Porta **Control**™

Administrator Guide
Maintenance Release 13

Copyright notice & disclaimers

Copyright © 2000-2006 PortaOne, Inc. All rights reserved

PortaControl Administrator Guide, November 2006
Maintenance Release 13
V1.13.7

Please address your comments and suggestions to: Sales Department, PortaOne, Inc. Suite #400, 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

This document provides PortaControl users with the most common examples and guidelines for setting up a VoIP network. The last section of the document answers the most frequent questions users ask after running PortaControl for the first time.

Where to get the latest version of this guide

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

Caution indicates that the described action might result in program malfunction or data loss.

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 in the paragraph.

Tips provide information that might help you solve a problem.

Hardware and software requirements

Server System Recommendations

- 2 (two) UNIX Servers: for RADIUS and for Web Interface. Both servers host the database, thus increasing system reliability.
- A minimum of 60 GB of available disk space. On the average 200,000 CDRs take up about 1 GB of disk space (this includes database files, indexes and binary logs, raw Radius detail files, billing engine log files and other related information). RAID is recommended to improve performance and reliability.
- A processor running at 2.8 GHz or greater. Additional processor speed is needed for networks with a high call volume.
- At least 512MB of RAM, 1 GB recommended.
- At least one USB port.

For additional details and configuration advice, see the *Hardware Recommendations* topic on our forum:
http://forum.portaone.com/index.php?showtopic=3

For information about whether particular hardware is supported by FreeBSD 6.1 from the JumpStart Installation CD, consult the related document on the FreeBSD website:
http://www.freebsd.org/doc/en_US.ISO8859-1/books/faq/hardware.html

Client System Recommendations

- OS: Windows 95-XP, UNIX or Mac OS X
- Browser: Internet Explorer 6.0 or higher, Netscape 7.2 / Mozilla 1.7 or higher supporting DOM and with JavaScript enabled.
- Spreadsheet processor (MS Excel)
- Display settings:
  - Minimum screen resolution: 1024 x 768
  - Color palette: 16 bit color (minimum)

NOTE: To view downloaded CSV (Comma-Separated Values) files in Windows, please do the following to match the system's default list separator: My Computer -> Control Panel -> Regional Settings -> Number -> List Separator type ",".

Installation

PortaControl is provided via a jump-start installation CD. This CD contains installation media for FreeBSD (6.1-stable branch with the latest security bug fixes), supplementary packages necessary for convenient system administration and maintenance, and PortaControl software packages.

PortaControl installation and configuration are automated and integrated within the main installation process. This allows you to install a completely functional PortaControl environment (two servers) from scratch in less than one hour!

For detailed installation instructions, please refer to the **PortaControl Installation Guide.**

What's new in Maintenance Release 13?

This release includes several new features and improvements:
- Migration to MySQL 5.0 database for improved reliability and performance
- Support for three separate rating periods (peak, first off-peak and second off-peak – e.g. night time and weekends)
- Advanced routing calculation for bilateral traffic exchange models
- Special call routing mode when original DNIS is preserved (required when routing multiple DIDs to IP PBXs such as SPA-9000)
- Ability to apply different rate plans for follow-me and transferred calls
- Many other minor improvements to the web interface

Important upgrade notes

We try to make the process of upgrading as easy as possible, and to keep our releases backward compatible. There are just a few things you should note when upgrading:

- Starting with Maintenance Release 13 PortaControl will use version 5.0 of MySQL database. The PortaControl code will be automatically updated to comply with the new version of MySQL, but if you have any custom-developed applications it is recommended to check whether they are fully compatible with MySQL 5.0. Please see MySQL documentation for more details

.

1. System Concepts

Basic idea

PortaControl is essentially a mediation software. It communicates with
your VoIP network, provides your gateways with authentication or
authorization, and collects data about calls being carried out. It creates the
CDRs (Call Detail Records) for completed calls, giving general
information about each call (such as source and destination number) plus
billing information.
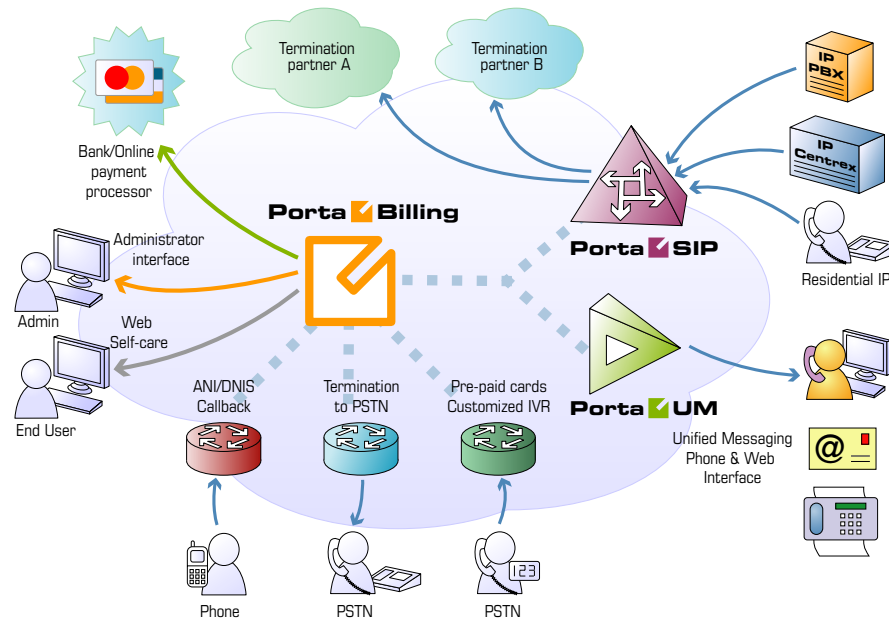
PortaControl

If you already have an existing billing system that charges your customers
for other services (e.g. DSL services), generates invoices and tracks
customer payments, it might be inconvenient to install a parallel billing
system just for VoIP services. At the same time, in order to operate a
VoIP service you need to make use of a significant amount of
provisioning data: VoIP gateways, calling rates, termination costs, routing
structure, call feature availability, and so on. Obviously this data must be
stored and properly managed in some system.

PortaControl provides a solution, allowing you to manage everything you
need to operate your VoIP service without providing any billing-specific
functionalities (such as invoice generation), so as not to interfere with
your current billing system. You will enter various service parameters into
PortaControl, so that your customers can make phone calls. After a call is
made, PortaControl will produce a CDR for it. Later these CDRs will be
imported into your existing billing system, appearing on the customer's

phone bill together with any other applicable charges. So PortaControl does billing only in the context of an individual call.
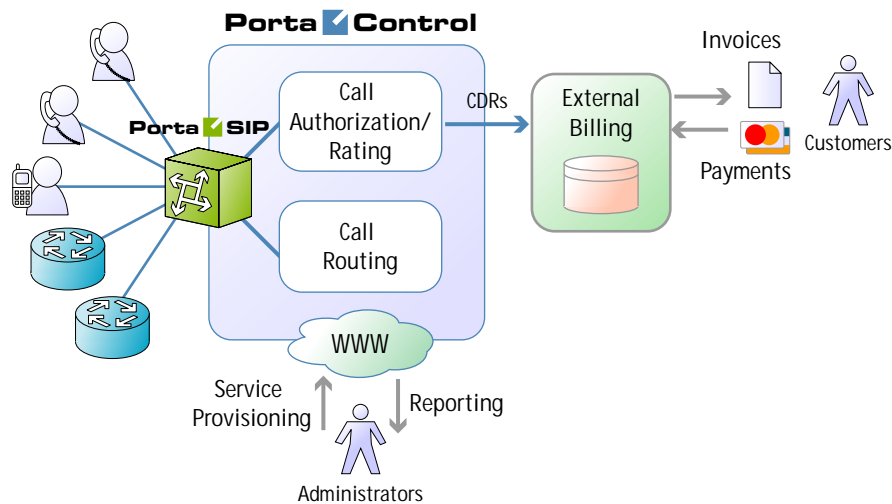
Essentially, PortaControl performs the following functions:
- Service provisioning via web interface (creating customers, end-users, termination carriers, their rates, etc.)
- VoIP user authentication
- Real-time call authorization
- Call routing
- Email alerts to administrator
- Management of advanced call services (e.g. follow-me)
- Call rating (producing CDRs)
- Generating text CDR files for further processing

Call legs

Billing is done based on the information provided by the elements of your VoIP network: gateways, SIP server, and so on. This information is supplied to the billing engine as accounting records. Usually the gateway treats incoming and outgoing calls as two separate entities, i.e. **call legs**. Each of the call legs produces a separate accounting record, so that, for instance, billing engine receives 4 accounting requests in case of a "classic" VoIP call arriving from PSTN to gateway A, traveling over the Internet to gateway B, and terminating in the PSTN network once again. Of course there must be some way of knowing that these four records are actually part of the same logical call. Therefore, each of them includes a special attribute that carries a unique ID for the call. This ID is used by the billing engine to add the call legs together, and is also extremely useful for troubleshooting. For H323 calls, this attribute is called `h323-conf-id`, while for SIP calls it is `call-id` (however, for backward compatibility Cisco gateways and PortaSwitch assign a `h323-conf-id` for every SIP call).

4-tier billing scheme

Four separate entities will be billed for every call registered in the system:

- Account (identified entity using the service)
- Customer (an account owner)
- Reseller (optional)
- Vendors (service providers, e.g. your termination partners or service providers for incoming telephony lines, both local and toll-free)

This allows you to:

- Bill the end user (the owner of a prepaid calling card, phone number or office VoIP gateway)
- Bill the customer to whom the end user belongs
- Bill the reseller who owns that customer (if applicable)
- Calculate your expenses for termination of this call to the vendor

System architecture
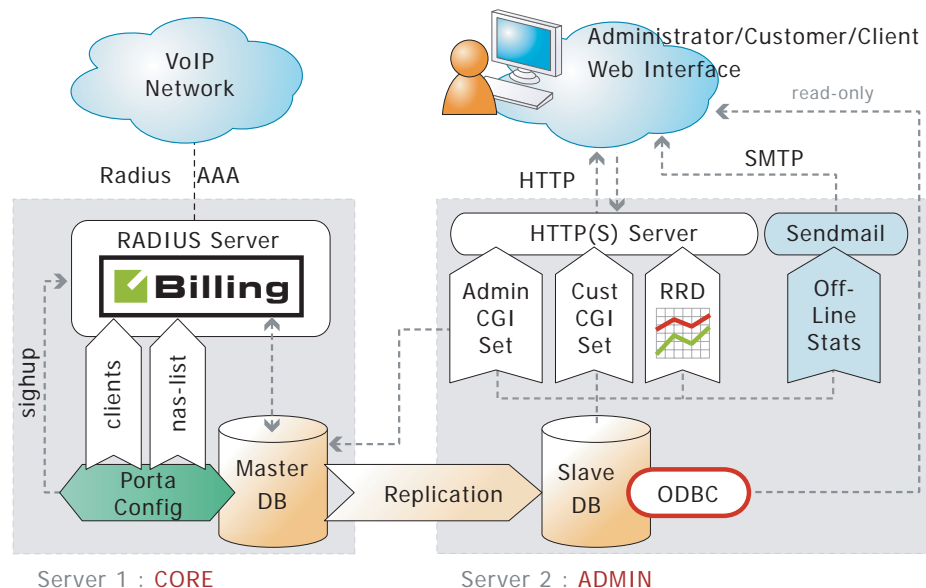
PortaControl consists of two main logical components: the master (or core) server (left) and the slave (or admin, or web) server (right). Typically, the master and slave servers are installed on two separate physical computers to provide data redundancy and load sharing. Although an "all in one" combination (where both servers run on a single physical computer) is also possible and supported, this is recommended only for evaluation or demo systems.

Master server

The master server hosts the master database and radius daemon. Radius has a billing engine embedded into it, and so after it receives a request from the gateway it passes processing on to the billing engine and sends back a response.

Billing engine

The billing engine **provides:**
- Authentication – tells the gateway which PIN, IP or the like is allowed to use services, and returns attributes such as current balance.
- Authorization – tells the gateway which PIN, IP or the like is allowed to call a particular number, and returns attributes such as maximum allowed call duration.
- Account processing – based on the information from the gateways, this bills the call and writes CDRs to the database.

The billing engine also keeps a cache of current calls in its memory. This allows PortaControl to:
- Detect and prevent multiple logins for the same debit account.
- Combine information from different gateways, gatekeepers and proxies in order to obtain complete information about a call.
- Reuse information from earlier steps in order to boost performance.
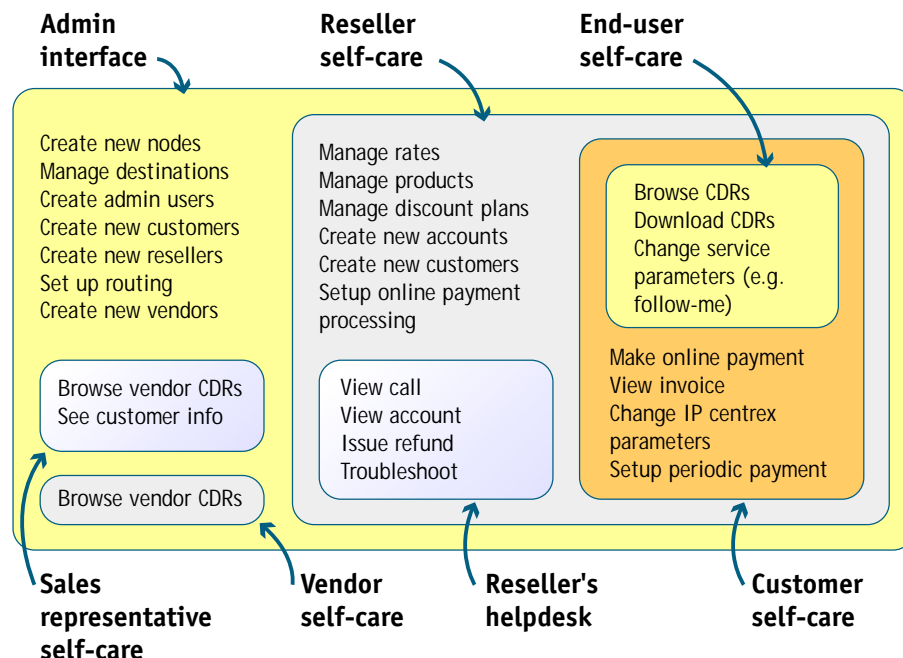
Slave server

The slave server hosts the following:
- Web interface, consisting of:
  - Admin interface
  - Account self-care
  - Customer self-care
  - Reseller's helpdesk (self-care; customer care)
  - Vendor self-care
  - Representative self-care
  - Online web signup interface
- Slave database
- Scripts for calculating various statistics
- Optional IVR (TFTP) server

However, should your co-location provider be having a problem with
network connectivity, both servers will become inaccessible to your VoIP
network installed elsewhere. This can be avoided by installing the standby
server in a different physical location, connected to a different network
(and preferably to a different ISP). In that case, even if the whole hosting
center including your main server is down, you can still continue network
operations via the standby server, since this will not be affected by the
outage.

PortaControl web interface

Different operations are available for different types of users who access
the system (also, the abilities of an individual user can be adjusted via his
access level settings). In general, administrators (PortaControl users) can
potentially perform all possible configuration tasks, while others are
limited to their own entity and related objects only. For instance, a reseller
can access any of his sub-customers or their accounts, while a sub-
customer can only access information about himself and his accounts on
his self-care pages. The self-care interface imposes multiple security-
related limitations on a user's activity, e.g. a sub-customer can view his
own balance on the self-care pages, but cannot modify it.

Admin interface   **Reseller self-care**   **End-user self-care**

Create new nodes
Manage destinations
Create admin users
Create new customers
Create new resellers
Set up routing
Create new vendors

Manage rates
Manage products
Manage discount plans
Create new accounts
Create new customers
Setup online payment
processing

Browse CDRs
Download CDRs
Change service
parameters (e.g.
follow-me)

Browse vendor CDRs
See customer info

View call
View account
Issue refund
Troubleshoot

Make online payment
View invoice
Change IP centrex
parameters
Setup periodic payment

Browse vendor CDRs

Sales representative self-care   **Vendor self-care**   **Reseller's helpdesk**   **Customer self-care**

For improved security, the PortaControl web interface is divided into
several separate areas, each serving its own purpose. Thus a login name
and password which allow a user to access the customer self-care interface
cannot be used to log in to the admin interface; the login name and

password for the account's (end-user's) self-care interface are not
applicable to customer self-care; and so forth.

Account and account types

An account identifies the end-user who is using the service. For example,
a prepaid calling card is an account identified by a PIN, while a customer's
VoIP gateway could be an account identified by its IP address, and a SIP
phone is identified by its phone number. There are two main types of
accounts (debit and credit) plus an auxiliary one (voucher).

Debit

Typically used for: prepaid calling cards.

- The balance shows the available funds on the account. Initially,
  the balance is equal to the "opening balance" (typically the prepaid
  amount), and decreases with every call made.
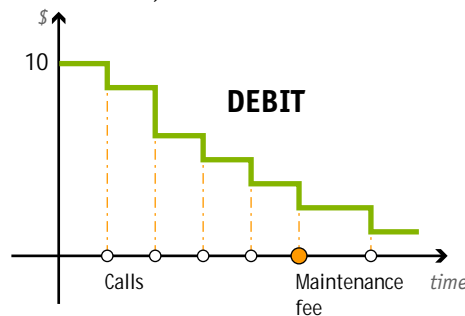
- The account is unusable when the balance hits zero.
- A fraud protection feature prevents simultaneous logins to the
  same debit account.

Credit

Typically used for: postpaid services.

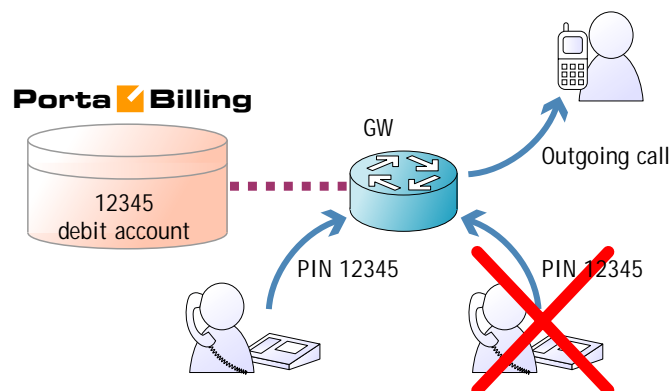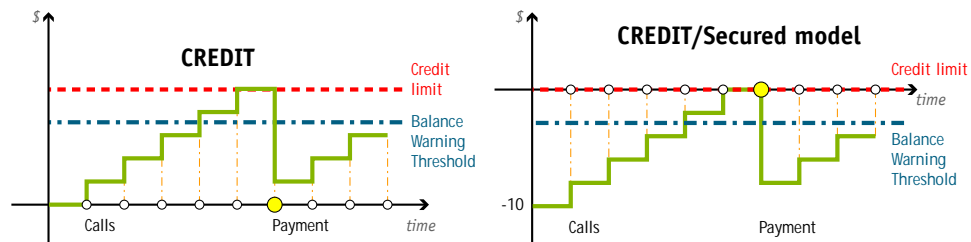- The balance reflects how much the account owner owes you. It starts with the "opening balance" and goes up with every call made, and decreases when a payment is received.

- The account is unusable when the balance reaches the "credit limit".
- The balance can also be negative – this means you owe money to the customer (e.g. the customer has made a deposit). By setting the credit limit to zero you can provide "secured" services (as described earlier), in which case a customer can only make calls when there is a deposit remaining in his account.
- Unlimited number of simultaneous logins.

Voucher

Typically used for: refill debit or credit accounts for a customer using the same account over a long time, e.g. ANI-based authentication.

This can be used to "recharge" (increase the available amount of) a credit or debit account. In this case, the balance of the voucher account is transferred to the main account. An account of this type cannot be used for making calls. Recharging may be done on the account self-care pages or via a special IVR script.

NOTE: The default Cisco debit card application does not support recharge vouchers, so you must add this functionality to the IVR yourself, or purchase PortaOne's "Voucher recharge" script from store.portaone.com. You may also use the voucher recharge IVR provided with PortaUM.

Account batches

Accounts can be grouped into batches. Each batch has its own descriptive name, giving you two ways to identify an account:

- By **Account ID**
- By a combination of **Batch Name** and the account's **Control Number** (a sequential number within the batch)

When creating new accounts, you can either create them in a new batch or add them to an existing batch. In addition to better account tracking, this also permits easy modifications of a large number of accounts. You can

block or unblock, change the balance, or perform other operations for either the whole batch or a portion of the batch. Note that a batch always belongs to a customer, so you cannot mix different customers' accounts within the same batch, and you cannot have two batches of the same name under two different customers.

Account life cycle

Three parameters define the life cycle of an account:

- Every account has an **activation date**. This defines the earliest date of activation (e.g. you print Christmas promotion cards in November, but set the activation date for December 23rd, so that no one can start using them earlier).
- **Expiration date** defines the date after which the account can no longer be used (e.g. your Christmas promotion cards expire on January 1st). Expiration date is optional, so you may also create accounts which never expire.
- Finally, there is the **lifetime** parameter, which defines the number of days the account remains active after its date of first use. If, for instance, an account has a lifetime of 90 days and is first used on May 1st, then it cannot be used after July 30th. Lifetime is also an optional parameter. If you wish to allow your customers to use the service perpetually, leave this parameter blank.

  Lifetime and expiration date work in conjunction – whichever comes earlier defines the end of the service. If an account has an expiration date of 01-Jan-2007 and a lifetime of 60 days, and is first used on 09-Dec-2006, then the account has only 23 days of service left, and will expire on January 1st.
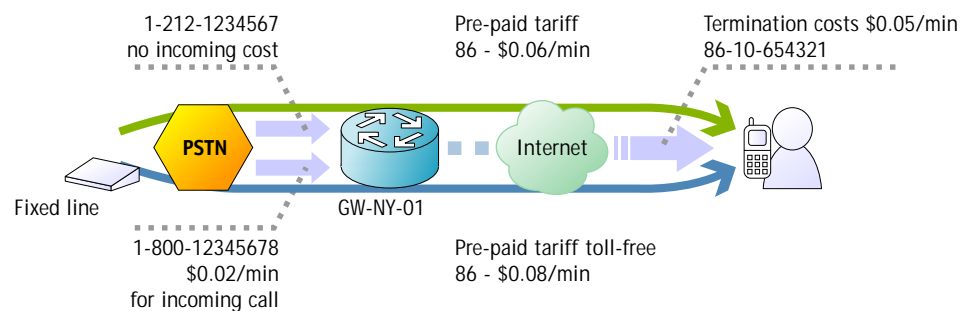
  Another important point regarding lifetime is that a voucher recharge operation extends the life cycle of an account. Thus, if an account has a 60-day lifetime, and the account owner makes a voucher recharge 57 days after the date of first use, service will be extended for another 60 days, with the date of the last recharge as the starting date of a new lifetime period.

Products and accessibility

A product is a combination of services that you provide to a customer for a price. For example, you decide to sell calling cards with 6 cents/minute calls to the China for calls to a local access number in New York, and 8 cents/minute for calls to China made via a toll-free line. In this case, your product will include two types of service:

- access via the local New York number, and
- access via the toll-free line

with price parameters associated with each service.

1-212-1234567   Pre-paid tariff   Termination costs $0.05/min
no incoming cost   86 - $0.06/min   86-10-654321

PSTN   GW-NY-01   Internet

Fixed line

1-800-12345678   Pre-paid tariff toll-free
$0.02/min   86 - $0.08/min
for incoming call

Maintenance fees

Maintenance is a property of the product which allows you to charge your customers periodic fees. A maintenance fee is applied to all active accounts (accounts which have been used at least once) for a specific product. A maintenance fee is applied for the first time on the day after an account is used for the first time, and at periodic intervals thereafter. If you change the maintenance fee for a product, this will affect maintenance fees that are charged in the future (but not past ones). It is possible to temporarily disable maintenance fees and then start charging them again from a certain date.

NOTE: Maintenance fees remain an important tool in the prepaid card business model. For subscription fees in a postpaid scenario, a new method is available in Maintenance Release 12: subscriptions. Please see below for more details.
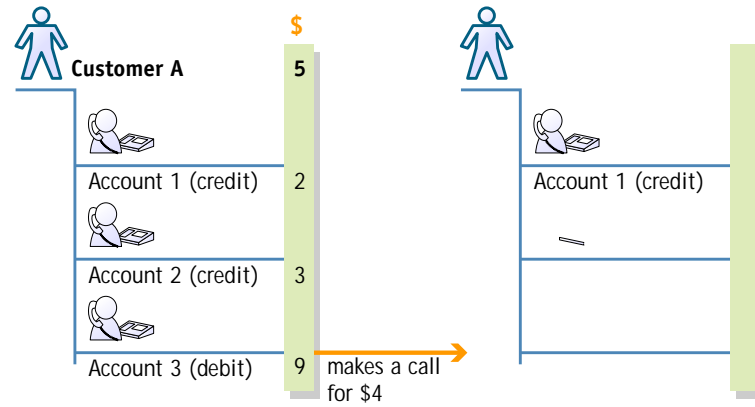
Customers

Retail customers

The customer is an owner of accounts with whom you interact in your business. If, for example, you are selling thousands of prepaid cards (each card being represented in PortaControl as an account), you do not have to create a customer for each of the cards. Rather, the cards will be owned by the customer, who represents the dealer or distributor of the cards. You can have many such customers if you use multiple distributors. Even if you do not use distributors at all, and sell all of your cards directly to end users, it is still a good idea to create a separate customer object for this purpose, so that these accounts will not be mixed with others.
For each call made by a credit account, the customer's balance is charged for exactly the same amount by which the account was billed (retail customer accounts must use the same currency as the customer). This provides an easy way to track the total balance for a group of postpaid accounts.
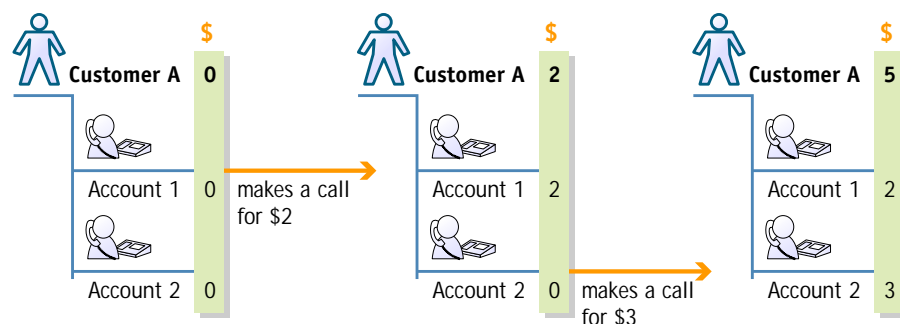
NOTE: If a call was made by a debit (prepaid) account, this does not affect the retail customer's balance (see below), since prepaid accounts (cards) are normally paid for by your customer at the time of purchase. This also allows your prepaid card

distributor to have postpaid accounts (for example, for internal office use) and to be billed for them. Or, your postpaid customer with multiple IP phones can purchase additional prepaid cards for his temporary workers.

reseller's name and logo on these pages, etc. Reseller collects money from his sub-customers and is responsible for paying invoices, issued by the PortaControl owner. If reseller exceeds his credit limit – this automatically blocks all activities of his sub-customers.
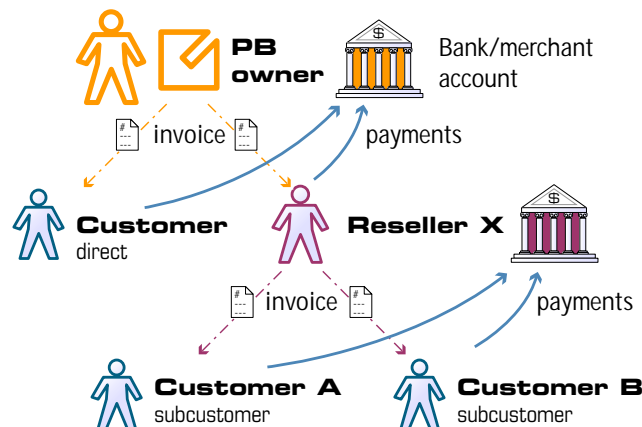
Typically used for: Service resellers, customers who would like to offer your service under their own brand name, while the actual service is provided on your VoIP network and all the billing of their accounts using different tariffs is done in PortaControl

NOTE: Billing via the reseller model implies an additional load on the billing engine, compared to a retail customer. If performance is an issue, avoid creating resellers when this is not necessary.

