September, 24th 2018

Alexander Stock
Cloud Infrastructure Architect

Billing with Cloudstack

About Me

- Cloud Infrastructure Architect @itelligence
- Experience in Vmware, KVM, Nagios and Ansible
- Working with CloudStack since 2015
- Mail:
  alexander.stock@itelligence.de

CloudStack Berlin & Dresden, Germany
https://www.meetup.com/german-CloudStack-user-group
Ansible Dresden, Germany
https://www.meetup.com/Ansible-Dresden

itelligence Worldwide in Numbers

Founded

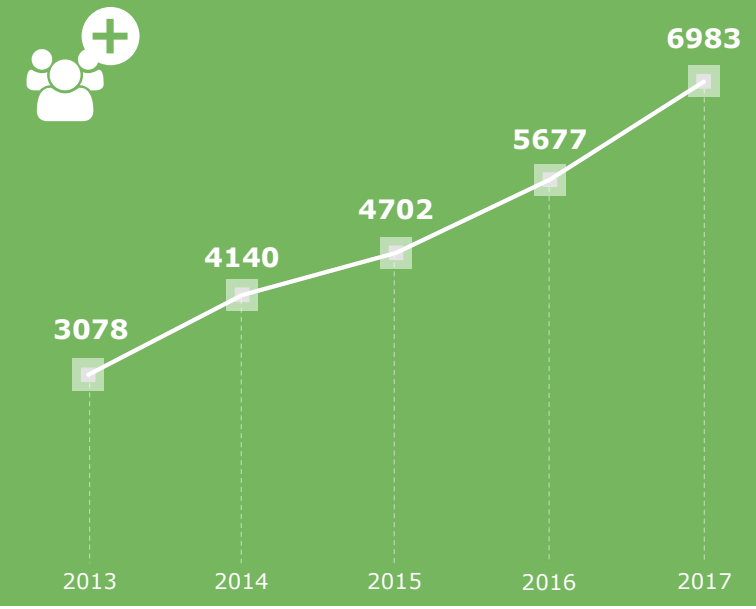
1989

Employees

7,300+

(as of June 2018)

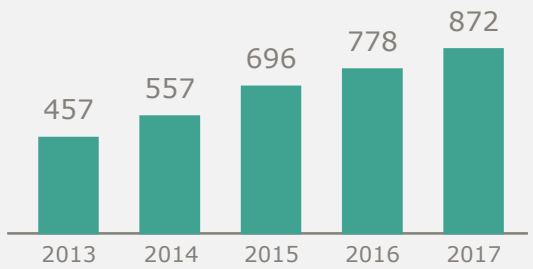
Development of Headcount



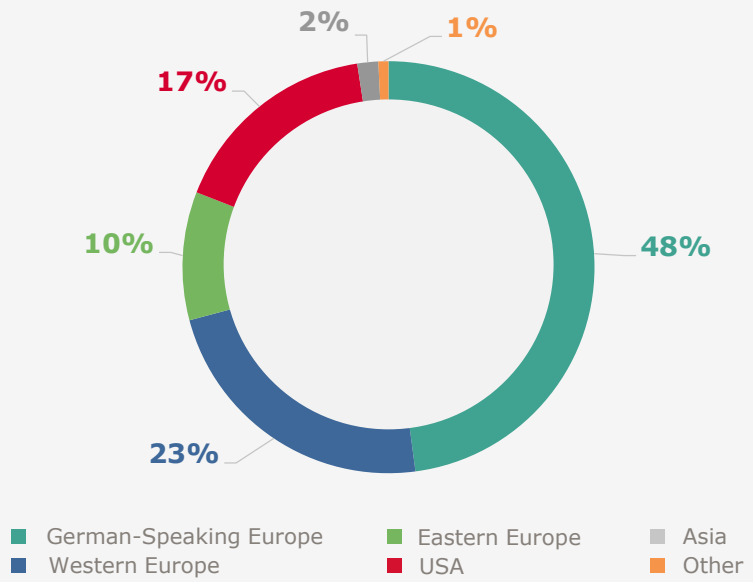
Revenue

872.2 MEUR
(2017)

Ø17,5% Revenue Growth
CAGR over the past 5 years



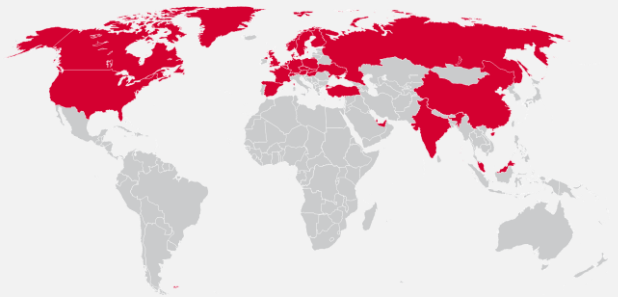
Revenue per Region (2017)



Customers

6,000+

25 Countries



Agenda

1. Short introduction to cloudstack billing functions
2. Our infrastructure and customer design
3. Our first steps with cloudstack billing
4. New approach as a webservice
5. Outlook

