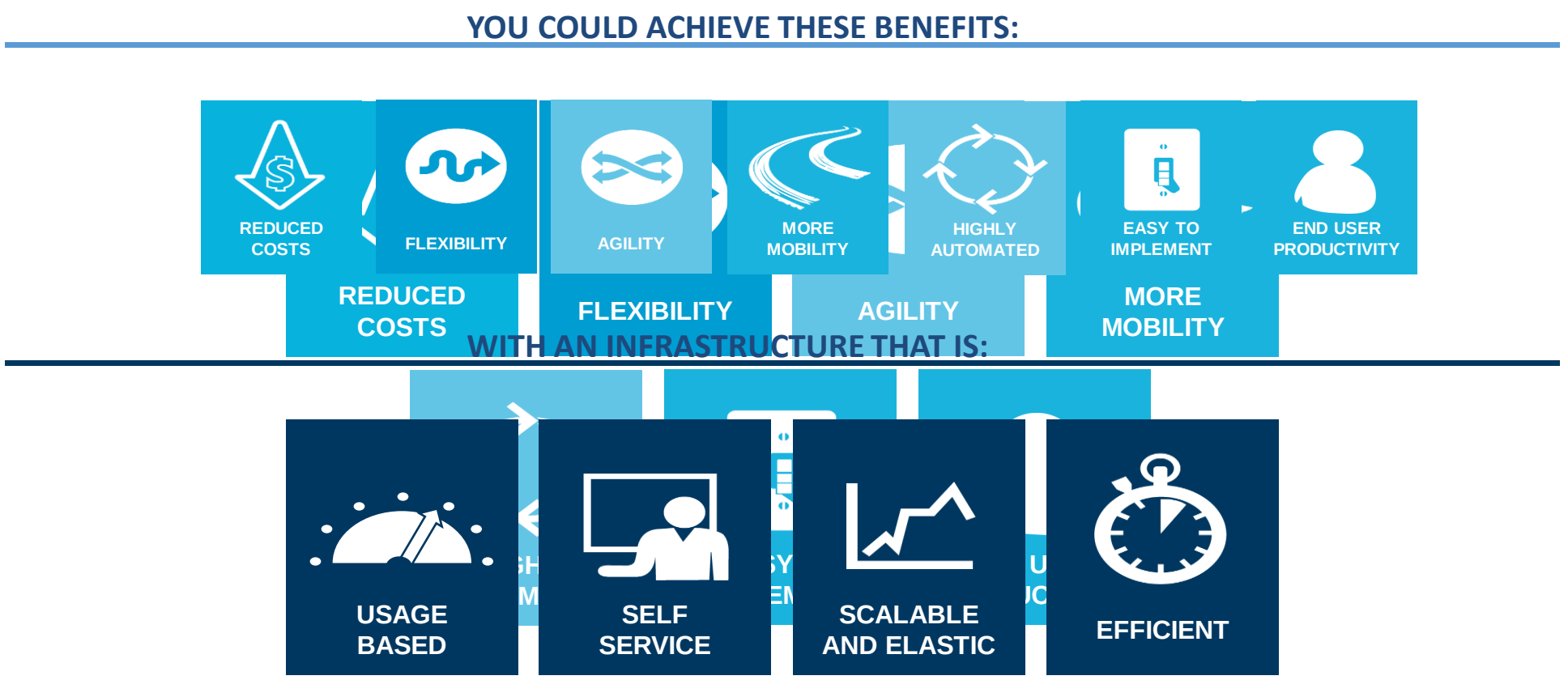
Platform as a Service (Paas)

Cloud based networks

vitmma 02

Intro – clouds revisited

Business Benefits of the Cloud



Cloud Computing Terms

IaaS

Infrastructure-as-a-Service

host

PaaS

Platform-as-a-Service

build

SaaS

Software-as-a-Service

consume

Defining the Cloud

Deployment Models

- Private Cloud
- Public Cloud
- Hybrid Cloud

Service Models

- Infrastructure as a Service (IaaS)
- Platform as a Service PaaS
- Software as a Service SaaS

Paas – what it is

Platform as a Service – the services

- app deployment
- scaling (horizontal, vertical, auto)
- load balancing
- health monitoring, auto recovery
- logging service
- external/internal services, marketplace

Advantages of Paas – the Platform assures

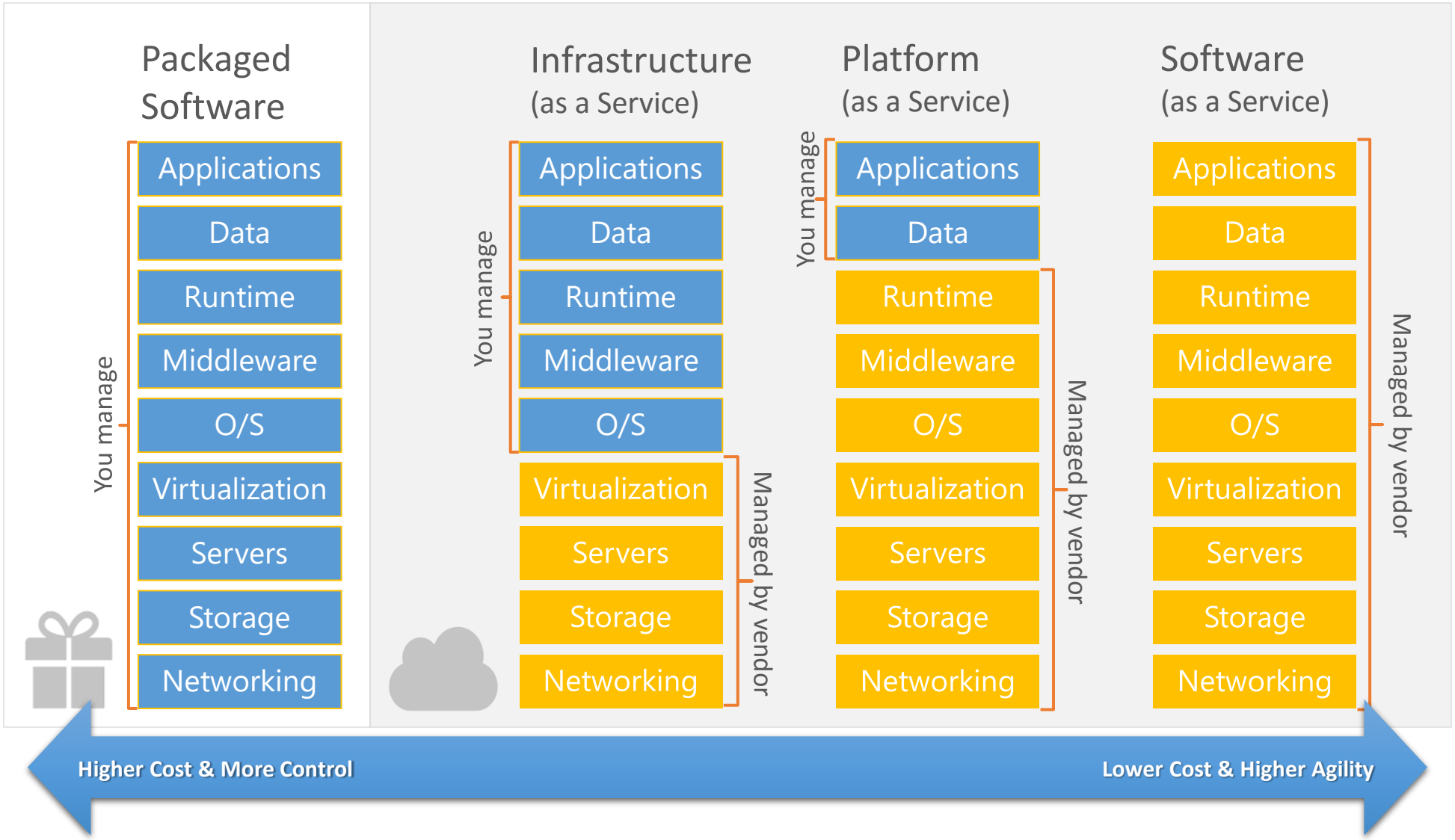
- deployment
- load balancing
- high availability
- log aggregation
- scaling
- Image mgmt –
  - Libraries, kernel versions
  - Security updates