Both retail customers and resellers are billed according to the credit model. Their balance reflects the amount they owe you; thus it starts from a certain value (typically 0) and goes up with each call made, or down with each refund or payment. If the customer's balance reaches the maximum credit amount (credit limit), no further calls for this customer's accounts will be authorized. The only exception are a retail customer's debit accounts – since these are prepaid, and their calls do not affect the customer's balance, they can still make calls even if the customer has exceeded his credit limit.

Summary of customer types

The following are all types of customers (all of whom have the same attributes, such as balance or address info):

- Reseller – somebody who resells your services.
- Direct customer – a customer who directly communicates with your company, i.e. receives bills from you and pays to your accounts. A direct customer can be either a reseller or a retail customer.
- Sub-customer – a retail customer who is owned by a reseller.

Customer status

A customer may have the following status:

Status	Description
🔺 **Credit exceeded**	This indicates that the customer's balance is above his credit limit, so he will not be able to make any outgoing calls, unless these are free calls (e.g. toll-free numbers).
❄ **Payment frozen**	Similar to the item above, this indicates that the customer's periodical payment has been suspended due to repeated errors (e.g. credit card cancellation).
⛔ **Blocked**	The customer has been blocked by the administrator; no call services will be provided until the administrator removes the block. The customer still has access to his self-care pages. Blocked customers are not subject to maintenance charges, but subscription fees still apply.
🔧 **Suspended**	Services for this customer have been suspended because of an overdue invoice. Once payment is received (whether the customer makes an online payment, or a periodical payment is applied, or the administrator enters a cash/check payment manually) in an amount covering the overdue invoice, the suspension is automatically lifted.
🔒 **Closed**	The customer's account has been closed, and is kept in the database only for informational/regulatory purposes. No further operations are possible with this entity.

Customer classes

Customer classes allow you to define a set of parameters which will be shared among a certain category of customers. For instance, suppose your invoice term for retail customers is "Net 21 days", while for business customers it is "Net 30 days". If your operators enter these values manually for each customer, there will inevitably be mistakes. Instead, you can define two separate customer classes, one named "Retail" and the other "Business", and define these parameters within them. After that, your operators need only assign a specific class to a given customer in order for the customer to automatically inherit all the class properties (grace period, invoice template, and so on).

Vendors

Vendors are your service providers, for example termination partners or incoming toll-free line providers. Every time a call travels from your network to a vendor (via telephony or VoIP), a cost is associated with it. So at that point PortaControl will charge the account and customer for the call, and also calculate your termination costs.

Connections to a vendor

Connections define points where calls travel from your network to a vendor. (Unless it is a direct call between two users on your network, there is always a connection to a vendor.) For connections via IP, we need to know the IP of the remote node (where the call is being sent to), while for telephony we need to know the gateway and, optionally, the name of the port the phone line is connected to. A connection defines a termination cost, i.e. the tariff according to which a termination partner charges you.

Connections from a vendor

It may happen in your business that you incur costs not only when terminating calls, but also when a carrier delivers calls designated for one of your customers to your network. Typical examples of situations in which such charges occur include the following:
1. You purchase a toll-free number from a carrier and interconnect with it via E1/T1. When a customer dials this number, the call is delivered to your gateway, and you can then provide a service (e.g. a prepaid card IVR) to the customer.
2. You purchase a set of phone numbers (DIDs) to be distributed to your customer's IP phones. When a number is dialed anywhere in the PSTN world, it will travel to the carrier who owns these numbers. The carrier will then forward the call either directly to your SIP server via an IP, or to your VoIP gateway via a PSTN trunk.

In both cases, you will be charged by the carrier who provides you with the service according to call duration, and these expenses must be reflected in PortaControl in order to show your cost/revenue figures correctly.

NOTE: Account and customer are billed **only** for outgoing connections. Connections from a vendor are purely for calculating costs. There is one exception, however: see "Charge reseller for incoming calls" in the "How to..." section
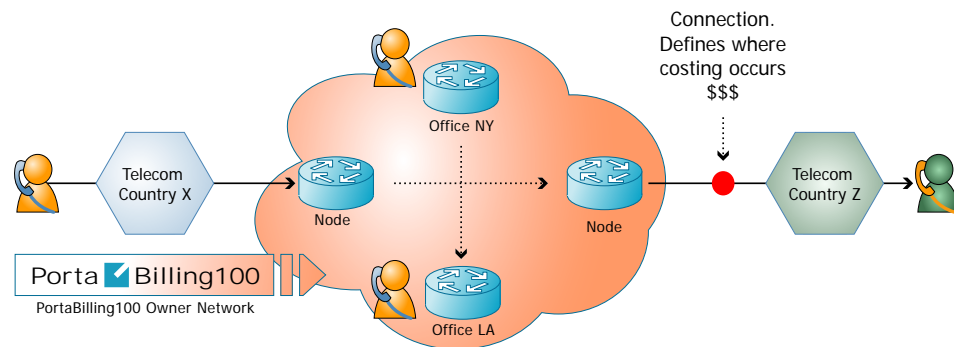
Figure 1-1

Connection load

PortaControl can build load graphs for every connection, so that you can monitor average load and quality parameters (e.g. ASR) for each one.

Account ID-based billing

Modern business involves diverse types of services, which require different types of billing. Calls made by prepaid cards are billed by PIN number, calls from a customer's GW should be billed by IP address, some wholesale customers are identified only by the port they are connected to, and so on.

It is easy to find out that a call has been made. The important matter is to correctly determine who made the call and, therefore, who should be charged for it. PortaControl uses a simple, flexible yet powerful method to determine a call's owner. Identification of the account (Account ID) should be sent to the billing system in the **User-Name** attribute.

For example, in the case of prepaid calling cards, User-Name will contain the PIN, IP address for IP-based billing, and so on. Of course, an account with such an account ID must exist in the database. This way of utilizing the User-Name attribute is actually default behavior for Cisco call handling applications, and the latest firmware releases of Quintum should do the same.

Due to such an approach, logic in the billing is independent from service-specific issues. You are not limited to any pre-defined billing schemes in PortaControl; rather, you can design your own. Today you can bill by PIN or IP address, and tomorrow you will additionally be able to bill by h323-id. All you need to do is pick the proper application to handle the call, so that a correct identification will be sent in the User-Name.

Unfortunately, for some types of service User-Name is not sent to the billing properly on Quintum gateways. In this case, it is possible to employ username replacement rules on the billing side to emulate this. See the relevant section in the "How to:" chapter for more details.

Nodes

Nodes are your gateways for carrying out calls. Perhaps the most important feature of a node is that it can ask the billing for authentication or authorization and send accounting information to the billing system. It is very important that we exchange AAA data **only** with trusted gateways, also called **trusted nodes.** Usually, if this gateway is not owned by your company, but rather by your partner or customer, it will not be considered trusted, and so will not be created in PortaControl as a node.

Node ID, NAS IP address, and Radius source IP

What is the difference between these terms?
- **Radius Source IP** address is the address that radius requests (UDP packets) come from. Radius will accept requests only from those IP addresses which are listed as **Radius source IP** for one of the nodes. This is used as protection against denial-of-service attacks and attempts to send fraudulent information to the billing system.
- **NAS-IP-Address** is the IP address gateway used for VoIP purposes. This address will be present in the radius data as the **NAS-IP-Address** attribute, and is used to identify which node sent the request. Usually **Radius Source IP** is the same as **NAS-IP-Address**, except in a situation when your gateway has two network interfaces, using one (internal) to communicate with the billing system and the other (external) for VoIP traffic.
- All of the incoming VoIP calls should be authenticated (see **Remote IP authentication** below). This also applies when a call comes from your gateway A to your gateway B. When node B (the terminating node) consults PortaControl as to whether the call should be allowed, it will send an authentication request to PortaControl with a User-Name containing some identification of the remote gateway. What information is used for identification depends on the application, but typically this is the IP address of the remote gateway. PortaControl will check if there is a node with such a **Node ID**, and, if so, the call will be allowed.

NOTE: You can only specify your own custom **Node ID** if you have enabled this advanced feature. Normally this is not required, and by default **Node ID** equals the **NAS-IP-Address**.

In most cases, all three parameters (NAS-IP-Address, Radius Source IP and Node IP) are the same.

Remote IP authentication

When your gateway is installed and connected to the Internet via the network interface, it can accept incoming VoIP calls and send them onward. This is fine, assuming that calls are coming in from one of your customers; but potentially *anyone* on the Internet can send you traffic if they know your gateway's IP address. This could happen due to a mistake, or it could be done intentionally; in any case, it is not good for your business if this happens. How to prevent this and make sure that only your customers and yourself are able to use your VoIP network?

There are a few methods which enable you to do this:

1. Set up an access list (ACL) on your gateways so that VoIP traffic will be allowed **only** from certain IP addresses/subnets. This is a very reliable method. Unfortunately it has one drawback: it is difficult to maintain, since everything is configured on the individual gateways. For example, if you have 12 gateways and you wish to add a new customer, you have to change the ACL 12 times.
2. Set up an application to handle incoming VoIP calls. This application will authenticate the remote party (by IP address or some other parameter) in PortaControl and, if authentication is successful, will connect the call. There are several advantages to this approach:
    o Information about authorized gateways is stored in one place, so it is easy to maintain.
    o The application can include data required for IP-based billing in the call information (so the User-Name attribute will contain an Account ID).
    o It may be combined with authorization, so that you can also check if a remote gateway is allowed to call a particular number.
    o You can use more sophisticated ways of identifying a remote gateway. For example, instead of IP address you can use an IP+MAC address combination.
3. An improved version of the previous item is a solution in which all calls to your network arrive to a proxy (such as PortaSIP) which will take care of call authorization, while your gateways will only accept incoming calls from that proxy. Again, the easiest way to implement this is by using IP authentication on both the proxy and gateways.

We assume that you will use either method 2 or 3, i.e. your terminating gateway will authenticate incoming VoIP calls (to check that the remote

gateway is allowed to send you traffic). This may be done by sending an authentication request to PortaControl, where the User-Name contains an identification of the remote gateway; typically this is an IP address. You can use the Cisco application remote_ip_authenticate.1.1.1.tcl provided as part of the TCL Ware package, or you can purchase an improved version called **Advanced Remote Authenticate** from PortaOne, Inc.

Virtual environments

It is possible to have multiple isolated PortaControl instances, or virtual environments, within the same PortaControl installation. This functionality allows you to use PortaControl as an ASP (Application Service Provider) platform, or to create test environments which do not affect billing production activities.

Each PortaControl virtual environment contains a complete set of data specific to that environment. A user who logs in to a user account for a specific environment will see only the users, customers, tariffs, accounts, CDRs, statistics, and so on for that environment. There are a few restrictions on the use of virtual environments to ensure their correct functioning:

- The login name for Admin users must be unique across all environments. For example, you may not have a user 'jack' in both environment ABC-Production and environment ABC-Test. Rather, two users will have to be created, possibly 'abc-jack' and 'test-jack'. The same applies to login names used by your customers/accounts/vendors on their self-care pages.
- A gateway (node) may be registered to only one environment. This is controlled via the NAS-IP-Address, so you cannot have a node with the same IP address in two environments.

You may have multiple customer, vendor or account entries with the same name across environments. For example, you may have an account '12345' in both 'ABC-Production' and 'ABC-Test' (from the previous example). In this case, the decision as to which account (from which environment) is determined by which gateway sends the accounting record (i.e. one registered in ABC-Production or in ABC-Test environment).

You may create new environments or delete them using the `porta-admin.pl` script on the master server in the home directory of the `porta-billing` user. To do this, just run the following commands on the master server:

```
cd ~porta-billing
./porta-admin.pl create <environment-name>
```

A new environment is created empty, except for a single user account which you will use to login and then create other users, tariffs, customers, and so on. By default, the username of this user is *<env>*-root and the password is the same, where <env> is the name of the environment you have just created. So if you execute the command "./porta-admin.pl create abc", a new environment abc will be created, and to login to it you will use username abc-root and password abc-root.

Destinations, rates and tariffs

Destinations, rates and tariffs are the essential parameters which define how a call should be billed. The call is billed according to the destination number (Called-Station-Id, or CLD, or DNIS).

Destinations

Destinations are a list of all possible phone number prefixes to be used in your system. You generally need a new phone prefix when you have a new service area for calls which are to be treated differently than others. For example, if you start providing calls to the Czech Republic, you should add destination 420 and specify it as Czech Republic. Later, if you plan to charge calls to Prague differently than calls to the rest of the Czech Republic, you might need to add another destination with the phone number prefix 4202. All such destinations have to be entered into the system before you can use these prefixes. This prevents errors and helps you to improve data quality. It is recommended that all of your prefixes be defined in the E.164 format.

It is virtually impossible to have an "ultimate" destination list that would contain all prefixes for the entire world. First of all, it is quite difficult to gather and maintain such information. Second, and even more important, is the fact that having "all possible" prefixes would not offer us any real benefits, and would only make rate maintenance more difficult. For instance, if vendor A provides you with a rate of 0.10/min for calls to 420 (Czech Republic) and a 0.18 rate for calls to 420602 and 420603, you will need to create three prefixes: 420, 420602 and 420603. If you also create the prefix 4205 (Czech Republic, Brno) it will only take up space in the database, as it will not ever actually be used.

Therefore, we recommend that you follow a "required minimum" approach: only create those destinations for which you will need specific rates for your carriers or customers.
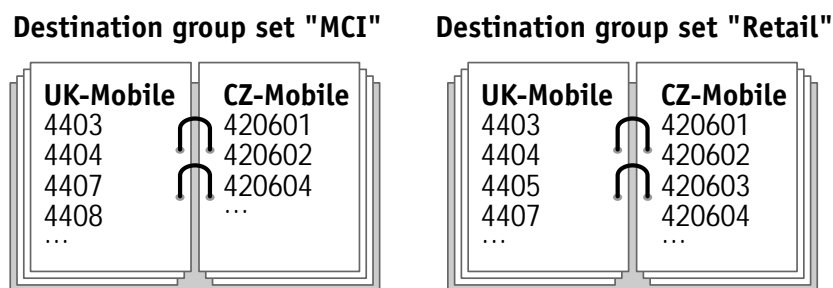
Destination groups

Sometimes you will have several prefixes for the same target destination, e.g. 420602, 420603 and 420737 for mobile numbers in the Czech

Republic. All of these prefixes can be defined as the single destination group "CZ Mobile", so when you enter a new rate for this destination group it will, in effect, create rates for each individual prefix.

Destination group sets

In happens quite often that different partners will assign a different meaning to the same destination groups. For instance, your partner A says that "CZ Mobile" is 420602, 420603 and 420737, but your partner B also considers prefix 420777 to be part of "CZ Mobile". So you will need to have two versions of "CZ Mobile" – one for partner A and one for partner B. In this situation, you will create two separate destination group sets ("A" and "B"), as well as different destination groups inside those sets.

Destination group set "MCI"   **Destination group set "Retail"**

UK-Mobile	CZ-Mobile		UK-Mobile	CZ-Mobile
4403	420601		4403	420601
4404	420602		4404	420602
4407	420604		4405	420603
4408	…		4407	420604
…			…	…

You can imagine each destination group set as a binder, with every page in that binder describing a certain destination group.

Rates

A rate is a combination of billing parameters for a specific destination. For example, you can specify that calls to 420 are charged 0.09 USD/min during peak time and 0.07 USD/min during off-peak time, while calls to 420609 should not be allowed at all

Tariffs

A tariff is a complete set of rates for a specific account, customer or vendor. Thus it should include every possible destination to which you want to let him call.

It may be that the tariff will contain some prefixes that are part of more generic ones (for example, you will have a rate for the **420** prefix and for the **4202** prefix). In this case, the longest prefix match takes priority. What happens if a destination is not included in the tariff, and the customer tries to call there? There are two possibilities:
  o   If outgoing calls are authorized via PortaControl (for example, in the prepaid card's IVR) the customer will not be authorized to call this destination.

> o   If there is no authorization, the customer will actually be able to make a call, but PortaControl will not be able to bill it properly, since required information is missing. An email alert will be sent and a special CDR will be written into the database, so you will still have an overview of this call. It is highly recommended that you always use authorization of calls via PortaControl.

Relationship between destinations, rates and tariffs

Let's move from VoIP to a simpler scenario. Imagine you are the owner of an office supply store. You have to manage your inventory and price lists for customers and resellers.

- First of all, you must create a catalog of all the items you intend to sell. This is your internal document containing entries such as "20001 – ball-point pen, blue; 20002 – ball-point pen; black; 50345 – stapler;" and so on. Note that this is just a description of the items, without prices, since the price will depend on the specific vendor/customer.

- You will receive price lists from your suppliers and convert them into data files for every vendor, such as "Office Depot: 20001 – $0.35; 20002 – $0.35; etc.". Note two things: there is no need to include an item description in every file – you can always extract this from the catalog of items you have created above. Also, each data file will contain only some of the items, i.e. those which are provided by this particular supplier.

- The next step is to create similar price lists, but with the prices you will apply to your customers. Of course, you might have several such price lists: e.g. one for retail customers, another for business customers, another for resellers, and so on. Every data file will contain all of the items you offer to this category of customers, including prices.

Now let's come back to the VoIP business:
- *The catalog of items* corresponds to **destinations** in PortaControl. Every destination is similar to an item description in the catalog, i.e. it provides general information (e.g. 420 – Czech Republic proper, 420602 – Czech Republic mobile, and so on).

- *The price list (for vendor or customer)* is equal to a **tariff** in PortaControl. The price list includes all items the customer may purchase and all destinations the customer can make calls to. All tariffs are identical in structure, but some of them will be used to calculate your costs (vendor tariffs), while others will be used to charge your customers.

- *A single line item in the price list* is equivalent to a **rate** in PortaControl. It gives the price per minute and other rating parameters for a specific destination.

Therefore, the standard sequence for setting up your service is as follows:
- Define the destinations you will use in your business. Basically you will need to define every unique prefix used by your vendors (an easy way to do this is by using PortaControl's default destination set and PortaControl templates).
- Create a tariff for every vendor using the rate list they supply.
- Create a tariff for every customer/product. Once again, note that each tariff may contain a different set of destinations. For instance, the tariff for your vendor ABC may contain different rates for 420 (0.10/min), 4202 (0.09/min) and 420602 (0.18/min). But you can just list a single rate for your customer in the tariff, i.e. 420 – 0.20/min.

Relationship between tariffs and destination group sets

As mentioned before, multiple destination group sets may be defined in the system, with some destination groups (e.g. CZ-Mobile) defined for each of them. So, when you enter a new rate for CZ-Mobile in a specific tariff, which destination group definition should be used? To avoid such ambiguities, you will assign a destination group set for every tariff, which will be used to create new rates (if you do not assign a destination group set to the tariff, then you can only enter rates for individual prefixes). Thus when you attempt to create a new rate for destination group CZ-Mobile, the following sequence of events takes place:
- PortaControl locates the definition of the CZ-Mobile destination group in the destination group set associated with this tariff.
- For every destination included in this destination group, the new rate is inserted.

In other words, the result is the same as if you had created multiple rate entries manually. Rates which are created using a destination group do not differ in any way from rates created for a single destination. One important consequence of the foregoing is that if you change the destinations included in a destination group, it will not affect the previously created rates. Thus if you had 420602 and 420603 in the CZ-Mobile destination group, and you now add 420737, this will not affect any of the tariffs. In order to have correct billing for the 420737 prefix, you must go to the corresponding tariff and add a new rate for the CZ-Mobile prefix.

Call billing parameters

Cost-based and volume-based billing

The traditional method of billing is cost-based billing. This means that, once a call is completed, the billing engine calculates the charges for the call based on price parameters for the call destination and duration. This amount is then applied to the account, customer or vendor balance. After Maintenance Release 8, PortaControl supported volume-based tariffs. This meant that you could sell traffic to your customers in volume lots (e.g. "10,000 minutes to UK-London, 5,000 minutes to UK-Manchester") and the system would track the total call duration to the specific destination, regardless of the call cost. However, the volume discount plans which were introduced in Maintenance Release 10 allow you to accomplish the same and much more, so that volume based tariffs are now discontinued.

Cost-based billing

Peak and off-peak prices

It is possible to have two different sets of prices – for peak and off-peak time. Off-peak periods are defined using the powerful and flexible Time::Period module. An Off-peak Period Wizard is also available, allowing you to perform period definition the easy way. See the **PortaControl Web Reference Guide** manual for more details.

The fundamental concept here is the period definition, which specifies a certain interval in time (typically a repeating one). In each period you can specify conditions for:
- Time of day
- Day of week
- Day of month
- Month

If you do not specify a condition for a certain element (e.g. for **Month**) then it is considered that there are no limitations on this element, e.g. "from 8pm to 8am" will apply in the same fashion throughout the year. If you define conditions for several elements, these must be satisfied at the same time, e.g. "from 8pm to 8am, Monday-Friday" means that an event fits into this interval if it happens between 8pm and 8am **and** from Monday through Friday, e.g. 6am Saturday is excluded.
An off-peak period is a conjunction of several definitions (in the off-peak period description, you can even see that they are joined by a logical **OR** statement). Thus a given moment is considered to be within the off-peak period if it satisfies at least one of them. For instance, for the period "from 8pm to 8am, Monday-Friday OR Saturday-Sunday" both 6am Saturday and 6am Friday are included.

By default, everything is considered to be peak time. You may use one of the three available options to determine whether a peak or off-peak rate should be used. When defining an off-peak period, you may choose for off-peak rates to be applied if:

- a call starts during off-peak time
- a call finishes during off-peak time
- a call both starts and finishes during off-peak time (the least flexible option)

Staring with Maintenance Release 13, there is an extra feature related to off-peak period usage. By enabling the `MultiOffPeakPeriod` option in the `[Tariff]` section of `porta-admin.conf` you can define two independent off-peak periods (e.g. night time and weekend time). In order to utilize these two periods, select the special **pricer** element in the rating formula as **price per minute** instead of **price_first**/**price_next**. PortaControl will then check whether the call fits into the main period; in this case, **pricer** will be set to the **Off-peak price next** value. If the call does not fit into the main period, then the alternative period will be checked; if the call fits into this period, **pricer** will be assigned the **Off-peak price first** value. Otherwise, peak rates are applied, and **pricer** will be populated with the value from **Price_next**.

So, for instance, in order to use the 0.10 rate during peak hours, 0.08 during weekends and 0.06 during night time, you should specify the main off-peak period as "Saturday and Sunday" and the alternative one as "8pm-8am", construct a formula such as "N intervals of 60 seconds at pricer USD/min", and enter the following in the rate data:

- Peak price_first, price_next = 0.10
- Off-peak price_first = 0.06
- Off-peak price_next = 0.08

(Yes, it is correct – the **main** off-peak period price is specified in the **price next** parameter).

Charging calls – the traditional method

With Maintenance Release 9 a new, more flexible rating formula has been made available (see the next section); however, the traditional method is still fully supported for backward compatibility.
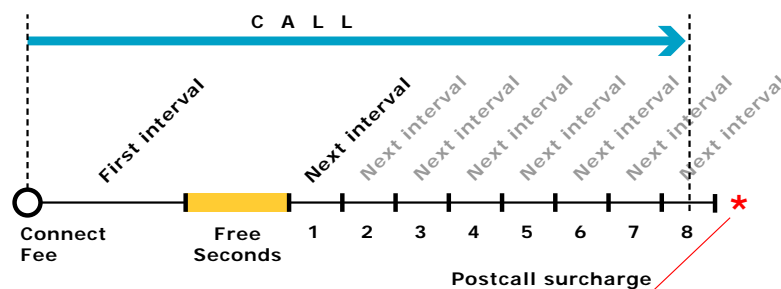

Figure 1-2

Figure 1-2 demonstrates how calls are charged. A **Connect Fee** is charged immediately upon connection, and all calls shorter than the *First interval* will be rounded to **First interval** seconds. Free seconds are granted after the *First interval*, so this part of the call is not charged. Calls longer than (*First interval+Free seconds*) will be rounded up to multiple **Next interval** seconds. After that, the **Post Call Surcharge** is applied. The call illustrated in the figure above will be charged using the following formula:

```
Amount_Charged = (Connect_Fee +
  First_Interval * Price_First/60 +
 8 * Next_Interval * Price_N/60) * (1+Post_Call_Surcharge/100)
```

Parameters such as **First interval, Next interval**, **Price First** and **Price Next** can be specified per destination. **Connect Fee**, **Free Seconds** and **Post Call Surcharge** are defined on a per-tariff basis, and so will be the same for all destinations within a tariff.

Attention: The billing unit may be any length, but price must always be entered on a per-minute basis. This allows better operations with tariffs, for example, comparing two tariffs with different billing intervals or entering rates into the system from an external source.

Charging calls – the rating formula method

This method gives you maximum flexibility in call rating. You create a call formula which is then applied to the call. This formula can consist of an unlimited number of charge elements (the only limit is the length of the formula text). These elements are:

- **Interval** - charged part of remaining call duration; the parameters for this element are:
  - o **Duration** – rounding period (in seconds)
  - o **Count** – number of rounding periods in the interval
  - o **Price per minute** – automatically prorated according to the rounding period duration

  The charge for the interval is calculated as:
  `Total = N * (Duration/60) * Price_per_minute`,
  Where `N` is either equal to `Count` (if `Call_Duration` is more than `Count*Duration`) or else `Call_Duration/Duration` (rounded up). Thus you can create intervals such as "First 10 minutes – price per minute is $0.05 and we will round up to 30 seconds" – in this case, the duration will be 30, the price per minute 0.05 and the count 20 (because there are twenty 30-second intervals in 10 minutes).
- **Fixed Surcharge** – fixed amount to be added to the total call cost
- **Relative Surcharge** – the total call cost (at this moment) will be increased by X%

Formula elements are applied until there is no more non-charged call duration left. The surcharges which follow a certain interval are applied only if this interval has been fulfilled. "Fulfilled" means that the interval is covered completely, i.e. the remaining call duration was enough to cover the `Count * Duration` time. Let's take an example; if we construct the formula:

- 3 x 60 seconds, 0.10/min
- Fixed surcharge 0.05
- N x 60 seconds, 0.10/min

then, when a call is made for 1 minute and 5 seconds, the charged amount will be 0.20 – in the first interval we use two increments of 60 seconds each, with price per minute 0.10. Since this first interval is to contain three increments, and we have used only two, the surcharge which follows the interval is not applied.

When a call is made for 4 minutes and 20 seconds, the total charged amount will be 0.55: three increments of 60 seconds each, a surcharge of 0.05, and then another two increments of 60 seconds with price per minute 0.10.