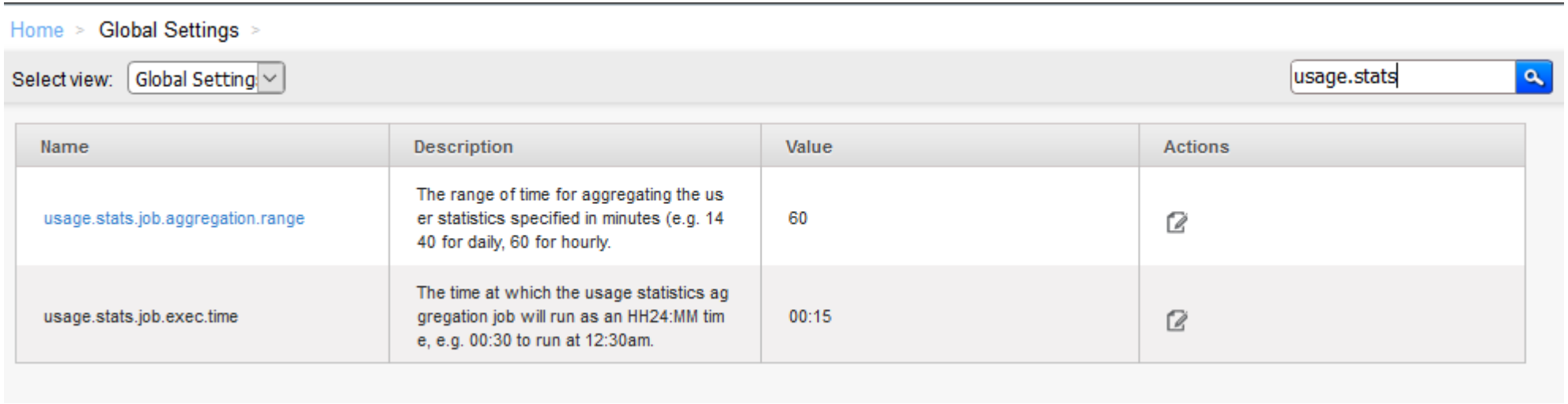
Short introduction to cloudstack billing functions

Short introduction to cloudstack billing

- What information does Cloudstack provide
  - Following metrics can be queries for a specific timerange:
    - CPU and memory usage of a VM.
    - Disk size of a volume and snapshots
    - Network Usage of a virtual router
    - IP address usage
    - Template, ISO usage
    - Loadbalancer/VPN usage

Short introduction to cloudstack billing

- The usage service:
  - Runs periodic jobs to generate usage records
  - Standard period: 24 hours
  - Standard settings: VM/Volume with runtime < 24 hours will not be tracked
  - Can be changed in global Cloudstack settings:



Short introduction to cloudstack billing

- How does Cloudstack generate usage records
  - Step1:
    - Every event like „create“, „destroy“, „start“ or „stop“ will be written to cloud.usage_event table
  - Step2:
    - Usageserver: copy new events to various table in cloud_usage database (helpertables)
    - Aggregate all data in cloud_usage.cloud_usage
    - Records can now be queried over the API

Short introduction to cloudstack billing

- How does cloudstack provide these information over API
  - **listUsageTypes (get mapping for usagetypes)**
    - Output:
      - usagetypeid
      - description
  - **listUsageRecords (get records):**
    - Input:
      - startdate (Date in Format: yyyy-MM-dd HH:mm:ss)
      - enddate (Date in Format: yyyy-MM-dd HH:mm:ss)
      - type (Integer for the specific usage: VM, Volume…)
      - domaindid
      - projectid
      - usageid

Short introduction to cloudstack billing

- How the information is structured:
  - Sample for type 1 (Running VM):

```
name = itcS4P
cpunumber = 24
cpuspeed = 2000
description = itcS4P running time (ServiceOffering: 33) (Template: 239)
domain = itelligence
domainid = 36493690-f727-4e79-8aa8-db20a68bfee1
enddate = 2018-06-24'T'23:59:59+00:00
memory = 256000
offeringid = b17f9251-45c9-4b85-a2ac-9af12f7fd422
project = gbs01
projectid = 62ab2fb4-6c9b-48ea-9f4c-b8d430dd06c1
rawusage = 1
startdate = 2018-06-24'T'23:00:00+00:00
tags:
templateid = 0c45e334-5018-477c-b87b-9229e6c33e35
type = KVM
usage = 1 Hrs
usageid = 75871a3e-22ed-4acc-8a80-ffb0d4ac8ec4
usagetype = 1
virtualmachineid = 75871a3e-22ed-4acc-8a80-ffb0d4ac8ec4
zoneid = b12744d5-a8f3-4328-91f7-3419bfea12b2
```

Our infrastructure and customer design

Our infrastructure and customer design

- Our Setup:
  - We offer: automated Application/SAP setups
  - Cloudstack 4.11.0
  - Advanced Networking
  - KVM as Hypervisor
  - Ceph as Storage Backend
  - Check_MK as Monitoring Solution
  - Ansible is responsible for deploying and configuring our VMs
  - We use projects the seperate resources for the customer
  - We have a self written user portal which manages cloudstack and ansible
  - Customers doesn´t have Cloudstack access