Advantages of Paas – the Platform assures

- deployment
- load balancing
- high availability
- log aggregation
- scaling
- Image mgmt –
  - Libraries, kernel versions
  - Security updates

Tenant's problem: just develop your own app

Compare the *aaS-es



Comparing the *aaS-es

- Iaas: The end user maintains control of the operating system and applications on the hardware.
- Paas: end user has to development, testing, deployment, and ongoing maintenance of applications
- Saas: end users pay on a per-use basis

Paas in a Nutshell

Comprehensive set of services that enable you to quickly build, deploy and manage applications across a global network of Microsoft-managed datacenters

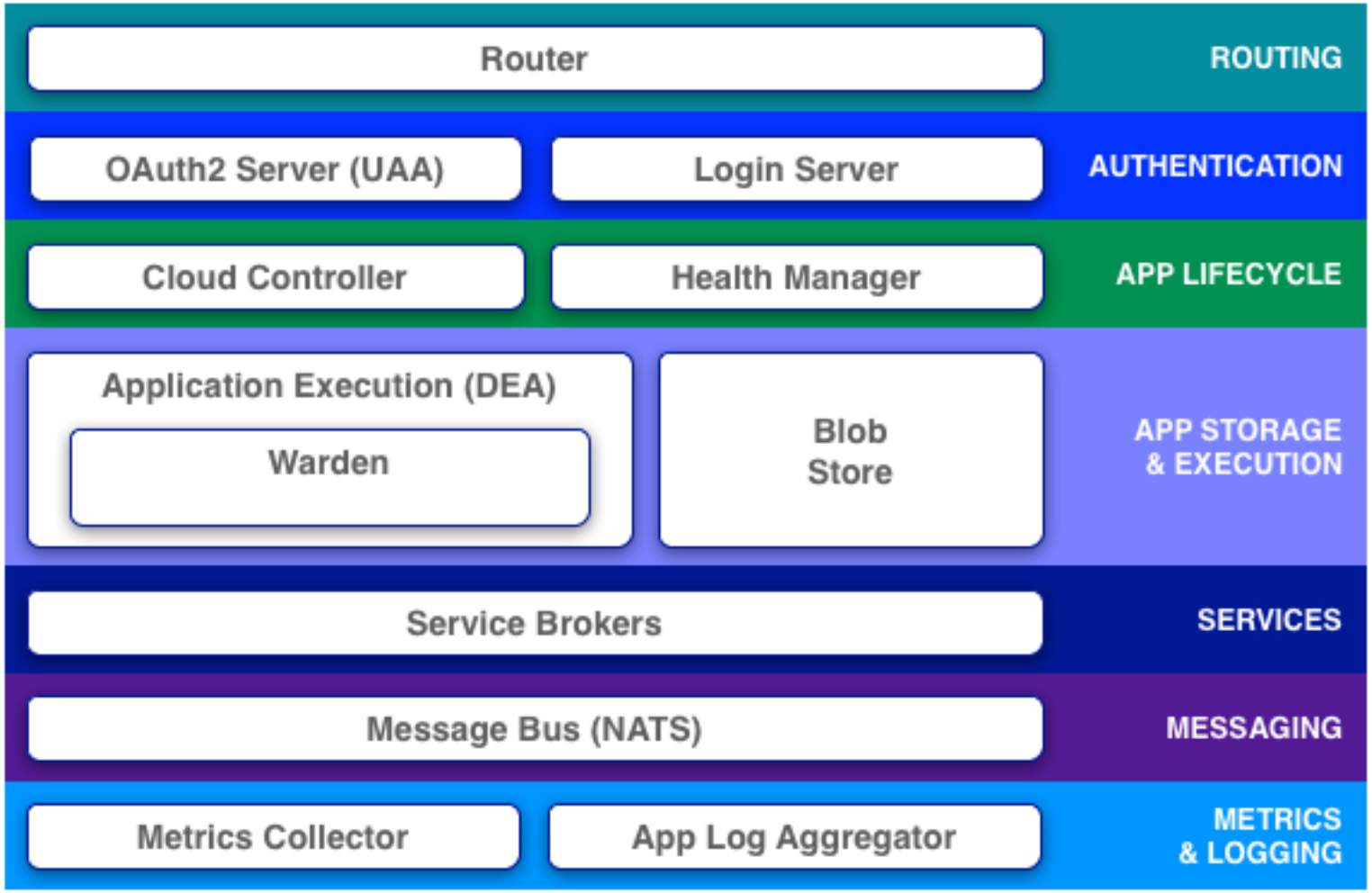
- Services that help develop and test apps
  - infrastructure is maintained by the provider
- Reduced infrastructure complexity
  - more effective overall application development
- Runtime environments are usually lock-in free
  - but might create lock-ins to provider specific infrastructure
- Usually simple network topology and access control
  - build your services as they would be open to the Internet
- The features and services provided vary a lot
  - from simple customizable runtime (CloudFoundry) to full marketplace of services (Heroku)