There is one exception to the surcharge application rule above: if the last formula element is a surcharge, it is always applied. This allows emulation of the "post-call surcharge" from earlier releases.
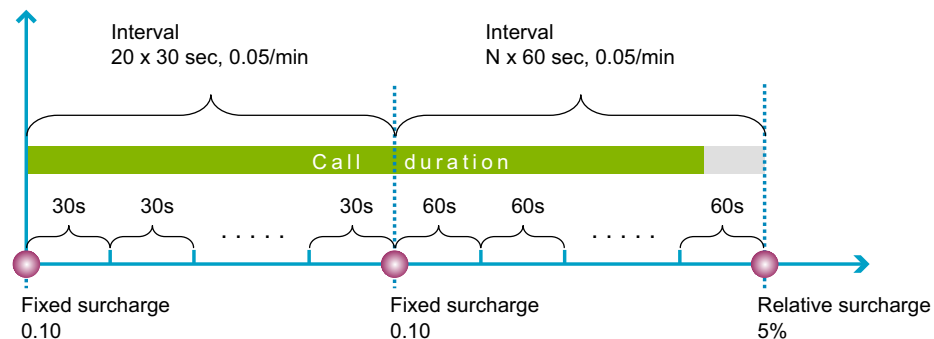

Figure 1-3

Figure 1-3 demonstrates how calls are charged. For example, the rating scheme "Charge 0.10 connection fee, then first 10 minutes should be rounded up to half a minute with price per minute 0.05. If the customer calls for more than 10 minutes, apply 0.10 fee and round the rest of the call up to 1 minute with price per minute 0.05; apply 5% post-call surcharge at the end of the call" will be translated into the following formula:

- Fixed surcharge 0.10
- Interval 20 x 30 seconds, price per minute 0.05
- Fixed surcharge 0.10

- Interval N x 60 seconds, price per minute 0.05
- Relative surcharge 5%

Most often you would like to set up a general call rating scheme (rounding, surcharges, etc.) immediately, even though you foresee that your rates per minute will change in the future. In this case, you may construct the formula as you would normally do, but instead of entering a fixed price per minute in the formula you can specify: "use the current value of the **Price_Next** parameter". Later, when you decide to change your rates, it will be much easier to download the current rate data, alter just one column (**Price_Next**), and upload the file again, rather than change the formula in every row. Also, using the pseudo-variables First Price and Next Price is required to ensure that the correct rate is used during the off-peak period, since the system will then automatically populate the variables with the values you specified for either peak or off-peak prices. Of course, if you entered a constant price value into the formula it will remain unchanged regardless of the off-peak period.

Using formula parameters linked to rate values has another advantage: if you have different prices or interval durations in peak/off-peak periods, the system will automatically use the one applicable.

Tricky vs. honest charges

Should surcharges affect the time announced to the user? For example, if an account has a $10 balance, and the price per minute is 0.10 but there is a 0.20 connection fee (or 5% post-call surcharge), should the customer hear: "You have 100 minutes"? Should he be allowed call for 100 minutes, or less? PortaControl allows calculation of two call durations during call authorization:

- The call duration to be announced to the customer (`h323-credit-time`). This is a default parameter, and historically has been used to limit the maximum call duration as well.
- The actual call duration (when the call should be disconnected – `h323-ivr-in DURATION`).

So the difference between the "honest" and "tricky" charge is that the former affects both values for call duration, while the latter only affects the "actual" call duration. So, in our example above ($10 balance, rate 0.10/min):

- if the fixed charge 0.20 is honest, then the customer will hear "You have 98 minutes", and will be disconnected after 98 minutes.
- if the fixed charge 0.20 is tricky, then the customer will hear "You have 100 minutes", but will still be disconnected after 98 minutes.

⚠️ You are free to use either honest or tricky charges, depending on how your business model is set up. Please note that it is your responsibility to make a proper disclaimer to customers when using tricky charges.

NOTE: In order to implement this functionality, your gateway/IVR should be able to work with two call duration values. This feature is available in recent releases of the Quintum firmware. For the Cisco platform, you will have to modify the prepaid IVR TCL script.

Random charges

When using the tricky charges described above, one problem is that customers may still discover that they are being charged over and above the per-minute rate by comparing the amount deducted from their account to the estimated charges. PortaControl allows you to make this detection much more difficult, by applying special charges only randomly. As a result, special charges will apply not to all calls, but rather only to some of them, which makes it virtually impossible to "reverse-engineer" your system of charges.

In creating a rating formula, you may assign a probability to each surcharge (fixed or relative). This parameter defines the chance of this surcharge being applied, e.g. a probability of 25% percent means that there is a one in four chance of its being applied when calculating the total charged amount for a call, such that, in the final analysis, this surcharge will, on average, be applied to every fourth call. Another example is the random charges configuration shown below:

```
Rating Sequence

5 intervals of 60 seconds, 0.10 USD/min
Fixed surcharge: 0.15 USD. Type: Tricky. Probability: 25%
N intervals of 60 seconds, 0.10 USD/min
Relative surcharge: 30%. Type: Tricky. Probability: 20%
```

Figure 1-4

Since every random charge is evaluated independently, this formula will produce the following result (an average calculated based on a large number of calls):
- in 60% of calls, no surcharges will be applied at all
- in 5% of calls, both the fixed surcharge and relative surcharges will be applied
- in 15% of calls, the post-call surcharge will be applied
- in 20% of calls, the fixed surcharge will be applied

Random disconnects

Since fees such as the connect or disconnect fee are applied once per call, it is obvious that you will earn more profits on many short calls than on a

few long ones. Also, you may not wish to allow your customers to make too long calls, as this occupies incoming phone ports on your gateway. So your goal is to disconnect some calls so that customers will have to redial. If the disconnect occurs at a certain moment in time (e.g. the 20th minute of a conversation), it will be very easy for the end-user to notice. Therefore, the disconnect should be made to appear as if it had happened naturally, that is, randomly.

The random disconnect element in the PortaControl rating formula allows you to specify a certain probability according to which calls will be disconnected at a random moment of time during a certain period (for instance, between the 10th and 20th minute).

Let's take a look at the example below:

```
                          Rating Sequence

Fixed surcharge: 0.20 USD. Type: Tricky
10 intervals of 60 seconds, First price USD/min
Call disconnect: Dispersion: 300 sec. Probability: 20 %
30 intervals of 60 seconds, Next price USD/min
Call disconnect: Dispersion: -900 sec. Probability: 100 %
N intervals of 60 seconds, Next price USD/min
```

Figure 1-5

There are two call disconnect elements. The first one specifies, with a probability of 20%, that the call should be disconnected within 5 minutes (300 seconds) **after** this element is entered into the formula, i.e. the disconnect will happen sometime between the 10th and 15th minute of the call. The second element specifies that the call will **always** be disconnected (100% probability) 15 minutes (900 seconds) **before** this element is entered into the rating formula (i.e. the dispersion parameter is negative); so the disconnect will occur sometime between the 25th and 40th minute of the call. Thus, when this formula is applied in the actual tariff, the result will be that there are no calls longer than 40 minutes, i.e. every call will be cut off at some random instant between the 25th and 40th minute, while some calls will even be disconnected earlier (between the 10th and 15th minute).

Too short calls

Unfortunately this is a common problem: for some destinations (usually where analog lines are used) your vendor will report the call as connected once the remote phone starts ringing. So, even if no one picks up the phone, such calls will still be considered successful by your system, with a duration (for instance) of 15 seconds, and the customer will be charged for them – which will probably lead to disputes and arguments. Using the new call rating formula, you can specify that short calls to a specific destination be disregarded. So, if you set the **Do not bill calls shorter**

than parameter to 20 for a certain destination, and if a call lasts less than 20 seconds, it will not be billed at all. However, if the call was made for 20 or more seconds, it will be charged for the full call duration.

Add duration

This feature is similar to the "relative surcharge" discussed earlier, but works in a different way. The main disadvantage of post-call surcharges is that the time charged does not correspond to the officially-quoted price per minute.

For instance, if you are using 30-second rounding, a 0.10/min rate and a 10% post-call surcharge, and the customer makes a call for 4 minutes and 52 seconds, he will be charged $10*(30/60)*0.10 + 10\% = 0.55$. However, on the CDR this will be represented as "charged time: 5:00 minutes, charged amount: 0.55" - which might lead to a dispute, since you have quoted the customer a rate of 0.10/min.

The Add duration feature allows you to increase the call duration before any other charges are applied, so it will look as if the call was actually made for the longer duration. Thus, taking the example above, what will happen if we replace the relative surcharge with an add duration of 10%? First of all, the call duration will increase by 10%, becoming 5:21. Then this call will be rated using 30-second intervals, i.e. it will be rated $11*(30/60)*0.10 = 0.55$, and the CDR will read: "charged time: 5:30 minutes, charged amount: 0.55" - therefore it will be perfectly in line with your quoted rates.

Using "billing tricks"

Please note that you are responsible for finding out which call rating features are allowed in your business and providing your customers with the required disclaimer.

Using the old rating method with the new formula

The old method of call rating can be represented by the following rating formula:
- Fixed surcharge, equal to the *Tariff connect fee*
- 1 interval with *First_Interval* duration and *Price_First* price per minute
- 1 interval with *Tariff free seconds* duration and 0 price per minute (if the free second parameter in the tariff is omitted, this interval is skipped)
- N (until the end of the call) interval with *Next_Interval* duration and *Price_Next* price per minute
- Relative surcharge, equal to the *Tariff post-call surcharge*

Volume-based billing

Volume-based tariffs

This feature has been replaced by volume-based discounts; see details below. Maintenance Release 13 of PortaControl will support existing billing configurations involving volume-based tariffs, but you are strongly encouraged to migrate toward a scheme utilizing volume-based discounts, as this provides far better management of traffic volumes.

Volume-based discounts

Sometimes you want to adjust rates based on the volume of traffic your customer has already sent you – volume-based discounts are ideal for this. Key features of volume-based discounts include the following:

- A discount is defined per destination group. For example, if you wish to provide special rates for calls to "ex-USSR countries" you can create a destination group that will contain prefixes for Russia, Ukraine, Belorussia and others, and define a single discount rule for this destination group.
- Multiple discount steps per destination group. For example, you can define a discount rule stating: first 200 minutes free (100% discount), then up to 500 minutes at the normal rate (0% discount), then a 10% discount for every minute up to 1,000 minutes, and a 20% discount for every minute over 1,000.
- You can use thresholds based on call time (e.g. 10% discount after 200 minutes) or on the call cost (10% discount after more than $10 worth of calls).
- Several discount rules may be grouped into a volume discount plan. For instance, your service may include 300 free monthly minutes to US&Canada, 100 free monthly minutes to Western Europe, and a special offer of 15% off calls to India after the customer has called for more than 200 minutes. Each of the conditions above (e.g for calls to US&Canada or India) is represented by an individual discount rule. Taken all together, they will be grouped as discount plan "EasyCall", which can then be applied to a specific account.
- For better management, you can associate a volume discount plan with a certain product. This discount plan will then be applied by default to every account using this product. You may also override the discount plan setting for a specific account, so that the system uses a specifically assigned discount plan instead of the one defined in the product.
- Discount plans may also be assigned at the customer level. If both account and customer discount plans are engaged for a certain call, PortaControl will combine them. For instance, if the account

discount plan provides a 30% discount, and the customer discount plan provides a 20% discount, the result is that a 50% discount will be applied to this call.

- Discount counters (amount of currently used minutes) may be automatically reset in accordance with the customer's billing period. This makes it very easy to implement services such as "200 free minutes per month".

The discount is applied to every minute which exceeds the threshold. Let's look at the following example:

- You have the discount rule "first 200 minutes – normal rate $0.20/min, 15% discount after 200 minutes" for calls to Israel.
- During one month the customer spends 230 minutes on the phone.

In this case, the customer will be charged $200*0.20 + (30*0.20 – 15\%) = 40 + 5.1 = 45.1$. Sometimes people get confused, and will assume that the charges in this case should be $230*0.20 – 15\% = 39.1$, which is **incorrect**.

Subscriptions

The **Subscriptions** module allows you to charge your customers periodic fees for using the service. If your advertisement states something like: "only $9.99 per month, and just $0.99 extra per month for Voicemail service", two subscription plans are involved here. When you define a subscription plan, you define various parameters that specify what effect the plan will have on the customer. Every subscription plan includes the following:

- Name and description – Used by your administrators to better manage various subscription plans.
- Invoice line description – The subscription name your customers will see on the invoice.
- Activation mode – Specifies when the subscription is active and charges start to be assessed. Typically, a subscription is activated immediately upon its creation (or on the start date, if a future start date has been specified). PortaControl supports an additional mode of subscription activation based on the account's first use. This allows you to avoid problems when there is a delay between the time a customer signs up for a service and the time he is actually able to use it (e.g. he uses online signup to purchase a residential SIP service, but his IP phone will only be delivered by FedEx five days later).
- Activation fee – The amount to be charged when the subscription becomes active.

- Minimum subscription period and early cancellation penalty – It is common practice to lock a customer's contract for a certain period (e.g. if you provide the customer with a free IP phone, you want to make sure he keeps paying monthly fees until he has paid back the cost of the phone). If the subscription service is canceled earlier than the interval specified, the customer will be charged a cancellation penalty.
- Rounding – To avoid unusual subscription charges like $1.26789, you can specify a rounding pattern for them (in a similar way as you have done with rating in tariffs). For example, the rounding pattern XXXXX.XX000 will round up all subscription charges to whole cents.
- Of course, every subscription also includes a definition of periodic charges applied while the customer has an active subscription; see below for more details.

Applying subscription charges

Traditional method

This a very common and widely popular method. When a customer's billing period is closed, PortaControl calculates the applicable charges for all subscriptions which were active during that period, and applies these to the customer's balance.
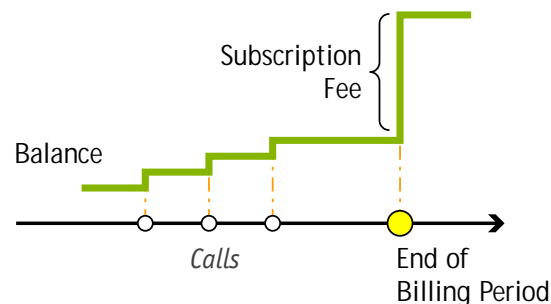
As a result, at the moment the billing period is closed there may be a significant increase in the customer's balance.

Progressive method

While the situation described above is acceptable for postpaid customers with a large credit limit, or if the customer has a credit card which may be charged automatically when the billing period is closed, it may create certain problems for prepaid customers. If such a customer does not have sufficient available funds at the end of the billing period, his balance will exceed the credit limit, and his outgoing calls may be blocked. PortaControl provides a solution to this problem, allowing you to conveniently offer prepaid VoIP services in combination with subscription fees.

Balance

SC   Calls   SC   Calls   SC   End of
                                    Billing Period
SC = Subscription Charges
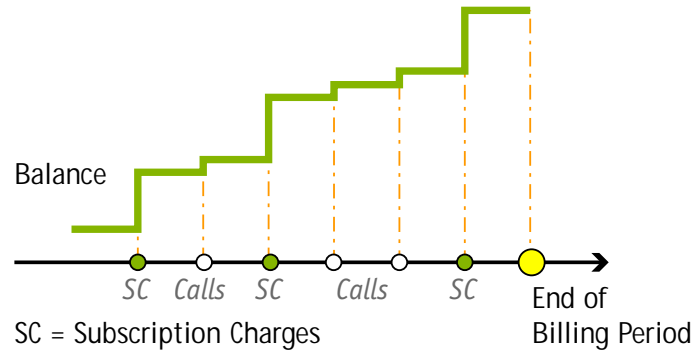
To avoid the unpleasant situation of a sizeable charge at the end of the billing period, PortaControl's progressive charges are applied continuously, so that, at any given moment in time, the amount covering the interval from the beginning of the billing period until the current day has already been charged to the customer. For example, if a customer's monthly fee is 9.99, on the first day of the month he will be charged 9.99 / 30 = 0.33. On the next day the charge will be 0.66, and so on until the end of the month, when the charge will equal the full amount of 9.99. Thus the customer is effectively charged a small portion of the total fee every day. His balance grows slowly, giving him enough time to react and deposit more money into his account.

Periodic subscription fees

As mentioned above, periodic fees are charges applied continually throughout a subscription's lifetime. Since your customers may use different billing periods, PortaControl gives you the flexibility to assign subscription costs individually for each period. For instance, if your monthly rate is 19.99, you will not want to offer a weekly rate of 4.99, since the increased maintenance required by customers with a short billing period justifies increased rates.. PortaControl allows you, for instance, to define your monthly rate as 19.99, bi-weekly as 10.99, weekly as 6.99 and daily as 1.99 for the same subscription plan. PortaControl will automatically use the correct base value according to the customer's billing period.

Promotional periods

Yet another common business practice is offering special rates for an initial period following signup (e.g. "Only 9.99/month!", while the disclaimer states: "For the first six months only, after which the standard rate of 29.99/month applies."). PortaControl allows you to define an unlimited number of subscription periods, with different subscription fees for each. For example, you could create a subscription which offers free service for the first three months, a rate of 19.99 for the next nine months, and 12.99 thereafter; or something even more complex.

Applying subscriptions

Since a product defines the way you provide a service to the end-user, and subscriptions define the charges for this service, it is obvious that the two must be interconnected. There is now a new property of products – the subscription list – that defines which subscription plans are to be included in a given product. For instance, your EasyCall product may only include one subscription plan, e.g. "SIP calling – 9.99", while the SmartCall product might include "SIP calling – 9.99", "Voicemail – 1.99" and "Follow-me – 0.99". A SmartCall customer will be charged 12.97 in total, but there will be three separate invoice lines, one for each subscription plan. Thus, when an account is assigned the SmartCall product, three subscription records are created for it (SmartCall, Voicemail and Follow-me), while only one subscription record is created for an account with the EasyCall product. These **subscription records** (not to be confused with **subscription plans**) reflect the subscription's actual status with regard to this account.
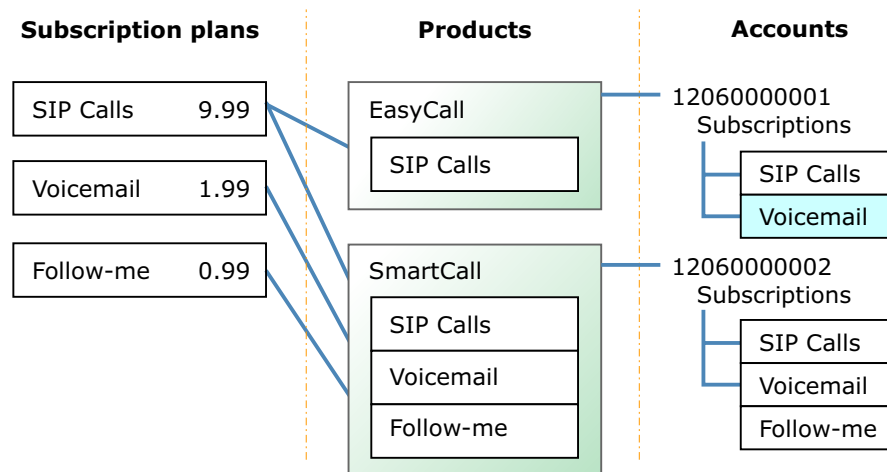
Subscriptions which an account receives as part of its product are called mandatory subscriptions, because it is not possible to remove or cancel these subscriptions so long as the account uses the product. You may, however, add some optional subscriptions to an account (e.g. if an EasyCall product user wishes to receive the Voicemail service). These optional subscriptions can be added or canceled as you wish.

You can also assign subscriptions to a specific customer. Here the procedure is exactly the same as with an account. Since a customer cannot have a product assigned directly to him, there are no mandatory subscriptions in this case, i.e. all of a customer's subscriptions are optional.

Every subscription has parameters such as activation date, cancellation date and (most importantly) the "billed to" date, which defines the period for which subscription fees have already been charged.

Activating a subscription

By default, a subscription is automatically activated when its start time arrives, with one exception. Namely, in the subscription plan properties you may specify that the subscription will become active only when the account is used for the first time. In this case, the subscription is activated either on the start date or the account's date of first use, whichever comes later.

Canceling (closing) a subscription

A subscription which is not yet active can simply be deleted (for instance, a customer has signed up for a new service beginning next month, but then changes his mind). A subscription which is already active cannot be deleted; when the subscription is canceled, the cancellation date is entered as the end date of the subscription. Typically, the cancellation date will be the day on which cancellation took place, but you can also schedule cancellation for a future date (e.g. a customer wishes to use the service until the end of the present month). The record of the canceled subscription allows PortaControl to correctly charge for closed subscriptions. For example, if a customer subscribed to the service on the 3rd and cancelled it on the 7th, he will still be charged for five days of service.

Product modifications

Changing the subscription plan definition of a product

If you modify the list of subscription plans for a given product, this will not affect existing accounts with this product. This basically allows you to sell the same product over an extended period of time, simply adjusting the subscription plans within the product according to current promotional offers, without affecting any old customers.

Changing an account's product and subscriptions

A change of product for an existing account, e.g. from product A to B, will have the following effect on this account's subscriptions:
- Mandatory subscriptions present in both A and B will remain unchanged.
- Mandatory subscriptions which were part of product A (old), but not part of product B (new) will be closed (with the current date as finish date).

- Mandatory subscriptions under product B which the account does not yet have will be created as new ones (with the current date as start date).
- Optional subscriptions will remain unchanged.

Applying subscription fees

When a customer's billing period is closed, PortaControl calculates charges for any subscriptions that were active during this period. For each subscription, three types of charges may be made:

- Activation fee – if the subscription plan has an activation fee defined, and if the subscription was started during this period
- Cancellation fee – if the subscription plan has a minimum subscription period and early cancellation fee defined, and if the subscription was canceled during this period (where the total subscription duration is less than the minimum subscription time)
- Subscription fee – a periodic subscription fee, pro-rated to the actual duration of the service used; please consider the following examples, all of which apply to the "SIP calls" subscription with a monthly periodic fee of $9.99:
  - Customer A, with a monthly billing period, activated the subscription on April 12th. On May 1st he will be charged $6.33, because he used the service for 19 days (12..30) and (19/30) * 9.99 = 6.33
  - Customer B, with a monthly billing period, activated the subscription on April 12th, and canceled it on the 25th. On May 1st he will be charged $4.66, because he used the service for 14 days (12..25) and (14/30) * 9.99 = 4.66
  - Customer C, with a monthly billing period, activated the subscription in March and canceled it on April 23rd. On May 1st he will be charged $7.66, because he used the service for 23 days (1..25) and (23/30) * 9.99 = 7.66

All these types of charges will be reflected on the invoice as separate invoice lines.

Incompletely used promotional periods

If you offer promotional rates for X billing periods, then every billing period in which the service was used counts towards it, regardless of the amount of days used in a particular billing period. Thus, if you offer a promotional period for the first four weeks, and customer subscribes on a Wednesday, his first week (of 5 days) and the three full weeks following it will be charged according to the promotional rate. Starting from the fifth week the default rate will be used.

In order to avoid potential misunderstandings when a customer expects a longer promotional period, the best practice is to use an anniversary

billing cycle, as this eliminates the problem of "incompletely used" promotional periods.

Invoicing

At the end of the billing period, PortaControl can produce an invoice for your customers. An invoice reflects all the transactions (calls, payments, refunds, subscription charges, and so on) which occurred during this period, and serves as the primary record of services provided to a specific customer, as well as his current status.
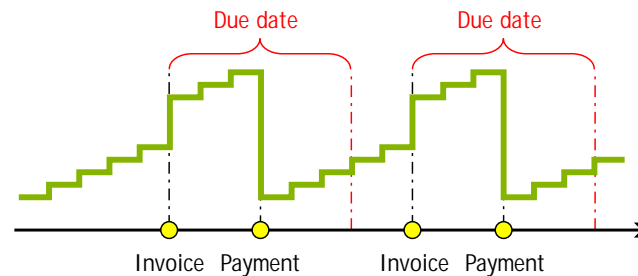
Billing periods

A billing period defines how often invoices/statements will be generated and when exactly they are to be produced. PortaControl supports the following billing periods

- Daily – Covers a 24-hour period.
- Weekly – Covers a 7-day period (Monday through Sunday).
- Bi-weekly – Covers a period from the 1st to the 15th or from the 16th to the last day of the month.
- Monthly – Covers the period from the 1st of the month to the last day of the month.
- Monthly (anniversary) – Covers the period from the Nth day of the month to the day before the Nth day of the following month. N is the day of the month when the customer was created; thus if a customer was created on January 6th, his invoices will always cover the period from the 6th of the current month to the 5th of the following month.
  To avoid complications for customers who were created on the 29th, 30th or 31st day of the month, their first billing period will cover the time until the 1st day of the month after the next one, and thereafter will always cover the period from the 1st to the 1st. For instance, if a customer was created on March 30th, his first invoice will cover the period up to May 1st, while his next invoice will cover the period from May 1st to May 31st.
- 30 days – Every billing period is exactly 30 days, so if a customer was created on March 20th, his first invoice will cover the period

from March 20th to April 19th, the second invoice will cover the period from April 20th to May 20th, and so on. This invoicing method allows you to make subscription fees more straightforward compared to regular monthly billing, where the same monthly fee is applied to longer (e.g. March) as well as shorter periods (e.g. February or April).

Every billing period is adjusted to the respective customer's (or vendor's) time zone. So, in the case of a customer with the Los Angeles time zone and a weekly billing cycle, the billing period will start at midnight on Monday Los Angeles time, while for a customer with a weekly billing cycle and the Singapore time zone it will start at midnight on Monday Singapore time. Thus while both invoices will cover 7 days (168 hours), they will actually refer to different intervals of time, with a 15-hour difference.

Closing a billing period

Or, in other words: when will invoices actually appear in the system?

By default, invoices/statements are generated on the day after the last day of the billing period, e.g. invoices for customers with a weekly billing cycle are generated on Monday, while invoices for customers with a monthly billing cycle are generated on the first day of the month. A transaction is included in a certain billing period if it **started** during this period; this means that calls are considered as falling within a certain period according to their connect time. For example, if a call starts at 23:55 (11:55pm) on March 31st and finishes at 00:43 (12:43am) on April 1st, then this call belongs to the March billing period.

For this reason, a billing period cannot be closed the next day at midnight sharp; there might be calls in progress which started just a few minutes ago, and which should still be included on the current invoice. PortaControl waits a sufficient amount of time before closing a billing period, to ensure that all calls have been completed. By default, this interval is six hours, but it can be changed in the `porta-admin.conf` configuration file.

Also note that since statements are generated in the time zone of the customer/vendor, the billing period is closed in that time zone. Thus if your system time zone is Singapore, you cannot expect to see invoices for your US customers on Monday morning, since it is still Sunday evening then in the US. Finally, in order to provide optimal system response time for your online users, PortaControl only performs resource-intensive calculations (such as creating statistics/invoices) during the period specified in your configuration as "off-peak". So, in the example above, if your "statistic calculation" hours are defined as 2:00am to 7:00am, you will receive invoices for US customers on Tuesday morning.

Invoice templates

You can design multiple invoice templates, so that each template has its own layout, language, logos/pictures, and the like. Every customer will be assigned a specific invoice template, according to which PortaControl will create a PDF file that can be emailed to the customer and/or downloaded from the PortaControl web interface (by both the administrator and the customer himself). It is also possible to use a special "Do not create invoice" option for a customer. In this case, PortaControl will only produce CDR summaries, but no actual invoices will be generated.

Please consult the **PortaBilling Templates Guide** for more information about creating and managing invoice templates.

