PORTA ONE

Porta ✓ Billing®

40
Maintenance Release

Templates Guide

Copyright Notice & Disclaimers

Copyright © 2000-2014 PortaOne, Inc. All rights reserved

PortaBilling® Templates Guide, May 2014
Maintenance Release 40
V1.40.2

Please address your comments and suggestions to: Sales Department, PortaOne, Inc. Suite #408, 2963 Glen Drive, Coquitlam BC V3B 2P7 Canada.

Changes may be made periodically to the information in this publication. Such changes will be incorporated in new editions of the guide. The software described in this document is furnished under a license agreement, and may be used or copied only in accordance with the terms thereof. It is against the law to copy the software on any other medium, except as specifically provided in the license agreement. The licensee may make one copy of the software for backup purposes. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopied, recorded or otherwise, without the prior written permission of PortaOne Inc.

The software license and limited warranty for the accompanying products are set forth in the information packet supplied with the product, and are incorporated herein by this reference. If you cannot locate the software license, contact your PortaOne representative for a copy.

All product names mentioned in this manual are for identification purposes only, and are either trademarks or registered trademarks of their respective owners.

Table of Contents

Preface

The PortaBilling® Templates Guide gives a detailed explanation of the template module. This manual also 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 upon major releases only, and does not always contain the latest material on enhancements that occur in-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/resources/documentation/**

Conventions

This publication uses the following conventions:
- Commands and keywords are given in **boldface**
- Terminal sessions, console screens, or system file names are displayed in `fixed width font`

The **exclamation mark** draws your attention to important information or actions.

NOTE: Notes contain helpful suggestions about or references to materials not contained in this manual.

Timesaver means that you can save time by performing the action described here.

Tips provide information that might help you solve a problem.

1. Template Concepts

Real Life Situation

Before discussing some technical details about PortaBilling® and templates, let's take a look at a "real life" example which will help us learn more.

Situation

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

Solution 1: Manual Entry

In order to do this, you decide to hire a temporary worker to type in all the data. This person does not know much about your business, so you will have to give him or her detailed instructions on how to proceed. These might be similar to the following:

1. This is an Excel file from our vendor.
2. After you open it, in the first row of the second worksheet, entitled "Rates", you will see a title and a list of rates underneath it.
3. The value in column C (entitled "Prefix") goes in the "Destination" field on the web interface, but do not enter **011** at the beginning of the number, if for example, your vendor provides the phone prefix for the Czech Republic as **011420**. The value in column F (entitled "Price per minute") goes in the "Price 1" and "Price N" fields on the web. There is no "Effective From" data in the file, so just enter the current date in that field on the web.

Does this sound complicated? Let's now take a look at what you have just done:

- You have specified the file format which the data is in ("**an Excel file**").
- You have specified where the required data is to be found within that file ("**in the first row of the second worksheet, entitled 'Rates'**").
- And, finally, for each data element which needs to be imported into the system (Destination, Price) you have specified:
  - Which column the data is in ("**column C (entitled 'Prefix')**").
  - The format of the data (whether it needs to be changed somehow prior to being entered into the system) ("**do not enter 011 at the beginning**").

Now, let's take a look at an alternative way of handling the same process more easily, faster, and with fewer errors.
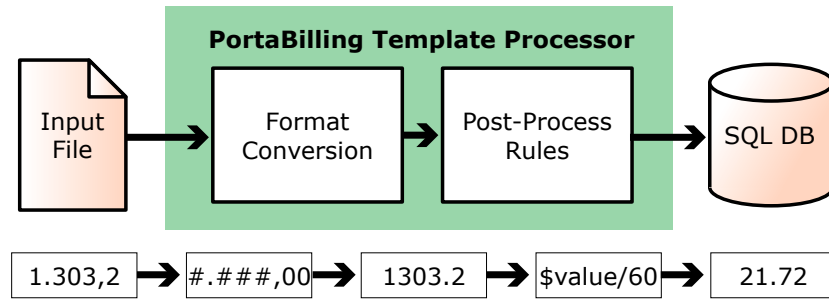
Solution 2: Template Engine

Instead of having your employees type in all the data, you can entrust this job to PortaBilling®! The PortaBilling® rate upload wizard and templates allow you to do the following:

- Upload your XLS or CSV rate file to the server and immediately see the result of file processing (in order to verify that the file was recognized correctly, and to adjust parameters such as the field delimiter).
- Using a drag-and-drop interface, you can place markers on columns in the file to identify where particular groups of data are located in it (e.g. the destination prefix in the third column, and the price in the fifth column). In fact, the rate upload wizard will try to automatically recognize data elements, so most of the time you will only need to confirm the proposed column assignment.
- You can adjust the format which each data element is in (for example, whether the phone prefix is given as **011420** or as **420**). You can also apply post-processing rules (analogous to formulas in spreadsheet editors).
- The upload wizard will process all the rate information and allow you to compare new rates with the existing rates and make adjustments if necessary.
- If there are new destination prefixes in the file (not in the database yet), they can be created automatically during rate upload. Also, you can assign new prefixes into specific destination groups during the upload process.
- Once the new rates are uploaded into the system, PortaBilling® saves all of your changes as a template associated with this tariff. Thus, the next time you receive a new file from the vendor and need to upload it, everything is ready and no special actions are required.

Template Engine

The template engine allows you to define and use data conversion rules (templates) during the rate upload process. There are two methods of template-based data processing and, accordingly, two types of templates.

Import Data into PortaBilling (Upload)

PortaBilling Template Processor

Input File	→	Format Conversion	→	Post-Process Rules	→	SQL DB

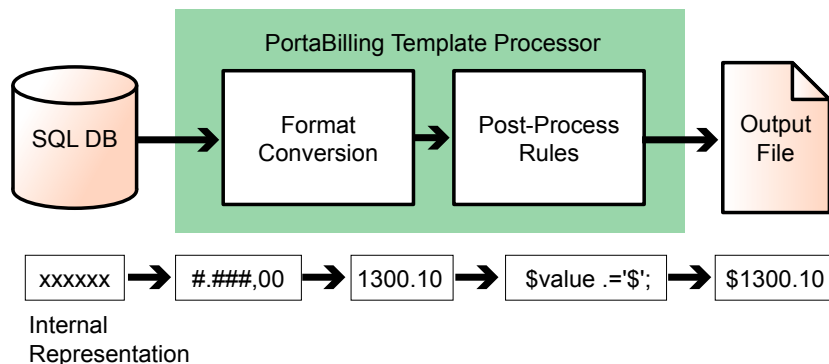| 1.303,2	→	#.###,00	→	1303.2	→	$value/60	→	21.72

Such a template (called an upload template) is defined via a user-friendly rate upload wizard during the first upload of rates from a tariff with a specific format.

The Rate upload wizard provides:
- Rate upload with visual results review at each successive step
- Rate change analysis
- Automatic creation of new destinations
- Prefix assignment to destination groups
- Rate update notifications
- Definition of changes to apply to the data before entering it into the database
- Saving all of the steps performed as a template associated with this tariff that can be used later on during subsequent rate uploads

When rate data is being uploaded into a tariff, the template engine parses the input file according to its type, locates the data inside the file, converts it into an internal representation according to the specification predefined during the first upload, applies post-processing rules and then writes the data to the database.

Export Data from PortaBilling® (Download)

PortaBilling Template Processor

SQL DB	→	Format Conversion	→	Post-Process Rules	→	Output File

xxxxxx	→	#.###,00	→	1300.10	→	$value .='$';	→	$1300.10

Internal Representation

Such a template (called a download template) defines:
- The output data format
- How the data is to be processed before being downloaded into the file
- How the data should be arranged in the output file

When data processing takes place, 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, entering the data in the appropriate places in the file.

Template Components

Data for an invoice or a tariff may have a very complex structure. In order to facilitate such cases, data within the template is split into different groups, called "template components", so that it can be managed separately.
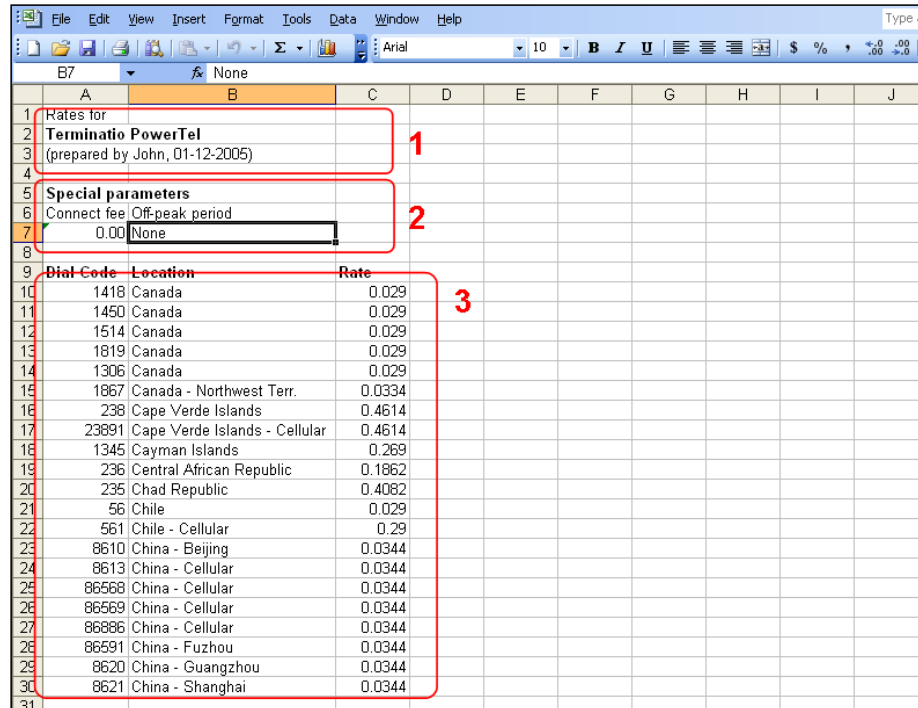
Look at the sample invoice in the picture below:

Page 1 of 2

Porta Software Ltd
902 - 2181 West 38th Ave.
Vancouver
V6M 1R8

Invoice

Date	Invoice #
2004-03-30	22

Bill To:

Retail test Customer
Mr. Vasiliy N Ivanov
Wall str., 22
New-York
USA

Statement Period			
From	To	Terms	Due Date
2004-02-11	2004-02-11	Due on receipt	'Invoices.due_date' not

Description	Amount
Voice Over IP Services	12.17 USD
GST 7%	0.85 USD
PST 7.5%	0.91 USD
Previous balance	0.00 USD

We can clearly identify a few independent groups of data here:
1. **Invoice header** – This is present only once in the file, and contains global information.
2. **Invoice line(s)** – Groups of data which repeat multiple times in the file, each time with different values.
3. **Invoice footer** – This is present only once in the file, and contains global information.

Here is another example, this time for a tariff:

Again, we have several different groups here:
1. **Header** – This is present only once in the file, and contains global information.
2. **Tariff parameters** – This is also present only once in the file, and likewise contains global information.
3. **Rate data** – A data grouping which repeats multiple times in the file, with different values.

Thus there are six possible template components:

Header

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

Page Header

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

NOTE: This component may not be available in *a single page with totals only* invoice templates.

Fields

Similar to the **Header**, this component contains global data for the object to be processed (tariff, invoice, etc.). Data fields are split between the **Header** and **Fields** simply for convenience, with the **Header** containing the main parameters, while **auxiliary** parameters are included in **Fields**. For instance, templates for tariff download contain specialized **Fields:** off-peak period definition, free seconds, connection fee, and so on.
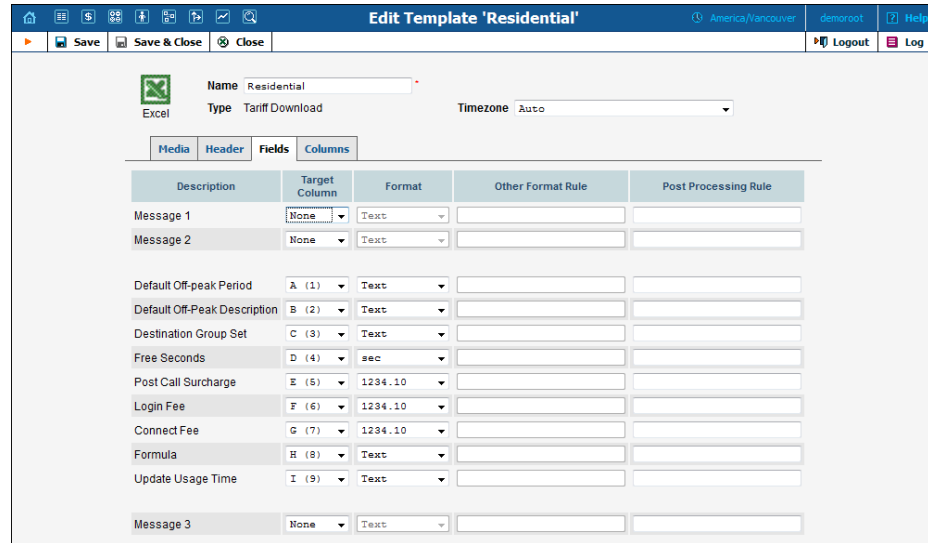
Columns (repeating groups)

This is the most important component, since it describes the variable portion of the 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, and so forth). However, there are many such rows in the file, and each row is an independent object. Thus, you define how one single row of data is to be formatted, and this will then 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 the *Template Groups* section.

Template Parameters

Template Type

The template type defines the intended use of the template:
- Whether it is for generating invoices (based on a standard or a custom invoice template) or for exporting data from PortaBilling®

- What kind of data is involved (invoice data or tariff data, etc.)

Template Media

This defines the file format to be processed, for example .CSV (Comma-Separated Values), .XLS (Microsoft Excel), or a printable form (a specific HTML, which might be converted into a PDF file).

Template Time Zone

Normally, when people specify a point in time, they name the day of the month, the month, the year, the hour and the minute, and, optionally, seconds as well. However, in order to be absolutely precise, this definition should also include a time zone.

Let's assume that your vendor sends you a data file containing his new rates and, for each rate, the date on which it becomes effective. This is fine; but is **01-May-2009 00:00** in the Pacific Standard time zone, or the Central European one? Obviously, time zone misinterpretation can render the rest of the data invalid. In most cases, specification of a time zone is not included in the date itself, but is defined externally.

For example, your vendor might tell you that all of the given times are in Eastern Standard Time. Or, there might be a note in the file saying this. In any case, the template engine needs to know what time zone is to be used. This is specified in the **Template time zone** parameter. If the parameter is set to "Auto", this means that the time zone of the user performing the upload is to be used. This is convenient when working with your own tariffs, but is not usually acceptable when uploading tariffs from vendors.

NOTE: **If the date includes a time zone (for example,** 2009-04-10 12:00:00 Australia/Sydney,**) this time zone will take precedence over the template's time zone. However, make sure in this case that the time zone specification is both supported and unambiguous. For example,** EST **could mean Eastern Standard Time in both the US (GMT -5) and Australia (GMT +10) – quite a big difference!**

Media Parameters

Media parameters define the data layout within the file, where the data for individual components should be located in the file, and whether they are present at all.
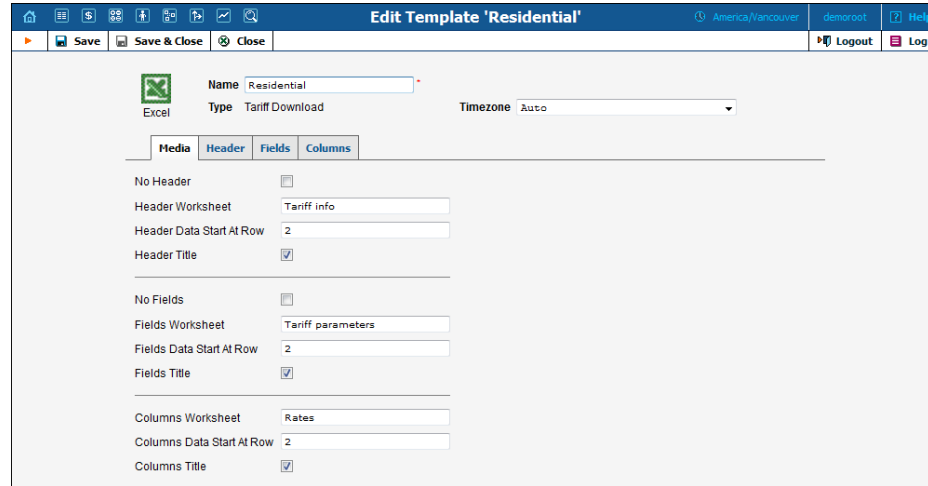
Here is a sample data file that matches this template. Note that "…Data start at row" means the first row with actual data, not including the title. For example, line 7 contains titles for column data, and so 8, the number of the first row with rate data (4066667….), appears in **Column Data Start At Row**.

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 the following:

Target Columns (for Download Templates)

Target columns define where to position the result value. If you want to skip the downloading of specific data, then choose **None** for this field. Multiple data fields can be populated from the same target column.

NOTE: This option is not available for download templates with printable form media, since in that case you will specify where you would like the value to appear using Layout Designer.

Format

A format describes data fields so that they may be converted from the database (default) format to a specific format during download.

For example, if you choose format "**011E.164**" for the "Destination" data field, then the template engine will know that you require the downloaded numbers to be in a US dialing format, so the international number must be prefixed by "**011.**" Therefore, "**011**" will be prefixed to the number "**4201234567**" during the download, so the number in the output file will be "**0114201234567.**"

In order to provide better quality control of data, PortaBilling® supports several data types. These define what kind of data appears in the data field, as well as possible ways to process it. Currently supported data types include:

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

A data type is predefined for each data field, so you do not need to specify one. For each data type in PortaBilling® there is a set of default formats, so that you will usually 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". You will then be able to specify a custom format rule.

"Other" Format Rule

If none of the PortaBilling® default formats suit your needs, you can choose "Other" and enter your own formatting rule. See below for a description of how a format rule works for data fields of a particular type.

Number

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

- A **#** sign or the digits 0-9 represent one digit of the input or output number.
- The first character from the *right* which is not a digit or # (e.g. . in #,###.00) specifies a decimal separator.
- The first character from the *left* which is not a digit or # (e.g. , in #,###.00) specifies a group separator.
- The number of characters after the decimal separator specifies the precision of the number. However, # indicates no zero padding, and 0 indicates zero padding.
- If the number has too many digits for the format specification, it will be rounded to the number of decimal digits specified 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 is specified, then no group separation will be done.
- It is sufficient to specify a group separator only once, even if your real numbers will be large and contain more than one digit group. All digits to the left of the decimal point are always shown, regardless of how many of them are actually shown in the format.