Toolchain as a Service

- Manage your project
  - Trello, Jira OnDemand, Sprint.ly, PivotalTracker, ...
- Create your code
  - Cloud9, Koding, Nitrous, ...
- Host your code
  - GitHub, Bitbucket, ...
- Build your code
  - Codeship, Travis CI, CloudBees, Drone, ...
- Test your code
  - BrowserStack, Sauce Labs, Xamarin Test Cloud, Blitz, ...
- Distribute your code
  - npm, Bintray, Maven Central, PyPI, Docker Hub, ...

Paas Architecture

Cloud Foundry - Architecture



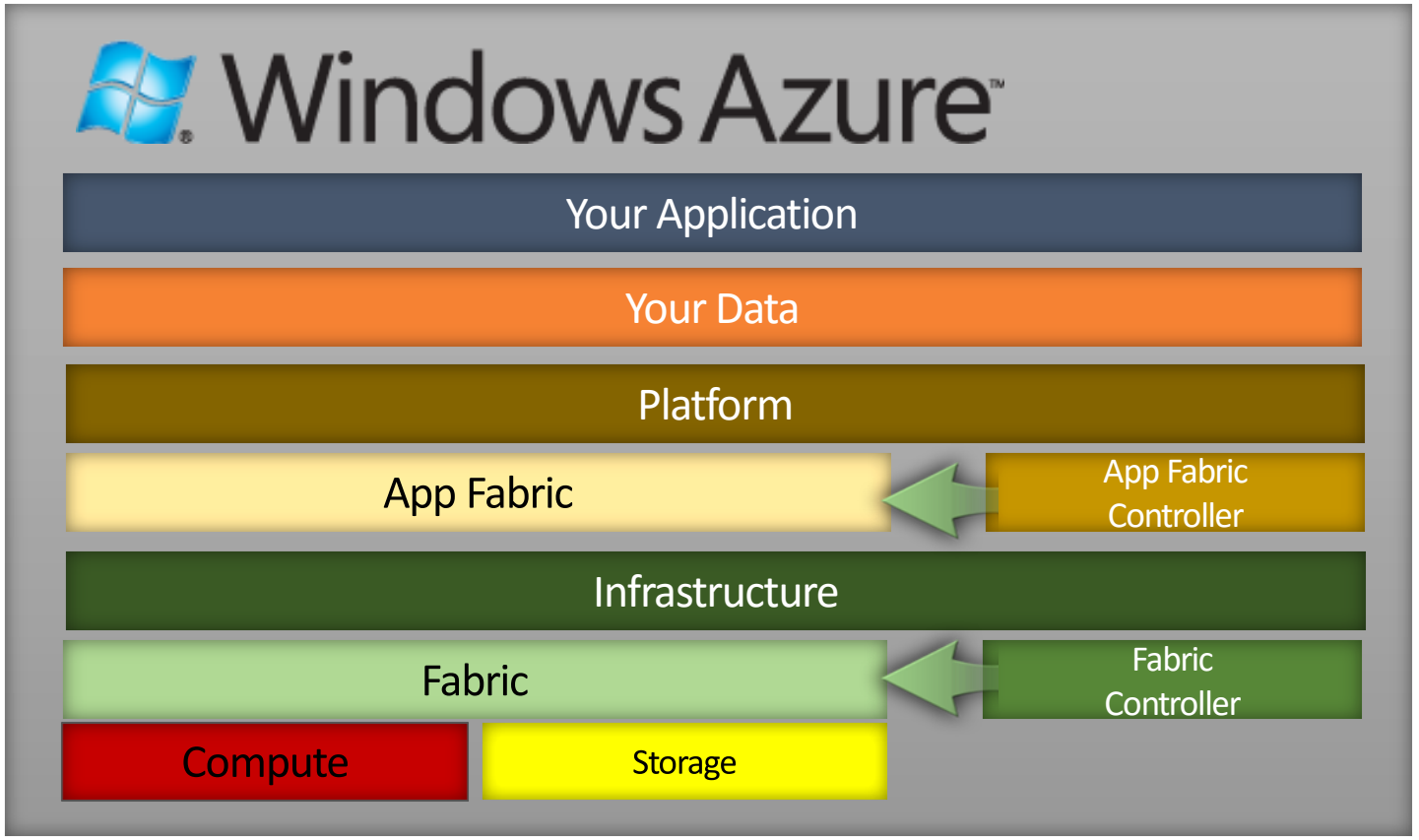
Heroku architecture

- Reverse Proxy by nginx
  - terminates SSL
  - forwards to cache layer
- HTTP Cache by Varnish
  - returns cached pages immediately
  - forwards to routing mesh
- Routing Mesh written in Erlang
  - routes to an existing dyno
  - spawns a dyno if none available
- Dyno Grid ('railgun' servers)
  - AWS hosted EC2 instances
  - multiple dynos per server