Invoice parameters

Every invoice contains global invoice data, which is stored as part of the invoice record in the database:
- Invoice number – unique identifier of an invoice
- From date – start date of the billing period
- To date – end date of the billing period
- Invoice date – date when the invoice was generated
- Due date – date by which payment should be received
- Payment terms – description of payment terms (e.g. "due on receipt")
- Invoice total – sum of all charges in this period minus credits/refunds
- Invoice amount due – amount the customer is supposed to pay you (see below for a detailed explanation of invoice balances)
- Invoice payment status – this specifies the following:
  - **Do Not Pay** – invoice amount is 0, no payment is required
  - **Unpaid** – no payment has been received yet
  - **Partially Paid** – payment has been received, but in an amount less than the amount due
  - **Paid** – invoice has been paid in full
  - **Overdue** – invoice is unpaid and past due
  - **N/A** – payment status is not applicable to this invoice.

Invoices also contain certain data extracted from other PortaControl objects, which are included in the invoice's PDF version:
- Information about the customer (invoice recipient)
- Information about the invoice issuer (your company or a reseller, if the invoice was issued on behalf of a reseller)
- Information about calls and other transactions included in the invoice

Invoice balance

PortaControl provides two methods for computing the invoice balance (amount due):

- Simple – In this case, the invoice's amount due is equal to the invoice total, and is calculated as the sum of all charges during the given period (no previous payments are taken into consideration). This is an optimal method for integration with an external bookkeeping system, where you keep track of your incoming payments via accounting software, and not PortaControl.
- Balance-aware – The invoice total is calculated as the sum of all charges during the given period (both call and non-call related), minus the sum of all credits/refunds. The invoice's amount due is calculated as: previous_balance + invoice_total - payments. This allows you to "carry over" a balance in the case of partial payments. For example, suppose a customer's March invoice was for $40. If he makes a payment of $30 on April 10th, makes calls for $25 during April, and is also issued a $3 refund, then his April invoice will have an invoice total of $25 - $3 = $22, while the invoice's amount due will be $40 + $22 - $30 = $32.

By default, PortaControl uses the balance-aware invoice generation method.

Charging the invoice balance

To simplify payment collection and improve cash-flow, PortaControl can charge a customer's credit card before closing an invoice. So if, as in the example above, the invoice's amount due was initially $32, the customer's credit card will be charged $32, with payment entered to the April billing period. As a result, the customer's invoice will be created with a zero amount due.
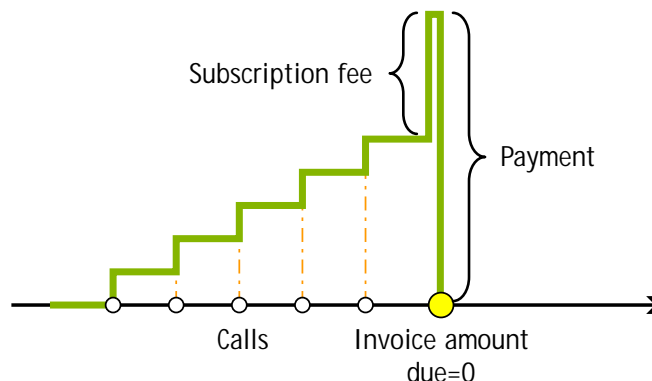
This type of periodic payment can be activated on a per-customer basis (using the Payment Info tab) and used in conjunction with other types of periodic payments (e.g. balance-driven).

Payment status

Every time a payment is recorded in the system (this could be a periodic payment, an online payment by a customer, or a payment entered manually by the administrator), in addition to modifying a customer's balance it will also be applied to one of his unpaid invoices. If the amount so applied equals the invoice's amount due, the invoice becomes "paid", while if the payment is less than the amount due, the invoice becomes "partially paid". Payments are applied to an invoice cumulatively; thus if an invoice is for $30, and the customer makes three payments of $10, $13 and $17, following this last payment the invoice will be "paid". If a customer has several unpaid invoices, the payment will be applied to the oldest one. If a payment exceeds the total amount of all unpaid invoices, the remaining sum will be assigned to the special customer property "unallocated payments", and applied to future invoices. For instance, suppose a customer receives his weekly invoice, with an amount due of $8.99. Since he plans to leave for a three-week vacation, he sends in a payment of $36. This entire amount is applied to the customer's balance, so that $8.99 will cover the existing invoice, while $27.01 will remain in "unallocated payments". When his next several invoices are created, they will show an amount due of zero and the status "paid in full".

NOTE: Unallocated payments do not represent a "cash reserve". When a payment is made, the amount is immediately applied to the customer's balance. Unallocated payments merely show that the customer "overpaid" you sometime in the past, and are used to correct the paid/unpaid status of future invoices.

Collection policy

Customer class allows you to define a policy for automated payment collection.

Important dates

- Issue date defines the moment when an invoice was issued; all other dates are generated based upon it.
- Invoice grace period indicates within how many days following the invoice issue date a payment must be received. The invoice due date is calculated as invoice_issue_date + grace period. The grace period can be zero, in which case the invoice is considered to be due upon receipt.
- Suspension time defines within how many days the customer will be suspended after an invoice becomes overdue. Suspension means that, while the customer's information will remain intact, he will not be able to use certain services, most notably sending or receiving calls.
- Closing time defines within how many days a customer account will be closed after an invoice becomes overdue.

Let's take a look at the following example. Customer A has a grace period of 21 days, a suspension time of 14 days, and a closing time of 21 days. A's invoice was generated on May 1st, so that the invoice due date is May 22nd. If A does not pay by that date, the invoice will become overdue; 14 days (2 weeks) after that – on June 5th – his account will be suspended; and on June 12th – 21 days after the due date, or one week after the suspension date – his account will be permanently closed.

Notifications regarding payment due

Since it may happen that a customer has lost or overlooked the original invoice, PortaControl can send automated notifications to a customer several times before the invoice due date, stating that payment has still not been received. You can configure any set of days, e.g. "10,7,1" will send notifications 10 days, 7 days and 1 day prior to the invoice due date.

Overdue invoice notifications

When an invoice is past its due date, PortaControl will continue notification attempts, but in this case using a different notification text. Here again, you have complete freedom in configuring this notification policy: for instance, "0,7,14" will send an alert to the customer on the due date, and then 7 and 14 days later.

Re-collection attempts

It could be that the initial attempt to charge a customer's credit card has failed due to a temporary problem (e.g. he exceeded his credit limit, having recently made a very expensive purchase, but has now paid his credit card bill). In such a case, it would be useful to try charging his card sometime later. However, since the merchant bank will usually charge you for every failed credit card transaction, this should not be done too often.

In the customer class definition you can specify when re-collect attempts should be made, e.g. "0,3,7" means that PortaControl will attempt to charge the customer's credit card on the due date, 3 days after the due date, and 7 days after the due date.

Suspension warning time

As a last resort to prevent service interruption, PortaControl will send another alert to the customer prior to service suspension. This parameter may also be configured in the customer class definition.

Customer account during collection

When a customer has an unpaid or overdue invoice, all call services are rendered as usual, subscription fees are charged for the current period, and new invoices are generated (for instance, if your customer has net

payment terms of 60 days, he may have two other (newer) invoices generated by the time his invoice becomes overdue).

When a customer has suspended status, his call services are blocked, but subscription fees are still applied (since his account is being provisioned, e.g. his voicemail is kept on the server).

When the customer's account is closed, no further activities, be it calls, subscription fees, self-care access, or any other, are available for his account.

Collection process
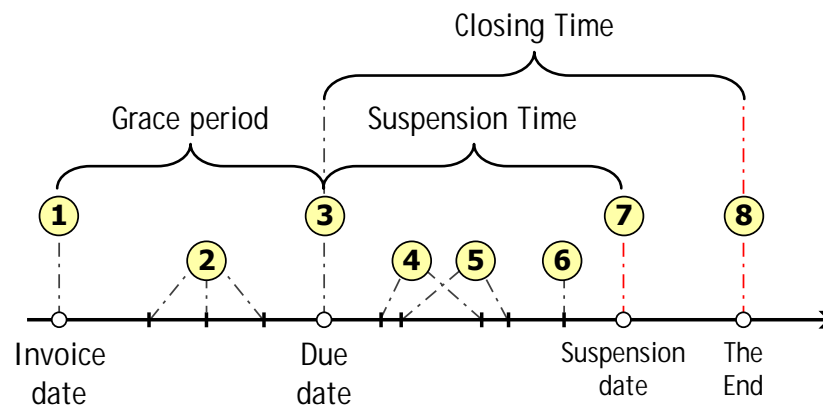
- At the end of the billing period an invoice is generated (**1**).
- If the invoice has a positive amount due, it is considered unpaid. When the invoice status is unpaid, a customer may be reminded of the approaching due date (**2**).
- On the due date, the unpaid invoice becomes **overdue** (**3**).
- Several collection attempts may be made (attempts to charge the customer's credit card on file for the amount due) (**4**).
- The customer may be reminded that the invoice is overdue (**5**) and that service may soon be suspended (**6**).
- On the suspension date, the customer's status changes to suspended (**7**), which automatically blocks his access to services.
- If the payment issue is not resolved, the customer's account is closed on the closing date (**8**).

Payments

PortaControl allows payments to be processed online, without the intervention of an operator, by charging the customer's credit card or debiting his bank account. Payments may be initiated by:

1. Your subscribers from the web interface (called "online payments" because payment is triggered by the customer, who is using the web interface at the moment the payment is made).
2. The PortaControl system itself. These payments are called periodic, because they usually happen automatically over a regular period of time. There are three sub-types of periodic payments:
   - Calendar payments, which happen every month, every week, and so on. This is a convenient way to charge a customer his full balance, or some other pre-defined amount, at a certain moment in time.
   - Balance-driven payments, which are triggered by certain conditions involving a customer's balance. When the customer's balance is higher than the specified threshold, payment is initiated automatically.
   - Automated payment, which is applied when closing an invoice. The customer's credit card is charged for the invoice's amount due, so that when the invoice is created it shows an amount due of zero

Payment flow

The following picture illustrates the individual components of the online payment system:
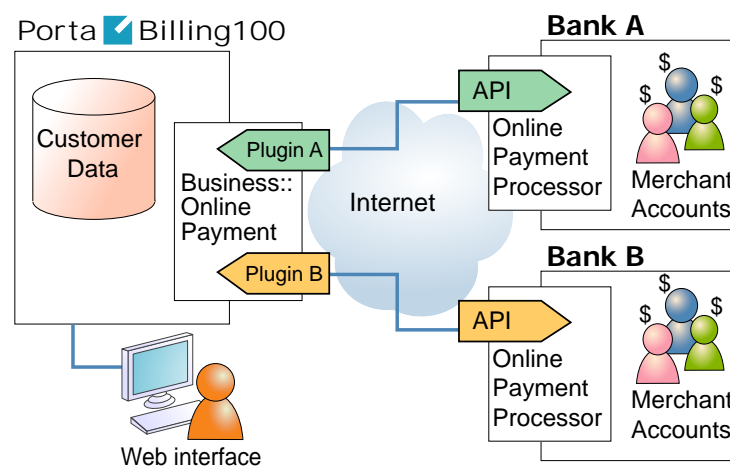
Merchant account

A merchant account must be opened at a bank. Its purpose is similar to that of a normal checking account. It stores funds you have received from credit card payments.

Online payment processor

A merchant account is usually enough to start processing credit card payments via charge slips. However, if you wish to initiate transactions from your own server via the Internet, this service is provided by online payment processors such as Authorize.Net. You establish an account with an online payment processor, providing it with your merchant account information, and in return you receive credentials (username, password, etc.) for using its API. So now your application can connect to the API's server and, upon providing valid authentication information, initiate transactions.

Business::OnlinePayment framework

Unfortunately, there are many different online payment processors, most of them using their own API which is different from that of other providers. So it seems that every e-commerce application such as PortaControl must be programmed to support them all, which is an extremely laborious task. Fortunately, there is a solution: Business::OnlinePayment is a framework which provides the necessary middleware between e-commerce applications and online payment processors.

- The application is provided with unified API methods, e.g. doTransaction, so the same code in the application can be used to communicate with any online payment processor.
- A communication plugin is created for each online payment processor. This plugin knows such specific parameters as the server's address, the correct way of encoding transaction parameters, the proper way of parsing the server's response, and so on.

On the application side, therefore, only a few things are required: loading the correct plugin and invoking a transaction method. After that control is handed over to the plugin, which handles all communication with the remote server.

The whole Business::OnlinePayment project is open-source, and is maintained by a large team of individuals. At the moment about 10 different plugin modules are supplied with PortaControl, and it is easy to create new plugins for online payment processors. You can find more information about the Business::OnlinePayment module and available plugins at **http://www.cpan.org/modules/by-module/Business/**.

Porta✦Control

PortaControl payment support

PortaControl stores required information, such as customer name and address, credit card number, and so on, in the database. Multiple merchant accounts are supported (for instance, your merchant account A accepts payments in USD, while account B takes payments in euros). You may define a "minimum allowed payment" and a list of supported card types (VISA, MC, etc.) for every merchant account.

When payment is initiated (either by the customer from web self-care, or automatically) the system connects to the online payment processor and performs the transaction. The online payment processor verifies the credit card information (and performs optional steps such as address verification or CVV control). Upon receiving confirmation that the transaction was successful, PortaControl writes transaction information to the PortaControl database and modifies the balance for the account/customer.

Supported payment methods

The choice of available payment methods depends on the online payment processor. In general, however, PortaControl supports the following payment methods:
- MasterCard
- VISA
- American Express
- Discover
- FirePay
- Switch/Maestro
- E-check (direct debit of bank account)

Templates

Templates are a powerful and flexible tool defining the relationship between data and its representation. Let's take a practical example: vendors send you their rates, and you have to enter this data into the system. Obviously, regardless of the particular numbering plans, formats and other options, this data has to be in the unified format used in PortaControl. Some vendors use Microsoft Excel, others text-based formats (for example, .CSV). Even if the same format is used, there will be different ways of positioning and formatting the data. For example, vendor A might have the phone prefix for the Czech Republic in the first column in his data file, entered as **420**. Vendor B might have it in column number three, entered as **011420**.

Of course, it is possible to write an external tool which would convert data files from these different formats into a common one. However, this would require some extra steps in data processing, so that every time you upload rates from a vendor you will have to perform a conversion first. In addition, this will probably involve extra cost, since not all users are capable of writing the required data conversion application.

This is why PortaControl has chosen a different approach. Instead of making your data suit PortaControl, you can do opposite: make PortaControl understand and process your data as is! Templates allow you to specify the following:

- Media format of the data (e.g. XLS, CSV or HTML).
- Where particular groups of data are located in the file (e.g. data on rates are on worksheet *Rates*, starting from the tenth row).
- Then you can specify which format each data element is in (for example, if the phone prefix is given as **011420** or as **420**). You can also apply post-processing rules (the analog of formulas in spreadsheet editors).
- For download templates (ones which describe how data extracted from PortaControl are to be represented), you can use our visual WYSIWYG layout editor.

Returning to the vendors in our example above, we can create one template for each of them and associate them with certain tariffs. Thus a corresponding template will automatically be used whenever you upload data for a given tariff.

For more details about templates, see the **PortaBilling Templates Guide**.

Translation rules

It is very often the case that you must make use of different numbering systems on your VoIP network. For example, Vendor A requires you to send all phone numbers with the technical prefix **12345#**, Vendor B requires that all phone numbers be in the format **011<*country-code><area-code><number>***, and, finally, when you send calls to the local telecom (Vendor C) you have to strip the country code and area code off completely. Of course, you would like to have this information in a unified format in your billing, so you will not need to enter multiple phone prefixes for the same destination.

PortaControl translation rules allow you to convert a phone number from a non-standard format to a unified one. A translation rule is a set of Perl regular expressions which are applied to the **Called-Station-Id** phone number before any other processing in billing takes place.

A translation rule is a set of s-regexp's; individual regexp's are separated by a semi-colon. The initial value is inside of the **$_** variable, and the result is considered to be the value found in **$_** after the last regexp has been evaluated. For example, here is a translation rule which will remove an initial double-zero (international dialing code) and also convert local numbers (ones starting with a single zero) into the E.164 format:

```
s/^00//; s/^0/420/;
```

Three types of translation rules:

- **Connection translation rules.** These are related to a particular connection, and convert a phone number from a unified format into a vendor-specific one during the routing phase, or from a vendor-specific format into a unified one during the call billing phase.

- **Node translation rules** (also called authorization translation rules). These are used to convert phone numbers from an IVR-specific format into a unified one during the authorization phase. Example: On your gateway A an IVR is running for prepaid cards. You use E.164 numbering in billing, so you expect all numbers to be dialed in E.164 as well (e.g. **42021234567** for the Czech Republic).
  However, very often your customers will dial this as **01142021234567** instead. Such calls will not be authorized, since there is no phone prefix defined in their tariff to match this number. To avoid this problem, you can set a translation rule for this node. Thus, when an authorization request comes in, we will first convert the phone number into the unified format (**42021234567**), and only then check the rates.

- **Customer translation rules.** These are applied in the same way as node translation rules (during authorization), but may be different for each customer (for example, one customer prefers to dial 00 before the number, another 011). If a customer has his own translation rule, then this will be used instead of the node-based rule.

Date and time information

Time zones

Due to the global nature of the VoIP business, it is very important to identify times with reference to time zones. For example, there is an 18-hour difference between 8pm Melbourne, Australia, and 8pm Seattle, USA – which could mean many thousands of calls and thousands of dollars incorrectly counted if the time zone is not taken into consideration.

This is why every object in PortaControl (e.g. user, customer, vendor or account) has a time zone associated with it. All dates are stored in the database in a universal, portable representation, and are converted into the required time zone when necessary. For example, say a call is made at 14:03 New York time (the time zone of your PortaControl server). The call is made using a calling card running on the America/Chicago time zone, and the account belongs to customer Easy-Cards-Reseller, whose time zone is America/Los_Angeles. The call goes to vendor Deutsche Telekom, which has Europe/Berlin as its time zone.
Thus, if each of these entities views information about the call, they will see it in their respective time zone, as follows:
- Account: 13:03
- Customer: 11:03
- Vendor: 20:03

This helps to present time information in a format suitable for the user. The time zone also affects the billing period. Thus if a customer has time zone Europe/Berlin and a monthly billing period, this means that his billing period covers one whole month, from 00:00 on the first day of the month until 00:00 on the first day of the next month **in his time zone** – Western European Time, in this example.

Also note that all time zones are expressed in relation to a geographical location in PortaControl, for example, Europe/Prague. This helps to avoid ambiguous abbreviations such as EST, which could mean both New York time and Melbourne time. Moreover, by using such a notation daylight savings time automatically comes into effect when applicable, so you do not have to think about whether it is EST (Eastern Standard Time) or EDT (Eastern Daylight Savings Time).

Date and time formats

Times and dates have different formats in different places in the world. For example, the same date will be represented as 20.05.2003 in Germany and 05/20/2003 in the US. It is important that the user be able to work

with dates in his native format in order to facilitate work and avoid data interpretation problems.

Each PortaControl user (admin, account, customer care) has his own date and time formats. When logging in to the web interface, all date and time information will be shown in the user's specified format. Additionally, all data input fields will also assume the date and time formats specified for the given user.

Call routing

When an entity on your network (for instance, a gateway or a SIP server) has to establish an outgoing call, it must find out where the call is being sent to. To do this, it can use its internal configuration (for example, dial-peer on the gateway) or some external entity (e.g. gatekeeper), or it may ask billing for a list of possible routes. This last method has several advantages: the routing configuration is in one central location, and billing can use information about termination costs to choose the best route (least-cost routing).
Currently, the following VoIP nodes can obtain routing information directly from billing:

- PortaSIP server (for SIP calls)
- Mera Voice Transit Switch (for H323 calls)

Control of routing on the Cisco gatekeeper (via GKTMP) or other equipment is not currently supported, but you can develop your own routing interface for PortaControl, which will communicate with PortaControl using RADIUS and then transfer this information to the originator in the correct format.

When a call goes through the PortaSIP server, the SIP server may:

- Direct the call to one of the registered SIP clients, if the called number belongs to the registered agent.
- Optionally, direct the call to the voice mailbox (PortaUM required) if the called number belongs to an account in PortaControl, but this account is not currently registered to the SIP server (is off-line).
- Route the call to one of the gateways for termination, according to the routing rules specified in PortaControl.

Routing of SIP on-net (SIP-to-SIP) calls

The SIP server automatically maintains information about all currently registered SIP user agents, so it is able to determine whether a call should be sent directly to a SIP user agent.

Routing of off-net calls

You can have different vendors for terminating off-net calls. For example, you can terminate calls to the US either to AT&T, via a T1 connected to your gateway in New York, or to a remote gateway from Qwest.

Rate routing parameters

Ordinarily, tariffs define the termination costs for each connection to a vendor. If you create a tariff with the **Routing** type, a few more fields will be added to rates in that tariff:

- **Route category** – you can split this into categories such as "Premium", "Cheap", etc. and use these categories in routing plans
- **Preference** – routing priority (0-10), higher values mean higher priority, 0 means do not use this rate for routing at all
- **Huntstop** – signals that no routes with a lower preference should be considered

This allows you to easily manage both termination costs and routing from a single location on the web interface. Thus, when such a routing tariff is associated with a connection, you can send calls for termination to all prefixes for which rates exist in the tariff.

Multiple routes

It is dangerous to have only one termination partner: if it is down, your customers will not be able to make any calls. Normally, you will try to find several vendors and enter their rates into the system. Each connection to a vendor (with routing tariff) will produce one possible route, and PortaControl will arrange these according to cost or your other preferences, thus providing fail-over routing.

Routing plans

Routing preferences in the rate allow you to specify that, for example, you would rather send a call to MCI than to T-Systems. However, this decision is "global", and so will apply to all calls made in your system. But what if you would like to use MCI first for customer A, while T-Systems should be the first route for customer B, and customer C should be routed to MCI only?

This can be accomplished using routing plans. A routing plan defines the routes for which categories are available, as well as in which order they should be arranged (route categories with a higher order value are tried first). For instance, in the example above MCI may be assigned as the "Normal" route category and T-Systems as the "Premium" category. After that, three routing plans will be created:

- **Quality** - includes first Premium and then Normal routing categories
- **Ordinary** - includes first Normal and then Premium routing categories
- **Cost-efficient** – includes only Normal routing category

So, depending on which routing plan is assigned to the current customer, the system will offer a different set of routes. There is one routing plan which always exists in the system, called **Default**. This is assigned to your customer unless otherwise specified. This routing plan uses all the available routes regardless of their route category.

Routing algorithm

The routing principle is simple:
- The SIP server (or MVTS, or some other entity) asks PortaControl for routing destinations for a given number.
- PortaControl checks every tariff with routing extensions associated with a vendor connection for rates matching this phone number. In each tariff the best-matching rate is chosen; this rate will define the routing parameters.
- A list of possible termination addresses will be produced (this will include the remote IP addresses for VoIP connections and IP addresses of your own nodes with telephony connections).
- This list will be sorted according to routing plan, routing preference and cost; entries after the first huntstop will be ignored.
- If there are several routes with identical cost/preference, load sharing will be applied, so that each potential route has an equal chance of being the first. Consequently, if your termination carrier provides you with three gateways to send calls to, at the end of the day each gateway will receive approximately the same number of calls (33%).
- A list of these IP addresses (with optional login and password for SIP authentication) will be returned to the SIP server. (To avoid extremely long delays, only a certain number of routes from the beginning of the list are returned; the default is 15, but this can be changed in `porta-billing.conf`).

Route sorting

How exactly does PortaControl arrange multiple available routes?
1. By route category. Only route categories which are included in the routing plan will be used, following the order given in the routing plan.
2. If you have multiple route categories within the routing plan, you can either merge them into the same group by assigning them the

same order value, or keep each one separate, with its own order value. Then routes within the same order group of route categories will be arranged according to preference.

3. For routes with the same preference, the system can arrange them according to cost (a comparison is made on the **Price_Next** rate parameter) so that cheaper routes will be among the first ones, or in random fashion.

Does PortaSwitch support LCR?

Yes, we support LCR – and much more besides. In fact, "just LCR" is the simplest type of routing PortaSwitch handles. If you decide not to use routing plans (one default plan for everyone) or routing preferences (same preference for all vendors), then the routing decision will be affected solely by the vendor's cost.

Compare the two illustrations below. The first one shows simple least-cost routing: all of the available carriers are arranged according to cost. In the second illustration, PortaSwitch routing preferences are used, so that the administrator can re-arrange the routing. In this case, carrier D has been moved down the routing list, while carrier A has been moved up to the top of the list, thus becoming the first route.

Routing configuration example
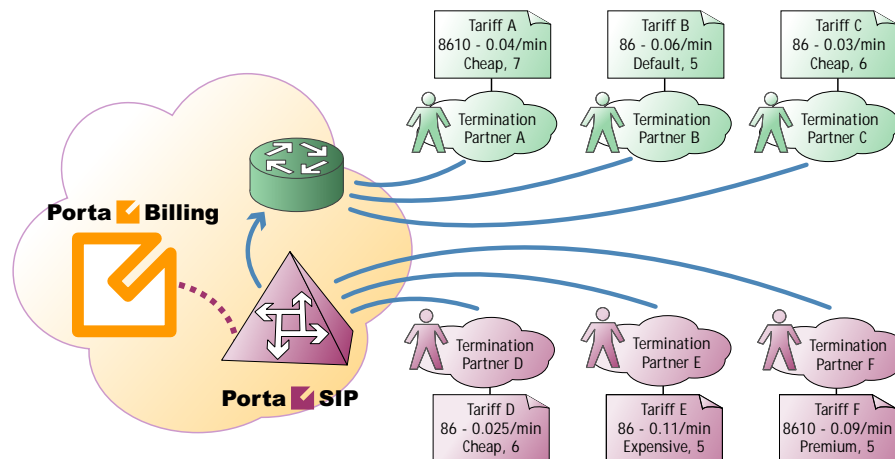
Consider the following example. If you have:
1. A "Standard" routing plan, which includes three route categories: "Default" (order 70), "Cheap" (order 40) and "Expensive" (order 10).
2. Six vendors (A, B, C, D, E, F) with the following rates (prefix, route category, preference, price):
   a. 8610, Cheap, 7, 0.04
   b. 86   , Default, 5, 0.06
   c. 86   , Cheap, 6, 0.03
   d. 86   , Cheap, 6, 0.025
   e. 86   , Expensive, 5, 0.11
   f. 8610 , Premium, 5, 0.09

then, when a customer with this routing plan makes a call to **8610234567,** the system will arrange the possible routes as follows:

Vendor	Parameters	Comment
B	Default, 5, 0.06	The "Default" route category is first in the route plan
A	Cheap, 7, 0.04	This vendor has the highest preference in the "Cheap" category
D	Cheap, 6, 0.025	This vendor has the same preference as vendor C, but a cheaper per-minute rate
C	Cheap, 6, 0.03	
E	Expensive, 5, 0.11	This is the only vendor in the last route category

(Vendor F was not included in the routing, since his route category is not in the customer's routing plan).

Number translation

There are many different phone number formats, some used by your customers, others by your vendors. How to deal with all of them without making mistakes? PortaControl offers a powerful tool called **translation rules** for converting phone numbers, with several different types depending on customers' needs.

Your network numbering plan

The key to avoiding problems with number formats is to choose a certain number format as the standard for your network and make sure that calls travel on your network only in this format. The ideal candidate for such a format is E.164 (of course it is highly recommended that you use this same format in billing as well). When a call arrives from your customer (with a phone number from a customer-specific number plan), PortaSwitch will convert the number into your network format. It will then travel on your network until it is sent to a vendor for termination. Just before this happens, it can be converted into the vendor-specific format.