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

Date and Time

This data type is used for fields containing either both date and time (e.g. **Effective from**) or just the date (e.g. **Invoice date**). Please note that when importing dates from an unusual format you can use the Time::ParseDate module to take care of all the dirty work. This module is able to examine a data string and find bits of data regardless of how they are placed or separated there. In order to use the Time::ParseDate module for parsing dates, choose "Auto" as the format.

Unfortunately, in some situations even Time::ParseDate is unable to properly identify the date. For example, the string "02-01-2009" could either mean February 1st or January 2nd. In such situations, an exact format specification is required. The "Other" format uses a syntax very similar to the one used in applications like Microsoft Excel. Its elements are the following:

- **YYYY** – Year (four digits, e.g. 2009).
- **YY** – Year (short format, e.g. 05). Note that this format can lead to potential data interpretation problems, since "10" could be either 2010 or 1910. The PortaBilling® template engine considers the 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 a leading zero for 1 through 9; for example, 01, 06, 14, 25.
- **D** – Day of the month without a leading zero for 1 through 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 clearly not work correctly.

- **SS** – Seconds.
- **TZ** – Time zone name (e.g. CET).
- **XXX** – Any combination of non-space characters; used in formats for parsing dates to skip a variable part of the date, for example, the name of a weekday (see examples below in Table 1-2).
- All other symbols in the format (spaces, delimiters like **/**, **-**, **:** or **;** and other characters) represent themselves. So they are placed as-is in the output date string, and must be found in the input string exactly as they were entered.

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

Date/time string	Description	Correct format specification
07/04/2009	July 4th 2009, US date format	MM/DD/YYYY
04-07-2009	July 4th 2009, European date format	DD-MM-YYYY
07-04-09	July 4th 2009, 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
2009-07-04 14:00:00	Date and time, ISO format	YYYY-MM-DD HH:MI:SS
07/04/09 2:00:00 pm	Date and time, US format	MM/DD/YY HH12:MI:SS AM/PM
1-May-2009 18:52	Date and time, abbreviated month name	D-MON-YYYY HH:MI
Sun, 16 Mar 09	Date and time, skip weekday name	XXX D MON YY
12th of March, 2009	Date with "as is" elements	Dth of MONTH, YYYY

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

The following table gives examples for using templates to convert values in the PortaBilling® database to the desired format.

Date/time in	Format specification	Resulting value
01-May-2014 12:34:56	MM/DD/YYYY	05/01/2014
01-May-2014 12:34:56	DD-MM-YYYY	01-05-2014
01-May-2014 12:34:56	MM-DD-YY	05-01-14
01-May-2014 12:34:56	HH12:MI AM/PM	12:34 pm
01-May-2014 12:34:56	HH:MI	12:34
01-May-2014 12:34:56	HH:MI:SS	12:34:56
01-May-2014 12:34:56	YYYY-MM-DD HH:MI:SS	2014-05-01 12:34:56
01-May-2014 12:34:56	MM/DD/YY HH12:MI:SS AM/PM	05/01/14 12:34:56 pm
01-May-2014 12:34:56	D-MON-YY	1-May-14
01-May-2014 12:34:56	D of MONTH, YYYY	1 of May, 2014
01-May-2014 12:34:56	YYYYMMDD HHMISS	20140501 123456

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

Other data types

For other data types (e.g. **Destination**, **Text** or **Interval**) there are no specific format elements. The "Other" format rule is regarded as a Perl macro; thus it is evaluated, and the result of the evaluation is considered a new value. For a detailed description of Perl macros, see the Post-processing rules section below. The original string value is received as a parameter, and the converted value should be returned in the required format.

The following table provides a description of which type of output value is required for other format rules, depending on the data field type.

Data field type	Other format rule in	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 other format rules.

Post-processing Rules

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

- A processing rule is Perl code, i.e. one or more Perl operations separated by semicolons.
- The 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 post-processing rules can do is to create new data which is not based on the input data. This is required, for example, when certain data fields (e.g. **Interval 1** and **Interval N**) are absent from the input file. If you know that **Interval 1** is 30 seconds, and **Interval N** is 6 seconds, you can simply assign these values as constants in the post-processing rule.

Let's look at a few examples:

Post-processing rule	Input data	Result
Remove all leading white space from the string		
`$value =~ s/^\s+//;$value`	Aruba	Aruba
Remove all leading and trailing white space from the string		
`$value =~ s/^\s+//;` `$value =~ s/\s+$//;$value`	Aruba	Aruba
Add 1 before the phone number, but only if it does not start with 0		
`$value =~ m/^0/ ? $value` `: '1'.$value;`	6041234567	16041234567
Take only the part of the input string following 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/\"//; $value`	Germany-'Berlin'	Germany Berlin
Use an external (custom) module to translate English country names into Czech		
`use Czech::Translate; transl_data($value);`	Austria	Rakousko

Table 1-5 – Examples of using post-processing rules.

NOTE: **This manual does not cover details of 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 a post-processing rule should have the same type as the input value it receives. The following table provides information about types of 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 post-processing rules.

Template Groups

Concepts

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

```
Page 2 of 2
                        Call Detals Records By Country
CANADA

3804424471821  160452152771   CANADA   Quebec     2004-02-12 08:08:25    38    0.10
3804424471825  160452152771   CANADA   Alberta    2004-02-12 08:09:07    3     0.49
3804424471825  160421971327   CANADA   British    2004-02-12 08:09:13    31    0.26
3804424471825  160421971325   CANADA   Ontario    2004-02-12 08:09:54    3     0.04
TOTAL By CANADA                                                          2     0.88
CZECH REPUBLIC

3804424471821  420608076502   CZECH    Mobile     2004-02-12 07:23:07    14    0.75
TOTAL By CZECH REPUBLIC                                                  0     0.75
UKRAINE

3804424471825  380688763541   UKRAINE  Mobile      2004-02-12 06:20:29   66    0.36
3804424471825  380442447182   UKRAINE  Kiev Region 2004-02-12 07:09:35   1     0.95
3804424471825  380503425341   UKRAINE  Mobile      2004-02-12 07:10:16   20    0.24
3804424471825  380664729846   UKRAINE  Mobile      2004-02-12 07:14:25   2     0.13
3804424471825  380672345123   UKRAINE  Mobile      2004-02-12 07:15:07   21    0.15
TOTAL By UKRAINE                                                         2     1.83
```

Group

Each group is a portion of data which has an identical value for some parameter or parameters. All of the input data will be sorted according to groups, and rows within the same group will be presented together. In the example above, three groups are visible, each of them containing all calls made to a certain country. Groups can also be nested; for example, the group "By Country" will contain several sub-groups "By Prefix". These are sorted in the list of groups according to their include order, so that the outermost group is on top, and the innermost at the bottom.

Group Break

This is a parameter which defines which group the data should belong to. All of the available data are sorted according to this field. In the example above, the list of all calls is sorted by the name of the destination country (so that calls made to **Slovenia** appear before calls made to **Ukraine**). Next, rows which have the same group break expression value are 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 for breaking the group (in the example above, this is the "Country" column). You can also create a custom group break, specifying any valid SQL expression.

Group Header

This is a template component which is inserted into the document immediately before the group data. Besides static text, variable fields (e.g.

"Destination") can also be used. In the example above, this is the row containing the name of the country.

Group Footer

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

Group Fields

These variable elements are very similar to data fields in the **Columns** section of a template (and are actually based on them). You can alter their parameters (such as format or post-processing rules) in exactly the same way as you would for standard elements of the **Columns** section.

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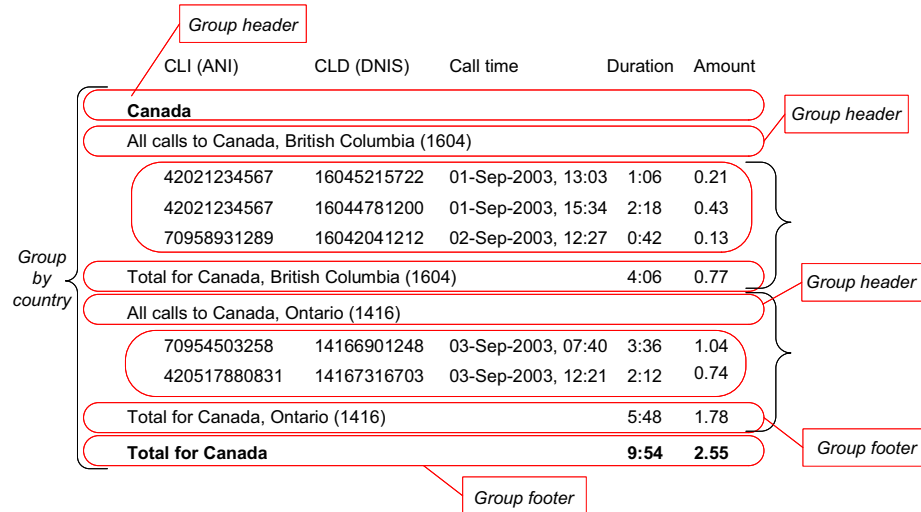
Since one group may consist of many rows of data, what would be the value in the group fields for the header/footer? The fields in the group header have the value of the first row in the group. You can specify how you would like to populate the value of fields in the group footer. Possible pre-defined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

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

You can always write your own custom total rules. These are very similar to post-processing rules, in that you can specify any Perl expression. The current value of the column is given in the **$value** variable, the current value of the total is given in the **$total** variable, and the result returned by the expression becomes the new total.

In the example below you see what a list of calls might look like if you were to use two groups:

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

Group header

CLI (ANI)	CLD (DNIS)	Call time	Duration	Amount

Canada   *Group header*

All calls to Canada, British Columbia (1604)

42021234567	16045215722	01-Sep-2003, 13:03	1:06	0.21
42021234567	16044781200	01-Sep-2003, 15:34	2:18	0.43
70958931289	16042041212	02-Sep-2003, 12:27	0:42	0.13

Group by country

Total for Canada, British Columbia (1604)   4:06   0.77   *Group header*

All calls to Canada, Ontario (1416)

| 70954503258 | 14166901248 | 03-Sep-2003, 07:40 | 3:36 | 1.04 |
| 420517880831 | 14167316703 | 03-Sep-2003, 12:21 | 2:12 | 0.74 |

Total for Canada, Ontario (1416)   5:48   1.78

Total for Canada   **9:54   2.55**   *Group footer*

Group footer

Things to Remember

XDR Number Limitation in the Invoice

In order to avoid high resource consumption while generating vast invoices, the number of rows in the invoice is limited to 1000 by default, but it can be changed via the configuration server.

Custom Fields

Sometimes service providers must supply customers with additional information (for example, tax code, contract number, etc.) on the invoices. In PortaBilling®, additional information can be filled in on the **Custom Fields** tab when you add a new customer or edit an existing one. Later this information can be added in the **Layout Designer** of the invoice template.

If you need to add extra fields to the invoice perform the following steps:

- Add necessary custom fields on the **Web Interface** page.
- Ensure that all required information is specified on the **Custom Fields** tab for the customers who require additional information on their invoices.
- Open the invoice template where you would like to add custom fields.
- Launch **Layout Designer**.
- Right-click on the layout designer sheet and select the desired custom field from the available list in the **Header** section.

The Suppress Columns Option

Suppress Columns allows to show in the invoice only totals per each, service such as total payments, total credits and total calls' duration and cost to specific countries.

In the figure below you can see an invoice example with the **Suppress Columns** option disabled.

12004577824	Payment/Refund transaction	E-Commerce Charge		21-02-00		-4.10
12004577824	Payment/Refund transaction	E-Commerce Charge		25-02-00		-2.15
TOTAL Payments						**-24.08**
Credits						
	Balance adjustment	Manual charge		09-02-00		20.50
	Balance adjustment	Manual charge		11-02-00		4.50
	Balance adjustment - Manual	Manual refund		12-02-00		-10.00
TOTAL Credits						**15.00**
Subscriptions						
12001234507	Subscription Fee	Follow Me	00-02-00	13-02-00		3.00
12001234507	Subscription Fee	SuperCall	01-02-00	01-03-00		7.99
TOTAL Subscriptions						**10.99**
CANADA						
12001234507	100438130379	CANADA	Newfoundland	2000-02-12 10:04:54	0:28	0.29
12001234507	88005554433	CANADA	Quebec	2000-02-12 10:08:25	1:10	0.70
12001234507	100471374925	CANADA	Quebec	2000-02-12 10:08:25	0:38	0.10
12004577824	100452152771	CANADA	Alberta	2000-02-12 10:09:07	0:38	58.48
12004577824	100421971327	CANADA	British	2000-02-12 10:09:13	0:31	0.20
12004577824	100421971325	CANADA	Ontario	2000-02-12 10:09:54	0:31	0.04
12001234507	8-800-SMS	CANADA	Quebec	2000-02-13 12:12:01	0:14	0.34
TOTAL By CANADA					**4:10**	**60.20**

In the figure below you can see an invoice example with the **Suppress Columns** option enabled.

Payments		
TOTAL Payments		-24.08
Credits		
TOTAL Credits		15.00
Subscriptions		
TOTAL Subscriptions		10.99
CANADA		
TOTAL By CANADA	4:10	60.20

Destination Upload

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

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

- The format is fairly simple.
- You almost never need to upload new destinations alone; normally, you will need to input new rates in the system which requires the new destinations, and in this case the PortaBilling® template processor will add new destinations automatically.

However, functionality for creating custom templates for destination upload or download is available, and may be included in future releases.

How to fill in country information for a destination

For each destination entered into PortaBilling®, it is necessary to provide information on which country (and, optionally, administrative subdivision in that country) the destination belongs to. This will help you keep better track of rates for a specific country, and will also be used to ensure a correct description in call details and statistics for you and your customers. PortaBilling® comes with a complete list of countries registered with the International Standards Organization, providing each country's ISO code (e.g. **US** for United States of America, **CA** for the Canada, and so on). When a new destination is manually entered using the PortaBilling® web interface, you can choose the country from a list. When importing many new destinations at once, country information must be provided for each in the form of the two-letter ISO code. Why not the country name? Unfortunately, country names can be very ambiguous. For example, *UK*, *United Kingdom*, *Great Britain*, *Britain* and *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 never know which version of the name your vendor will decide to use. Moreover, country names in local languages (e.g. *Velka Britanie* in Czech), spelling errors (e.g. *Untied Kigndom*), punctuation marks, and the like further complicate the matter. This is why PortaBilling® uses country codes instead, since they define exactly which country is meant.

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

TIP: When uploading destinations with empty country codes, PortaBilling® will try to find country information based on data which is already in the database. For example, if the prefix **61** already exists in the database, and is assigned to Australia, then when you upload a new

destination with prefix **61881** PortaBilling® will automatically fill in the country information for you.

NOTE: Of course, in order for this method to work effectively you need to have an initial set of destinations in the database already. PortaBilling® comes with a default set of almost 3,000 destinations, which covers most of the countries of the world and popular destinations in those countries. You can download it using the **Get default set** button in the Destinations screen, customize it according to your needs, and then upload it.

2. Rate Upload

Upload via the Rate Upload Wizard
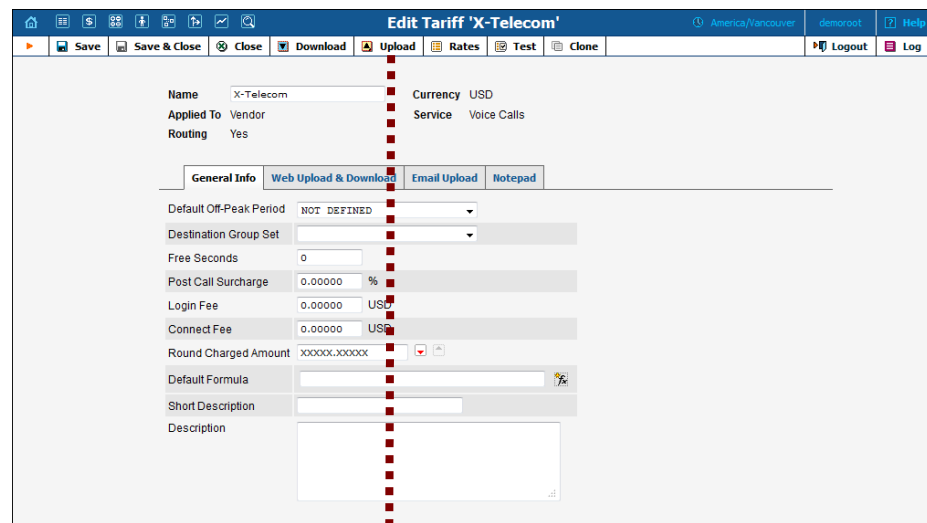
PortaBilling® offers the ability to program data processing rules (called templates) for each type of rate file you receive, so that you can eliminate the need for any manual data conversion and save yourself a lot of time. Also, the rate upload process is interactive and provides a comparison between new rates and existing ones, so that you can review the rates being uploaded for validity and accuracy.

Let's have a look at how a rate import is performed. We will perform the rate upload for the "X-Telecom" termination tariff, proceeding step by step. You can then use this as a model for uploading any other tariffs.

NOTE: In PortaBilling®, the proper name for the initial portion of the phone number (used to determine which location or type of phone network this number belongs to) is **destination** (e.g. 44 for the United Kingdom and 442 for London, United Kingdom). In the VoIP business the same entity is frequently called **prefix** – so we will use these terms interchangeably.

Start the Import

1. On the Tariff Management page, click on the tariff name.
2. In the Edit Tariff window, click the ![icon] **Upload** button. You will see a screen like the one below:

Let's now discuss the available options and their effect.

Add or replace rates

This radio button controls behavior in respect to rates which were already defined in the tariff at the time of import. The default is to **add** new rates into the tariff. So when the new rates appear, and there is an existing rate for a certain prefix, the new rate will simply override the old one. If a rate for a certain prefix is in the tariff but is not given in the file you received from the carrier, it will stay unchanged. This mode is frequently used, since most of the time your partners will send you rate updates only, i.e. the file will contain only a small subset of prefixes for which the price has changed.