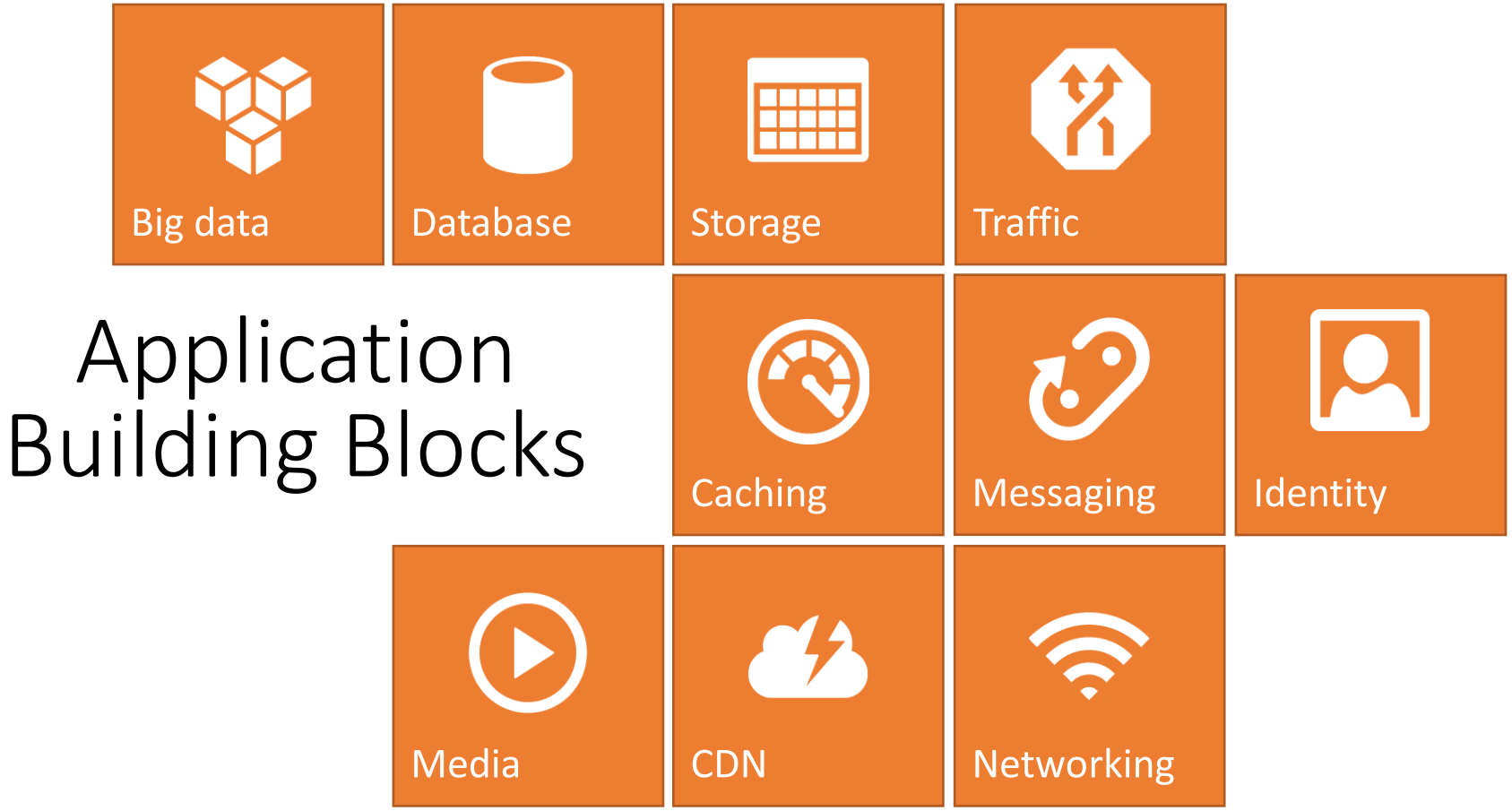
Heroku termonology

- **Application** source code and description of any dependencies
- **Procfile** list of process types – named commands to be executed
- **Deployment** sending application to Heroku using git or dropbox
- **Buildpack** compilation process that creates a slug from application
- **Slug** bundle of application, language runtime and compilation output
- **Dyno** isolated, virtualized Linux container for application runtime
- **Release** append-only ledger of slugs, config vars and add-ons
- **Config var** configuration data hosted independently of source code
- **Add-on** easily attachable third party cloud services
- **Logplex** collates logs from all running dynos and other components

