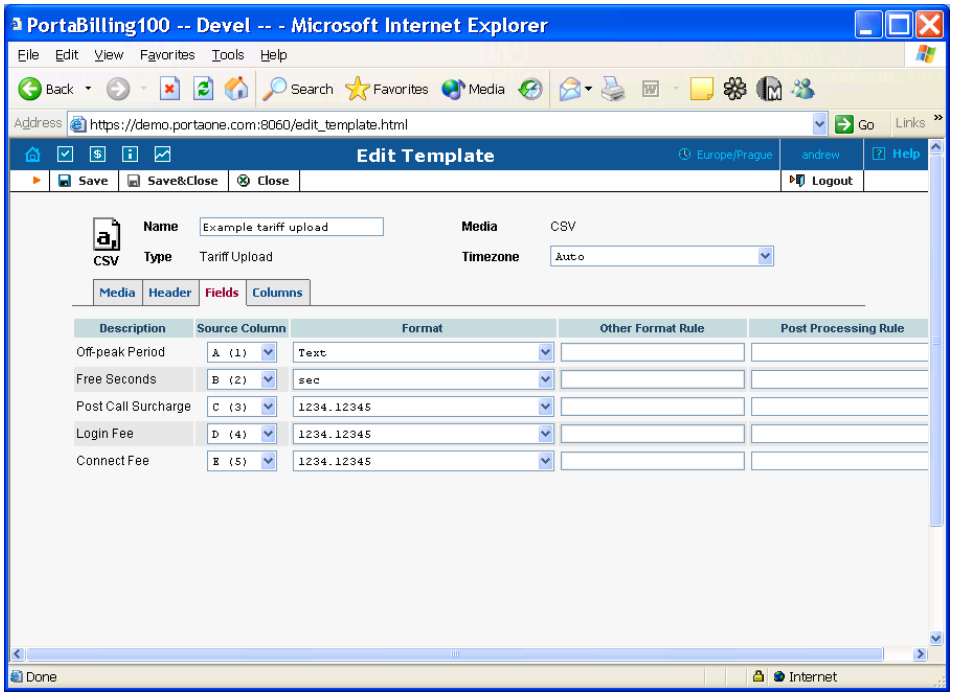


Figure 1-6

Columns (repeating groups)

This is the most important component, since it describes the variable portion of data. For example rate entries all have the same structure and format (phone prefix in the first column, country name in the second, price in the third, etc.) but there are many such rows in the file, and each row is an independent object. So you define how one single row of data is formatted, and then it will be applied to all rows in the file.

Footer

Contains additional data about the object, for example invoice totals.

Page Footer

Available only for printable forms – data fields to be placed at the end of each printed page.

Groups

See *Template groups* section.

Template parameters

Template type

The template type defines the intended usage of the template:

- If it will be for data import (***upload***) or for export of data from PortaBilling (***download***)

- What kind of data will be involved (invoice data, tariff data, destinations, etc.)

Examples of template types are "Tariff Download" and "Destination Upload"

Template media

Defines the file format to be processed, for example .CSV (Comma-Separated Values), .XLS (Microsoft Excel) or Printable form (specific HTML, which could be converted to PDF).

Template time zone

Normally, when people specify a point in time, they declare the day of the month, month, year, hour and minute, and optionally seconds. However to be absolutely precise, this definition should also include the time zone. Let's assume that your vendor sends you a data file, which contains his new rates and for each rate, the date when it becomes effective. This is good, but is **01-May-2003 00:00** in the Pacific Standard time zone or Central European Time? Obviously time zone misinterpretation can make the rest of the data incorrect. In most cases specification of the time zone is not included in the date itself, but specified externally. For example, your vendor might tell you: "All of the given times are in Eastern Standard Time". Or there might be a note in the file saying this. In any case, template engine needs to know. This is specified as the **Template time zone** parameter. If this parameter is set to "Auto" it means, that the time zone of the user, performing the upload is used. This is convenient when working with your own tariffs, but usually is not acceptable when uploading tariffs from vendors.

NOTE: If the date includes time zone (for example **2003-04-10 12:00:00 Australia/Sydney**) this time zone will have precedence over the template's time zone. However, in this case make sure that time zone specification is supported and unambiguous. For example **EST** could mean both Eastern Time in the US (GMT -5) and Eastern Time in Australia (GMT +10) – quite a big difference!

Media parameters

Media parameters define the data layout inside of the file, where the data for the individual components are located in the file, and if they are present at all.

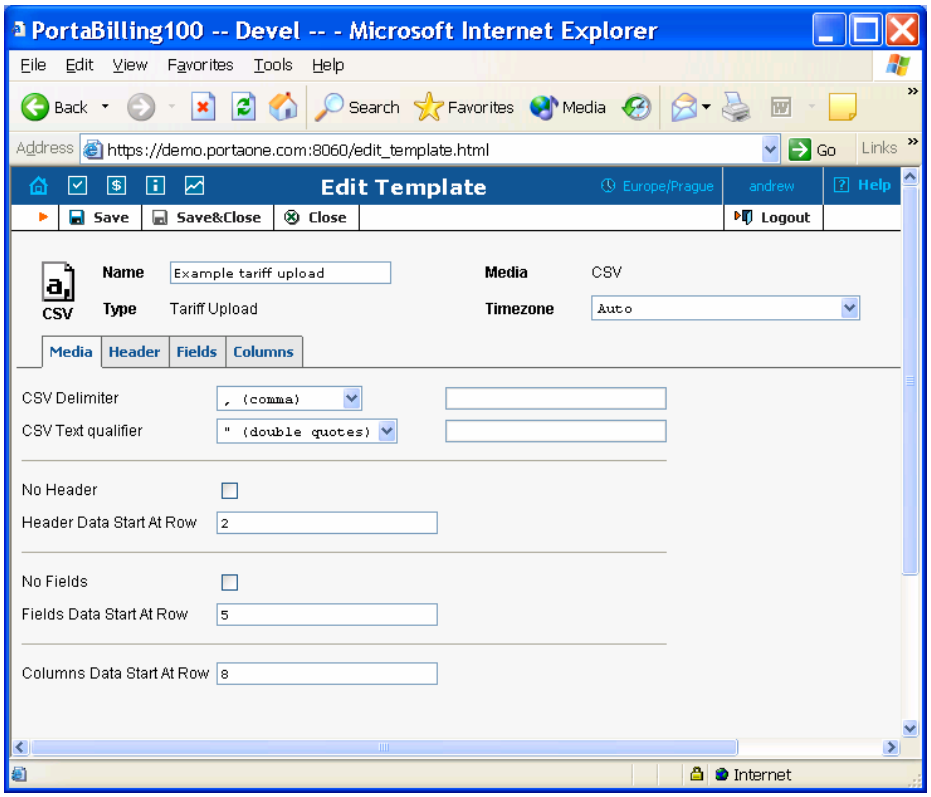
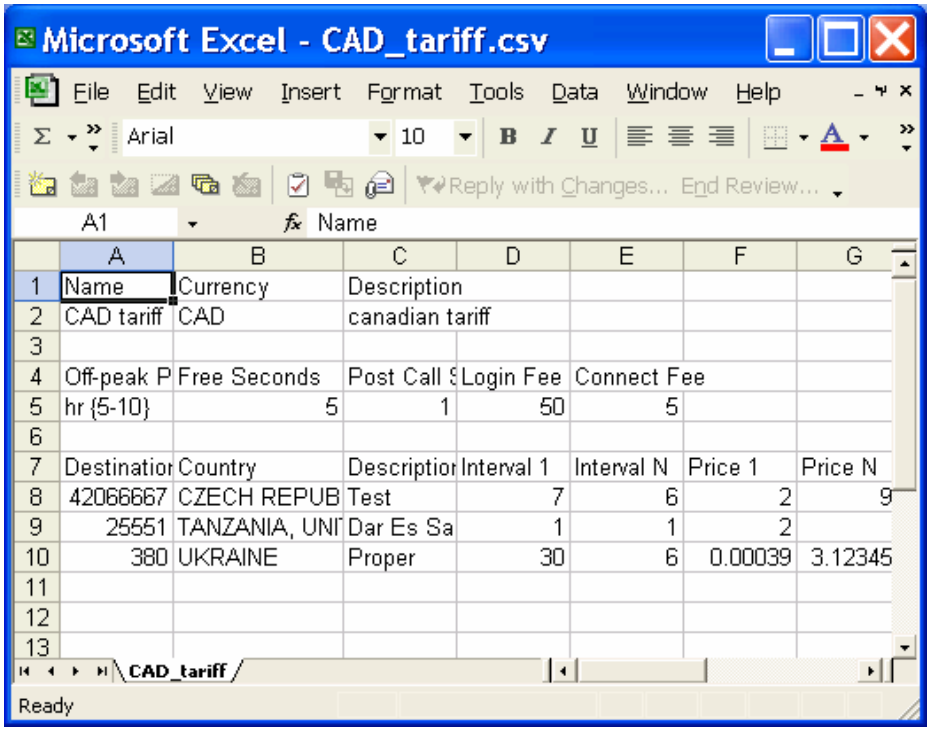


Figure 1-7

Here is an example data file that would match this template. Note, that "…. Data start at row" means the first row with the actual data, not including the title. For example line 7 contains titles for column data, therefore in the **Column Data Start At Row** is 8, the number of the first row with rate data (4066667….).



	A	B	C	D	E	F	G
1	Name	Currency	Description				
2	CAD tariff	CAD	canadian tariff				
3							
4	Off-peak P	Free Seconds	Post Call S	Login Fee	Connect Fee		
5	hr {5-10}	5	1	50	5		
6							
7	Destinatior	Country	Descriptior	Interval 1	Interval N	Price 1	Price N
8	42066667	CZECH REPUB	Test	7	6	2	9
9	25551	TANZANIA, UNI	Dar Es Sa	1	1	2	
10	380	UKRAINE	Proper	30	6	0.00039	3.12345
11							
12							
13							

Figure 1-8

Template data fields

Data fields describe a single element of data, such as "invoice number", "price" or "company name". For each data field in the template you specify:

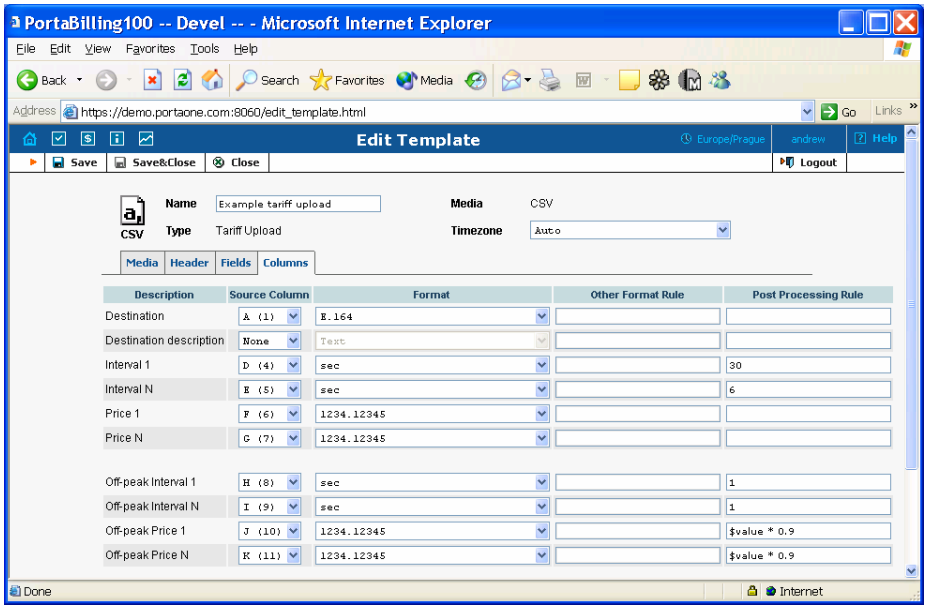


Figure 1-9

Source column (for upload templates)

Source columns define which column in the original data contains the desired value. If the data for this field is absent in the file, choose **None**. Multiple data fields can be used for the same source column. For example, if your input file has a single price for calling a specific destination both **Price 1** and **Price N** will use it as a source column.

Destination column (for download templates)

Destination columns define where to position the result value.

NOTE: For the download templates with the printable form media this option is not available since there you specify where you would like the value to appear using Layout Designer.

Format

A format describes the data fields so that:

- during upload they may be converted from that format to the unified representation in the database, and
- during upload they may be converted from the database (PortaBilling default) format to the specific format.

For example, if, for the "Destination" data field, you choose format "011**E.164**", then the template engine will know that destinations in the input file data are prefixed with the "011". So, for example, the destination "**01116041234567**" number would have "011" stripped off during upload, and the number inserted into the PortaBilling will be "**16041234567**". And during the download exactly the opposite will happen: to the number "**16041234567**" will be prepended "**011**", so the number in the file will be "**01116041234567**".

In order to provide better data quality control, PortaBilling supports several data types. Data type defines what kind of data are in the data field and possible ways to process them. Currently supported data types are:

- Destination (phone number).
- Date/time
- Number
- Time interval (number of minutes/seconds)
- Text
- Boolean (yes/no)

The data type is predefined for each data field, so you do not have to specify it. For each data type in PortaBilling there is an set of default formats, so normally you do not have to invent anything from scratch. If you encounter a data format for which there is no suitable data type in PortaBilling you may choose "Other". Then you will be able to specify a custom format rule.

Other format rule

If none of the PortaBilling default formats suit your needs, you can choose the "Other:" format and put your own formatting rule here. See below for description how the format rule work for the data field of particular type.