In some cases, however, you need to ensure following import that the tariff structure matches the provided file exactly, so that there are no "old" rates remaining. This is normally done when your carrier changes the structure of the prefixes and sends you a "replacement" rate update. In this case, you will want to use **replace** mode, which uploads all the new rates and then marks rates for all prefixes not present in the uploaded file as discontinued.

NOTE: To prevent a situation where some rates would be discontinued erroneously because a new rate was not uploaded due to a data error (e.g. a typo in the price column), discontinuation of rates will take place only if there were no errors in uploading the new rates.

In our example, we assume that a normal, "incremental" rate import is being used; thus we will be **adding** rates.

Effective from

PortaBilling® allows each rate to be assigned an "effective from" date, so that an automated rate change can be easily scheduled for the future. Thus

if your carrier sends you a file which indicates the date when a specific rate becomes effective, this can be extracted from the file and added to the rate information in PortaBilling®. Sometimes it may happen, though, that you receive the file later than expected, when the moment at which the rate was supposed to become effective has already passed. In order to protect data integrity, it is impossible to upload the rate and assign an "effective from" date in the past, since CDRs for calls previously made will have already been rated according to the old rate and stored in the database (see the xDR rerating topic for more information about how this situation can be resolved). By default, a rate entry effective from a date in the past will be rejected and not inserted into the tariff. Alternatively, you may choose to insert these rates into the tariff, but effective from the current moment; to do so, please tick the appropriate check-box.

Skip duplicate rates

Sometimes carriers will include all prefixes in a rate update file, even if for some of them no change in pricing has occurred. If you upload the whole rate file, it will create duplicate rate rows. This has no effect on the way calls are billed, but it will increase the amount of entries in the rate table and, in the long run (if it occurs continually), have a negative effect on performance.

In order to prevent this, simply activate the option to skip duplicate rates. This will slightly increase the time it takes to upload the whole rate file (since PortaBilling® will have to check the already existing rates and compare them with the new ones). If rate parameters such as price, preference, and so on turn out to be identical, no extra (excess) rate will be inserted. We use this mode in our example (and certainly recommend using it whenever possible).

Finally, choose the rate file and click **Next**. The file is uploaded to the server, and we move on to the next step.

Review File Parsing

The purpose of this screen is to verify that the general structure of the file has been determined correctly, so that the data can be properly extracted. There are different ways to organize data files: for instance, CSV files, despite their name (comma-separated values) can use semi-colons or other characters as a separator. PortaBilling® attempts to make a best guess, but it is important to review the results and, if necessary, change the settings according to your preference.
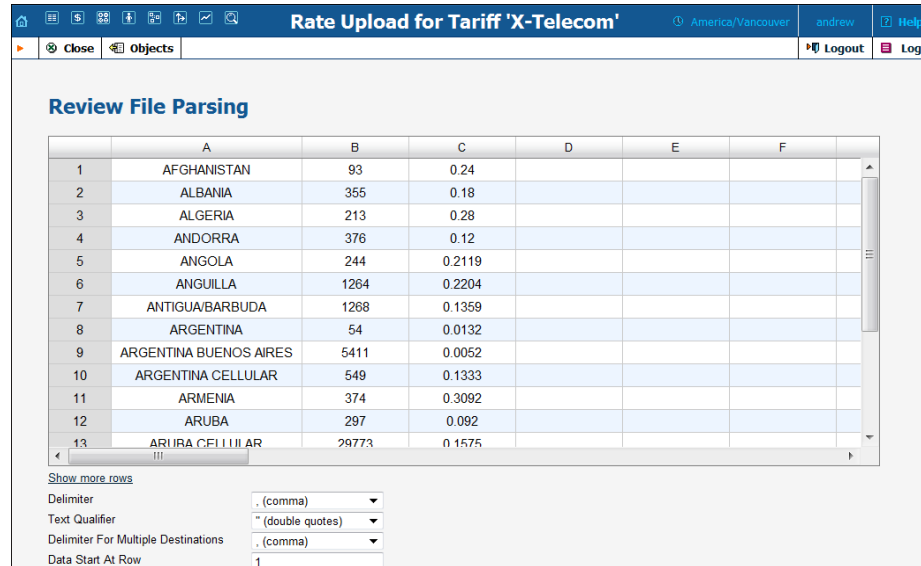
Multiple prefixes in the same cell

Typically one row in the table contains the rate information for a single phone prefix (destination) such as 447 or 447527. Sometimes you may receive a file from the vendor where multiple prefixes are placed in the same table cell as shown in the screenshot below.
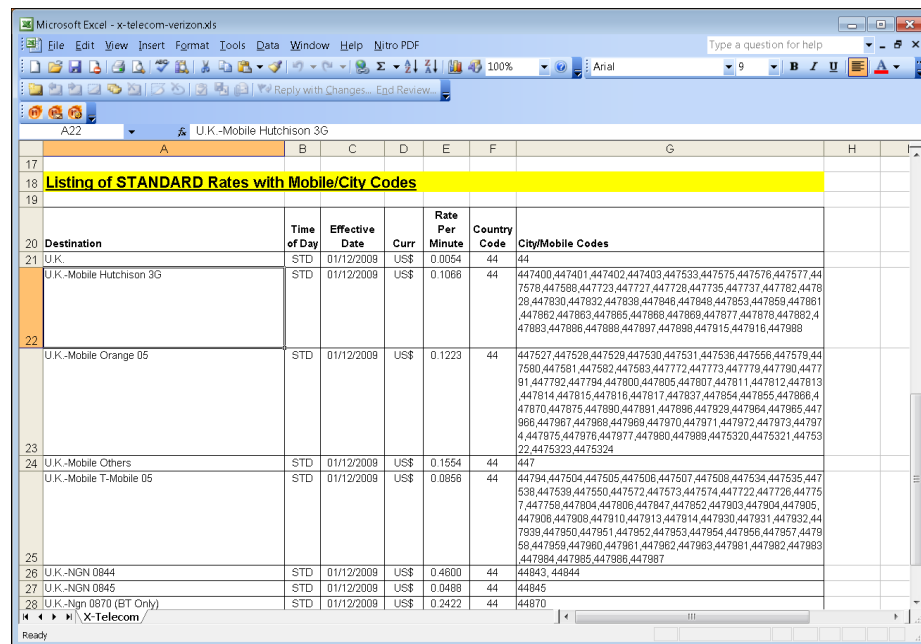
You can easily process such files – you just need to specify which character (e.g., comma (,), semicolon (;) or something else) will be used to separate the individual prefixes in the **Delimiter For Multiple Destinations** field. This way, during the upload, the system will treat each occurrence as if there is a separate row in the table for each prefix.
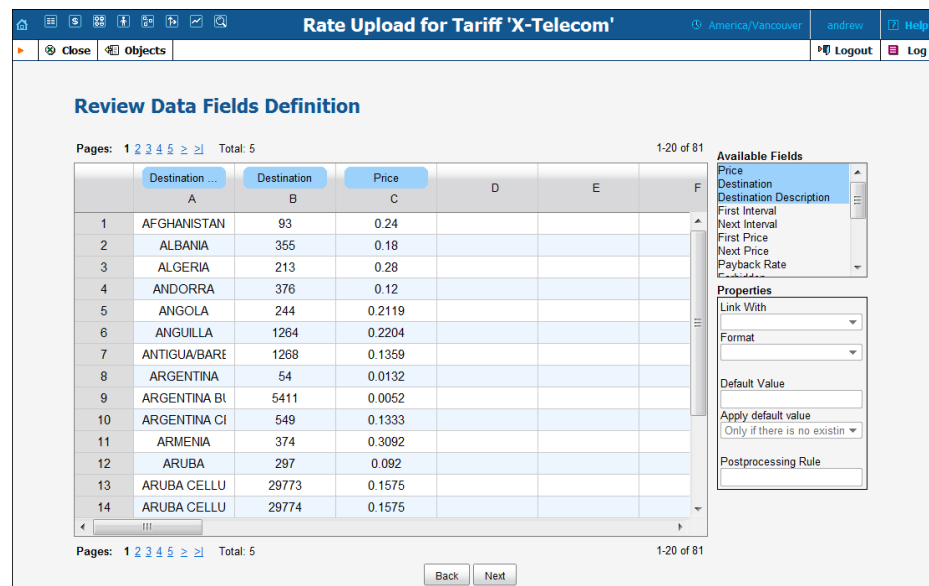
In our case, it looks as if the file was parsed properly, so we can click **Next** and move on to the next step

Review Data Fields Definition

This is a very important step, since we need to identify where individual data elements (e.g. destination prefix or price) are located in the file, and map them to the attributes (data fields) for the rate. You can imagine what would happen if the system were to confuse them.

PortaBilling® applies some artificial intelligence, and usually guesses quite well where the specific columns are; but, obviously, you need to check whether the data fields have been assigned to the proper columns, and adjust them if necessary.

On the right you see the **Available Fields** select menu, which represents all the attributes of the rate record. You can drag and drop fields from here to the table headers in order to associate a particular data field with a column. A blue label will then remain at the top of the column, signaling that the values in this column (e.g. column A) will be used for the specified rate data field (e.g. **Destination**). Also, the data field in the list will be highlighted – this allows you to quickly see which data fields are already assigned.

You can also drag a label to a different column (thus associating the data field with a different column) or back to the **Available Fields** menu (the data field will then not receive any input values from the file).

The **Format** select menu allows you to adjust the way PortaBilling® interprets the values in the table cells, e.g. whether a leading 00 should be stripped off or considered part of the phone prefix, or whether a comma (,) or a dot (.) should be considered as a decimal separator for numbers.

Finally, you have the post-processing rule. Here you can write a mini-program in Perl which will alter the original value, e.g. append or remove characters, increase or decrease numeric values. It can also perform advanced operations, e.g. look up the current exchange rate for the vendor's currency and convert the value into the currency you use.

What if the file does not contain a value for a certain data field?
This is another common situation: the carrier may not have included some important information (e.g. rounding intervals) in the file, or it could be that you simply need to assign the value based on your own decision (e.g. when uploading rates from carrier ABC, you will mark them as the Cheap route category). In this case, you may leave the data field unassigned to any column in the file, and simply type a value in the **Default Value** field.
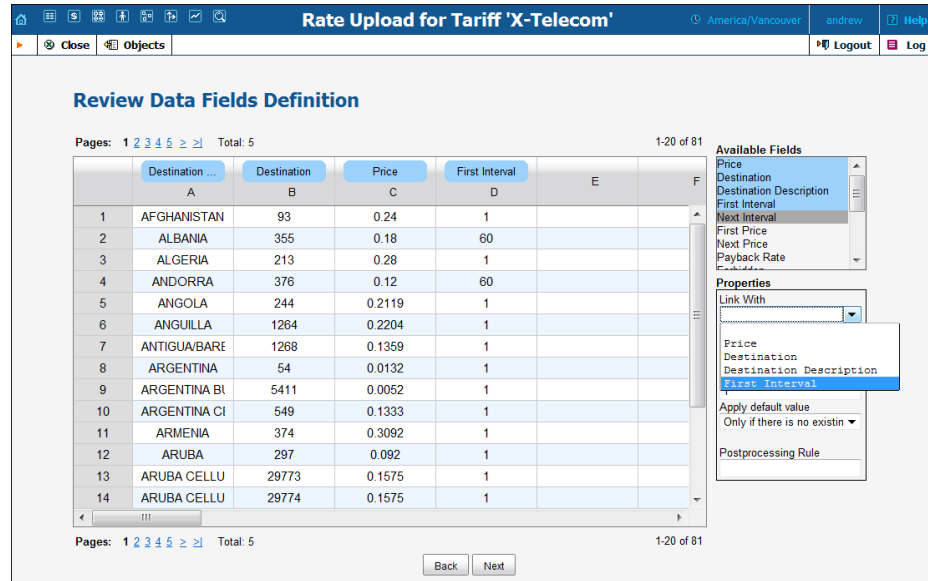
		Destination ...	Destination	Price	D	E	F
		A	B	C			
	1	AFGHANISTAN	93	0.24			
	2	ALBANIA	355	0.18			
	3	ALGERIA	213	0.28			
	4	ANDORRA	376	0.12			
	5	ANGOLA	244	0.2119			
	6	ANGUILLA	1264	0.2204			
	7	ANTIGUA/BARB	1268	0.1359			
	8	ARGENTINA	54	0.0132			
	9	ARGENTINA BI	5411	0.0052			
	10	ARGENTINA CI	549	0.1333			
	11	ARMENIA	374	0.3092			
	12	ARUBA	297	0.092			
	13	ARUBA CELLU	29773	0.1575			
	14	ARUBA CELLU	29774	0.1575			

Rate Upload for Tariff 'X-Telecom'

Review Data Fields Definition

Pages: 1 2 3 4 5 > >| Total: 5   1-20 of 81

Available Fields
Destination Description
First Interval
Next Interval
First Price
Next Price
Payback Rate
Forbidden
Hidden

Properties
Link With

Format
sec

Default Value
60

Apply default value
Only if there is no existin

Postprocessing Rule

Pages: 1 2 3 4 5 > >| Total: 5   1-20 of 81

What if you need to associate a single column with more than one data field?
For instance, the carrier only supports a single "rounding interval" rating parameter, and so they provide a single column only (column D, in our example). In PortaBilling®, however, you have **First Interval** and **Next Interval**. In this case, you will associate one data field with the column, and then click on the other data field and link it to the first one (select the first field name in the **Link With** menu).

What if you want to "preserve" some of the already assigned rate attributes?

Quite often you will initially upload all the rates from a given carrier with a default preference (e.g. 5) or route category (e.g. Default). After that, you start to fine-tune your routing based on quality reports, testing, customer feedback, and so on. Let's assume that, in this particular vendor's tariff, the preference for prefix 44 was changed to 8 (to reflect the fact that this is our preferred vendor for this destination) and the preference for prefix 447 was changed to 3 (here we actually decided to move the vendor to the bottom of the routing list for this particular destination). Now, if we were to simply upload the rate file, it would assign the default value (5) for preference to all the new rates, and our subsequent routing adjustments will be lost.

To prevent this, you may specify that the default value should only be applied if there are no existing rates for this prefix in the tariff.
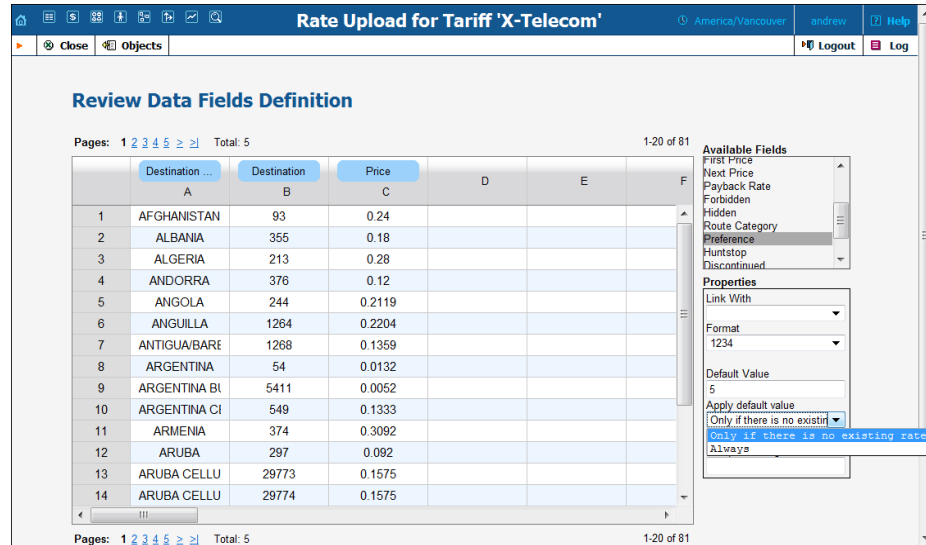
Thus the parameter value from the existing rate will be copied into the new one, so that the new rate for 44 will retain preference 8, and the rate for 447 – preference 3.

Tip: In PortaBilling® you configure the rate information using two parameters: the price for the first interval (**First Price**) and the price for subsequent intervals (**Next Price**). Most of the wholesale carriers do the rating using only one parameter: price per minute. So when importing such a file into PortaBilling® you would need to assign the **First Price** data field to the column that contains the price and then link the **Next Price** to it. Since it is a very common operation, for your convenience the rate import wizard contains a special data field: **Price**. If this "virtual" data field is assigned to a column, then the value found there will be used for both **First Price** and **Next Price** rate parameters.

Ready to move forward?

In order to proceed, you must assign the **Destination** and **Price** data fields to columns in the file.

The **Destination Description** field is not required for the actual rate import, but it will be very useful later on if there are any new destinations created.

Tip: In order for the destination description to be *automatically* filled in it is needed to create a "**Destination Description**" column and fill in some descriptive information in the file before the upload procedure.

Click **Next** to proceed.

Review Rate Information

The rate import wizard will process the rate data according to the rules previously specified and present the results on the screen for review before storing them in the database.

The **Status** column shows the comparison between the current rate and the new rate in the PortaBilling® file so you can quickly understand exactly how the rate will change. A red arrow indicates a price increase (this destination is more expensive) and a green arrow indicates a price decrease (making it cheaper to call that destination). In **Price** field you can also view the actual price differences.

You can edit the rate data right here (for instance, you negotiated a special last-minute rate change with the carrier or you wish to change route category) – just double-click on the specific cell in the table and enter a new value.

	Destination	Country Description	Status	Price		Off-Peak Price		Interval		Off-Peak Interval	
				First	Next	First	Next	First	Next	First	Nex
1	447	UNITED KINGDOM Mobile	↑	**0.26300(+0.01000)** _0.25300_	0.25300	0.25300	0.25300	1	1	1	
2	4470020	UNITED KINGDOM UNITED KINGDOM		**0.35300**	0.25300	0.25300	0.25300	30	6	1	
3	4470431	UNITED KINGDOM UNITED KINGDOM		0.46000	0.46000	0.46000	0.46000	1	1	1	
4	447060	UNITED KINGDOM UNITED KINGDOM		0.25300	0.25300	0.25300	0.25300	1	1	1	
5	4470740	UNITED KINGDOM UNITED KINGDOM		**0.25200**	0.25300	0.25300	0.25300	30	6	1	
6	4470775	UNITED KINGDOM UNITED KINGDOM		0.46000	0.46000	0.46000	0.46000	30	6	1	
7	4471	UNITED KINGDOM UNITED KINGDOM		0.41029	0.41029	0.41029	0.41029	30	6	1	
8	4472	UNITED KINGDOM UNITED KINGDOM		0.41029	0.41029	0.41029	0.41029	30	6	1	

As you can see in the screenshot, for the 447 destination, the price per minute has increased by $0.01. Also, the value of the **Preference** parameter of this rate is 3 (while most other rates have preference 5) – this is because there is an existing rate for this destination, and it is already assigned preference 3.