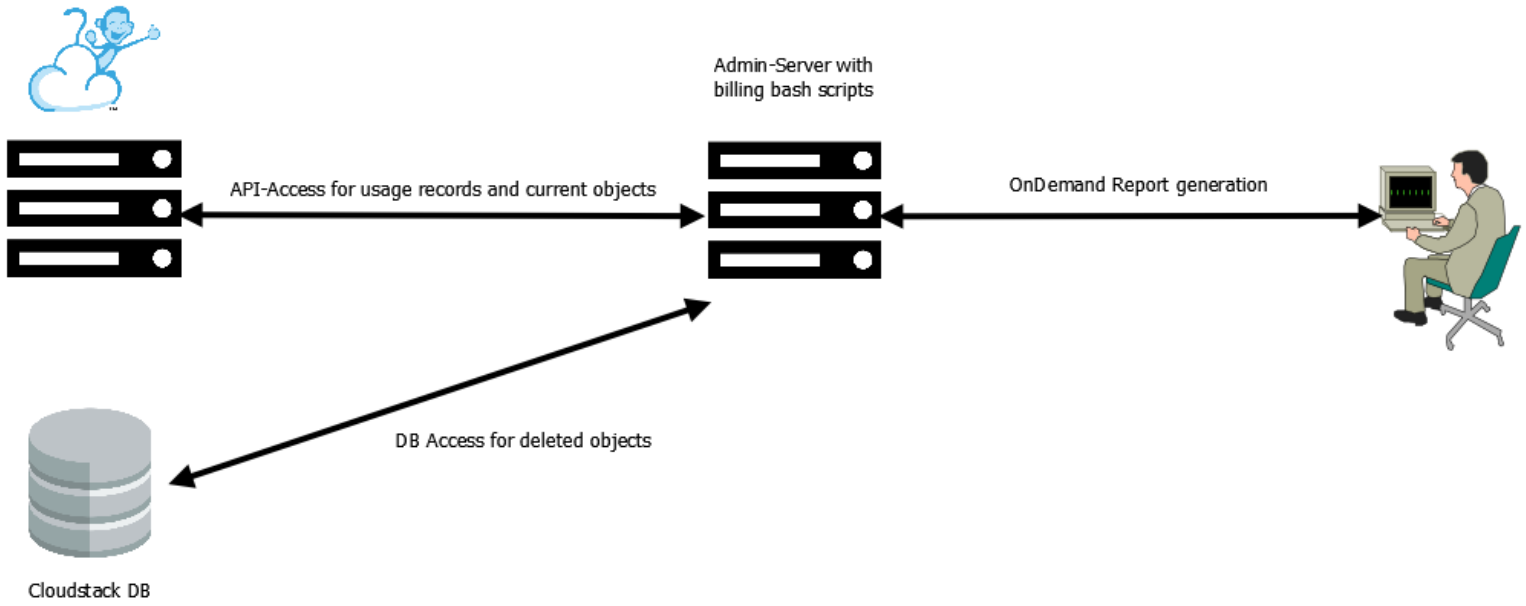
Our infrastructure and customer design

- Structure of resources
  - Cloudstack Domain:
    - respresents the customer like the internal customer „itelligence“
    - useraccounts will be setup per customer in our LDAP structure
  - Cloudstack Project:
    - projects are used to separate different customer landscapes inside domains
    - infrastructure Services instance in each project/landscape (automation,mon,bkp)

Our first steps with cloudstack billing

Our first steps with cloudstack billing

- Quick solution to get an overview of the consumed resources
- Former teammember wrote bash scripts in a very short amount
- Queried data from cloudmonkey (API) and directly from the MySQL Database
- Small part of Jasper Reports used to generated documents in different formats
- Files were sent via mail to the administrators



Our first steps with cloudstack billing

- Problems:
  - High amount of bashscripts which depended on each other were hard to maintain
  - No unique datasource (API + some informations from MySQL)
  - No own database for prices/discounts and other informations
  - No UI/API
  - All information had to be queried from cloudstack each time

- Solution attempt:
  - Own database
  - Sync with Cloudstack
  - Build UI/API

New approach as a webservice

New approach as a webservice

- Features of the billing system:
  - Creation of different reports wich can also work with filters:
    - Startdate and enddate
    - Domainid, Projectid, Virtualmachineid, Volumeid
  - Export reports in different formats
  - Sync of usage and metadata from cloudstack → implemented as cronjob
  - Managing prices for Resources (CPU,RAM,DISK) Services (Backup, Monitoring) and Packages (Gold-Support, Silver-Support)
  - Managing the allocation of services to packages (Tags on VM objects in Cloudstack)
  - Managing Discounts on Domain, Project, VM and Volume level
  - Webui and API with LDAP and local User authentication
  - Permissions based on API endpoints which can be managed via the UI