Number

For the numeric data fields (for example "Price 1" or "Connect fee") the value you put into the **Other Format** field is considered to be a format specification **###,###.##**, where:

- A **#** sign or digits 0-9 represent one digit of the input or output number.
- The first character from the right (non # or digit) specifies decimal separator
- The first character from the right (non # or digit) specifies group separator
- The number of characters after the decimal separator specifies precision of the number. But # means no zero padding, and 0 means zero padding.
- If the number has too many digits compared to the format specification, it will be rounded to the number of decimal digits specifies in the format
- A group separator (if present) specifies which character should be used to split groups of 3-digits in the number. If none specified, then no group separation is done.
- It is enough to specify the group separator only once, even if your real numbers will be large and will contain more than one digit group. All of the digits to the left of the decimal point are always shown, regardless of how many digits to the left of the decimal point actually are in the format.

This is mostly important for the download templates, when format or precision you use to present numbers in documents is different from the format data is stored in the database. For upload templates number format specification is mainly used to handle non-standard group and decimal separators. See the table below for the examples of using formats to convert numbers.

Format	Number	Result	Comment
#,###.##	1.6	1.6	No zero padding
#,###.00	1.6	1.60	Zero padding
#,###.00	34567.6	34,567.60	Digit groups
####.00	34567.6	34567.60	No digit groups
#.#	2.382	2.4	Rounding
#.#	2569.31	2569.3	Significant part is never truncated
#.###,##	2569.31	2.569,31	German format

# ###,##	7892569.31	7 892 569,31	Yet another custom format

Table 1-1 - Examples of the number formats.

Date and Time

This data type is used for fields containing either both date and time (e.g. **Effective from**) or just date (e.g. **Invoice date**). Please note, that when importing dates from an unusual format, it is possible to use the functionality of the Time::ParseDate module and hand off all the dirty work to it. This module is able to look at your data string and find bits of data regardless of how they are placed there or how they are separated. In order to use Time::ParseDate module for parsing the dates, choose "Auto" as a format. Unfortunately in some situations even Time::ParseDate is unable to properly identify the date. For example string "02-01-2003" could either mean February, 1st or January 2nd. In such situations exact format specification is required. The "Other" format uses syntax very similar to the one used in applications like Microsoft Excel. The elements are:

- **YYYY** – year (including century)
- **YY** – year (without the century). Note, that this format is depreciated, since it lead to the potential data interpretation problems, since "10" could be either 2010 or 1910. PortaBilling template engine considers year values 00…30 to be in the 21st century, and 31..99 to be in the 20th century.
- **MM** – number of the month (01…12)
- **MON** – abbreviated name of the month (e.g. `Jan`). Only English names of the months are currently supported.
- **MONTH** – full name of the month (e.g. `January`). Only English names of the months are currently supported.
- **DD** – day of the month with the leading zero for 1, … 9 – for example 01, 06, 14, 25.
- **D** – day of the month without the leading zero for 1, … 9 – for example 1, 6, 14, 25.
- **HH** or **HH24** – hour (24-hour format).
- **HH12** – hour (12-hour format).
- **AM/PM** – AM or PM indicator.
- **MI** – minutes.

NOTE: Very often people make the mistake of using **MM** as a format for minutes. This is incorrect, as **MM** is a format for month. Thus, **HH:MM:SS** will certainly not work icorrectly.

- **SS** – seconds.
- **TZ** – time zone name (e.g. `CET`)
- **XXX** – any combination of the non-space characters. Used in the formats for parsing the dates to skip some variable part of the

date, for example name of weekday (see examples below in the Table 1-2)

- All other symbols in the format (spaces, delimiters like **/**, **-**, **:** or **;** and other characters) represent themselves. So they will be placed as-is in the output date string and have to be present in the input string exactly how they were entered.

Let's take a look at some examples for the importing dates into the PortaBilling:

Date/time string	Description	Correct format specification
07/04/2003	July 4th 2003, US date format	MM/DD/YYYY
04-07-2003	July 4th 2003, European date format	DD-MM-YYYY
07-04-03	July 4th 2003, US date format, short year format	MM-DD-YY
11:05 am	Time, 12-hour format	HH12:MI AM/PM
11:05	Time, 24 hour format	HH:MI
18:58:00	Time with seconds	HH:MI:SS
2003-07-04 14:00:00	Date and time, ISO format	YYYY-MM-DD HH:MI:SS
07/04/03 2:00:00 pm	Date and time, US format	MM/DD/YY HH12:MI:SS AM/PM
1-May-2002 18:52	Date and time, abbreviated month name	D-MON-YYYY HH:MI
Sun, 16 Mar 03	Date and time, skip the weekday name	XXX D MON YY
12th of March, 2003	Date with "as is" elements	Dth of MONTH, YYYY

Table 1-2 - Examples of the date and time formats for upload.

The following example shows examples of using templates to convert values in the PortaBilling database into the desired format

Date/time in the PortaBilling	Format specification	Result value
01-May-2003 12:34:56	MM/DD/YYYY	05/01/2003
01-May-2003 12:34:56	DD-MM-YYYY	01-05-2003
01-May-2003 12:34:56	MM-DD-YY	05-01-03
01-May-2003 12:34:56	HH12:MI AM/PM	12:34 pm

01-May-2003 12:34:56	HH:MI	12:34
01-May-2003 12:34:56	HH:MI:SS	12:34:56
01-May-2003 12:34:56	YYYY-MM-DD HH:MI:SS	2003-05-01 12:34:56
01-May-2003 12:34:56	MM/DD/YY HH12:MI:SS AM/PM	05/01/03 12:34:56 pm
01-May-2003 12:34:56	D-MON-YY	1-May-03
01-May-2003 12:34:56	D of MONTH, YYYY	1 of May, 2003
01-May-2003 12:34:56	YYYYMMDD HHMISS	20030501 123456

Table 1-3 - Examples of the date and time formats for download.

Other data types

For the other data types (e.g. **Destination, Text** or **Interval**) there are no specific format elements. The "Other" format rule is considered to be a Perl macro, so it is evaluated and the result value of the evaluation is considered a new value. For detailed description of Perl macros' see the Post Processing rule. It receives the original string value as a parameter, and should return the converted value in the required format. The following table provides a description of which type the output value is required to be for the other format rules depending on the data field type.

Data field type	Other format rule is	Output value
Destination	Perl code	string
Number	Number format	N/A
Boolean	Perl code	string ('Y' or 'N')
Interval	Perl code	number
DateTime	Date/time format	N/A
Text	Perl code	string

Table 1-4 - Output value type for the other format rules.

Post processing rule

Sometime you need to perform some very specific conversion which is not possible with the standard PortaBilling formats. In this case you can use post processing rules. A post processing rule is a piece of Perl code which will be executed at run-time and the value it returns will become the final value for the field. This is something similar to formulas or macros in Excel, and gives you virtually unlimited ability to change your data the way you need. There are a few requirements for writing post processing rules:

- A processing rule is Perl code – one or more Perl operations separated by semicolons.
- Input value (to be processed) is in the **$value** variable.

- The final result is the value returned by the last expression in the list.

TIP: One of the most useful things you can do with the post processing rules is to create some new data, which is not based on the input data. This is required for example when some data fields (e.g. **Interval 1** and **Interval N**) are absent in the input file. Nevertheless you know that **Interval 1** is 30 seconds, and **Interval N** is 6 seconds. You can just assign those values as constants in the post processing rule.

Let's take a few examples:

Post processing rule	Input data	Result
Remove all leading white space from the string		
`$value =~ s/^\s+//;`	Aruba	Aruba
Remove all leading and trailing white space from the string		
`$value =~ s/^\s+//;` `$value =~ s/\s+$//;`	Aruba	Aruba
Add 1 in from of the phone number, but only if it does not start with 0		
`$value =~ m/^0/ ?` `$value : '1'.$value;`	6041234567	16041234567
Take only part of the input string after the hyphen		
`my @a = split('-',` `$value, 2); $a[-1];`	Norway-Oslo	Oslo
Assign a constant value to the field		
`$value = 30`	None	30
30	None	30
Replace all hyphens with spaces and remove all quotes		
`$value =~ s/-/ /g;` `$value =~ s/\"`	Germany-'Berlin'	Germany Berlin
Use an external (custom) module to translate English country name into Czech		
`use Czech::Translate;` `transl_data($value);`	Austria	Rakousko

Table 1-5 - Examples of post process rule usage.

NOTE: This manual does not cover details of the programming in Perl. There are excellent books on this subject, for example:

- **Programming Perl**, by Larry Wall, Tom Christiansen, Jon Orwant (ISBN 0-596-00027-8)
- **Learning Perl**, by Randal L. Schwartz, Tom Phoenix (ISBN 0-596-00132-0)
- **Perl Cookbook**, by Tom Christiansen, Nathan Torkington (ISBN 1-56592-243-3)

Values returned by the post processing rule should have the same type as the input value it receives. The following table provides information about

types of the input values for post-processing rules depending on the data field type.

Data field type	Input value ($_)
Destination	String
Number	number
Boolean	string ('Y' or 'N')
Interval	number
DateTime	unixtime
Text	string

Table 1-6 - Input value type for the post process rules.

Template groups

Concepts

Template groups are necessary for doing sorting/subtotals on the large sets of data. For example, we need to make a list of all of the phone calls, with subtotal for each country.

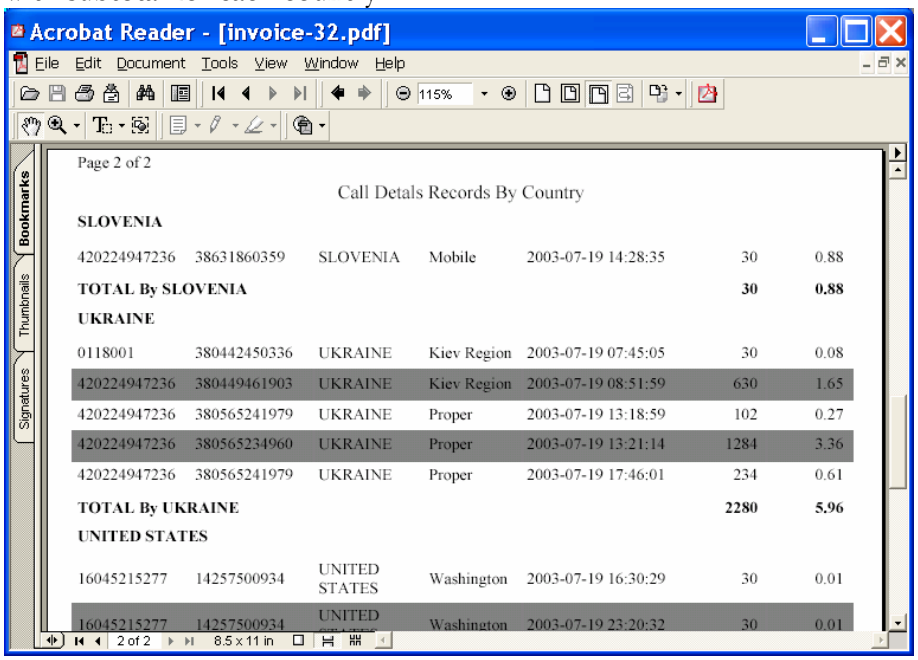


Figure 1-10

Group

Each group is a portion of data which have the identical value of some parameter or parameters. All of the input data will be sorted according to the groups, and rows in the same group will be presented together. In the example above there are three groups visible, each of them contains all of the calls made to the certain country. Groups can be nested, for example group "By Country" will contain several sub-groups "By Prefix". In the

list of groups they are sorted according to their include order, so that the outmost group is on the top, and the inmost at the bottom.

Group Break

This is the parameter, which defines to which group data should belong. All of the available data will be sorted according to this field. In the example above, the list of all calls made will be sorted by the name of the destination country (so that calls made to **Slovenia** will appear before calls made to **Ukraine**). Next, rows which have the same group break expression value will be joined in the same group, as seen in the example above for all calls made to Slovenia, Ukraine or the US.
Normally you will just choose one of the available columns to break group on (in the example above that would be the "Country" column). You can also create a custom group break, and specify any valid SQL expression.

Group Header

This is a template component which will be inserted into the document immediately before the group data. In addition to the static text, variable fields (e.g. "Destination" can be used. In the example above this is the row which contains the name of the country.

Group Footer

Similar to the group header, this element will be placed immediately after the last row of the group. It can consist of the static text and variable fields. In the example above this is the row which contains "TOTAL by" and the name of the country, and the total number of seconds and amount for this country.

Group Fields

These variable elements are very similar to the data fields in template (and they are actually based on them). They have their own name, so you can choose format and apply the post processing rule in exactly the same way as you would for the ordinary elements.