WinAzure - Architecture



Building Block Services



Application Building Blocks

Use case: Win Azure

Three Main Components

Virtual machines

Cloud services

Web sites

Virtual Machines

Virtual Machines

Windows Server and Linux

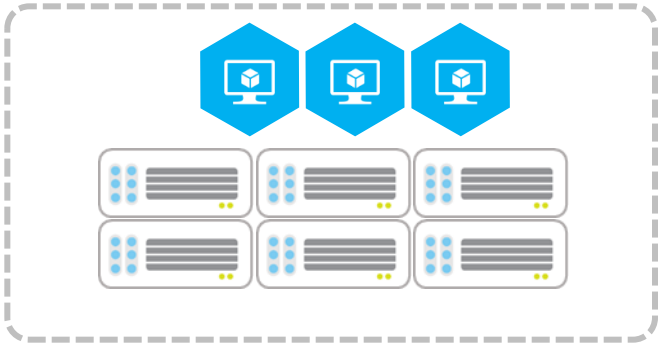
Flexible Workload Support

Virtual Private Networking

Virtual machine portability

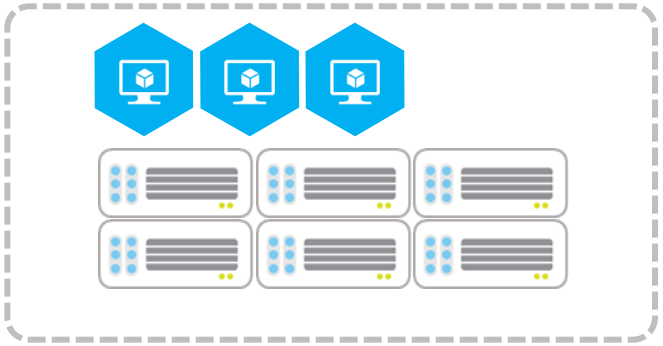
Windows Azure

Your Data Center