New approach as a webservice

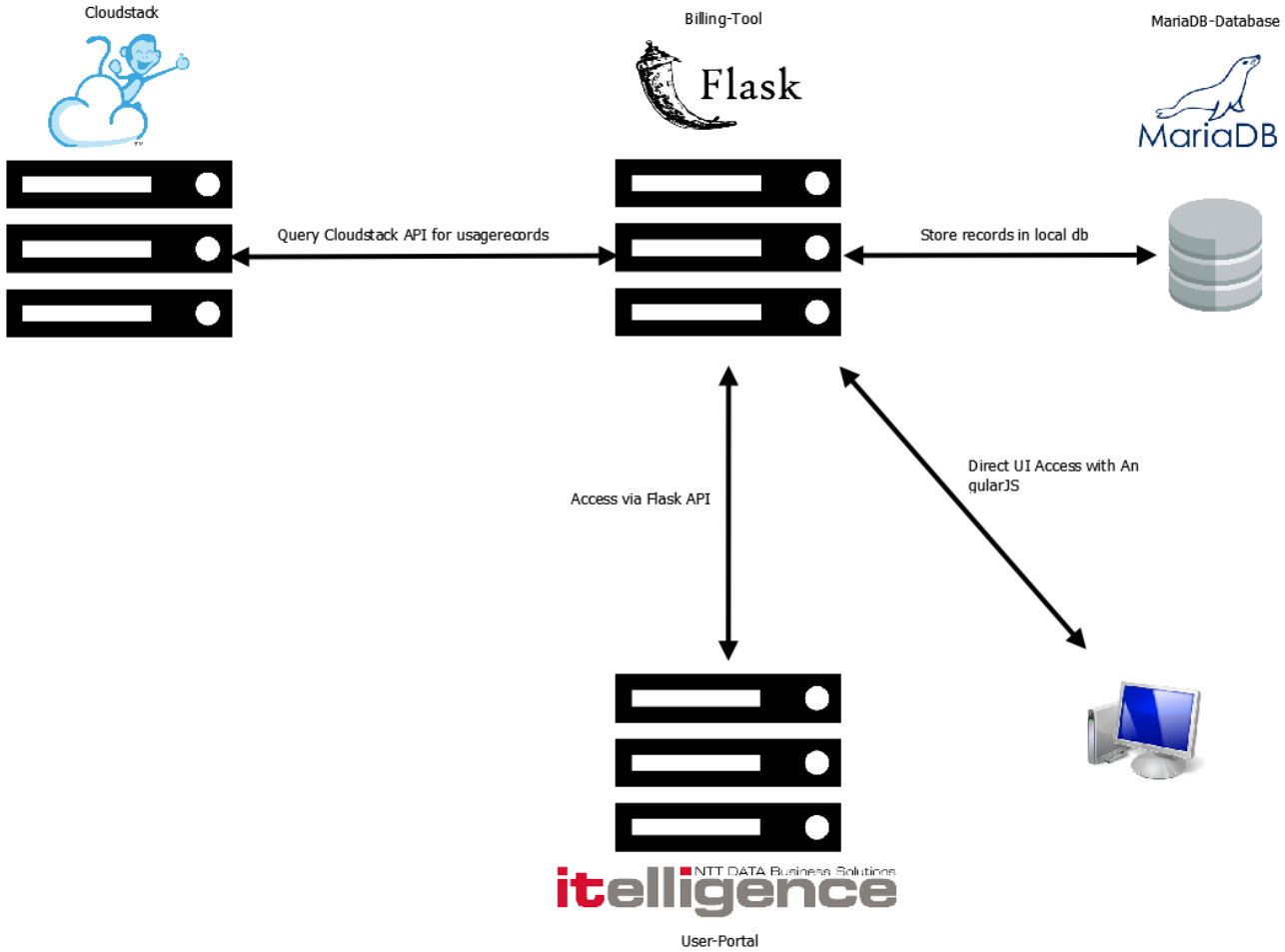
- Syncronisation of Tags:
  - All tags of a VM or volume will be synced
  - We define special tags to represent Servicelevels or packages
  - Billing can be deactivated through tags like „Billing:noCPU“, „Billing:noRAM“ or „Billing:noDisk“
  - Creation and removed dates will be considerd in the reports
  - For next version we maybe move tagging directly to billing-tool

New approach as a webservice

- Used components:
  - Database: MariaDB
    - Reasons: Knowleadge already existing, good and solid foundation
  - Backend Framework: Flask (Python)
    - Reasons: perfect for building APIs, not overloaded as other frameworks, no steep learning curve
  - Frontend Framework: AngularJS + Bootstrap
    - Reasons: Knowleadge already existing, easy to learn