Click **Next** in order to proceed.

Managing New Prefixes

This screen only appears if the rate file contains any prefixes you haven't yet defined in your system and it allows you to add new prefixes quickly and efficiently. This process takes a very little time and reduces the chances for error.
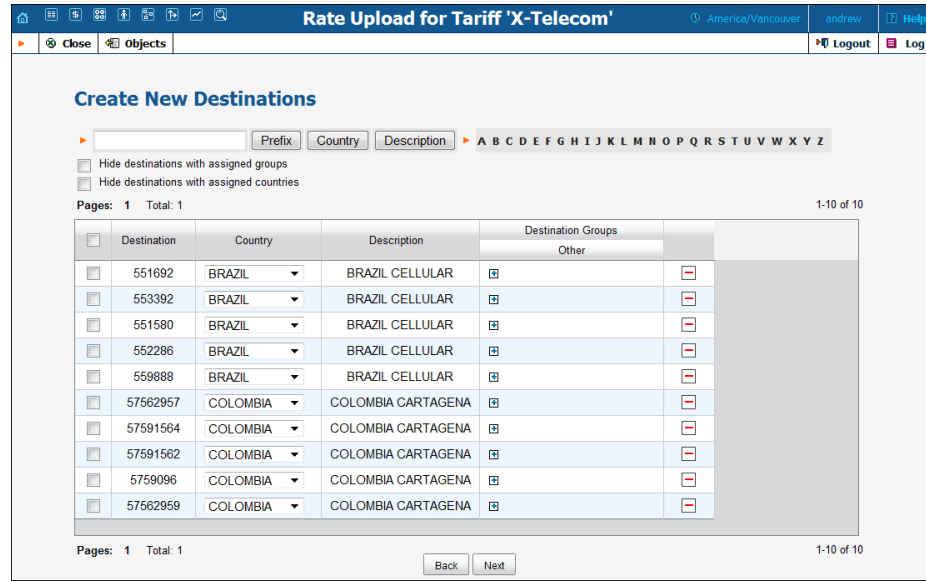
Carriers frequently send rate files containing prefixes that have not yet been defined in your system. These prefixes must be added into the system in order for rates to be created for them – so the big questions are: 'how will this affect the routing and billing on your system in general?' and 'how will other carriers and customers be affected?'

When adding a new destination, extra parameters may be provided in addition to the actual prefix - see below.

Assigning the country

Each destination is associated with a country. For a new destination the association has to be made before it can be inserted into the database. It is a good idea to upload the default set of destinations provided with PortaBilling®, or some other set that contains, at the least, destinations for each country, e.g. United Kingdom – 44, Germany – 49, etc. Then PortaBilling® can correctly assign the country name to the majority of new prefixes. Thus you do not need to choose the country for *each* destination manually (e.g. if you try to upload a destination 33040, but there is already a more generic one 33 in the system and the country has been defined as France – the upload wizard will propose assigning the number 33040 to France as well). If required, a country for the new destination may be chosen from the list on the **Create New Destinations** page.

NOTE: This is a required step – the rate import cannot proceed until each of the new prefixes has been assigned to a country.

Sadly, both the US & Canada pose the biggest challenges, since they (plus a handful of other countries, like Bermuda) share the prefix 1. This makes it impossible to guess the country code just by looking at the phone

number. For instance, 1204 is Manitoba, Canada and 1205 is Alabama, US! In these cases you will need to manually assign country codes for these destinations. The good news is that new area codes for the US & Canada do not appear that often.
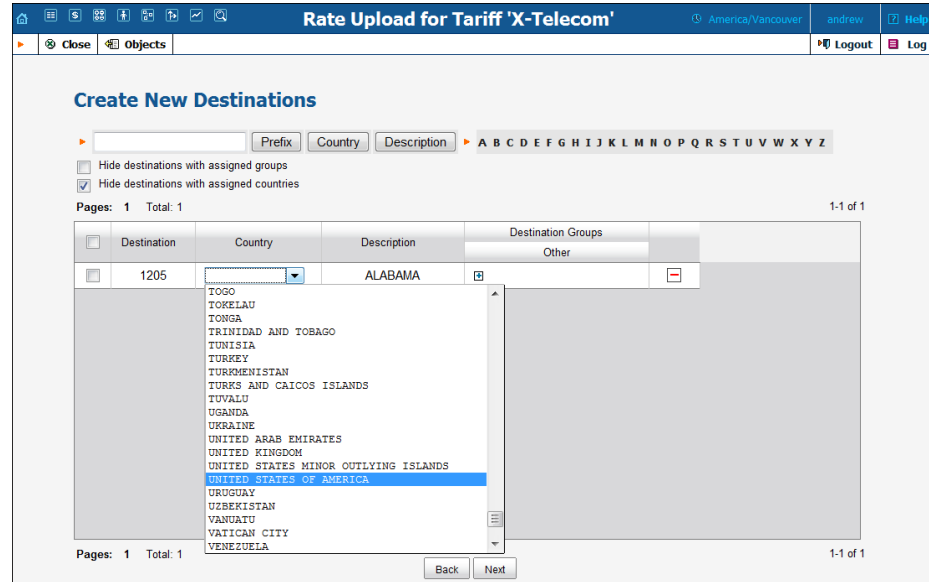
In our example we will assign the 670 destination to the country East Timor.
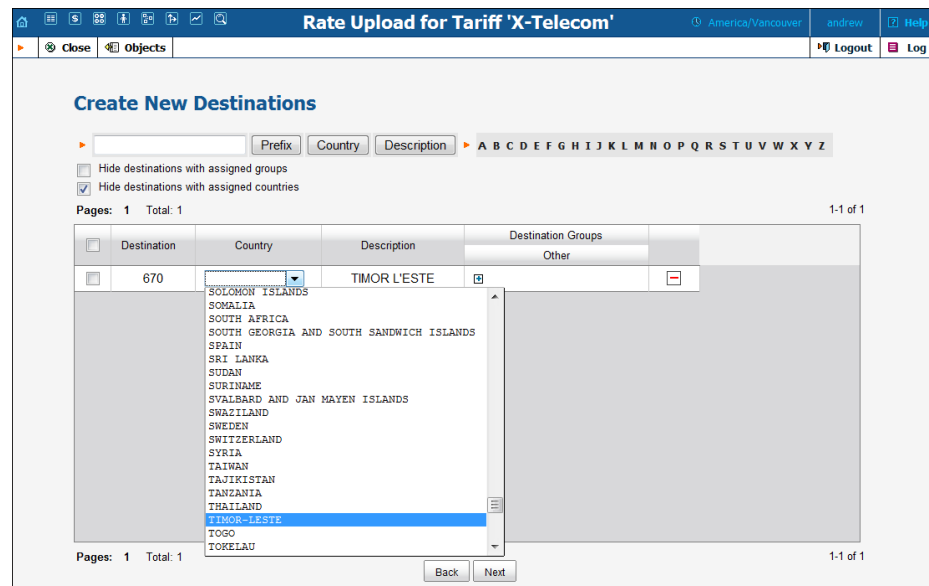
Including a destination into a destination group

Destination groups are invaluable for saving hours of time and for avoiding mistakes, since one can change the rate for a whole group with just a single click instead of modifying rates for many destinations (and

many could literally mean thousands), manually. So for instance, when the new destination 4473 is being added – it may be a good idea to immediately include it into a destination group (within the destination group set that's associated with this tariff) – UK-Mobile in this case. Then the next time someone updates the UK-Mobile rate, the change will be applied to this prefix as well. See the screenshots below for the sequence of actions.
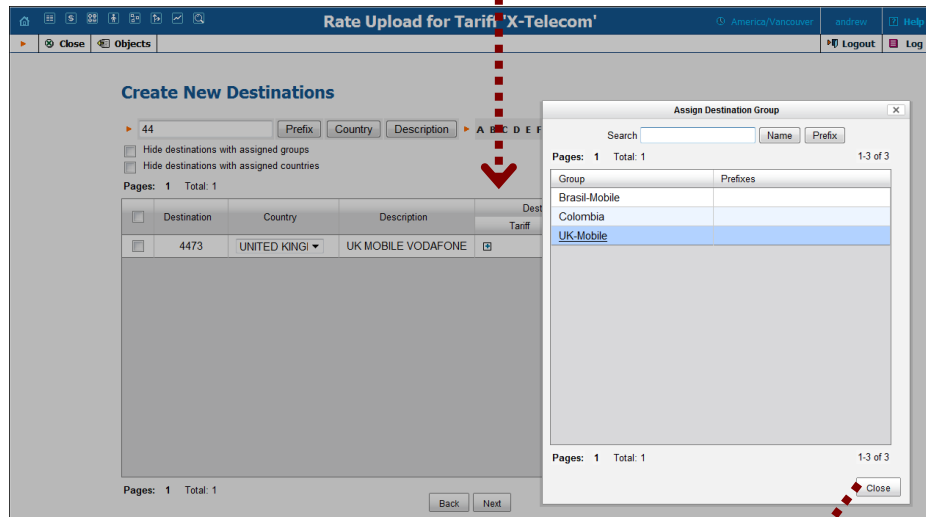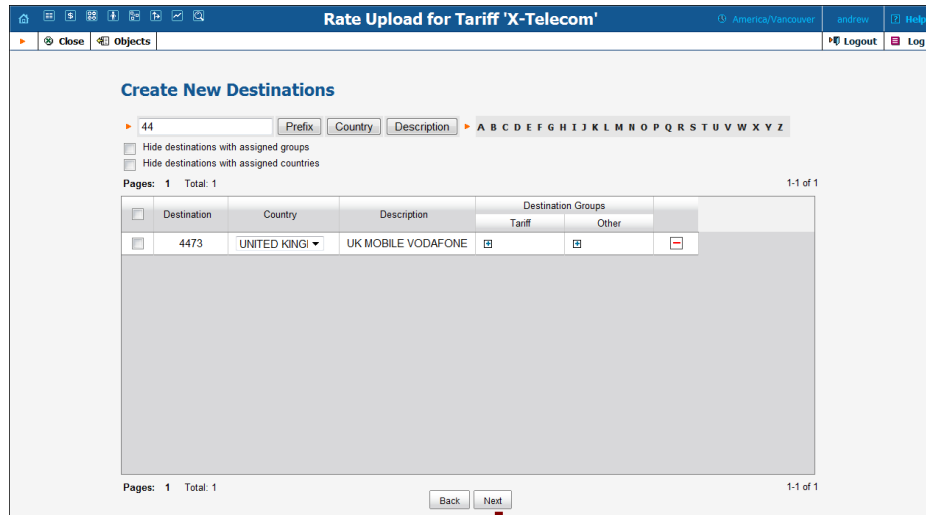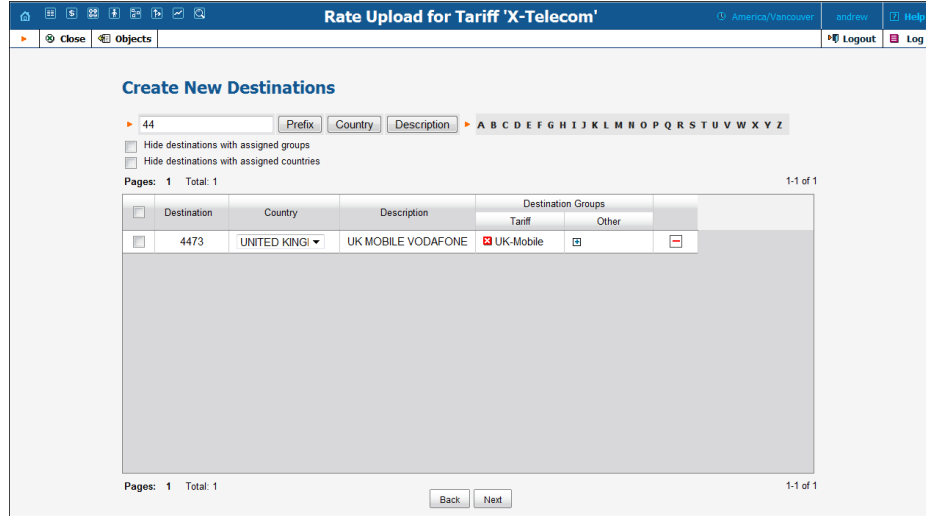
In some cases it may also be beneficial to include this destination into groups that are in other destination group sets. In our example, the new destination 4473 (UK-Mobile-Vodafone) is detected while uploading rates into the tariff of carrier X-Telecom (which has the X-Telecom destination group set associated with it). You may want to immediately add it into the UK-Mobile destination group in the Retail-Customers destination group set. This way the next time your colleague is generating rates to be applied to your customers – this prefix will appear and he can apply the special pricing for it.
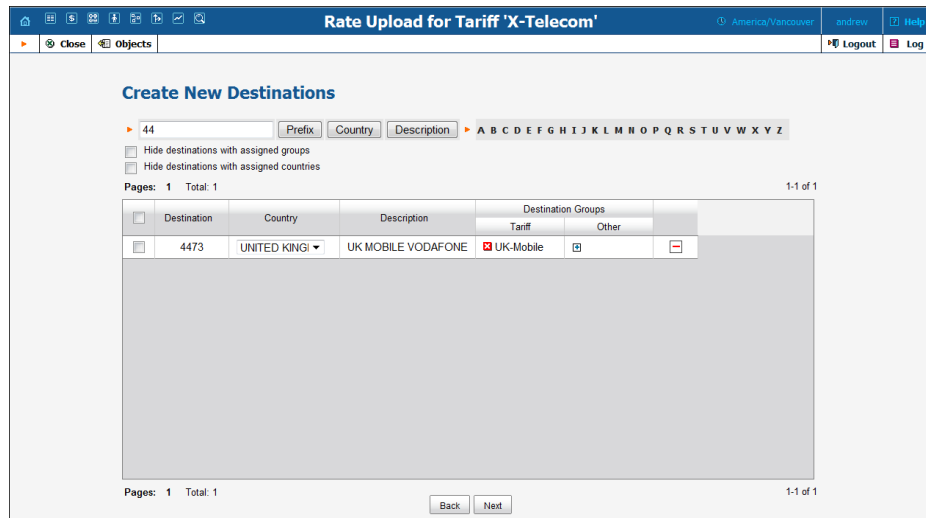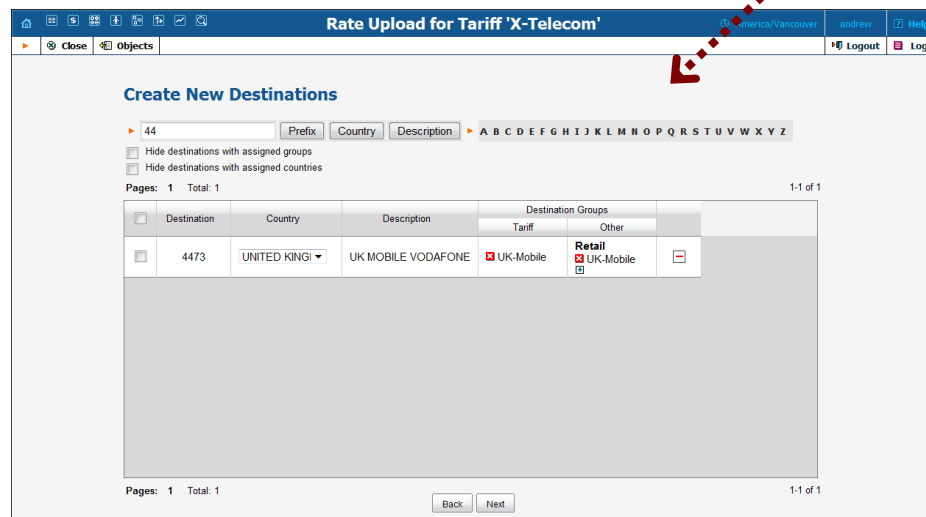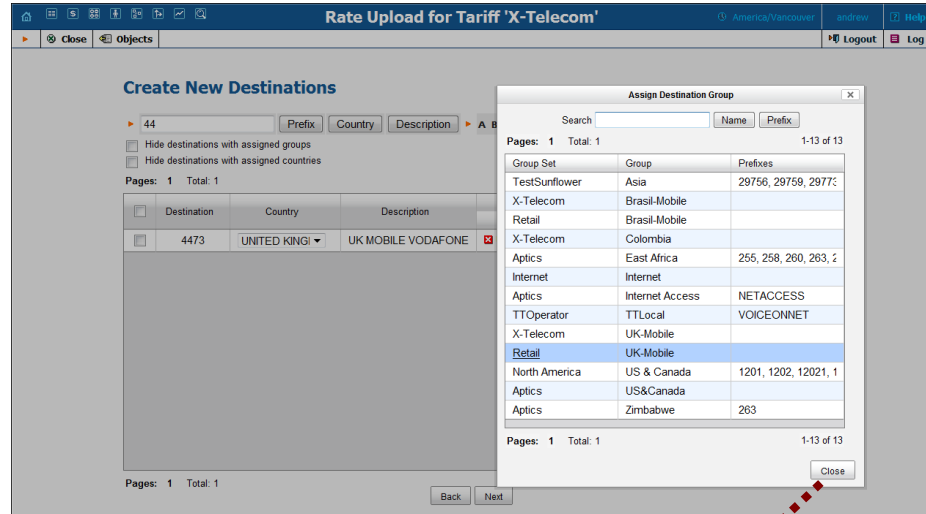
Multiple similar destinations

In our example you may notice that there are several destinations (55123, 55124, 55125, etc.) which look very similar. That's because they belong to the same country (Brazil) and they are in the same destination group (Brazil-Mobile).
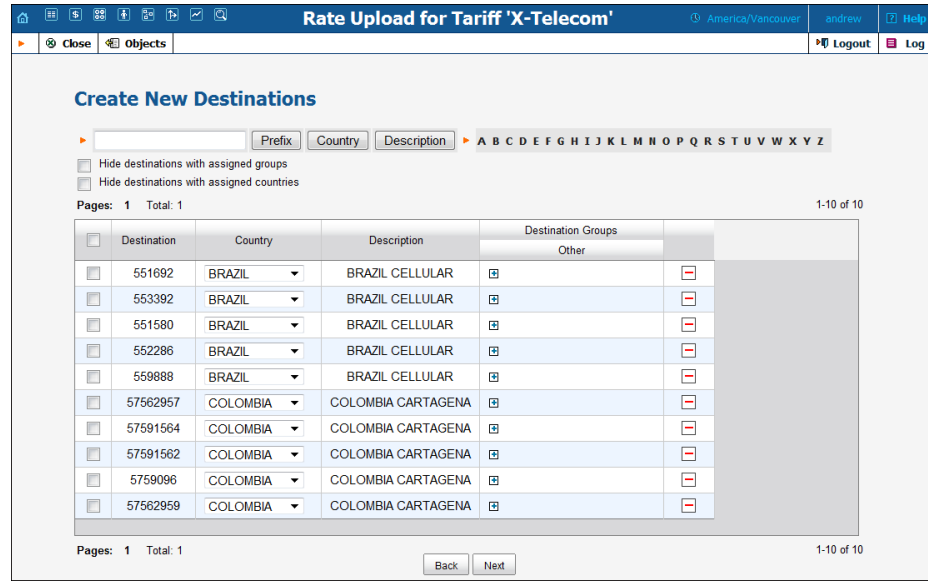
In order to save time and avoid assigning parameters to each one of them manually, you can do the following: assign the required parameters to just one of them, select the others using the checkbox on the left side of each row and then move the mouse over to the rows that don't have parameters assigned. A popup menu will appear – choose **Apply to Selected** and this will automatically change all the rows.