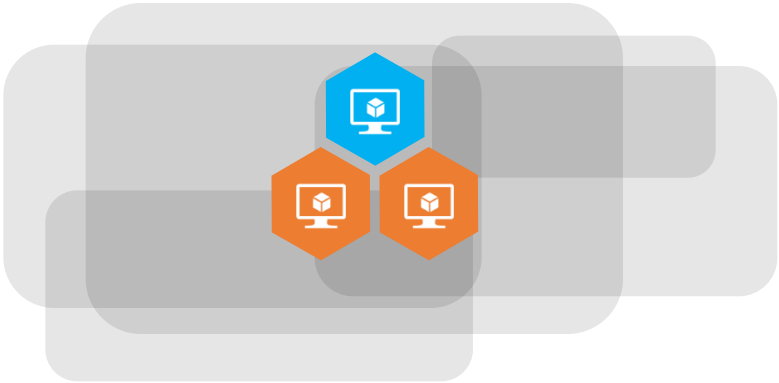
Windows Azure

Your Data Center

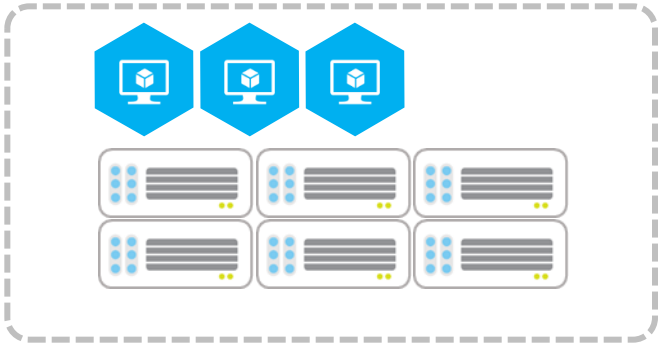


Windows Azure

Other Service Providers



Your Data Center

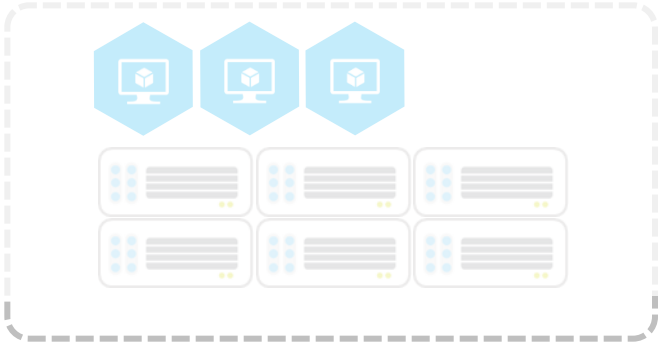


Windows Azure

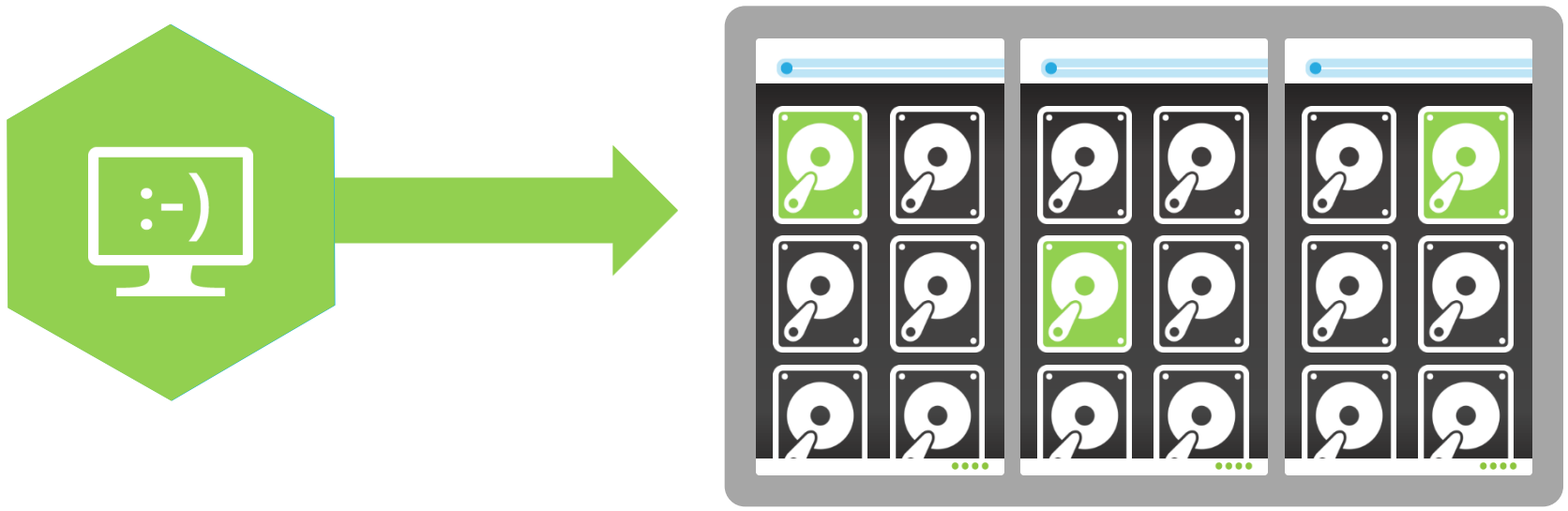
Other Service Providers

no lock-in

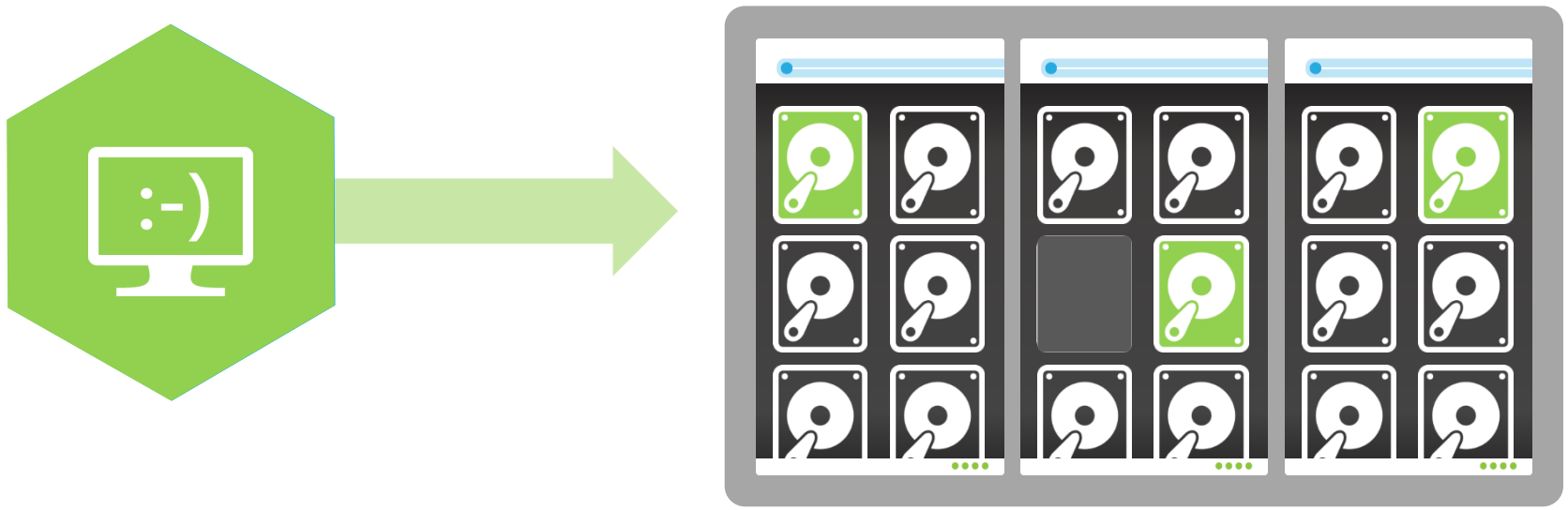
Your Data Center



Windows Azure Storage



Windows Azure Storage