Group totals

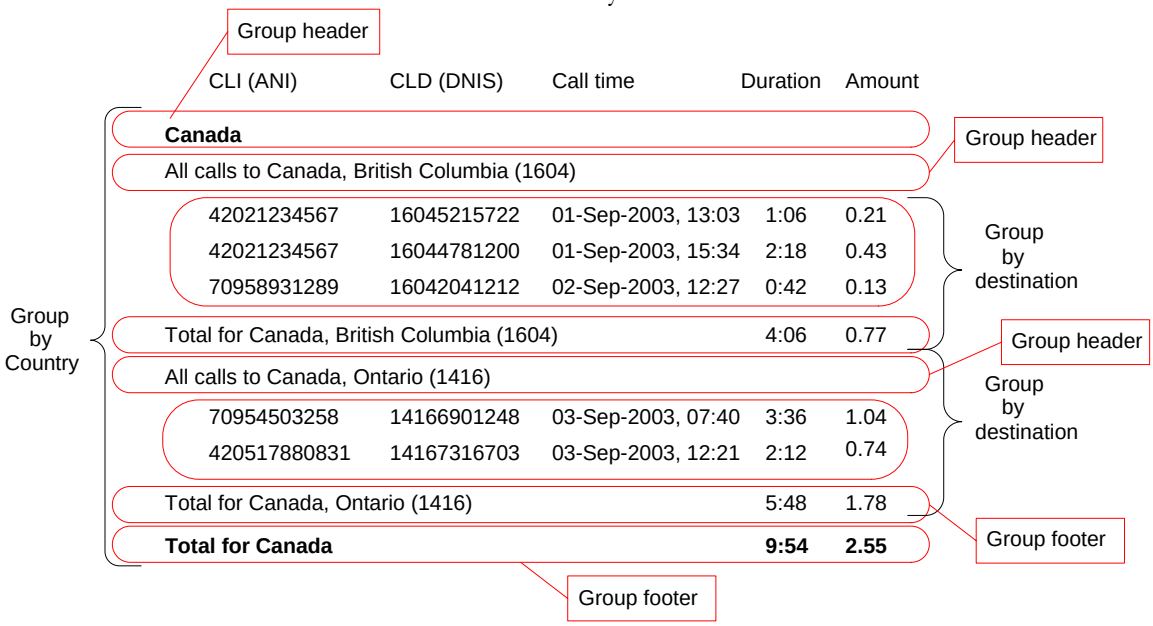
Since one group may consist of many rows of data, what would be value in the group fields in the header/footer? Fields in the group header have the value of the first row in the group. You can specify how you would like to populate the value of the fields in the group footer. Possible pre-defined options include:

- Summary (sum of the values for the corresponding column across all rows)
- Count (number of the rows in the group)
- Last value (value of this column in the last row)

You can always write your own custom total rules. This is very similar to the post processing rule: you can specify any Perl expression. The current

value of the column is in the **$value** variable, the current value of the total is in the **$total** variable, and the result returned by the expression becomes the new total.
In the example below we see what a list of calls might look like if we use two groups:

1. Break group by country (so that each group contains all of the calls made to the same country)
2. Break group by destination. This group is inside the first group, so it will contain all of the calls made to the specific destination within a certain country.



Things to remember

Destination upload

Why can't I use custom templates for destination upload/download?

There is one format available for uploading destinations at the moment, since:

- The format is fairly simple
- You almost never simply need to upload just new destinations; normally you need to input new rates in the system which requires the new destinations, and in this case the PortaBilling template processor will provide you with list of new destinations in the proper format automatically.

However, functionality to create your custom templates for destination upload or download is available, and might be included in the future releases.

How to fill in country information for the destination

For each destination entered into the PortaBilling it is necessary to provide information on which country a destination belongs to (and optionally an administrative subdivision in that country). This will help you to better keep track of the rates for the specific country and also will be used to provide a correct description in the call details and statistics for you and your customers. PortaBilling comes with the complete list of the countries which are registered with the International Standards Organization and have allocated their country ISO code for them (e.g. **US** for United States of America, **CA** for the Canada and so on). When new destination is manually entered using the PortaBilling web interface, you can pick country from the drop-down select menu. When importing many new destinations at once, the country information has to be provided for each of them in the form of the two-letter ISO code. Why not the country name? Unfortunately country name could be very ambiguous. For example, *UK*, *United Kingdom*, *Great Britain*, *Britain*, *England* are all synonyms for the same country (which could also could be called *The United Kingdom of Great Britain and Northern Ireland*) and you may never know which version of the name your vendor will decide to use. Plus the country names in the local languages (e.g. *Velka Britanie* in Czech), spelling errors (e.g. *United Kigndom*), punctuation marks and so on complicate the matter. That is why PortaBilling uses country codes instead, which exactly describe which country is intended.

Does this mean that you now have to manually fill in country codes for several hundred missing destinations? No, there is a better way!

TIP: When uploading destinations with empty country codes, PortaBilling will try to find country information basing on the data which are already in the database. This means, for example, if the prefix **61** already exists in the database with the country assigned as Australia, that when you upload a new destination with prefix **61881** PortaBilling will automatically fill in the country information for you.

NOTE: Of course, for this method to work efficiently, you need to have some initial set of destinations already in the database. PortaBilling comes with the default set of almost 3000 destinations, which covers most of the countries in the world and popular destinations within those countries. You can download it using **Get default set** button in the Destinations screen, customize it according to your needs and then upload.

Tariff download and upload

There are few helpful features regarding the upload of tariffs.

New destinations files are created automatically

When you try to upload a rate for a phone prefix which is not entered in the destinations, that rate will not be inserted. However PortaBilling will create list of such missing prefixes in the format suitable for upload.

Integrity checks during the tariff upload

It is a very common error to upload the wrong file (e.g. from a different vendor or in a different currency). In order to prevent this situation, the following improvement has been made in the Portabilling100:

- It is possible to include tariff name and tariff currency in the data. The default templates for upload and download of tariffs, provided with the Portabilling100, include it as well.
- When you try to upload a tariff, the template processor compares **Tariff name** and **Currency** in your file to the tariff you are trying to upload. If they do not match, the upload will be rejected. This prevents you from such errors.

This feature works only if you have **Tariff name** and **Currency** in your file.

NOTE: PortaBilling 2.3 and older tariff export format did not include tariff name or currency. So this option is not available when importing tariffs from PortaBilling version 2.3 or older.

You can only upload rates effective immediately or sometime in the future

Since billing is done in the real time, a call is billed as soon as possible and a CDR is written to the database. So in order to modify CDRs which are already in the database re-feed should be used. So if you are copying data between tariffs and there is an **Effective from** column included in the data, make sure that you replace values which are in the past with an empty value. An empty value in the **Effective from** column means "immediately".

2. Invoicing

How does invoicing in the PortaBilling work?

PortaBilling provides ready to use invoicing functionality. You can design how your invoices should appear; invoices are created automatically based on the calls the customer has made; invoices are stored as a PDF file and can be viewed online, or printed. They are also automatically emailed to the customer if he has an email address set up in his properties.

Who is invoiced?

Only customers (both resellers and retail customers) are invoiced, since only they have all the required information for invoicing, such as address, tax payer ID and so forth. Individual accounts are not invoiced.

How often?

Each customer has his billing period, which could be:

- Daily
- Weekly
- Bi-weekly
- Monthly

When the current billing period is over (for example on the first day of the month) a summary of all the calls made by customer is calculated. Customer call statistics are generated and stored on the server (optionally they could also be automatically mailed to the customer) and the invoice is created.

Note, that the billing period is based on customer's time zone. For example if customer has a weekly billing period and his time zone is Australia/Melbourne – his statistics would cover period from 24-Mar-2003 00:00 Melbourne time (23-Mar-2003 14:00 GMT) until 31-Mar-2003 00:00 Melbourne time (30-Mar-2003 14:00 GMT). For another customer, who also has the weekly billing period, but New York time zone this billing period will cover calls made in the period 24-Mar-2003 00:00 New York time (24-Mar-2003 05:00 GMT) until 31-Mar-2003 00:00 New York time (30-Mar-2003 14:00 GMT).

When are statistics available?

Statistics are generated with a sufficient delay to make sure that the billing period has ended for a particular customer (according to that customer's time zone). Also, statistics calculation takes place only at a certain time of day (usually during the night hours, when there is the least amount of activity in the system). Weekly invoices cover Monday-Sunday. Assuming that your system does statistical calculation from 2am until 8am Central European Time every day, then statistics for those customers who have a time zone in Asia or Europe should be available each Monday morning.

For customers with a time zone in the US, the billing period will not have ended yet when statistics are calculated on Monday morning; thus they are created on Monday night and made available Tuesday morning.

There is currently an invoice-on-demand functionality in PortaBilling, i.e. invoices should match customer billing periods. However, if you want to include a certain amount (for some special service) in the invoice for your customer manually, this is easy to do: simply perform a balance adjustment. This will modify the customer's balance and will also create a CDR record, which will be included in the customer's next invoice.

Invoice total

There are two methods available for calculating the invoice total:

All charges in the period

This is the simpler method (and the only one which was available prior to PortaBilling100 Maintenance Release 9). The invoice contains all of the call charges in the specified period, with the invoice total equaling the sum of these charges. Balance adjustments, refunds and payments made by a customer via the online payment system are not included, since these refer to transactions which have already been made and should not occur again. So, for example, if during August a customer makes calls for $120 and pays $100 for his services in July on August 10th, his invoice will show $120. Of course, this payment will be recorded in the system and affect the customer's balance.

Including the previous balance

The invoice total will be calculated as:

- Previous invoice total
- + Sum of all charges during the period (call related or manual charges)
- - Sum of all payments or refunds made during the period

So, taking our previous example, if the customer's invoice for July was $110, and he paid $100 and made calls for $120 during August, his total on the August invoice will be $110 + $120 - $100 = $130

The invoicing modes are controlled by the `New_Style_Invoice_Calculation` switch in the `porta-admin.conf` file.

What the invoice is based on

For a reseller, the call charges on the invoice will be based on the sum total of all his CDRs for a given period.

For a retail customer, the call charges on the invoice will be the sum total of all the CDRs (call charges and maintenance fees) for the credit accounts which this customer has (debit accounts are prepaid, thus they have been already paid in full and there is no need to invoice for calls).

Invoice template

The invoice template is a specially formatted HTML document, which is created using the PortaBilling Layout Designer (see the following chapter for a detailed description of the layout designer). It defines how the invoice should appear with information on fonts, colors, pictures, placement of the components. So when you view the invoice on the web interface, or when the invoice is automatically mailed to the customer – the template engine takes the invoice template and just fills data fields (such as **Header.Invoice Number**) with their real values (for example **1001** for the invoice number). Result of this processing could be a HTML document or PDF file.
Requirements for the invoice are very different from country to country (and even in the same country). The default invoice template, supplied with the system, is suitable for British Columbia, Canada and provided just as an example. Make sure you create your own invoice template and assign it to customers.
Two predefined invoice templates are supplied with PortaBilling: "Single-page invoice" and "Multi-page invoice" – what is the difference between them?

Single page invoice

This is a traditional-style invoice, similar to the invoice template in the older PortaBilling version. It contains one page with an invoice header (your company name, customer name, etc.), invoice fields (invoice number, invoice date) and invoice footer (subtotal, total).

Multi-page invoice

This template's first page is identical to the single-page invoice template. In addition to this, it may contain multiple pages with details of the calls related to the invoice. In the layout designer, you can use the  button to switch between the main page and the call details page.
You can include all of the calls made by the customer, or only summaries.

Billing, invoicing and taxes

Does the PortaBilling perform billing including or excluding taxes?
Actually PortaBilling is a rating engine, so it just depends on how you will be using it. PortaBilling does all the calculations basing on the rates you enter into the system, tariff parameters and call duration – in this way there is no difference if price already includes taxes or no.
So there are two scenarios available for dealing with the taxes, both have their own advantages and drawbacks.

Calculating the CDRs without the tax calculations

You enter rates into the PortaBilling without the taxes. CDRs are calculates without the taxes. When you create an invoice for your customer, you can include all the necessary tax calculations there. Advantage of this method is unlimited flexibility – you can implement support for any taxation scheme you need. The big disadvantage of this method is though, that since all of the CDRs exclude taxes, the customer balances will also be without tax. It makes it difficult to match customer balances in PortaBilling with the actual balances, since customers pay you amounts including the tax. Also this method does not work well with the pre-paid cards – when the customer buys a $10 calling card, he expects to spend $10 on calls and he is interested what his price per minute is without complications like taxes.

Calculating the CDRs with the tax calculations

You enter rates into the PortaBilling including the taxes, so it is "final" price. CDRs are calculated accordingly (with tax being included). In this case balances are correct, and pre-paid products work fine. The only complicated part is to produce proper tax information on the invoice (if it is required), since it has to be done the opposite way (i.e., you have to deduce tax from the amounts).

This is the recommended method, since it provides accurate billing for both the pre-paid and post-paid products. The default invoice template supplied with the PortaBilling is an example how you could set up invoicing in this way.

Reverse calculating taxes basing on the invoice total

In the event you wish to price with the tax included in the rates, you need to show proper tax information on the invoice. There are 5 data fields available for different types of taxes (for example GST and PST in Canada) which you can fill in using different formulas. They all receive a total invoice value as the input value, and then you can calculate the actual tax value by your own formula in the **post-processing rule**.

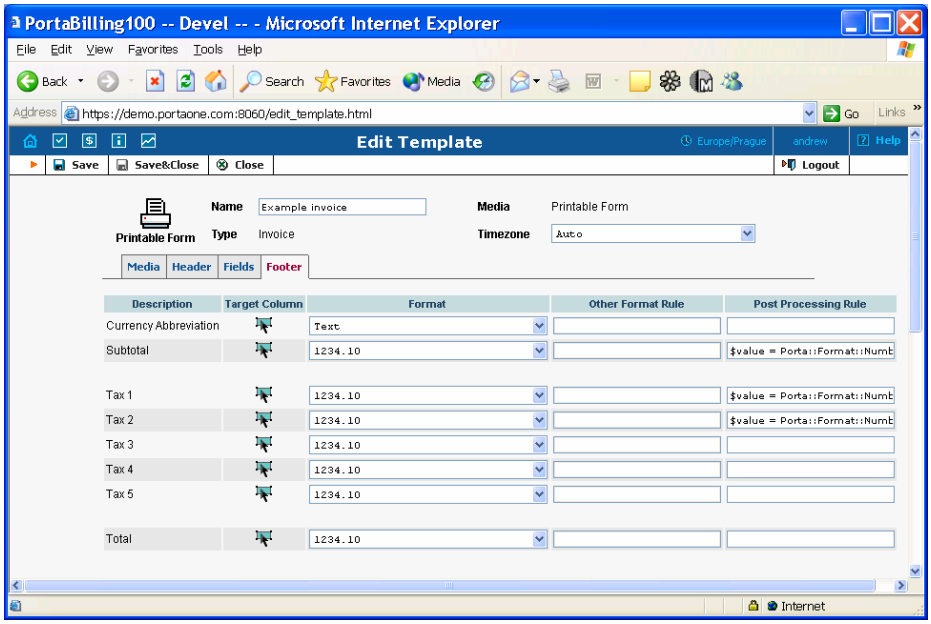


Figure 2-1

There is function in the PortaBilling template library which can be used to easily calculate the amount of certain tax types. The function `Porta::Format::Number::backcalcTax` takes three parameters:

- Total amount
- Total tax percentage
- Percentage value of this tax type

Let's take the example of British Columbia, Canada. We would need to have two types of tax: GST (7%) and PST (7.5%). Then we just have to assign for the **Tax1** data field post processing rule of

```
Porta::Format::Number::backcalcTax($value, 7 + 7.5, 7.0);
```

and for the **Tax2** post processing rule would be

```
Porta::Format::Number::backcalcTax($value, 7 + 7.5, 7.5);
```

As you can see it is quite easy. Instead of writing `7 + 7.5` we could have written `14.5`, but why do the calculations if template processor can do it for us?