Please see how the change is applied to several prefixes for Brazil below.

Tip: You can filter the list of new destinations by country or hide the destinations that have already been assigned a country or destination group. For this action, use the **Select All** button in conjunction with **Apply To Selected** to quickly apply settings to hundreds of destinations.

If I add a new destination while uploading rates for carrier ABC will it affect my other carriers?

Customers who are used to old-style routing systems are very concerned about adding new prefixes, since on many of those, a carrier with a longer prefixes in their tariff would all of a sudden get priority routing (although it is not the cheapest or the one with better quality). Attempts to "harmonize" the prefixes (to ensure that all carriers use the same prefix set) require lots of manual labor and are Sisyphean in scope: the moment you finish changing the prefixes and things are finally looking good, one of the carriers sends you an updated prefix list and you have to start the task all over again.

This is not the case with PortaBilling®, which uses a true real-time routing engine giving each carrier their own set of prefixes for the tariff. When the routing is calculated, available routes are compared based on cost, preference and route category – never on the prefix length. So if you upload the rate for 4471 for carrier A (and other carriers only have prefixes 44 and 447) – when a routing for a call to 4471234567 is computed, the 4471 rate will be compared by price and preference with the rates for 44 or 447 from other carriers (who don't have the 4471 prefix). Consequently, the system always produces a correct routing and you don't have to worry about adding new prefixes that are only used by a few carriers.
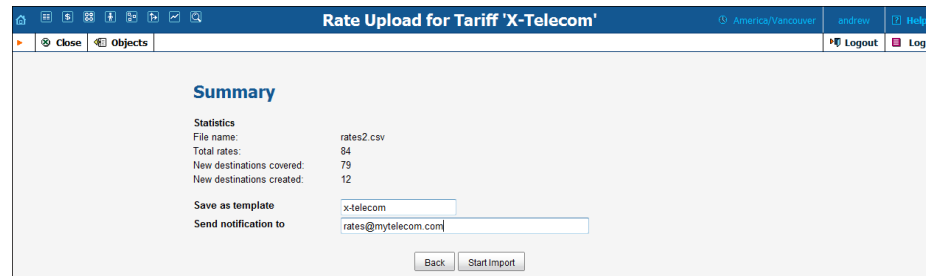
I do not want to add a destination right now, what do I do?

You can delete the destination from the list of destinations that will be automatically created during the import by using the ⊟ delete button.

Note that in this case, the rate for this destination will not be imported (since there is no destination to associate it with). To undo the delete destination action, click the ⊞ button, respectively.

Summary

On the **Summary page**, you will see an overview of information about the number of rate records to be processed. You can also specify a name under which the current import settings will be saved as a template. Later you can re-use them when uploading rates into this or other tariffs.

You may specify who should receive a report about the rates being uploaded, and then click **Start Import** to launch the rate import process. The report will contain following information:

- Total rates processed - the total number of rates in the imported file
- Total rates created - the total number of rates created including the newly created rates (the rates which were not present in the tariff prior to the import) and the overwritten ones
- Old rates overridden - reflects the total number of rates, which were updated (i.e. the previous rates were replaced by the new ones for some destination)
- Skipped (due to errors) – the number of rates that contained errors (e.g. a typo in the price column) and therefore were not added to the tariff. The information about the occurred errors you can see in the e-mail with the tariff upload report
- Skipped (duplicates) – the number of skipped duplicate rates. It means that if some rates (rate parameters) in the uploaded file were identical with the ones already present in the tariff, the system skipped the duplicates i.e. no extra (excess) rate were inserted into the tariff
- Rates for destinations, previously absent in this tariff, created – the number of rates created for destinations which were not in the tariff prior to the import
- Destinations, previously absent in the system, created – the number of new destinations created in the system (billing environment)

In the attached file you can see the detailed information about the result of the rate upload procedure in order to quickly understand exactly how the existent rates were changed. You can also see information about new rates added if there are any. See the example of the file on the screenshot below:

Usage of the Effective from Field

You can only upload rates that are effective immediately or for the future

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

You can schedule the discontinuation of a rate

It is possible to schedule a rate for a certain prefix to be discontinued on a given date in the future. This simplifies the management of the vendor's tariffs, since an announcement that a rate for a prefix is no longer available is typically received in advance. If you perform the upload immediately after receiving the updated rate sheet, the system will continue using the current rate for that prefix until the moment it actually becomes inactive (the date and time specified in the **Effective from** field).

When Is Importing Tariff Data Not Possible?

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

File format not supported by PortaBilling®

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

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

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

No Phone Prefix Information in the File

Unfortunately, this is a very common situation: your partner sends you a list of rates, but only includes the country or location name, as in the example below.

1. Go to **Destination Group Sets** and add a new destination group set to be assigned to this tariff.

2. After that, create destination groups. Click the **Add** button and fill in the name of the destination group that will coincide with the name of the destination (the country or location name) in the

tariff and specify the prefixes for this destination group. Create a destination group for every name provided in the tariff.

3. When creating a new tariff, select the already created destination group set in the **Destination Group Set** field and click **Save**.
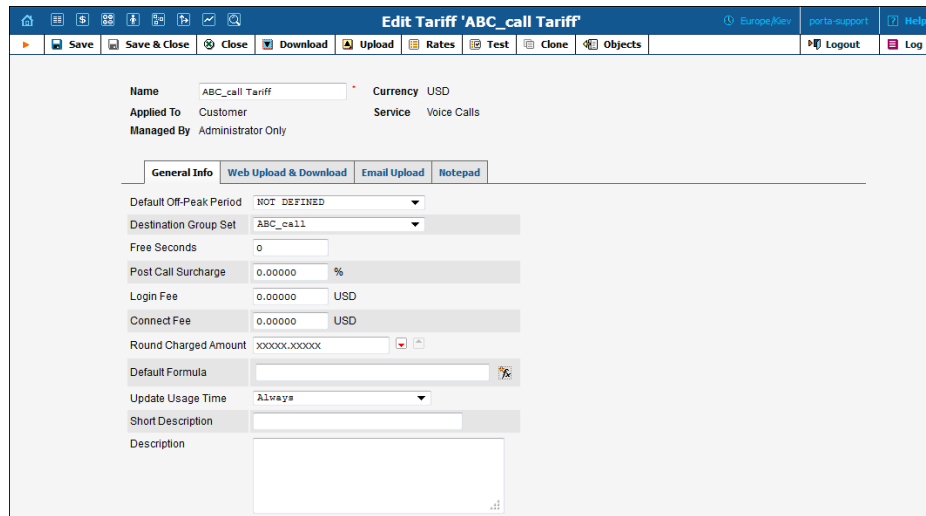
4. Click **Upload** and select the document that contains rates. You can see an example of rates below.

5. Click **Next** to proceed with a review of file parsing.

6. Click **Next** to review the data field definition. When you locate the "Destination" column A, select it and drag it back to the **Available Fields** list and assign the "Destination Group" instead.

7. Click **Next** to view the rate information. Make sure the rates coincide with the predefined destination groups and click **Next**.
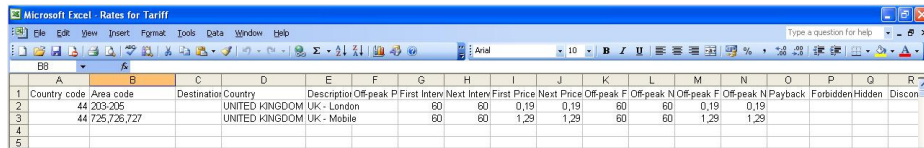
How to ...

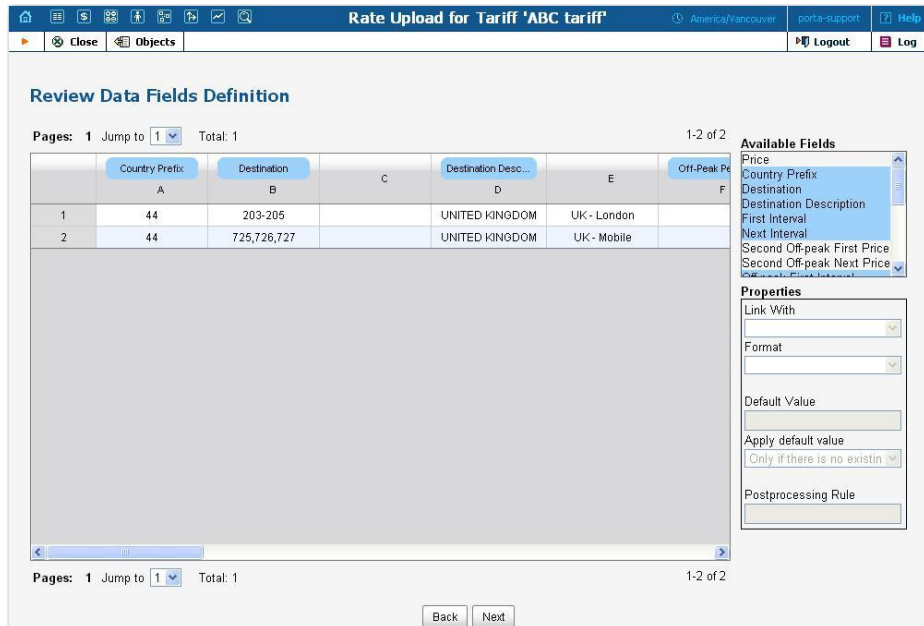## ...upload tariff rates with country code and area codes in separate columns

Sometimes vendors may send you tariffs in a format with country codes and area codes in separate columns. Starting from MR26, there is no need to edit manually .xls files in order to merge these two columns into one before upload.

You can upload rates with country code and area codes that are in separate columns. During the upload, the value of the **Country Prefix** column will join each destination value of the **Destination** column. For area codes you can use comma-separated values or the values separated by dashes (see an example below). The .xls file in this case should look like the following:

During upload on the **Review Data Fields Definition** page, assign the **Country Prefix** data field to the column with country codes and the **Destination** data field to the column containing area codes.

The result of performing the above-mentioned actions should look like the following:

…upload US inter- / intra- state rates

You may also specify whether the destination pertains to this category before the rate upload. Simply specify "_USInterIntraState_" in the **Off-peak Period** field. For interstate rates specify the price in the **First Price** and **Next Price** fields; for intrastate – in the **Second Off-peak First Price** and **Second Off-peak Next Price** fields. During the upload, the system will automatically recognize the destinations marked as inter / intra state and appropriate rates will be applied.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
4	Default Off	Default	Destination C	Free Seco	Post Call Surcharge	Login F	Conne	Formula	Update Usage Time						
5				0		0	0	0	Always						
6															
7	Destination	Destina	Country	Descriptio	Off-peak Period	First Int	Next Ir	First Price	Next Price	Off-pea	Off-p	Off-pea	Off-p	Second Off-peak First Price	Second Off-peak Next Price
8	1352		UNITED STA	Florida	_USInterIntraState_	1	1	1.5	1.5	1	1	1	1	0.5	0.5
9	1202		UNITED STA	Kentucky	_USInterIntraState_	1	1	2.22	0.7	1	1	1	1	0.3	0.3

After successfully configuring the inter / intra state rates for calls between different states the interstate price will be used. For calls within the state, the intrastate price will be used.

3. Invoicing

How Does Invoicing Work in PortaBilling®?

PortaBilling® provides a ready-to-use invoicing functionality. You can design how your invoices will look. Invoices are created automatically based on calls the customer has made; they 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 information required for invoicing, such as address, taxpayer ID, and so forth. Individual accounts are not invoiced.

How often?

Each customer has his own billing period, which could be:
- Daily
- Weekly
- Bi-weekly
- Monthly (anniversary or on the first day of every month)
- 30 calendar days

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

Note that the billing period is based on the customer's time zone. For example, if the customer has a weekly billing period and his time zone is Australia/Melbourne, his statistics would cover the period from 24-Mar-2009 00:00 Melbourne time (23-Mar-2009 14:00 GMT) until 31-Mar-2009 00:00 Melbourne time (30-Mar-2009 14:00 GMT). For another customer, who also has a weekly billing period but is in the New York time zone, this billing period will cover calls made from 24-Mar-2009 00:00 New York time (24-Mar-2009 05:00 GMT) to 31-Mar-2009 00:00 New York time (30-Mar-2009 14:00 GMT).

When are statistics available?

Statistics are generated with a sufficient delay to make sure that the billing period for a particular customer (according to that customer's time zone) has ended. Also, statistics calculation takes place only at a certain time of day (usually during night hours, when there is the least amount of activity

in the system). Weekly invoices cover Monday through Sunday. Assuming that your system does statistical calculation from 2am until 8am Central European Time every day, then statistics for those customers whose time zone is 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 no invoice-on-demand functionality in PortaBilling®, i.e. invoices should match customer billing periods, and are generated automatically. 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 also create a xDR record, which will be included with the customer's next invoice.

Invoice Total

There are two methods available for calculating the invoice total:

All charges during the period

This is the simpler method - the invoice contains all of the call charges for 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 will 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, the customer's payment will be recorded in the system, and will affect his 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

Invoicing modes are controlled by the `New_Style_Invoice_Calculation` configuration option.

Basis of Invoices

For a reseller, the call charges on the invoice are based on the sum total of all his xDRs in a given period.

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

Invoice Template

The invoice template is a specially formatted HTML document, which is created using either the PortaBilling® layout designer or an external tool (e.g. any third-party HTML editor). We suggest generating large invoices (hundreds of pages and more) using the external template uploaded into PortaBilling® as it requires less computing power for PDF file generation in comparison with the layout designer. See the following chapters for a detailed description for each approach. It defines how the invoice should look, with information on fonts, colors, pictures, and placement of the components. When you view the invoice on the web interface, or when the invoice is automatically emailed to the customer, the template engine takes the invoice template and simply fills in the data fields (such as **Header.Invoice Number**) with their real values (for example, **1001** as the invoice number). The result of this processing is a PDF file.

Requirements for invoices vary greatly from country to country (and even within the same country). The default invoice templates supplied with the system are suitable for British Columbia, Canada, and are provided as examples only. Be sure to create your own invoice template and assign it to your customers.

Two predefined invoice templates for Layout Designer are supplied with PortaBilling®: "A single page with totals only" and "First page with totals + usage details attached on additional pages." What is the difference between them?

A single page with totals only invoice

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

First page with totals + usage details attached on additional pages invoice

This template's first page is identical to the **single page with totals only** invoice template. In addition to this, it contains multiple pages with details of calls related to the invoice. You can use the button in Layout Designer to switch between the main page and the call details page. All of the calls made by a customer can be included, or only summaries.

Billing, Invoicing and Taxes

Does PortaBilling® perform billing with or without taxes? Actually, PortaBilling® is a rating engine, so it all depends on how you use it. PortaBilling® does all the calculations based on rates you enter into the system, as well as tariff parameters and call duration. So, in this sense, it makes no difference whether the price includes tax or not.

Thus, there are two scenarios available for dealing with taxes, and each has its own advantages and drawbacks.

Calculating Taxation CDRs at the End of the Billing Period

If you enter rates without tax into PortaBilling®, the CDRs for calls (and other services) will not include the taxes. You can include all the necessary tax calculations when you create an invoice for your customer. The calculation of taxation CDRs is performed prior to invoice calculation at the end of the billing period. The customer's actual balance will be identical to the invoice total and will reflect all proper tax charges.

The advantage of this method is that it gives unlimited flexibility: you can implement support for any taxation scheme you need. Its disadvantage is that since all the CDRs exclude taxes, customer balances do not include taxes until the end of the billing period when a taxation CDR is created. Be advised that this method will not work with prepaid cards: if a customer buys a $10 calling card, he naturally expects to spend $10 on calls and wants to know what his price per minute will be – without complications like taxes.

Calculating CDRs with Taxes

If you enter rates into PortaBilling® including taxes, you will get a "final" price, and the CDRs are calculated accordingly (with tax included). In this case, the balances are correct, and prepaid products work fine. The only complication is providing proper tax information on the invoice (if this is

required), since now it has to be done backwards (i.e. you have to deduct tax from the amounts).

This is the recommended method, as it provides accurate billing for both prepaid and postpaid products. The default invoice template supplied with PortaBilling® provides an example of how to set up invoicing in this way.

Back-calculating Taxes Based on Invoice Total

If you wish to include tax in your rates, you need to show proper tax information on the invoice. Five data fields are available for different types of taxes (for example, GST and PST in Canada), which you can fill in using different formulas. These all receive a total invoice value as their input value; then you can calculate the actual tax value using your own formula in a **post-processing rule**.

The PortaBilling® template library has a function which can be used to easily calculate the amount of certain types of tax. The `Porta::Format::Number::backcalcTax` function consists of three parameters:

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

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

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

while for **Tax2** the post-processing rule would be:

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

As you can see, this is quite simple. Instead of writing `7 + 7.5` you could have written `14.5` – but why do the calculations if the template processor can do them for you?

