Blurring the IaaS/PaaS Divide

Google Cloud Platform Developer Roadshow - 2014
Today: Cloud as a binary choice

Flexibility

Your code

IaaS

Agility

Your code

PaaS
Next Gen Cloud

Flexibility and Management
Managed VMs

- Flexibility of Compute Engine
- Access to native resources, machine types
- Auto-management of App Engine
- High Availability
- Image updates, security patches
- Access to all Managed Services
- Collocation with other services
- Developer productivity
Replica Pools

1 template $\Rightarrow$ many machines

Template
- image
- metadata
- bash cmds

Replica Pool
- actuator

Resource View
- Load Balancer
- VM

create/destroy
Compute as a spectrum

Flexibility

Your Code

Google Compute Engine

Agility

Manage your infrastructure
Compute as a spectrum

Flexibility

Your Code

Your Code

Replica Pools

Provisioning and health checking

Agility

Google Compute Engine

Manage your infrastructure
Compute as a spectrum

Flexibility

Your Code

Your Code

Your Code

Managed VMs

Replica Pools

Google Compute Engine

Agility

OS management, deployments, logging and monitoring

Provisioning and health checking

Manage your infrastructure
Manage your infrastructure

Flexibility

Your Code

Your Code

Your Code

App Engine Managed Runtimes

Managed VMs

Replica Pools

Google Compute Engine

Agility

Manage your serving stack

OS management, deployments, logging and monitoring

Provisioning and health checking

Manage your infrastructure
Managed VMs Demo
Managed VMs Demo

Sudoku Example
OpenCV

OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can take advantage of the hardware acceleration of the underlying heterogeneous compute platform. Adopted all around the world, OpenCV has more than 47 thousand people of user community and estimated number of
Peter Norvig’s Sudoku Solver

Solving Every Sudoku Puzzle

by Peter Norvig

In this essay I tackle the problem of solving every Sudoku puzzle. It turns out to be quite easy (about one page of code for the main idea and two pages for embellishments) using two ideas: constraint propagation and search.

Sudoku Notation and Preliminary Notions

First we have to agree on some notation. A Sudoku puzzle is a grid of 81 squares; the majority of enthusiasts label the columns 1-9, the rows A-I, and call a collection of nine squares (column, row, or box) a unit and the squares that share a unit the peers. A puzzle leaves some squares blank and fills others with digits, and the whole idea is:

A puzzle is solved if the squares in each unit are filled with a permutation of the digits 1 to 9.

That is, no digit can appear twice in a unit, and every digit must appear once. This implies that each square must have a different value from any of its peers. Here are the names of the squares, a typical puzzle, and the solution to the puzzle:
app.yaml

application: compute-demo
version: 1
runtime: python27
api_version: 1
threadsafe: yes

handlers:
- url: /favicon.ico
  static_files: favicon.ico
  upload: favicon.ico

- url: .*
  script: main.APP

libraries:
- name: webapp2
  version: "2.5.2"
- name: jinja2
  version: latest

backend.yaml

application: compute-demo
version: 1
module: solver
runtime: python27
vm: true
api_version: 1
threadsafe: yes

handlers:
- url: .*
  script: main_solver.APP

libraries:
- name: webapp2
  version: "2.5.2"

vm_settings:
  machine_type: n1-standard-1
  apt_get_install: python-opencv
app.yaml

application: compute-demo
version: 1
runtime: python27
api_version: 1
threadsafe: yes

handlers:
- url: /favicon.ico
  static_files: favicon.ico
  upload: favicon.ico

- url: .*
  script: main.APP

libraries:
- name: webapp2
  version: "2.5.2"
- name: jinja2
  version: latest

backend.yaml

application: compute-demo
version: 1
module: solver
runtime: python27

vm: true
api_version: 1
threadsafe: yes

handlers:
- url: .*
  script: main_solver.APP

libraries:
- name: webapp2
  version: "2.5.2"

vm_settings:
  machine_type: n1-standard-1
  apt_get_install: python-opencv
$ gcloud preview app deploy .

https://developers.google.com/cloud/sdk/gcloud-app
Live Demo!

[Image of a web interface with options for Sudoku Solver, Capture and Solve, and Upload and Solve. A video source selection is also visible.]
Sudoku architecture

- App Engine
- Task Queue
- Backend workers
- VM

Google Cloud Platform
from google.appengine.api import modules

modules.set_num_instances(42)

https://developers.google.com/appengine/docs/python/modules/
https://developers.google.com/appengine/docs/java/modules/
$ git clone
https://github.com/GoogleCloudPlatform/appengine-opencv-sudoku-python
IaaS or PaaS is a false dichotomy

Removing the split with great new technology

This is only the beginning!

https://developers.google.com/appengine/docs/managed-vms/

http://goo.gl/ykbnsd