Finally, the only thing is necessary is to place **Footer.Tax1** and **Footer.Tax2** fields on the invoice where we want them to appear, and label them appropriately.

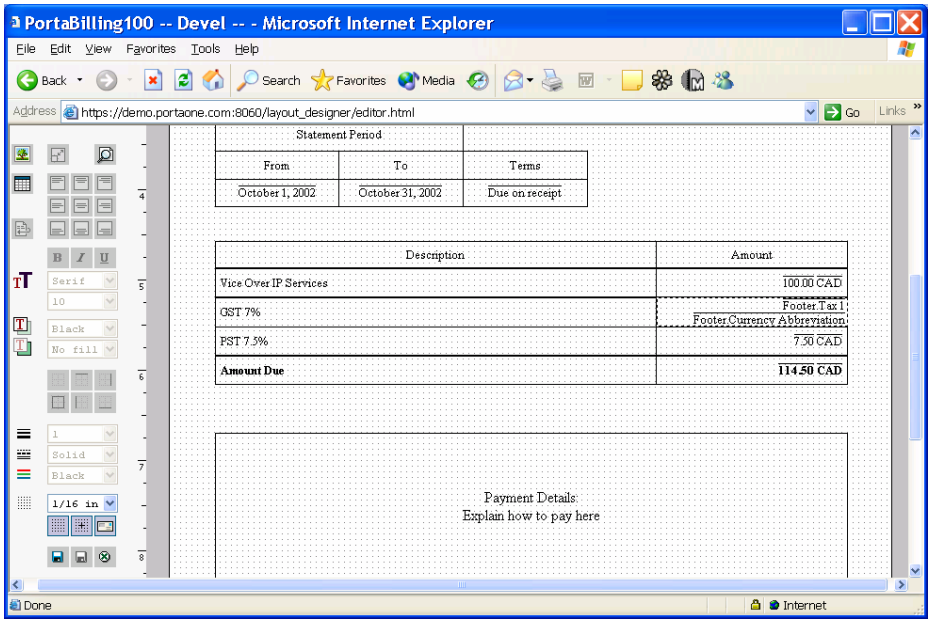


Figure 2-2

3. Layout designer

Overview

Layout designer is a visual (WYSIWYG) tool for creating document templates. It is implemented using only Javascript and DOM HTML extensions, so it will work in the MS IE or Mozilla web browser, no installation of extra components on the user's computer is required. When you design the template in the layout designer, you define how the document should look like: location of the components, pictures, fonts, colors. You also place the data fields on the document. These are the containers for the real data, which define where the data element should appear. For example, on the template for the invoice you put **Invoice.number** data field in the place where you would like invoice number to be printed on your invoices (top-right corner in the example below). When the actual invoice will be generated basing on this template, the **Invoice.number** will be replaced with the actual invoice number (e.g. **557**). Make sure you insert the data field using the pop-up menu, because if you just type **Invoice.number** it will be regarded as a text string and presented on the invoice exactly as you have typed it.

NOTE: During the design data fields will be represented on the screen by the example data, e.g. **25** for the invoice number. This data is provided only for your convenience and are used only in the design mode. All of example data are shown in the over lined font, which makes it easy to recognize them.

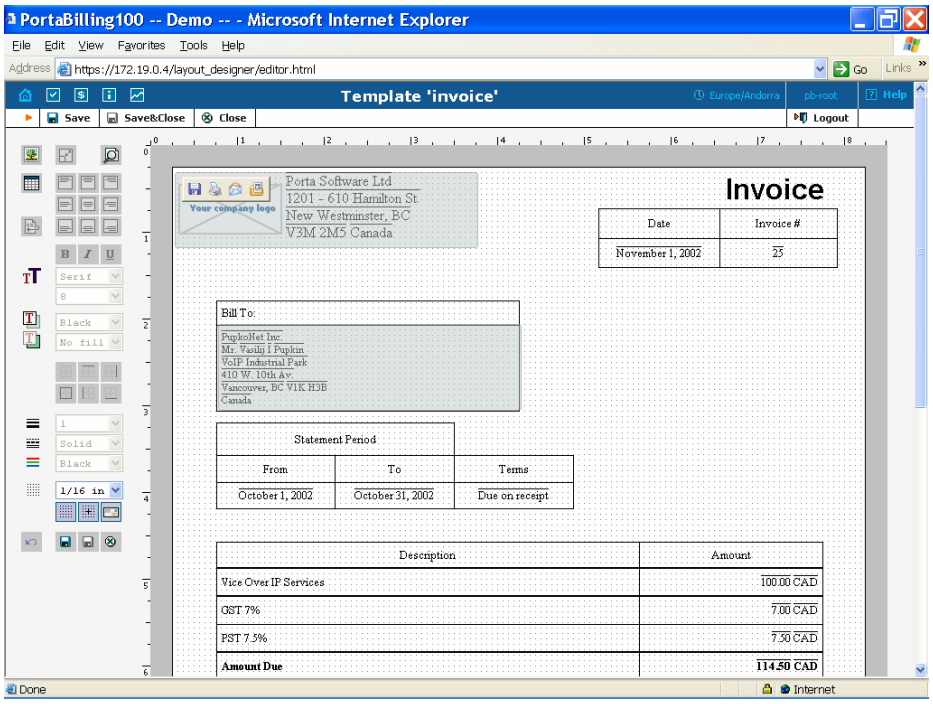


Figure 3-1

Layout designer consists of two main parts:

- Toolbar (on the left). Contains buttons to controls for manipulating content in the editor window.

- Worksheet (on the right). Contains the document itself.

General document properties

Page setup

In order to access page setup properties, right click on the template window, on the clear background (and not on the object such as table or image).

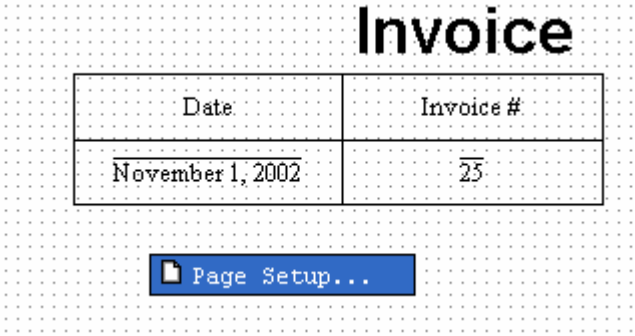


Figure 3-2

Choose "Page Setup" item.

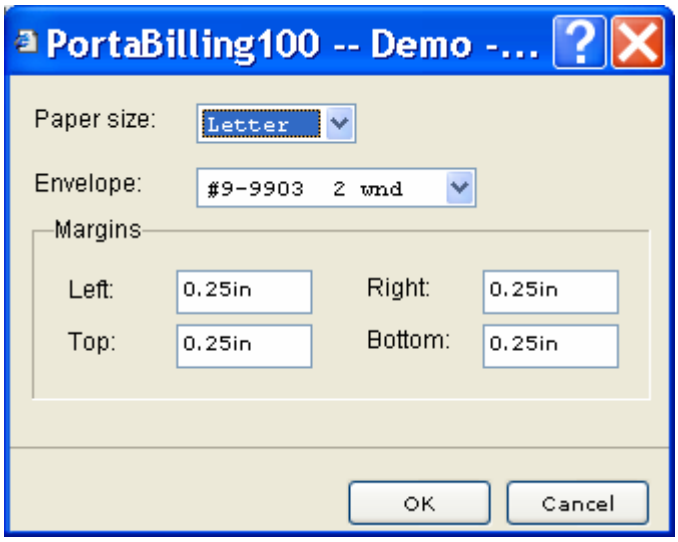


Figure 3-3

You will be able to choose paper format, margins and position of the envelope's windows. Consult the online help for the detailed description of the different envelope styles.

Grid properties

Grid helps you to easy adjust objects in the template. There are few buttons at the toolbar which control grid's behavior:

-  - turn the grid on or off (by default is on).

-  - turn snap to grid on or off. When snap to grid is on, object edges will always be on one of the grid lines, so you cannot put object between the grid lines when you drag it or resize.
- Select menu allows you to choose size of the grid cells.
-  - shows or hides envelope window on the template.

Working with the images

You can add your images to the template. These images will be stored on the server then and used by any one who will be browsing document, created from the template. To add a new image:

- Click on the empty background of the template (so no other object such as picture or table is selected). **Add picture** button should become enabled.
- Click on the  **Add Picture** button.
- **Open File** dialog will appear, choose the picture on your local disk. Any picture format, which is supported by the popular web browsers can be used (.JPG, .GIF, .PNG).
- Click **Open** and the picture will be transferred to your template. You can now move it or resize.

Placing the data elements

The main object of the Layout Designer is Table, the placeholder for all inserted information. It contains text or data fields, which otherwise could not be placed into the template. To add a new table to the template:

- Click on the empty background of the template (so no other object such as picture or table is selected). **Add Table** button should become enabled.
- Click on the **Add Table** button.

A new table will be added to the template. This is a very basic table, which consists only from a single cell – however you can change it. To move the table to the desired place, simply drag it there (click on the table and move the mouse without releasing the mouse button). To do advanced operations with the table (e.g. add a row), right click on the table – so the menu will appear.

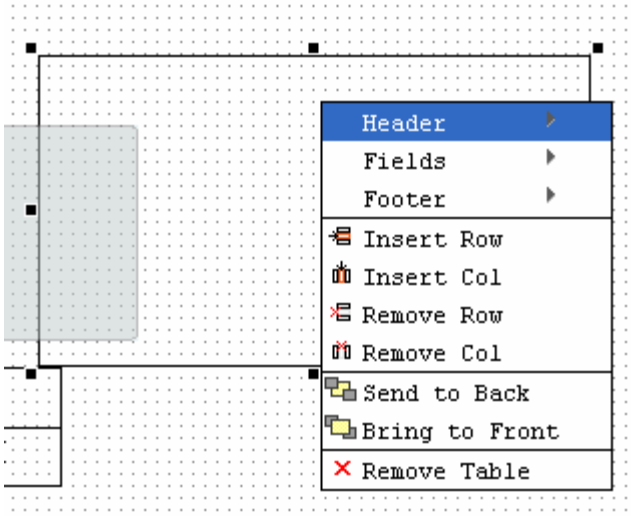


Figure 3-4

Table or table cell within a table can be moved and scaled dragging the cells handlers. Double click a table or a cell within table to enter the Cell Editing mode, where you can type something in or edit cell's text. To change attributes of the cell's text, make sure the cell is selected (8 black square handlers are visible) and set text attributes in the toolbar on the left.

Changing text attributes of a cell from the Toolbar applies to all text in the current cell. To change attributes of part of text, enter the cell editing mode (double-click), select part of the text, and use keyboard shortcuts listed below.

To insert the data fields into the cell, do the following:

- Right-click on the cell where data field should be inserted.
- In the top part of the menu click on the component name to which data field belongs (e.g. **Header**). Sub menu with the list of all data fields in this component will appear.
- Click on the name of the data field you would like to insert.

You can always enter into the cell edit mode and add text before or after the data field. Also, by removing all of over lined data field name you can completely delete the data field from the cell.

Changing the text attributes

You can apply multiple formatting styles to the text elements in the table (this includes both static text and data fields), notably:

- Text alignment within the cell
- Text font and font attributes, text color
- Table cell borders and border style

Consult the online help for the complete list of all possible text operations.

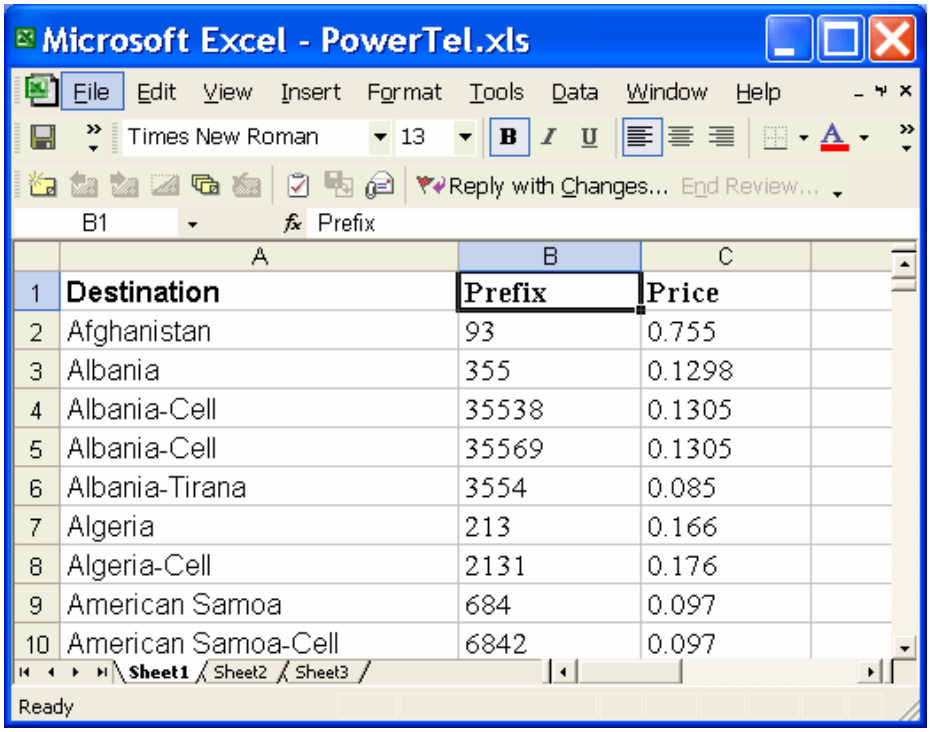
4. Practical session

In the following chapter we will try to solve situations very similar to what you might encounter. Of course, details will likely be different in your case, but he we will cover general techniques.