Customer-based translation rules

Very often your customer will have his own numbering format, for example, dialing 00 for international numbers, while dialing just the phone number for local calls. Customer-based translation rules allow you to convert a number from a format specific to this particular customer. There is a special dialing rules wizard available to make such configuration easier, so that customers can even do this themselves. Customer-based translation rules have two applications:

- When a number is submitted for authorization, these rules will be applied, with the resulting number used to search rates. Thus, if your customer dials 0042021234567, you can convert it to 42021234567 and find the correct rates for the 420 prefix.
- This number will be returned to the node which requested it.

Connection-based outgoing translation rules

If your vendor requires a special number format (e.g. tech-prefix), it is possible to apply this rule to convert the number. When billing returns a list of routes, the phone numbers for routes for this connection will be converted. This only works for a routing model in which the VoIP node (e.g. PortaSIP) requests billing for routing information. If your gateway uses dial-peers or an external gatekeeper for routing, then you must configure number translation there.

Connection-based translation rules

When the call has been terminated to the vendor in a vendor-specific format, it will be reported to billing in this same format (e.g. 7834#42021234567). Now it is necessary to convert this number to the proper format for billing (4202134567), which may be done using connection translation rules. These rules will be applied to all calls which go through a given connection (even those routed there using dial-peers or other external tools).

Node-based translation rules

These serve the purpose of converting a number from a custom format used by the customer into billing's internal format during authorization, depending on the gateway. For example, on a gateway in Prague, Czech Republic, there may be the translation rule "strip leading 00", while on a gateway in Moscow, Russia, the rule will be "strip leading 810 or replace leading 8 with 7".

Since customer-based translation rules were introduced, node-based translation has become obsolete. Therefore, a node-based translation rule is applied only if there is no customer-based translation rule defined for a given customer.

Number translation on your network

Below is an illustration of how different translation rules are applied during a call.
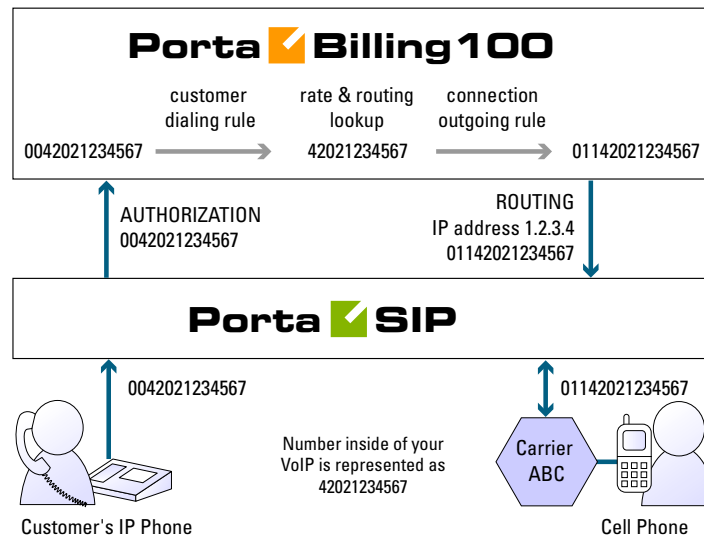
1. The customer dials a phone number on his SIP phone. He enters the number in the same format he uses on his conventional phone, i.e. 0042021234567.
2. The number is delivered to the PortaSIP server and translated using the **customer's dialing rule**, which states that the international dialing prefix for this customer is 00. So the number becomes 42021234567 (E.164 format). This number is used to search for the customer's rate for this destination.
3. PortaSIP then requests routing for this number. Carrier ABC is defined for terminating calls to the Czech Republic in PortaControl. However, this carrier requires the number to be in US dialing format, so the international number must be prefixed by 011. An **outgoing translation rule** s/^/011/; to carrier ABC has been defined, and is now applied to the phone number, with the result 01142021234567. Note that there may be several carriers who can terminate this call, each with its own numbering format. In such a case there will be several alternative routes with different phone numbers.
4. PortaSIP attempts to establish a connection to remote gateway 1.2.3.4 using phone number 01142021234567.
5. After the call is completed, PortaSIP sends an accounting request to PortaControl, stating that a call to remote gateway 1.2.3.4 has just been completed. PortaControl finds a connection to vendor ABC with remote IP address 1.2.3.4, and applies the **translation rule** s/^011//; for this connection in order to convert the number from the vendor-specific format into your billing format. Thus 011 is removed from 01142021234567, and the number becomes 42021234567. PortaControl searches for the vendor and customer rates for this number and produces the CDRs.

Active call monitoring

PortaControl can collect information about calls in progress from the PortaSIP server, Cisco/Quintum gateways and other VoIP nodes. The only thing required on the VoIP node side is the capability to send Start accounting records, in addition to Stop records.

This enables several options:
- You may see a list of calls in progress on the PortaControl web interface.
- You may forcibly disconnect a specific call (if the call is routed via PortaSIP).
- You may specify the maximum allowed number of simultaneous calls for each customer. PortaControl will reject any call attempts above the allowed number of concurrent calls.

IP device provisioning and inventory

If you provide your VoIP customers with IP phone equipment, you know how laborious and yet important the task of performing initial configuration is. If the equipment is not configured properly, it will not work after being delivered to the customer. Or, even if it works initially, problems will arise if you need to change the IP address of the SIP server. How can you reconfigure thousands of devices that are already on the customer's premises? There are two ways to manage the device configuration.

Manual provisioning

The administrator must login to the device provisioning interface (typically HTTP) and change the required parameters. There are several drawbacks to this method:
- The IP phone must be connected to the Internet when the administrator is performing this operation.
- The administrator must know the device's IP address.
- The IP phone must be on the same LAN as the administrator, or on a public IP address (if the device is behind a NAT/firewall, the administrator will not be able to access it).

Due to these reasons, and since every device must be provisioned individually, this method is acceptable for a testing environment or small-scale service deployment, but totally inappropriate for ITSPs with thousands of IP phones around the world.

Auto-provisioning

This approach is a fundamentally different one. Instead of attempting to contact an IP phone and change its parameters (pop method), the initiative is transferred to the IP phone itself. The device will periodically go to the provisioning server and fetch its configuration file.

Cisco ATA Expert

This utility allows you to simplify manual provisioning of a Cisco ATA 186/188 device and browse the device configuration in a user-friendly format. For example, instead of entering values such as 0, 1, 2, 3, etc. for codec selection, you can choose names such as "G.729" from a select menu. This tool is convenient for single-time configuration of a device, or for troubleshooting.

IP Phone Provisioning

When you use auto-provisioning for an IP phone, instead of entering the same values for codec, server address, and so on into each of a thousand user agents, you can simply create a profile which describes all these parameters. Then PortaControl can automatically create a configuration file for the SIP phone and place it on the provisioning server.
The only configuration setting which is required on the IP phone side is the address of the provisioning server, i.e. where it should send a request for its configuration file. When the IP phone connects to the Internet, it will retrieve a specific configuration file for its MAC address from the TFTP or HTTP server and adjust its internal configuration.
If you decide later to change the address of the SIP server, you need only update it once in the profile, and new configuration files will be built for all user agents. Each user agent will then retrieve this file the next time it goes online.

Porta Billing   **Provisioning server**

TFTP   HTTP
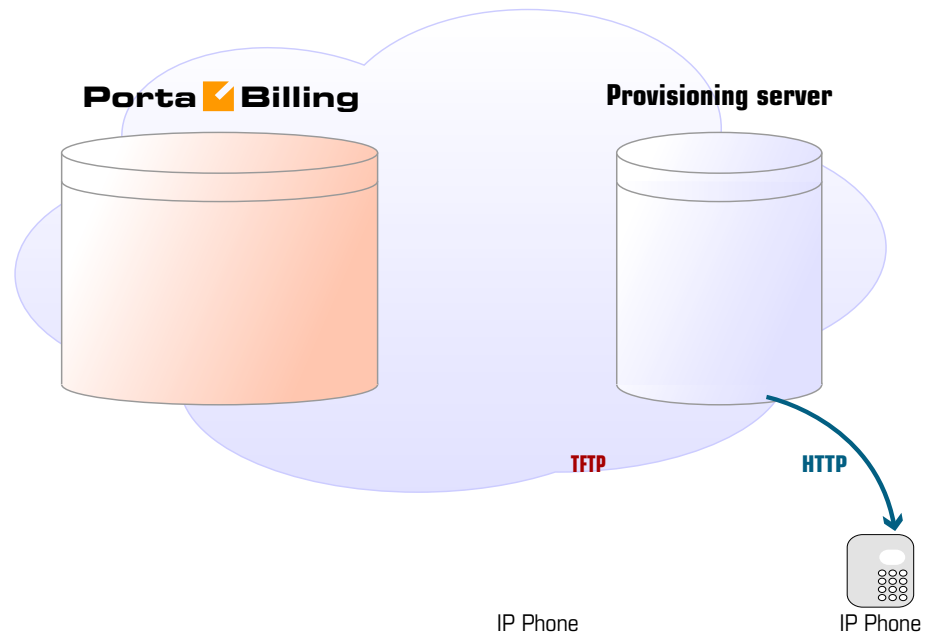
IP Phone   IP Phone

Follow-me services

The follow-me feature allows you to receive calls even if your IP phone is offline at the moment. You can specify several alternative destinations for a single destination number (account). Follow-me is activated when:

- IP phone is offline (not registered)
- IP phone replies with an error code (i.e. the line is currently busy because you are making another call)
- No answer is received within a certain interval (usually 20 seconds) – the phone may be online but nobody answers, or there is a network outage

For instance, if you do not pick up your IP phone (or the IP phone is unreachable due to a network error) the call would be forwarded to your home phone; if not answered within 30 seconds, it would be forwarded to your mobile phone, and so on. For each of these phone numbers you can define the period when a given phone should be used; for example, calls should be forwarded to your home phone only from 8 in the morning until 9 in the evening.

Billing follow-me calls

From the billing perspective, a follow-me call is treated as two separate calls. Thus, if party A calls party B, and B has a follow-me setup for phone number C, the following will occur:

1. PortaControl will check if A is authorized to call number B and for how long (based on A's rates and the available funds in A's account).
2. If follow-me is currently active on B's account, PortaControl will check if B is authorized to call C and for how long (based on B's rates and available funds).
3. After the call is completed, the two accounts are charged and CDRs are produced accordingly: one is for account A, for a call to destination B, the other for account B, for a call to destination C.

For A, this call looks like any other call made to B. If B is a number in the US, it looks like a call to the US, and A is charged according to US rates, even if the call was actually sent to a mobile phone in the Czech Republic. For B, the follow-me call is authorized and billed according to the same rules as a normal outgoing call from this account. So, for instance, if B is allowed to make outgoing calls only to US&Canada, and tries to set up a follow-me number to India, this number will not be usable. If multiple follow-me numbers have been defined, each one will be authorized independently. So if B currently has $1 available, and this is enough to make a 5-minute call to the Czech Republic or a 3-minute call to Russia, the call will be automatically disconnected after 5 or 3 minutes, respectively.

Follow-me vs. redirect number

What is the difference between the follow-me and "redirect number" properties of an account? Both seem to serve a similar purpose. However, redirect numbers, which appeared quite some time ago, had several drawbacks:

- Different gateways/applications had different kinds of support for this feature. For instance, the default Cisco debit card application did not support this feature at all.
- Using only a single phone number as a parameter did not allow flexible services.

For this reason, a new, flexible, robust solution was required, and so the follow-me feature was implemented in PortaSwitch. The redirect number feature is now obsolete, and information in the redirect number field is no longer used by PortaSwitch. PortaControl still returns the value of the redirect number in the h323-redirect-number RADIUS attribute for backward compatibility, so it can still be used by some external applications, e.g. TCL script on Cisco gateway.

Online web signup

PortaControl allows you to automate the process of subscribing new customers to the service, so that customers can do everything necessary online without requiring any assistance from your administrator. A template-based API for online signup allows you to customize page layout and put a web signup front-end on multiple websites (web portals, web pages of your resellers or affiliates, etc.). The following diagram illustrates the online web signup process:
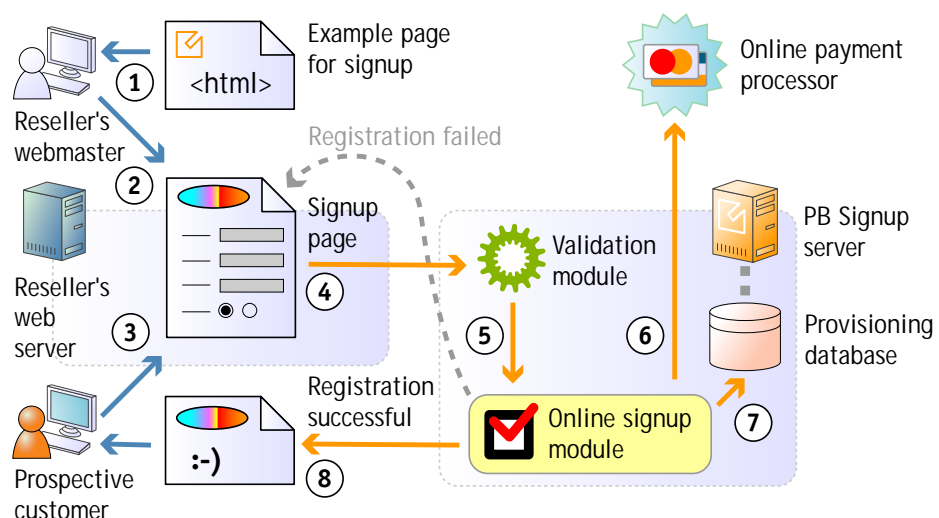
1. You are provided with a sample HTML template for the web signup front-end - the web page which will be visible to your customers (for more details, see the *How to:* section of this guide).
2. This page can be customized by the webmaster (changes in language, layout, colors, fonts, pictures); the only entities which should remain unchanged are the names of the HTTP parameters. Also, since this page is pure HTML, there are no limitations as to the web programming tools which may be used to produce the online web signup front-end: PHP, Perl, ASP, ColdFusion, and so on. After this stage is complete, the web pages are uploaded to the subscription web server.
3. When a potential customer comes to your web portal and decides to subscribe to the service, he is presented with a signup page (or pages) where he will provide his address, credit card information, and other required information. For extra convenience to the user, this process may be divided into several steps: first the user will choose a product and click Proceed, then provide his address information on the next page, and then fill in his credit card information on the third page. Finally, there will be a "confirmation" page, where the user may check to see that all the information is correct. (This page will contain all the required information, either as hidden variables or as data fields visible to the end-user).
4. When the user clicks the "Sign up" button on the signup page, his browser will send a request to the PortaControl online signup server, where this request will be processed.
5. First, the signup server loads a validation module specific to this particular web signup portal and performs a data integrity check. Validation modules can be customized by the PortaControl administrator. Generally, these involve checks as to whether all the required information (e.g. zip code) has been provided, but they may also include advanced fraud protection checks. For instance, signup may be rejected if someone with a Pakistan IP address attempts to sign up as a customer with a US billing and shipping address.
    If the validation check fails, the user's web browser is redirected to the initial signup page (with the error code passed as a parameter). Thus he will see the same web pages as before, with a message explaining the error, and may be given a chance to re-enter his information.
6. PortaControl contacts the online payment processor to charge the customer credit card for the required amount. If the transaction fails (e.g. invalid card number, no funds available) the user is once again redirected to the initial signup page.
7. At this stage, new customer and account objects are created in the database (this may involve allocating a new phone number for the account from among those available). Optional actions (e.g. enter a record to the provisioning database that a certain IP phone

must be shipped to the customer via FedEx/UPS) may also be done at this point.

8. The user is redirected to the web signup front-end, with important service attributes (e.g. assigned phone number, password, etc.) being passed as parameters. This request is processed by the front-end site's "signup successful" script, and the user sees a page informing him that the signup process has been completed and providing further information about the service (e.g. his phone number).test

Consequently, even if PortaControl participates in the process, it remains invisible to the end user. The user accesses the web front-end, fills in the required information, clicks "Sign up", and then, after a short pause, is shown either a "Congratulations!" or "Sorry, there was an error!" page on the same web portal.

Custom reports

If your business process requires that specific information from PortaControl be presented in a certain format (e.g. your marketing department needs to know the top 10 countries based solely on calls longer than 15 minutes, including the ratio between these calls and all other calls for each country), and this information is not available among PortaControl's default reports, then there are two options available:

- Create your own custom report using tools such as Crystal Reports (for more details, see the *How to …* section). This assumes that you have sufficient experience with relational databases and reporting tools.
- Contact the PortaOne team, so that a custom report can be developed for you and uploaded to your system. In this case, no technical knowledge on your side is required.

PortaControl's custom report module allows the creation of "report" objects, which can later be uploaded to your system and then executed by your administrators. In this way, you can maintain an unlimited number of different reports specific to your business.

Report type

Report type defines the purpose of a report: does it show the most popular destinations, or the customers who make the most calls? Several types of reports are already supplied with PortaControl, including:

- Most popular destinations of a specific customer
- Total amount of minutes and charges for a specific customer

Other types of reports can be developed on request. When a new report object is uploaded to your system, a new type of report will appear. Report type defines:

- What types of input parameters are required by the report (e.g. customer name, time interval, etc.).
- What the output values of this report are. Since the report results are displayed as a table, this defines what the columns represent in this table, while every row in the table represents another line of report data.

Report query

Whereas report type defines a report's general characteristics (e.g. "Most popular destinations"), by executing a report query you may obtain a specific result (e.g. "What were the most popular destinations for customer John in the last two weeks?").

Sometimes you will simply execute a query once (for instance, you would like to quickly verify that a particular customer did indeed make more than 1,000 calls yesterday), while other types of queries will be made routinely (e.g. every morning your sales manager checks the previous day's statistics for his VIP customers). To accommodate both types of needs, PortaControl allows you to create a new query and then:

- Run it once in your web browser to see the results immediately.
- Save the query (including input parameters) for later use. When you need to re-run the query sometime in the future, you simply open and execute it, i.e. you do not have to fill in any parameters. This significantly reduces the time required to prepare complicated reports, and allows you to minimize the chance of error. The administrator creates the query just once, and then inexperienced users need only execute it.

Also, to save you time when preparing multiple similar reports, PortaControl allows you to create a query as a copy of another existing query. After you have defined a complex report with many input parameters for customer A, simply create a copy of it and change the customer name to "B". Now you have reports for both customers, either of which can be executed with just one mouse click.

Query execution

There are several methods for executing a query and obtaining report results:

- You may request query execution from the PortaControl web interface, and see the result in your web browser.
- You may request that a query be executed in the background, with the results delivered to you via email. This is the preferred method for complicated reports involving large arrays of data, since in this case PortaControl can schedule query execution in such a way that it does not interfere with other tasks. This allows you to avoid performance degradation when several users are trying to execute

reports concurrently; PortaControl will execute all these reports sequentially.

This is the recommended method for executing reports.

- You may schedule a query to be executed at a certain moment in time or periodically, with the report results delivered to you via email. This allows you to automate reports that you use on a continuous basis. For instance, you may schedule a "Total minutes per customer" report for 3am every day; then every time you open your mailbox in the morning, you will find all the required information. You can view a list of scheduled query executions and their status on the PortaControl admin interface.

Query results

When you execute a report query from the web interface, its result is stored in PortaControl. Thus if you did not save the results initially, you need not run the report again, as the data for the last ten report executions is always available.

2. Entering Data in PortaControl

Cost / revenue

There are two independent parts of the billing configuration:
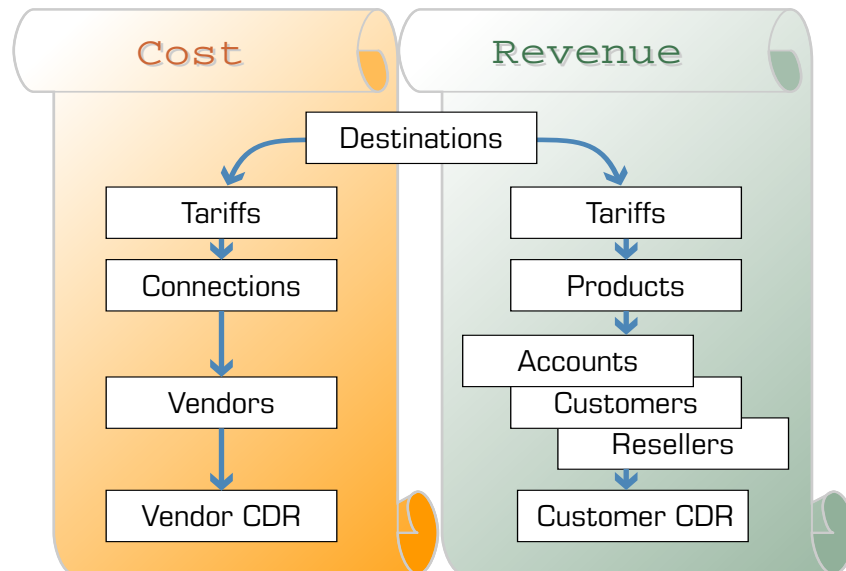
The Cost part provides you with information about expenses associated with running your VoIP business: call termination, incoming toll-free lines, etc. The Revenue part calculates charges applicable to your customers. Note, that these parts are independent (the only common item is Destinations). So, the amount charged to your customer is unrelated to your termination expenses and vice versa.

PortaControl data model concept

The illustration below shows some major elements and their interrelationships. For the sake of clarity, only the most important tables and columns are shown.
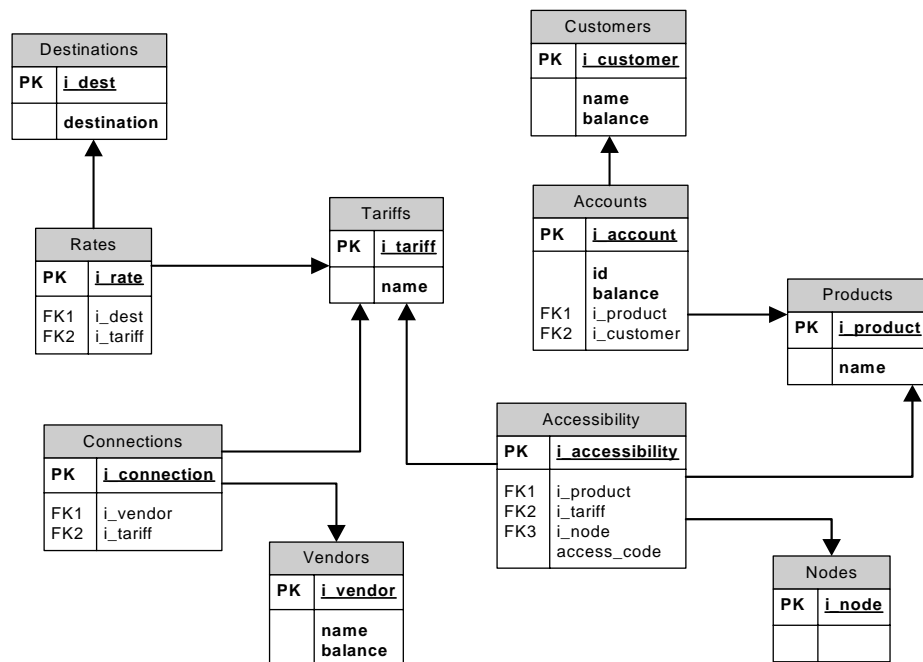
- Destinations define possible phone number prefixes which will be used in the system.
- Rate tables store information about rating parameters for a specific destination with a specific tariff.
- Tariffs serve as a pricelist to be assigned to a specific customer or vendor
- The relationship between vendors and tariffs is defined via connections. A tariff assigned to a specific connection will be used to calculate termination costs for a given vendor.
- Every account is assigned a certain product, which defines how the account will use the service and how it will be charged for this.
- A product is related to tariffs via an accessibility table, so that a single product may be linked with multiple tariffs.
- Accessibility specifies which tariff is applied to the account for calls made on a specific node (gateway) and access number.

Billing entities hierarchy
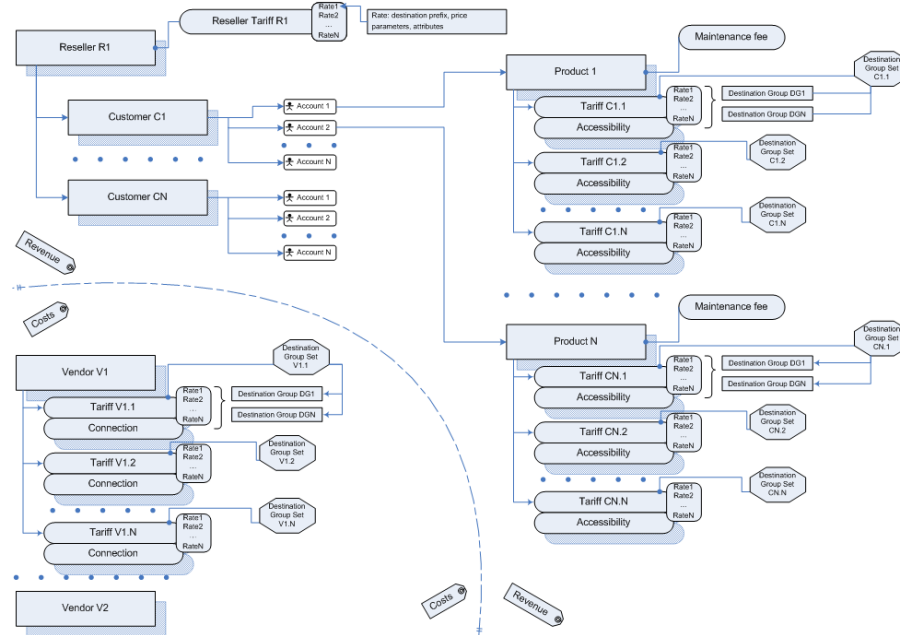
The diagram above presents a hierarchical overview of the billing information stored in the system. On the cost side, under vendors, are located connections, with a tariff associated with each connection. On the revenue side, accounts are grouped under a customer (while a customer can, in turn, be placed under a reseller). The account is linked with the tariff (rate plan) via its product; note that this relationship is "one to many", so that different tariffs can potentially be applied to the same account.

Step 1 – "Your Network"
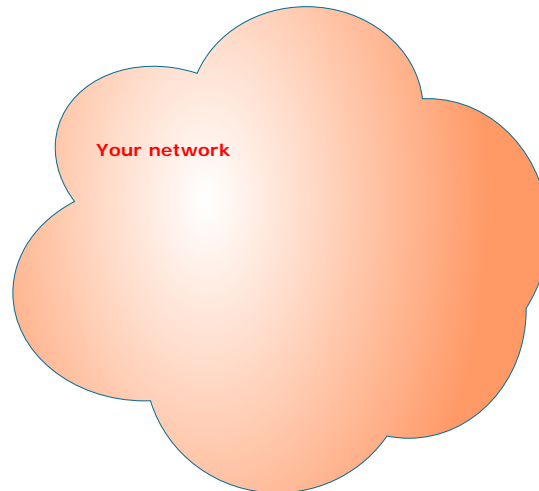
PortaControl is designed for an IP telephony service provider which we will call the **Company**. All information kept within the system is used to calculate the financial status between the Company and its **Vendors** and **Resellers** or **Direct Customers**.