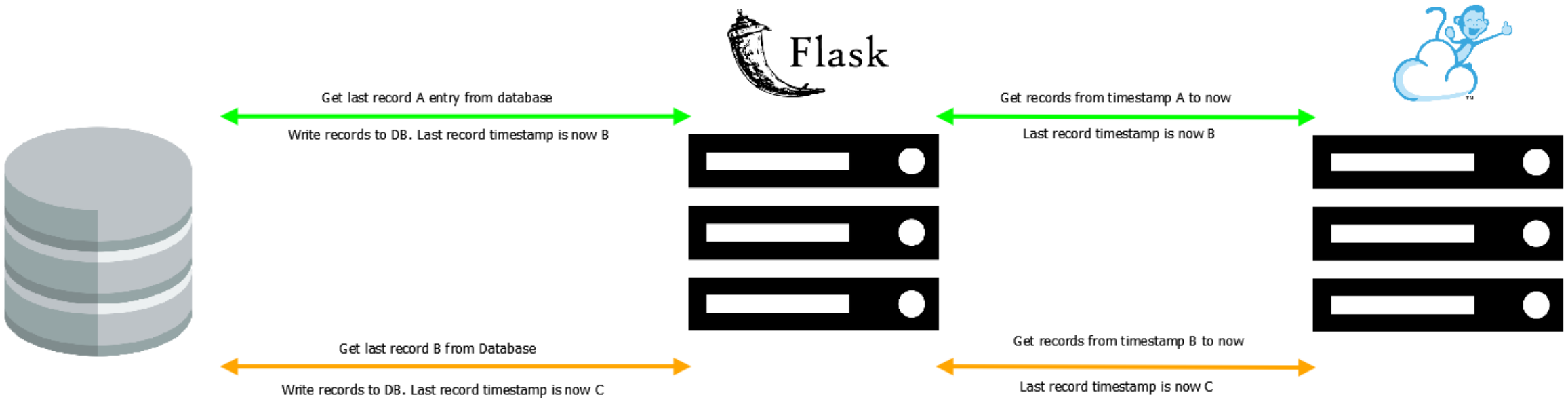
New approach as a webservice

- The big picture:



New approach as a webservice

- The sync process:

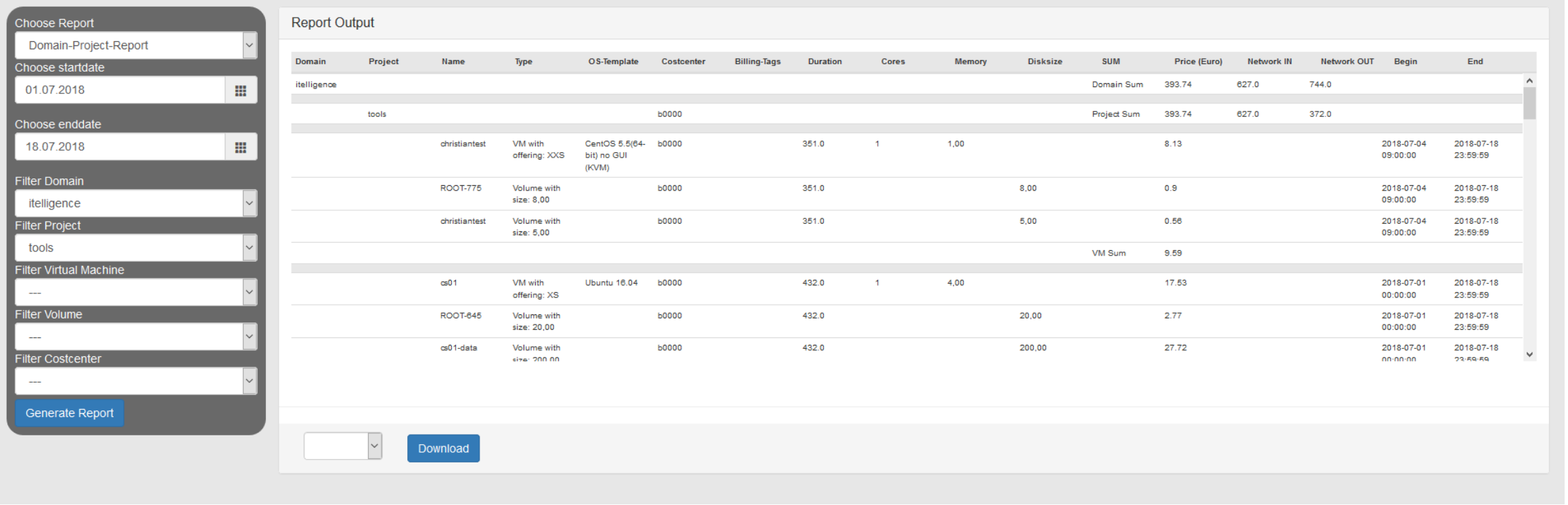


New approach as a webservice

- Functions of Report-Module:
  - Selection of different reporttypes
  - Selection of timerange (start,end)
  - Selection of filters (domain,project,vm…)
  - Filter by CostCenter (special for our needs)
  - Export to CSV and XLS (PDF planned)

New approach as a webservice

- Reports:



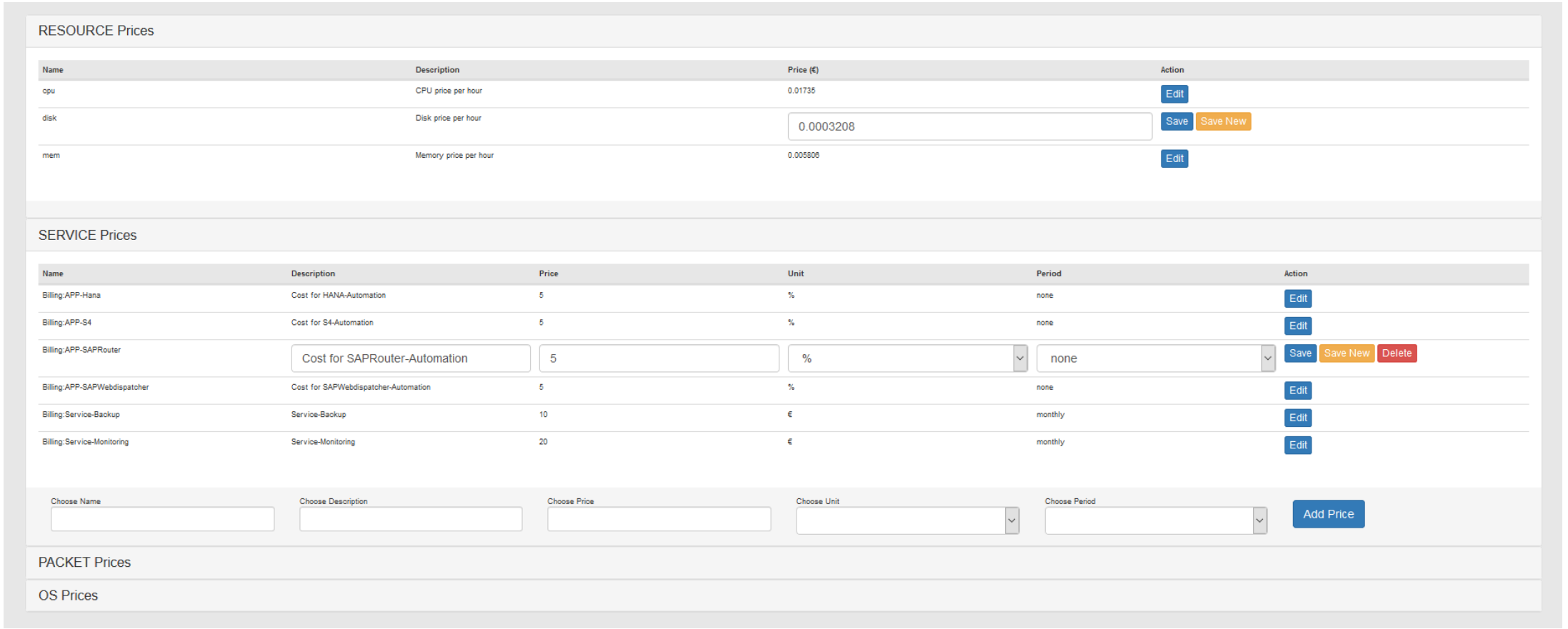
Domain	Project	Name	Type	OS-Template	Costcenter	Billing-Tags	Duration	Cores	Memory	Disksize	SUM	Price (Euro)	Network IN	Network OUT	Begin	End
itelligence											Domain Sum	393.74	627.0	744.0		
	tools				b0000						Project Sum	393.74	627.0	372.0		
		christiantest	VM with offering: XXS	CentOS 5.5(64-bit) no GUI (KVM)	b0000		351.0	1	1,00			8.13			2018-07-04 09:00:00	2018-07-18 23:59:59
		ROOT-775	Volume with size: 8,00		b0000		351.0			8,00		0.9			2018-07-04 09:00:00	2018-07-18 23:59:59
		christiantest	Volume with size: 5,00		b0000		351.0			5,00		0.56			2018-07-04 09:00:00	2018-07-18 23:59:59
											VM Sum	9.59				
		cs01	VM with offering: XS	Ubuntu 16.04	b0000		432.0	1	4,00			17.53			2018-07-01 00:00:00	2018-07-18 23:59:59
		ROOT-645	Volume with size: 20,00		b0000		432.0			20,00		2.77			2018-07-01 00:00:00	2018-07-18 23:59:59
		cs01-data	Volume with size: 200,00		b0000		432.0			200,00		27.72			2018-07-01 00:00:00	2018-07-18 23:59:59

New approach as a webservice

- Functions of Price-Module:
  - Manage prices for:
    - Resources (CPU,Memory,Disk)
    - Services (Backup,Monitoring…)
    - Packages (Offers which represent service levels which include services)
    - OS Images (SLES,Windows…)
  - Create new prices (valid from time of adding)
  - Change prices (will be valid also for old reports)
  - Change prices with new tupel (will be valid for new reports)
  - Select unit and period of price

New approach as a webservice

- Prices:



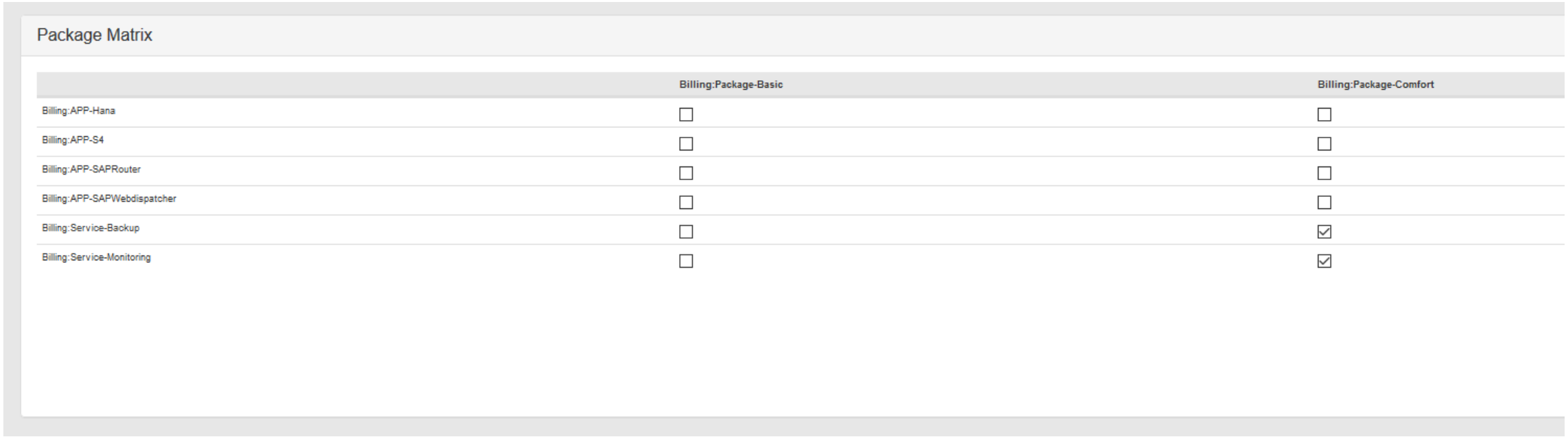
New approach as a webservice

- Functions Package-Matrix-Module:
  - Add or remove services from packages
  - Prices of services included in packages

    will not be included when customer has booked the package
  - Information for services and packages of each VM/Volume taken from Cloudstack tags

New approach as a webservice

- Package Mapping:



Package Matrix

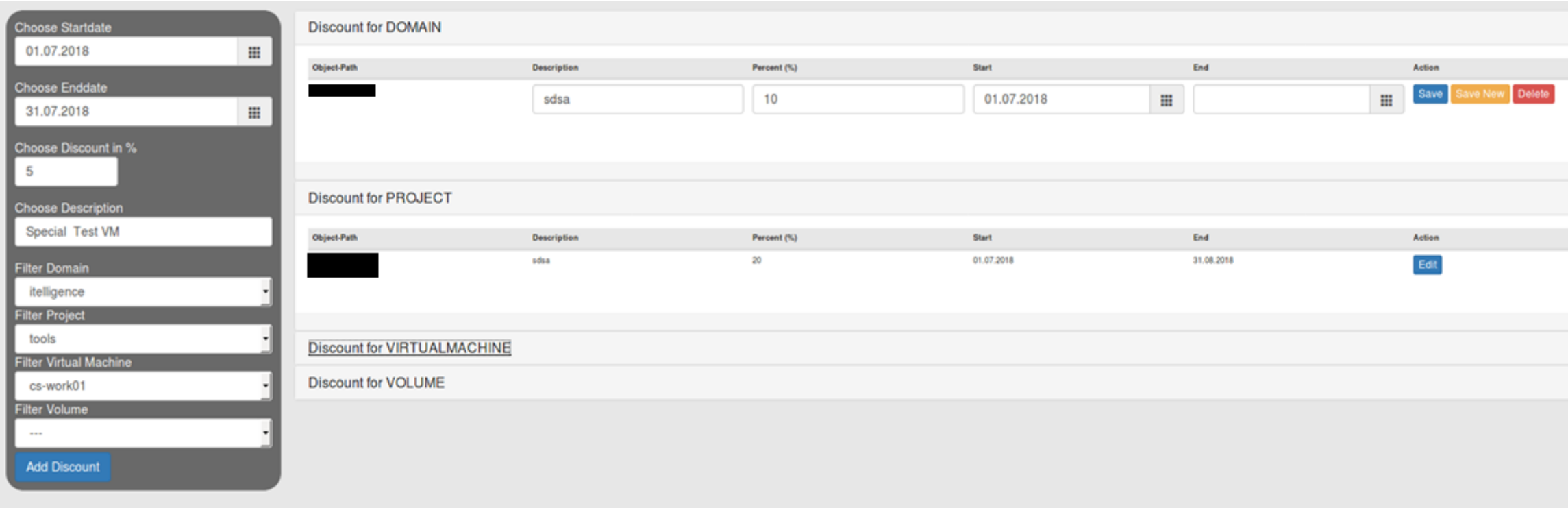
	Billing:Package-Basic	Billing:Package-Comfort
Billing:APP-Hana	☐	☐
Billing:APP-S4	☐	☐
Billing:APP-SAPRouter	☐	☐
Billing:APP-SAPWebdispatcher	☐	☐
Billing:Service-Backup	☐	☑
Billing:Service-Monitoring	☐	☑

New approach as a webservice

- Functions of Discount-Module:
  - Manage discounts for domains, projects, VMs and volumes
  - Choose duration for discounts (start, end)
  - Change discounts (save for all reports or save for new reports)
  - Permanent discounts are also possible