Finally, it is only necessary to put **Footer.Tax1** and **Footer.Tax2** fields on the invoice where you want them to be, and label them appropriately.
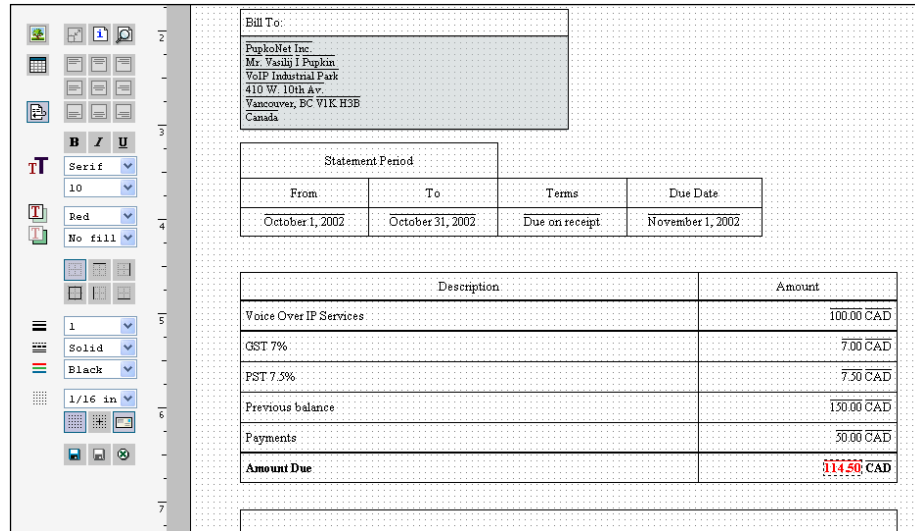
NOTE: The default **Invoice** template represents an invoice sample that is applicable for BC, Canada locale. So, be sure to customize it to your needs.
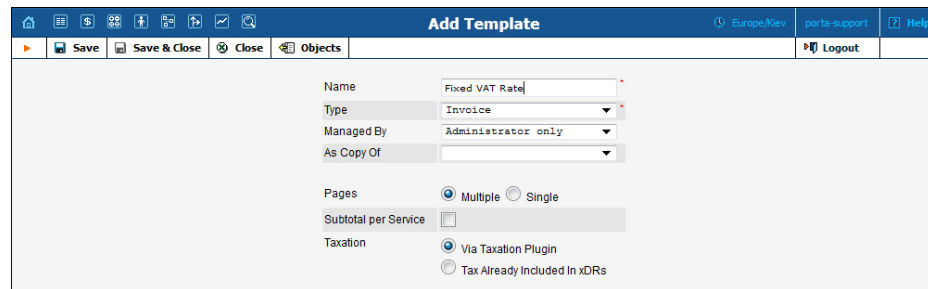
NOTE: If any plug-in for calculating tax amounts is integrated with the billing system (like the one for BillSoft EZTax suite, etc.), back-calculation of taxes via the template should not be used, since it will produce incorrect results.

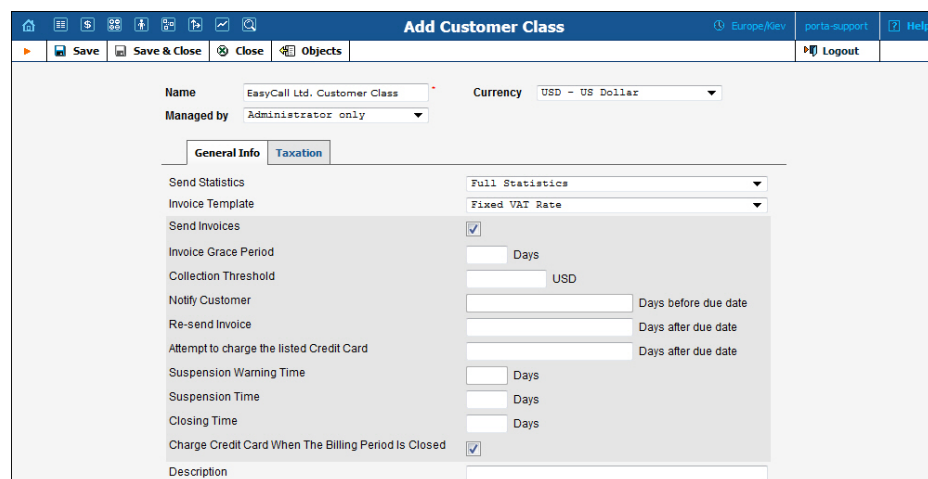## Applying fixed VAT rate using customer classes

Let's consider the example of including a fixed tax into a customer class, which will then be applied to the customer:
1. Create a new invoice template.
   - Tick **Via Taxation Plugin** in the **Taxation** tab.
   - Click **Save**.
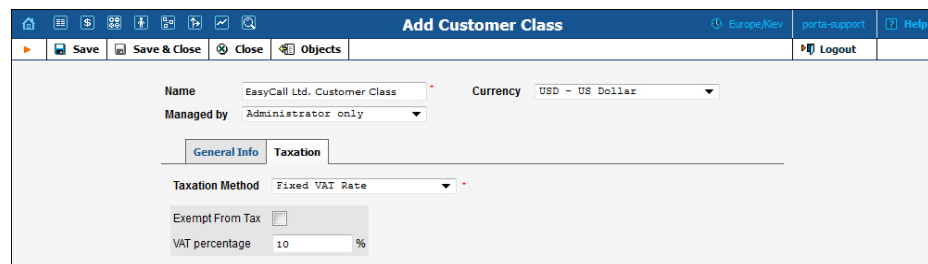2. Go to the **Customer Classes** page.

- Create a new customer class or modify an existing one.
- Select the created above template in the **Invoice Template** field.
- Go to the **Taxation** tab on the same page.
- Leave **Exempt From Tax** box empty.
- Specify the desired percentage in the **VAT percentage** field and click **Save**.

Applying fixed VAT rate without using customer classes

To apply a fixed tax directly to the customer, do the following:
- Go to the **Taxation** tab on the **Customer** page
- Select **Fixed VAT Rate**
- Choose **No** in the **Exempt From Tax** field
- Specify the required tax percentage.

NOTE: Make sure you are using an invoice template for the exclusive method (Via Taxation Plugin).
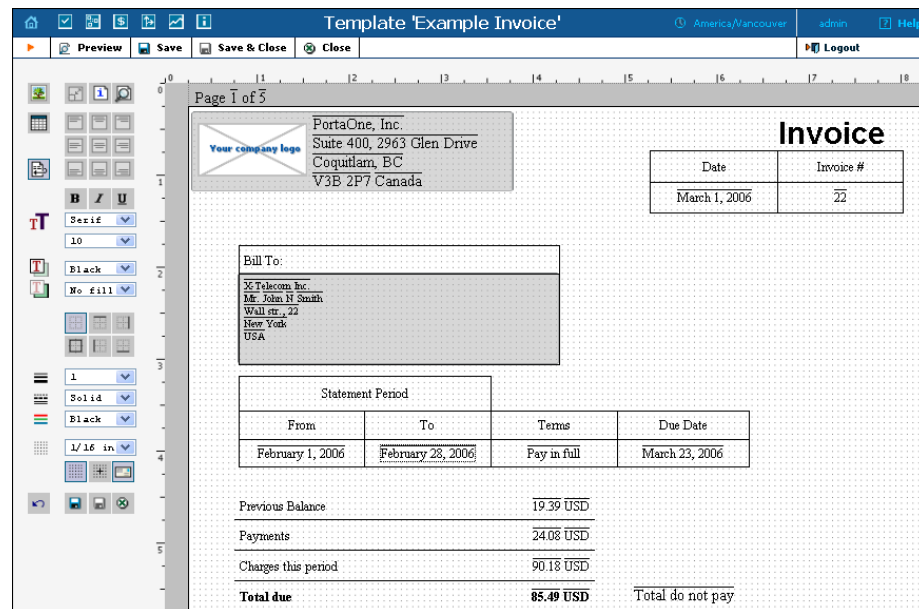
4. Layout Designer

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, and no installation of extra components on the user's computer is required.

When you design a template in Layout Designer, you define how the document should look: the location of components, pictures, fonts, and colors. You also place the data fields on the document. These are containers for real data, defining where the data element should appear.

For example, on the invoice template you put an **Invoice.number** data field in the place where you would like the invoice number to be printed (top-right corner, in the example below). When the actual invoice is generated based on this template, **Invoice.number** will be replaced by the actual invoice number (e.g. **557**). Make sure you insert the data field using the pop-up menu, because if you simply type **Invoice.number** it will be regarded as a text string, and so will appear on the invoice exactly as you have typed it.

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

Layout Designer consists of two main parts:
- Toolbar (on the left). It contains buttons and controls for manipulating content in the editor window.
- Worksheet (on the right). It contains the document itself.

General Document Properties

Page Setup

In order to access page setup properties, right click on the empty background of a template window (i.e. not on an object such as a table or image).
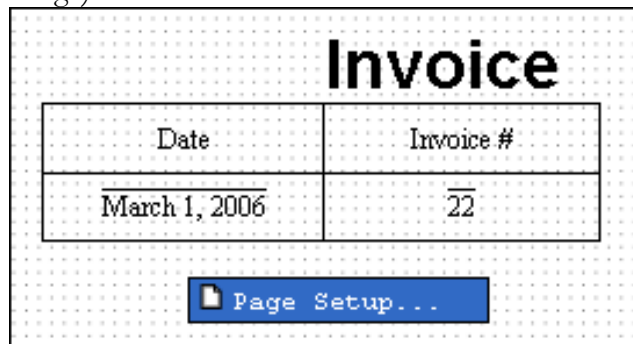
Choose the "Page Setup" item.

You can choose the paper format, margins, and the position of envelope windows. Consult the online help for a detailed description of different envelope styles.

Grid Properties

The grid helps you easily adjust objects in the template. There are a few buttons on the toolbar which control the grid's behavior:

- ▦ - Turn the grid on or off (by default it 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 place an object between the grid lines when dragging or resizing it.

- The select menu allows you to choose the size of grid cells.
- ■ - Shows or hides the envelope window on the template.

Working with Images

You can add your own images to the template. These images will then be stored on the server for use by anyone browsing documents created from the template. To add a new image:

- Click on the empty background of the template (i.e. no other object, such as a picture or table, should be selected). The **Add picture** button will become enabled.

- Click on the **Add Picture** button.
- The **Open File** dialog will appear. Choose a picture on your local disk. Any picture format which is supported by popular web browsers can be used (.JPG, .GIF, .PNG).
- Click **Open** and the picture will be transferred to your template. You can now move or resize it.

Placing Data Elements

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

- Click on the empty background of the template (i.e. no other object, such as a picture or table, should be selected). The **Add Table** button should become enabled.
- Click on the **Add Table** button.

A new table will then be added to the template. This is a very basic table, consisting only of 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 perform advanced operations on the table (e.g. adding a row), right click on the table so that a menu appears.
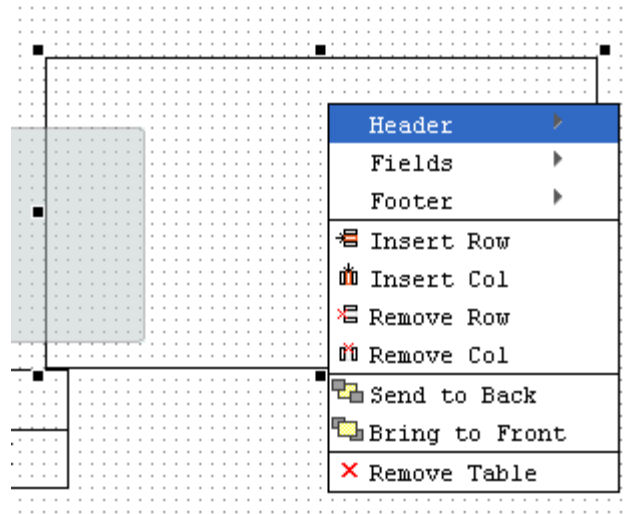
A table or a cell within a table can be moved and scaled by dragging the cell handles. Double click a table or a cell to enter cell editing mode, where you can type in or edit the cell's text. To change the attributes of the cell's text, make sure the cell is selected (8 black square handles are visible) and set the text attributes in the toolbar on the left.

Changing a cell's text attributes in the toolbar will affect all text in the current cell. To change the attributes of part of the text, enter cell editing mode (double-click), select part of the text, and use the keyboard shortcuts listed below:
- Ctrl + B - Bold
- Ctrl + I - Italic
- Ctrl + U - Underlined

To insert data fields into the cell, do the following:
- Right-click on the cell where the data field should be inserted.
- In the top part of the menu, click on the component name to which the data field belongs (e.g. **Header**). A submenu with a 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 go into cell editing mode and add text before or after the data field. Also, by removing all of the over lined data field name, you can completely delete the data field from the cell.

Changing Text Attributes

You can apply multiple formatting styles to text elements in a table (this includes both static text and data fields), namely:
- Text alignment within the cell,

- Text font and font attributes, text color,
- Table cell borders and border style.

Let's consider the example of changing the "Bill To:" table text attributes:

- align the text to the middle left position
- change font style to Bold, font Family to Fantasy and set Font Size to 9
- Apply green Foreground Color and wheat Background color
- Set square Table Border.

Then click **Save**. Now you can generate a preview by pressing the **Preview** button.
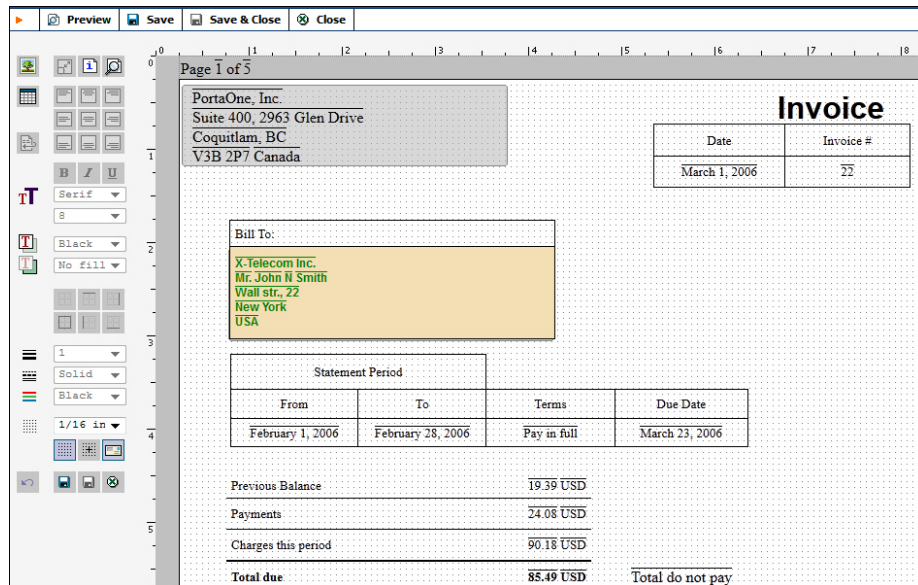
Consult the online help for a complete list of all possible text operations.

5. External Invoice Templates

As an alternative to the WYSIWYG invoice layout designer (built into the system), you can use another approach in which an invoice template can be designed as an HTML+CSS file and then uploaded into the system. This simplifies the process if the invoice template is created by an external design agency (that does not have access to PortaBilling®) and allows advanced template customizations (e.g. arrangement of data to exactly match "legacy" invoices or insertion of dynamic content such as banners) by third-party developers. Also we suggest using this approach for large invoices (hundreds of pages and more) as it requires less computing power for PDF file generation in comparison with the WYSIWYG invoice layout designer.

Industry-leading PDF generation software PrinceXML is used to produce the actual PDF files. The license for PrinceXML is provided as a part of an OEM agreement for all PortaOne customers.

Creating an External Invoice Template

To create an external invoice template follow these steps:
1. Create an HTML+CSS file for your custom template or modify an already existing example according to your requirements. Several examples can be found in the following folder: /home/porta-admin/templates/custom_invoice/default
2. Prepare image files that you will insert into the template. The images must be defined in the HTML+CSS file, for example with the following tag:
   ```
   <img class="advert" src="[% image_path %]ad.jpg" />
   ```
3. Upload the new invoice template on the PortaBilling® administrative web interface.
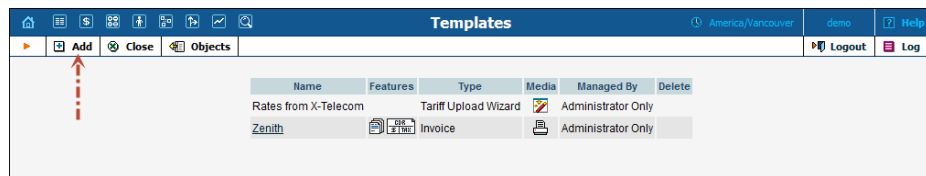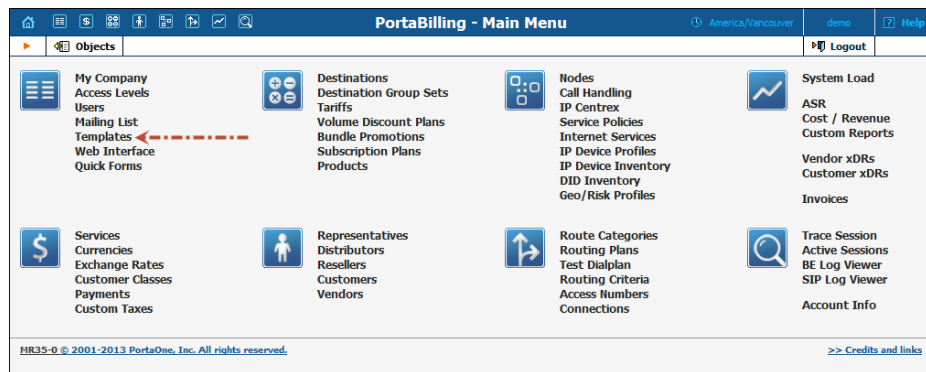
4. Assign the uploaded invoice template to a new or existing customer.

5. Check the look of the invoice template by generating an out-of-turn invoice for a test customer (optional). For this you may add a manual charge or manual credit transaction on the **Balance Adjustments** tab.

6. Below is an example of the invoice generated using a custom invoice template.

This external invoice template can be downloaded from
www.portaone.com:

`http://portaone.com/resources/invoice_template.zip`

In **APPENDIX A. Variables That Can Be Used in an External Invoice Template**, you can find a list of variables and their descriptions that can be used to show different information in the invoice.