Importing rates from your vendors into PortaBilling

Input data

Your vendor PowerTel sends you list of his new rates for termination. You need to import them into the PortaBilling.



	A	B	C
1	**Destination**	**Prefix**	**Price**
2	Afghanistan	93	0.755
3	Albania	355	0.1298
4	Albania-Cell	35538	0.1305
5	Albania-Cell	35569	0.1305
6	Albania-Tirana	3554	0.085
7	Algeria	213	0.166
8	Algeria-Cell	2131	0.176
9	American Samoa	684	0.097
10	American Samoa-Cell	6842	0.097

Figure 4-1

Input data analysis

First of all let's try to identify if file I suitable for upload at all, which structure is has, and where the data is located.

- The file is in Microsoft Excel format.
- There is no tariff name, currency or description in the file. In other words there is not any **Header** data.
- There is no information about off-peak periods, connect fees or other **Field** parameters either.
- There is data for the rates. It is located on the worksheet called **Sheet1**. Each rate is on a separate row, and the two most important parameters: phone prefix and price are present. The first row with data is at the row **2**. The missing information (for

example duration of the billing intervals) should be obtained by some other means (for example contacting the vendor directly).

So the file contains necessary data and may be imported.

Create a template

From the main menu, choose **Templates** to open the template management screen.

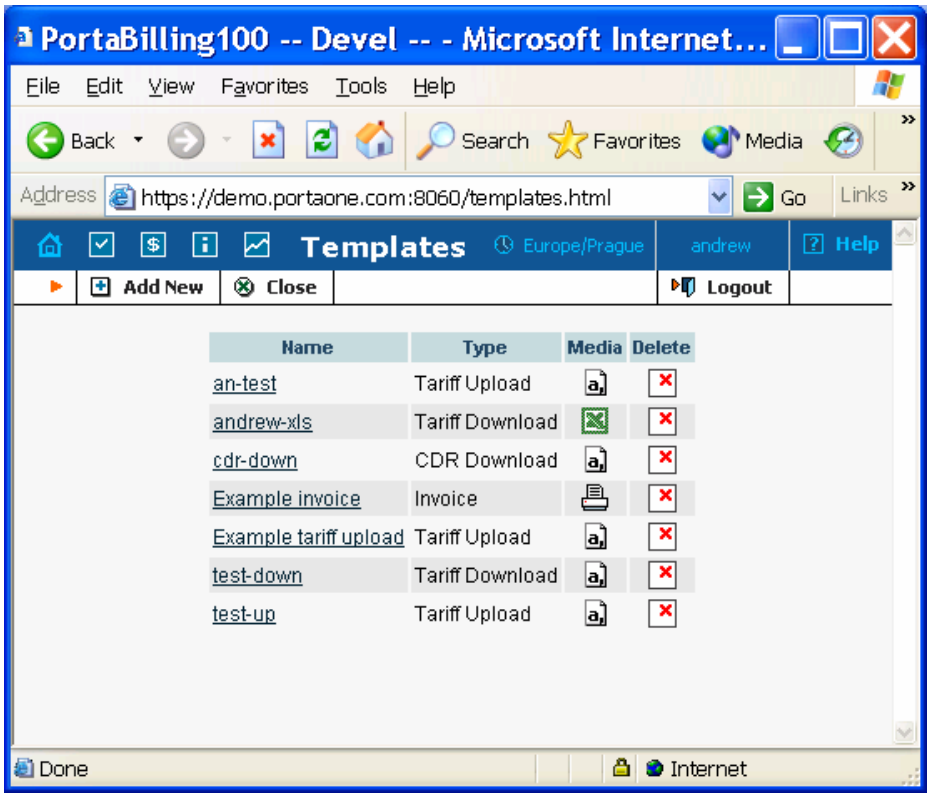


Figure 4-2

Click **Add new** and fill in information about the new template. The template type is **Tariff upload**, the media is **XLS**, and the template is created as a copy of the **Tariff upload XLS** (default PortaBilling template). Press **Save**.

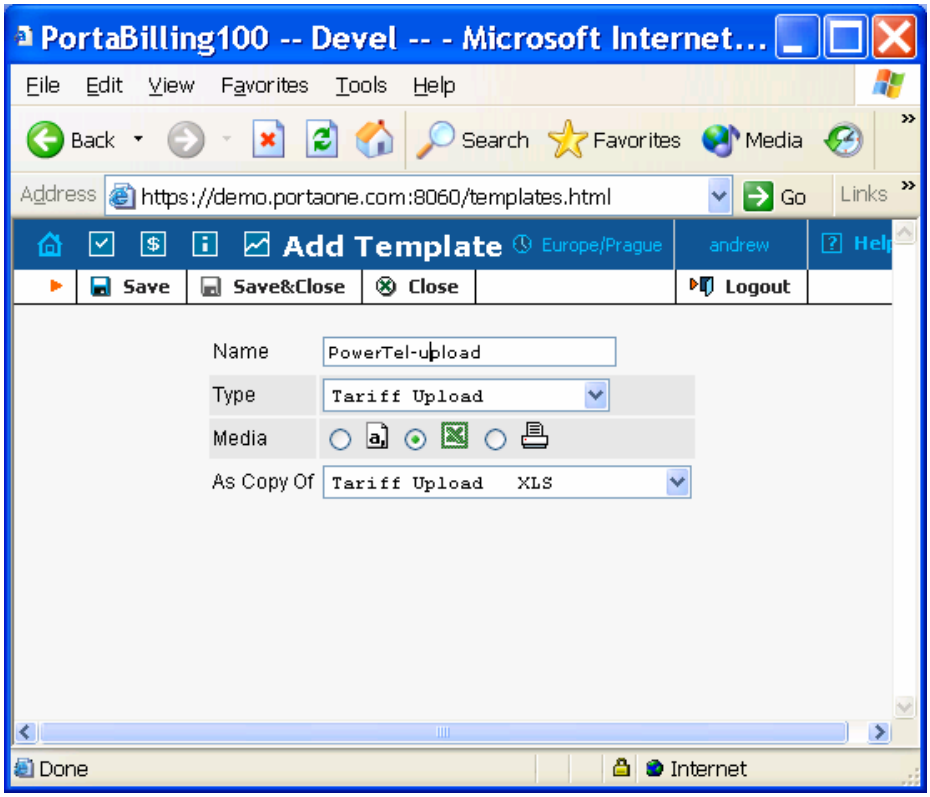


Figure 4-3

Template parameters

After the new template is created, we need to fill in media parameters and formats for the individual data fields.

Media parameters

Our media parameters would be:

- **No Header**, since there is no header data in the file. This completely disables the component, so values in the **Header Worksheet** and **Header Data Start At Row** are ignored.
- **No Fields**, since there is no field data in the file.
- **Column Worksheet** is "Sheet1" and **Column Data Start At Row** is "2"

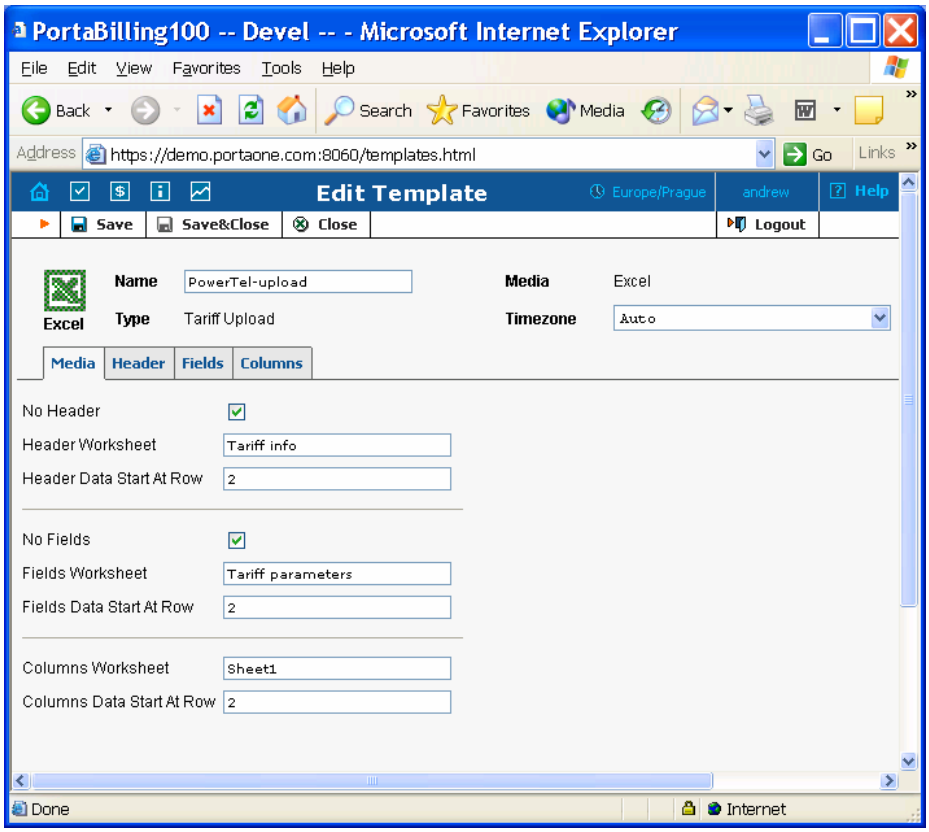


Figure 4-4

Header

Since there is no header data in the file, and we activated **No Header** switch in the media parameters, this tab can be ignored.

Fields

Since there is no field data in the file, and we activated **No Fields** switch in the media parameters, this tab can be ignored.

Columns

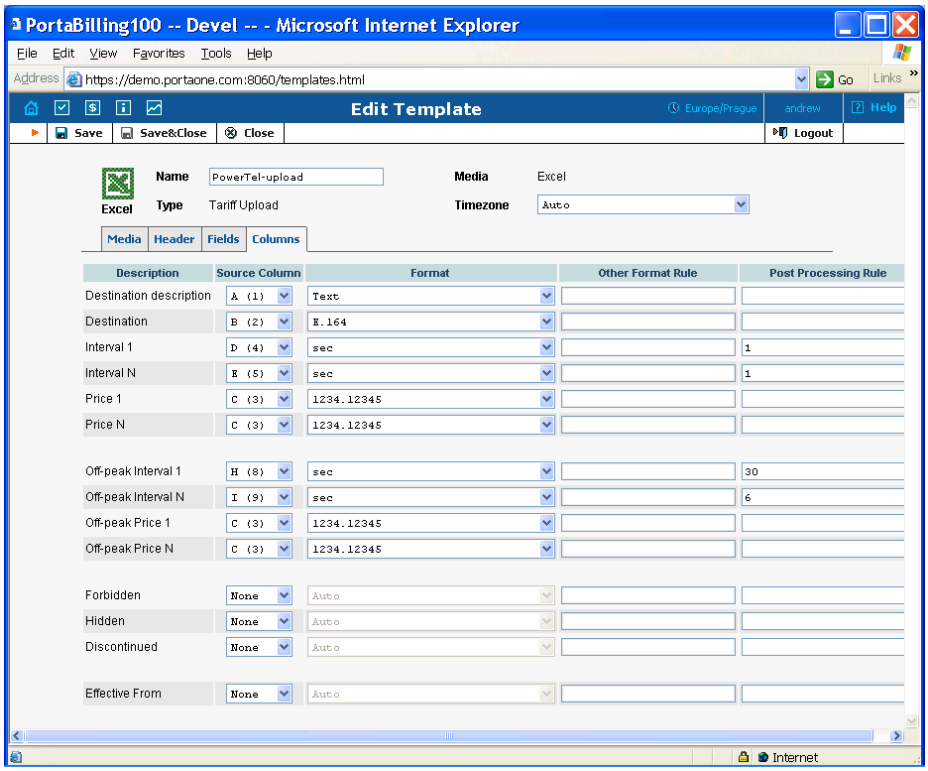


Figure 4-5

- **Destination description** – data is in the first column, so we choose **A (1)** as a source column. No extra formatting is reqired. **Hint**: in order to extract the location description ("Tirana") from the destination description "Albania-Tirana" the following post-processing rule can be used: `$value =~ s/^.+\-//; $value;`
- **Destination** – data is in the second column, so we choose **B (2)** as a source column. Since phone prefixes are already in the E.164 format, we choose **E.164** as a format, and no post processing rule is required.
- **Interval 1** and **Interval N** – there is no data for them in the file, so we choose **None** as a source column. Let's assume that vendor provided us with the information that he uses 1 second billing intervals during peak hours and otherwise 30/6 billing intervals. Then we just have to put those values as constants in the post processing rule.
- **Price 1** and **Price N** – since, in our case, the price (per minute) for the first interval and the rest of the intervals are the same, we use the same source column **C (3)**. We use format **12345.12345** since numbers are with "**.**" as a decimal point.
- **Offpeak Interval 1** and **Offpeak Interval N** – we fill them with constants, similar how it was done with the **Interval 1** and **Interval N**.