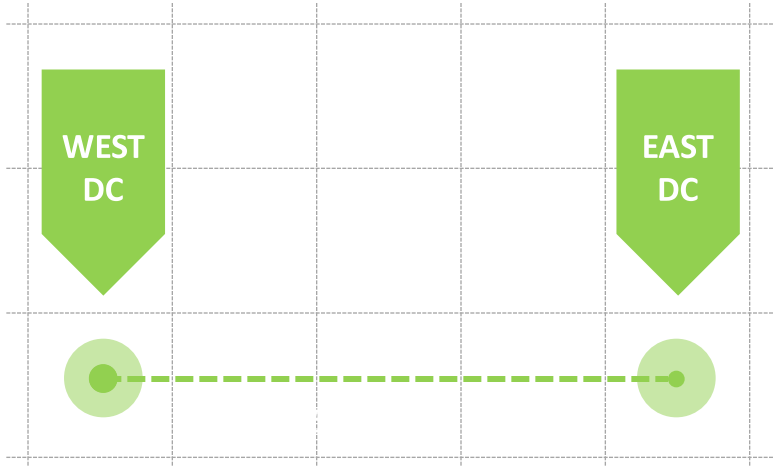


Windows Azure Storage

VM with persistent drive

Reliable and always on

Windows Azure Storage



Continuous storage geo-replication

Windows Azure Storage

Web Sites

Web sites

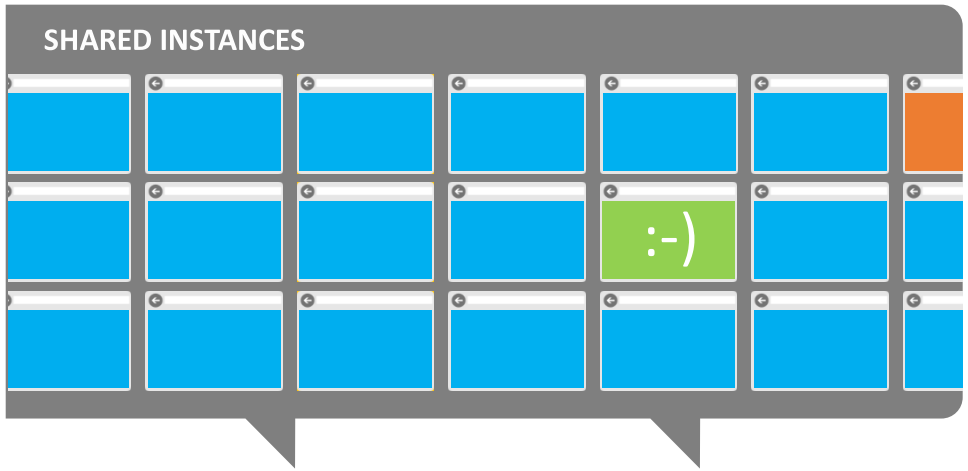
Build with ASP.NET, Node.js or PHP

Deploy in seconds with FTP, Git or TFS

Start for free, scale up as your traffic grows

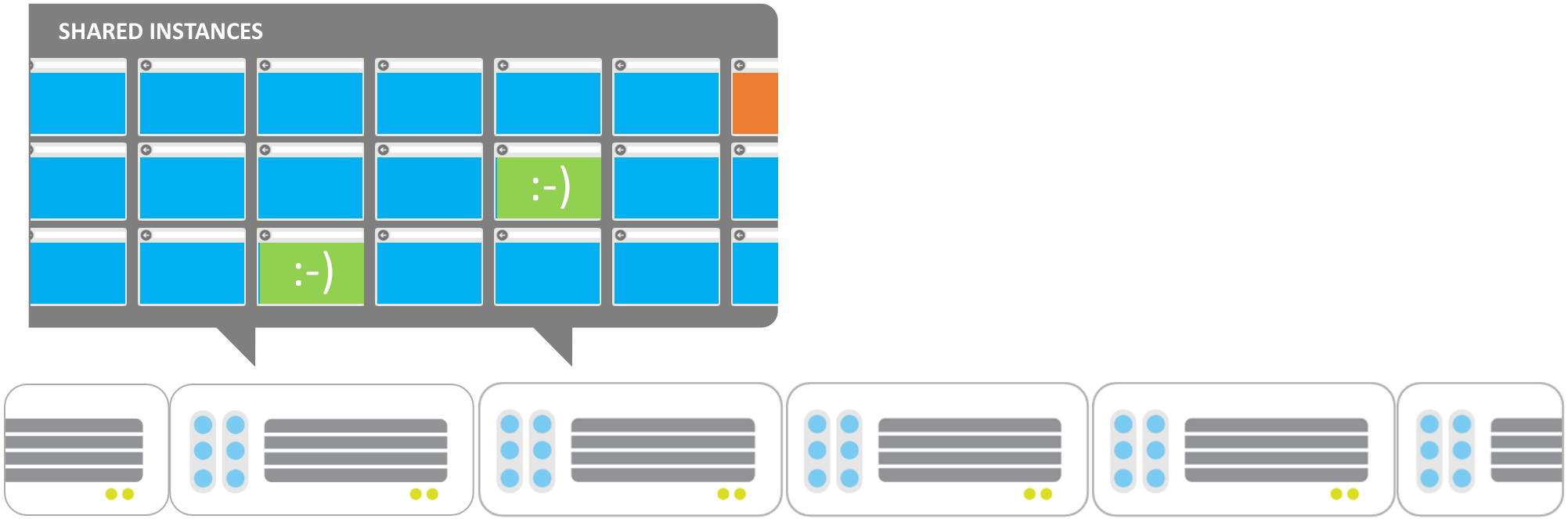