In **APPENDIX B. External Invoice Template Example**, you can find the invoice.tmpl file. This example is analogous to a default template of the invoice layout designer.

In **APPENDIX C. Advanced Template Customizations**, you can find examples of advanced template customizations (e.g. arrangement of data to exactly match "legacy" invoices.)

Updating an Earlier Uploaded Template

To introduce changes and adjustments into an invoice template that was already uploaded follow these steps:

1. Update the HTML+CSS file for your custom template according to your requirements.
2. Prepare new image files that you will insert into the template.
3. Upload the updated invoice template on the PortaBilling® administrative web interface.
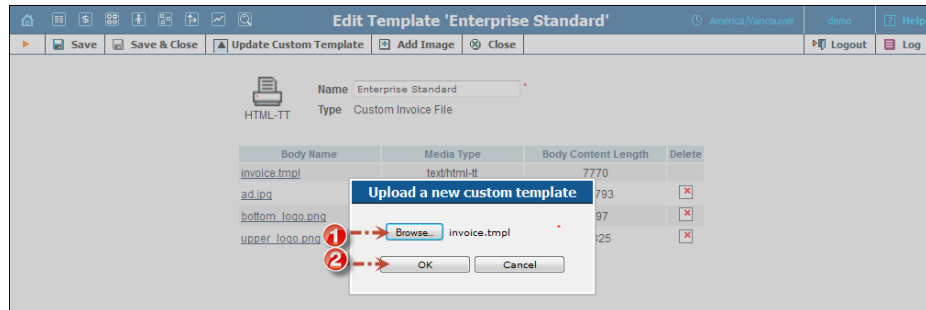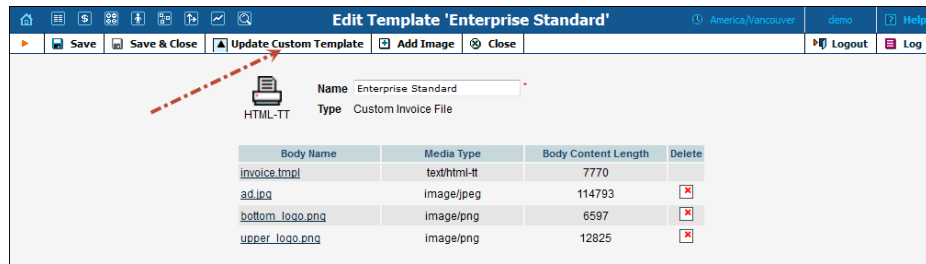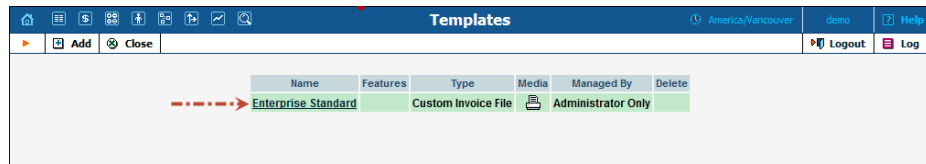
4. Delete images that are no longer used and upload new images defined in the invoice template.
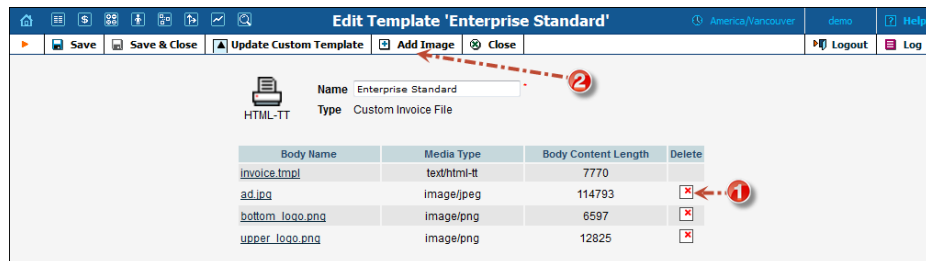
6. Templates for Outgoing Notification Messages

Notification Templates
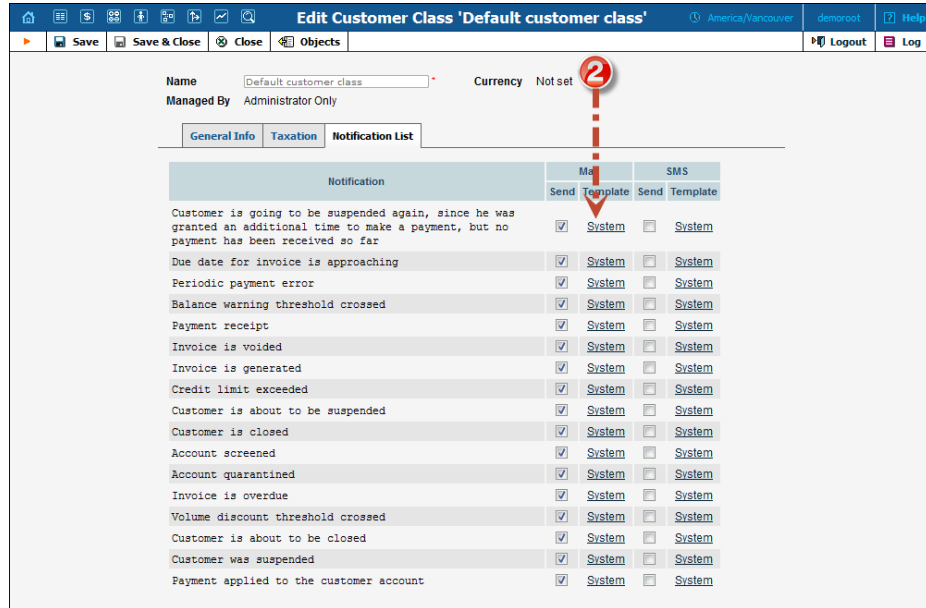
There are many types of email and SMS messages sent from PortaBilling® to your administrative and end users: notifications about a balance being dangerously close to the credit limit, emails with statistics and / or an invoice, different alerts sent to administrators and so on. It is possible to customize the language and content of these messages.

Notification Templates per Customer Class

You may find notification templates on the **Notification List** tab of the **Customer Class** page. On this tab you can view the name of the template, sending preferences and template status: **System** (the default template) or **Custom** (the modified one). Upon clicking on the **Custom / System** link you will be redirected to the **Edit Notification Template** page that allows you to view the type of notification template and modify the subject, body, format and post processing rule for variables (**PP Rule** on the web interface).
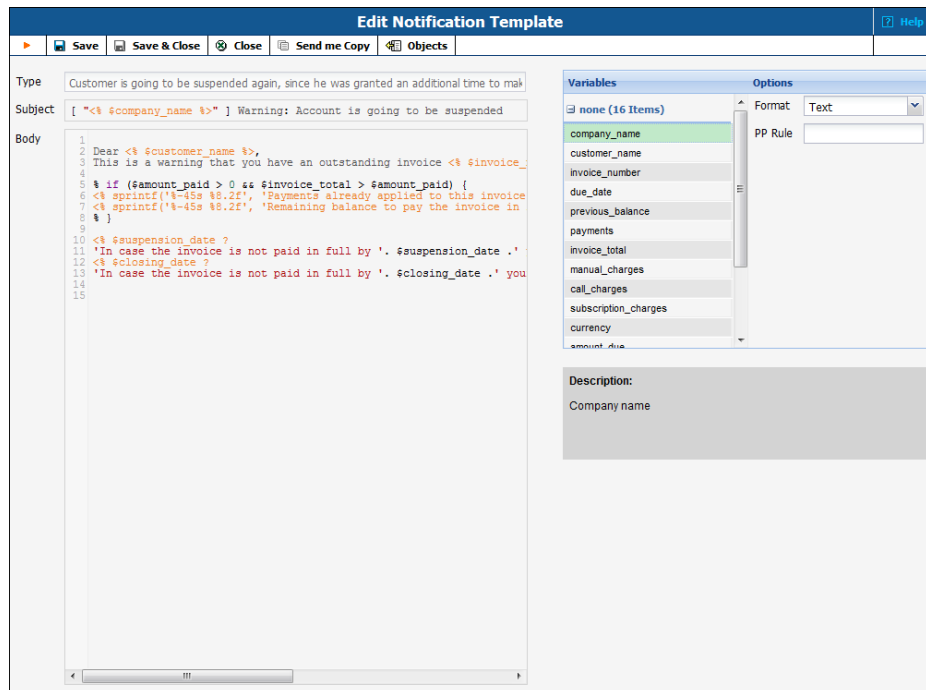
In the **PP Rule** field you can specify Perl expressions just as in other templates within PortaBillling®.

Mail Template Syntax

NOTE: Mail templates use basic HTML::Mason syntax; you can find more information on the **HTML::Mason homepage**.

Here is an example of the default **Balance warning threshold crossed** template:

Subject: Credit limit warning

Body:

```
Dear <% $greetings %>,

We would like to warn you that the current balance on your
account <% $companyname ? $companyname : $name %> is <% $balance
%><% $currency %>.
Your credit limit is <% $credit_limit %><% $currency %> and when
your balance reaches it, your service will be blocked.
This is an auto-generated warning, your current warning
threshold is <% $credit_limit_warning =~ /%/ ?
$credit_limit_warning : $credit_limit_warning.$currency %>.
```

Variables:

```
greetings
companyname
name
balance
credit_limit
credit_limit_warning
currency
```

Each template contains **subject, body** and **variables** sections. The **variables** section defines the parameters (e.g. Current customer balance) that can be used in the body section.

The **subject** section contains the text to be entered in the subject line of the outgoing email (e.g. Credit limit warning).

The **body** section contains the text which will be used as the email body text. Any normal text entered here will appear unchanged in the email. If you want to include the current value of a certain parameter in the text, enclose the parameter variable in `<% %>` as shown below:

```
Your current credit limit is <% $credit_limit %>.
```

In fact, you may include mini-pieces of Perl code in `<% %>`. For instance, the line below includes a conditional statement, and will produce either a company name (if one is defined) or a customer name:

```
<% $companyname ? $companyname : $name %>
```

In **APPENDIX D. Available Email Templates**, you can find email templates and their descriptions.

7.   Appendices

APPENDIX A. Variables That Can Be Used in an External Invoice Template

Template Name	Description
Environment for direct customers, Reseller for sub-customers	
env.i_env	The unique ID number of the environment
env.name	Environment name
env.lname	Proper, legally-recognized form of the company's name
env.lcountry	Country in which the company is legally registered
env.lcity	City in which the company is legally registered
env.subdivision	Country subdivision name
env.iso_3166_2	Country subdivision code
env.laddr1	Address where this company name is legally registered
env.laddr2	...
env.lzip	Postal code for the company's registered location
env.tax_id	Locally-designated tax number
env.iso_4217	The company's internal currency
env.companyname	The conventional form of your company's name
env.addr1	Street address of the company
env.addr2	Street address of the company
env.addr3	Street address of the company
env.addr4	Street address of the company
env.country	Country name
env.iso_3166_1_a2	Country code
env.phone	Primary telephone number for the company
env.email	Primary e-mail contact
env.faxnum	Primary fax number
env.web	Web address of the company.
Customer information	
customer.i_customer	Unique numeric number of customer
customer.name	Customer ID, defines the customer name as it will appear in the PortaBilling® system
customer.companyname	The conventional form of company's name
customer.salutation	Something like 'Mr.', 'Ms.'
customer.firstname	First name

customer.lastname	Last name
customer.midinit	Middle name
customer.lname	Either company name, or first name + last name
customer.country	Country name
customer.city	City name
customer.address	Full address line
customer.baddr1	Address line 1
customer.baddr2	Address line 2
customer.baddr3	Address line 3
customer.baddr4	Address line 4
customer.baddr5	Address line 5
customer.zip	Postal code
customer.cont1	Primary contact
customer.email	Primary e-mail contact
customer.phone1	Primary telephone number
customer.cont2	Alternative contact
customer.phone2	Alternative telephone number
customer.faxnum	Primary fax number
customer.TZ	Time zone
customer.out_date_format	Preferable date format
customer.out_date_time_format	Preferable date-time format
customer.out_time_format	Preferable time format
customer.balance	Customer's balance at the time of PDF generation
customer.tax_id	Locally-designated tax number
customer.billing_period	Defines the frequency of invoicing for this customer
customer.iso_4217	The customer's currency
customer.opening_balance	The starting balance for this customer
customer.bp_charge_cc	Y or N, defines if the customer's credit card will be charged when the invoice is made
customer.creation_date	Date when customer was added into PortaBilling® system
customer.credit_limit	Customer's current credit limit
customer.discount_rate	Amount of discount applied by default to all subscriptions of this customer
customer.i_customer_class	Numeric ID of the customer class assigned to customer
customer.i_customer_type	1 (Retail), 2 (Reseller), 3 (Distributor)

customer.i_lang	Localization language, 2-characters ISO639-1 code (examples: en, es)
customer.login	Customer self-care login
customer.notepad	Text entered in Notepad field on web
customer.custom_field1	Value of the first custom field
customer.custom_field2	Value of the second custom field
...	...
customer.custom_fieldN	Value of the Nth custom field
Invoice information	
invoice.i_env	Environment ID
invoice.i_invoice	The unique ID number of the invoice
invoice.invoice_number	Visible invoice number
invoice.i_customer	Reference to Customer's ID number
invoice.period_from	The start date of invoice period
invoice.period_to	The end date of invoice period
invoice.issue_date	The date when the invoice was created
invoice.due_date	Due date of the invoice
invoice.due_days	Days left to the due date
invoice.payments	Sum of payments made in the invoice period
invoice.manual_charges	Sum of manual charges made in the invoice period
invoice.subscriptions	Sum of subcription charges made in the invoice period
invoice.taxes	Sum of taxes, non-zero only if taxation plugin used
invoice.calls	Sum of charges for all other services
invoice.amount_net	Total amount for the invoice period
invoice.previous_balance	Due amount from the previous invoice
invoice.amount_due	Due amount for the current invoice
invoice.i_invoice_status	Invoice status, refer to value from Invoice_Status_Attribs table
invoice.invoice_status_desc	Invoice status description
invoice.is_void	N for normal invoices, Y for voided
invoice.iso_4217	Currency
invoice.tax_method	1 if taxation plugin used, 0 if taxes are included in rates
Service totals for given invoice	

services	Array indexed by i_service number, value is 'Service totals'
Service totals (allow to get xDRs for the chosen Services)	
i_service	The unique ID number of the Service
name	Service name
i_service_type	The unique ID number of the service Type
rating_base	Either 'quantity' or 'session-time'
period_totals.records	Accumulated number of records
period_totals.quantity	Sum of quantity-based service
period_totals.amount	Accumulated sum of charged amount
period_totals.tax_amount	Accumulated sum of taxes
period_totals.total_amount	Accumulated sum of charges + taxes
next_row	Iterator, which returns next xDRs row for this service, or undefined if there are no more records
xDR row	
account_id	Master Account ID
account_batch	Account Batch name
xdr_account_id	Account ID in xDR, may be Account Alias
xdr_id	ID of xDR
xdr_connect_time	Connect date-time
xdr_disconnect_time	Disconnect date-time
xdr_cli	CLI
xdr_cld	CLD
xdr_h323_conf_id	H323-conf-id
xdr_h323_incoming_conf_id	incoming H323-conf-id
xdr_originating_ip	Originating IP
xdr_call_id	Call-ID
xdr_charged_amount	Charged amount
xdr_charged_quantity	Service quantity charged
xdr_used_quantity	Service quantity used
xdr_peak_level	Peak level applied
destination_destination	Destination used
destination_description	Destination description
destination_group_name	Destination group name
country_name	Country corresponding to destination

country_subdivision_name	Country subdivision corresponding to destination
tariff_i_tariff	Tariff ID used
tariff_name	Tariff name used
Other variables	
image_path	Path to the template images, used to make a full link to the included image

APPENDIX B. External Invoice Template Example

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html>
[% USE format -%]
[% money = format('%.2f') -%]
<html>
<head>
<style type="text/css">
@page {
    size: A4 portrait;
    margin-left:   0.25in;
    margin-right:  0.25in;
    margin-top:    0.25in;
    margin-bottom: 0.25in;
}

@page {
    @top-left {
        content: "Page " counter(page) " of " counter(pages);
        font-family: serif;
        font-size: 9pt;
    };
}

table { border-collapse: collapse; }
body { font-family: serif; }

.abs { position: absolute; }

.logo { top: 10mm; left: 10mm; }
.company { top: 10mm; left: 40mm; width: 50mm; }
.invoice { top: 10mm; left: 150mm; font-size: 24pt; font-weight: bold; }
.date-num {
    top: 20mm;
    left: 120mm;
    width: 70mm;
}
.date-num td {
    border: black solid 1px;
    font-size: 10pt;
    text-align: center;
    vertical-align: center;
    height: 8mm;
}

.bill-to { top: 50mm; left: 10mm; width: 80mm; }
.bill-to td { border: black solid 1px; padding-left: 3px; }
.bill-to-header { height: 10mm; }
.bill-to-body { height: 30mm; }

.statement { top: 95mm; left: 10mm; width: 130mm; }
.statement-header1 { table-column-span: 2; border: black solid 1px;
text-align: center; height: 10mm; font-size: 10pt; }
```

```
.statement-header2 { border: black solid 1px; text-align: center;
height: 7mm; font-size: 10pt; }
.statement-row { border: black solid 1px; text-align: center; height:
7mm; font-size: 10pt; }

.due { top: 120mm; left: 10mm; width: 100mm; }
.due .label { width: 70mm; border-bottom: 1px solid black; height: 10mm;
font-size: 10pt; }
.due .amount { border-bottom: 1px solid black; height: 10mm; font-size:
10pt; text-align: right; }
.due .total { font-weight: bold; border-top: 3px solid black; border-
bottom: none }

.rule { top: 160mm; left: 15mm; width: 160mm; height: 1px; }
.rule td { border-top: 1px solid black; }

.totals { top: 160mm; left: 80mm; width: 100mm; }
.totals .label { width: 75mm; border-bottom: 1px solid black; height:
10mm; font-size: 10pt; }
.totals .amount { border-bottom: 1px solid black; height: 10mm; font-
size: 10pt; text-align: right; }
.totals .total { font-weight: bold; border-top: 3px solid black; border-
bottom: none }
.totals .subtotal { border-top: 3px solid black; }

.details { top: 225mm; left: 10mm; width: 170mm; height: 30mm; border:
1px solid black;
    font-size: 10pt; text-align: center; vertical-align: top; padding-
top: 2mm; }