Step 2 – "Nodes"
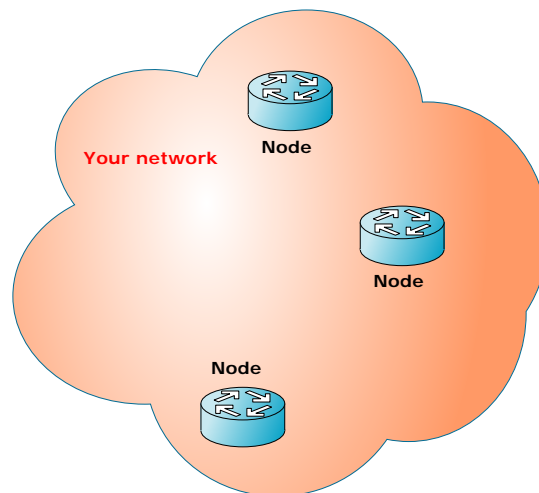
The **Network** is defined as all network elements, or **Nodes**, that belong to the Company. Gateways in the Company's network are "trusted" gateways. These gateways must be listed in the Nodes registry, and are also required to be Radius clients so that PortaControl may receive accounting information.

Step 3 – "Vendors and Resellers (Customers)"
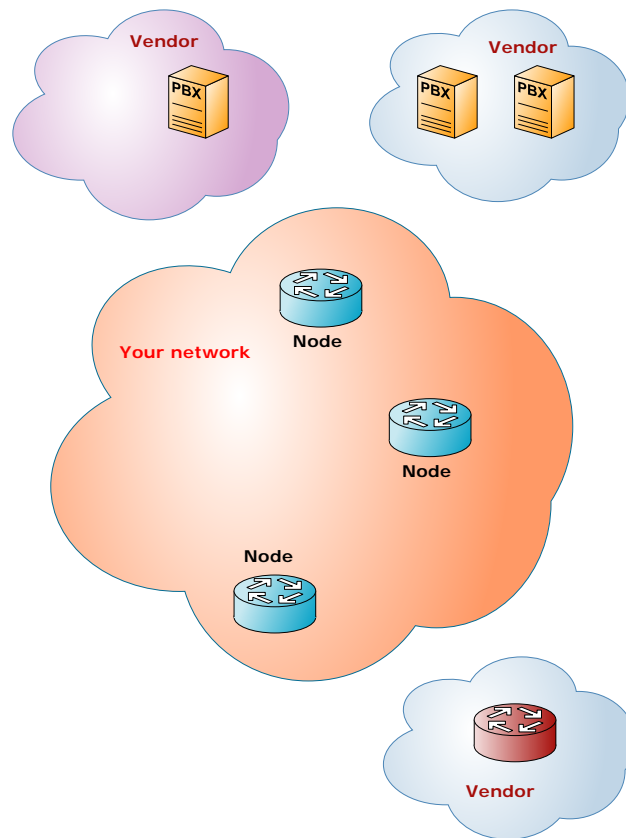
Vendors are those partners which provide VoIP traffic termination services to the Company. If the Company sends traffic to another service provider, then, in this case, the other entity is defined as a Vendor. Therefore, if there is bilateral traffic between the Company and another firm, the second firm must be registered as both a **Vendor** and a **Customer**.

Step 4 – "Connections"
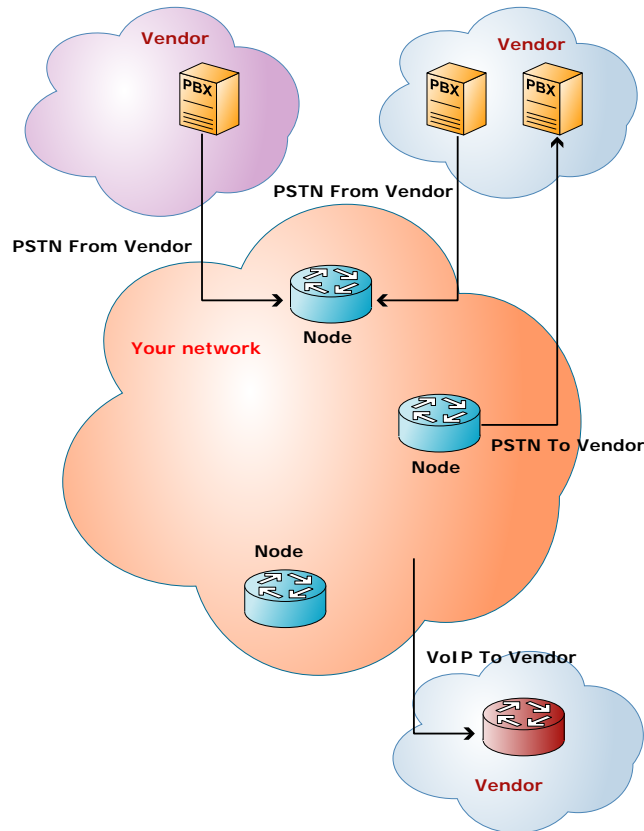
A **Connection** is a precise description of a point of contact between the Company's Network and the Vendor's. It is the point at which costing occurs and revenue is generated. There may be multiple Connections to the same Vendor, and each may, for example, have a different price. All calls within the Company's network are free, as they do not utilize Vendor Connections. The different types of Connections are categorized as

- "VoIP to Vendor", "PSTN to Vendor" – outgoing calls
- "VoIP from Vendor", "PSTN from Vendor" – incoming calls

Sometimes two entities are confused:

- The vendor who delivers calls from **your** customers to your network (providing you with a toll-fee line or DIDs) – this is a VoIP or PSTN from vendor connection
- The vendor who sends calls from **his** customers to your network for termination – in this case he is just another one of your customers, and not a vendor

Step 5 – "Vendors and Customers Part II"
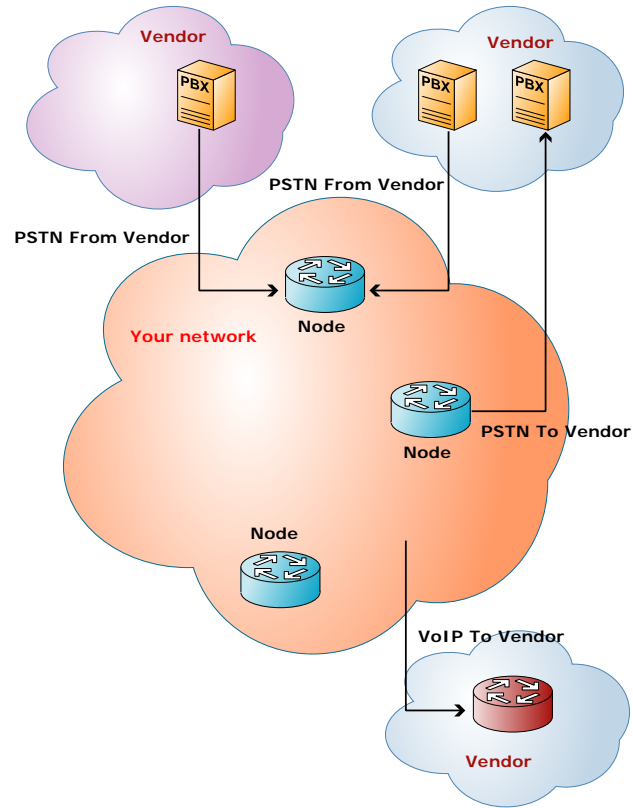
Step 6 – "Accounts"
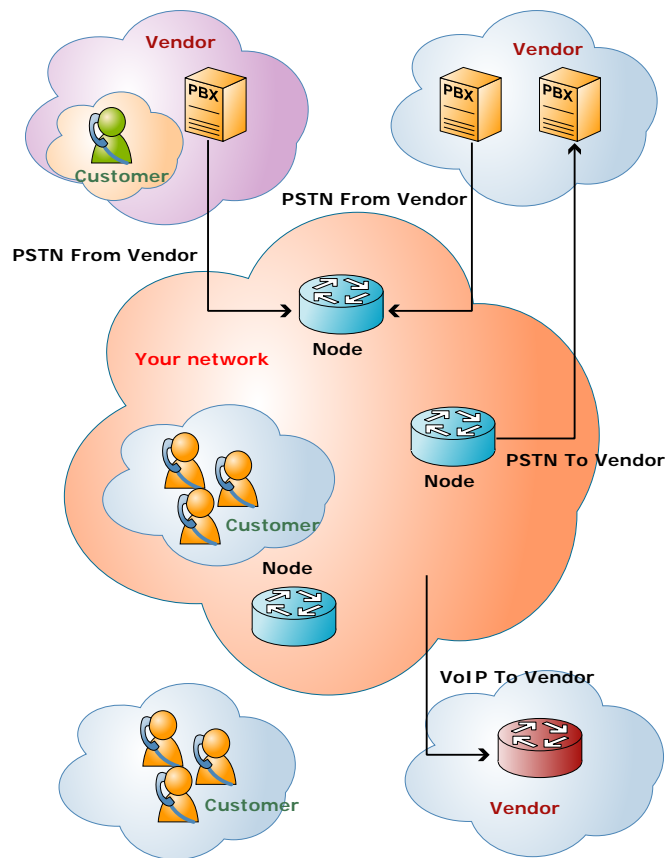
A Customer is composed of one or more **Accounts**. These Accounts may represent individual users of a product, such as prepaid cards or IP phones; or they may identify remote gateways.

Billing concept

Collecting data from the VoIP network
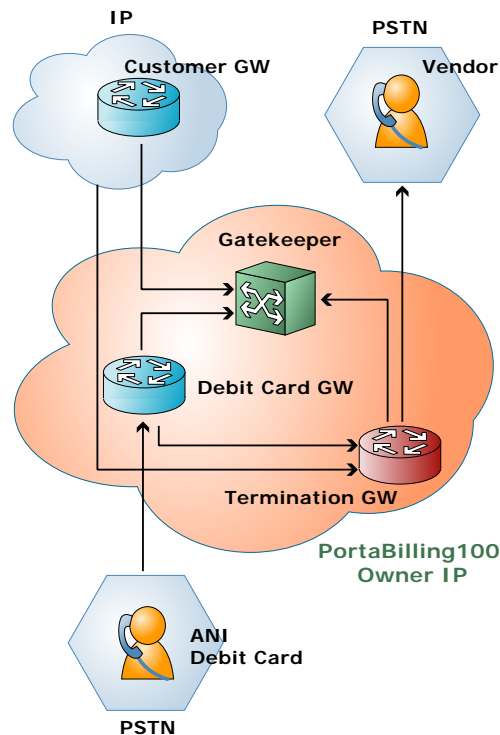
Gateways in the PortaControl owner network are "trusted" gateways. These gateways MUST be listed in the Nodes directory and be a Radius client.

Gateways which belong to your customers or vendors should not be created as nodes. Instead, a remote GW which belongs to your customer will be represented in the system as an account.

If the termination GW transmits an accounting record (connection matched), the User-Name must be verified to see if a trusted GW (Node IP=User-Name) is calling. If so, then the billing engine must wait for an accounting record from the trusted GW and use the User-Name from a trusted GW accounting record. If a trusted GW is not found, the call will be charged immediately to an applicable account. If an account is not found, a "security breach" mail alert will be sent.

Using different protocols (SIP and H323)

The billing itself is independent from the actual implementation details of those protocols used to carry out the call. PortaControl communicates with the nodes on your VoIP network (gateways, gatekeepers, proxies) using the RADIUS protocol and the VoIP VSA extensions for it. Thus, from the point of view of PortaControl, RADIUS requests related to SIP calls are nearly identical to those related to H323 calls (there are only a few different attributes).

PortaControl is able to process both types of requests, so no special configuration in billing is required to support SIP or H323 calls. You only need to add your gateway to the system as a trusted node, and then both H323 and SIP calls reported from that gateway will be processed properly.

Note that from the billing point of view, a SIP server is not different from any other gateway - it also issues authentication or authorization requests, and sends accounting information after the call is completed.

A node for the SIP server should therefore be created in your system, so that it can communicate properly with the billing. Also, products that are to be allowed to use SIP services should have an Accessibility entry which includes this node.

3. Statistics and Monitoring Tools

Billing server health monitoring

Entire system load

These statistics are accessible from the main admin menu via the **System load** link in the **Statistics** section.

This graph shows the number of calls registered by a production system every 15 minutes: normal calls with duration >0 (green areas) and calls with zero duration (red line). The most recent information appearing on the right-hand side of the graph is an hour old or less.
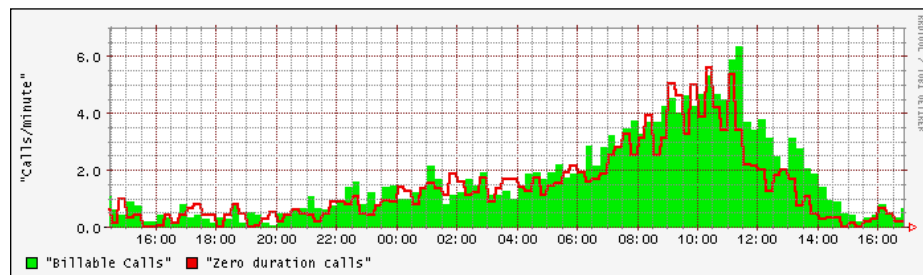
Figure 3-1

Number of calls

If the disconnect time on the CDR does not fall within the past 15 minutes, the call may not be finished, and hence will not be reflected on the graph. (See picture below, call types 3 and 4.)

Zero duration calls

The graph can also show possible problems in the system, such as an unexpectedly high number of failed calls.
By default, this graph displays statistics for the last 30 hours.

Total minutes statistics

The graph below gives you the ability to monitor how the call volume in your system changes each day.
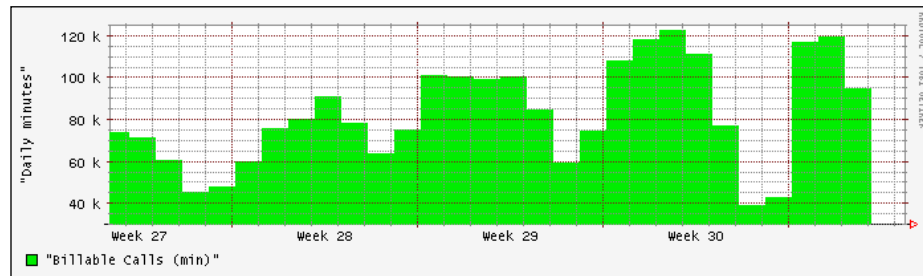
Master and slave MySQL server statistics

These statistics are accessible from the main admin menu via the
Database link in the **Statistics** section.
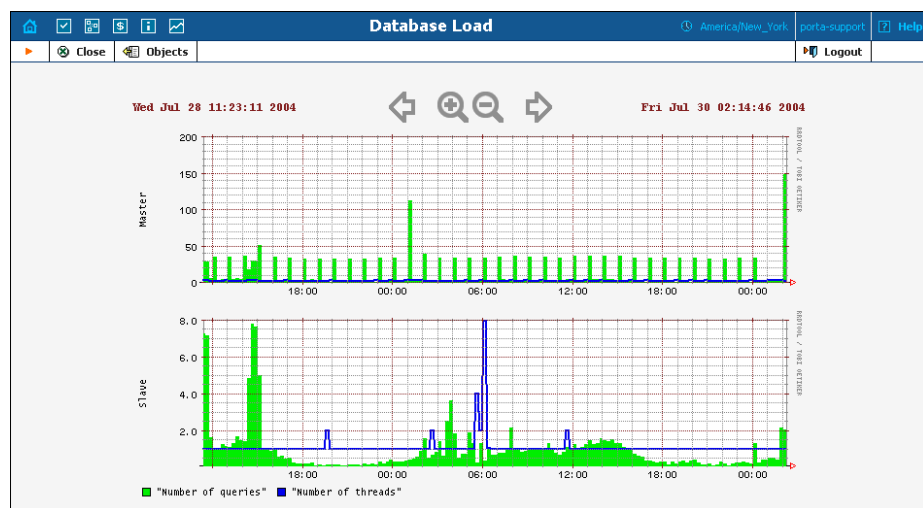
Figure 3-2

These two graphs show values for the status of the MySQL servers. The
green line indicates the number of queries processed by the server every
15 minutes, while the blue one shows the number of threads running on
the server.

Queries
The graph indicates how many times clients have queried the database.
On recommended hardware, this value may exceed 200 without any
difficulties. High peaks with a number of requests over 2,000 could
indicate improper configuration of the system or temporary problems.
Also, these peaks can appear when replication has been restored after a
system fault.

Threads
Normal values for these graphs are:
- for the Master server: <=2
- for the Slave server: >=1 and <=3

Normally, these values will depend on the processes running on the server, such as the Radius daemon, the Apache httpd server, statistics collection tools, and so on.

Larger values can indicate other client connections to the server. If the number of threads on the slave name server is 0, this means the replication process is down and requires administrator attention (see section 2.4 for details).

VoIP network performance statistics

Connection load

You can monitor load and quality parameters for individual connections. To see a list of the available connections, follow the **Connections** link from the **Statistics** section in the main menu.
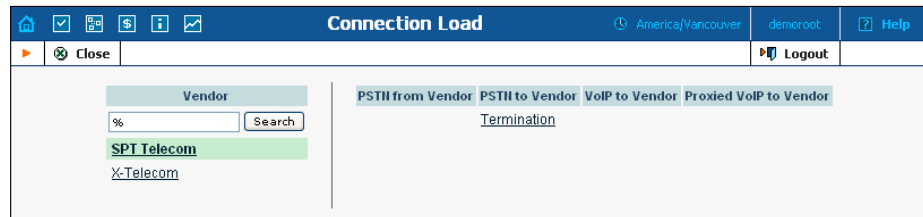
After you select a vendor and a particular connection, you will see its load graph:
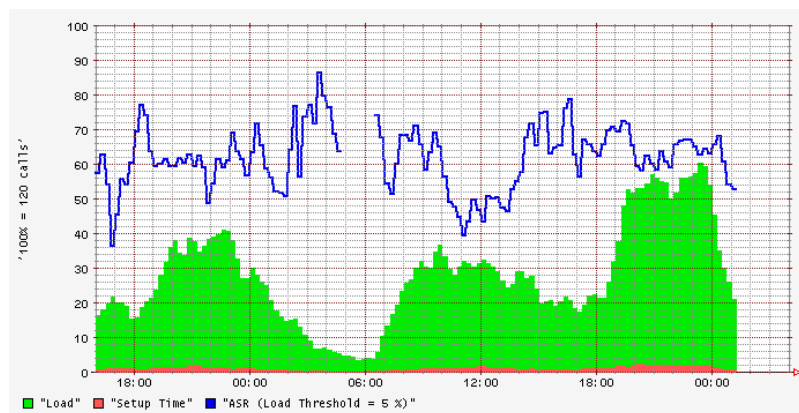
This graph shows the node load, setup time and ASR (Average Success Rate) statistics. The green area indicates node load, the red shows relative setup time, and the blue line indicates ASR. Information appears on this graph with a one hour delay.
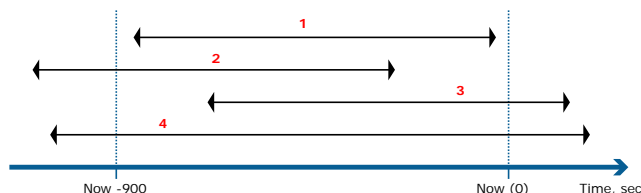
Assumptions: Calls with a duration of more then 3,600 seconds are rare, and do not affect our statistics very much; zero duration calls need an average duration of 10 seconds to be processed by the router.

Connection load calculations

For a better understanding of the calculations performed, consider the following diagram:
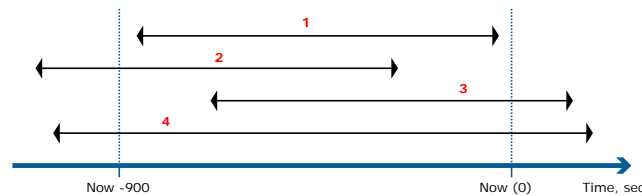
The total duration must be calculated for all calls processed by the system corresponding to the interval between "now - 1 hour 15 minutes", and "now - 1 hour".

Four different call types are recognized in the above diagram; an additional call type is "zero duration":

1. Sum of durations of all the calls.
2. Sum of durations between `t0` and `disconnect_time`.
3. Sum of durations between `connect_time` and `t1`.
4. Calculate sum as number of calls * 900 seconds.
5. Calculate sum as number of calls * 10 sec.

The maximum value in seconds which a connection can hold in the 900 sec. interval is calculated as capacity * 900sec. The connection load is calculated as the sum of durations for all call types divided by the maximum value in seconds. The result is shown as a percentage value (multiplied by 100).

ASR calculations

Assumption: Connection load less than 5% is not representative for ASR calculating.

ASR is calculated as the number of type 5 calls divided by the total number of calls. The result is shown as a percentage value (multiplied by 100).

Relative setup time calculations

Relative setup time is calculated as the ratio between the total setup time of all calls during the given period and the maximum value in seconds (capacity * 900sec).

ASR statistics

Follow the **ASR** link from the **Statistics** section in the main menu.

You can download pre-calculated statistics in .CSV format (for previous days) or obtain data online using **Custom Query**. For pre-calculated statistics, click on the vendor at left, then choose the statistics period from the calendar. Pre-calculated statistics will look like the following:

The following data is available:

- Total number of calls
- Number of calls with non-zero duration (billable calls)
- ASR
- Total call duration
- Average Length of Call (ALOC)

Numbers are aggregated per destination prefix, with a subtotal per country and a total for all calls.

Should you need statistics for today, or a report with certain other parameters (for example, divided by hour, so that you see how the ASR

evolved during the day), you can use the **Custom Query** report. This extracts data directly from the database, so use it with caution in order not to overload the server.

Note that both ASR and Cost/Revenue parameters are shown on the **Custom Query** report.

Billing statistics

Customer CDRs

Lists of all customers' CDRs are calculated daily, and are split into files according to each customer's billing period. These pre-calculated statistics can be automatically mailed to the customer. They are also available for download by the customer on the self-care pages, and for your staff on the admin interface.

Different types of CDR files are available, depending on the type of customer:

CDR files for retail customers

- **Invoice** – Calls made by credit accounts of this customer. These calls will be included on the customer's invoice for the corresponding billing period.
- **Debit** – Calls made by debit accounts of this customer. Since debit accounts are prepaid, calls made by them do not affect the customer's balance or invoice. Therefore, they are included in a

96

separate file so that the customer can easily monitor activities on his debit accounts.

CDR files for resellers

CDRs calculated using the reseller's wholesale tariff. These are the charges applied to the reseller and reflected on his invoice.

Choose the customer and then click on the calendar to obtain statistics for desired period. The .CSV file will have the following structure:

The following data is available:

- Account ID (if applicable; not available for a reseller's CDRs, but available for CDRs of a reseller's sub-customers)
- CLI or ANI (From)
- CLD or DNIS (To)
- Country and description of destination
- Call start time

- Charged time (in minutes:seconds)

NOTE: When browsing data in Microsoft Excel, extra trailing zeros might be added, for example **345:37** (three hundred forty-five minutes and thirty-seven seconds) will be shown as **345:37:00**. Use proper call formatting in Excel to eliminate this problem.

- Charged time (in seconds)
- Charged amount

NOTE: The call duration shown in the file is based on the charged duration of the call, not the actual call duration. So if, for instance, you bill by 6-second intervals, the total call duration will be higher than the actual duration of the calls.

Vendor CDRs

It is very useful to have a detailed list of calls for a specific vendor, in case of disputes over the amount of terminated traffic. These statistics are calculated daily, and are split into files according to the vendor's billing period.

Choose the vendor and then click on the calendar to download statistics for the desired period. The .CSV file will have the following structure:

The following data is available:

- CLI or ANI (From)
- CLD or DNIS (To)
- Country and description of destination
- Call start time
- Charged time
- Charged amount

Cost/Revenue statistics

These are essential tools for monitoring the growth of your business. Working with different partners, different currencies and different prices, it is very easy to make mistakes and carry out non-profitable calls.

Cost/Revenue reports allow you to monitor this and stay out of trouble. To access Cost/Revenue reports, follow the **Cost/Revenue reports** link from the **Statistics** section of the main menu.

There are five variants of pre-calculated reports:

- By customer and destination, subtotal per country
- By customer and destination, subtotal per customer
- By vendor and destination, subtotal per country

- By vendor and destination, subtotal per vendor
- By destination

The following data is available:

- Name of customer/vendor
- Destination prefix
- Country and description of destination
- Total number of successful calls
- Gross margin – The difference between total revenue and the cost for this destination
- Rated, sec. – The sum of the time charged for all calls to this destination in seconds
- Rated, min. – The sum of the time charged for all calls to this destination in minutes. This is the same value as in the previous column, only expressed in different units. Thus, for instance, if *Rated, sec* contains **180**, here **3.0** will be shown.
- Total Cost – Summary of the amount charged for all the vendor's CDRs for this destination. If the vendor uses a currency different than your base one, this will be converted using the current exchange rate.
- Total Revenue – Summary of the amount charged for all the customer's CDRs for this destination. If the customer uses a currency different than your base one, this will be converted using the current exchange rate. Note that, in the case of resellers, we use the value of the reseller's CDR, not the sub-customer's, since your revenue is what gets invoiced to the reseller.

In addition, you can use the **Custom Query** report, which is identical to the **Custom Query** report in the ASR statistics.

4. How to …

Login into PortaControl after the installation

The default username is "pb-root" and the password is "pb-root" as well. After you login for the first time, you will be requested to change the password – please remember the new password you choose.

Charge my calling card customers XX/min extra when they call a toll-free line

This is very easy to do with PortaControl's **Accessibility** feature:

1. Create two tariffs – one with your normal rates (for example 0.13 dollar/min for calls to Czech Republic) and the other one with "toll-free" rates, including your toll-free costs there (for example, 0.17 dollar/min for calls to Czech Republic).

TIP: When calculating the price for the toll-free line, it is not enough to add together your ordinary price plus the costs of one minute on the toll-free line. In order to make a 3-minute call, the customer will spend about 4 minutes on the line while listening to voice prompts, entering a PIN / destination and waiting for an answer. Additionally, situations where a customer will enter an incorrect PIN or be unable to reach his party at all should be taken into account. Therefore, usually:
TollFree_ Price = Ordinary_Price + X*Toll_FreeCost
where **X** is a ratio between the total duration of incoming toll-free calls and the total duration of outgoing calls.

2. Make sure your IVR script supports the "Access Code" feature (using the "PortaBilling Original CLD" RADIUS attribute). PortaOne's TCL developers can help you add this feature to your application.

3. Create a product for your calling cards and make two entries in Accessibility:
   - one with an **Access Code** equal to your toll-free number and "Toll-free tariff"
   - another with an **Access Code** equal to your local access number and "Ordinary tariff".

Authorize and bill my customers by the phone number they are calling from (ANI-based billing)

PortaControl gives you great flexibility in choosing how you would like to authorize and bill your customers. For ANI-based billing, you only need to do the following:

1. Create a tariff (or tariffs) and a product.

Note: If you are providing both prepaid cards and ANI-based billing, take measures to prevent fraud (e.g., someone could dial your IVR and enter their neighbor's home phone number as the PIN). See the *Prevent an ANI number from being used as a PIN* topic below.

2. Use the corresponding application on your gateway to handle the call. You can use one of the default Cisco applications (clid_*), create your own, or use PortaOne's "Advanced Remote Authenticate" script (available at **http://store.portaone.com**). The only important thing is that ANI (CLI) must be in the User-Name attribute in the AAA requests which go to the billing.

None of the default Cisco applications support call authorization, so the gateway will not consult billing as to whether the customer is allowed to call this particular destination and what is the maximum allowed call duration. Needless to say, this may lead to abuse of your service; moreover, there are several other drawbacks to the default Cisco applications. In order to avoid all these problems, it is recommended that you use PortaOne's "Advanced Remote Authenticate" script, or (if you need advanced IVR capabilities such as "announce available balance or number of minutes") modify the debit card application.