- **Offpeak Price 1** and **Offpeak Price N** – since the price is the same for on-peak and off-peak, we use the same source column and format as for the **Price 1** and **Price N**.
- **Forbidden**, **Discontinued**, **Hidden** – there is no data for these data fields in the file, so we choose **None** as a source column.
- **Effective from** – we choose **None**, so rates will be active immediately.

After we are done with the template parameters, click **Save&Close** and then go to the tariff management screen.

Create a tariff

From the tariff management screen press **Add New** in order to create a new tariff.

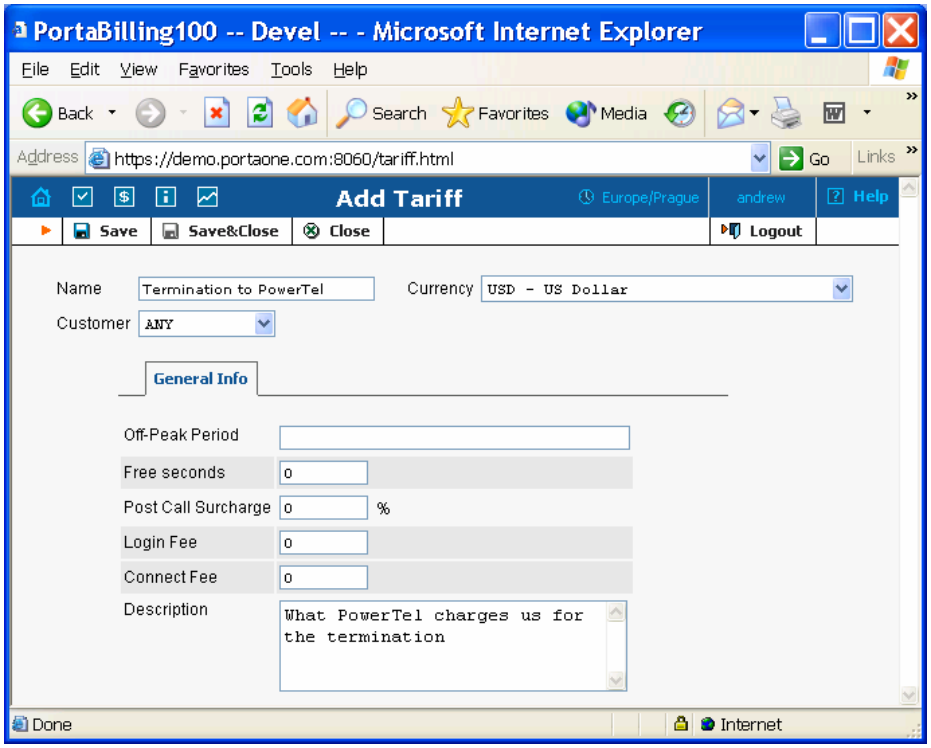


Figure 4-6

After tariff name and other parameters are filled, we can press **Save**, so tariff will be saved and extra tabs will become available. Now in the **Default Upload Template** let's choose the template we have just created – **PowerTel-upload** and click **Save**.

NOTE: Always save the changes in tariff information by clicking save before proceed to the upload of the tariff data.

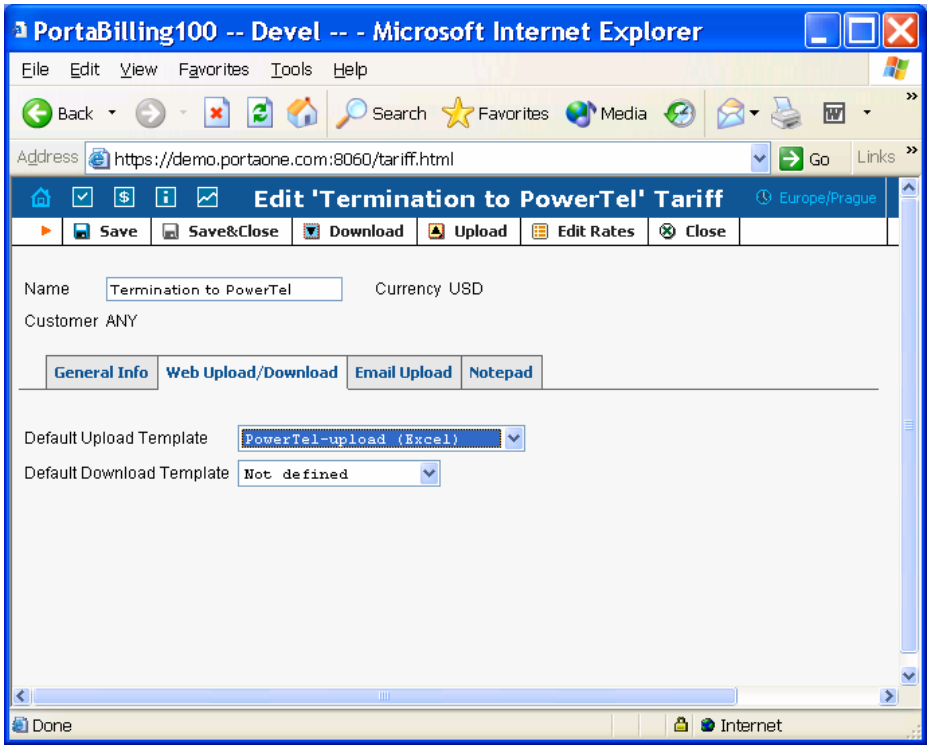


Figure 4-7

Now we are ready to upload the tariff data.

Upload tariff data

Click on **Upload**, so the upload screen will open. It is possible to type the name of the input file manually, but it is more convenient to use **Open file** dialog (click on the **Browse…** to invoke it).

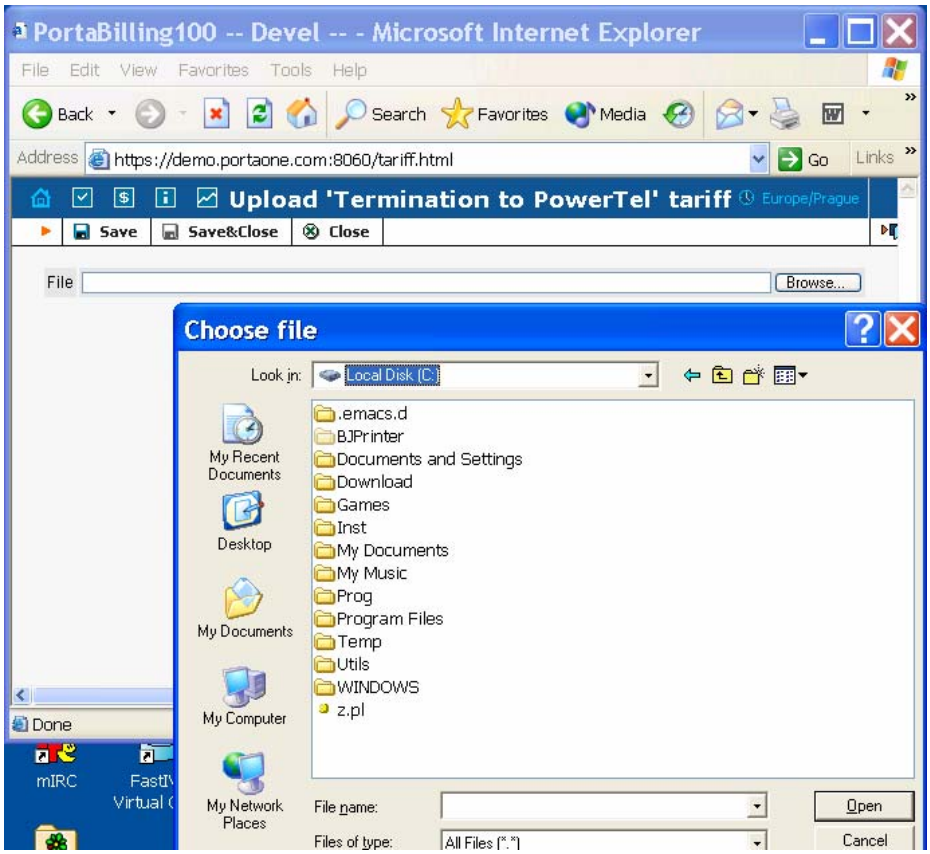


Figure 4-8

After the file name is entered into the form, we can click **Save&Close**. This will transfer the file to the server, where the processing will start.

NOTE: If you had the input file open in Excel, make sure that you **Save&Close** it first. Excel locks the file, so no other applications can access it.

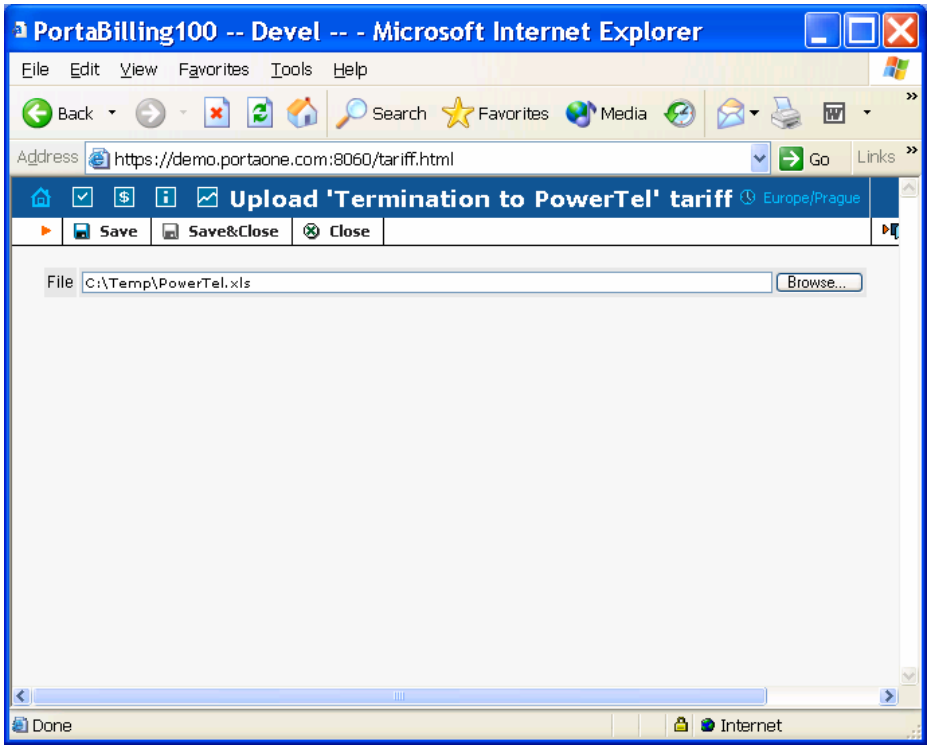


Figure 4-9

Dealing with the upload errors

Let's assume that there have been some errors during the upload. We will see the following message then.

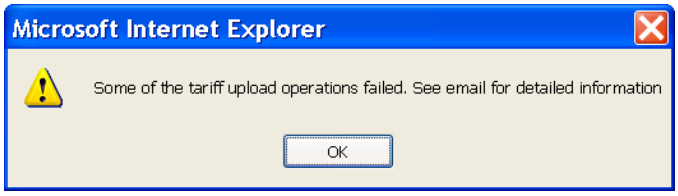


Figure 4-10

Types of errors

There are few main reasons why rate will not be inserted:

- There is some error in the input data. For example number does not match the format, so some field is empty. All such rows will be grouped together in a single file, so simplify their correction and re-importing. So if we have tried to upload 5000 new rates and 30 of them had some error, we have to edit the file which contain only 30 lines, and upload again only that file, and not the original one.
- A destination is missing. For example, we have tried to upload rate for the **420602** (mobile in the Czech Republic). However, for the purpose of maintaining data quality, each phone prefix first have

to de defined as a destination. At this step some extra information associated with the prefix, for example the country name of the custom description. So if destination **420602** does not exist yet, this rate cannot be inserted. In this case we will have to create a destination first, and then try to upload the rate again.

- Attempt to upload rate with an "effective from" date in the past. Since PortaBilling does real-time billing, as soon as call has been completed, CDRs are created and accounts/customers/vendors are billed. So if you try to upload rate with the effective time 1 hour before the current time, this would not change anything, since the calls which were made during this hour have been billed already. Moreover, it could only lead to confusion later – since it might seem that calls were billed by some rates, which were not effective at the moment. So you can only upload rates with the effective time "immediately" or in the future.

NOTE: You can upload many rates for the same destination within the same import session, provided that their effective from dates are different. For example, you can upload rate, effective immediately, effective 01-Apr-2003, effective 01-May-2003 and so on. It does not make much sense to upload more than one rate for the same destination effective immediately in one upload session.

Import result in the email message

In any case, you will receive the email which contains detailed information about the import results. The email message would look similar to the one on Figure 4-11. There will be detailed log, explaining why each particular row could not have been uploaded. Also there will be one or more files as attachment.