New approach as a webservice

- Discounts:

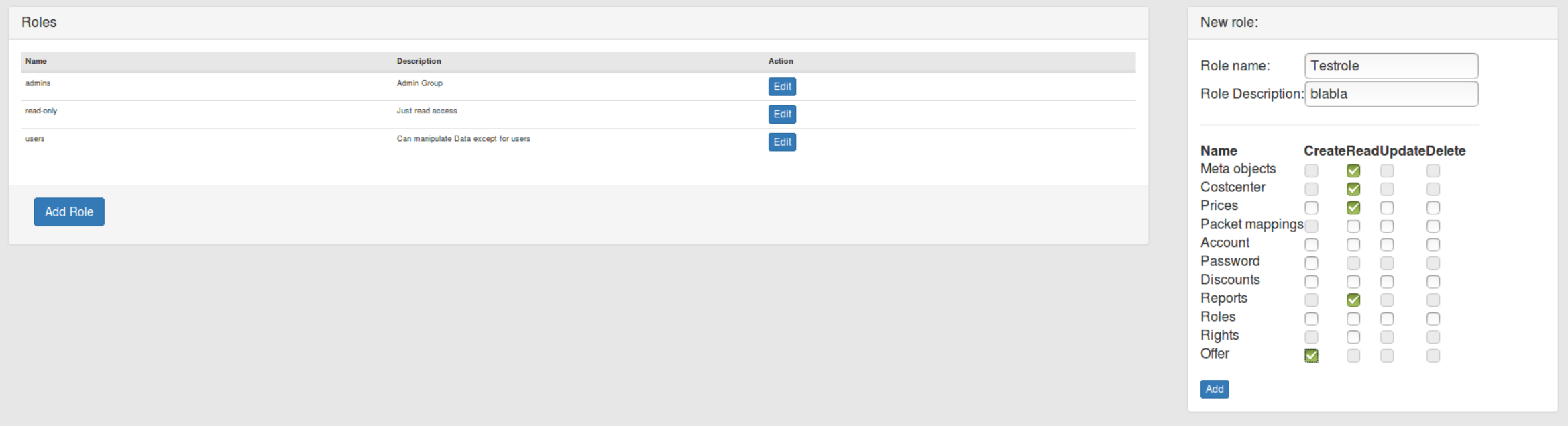


New approach as a webservice

- Functions of Role-Module:
  - Management rights of roles for accessing the system
  - Rights management for different categories (Create,Read,Update,Delete)
  - Roles can be later attached to users

New approach as a webservice

- Access (Roles):

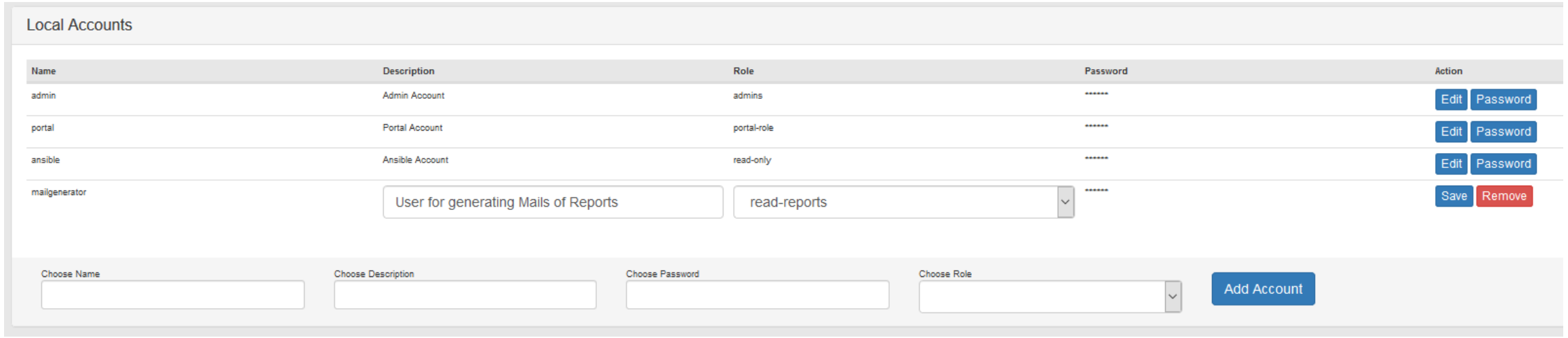


New approach as a webservice

- Functions of User-Module:
  - Manage local users (MariaDB)
  - Change/Set passwords for users
  - Change/Set Role of a User
  - Also planned for LDAP users

New approach as a webservice

- Access (User):



Outlook

- Build forecasting module for planned installations
- Add new export formats like PDF
- Improve UI
- Enable automatic mail generation for the customer
- Add visualization for cloudstack statistics
- Enable hardlinked prices for special customers

Questions?

Contact
Alexander Stock
Cloud Infrastructure Architect
alexander.stock@itelligence.de

itelligence Global Managed Services GmbH

We make the most of SAP® solutions!

Copyright itelligence AG - All rights reserved