Here is a sample:

```
aaa new-model
aaa authentication login h323 group radius
aaa authorization exec h323 group radius
aaa accounting connection h323 stop-only group radius
!
gw-accounting h323 vsa
!
call application voice ani tftp://…/portaone.tcl
call application voice ani authenticate-by ani
call application voice ani skip-password yes
call application voice ani authorize yes

!
dial-peer voice 1 pots
 application ani
 incoming called-number .
```

```
 port 1:D
!
gateway
!
```

3. Create a customer who will own these accounts.

4. Create accounts with an Account ID identical to the phone number from which the service is to be used.

TIP: Check in which format ANI (CLI) numbers are reported by your gateway. For example, the phone number 42021234567 might be reported as "21234567", "021234567", or something different. You must use exactly the same format for the Account ID (or change your application so as to convert it to the desired format). Only one extra item is required in the application configuration for PortaOne's "Advanced Remote Authenticate" – the translate parameter. The following example assumes that phone numbers arrive in the local format, prefixed by 0, e.g. 021234567 instead of 42021234567:

```
call application voice ani translate "/^0/420/"
```

Bill customers who are connected via T1/E1 directly to a port on my gateway

Typically, you do not use authentication or authorization for such "port-based billing", since you are always sure that your customer is on the other end of the physical line. Although it might be a good idea to implement call authorization so that you can control which destinations the customer is allowed to call to, we will not discuss this here. For port-based billing, you need only do the following:

1. For each such customer, create a tariff that you want to use to bill the customer and a product. Only one row in Accessibility is necessary, with the node "ANY" and the tariff you have created.

2. Now you must make sure that each call made by that customer is tagged as belonging to him. Use the corresponding application on your gateway to handle the call. You can use the PortaOne's "Session name" application, or create your own.

The often difficult part in gateway configuration is designing dial-peers to match certain voice ports. Dial-peers should resemble the following configuration:

```
dial-peer voice 1 pots
    application client_on_port_1
    direct-inward-dial
```

```
    port 1:D
```

They will be listed with an operation status of down, as the port specification must be accompanied by a number matching a specification, for example:

```
dial-peer voice 1 pots
    application client_on_port_1
    incoming called-number .
    direct-inward-dial
    port 1:D
```

This dial-peer is not adequate for our needs, because any number matching the specification will prefer the port specification, i.e. this dial-peer will match any port. The solution is to use a number-matching scheme that will not match any number.

```
dial-peer voice 1 pots
    application client_on_port_1
    incoming called-number A
    direct-inward-dial
    port 1:D
```

This solution may be considered a "hack", yet it is functional and secure. For increased security, you may specify a 32-character string with a random sequence of "ABCD" characters. The probability of receiving such a number is near zero. So the configuration should look as follows:

```
aaa new-model
aaa authentication login h323 group radius
aaa authorization exec h323 group radius
aaa accounting connection h323 stop-only group radius
!
call application voice client_on_port_1
flash:app_session_name.2.1.0.tcl
call application voice client_on_port_1 user-name abc_ltd
!
call application voice client_on_port_2
flash:app_session_name.2.1.0.tcl
call application voice client_on_port_2 user-name xyz_inc
!
gw-accounting h323 vsa
!
dial-peer voice 1 pots
    application client_on_port_1
    incoming called-number A
    direct-inward-dial
    port 1:D
!
dial-peer voice 2 pots
    application client_on_port_2
    incoming called-number B
    direct-inward-dial
    port 2:D
!
gateway
!
```

1. Create customers who will own these accounts.

2. Create accounts with an Account ID identical to the name you entered in the configuration for the applications (**abc_ltd** and **xyz_inc** in our example).

Authenticate and bill my customers by the IP address of their gateway

This is another example of how easy it is to implement different billing schemes with PortaControl. For IP-based billing, you just have to do the following:

1. Create a tariff and a product. Since "Node" and "Access Code" only make sense for incoming telephony calls, set node to "ANY" and leave Access Code blank for the IP-based billing product.

2. Use the corresponding application on your gateway to handle the call. You can use the Cisco application remote_ip_authenticate, PortaOne's "Advanced Remote Authenticate" (recommended), or create your own. The only important thing is that the IP address of the remote GW be in the User-Name attribute in the AAA requests which will go to the billing. Here is a sample configuration:

```
aaa new-model
authentication login h323 group radius
aaa authorization exec h323 group radius
aaa accounting connection h323 stop-only group radius
!
gw-accounting h323 vsa
!
call application voice remote_ip tftp://…/portaone.tcl
call application voice remote_ip authenticate-by ip
call application voice remote_ip_auth password cisco
!
dial-peer voice 11 voip
  application remote_ip
  incoming called-number .
!
```

3. Create a customer who will own these accounts.

4. Create accounts with an Account ID identical to the IP address of the remote gateway, and enter cisco as the VoIP password for this account. (The password cisco has been chosen only for backward compatibility with the Cisco remote_ip_authenticate script. With PortaOne's "Advanced Remote Authenticate" you can use a different password; just change the list in the application config).

Charge a reseller for incoming calls

A very common situation is that in which you provide the reseller with billing, gateways and other required components. When an end-user makes a call, you charge the reseller according to your wholesale rates. But now the reseller wishes to give his subscribers the ability to use either a local access line or a toll-free line. The reseller can use the accessibility feature to charge subscribers by different tariffs, depending on the access number. But what amount do you charge the reseller? Obviously, when somebody calls a toll-free line and then calls to China your cost is higher than when somebody calls a local access line and then calls the same number in China. This fact should be reflected in the reseller charges.

PortaControl allows you to charge your reseller not just for outgoing calls, but also for incoming ones. This is done when the call crosses PSTN from the vendor connection. In order to implement this, follow these instructions:

1. Create a tariff which defines your incoming line costs.

2. Create a separate tariff which describes charges for incoming access numbers, and which you would like to apply to your reseller (for instance, incoming calls to 1800 numbers will cost your reseller 0.02/min, while calls to 1718 and 1206 numbers will be free).

3. Activate the "incoming charges" feature (disabled by default).
    *   On the PortaControl master, edit the `/home/porta-billing/etc/porta-billing.conf` and make sure the following line is present in the `[Features]` section:
        `ChargeCustomerForIncomingCalls=yes`
    *   On the PortaControl slave, edit the /home/porta-admin/etc/porta-admin.conf and make sure the following line is present in the [Customers] section:
        `ChargeCustomerForIncomingCalls=1`

4. Define a **PSTN from Vendor** (or a **VoIP from Vendor**) connection, which will describe the point where calls are delivered to your network from the vendor, and therefore have costs associated with them. Associate the tariff created in step 1 with this connection.

5. Edit your reseller information. Now, on the **Additional Info** tab, you will see a new select menu – **Incoming tariff**. Choose the tariff you created in step 2, then click ◼ **Save&Close**.

Now, when an account belonging to the reseller makes a call to the toll-free number 18001234567 (and there is a matching PSTN from the

vendor connection), then makes an outgoing call to the Czech Republic (42021234567) and hangs up, the following charges will be applied:

- The account will be charged for an outgoing call to 42021234567 according to his rates to the Czech Republic; a CDR will be written to the database.
- For the vendor who terminated this call to the Czech Republic, a CDR will be written which describes your termination cost.
- For the vendor who provided the toll-free line, a CDR will be created which describes your incoming call cost.
- For the reseller, two CDRs will be created (and his balance will be modified accordingly):
    - for the call to 42021234567, according to his "outgoing" tariff (wholesale rates)
    - for the call to 18001234567, according to his "incoming" tariff.

    Please note that these two calls will usually have a different duration. The incoming call will be longer, because the customer needs time to enter the prompts, wait to be connected, and so on. In this case you charge the reseller for the actual incoming call duration, i.e. exactly the same duration you are charged by the toll-free line vendor.

Deal with technical prefixes and numbering formats

Different termination partners often require that you send them numbers in some specific format. For example, your termination partner might require usage of the technical prefix *58901#* (so you have to send him 58901#42021234567 instead of 42021234567), while your local phone provider requires that outgoing call numbers be dialed without the country code (so a call to 42021234567 has to be dialed as 21234567). The more partners you have, the more likely it is you will run into these problems with different numbering formats. Of course it is better to use a single numbering format internally. Our recommendation is to use the E.164 format in billing and on your network. An E.164 compliant number is one consisting of *<countrycode><areacode><phone#>*. Examples of valid E.164 numbers are ***42021234567*** and ***16049876543***. The following numbers are not valid E.164 numbers: 6049876543 (NANP format), 01142021234567 (US overseas dialing format), 021234567 (local format).

PortaControl provides you with a powerful tool for converting all outside numbering formats into the unified format which is to be used in billing. For every connection to a vendor you can specify a translation rule that

will convert the number into E.164 format. Here are some ready-to-use examples of translation rules:

- Convert NANP (North American Numbering Plan) phone number (area code + phone number, e.g. **604 888 7766**) into E.164:
  ```
  s/^/1/;
  ```
- Convert European international dialing format (00 + country code + area code + phone number, e.g. **00 1 604 888 7766**) into E.164:
  ```
  s/^00//;
  ```
- Convert North American international dialing format (011 + country code + area code + phone number, e.g. **011 1 604 888 7766**) into E.164:
  ```
  s/^011//;
  ```
- Convert Australian international dialing format (0011 + country code + area code + phone number, e.g. **0011 1 604 888 7766**) into E.164:
  ```
  s/^0011//;
  ```
- Convert tech prefix format (tech prefix + country code + area code + phone number, e.g. **6789# 1 604 888 7766**) into E.164:
  ```
  s/^6789#//;
  ```
- Convert European domestic dialing format (0 + area code + phone number, e.g. **0 5 888 7766**) into E.164 (assuming that the country code is **44**):
  ```
  s/^0/44/;
  ```

Always test your translation rules before entering them into the billing. A special test window is available on the web interface. To access it, click on the test icon ⍰ next to the translation rule field.

This will enable you to check if your translation rule has the proper syntax, and you can also immediately see if it performs the translation you need.

Locate the h323-conf-id for a call

Unique call IDs are extremely important for troubleshooting. They allow you locate a call in the database and find call information in the billing engine logs or RADIUS detail files. Finally (and this is the most important thing), when reporting call problems to the PortaOne Support team or your business partner, a call ID helps to prevent confusion and allows the problem to be solved quickly.

In order to find the h323-conf-id for a call, do the following:
1. From the main menu, go to **Trace Call**.
2. Using the CLD and from/to date filters, make sure the required call is shown in the list of calls on the screen.
3. In the row containing call information, click on the 🔍 **View** icon (far left column). You will be shown detailed information about the call.
4. The first field on the screen should be the h323-conf-id, which is a combination of four hexadecimal numbers separated by spaces, e.g. `5FF7F6D1 715E02C6 A40990F3 C823E27E`.

Troubleshoot an incorrectly billed call

1. Make sure that someone in your organization is subscribed to the PortaControl mail alerts (especially "Missing critical information"). This will help you to detect problems early.

2. Find the h323-conf-id for this call. This is a unique ID (a string of four hex numbers) generated by the gateway when the call was started, which will help you to exactly identify this particular call among all the others. You can find this ID by doing one of the following:
   • Looking at the statistics on your gateway.
   • Going to the radius details directory (`/var/log/radacc`), entering the subdirectory for the corresponding gateway, and using a program like `more` or `less` to browse the file. You can search by the number you dialed or account (PIN) number, or just browse the records near the end of the file.
   • Browsing the PortaControl log (`/var/log/porta-billing.log`) with `less` utility for the number you dialed, account (PIN) number, or just browsing the records near the end of the file.

3. Look in the PortaControl log (`/var/log/porta-billing.log`) or use **View logs** on the web interface to find all of the information about call processing there.

Note: Normally we receive information about each of the call legs separately, so it is necessary that you check the log entry regarding the processing of **all** call legs and the final call clean-up.

4. The following are typical error situations:

- **Call was not billed at all**. It was considered an "unresolved" call, because we did not detect that it went to any of the vendors. Check that you have correctly defined connections to your vendors.

- **Call was billed only to vendor, not to an account or customer** (and you received an email alert). The billing engine needs an Account ID in the User-Name attribute to correctly identify the account and customer. Check the logs to see what was in the User-Name attribute. Typical situations are:

    1. **Incorrect value in User-Name** (for example, phone number instead of IP address of the remote gateway). Cause: the required application was not used to handle the call. Remedy: check that the application is configured and associated with the corresponding *incoming* dial-peer.

    2. **On some of the call legs there is a correct value in the User-Name, while on others there is an IP address.** When a call is originated on gateway A and terminated on gateway B, then a "real" username appears only in the accounting from gateway A. In gateway's B accounting the username will be the IP address of gateway A, because gateway A had to authenticate itself before being able to make the call. This is a perfectly normal situation. PortaControl recognizes this, and will replace a username identical to the node's ID (IP address) with a "real" username from the other call legs.

    3. **Value in User-Name is correct, but reports "Did not find account/customer".** Check the accessibility for the account's product. It may be that, even if you have such an account, its accessibility is incorrect.

- **Call was billed, but the phone number in the billing is incorrect (not in E.164 format) and there is a "Mismatch in rates or destinations" error.** Cause: missing or incorrect translation rule on connection to the vendor. Remedy: assign a translation rule; read the "Deal with technical prefixes and numbering formats"section.

- **Call was billed and the phone number in the billing is correct, but there is a "Mismatch in rates or destinations" error.** Cause: missing rate for this phone prefix in the tariff. Remedy: create a rate for this destination in the tariff.

5. If you are unable to solve the problem by yourself, submit a problem report to **support@portaone.com**. Please make sure that you include the following in your email:
   - A detailed description of the call flow and what seems to be incorrect.
   - H323-conf-id of the call.
   - Relevant items from the `porta-billing.log.`

Create a custom TCL application

Sometimes you need a special call-handling or IVR functionality which is not available in the applications you already have. For example, you need a prepaid card IVR which will perform ANI authentication prior to PIN authentication.

First of all, check the TCL applications available as part of the Cisco TCLWare package, since what you are looking for (or something similar) is perhaps already there. If you find an application similar to what you need, you can use it as a template.

 A productivity tool for TCL IVR development is available from Porta Software Ltd. **FastIVR** emulates Cisco with regard to script execution, TCL API, and VoIP-related IOS commands. You can find out more at: **http://www.portaone.com/products/fastivr/**

Make the 'Periodic payments' tab appear in the customer/account info

The Periodical payments tab will only be shown if the rest of the system is configured to accept payments. This requires:
- At least one payment system defined in the **Company Info** section.
- A payment system assigning the currency of this account or customer.
- For accounts, the **E-commerce enabled** flag must also be switched on.

- On the **Payment info** tab, the Preferred payment method must be set and information about the credit card entered.

For sub-customers or accounts belonging to sub-customers, the requirements are the same, except that the payment systems are defined in the reseller information, not in Company Info.

Prevent an ANI number from being used as a PIN

The first method is the easiest one: use different gateways or access numbers for PIN-based and ANI services, and configure product accessibility accordingly. Thus even if a "hacker" calls your access number 12345 for prepaid cards and attempts to enter his neighbor's phone number as a PIN, the call will not be authorized, since, although such an account exists, its product only allows usage with the access number 12346.

However, you might need to create an advanced service in which a single access number can be used for both ANI and PIN-based services. When the customer calls in, the system checks his ANI and, if the ANI is OK, it asks for a destination number. Otherwise, it gives the option of entering either a PIN or a phone number + password. In this case, you must prevent account misuse in a different way:

- ANI accounts should **always** be created with a non-empty VoIP password. Prepaid cards should be created with an empty VoIP password.
- Modify the TCL script, so that when the first authentication by ANI is done, the billing will receive User-Name=ANI and a special flag "skip password". Thus, authentication will be successful if such an account exists, otherwise it will fail and the user will be prompted for a PIN. To activate the "skip password" feature in PortaOne's "Advanced Remote Authenticate" script, simply include the `skip-password yes` configuration parameter for this application.
- When a user enters a PIN, the PIN is provided in the User-Name, and the Password attribute is empty. The system checks for such an account, and since the password is empty for prepaid card VoIP, authentication is successful. If somebody tries to enter an ANI number as the PIN, authentication will fail because the password supplied does not match the one assigned to the account.
- If given the option "enter your registered phone number", the user will then enter both his phone number and password (the latter is required to prevent unauthorized usage of his account),

and both will be supplied to the billing. Authentication will be successful only if a correct account ID and password are provided.

Make a custom report from PortaControl

PortaControl provides you with an open data model. An ER-diagram of the database structure is not included in this document, but may be obtained from Porta Software Ltd upon request. If you want to prepare custom reports on your workstation or a non-PortaControl server, you will need to do the following:

- Make sure the remote computer has database drivers installed to access the PortaControl database. Normally you would use native MySQL connectivity on Unix-based hosts and ODBC on Windows-based hosts.
- For any data-mining solutions (extracting data from the database), use **only** the slave database.
- Use a tool like Crystal Reports, Microsoft Access or some custom application to retrieve data from the database, process it and submit it to the user.

Use ODBC to connect to PortaControl

ODBC (Open Database Connectivity) provides a way for client programs to access a wide range of databases or data sources. If you need extended customized reporting not available in PortaControl, you can do this using external tools such as MS Access or Crystal Reports.

Create a MySQL user to be used for reports

1. Login into your PortaControl slave server using ssh.

2. Start the MySQL command tool.
```
andrew@demo:/home/porta-admin$mysql -u root mysql
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 42122 to server version: 4.0.17-log
Type 'help;' or '\h' for help. Type '\c' to clear the
buffer.
mysql>
```

3. Create a new user using the GRANT command.
```
mysql> grant ALL PRIVILEGES  on `porta-billing`.* to
'reports'@'192.168.0.5' identified by 'pod23uk';
Query OK, 0 rows affected (0.02 sec)

mysql>
```

4.   Flush the privileges.
```
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.07 sec)
```

Installing the MySQL ODBC driver

1.   Download and run the installation package from:
     http://www.mysql.com

2.   Click **Next** on the information screens.
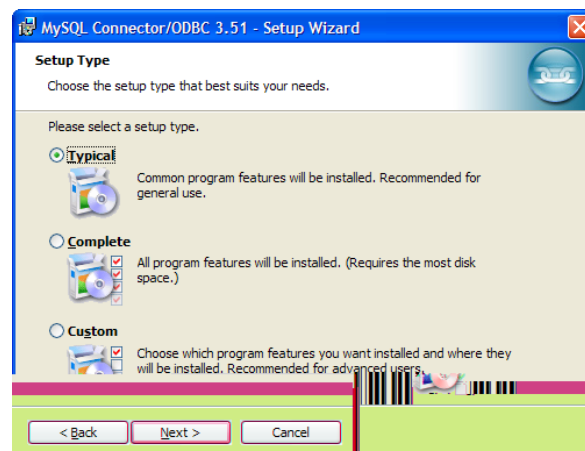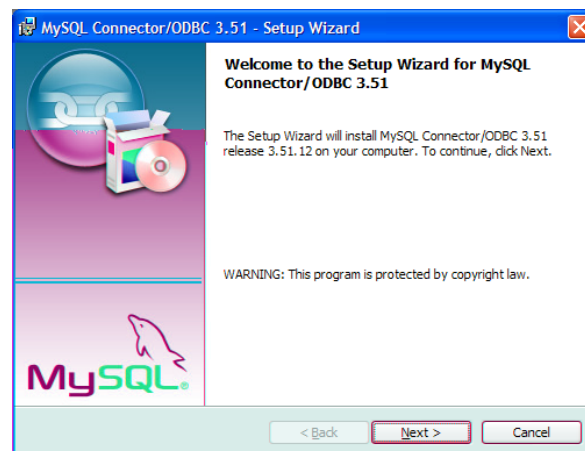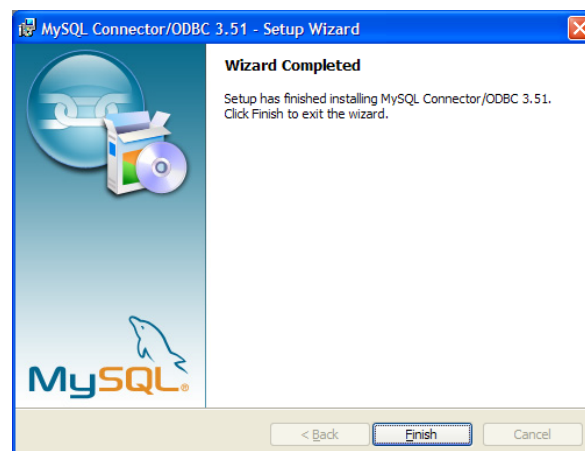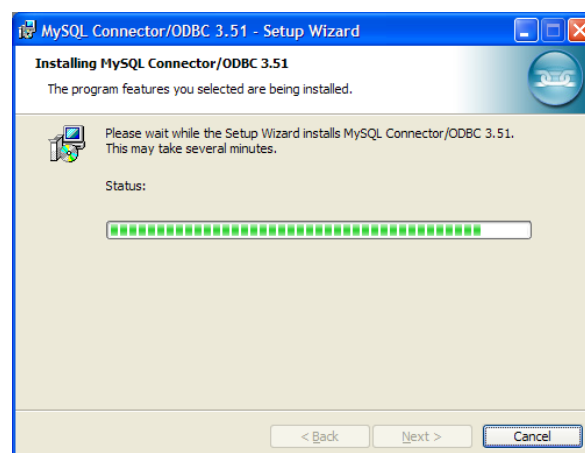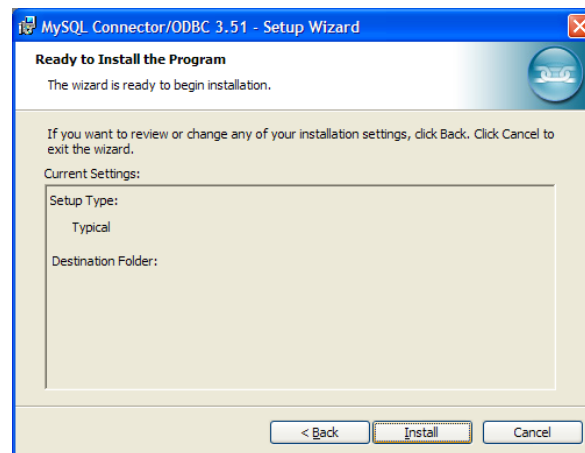
3. Click **Finish**.

Configuring ODBC

Before configuring the data source, create an MySQL user on slave DB with read-only permissions. Please examine the following document on how to add new user accounts to MySQL:
http://www.mysql.com/doc/en/Adding_users.html

1. Control panel -> Administrative tools -> Data sources (ODBC).

2. Select myodbc3-test and click **Configure…**". Fill in the configuration form.
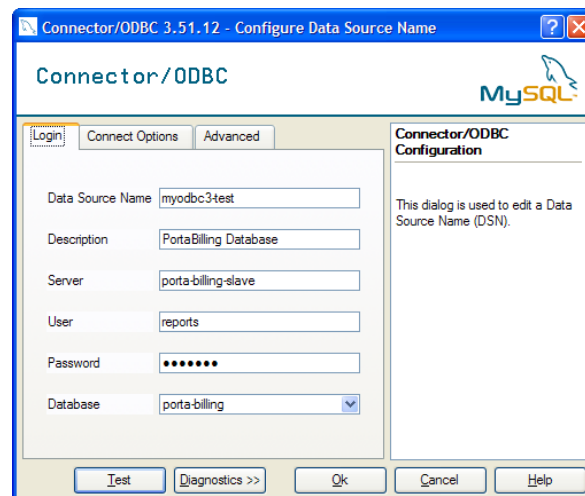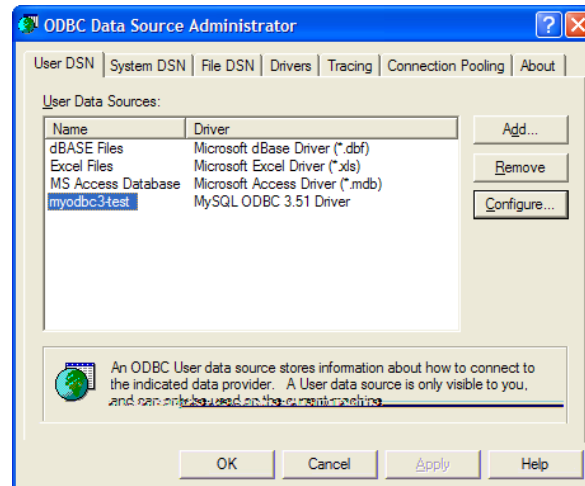
Important parameters include:
- **Host/Server Name (or IP)** – hostname (or IP address) of your **slave** server
- **Database Name** – `porta-billing`
- **User, Password** – username and password of the MySQL user you have created for reporting purposes
- **Port** – the port on which the database service is accessible; enter **3307** here

Note: This port number differs from the one used by default.

Using ODBC
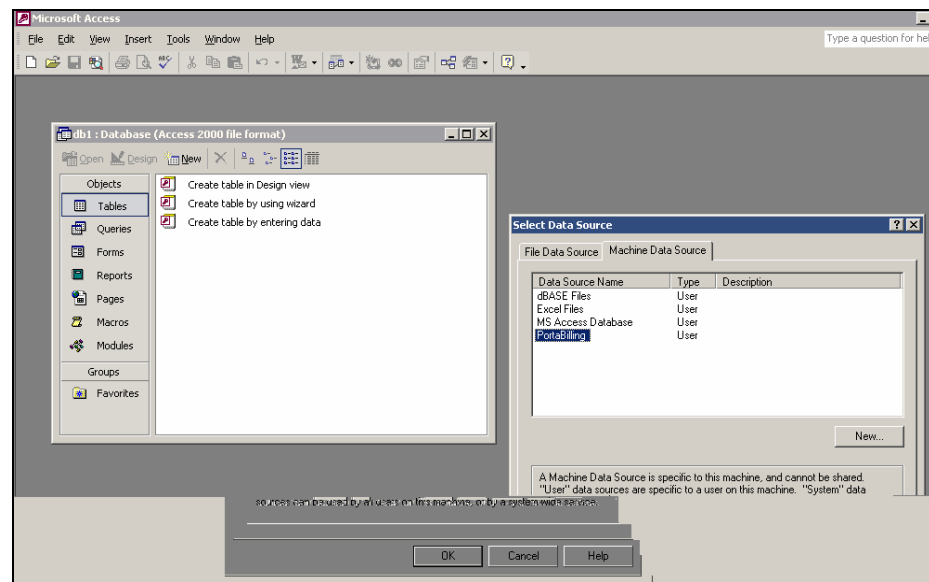
In MS Access:

1. Create a blank database.
2. Right-click in the table design view, and choose **Link tables…**
3. Choose **ODBC databases** from "Files of type…" list.
4. Select **Machine data source**.
5. Select **PortaBilling** and click OK.
6. Select the desired tables.
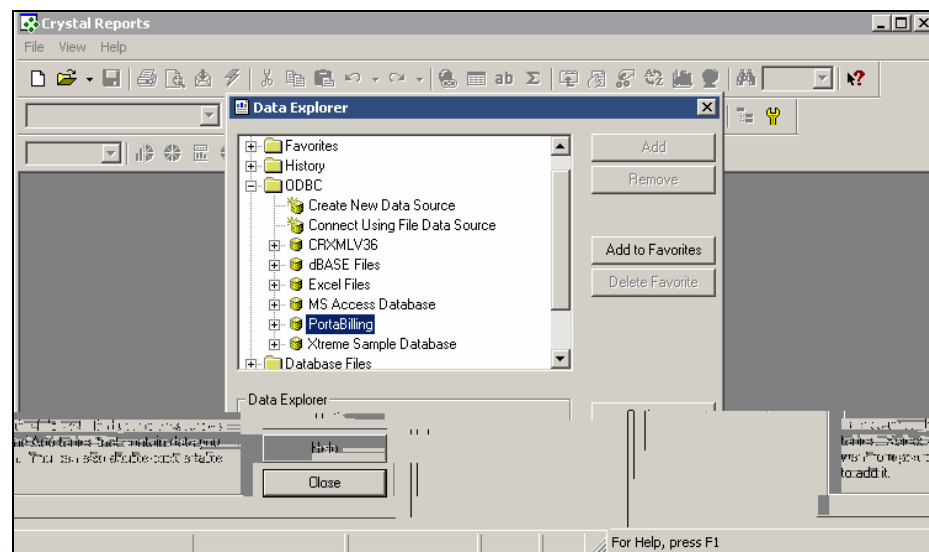
In Crystal Reports:

1. Create a *New Report*.
2. In *Data Explorer*, open **ODBC** branch.
3. Select **PortaBilling**.
4. Select the desired tables.