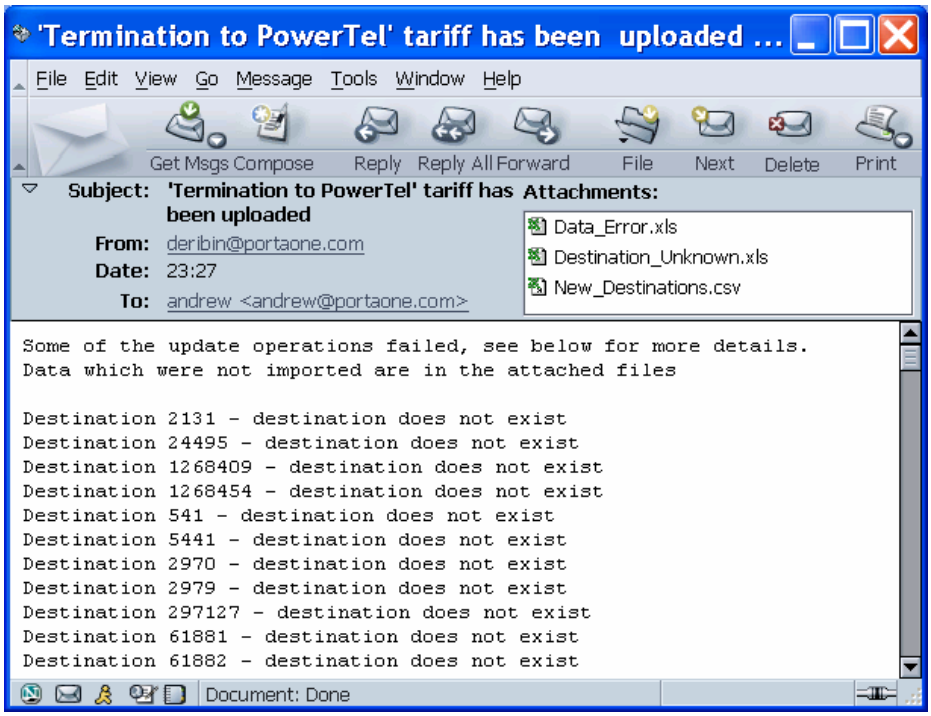


Figure 4-11

- **Data_Error File**: contains lines from the input file which were not imported because of incorrect data (missing value, incorrect format, etc.).
- **Destination_Uknown File**: contains lines from the input file which were not imported because the phone prefix for this rate does not exist in the database.
- **New_Destinations File**: contains a list of phone prefixes which seem to be missing in he database in a format suitable for destination upload.

The names of those files are defined by PortaBilling, and the extension depends on the actual input file format. For example, if you try to upload a CSV file, then Data_Error and Destination_Uknown will also have a .CSVextension.

Fixing the errors in data

Let's open the Data_Error.xls file and see its contents.

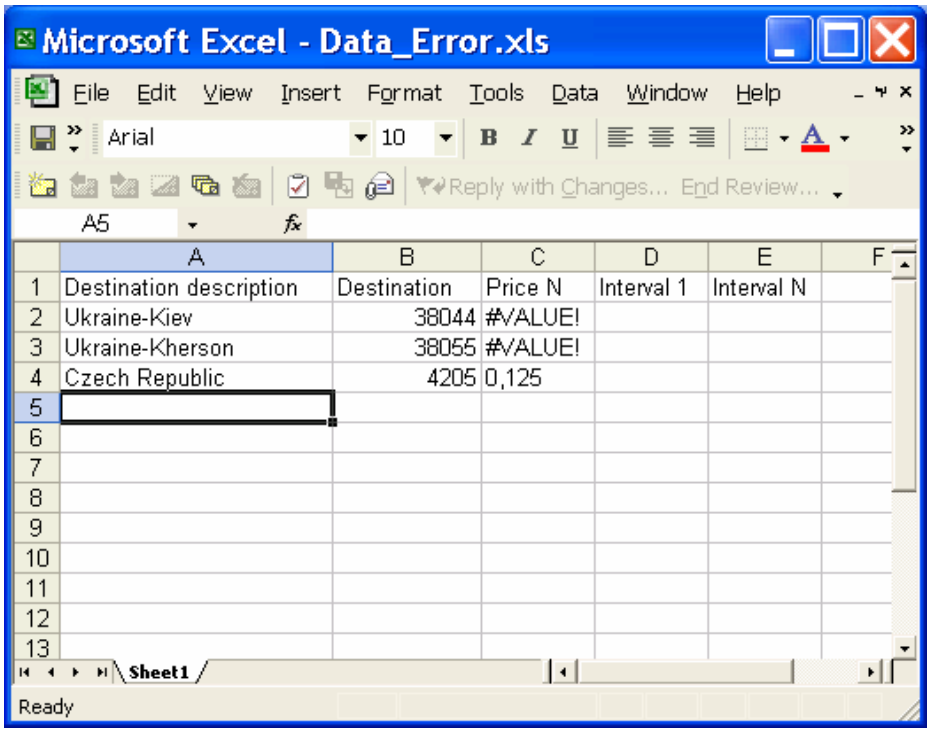


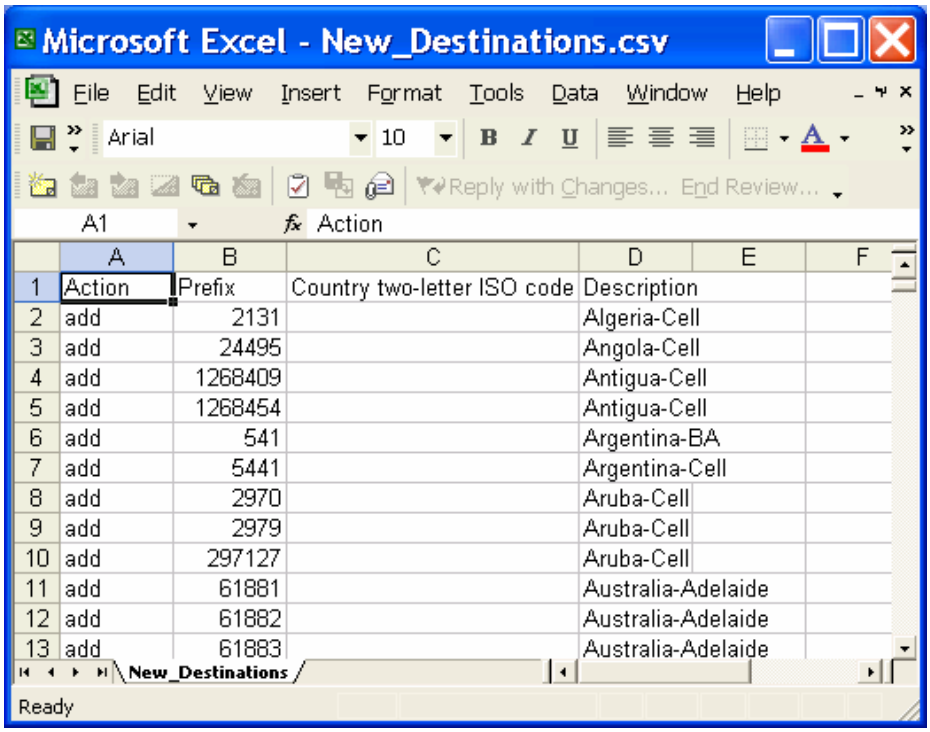
Figure 4-12

We can see, that rates for the destination in Ukraine were rejected, since there is some a formula error in the original file. And the price for the Czech Republic has "," (comma) instead of the "." (dot) in place of the decimal point. After we fix those errors, we can upload the file again and it should go smoothly this time.

Fixing the missing destinations

First, let's take a look at the New_Destinations.csv file.

NOTE: Regardless of the input file type (.CSV, .XLS), the file type for destination upload is always .CSV file as there is only one template used for destination uploads – the default one. It is possible to create a user-defined destination upload/download template, and it is will be possible to apply them to data in a future release.



	A	B	C	D	E	F
1	Action	Prefix	Country two-letter ISO code	Description		
2	add	2131		Algeria-Cell		
3	add	24495		Angola-Cell		
4	add	1268409		Antigua-Cell		
5	add	1268454		Antigua-Cell		
6	add	541		Argentina-BA		
7	add	5441		Argentina-Cell		
8	add	2970		Aruba-Cell		
9	add	2979		Aruba-Cell		
10	add	297127		Aruba-Cell		
11	add	61881		Australia-Adelaide		
12	add	61882		Australia-Adelaide		
13	add	61883		Australia-Adelaide		

Figure 4-13

We can see that phone prefix and destination description were transferred from the original input file. "Add" in the **Action** column says that we are going to add new destinations. What about the **Country two-letter ISO code** column, which is empty?

Country information for the new destinations

As you know from the previous chapters, PortaBilling is able to automatically match the country for the phone prefix basing on the data which is already in the database. For example, in our case, if in the database already has destination **61** which states that it is Australia, and we try to upload destination **61881** (line 11 in the on the Figure 4-13),PortaBilling will fill in the country information automatically.
So we need to go to the **Destinations** screen, choose **Upload** there, enter the name of the file and click **Save&Close**.

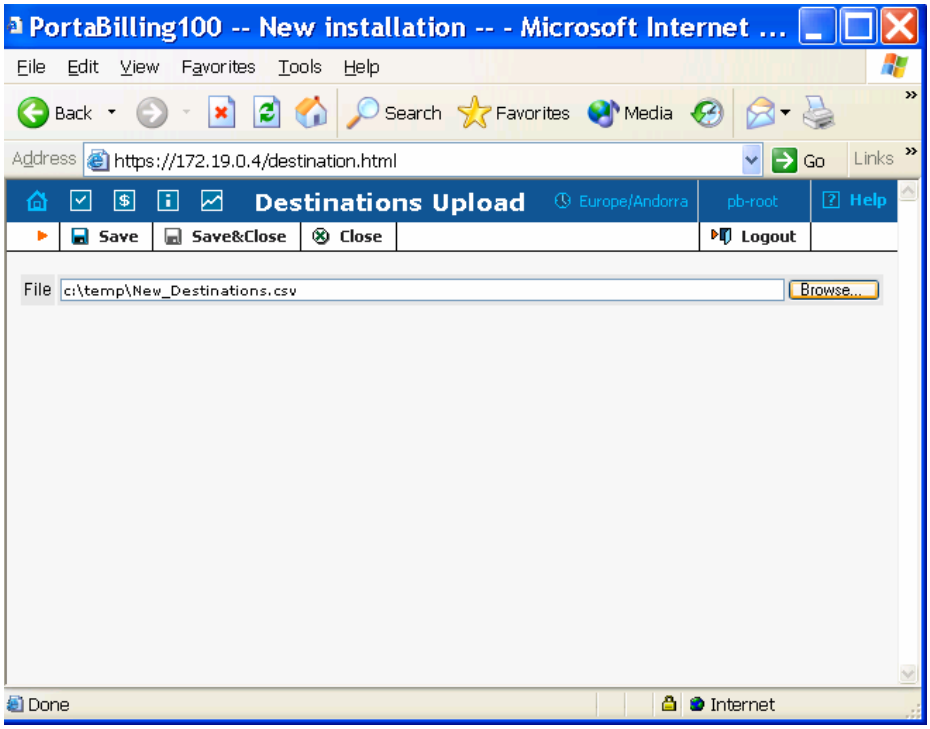


Figure 4-14

Soon we will receive email (similar to the one below) which contains information about the success (or failure) of the import.

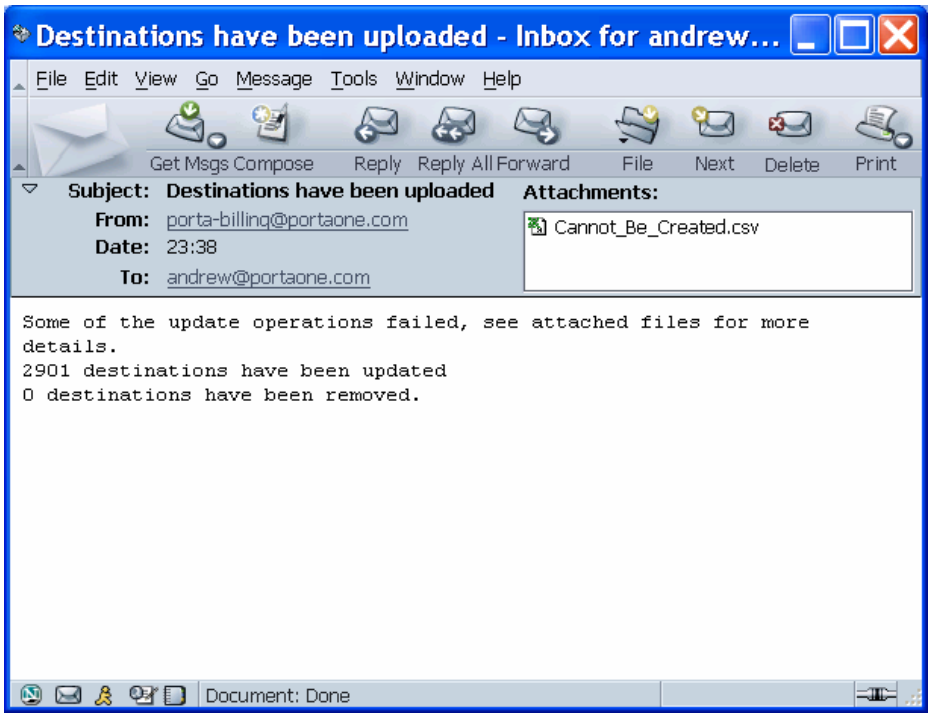


Figure 4-15

As we can see, most of our destinations were uploaded. The ones which system failed to create are in the attached file.