.disclaimer { top: 260mm; left: 10mm; width: 170mm; height: 20mm;
border: 1px solid black;
    font-size: 10pt; text-align: center; vertical-align: top; padding-
top: 2mm; }

.xdrs { width: 200mm; font-size: 9pt; }
.xdrs td { height: 7mm; text-align: left; }

</style>
</head>
<body>

<img class="abs logo" src="[% image_path %]parcel.gif"/>
<div class="abs company">
[% env.companyname | html %]<br>
[% env.addr1 | html %]<br>
[% env.addr2 | html %]<br>
[% env.addr3 | html %]<br>
[% env.addr4 | html %]
</div>

<div class="abs invoice">Invoice</div>

<table class="abs date-num">
<tr>
  <td>Date</td>
  <td>Invoice #</td>
</tr>
<tr>
  <td>[% invoice.issue_date | html %]</td>
  <td>[% invoice.invoice_number | html %]</td>
</tr>
</table>

<table class="abs bill-to">
<tr>
  <td class="bill-to-header">Bill To:</td>
</tr>
<tr>
  <td class="bill-to-body">
```

```
    [% customer.companyname | html %]<br>
    [% customer.salutation | html %] [% customer.firstname | html %] [%
customer.midinit | html %] [% customer.lastname | html %] <br>
    [% customer.baddr1 %]<br>
    [% customer.baddr2 %]<br>
    [% customer.baddr3 %]
  </td>
</tr>
</table>

<table class="abs statement">
<tr>
  <td class="statement-header1">Statement Period</td>
  <td></td>
  <td></td>
</tr>
<tr>
  <td class="statement-header2">From</td>
  <td class="statement-header2">To</td>
  <td class="statement-header2">Terms</td>
  <td class="statement-header2">Due Date</td>
</tr>
<tr>
  <td class="statement-row">[% invoice.period_from | html %]</td>
  <td class="statement-row">[% invoice.period_to | html %]</td>
  <td class="statement-row">Pay in full</td>
  <td class="statement-row">[% invoice.due_date | html %]</td>
</tr>
</table>

<table class="abs due">
<tr>
  <td class="label">Previous Balance</td>
  <td class="amount">[% money(invoice.previous_balance) %] [%
customer.iso_4217 %]</td>
</tr>
<tr>
  <td class="label">Payments</td>
  <td class="amount">[% money(0 - invoice.payments) %] [%
customer.iso_4217 %]</td>
</tr>
<tr>
  <td class="label">Charges this period</td>
  <td class="amount">[% money(invoice.amount_net - invoice.taxes) %] [%
customer.iso_4217 %]</td>
</tr>
<tr>
  <td class="label total">Total due</td>
  <td class="amount total">[% money(invoice.amount_due) %] [%
customer.iso_4217 %]</td>
</tr>
</table>

<table class="abs rule"><tr><td></td></tr></table>

<table class="abs totals">
<tr>
 <td class="label">Session charges</td>
 <td class="amount">[% money(invoice.calls) %] [% customer.iso_4217
%]</td>
</tr>
<tr>
 <td class="label">Subscription charges</td>
 <td class="amount">[% money(invoice.subscriptions) %] [%
customer.iso_4217 %]</td>
</tr>
<tr>
 <td class="label">Credits</td>
 <td class="amount">[% money(invoice.manual_charges) %] [%
customer.iso_4217 %]</td>
```

```
</tr>
<tr>
 <td class="label subtotal">Subtotal</td>
 <td class="amount subtotal">[% money(invoice.amount_net -
invoice.taxes) %] [% customer.iso_4217 %]</td>
</tr>
<tr>
 <td class="label">Taxes</td>
 <td class="amount">[% money(invoice.taxes) %] [% customer.iso_4217
%]</td>
</tr>
<tr>
 <td class="label total">Total</td>
 <td class="amount total">[% money(invoice.amount_net) %] [%
customer.iso_4217 %]</td>
</tr>
</table>

<div class="abs details">Payment Details:<br>Explain how to pay
here</div>

<div class="abs disclaimer">Fine Print / Disclaimer</div>

<div style="page-break-before: always;"></div>

<!-- xdrs -->

<!-- Payments -->
[% SET iter = services.2 -%]
[% SET row = iter.next_row -%]
[% IF row -%]
<table class="xdrs">
<thead>
 <tr>
   <td style="table-column-span: 5;"><b>Payments</b></td>
 </tr>
</thead>
<tbody>
[% SET total_amount = 0 -%]
[% WHILE row -%]
<tr>
  <td>[% row.account_id | html %]</td>
  <td>[% row.destination_description | html %]</td>
  <td>[% row.xdr_cld | html %]</td>
  <td>[% row.xdr_connect_time | html %]</td>
  <td>[% money(row.xdr_charged_amount) %] [% customer.iso_4217 | html
%]</td>
</tr>
  [% SET total_amount = total_amount + money(row.xdr_charged_amount) -%]
  [% row = iter.next_row %]
[% END %]
</tbody>
<tfoot>
 <tr>
   <td style="table-column-span: 4;"><b>Total Payments</b></td>
   <td>[% money(total_amount) %] [% customer.iso_4217 | html %]</td>
 </tr>
</tfoot>
</table>
[% END -%]


<!-- Credits/Adjustments -->
[% SET iter = services.1 -%]
[% SET row = iter.next_row -%]
[% IF row -%]
<table class="xdrs">
<thead>
 <tr>
   <td style="table-column-span: 5;"><b>Credits</b></td>
```

```
  </tr>
</thead>
<tbody>
[% SET total_amount = 0 -%]
[% WHILE row -%]
<tr>
  <td>[% row.account_id | html %]</td>
  <td>[% row.destination_description | html %]</td>
  <td>[% row.xdr_cld | html %]</td>
  <td>[% row.xdr_connect_time | html %]</td>
  <td>[% money(row.xdr_charged_amount) %] [% customer.iso_4217 | html
%]</td>
</tr>
  [% SET total_amount = total_amount + money(row.xdr_charged_amount) -%]
  [% row = iter.next_row %]
[% END %]
</tbody>
<tfoot>
 <tr>
   <td style="table-column-span: 4;"><b>Total Credits</b></td>
   <td>[% money(total_amount) %] [% customer.iso_4217 | html %]</td>
 </tr>
</tfoot>
</table>
[% END -%]


<!-- Subscriptions -->
[% SET iter = services.4 -%]
[% SET row = iter.next_row -%]
[% IF row -%]
<table class="xdrs">
<thead>
 <tr>
   <td style="table-column-span: 6;"><b>Subscriptions</b></td>
 </tr>
</thead>
<tbody>
[% SET total_amount = 0 -%]
[% WHILE row -%]
<tr>
  <td style="width: 30mm;">[% row.account_id | html %]</td>
  <td>[% row.destination_description | html %]</td>
  <td>[% row.xdr_cld | html %]</td>
  <td>[% row.xdr_connect_time | html %]</td>
  <td>[% row.xdr_disconnect_time | html %]</td>
  <td>[% money(row.xdr_charged_amount) %]</td>
</tr>
[% SET total_amount = total_amount + money(row.xdr_charged_amount) -%]
[% row = iter.next_row -%]
[% END -%]
</tbody>
<tfoot>
 <tr>
   <td style="table-column-span: 5;"><b>Total Subscriptions</b></td>
   <td><b>[% money(total_amount) %]</b></td>
 </tr>
</tfoot>
</table>
[% END -%]

<!-- Taxes -->
[% SET iter = services.12 -%]
[% SET row = iter.next_row -%]
[% IF row -%]
<table class="xdrs">
<thead>
 <tr>
   <td style="table-column-span: 6;"><b>Taxes</b></td>
 </tr>
</thead>
```

```
<tbody>
[% SET total_amount = 0 -%]
[% WHILE row -%]
<tr>
  <td>[% row.account_id | html %]</td>
  <td>[% row.destination_description | html %]</td>
  <td>[% row.xdr_cld | html %]</td>
  <td>[% row.xdr_connect_time | html %]</td>
  <td>[% row.xdr_disconnect_time | html %]</td>
  <td>[% money(row.xdr_charged_amount) %] [% customer.iso_4217 | html
%]</td>
</tr>
  [% SET total_amount = total_amount + money(row.xdr_charged_amount) -%]
  [% row = iter.next_row %]
[% END %]
</tbody>
<tfoot>
 <tr>
   <td style="table-column-span: 5;"><b>Total Taxes</b></td>
   <td>[% money(total_amount) %] [% customer.iso_4217 | html %]</td>
 </tr>
</tfoot>
</table>
[% END -%]

<!-- all other... 3, 4, 6, 7, 8, 9, 10, 14, 15 -->
[% FOREACH srv = services -%]
  [% NEXT UNLESS srv -%]
  [% SET i_service = srv.i_service -%]

  [% NEXT IF i_service == 1 -%]
  [% NEXT IF i_service == 2 -%]
  [% NEXT IF i_service == 11 -%]
  [% NEXT IF i_service == 12 -%]
  [% NEXT IF i_service == 100 -%]

    [% SET row = srv.next_row -%]
    [% IF row -%]
    <table class="xdrs">
    <thead>
     <tr>
       <td style="table-column-span: 7;"><b>[% srv.name | html
%]</b></td>
     </tr>
    </thead>
    <tbody>
    [% SET total_amount = 0 -%]
    [% SET total_time = 0 -%]
    [% SET prev_row = {} -%]
    [% WHILE row -%]
      [% IF prev_row AND prev_row.country_name != row.country_name -%]
      <tr>
        <td style="table-column-span: 5;"><b>Total by [%
prev_row.country_name | html -%]</b></td>
        <td>[% total_time %]</td>
        <td>[% total_amount %] [% customer.iso_4217 %]</td>
      </tr>
      <tr>
        <td style="table-column-span: 7;">[% row.country_name | html -
%]</td>
      </tr>
       [% SET total_amount = 0 -%]
       [% SET total_time = 0 -%]
      [% END -%]
    <tr>
      <td>[% row.xdr_cli | html %]</td>
      <td>[% row.xdr_cld | html %]</td>
      <td>[% row.country_name | html -%]</td>
      <td>[% row.destination_description | html %]</td>
      <td>[% row.xdr_connect_time | html %]</td>
```

```
        <td>[% row.xdr_charged_quantity %]</td>
        <td>[% money(row.xdr_charged_amount) %] [% customer.iso_4217 |
html %]</td>
    </tr>
        [% SET total_amount = total_amount + money(row.xdr_charged_amount)
-%]
        [% SET total_time = total_time + row.xdr_charged_quantity -%]
        [% SET prev_row = row -%]
        [% SET row = srv.next_row -%]
    [% END -%]
    [% IF prev_row -%]
    [% END -%]
    [% IF prev_row -%]
      <tr>
        <td style="table-column-span: 5;"><b>Total by [%
prev_row.country_name | html -%]</b></td>
        <td>[% total_time %]</td>
        <td>[% total_amount %] [% customer.iso_4217 %]</td>
      </tr>
    [% END -%]
    </tbody>
    </table>
    [% END -%]

[% END -%]

</body>
</html>
```

APPENDIX C. Advanced Template Customizations

You can use extra tools to customize templates in order to provide information in invoices in the most convenient and suitable for your customers form.

Below you will find a few examples of advanced template customizations (e.g. arrangement of data to exactly match "legacy" invoices.)

Subtotals

Subtotals can be used on large sets of data. For example, you might need to hide a list of all phone calls (as well as details for other services) and show only subtotals for each service.

Description	Cost - GBP (£)
Call charges	0.00
Subscription charges	0.67
TOTAL NET	**0.67**
VAT @ 20%	**0.00**
total gross	0.67

```
<!--BillingTable-->
    <table class="tableFullWidth billTable">
        <tr>
```

```
                    <th>
                        <div class="tableHeader"> Description</div>
                    </th>
                    <th class="even">
                        <div class="tableHeader"> Cost - GBP (&pound;)</div>
                    </th>
                </tr>
                <tr class="height3px"> <td colspan="3"></td> </tr>

<!--BillingTableBody-->
                <tr>
                    <td class="paddingleft10px">Call charges</td>
                    <td class="tableBillBodyRight">
                        [% money( invoice.calls ) %]</td>
                </tr>

                <tr >
                    <td class="paddingleft10px">Subscription charges</td>
                    <td class="tableBillBodyRight">
                        [% money( invoice.subscriptions ) %]</td>
                </tr>
                <tr>
                    <td colspan="3">
                        <div class = "divseparator">
                        </div>
                    </td>
                </tr>
                <tr class="billTableBottomMeasures billTableBottomFont">
                    <td class="paddingleft10px">total net</td>
                    <td class="text-right">
                        [% money( invoice.amount_net ) %]</td>
                </tr >
                <tr class="billTableBottomFont billTableBottomMeasures">
                    <td class="paddingleft10px">vat @ 20%</td>
                    <td class="text-right">[% money( invoice.taxes ) %]</td>
                </tr>
                <tr class="billTableBottomFontBold billTabelTotal
billTableBottomMeasures" >
                    <td class="paddingleft10px">total gross</td>
                    <td class="text-right">[% money( invoice.amount_net +
invoice.taxes ) %]</td>
                </tr>
        </table>
<!--BillingTable_END-->
```

Invoicing summary by country

You may show summary by country to your corporate customers who make many calls abroad.

INVOICING SUMMARY BY COUNTRY			
COUNTRY	**CALLS**	**MINUTES**	**TOTAL COST (£)**
AFGHANISTAN	4	1	£0.18
BAHAMAS	6	2	£0.35
BRAZIL	10	4	£0.08
PERU	8	1	£0.13
SRI LANKA	2	1	£0.02
UKRAINE	8	2	£0.12
UNITED KINGDOM	6	0	£0.09

```
[% SET iter = services.3 -%]
[% SET row = iter.next_row('country_name') -%]
[% IF row -%]

<table class="tableTitle">
        <tr>
                <td>INVOICING SUMMARY BY COUNTRY</td>
        </tr>
    </table>

<table class="tableRows">
        <thead>
            <tr>
                <th>COUNTRY</th>
                <th>CALLS</th>
                <th>MINUTES</th>
                <th>TOTAL COST (&pound;)</th>
            </tr>
        </thead>
        <tbody>
[% SET total_amount = 0 -%]
[% SET total_time = 0 -%]
[% SET prev_row = row -%]
[% SET calls = 0 %]
[% WHILE row -%]
    [% IF prev_row AND prev_row.country_name != row.country_name -%]
            <tr>
                [% IF prev_row.country_name == 'Not Applicable' -%]
                <td class="text-left">INTERNAL</td>
                [% ELSE -%]
                <td class="text-left">[% prev_row.country_name | html -
%]</td>
                [% END -%]
                <td>[% calls %]</td>
                <td>[% time( total_time / 60 ) %]</td>
                <td>&pound;[% money( total_amount ) %]</td>
            </tr>
        [% SET total_amount = 0 -%]
        [% SET total_time = 0 -%]
        [% SET calls = 0 %]
    [% END -%]
    <!--IF end-->
    [% SET total_amount = total_amount + row.xdr_charged_amount -%]
    [% SET total_time = total_time + row.xdr_charged_quantity -%]
    [% SET prev_row = row -%]
    [% SET row = iter.next_row('country_name') -%]
    [% SET calls = calls + 1 %]
[% END -%]
<!-- WHILE end -->
            <tr>
```

```
                      [% IF prev_row.country_name == 'Not Applicable' -%]
                      <td class="text-left">INTERNAL</td>
                      [% ELSE -%]
                      <td class="text-left">[% prev_row.country_name | html -
%]</td>
                      [% END -%]
                      <td>[% calls %]</td>
                      <td>[% time( total_time / 60 ) %]</td>
                      <td>&pound;[% money( total_amount ) %]</td>
               </tr>
          </tbody>
     </table>
<!-- end of the whole table -->
[% END -%]
```

APPENDIX D. Available Email Templates

The following section provides a brief summary of some email templates and their uses.

Template name	Description
account	
reset_password	Sent when a person uses the "Password recovery" feature on the web self-care interface to reset his password. This message contains the new password.
account_generation	
Accounts generation finished and Accounts generation error	Used to compose email reports about the account generation process. It is not recommended to change these.
alarms	
CALL_ERR_REP, CC_STAFF_REP, MERCH_ACC_ERROR, NO_XRATE, PAYMENT_ERR, REPLICATION_ERRO and others	Alerts sent to administrators subscribed to a specific mailing list. **NOTE:** Since most email alerts are triggered in the billing engine, they are configured in the billing engine templates on the master server.
cc_staff	
reset_password	Sent when a person uses the "Password recovery" feature on the web self-care interface to reset his password. This message contains the new password.
customer	
exceeded	Sent when a customer has exceeded his credit limit.
closed	Sent to confirm that a customer's account has been permanently closed.

reset_password	Sent when a person uses the "Password recovery" feature on the web self-care interface to reset his password. This message contains the new password.
send_statistics	Sent when a customer's billing period is closed, to provide him with information about calls during that period. (If the "Full statistics" option has been activated for this customer, an archived CSV file containing all CDRs is attached to the message.)
suspended	Sent when a customer has been suspended because of an overdue invoice.
warning	Sent when a customer's credit limit is over the warning threshold.
invoice	
new_invoice_notify	Sent when a new invoice is created or multiple times **before** the invoice due date to warn the customer that the invoice due date is approaching. Contains a brief summary of the invoice, with the actual invoice attached in PDF form.
overdue_invoice_resend_notify	Sent once or multiple times **after** the invoice due date to warn the customer that the invoice is overdue and service may be suspended soon.
payment	
auto_cc_charge_success	Sent when a credit card on file was successfully charged, contains a confirmation of the credit card transaction.
payment_posted	Notifies a customer that a payment was applied to his account.
periodic_payment_error	Sent to administrators subscribed to the "Payment error" mailing list when execution of periodic payment fails (e.g. credit card is blocked).
representative	
reset_password	Sent when a person uses the "Password recovery" feature on the web self-care interface to reset his password. This message contains the new password.
subscription	

web_subscription	Sent when a potential customer completes the online web signup process.
tariff	
uploaded and other	Used to compose email reports about the tariff upload process. It is not recommended to change these.
user	
reset_password	Sent when a person uses the "Password recovery" feature on the web self-care interface to reset his password. This message contains the new password.
vendor	
reset_password	Sent when a person uses the "Password recovery" feature on the web self-care interface to reset his password. This message contains the new password.