Use the redirect number feature

Each account in PortaControl has an **associated number** field. If this field is not empty, PortaControl will include an h323-redirect-number in each authentication confirm message and use an h323-return-code of 52. How to proceed in this case is determined by the gateway. The account owner is able to modify the associated number using customer self-provisioning.

To implement a simple call forwarding service (i.e. authenticate by DNIS and then forward the call to a number defined in the properties of the DNIS account), you may use PortaOne's "Advanced Remote Authenticate" script.

Customer support number

PortaControl allows you to enter the redirect number during the process of account generation so that you may specify different redirect numbers for different account batches. Different batches may be distributed in different regions, and only a slight modification to the prepaid card script will make it more intelligent in terms of choosing a customer support center.

IP telephony private line auto ringdown

In the case of a combination of ANI (CLI) authentication and immediate call forwarding, you may map phone lines from one world region to another. This is advantageous in that you will require only one local access number. Mapping is dependant on ANI (CLI).

Configure an outgoing connection to a vendor when sending calls using a gatekeeper so that the remote IP address is not known in advance

In this situation, you clearly cannot use a **VoIP to Vendor** connection, since the IP address of the remote gateway will be assigned dynamically. The solution is to set up an IPIPGW and use it to send traffic to this vendor.

If you only have one such vendor whose termination IP address is unknown beforehand, and are sure that everything sent to the unknown IP address is going to this vendor, you can use a special feature in PortaControl to handle this situation: create a "VoIP to Vendor"

connection and enter **ANY** as the remote IP address. Any outgoing VoIP call which goes neither to one of your trusted nodes nor any IP addresses explicitly defined in other connections will match this connection.

If you have more than one such vendor whose remote IP is not known in advance, this method is no longer applicable. If you are using tech-prefixes, so that the phone number changes based on which carrier is being used (e.g. a call to 42021234567 will go to vendor A as 12345#42021234567, and to vendor B as 9876542021234567), you can use connection match by prefix. In the example above, enter **PREFIX:12345#** in the Remote IP field for the connection to vendor A, and **PREFIX:98765** for the connection to vendor B. Such a prefix connection will be matched only if the call does not go via a connection with a specific IP address or to a trusted node.

Force PortaControl to disconnect after a customer calls over his credit limit

There is no need for PortaControl to do this, as the gateway is able to by itself. When the gateway authorizes an account to make a call in PortaControl, PortaControl returns a maximum credit time in the case of a successful authorization. When the gateway connects the call, it starts a timer; when the timer hits zero, it automatically disconnects the call. PortaControl sends **two** RADIUS attributes related to the maximum call duration. The `h323-credit-time` attribute is the standard one (announced credit time); there is also a custom `h323-ivr-in = DURATION` attribute (actual credit time). The value for the second attribute is computed with billing tricks taken into account. Thus, in the case of $10 available funds, a $0.10/min rate and a 10% post-call surcharge, this attribute will be 5400 seconds (90 minutes), while the h323-credit-time will be 6000 seconds (100 minutes).

If the application supports only the `h323-credit-time` attribute (e.g. Cisco default debit card application), then this time will be announced to the end user and used to disconnect the call, and `h323-ivr-in = DURATION` will be ignored.

Create accounts to be used for SIP services

There are no special requirements as to how such accounts should be created. You use the same interface to create and manage accounts for all services supported by PortaControl (H323, SIP). Thereafter accounts can use H323, SIP or SIP & H323 services, depending on their product's accessibility. So if you plan for accounts with a certain product to be able

to login to the SIP server and make outgoing calls, be sure to include the PortaSIP node with the appropriate tariff in the **Accessibility** for this product.

Integrate PB logins to your website

You can include the login form on your website using the following HTML code:

```
<form name=log method=post action=https://pb.mycompany.com/login.html>
      Login <input type="text" name="user" value="">
      Password <input type="password" name="password" value="">
      <input type=button value="Login" onClick="submit()">
      <input type=hidden name="redirectOnLoginError"
value="http://www.myweb.com">
</form>
```

`"redirectOnLoginError"` – in case of unsuccessful login, PortaControl will redirect the browser to the URL specified in the value attribute.

Add my company name and logo to every PortaControl web page

1. The custom HTML component file should be placed in the directory accessible by the web server; the best place is in the `~porta-admin/apache` directory. Login to the slave server and become a super-user, then type:
   ```
   cd ~porta-admin/apache
   mkdir brandpane
   ```
2. Create the HTML component which will be included on every PortaControl page. Edit the file `my.html` in the `~porta-admin/apache/brandpane` directory (a sample file is shown below).

   ```
   <table width=100% border="0" align="left"
   cellpadding="0" cellspacing="0" bgcolor="white">
       <tr>
           <td width="100%">
               <img src="/mylogo.gif">
           </td>
       </tr>
   </table>
   ```

3. Activate the brandpane feature in the PortaControl web interface config (edit the `~porta-admin/etc/porta-admin.conf` file on the slave server).
   ```
   [Web]
   …
   BrandPane=/brandpane/my.html
   …
   ```

If you are not familiar with editing text files on the Unix platform, ask PortaOne support to do this for you.

4. Restart the Apache web server (`apachectl restart`). Now you should see your company logo at the top of every PortaControl web page.

Add a reseller's logo to his customers' web pages

1. Activate the brandpane feature as described in the topic above.
2. Configure an additional virtual web server which will run on the reseller's domain name (e.g. **www.supercall.com**).
3. Include conditional statements in the brandpane component. These will analyze the domain name the customer is trying to access, and will then produce the company name and logo for your domain (**www.sipcall.com**), while for the reseller's (**www.supercall.com**) the reseller's name and logo will be produced. See the following example:

```
% if ($ENV{HTTP_HOST} =~ m/supercall.com/) {
<img src="/supercall.gif">
% } else { # our default domain
<img src="/sipcall.gif">
% }
```

Configure online web signup

Typical steps for online web signup configuration:

- Decide where you are going to host your online web signup page. (This could be your corporate website, for instance.) Determine the URL of the page which will submit the request to create an account in PortaControl. You will need to enter this in the "Subscription HTTP_REFERER" field. For instance, if your web signup page is running on: **http://www.mydomain.com/signup.html**, then this is exactly what you should enter in the field. If your web signup portal consists of several pages or screens, you should enter the address of the last page, i.e. the one which contains the Submit button invoking the PortaControl signup procedure. You will need an https server and a digital certificate in order to secure credit card transactions.
- Create a Web Signup HTML form using the example provided with PortaControl and put it on your web site. You can add or remove some fields in order to achieve the functionality you

require, i.e. you may request that an account ID be entered if you are providing ANI-based service, or may not, if you are in the debit card business, and so on.

- In the configuration file of the web server: `/usr/local/etc/apache/porta.httpd.conf`, uncomment the section about the virtual host for online signup; by default it runs on port 8500.
- Create a validation module or use the one provided with PortaControl. By customizing the validation module, you can implement the advanced restrictions you require (e.g. allow creation of accounts with a balance of 10, 20 or 50 dollars only, or ensure that lifetime is no more than 180 days). Place this module somewhere on the web server so that it will be accessible for mod_perl; for example, in: `/usr/local/lib/perl5/site_perl`. Thus, if you decided to name your module: My::PortaValidation::ResellerABC, the actual program code would reside in: `/usr/local/lib/perl5/site_perl/My/PortaValidation/ResellerABC.pm`

NOTE: Make sure that you restart the web server (`apachectl restart`) after making changes in the module.

Setting up online web signup for your company

- Ensure you have defined the online the payment processor and have it accociated with the desired currency.
- Fill in the form on the Product "Online Signup" Tab.

Setting up online web signup for your reseller

- Create a reseller.
- Create a set of tariffs and products managed by this reseller.
- Fill in the form on the Product "Online Signup" Tab.
- Login to the reseller self-provisioning interface and enter the merchant account data (**Company Info** -> **Payment Systems** tab).
- On the reseller self-care interface assign this online payment processor to the desired currency.

How it works:

- To start using your services, a user must go to your website and proceed to the online web signup page.
- Once the signup form is completed, the data will be posted to the PortaControl server.
- After receiving the signup request, PortaControl checks the HTTP header (provided by the customer's browser) and searches for a product with a matching "HTTP_REFERER", allowing further processing only if the host is found in the database. If not found, the transaction is rejected.
- PortaControl associates the Subscription Host with the product for the account that is being created.
- PortaControl will use the appropriate validation module to check if the supplied data are correct.
- The validation module will check the data, fill in any non-mandatory fields and return the data, along with the acquired status, to PortaControl.
- If the status indicates an error, then registration is rejected and the browser is redirected using "Continue URL".
- If the status is OK, then PortaControl proceeds with registration, creates the account/customer and attempts to execute the credit card transaction.
- If the status indicates an error, then registration is rejected, account/customer records are deleted and the browser is redirected using "Continue URL".
- If the status is OK, then PortaControl redirects the browser using "Continue URL".
- Each time the browser is redirected using "Continue URL", PortaControl provides a comprehensive set of parameters.

List of "Continue URL" parameters and possible values:

- `status` "OK", "ERROR"
- `message` – e.g. "CC rejected"
- `session_id` – unique session ID supplied by the web subscription server
- `account_id` – generated account ID

Sample files:

These can be found on your PortaControl installation in the following directories:

```
~porta-admin/apache/subscription/subscription.html
~porta-admin/apache/subscription/subscription_result.html
~porta-admin/site_lib/Porta/Subscription.pm
```

An online web signup HTML form can be found at the following URL:

```
https://mywebserver.myCompany.com:8500/subscription.html
```

where `mywebserver.myCompany.com` is the URL of your PortaControl Web Interface, and 8500 is the port on which the subscription virtual host is running.

5. Maintenance

Configuration files

There are two separate configuration files, one for the billing engine (master server) and one for the web interface (slave server).

Billing engine

The configuration file is called `porta-billing.conf` and is located in the `/home/porta-billing/etc` directory on the master server. It is automatically created by the installation program and populated with the initial configuration. You do not have to edit it unless you wish to alter the default behavior.

porta-billing.conf:
```
[Global]
Suspicious_Time_Threshold_sec=3599
```
Maximum acceptable difference in time sent by the gateway and current time on the server. Default is 1 hour.
```
SQL_Trace=0
```
Whether all SQL queries and their results should be logged to the PortaBilling log file. Use with caution, and only when a detailed debug is necessary, since this produces huge log files.

```
# Only full qualified email addresses, no local aliases please
#
[Email]
Last_Resort=admin@yourdomain.com
```
If a critical condition is detected (e.g. radius server is down) who should receive the email alert.
```
Header_From=porta-billing@yourdomain.com
```
What should appear in the **From:** field of email alerts.
```
Header_To=PortaBilling-MailList-Recipients:;
Max_Sequential_Emails=5
Look_Behind_sec=450
```
If the **Max_Sequential_Emails** in the **Look_Behind_sec** interval have already been sent, any extra email alert will be blocked. This is done to protect against email flooding.
```
Mail_Templates_Dir=Mail_Templates

[Master]
DSN=DBI:mysql:database=porta-billing
User=root
Password=xxxxx
```
Connect parameters for the master database.

```
# Simultaneous login prevention for debit accounts
#
[Fraud_Detection]
Active=Yes
```
Turn the fraud protection on.
```
Test_Mode=No
```
Turn the fraud detection on. Debit accounts will be allowed to login more than once, but an email alert will be sent.
```
Recovery_Time_sec=1800
```

After a debit account is successfully authenticated, how long should this information should be kept in the memory? Default is 30 minutes. A longer recovery time will increase the amount of memory required by the billing engine.

```
# Node IP Cache
#
[Node IP Cache]
Cache_Cleanup_Time_sec=600
```

How often information about nodes and connections should be updated from the database. You can force nodes/connections to reload by sending a HUP signal to the radius daemon.

```
# radcheck.pl
#
[Self Test]
Timeout_sec=5
Test_Frequency_sec=60
Errors_Before_Restart=3
Path_To_Lock_File=/var/run/radcheck.pid
NAS-IP-Address=127.0.0.1
Shared_Secret=SecretKey
User-Name=127.0.0.1
Password=cisco
```

This section defines parameters for the PortaBilling self-test suite. The script periodically sends an authentication request to the radius server, and expects a positive answer.

```
# radcheck.pl & porta-clients.pl
#
[Radius]
Host=127.0.0.1
Path_To_radius.sh=/home/porta-billing/radius_scripts/radius.sh
Path_To_Config=/var/db/raddb
Path_To_PID_File=/var/run/radiusd.pid
```

General parameters of the radius server required by the scripts as a self-check.

```
[Log]
# Where logs should go. Possible values are:
# Syslog, File and Console
Log_To=File
```

This writes a PortaBilling log to the file on disk. This is the default option, and normally is sufficient for both production and debugging. Use other ways of logging only for special debug purposes.

```
# For logging to file, give the filename.
# For logging to the syslog, give description in
# form of syslog:<facility>:<priority>
Log_Dest=/var/log/porta-billing.log
```

Path to log file. Default is: /var/log/porta-billing.log

```
#Log_Dest=syslog:daemon:warn
#NAS=13.232.103.106
#Username=0007777777
```

```
[Call_Cleanup]
```

Parameters in this section configure the lifetime of calls in the call cache.

```
Radius_Retransmits=5
Radius_Timeout=3
```

In order to know when we should no longer expect any packets, we need to know the maximum possible delay with which the request can be delivered to radius (the last possible retransmit).

```
Alive_Interval=60
```

If we bill by keep-alive requests, we need to know what is the interval between them.

```
Incoming_Session_Lifetime=10
```

Call lifetime after incoming call leg has been disconnected.

```
Default_Lifetime=120
```
Default lifetime of the call.

```
[Suggestions]
No_Remote_IP_Auth=no
```
Whether PortaBilling should warn you when it seems that your gateways are not authenticating incoming VoIP sessions.

```
[Features]
LogRequestEnvironment=no
```
Write in the log file which environment this RADIUS request is processed in. This is useful in the case of an ASP with many environments and cross-environment calls.
```
PatternFullStringMatch=no
```
When matching a port pattern in the connection, require that the NAS-Port-Name string should match the pattern completely, rather than finding a pattern somewhere in the middle of the NAS-Port-Name.
```
ChargeCustomerForIncomingCalls=no
```
Charge the reseller for ingress call legs (e.g. for toll-free numbers).
```
CheckVoucherOwner=yes
```
Vouchers can only be used for accounts under the same reseller
```
CheckDebitAccountOverDraft=yes
```
Send an alert when an overdraft is detected on a debit account.
```
ChargeAccountsWithCustomerTimezone=yes
```
Apply the customer's time zone when calculating volume discount counter periods.
```
UseAlertTime=yes
```
Use the alert-timepoint attribute to calculate the value stored in the setup_time columns in the database. Using alert-timepoint (as opposed to h323-setup-time) allows you to obtain reliable statistics about PDD.

```
ActiveCallsPresentation=yes
# ActiveCallsPresentationCleanup_sec=30
# ActiveCallsPresentationStalledCleanup_sec=86400

ProfitMonitor=no
ProfitMonitorLimit=0
CheckAvailableFundsDebit=no
CheckAvailableFundsCredit=yes
FullMatchPrefixForVolumeDiscount=no
```
When looking up applicable volume discounts for a call, check for any prefixes which match the dialed number (this increases load on the billing engine and database). The default method requires the prefix in the volume discount definition to be identical to the prefix used to rate this call in the tariff.
```
ReplaceCliWithUserName=no
```
For outgoing calls, make CLI identical to the authorization username (obsolete with the introduction of the per-account "Force CLI" setting in call features).

Admin server and web interface

The configuration file is called `porta-admin.conf` and is located in the `/home/porta-admin/etc` directory on the slave server. It is automatically created by the installation program and populated with the initial configuration. You might wish to edit it so that the system will better suit your needs.

porta-admin.conf:

```
[Global]
System_Name=MyBilling
```
Enter a distinctive name for your system here.
```
sendmail=/usr/sbin/sendmail
ZIP=/usr/local/bin/zip
UNZIP=/usr/local/bin/unzip
TempDir=/var/tmp/porta
maintainer=PortaBilling <porta-100@mybilling.com>
```
What should be in the **From:** field of outgoing emails.
```
Time_Zone=America/New_York
```
Default time zone (when creating new objects such as user, customer, etc.).
```
Debug=0

Check_Age=2
Pager=30
```
How many entries per page should be shown in tables. This is a master parameter, and can be altered for individual forms.

```
[Porta_Realm]
admin=443
accounts=8445
cc_staff=8446
customer=8444
vendor=8447
representative=8448
```
Ports that different parts of the web interface are using.

```
#admin_url=https://admin.mybilling.com
#accounts_url=https://account.mybilling.com
#cc_staff_url=https://helpdesk.mybilling.com
#customer_url=https://billing.mybilling.com
#vendor_url=https://carrier.mybilling.com
#representative_url=https://agent.mybilling.com
```
Fully qualified web URLs to different realms of the PortaBilling admin interface; for instance, if you would like to run each one on its own SSL virtual host (using default SSL ports)

```
[Master]
DSN=DBI:mysql:database=porta-billing;host=porta-billing-master
User=root
Password=xxxx
PrintError=1
```
Connect parameters for the master database.

```
[Slave]
DSN=DBI:mysql:database=porta-billing;mysql_socket=/tmp/mysql-
slave.sock
User=root
Password=xxxxx
PrintError=1
```
Connect parameters for the slave database.

```
[Backup]
Keep=1

[X-Rates]
# Max age (days)
Max_age = 60
Pager = 25
Base_Currency = USD

[Web]
```

```
# Login expiration (seconds)
Login_expire=172800
# Login expire if not used (seconds)
Passive_expire=86400

[Currency]
Pager = 25

# Default values for Card generator
```
Default values for "Account generator".
```
# Billing_model={Debit| Credit | Voucher}
# Preferred_language=<2-chars language name from iso-639-1>
[Cardissue]
Amount=1000
Balance=10
Life_time=90
Length=12
```
Default PIN length.
```
Billing_model=Debit
Preferred_language=en

# Default values for 'Add card manually'
# Billing_model={Debit| Credit | Voucher}
# Preferred_language=<2-chars language name from iso-639-1>
[Addcard]
Balance=0
Life_time=90
Batch=
Billing_model=Credit
Preferred_language=en

# Default values for Rates
# Pager Lines per page
# Cleanup_Interval -- interval for cleanup in days
[Rates]
Pager=50
Cleanup_Interval=90
```
How long old (not current) rates should be kept.
```
[CustomerWeb]
Pager=50
Interval=1

[AccountWeb]
Interval=1
Pager=50

[TraceCall]
Pager=50
Interval=1

[Invoice]
Pager=50
Interval=1

[Stats]
GrossMargin_Interval=1
# GrossMargin_Measure = day | month
GrossMargin_Measure=day
# GrossMargin_MaxAge = max age in days
GrossMargin_MaxAge=1000
ASR_days = 90
ASR_weeks = 52
```

```
ASR_month = 10

[RRD]
```
It is not recommended to change these parameters without a good grasp of RRD
principles.
```
Load_delay_seconds = 3600
Zero_call_length = 10
Load_delay_interval = 900
# % when ASR graph appears
ASR_Threshold = 5

[StatCalc]
Interval=0
Mask=%

[OnlinePayment]
PassFile=/home/porta-admin/etc/passphrase.txt
```
Encryption key for merchant account passwords.
```
Pager=25

[CDR]
```
How long CDRs should be kept in the database.
```
Keep_month=2
Keep_month_failed=2
```

Glossary / list of abbreviations

A

AAA – Authentication, Authorization and Accounting. A suite of network
security services that provides the primary framework through which you
can set up access control on your router or access server.
ACD – Automated Call Distribution
AD – Abbreviated Dialing
ADSL – Asymmetric Digital Subscriber Line
AIS – Automated Information System
ANI – Automatic Number Identification
API – Application Programming Interface
ART – Audible Ringing Tone
AS – Answer Supervision
ASP – Application Service Provider
ASR – Automated Speech Recognition
ATA – Analog Telephone Adapter
AVVID – Architecture for Voice, Video and Integrated Data

B

B2B – Business to Business
B2C – Business to Consumer
BH – Busy Hour
BHCA – Busy Hour Call Attempts
BHCC – Busy Hour Call Completion

BIOS – Basic Input/Output System
bps – Bits per Second
Bps – Bytes per Second

C

call leg – A link or hop along the route from the calling party to the destination (called) party.
CALEA – Communications Assistance for Law Enforcement Act
CDR – Call Detail Record
CGI – Common Gateway Interface
CHAP – Challenge Handshake Authentication Protocol
CLE – Customer Located Equipment
CLEC – Competitive Local Exchange Carrier
CLI – Command Line Interface
CLI, **CLID** – Calling Line Identification
CMS – Call Management Services
CPE – Customer Premise Equipment
CRM – Customer Relationship Management

D

DCE – Data Communications Equipment
DHCP – Dynamic Host Configuration Protocol
DID – Direct Inward Dialing
DMZ – Demilitarized Zone
DNS – Domain Name System
DSL – Digital Subscriber Line
DTE – Data Terminal Equipment
DTMF – Dual-tone multi-frequency. The use of two simultaneous voice-band tones for dialing (such as touch-tone).

E

E–1 – Europa-1, the European equivalent to T1 (but higher bandwidth).
E911 – Enhanced 911
ENUM – Electronic Numbering

F

FAQ – Frequently Asked Questions
FCC – Federal Communications Commission
FoIP – Fax over IP
FQDN – Fully qualified domain name. The full name of a system, rather than just its hostname. For example, test is a hostname, while test.portaone.com is an FQDN.
FXO – Foreign Exchange Office
FXS – Foreign Exchange Station

G

GK – Gatekeeper. The entity that maintains a registry of devices in an h323 network.
GMT – Greenwich Mean Time
GSM – Global System for Mobile Communications
GUI – Graphical User Interface
GUID – Globally Unique Identifier (also called h323–conf–id). The identifier for end-to-end calls. A new GUID is assigned to each new call. There may be numerous calls within a single session.

H

HDD – Hard Disk Drive
HTML – Hypertext Markup Language
HTTP – Hypertext Transfer Protocol

I

ICMP – Internet Control Message Protocol
IM – Instant Messaging
IMAP – Internet Message Access Protocol
IMS – IP Multimedia Subsystem
IOS – Internetworking Operating System
IP PBX – Internet Protocol Private Branch Exchange
ISDN – Integrated Services Digital Network
ISP – Internet Service Provider
ISV – Independent Software Vendor
ISDP – Internet Telephony Service Provider
IVM – Instant Voice Messaging
IVR – Interactive Voice Response
IXC – Inter–Exchange Carrier

J

JMS – Java Messaging Service
JVM – Java Virtual Machine

L

LAN – Local Area Network
LBS – Location Based Services
LCR – Least Cost Routing
LDAP – Lightweight Directory Access Protocol
LEC – Local Exchange Carrier
Long pound calls – see *Multi-session call*

M

MAC – Media Access Control

MAC address – The physical address of a device on Ethernet.
MG – Media Gateway
MGCP – Media Gateway Control Protocol
MIME – Multi–Purpose Internet Mail Extension
MMS – Multimedia Messaging Service
MOD – Movies on Demand
Multi-session call – A call in which several outgoing calls to different destinations can be made under the same incoming call to the IVR.

N

NANP – North American Numbering Plan
NAT – Network Address Translation
NBH – Network Busy Hour
NGN – Next Generation Network
NIC – Network Interface Card
NNX – The current general configuration for exchange codes within each area code.
NOC – Network Operations Center
NPA – Numbering Plan Area
NTP – Network Time Protocol

O

OEM – Original Equipment Manufacturer
OLTP – Online Transaction Processing
OS – Operating System

P

PABX – Private Automatic Branch Exchange
PAP – Password Authentication Protocol
PBX – Private Branch Exchange
PDF – Portable Document Format
PIN – Personal Identification Number
PnP – Plug and Play
POP – Point of Presence
POTS – Plain Old Telephony Service. Basic telephone service supplying standard single-line telephones, telephone lines, and access to PSTN.
PPP – Point-to-Point Protocol
PRI – Primary Rate Interface
PSAP – Public Service Answering Point
PSTN – Public Switched Telephone Network

Q

QoS – Quality of Service

R

RADIUS – Remote Authentication Dial-In User Service
RAM – Random Access Memory
RAS – Registration, Admission and Status protocol
RDBMS – Relational Data Base Management System
RFP – Request for Proposal
RFQ – Request for Quote
RJ – Registered Jack
RS–232 – Recommended Standard 232 (physical interface between DCE and DTE)
RTP – Real-Time Transfer Protocol
RTCP – Real-Time Control Protocol
RTSP – Real-Time Streaming Protocol

S

SBC – Session Border Controller
SCSI – Small Computer Systems Interface
SDK – Software Developers Kit
SDP – Session Definition/Description Protocol
SG – Signaling Gateway
SIGTRAN – Signaling Transport
SIM – Subscriber Identify Module
SIM Card – Subscriber Identify Module Card
SIP – Session Initiation Protocol
SIP-T – Session Initiation Protocol for Telephony
SLA – Service Level Agreement
SMP – Symmetric Multiprocessing
SMS – Short Messaging Service
SMTP – Simple Mail Transfer Protocol
SOAP – Simple Object Access Protocol
SOHO – Small Office/Home Office
SQL – Structured Query Language
SS7 – Signal System 7

T

T.37 – Standard for store-and-forward FoIP (Fax over IP).
T.38 – Standard for real-time FoIP (Fax over IP).
TAPI – Telephony Application Program Interface
TCL – Tool Command Language. A scripting language used for gateway products both internal and external to the Cisco IOS software code.
TCP – Transmission Control Protocol
TCP/IP – Transmission Control Protocol/Internet Protocol
Telco – A telephone company, such as LEC.
TFTP – Trivial File Transfer Protocol
TTS – Text-to-Speech
TVoIP – TV over IP

U

UA – User Agent
URI – Uniform Resource Identifier
URL – Uniform Resource Locator
USB – Universal Serial Bus

V

VAD – Voice Activity Detection
VAR – Value-Added Reseller
VLAN – Virtual LAN
VOD – Video on Demand
VoiceXML – Voice Extensible Markup Language
VoIP – Voice over Internet Protocol

W

WAN – Wide Area Network
WiFi – Wireless Fidelity
WPA – WiFi Protected Access
WWW – World Wide Web

X,Y,Z

XML – Extensible Markup Language