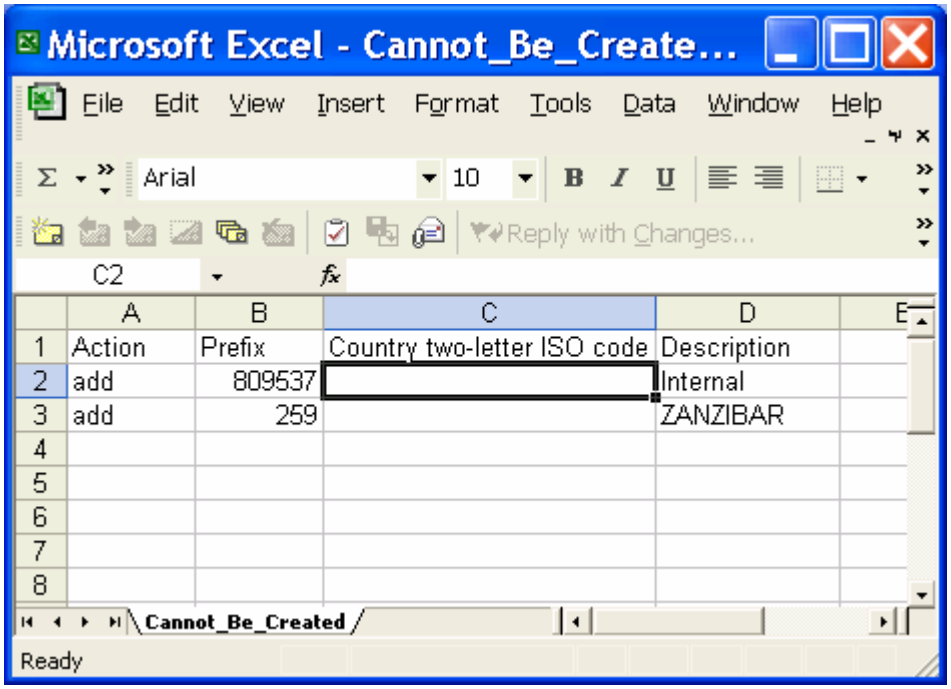
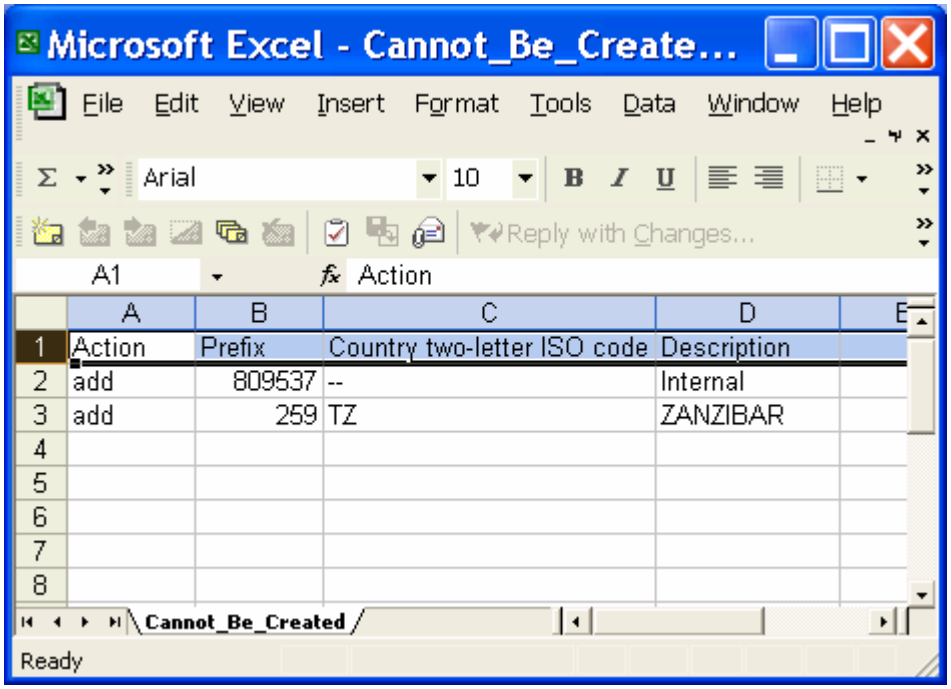


Figure 4-16

These two destinations were rejected, because there was no country code provided, and system was unable to fill it automatically – there is no existing prefix in the system, under which 259 or 809537 would fall under. So we have to fill country code manually, save the file and repeat the upload. File should look like the one on the Figure 4-15. Note, that '—' as the country code has special meaning "Not Available". It can be used for special destinations (for example Inmarsat or your company's internal numbering plan) which do not belong to any country.



	A	B	C	D
1	Action	Prefix	Country two-letter ISO code	Description
2	add	809537	--	Internal
3	add	259	TZ	ZANZIBAR
4				
5				
6				
7				
8				

Figure 4-17

Repeat the destination upload, but upload **Cannot_Be_Created.csv** file this time.

NOTE: Make sure that you perform upload as **supervisor** user, since only this user is allowed to manually override country information. If you are not a supervisor (e.g., an admin), the data you put in the **Country two-letter ISO code** column will be ignored.

After we have imported all of the missing destinations in the system we can upload the file again (**Destination_Unknown**) with the data which was rejected the first time due to missing destinations. This time it should import without any problems.
As a last step, verify that data for your tariff is correct. From the main menu go to **Tariffs** and then click on the (edit rates) icon to the right of the tariff description. After the rates screen appears, try to look up some rates. Type in some prefix, and then click on the **Prefix** button to show all rates for the destinations under this prefix. Or, alternatively, you can type country name and search by the country name (search is case-insensetive).

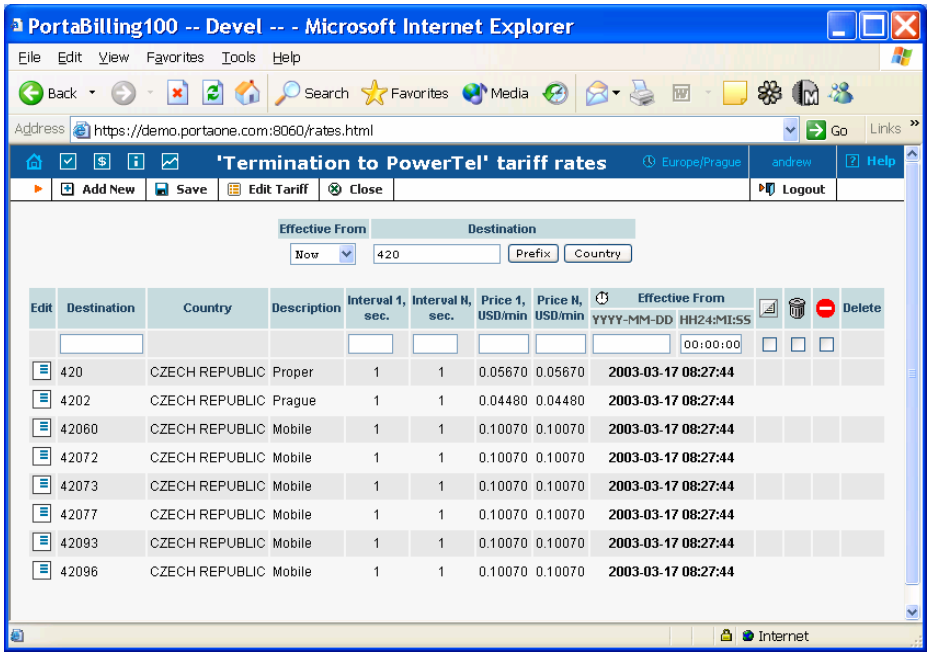


Figure 4-18

When is it impossible to import tariff data?

Despite all the flexibility and power of the PortaBilling template engine it may still be that it is not possible to directly import certain files. Let's take a look on the most typical cases.

File format is not supported by the PortaBilling

It could be that your vendor sends you data in PDF, Microsoft Word or some other format which is not currently supported by the PortaBilling template engine. PortaBilling currently supports

- Comma-separated values files (CSV)
- Microsoft Excel files (XLS)

So try to either ask your vendor to use different format, or do conversion by yourself before importing the data.

There is no phone prefix information in the file

Unfortunately this is very common situation, when your partner will send you a list of rates, but only include country or location name, like in the example below.

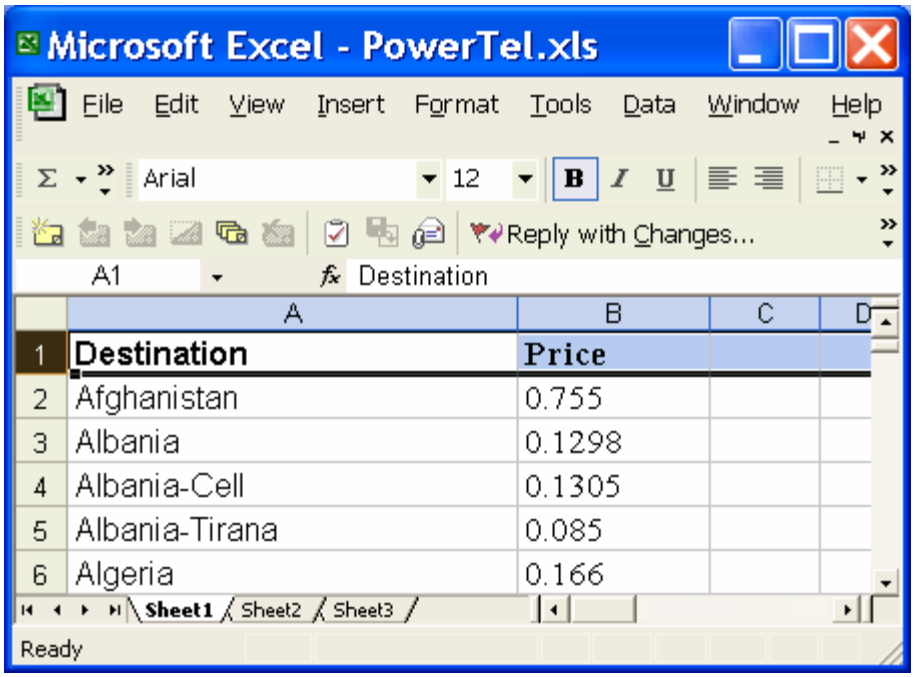


Figure 4-19

It is not feasible to import such data because you can never be sure which destinations were meant to be under the "Albania-Cell" description. Different vendors could use a different numbering for that location. The only solution here is to request from your partner to provide you with the list of prefixes.

There is more than one phone prefix per data line

Sometimes you are provided with data using a special coding for the prefixes. See the example below.

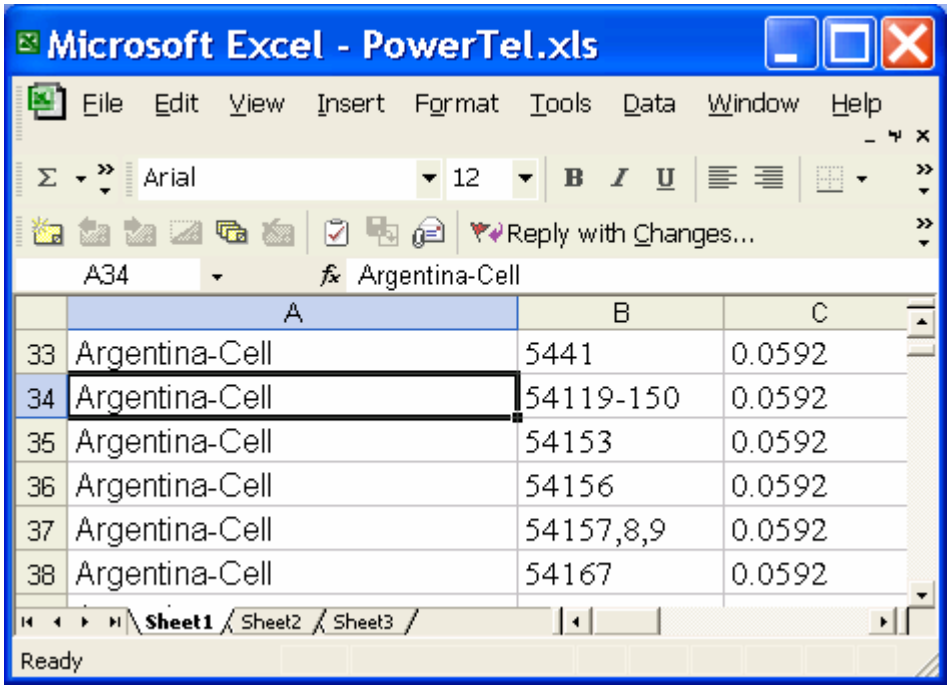


Figure 4-20

This is also not suitable for the uploading. It is necessary to convert this file first (using macros in Excel or some other tool) such that one line contains a single prefix. Cases like row 37 in the file on Figure 4-20 should be replaces with three lines, as shown below.

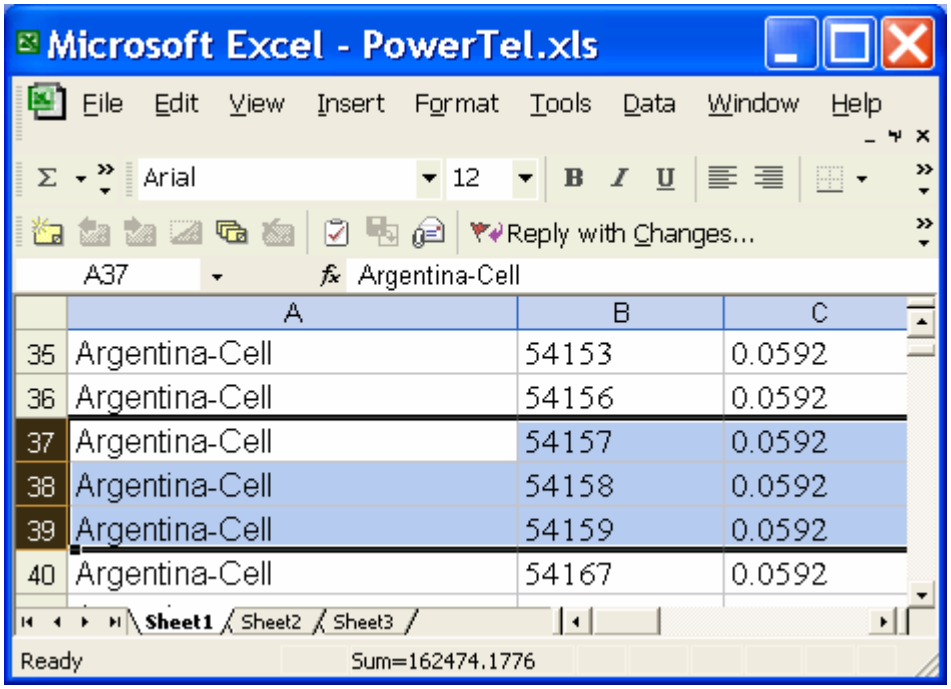


Figure 4-21