shared reserved

shared 1

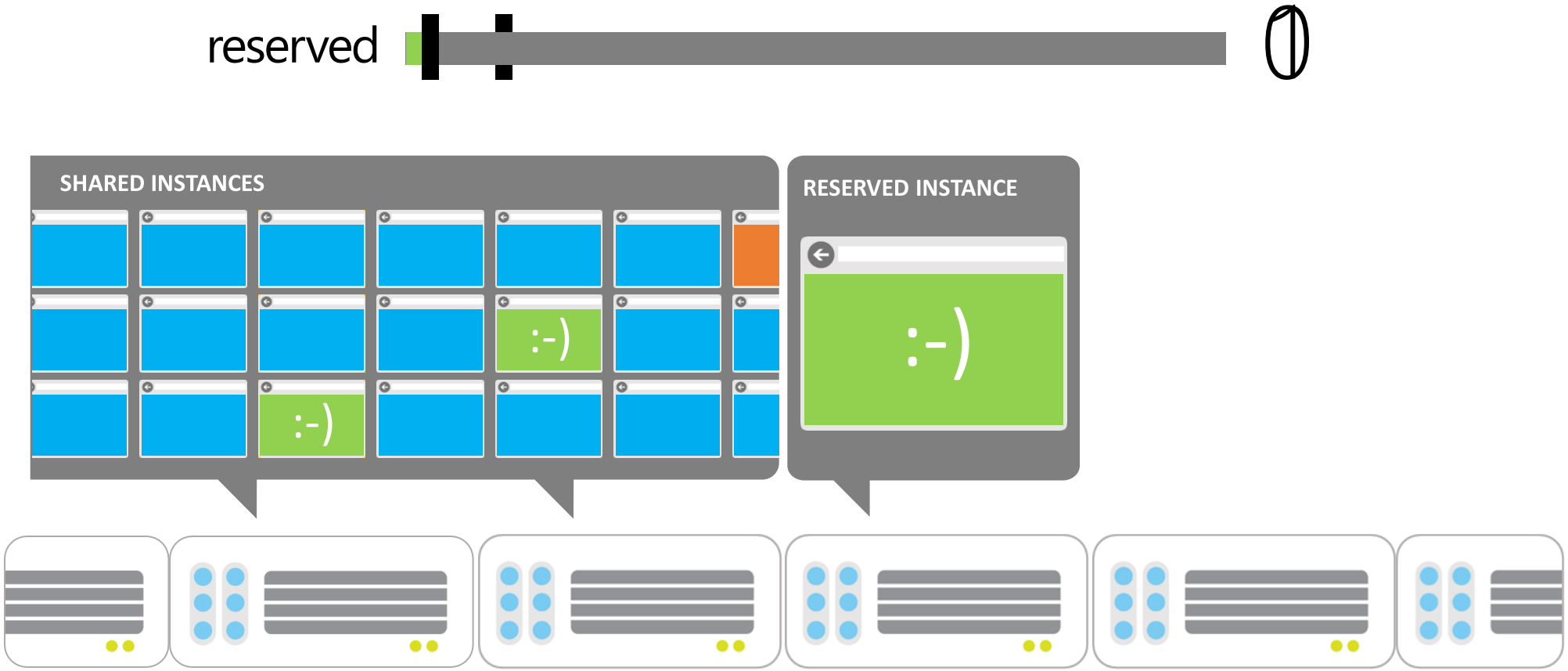


shared reserved

shared 2

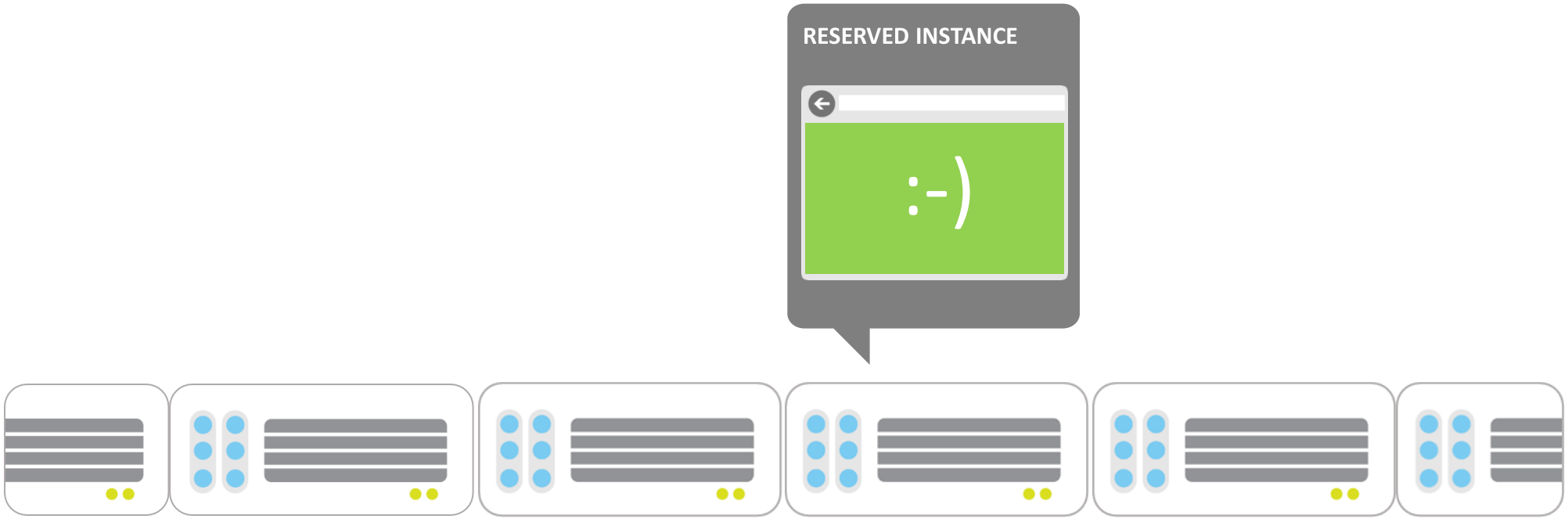


shared reserved



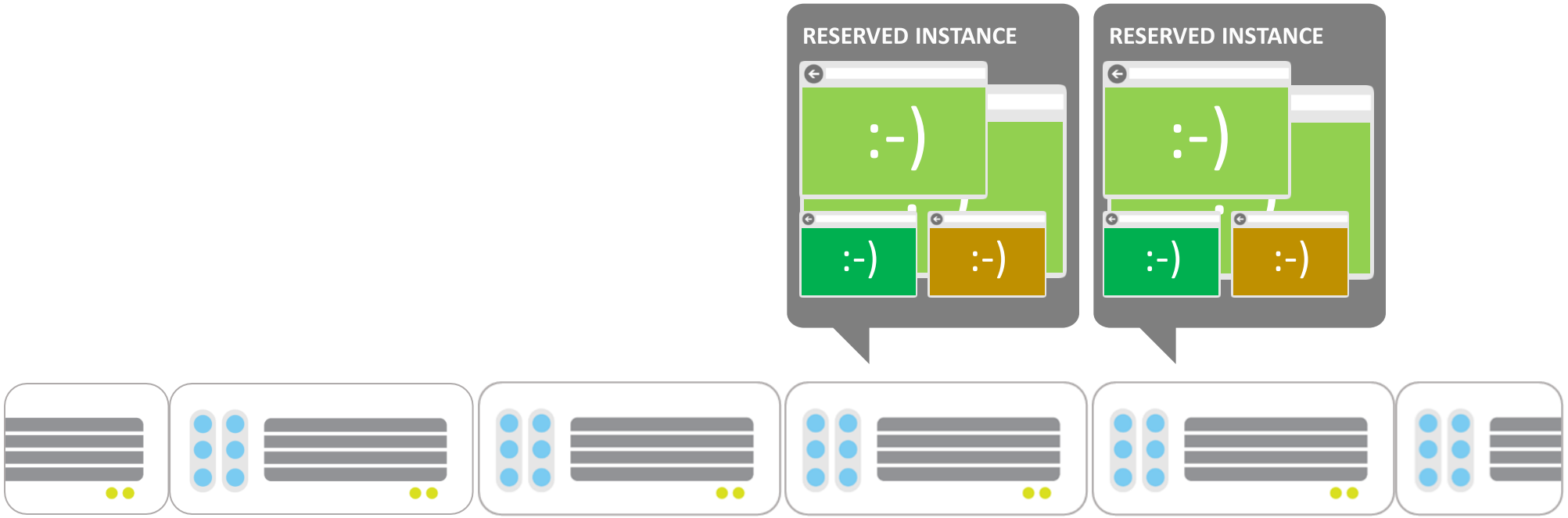
shared reserved

reserved 2



shared | reserved

reserved 2



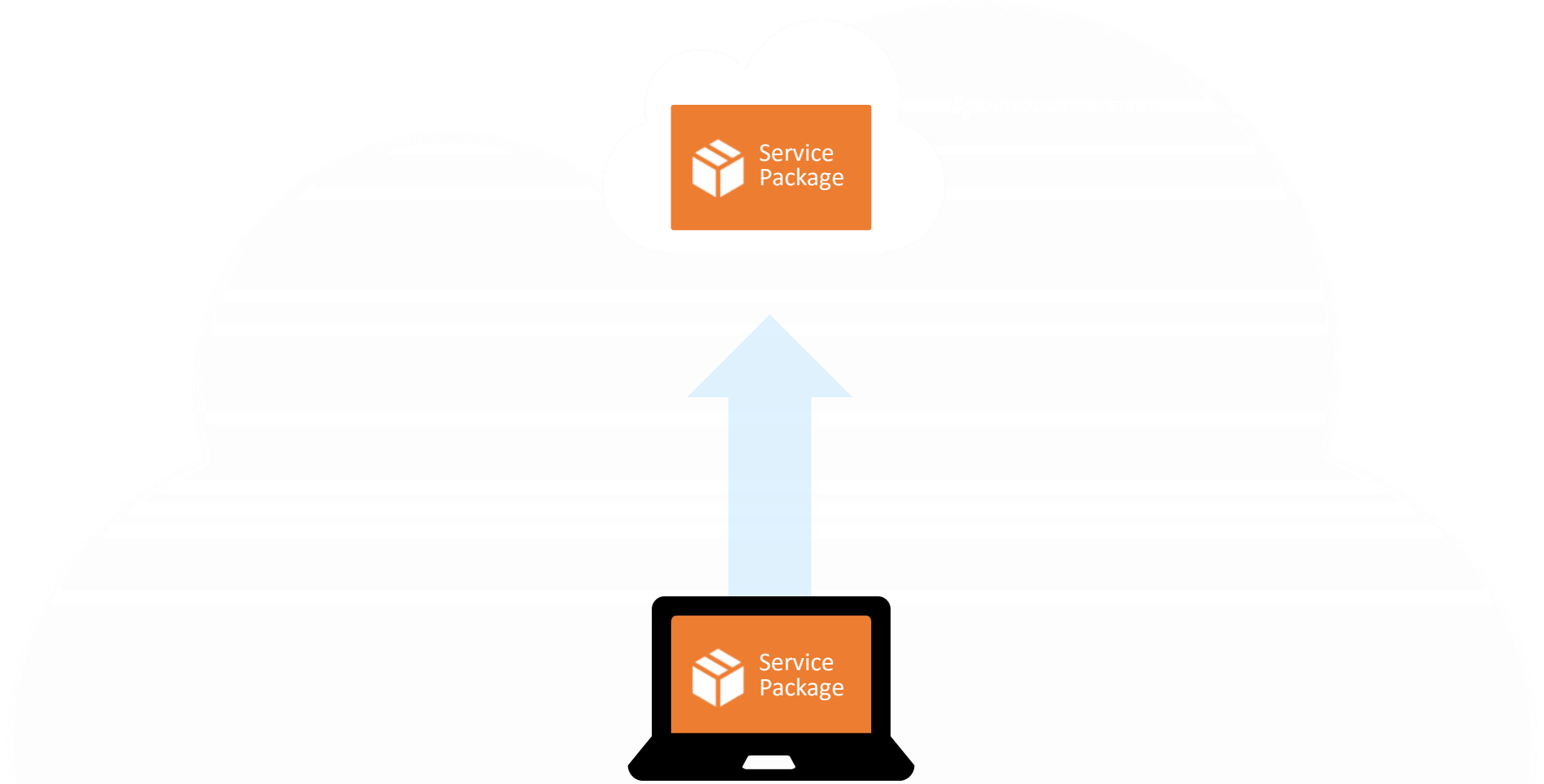
Cloud Services

Cloud services

Build infinitely scalable apps and services

Support rich multi-tier architectures

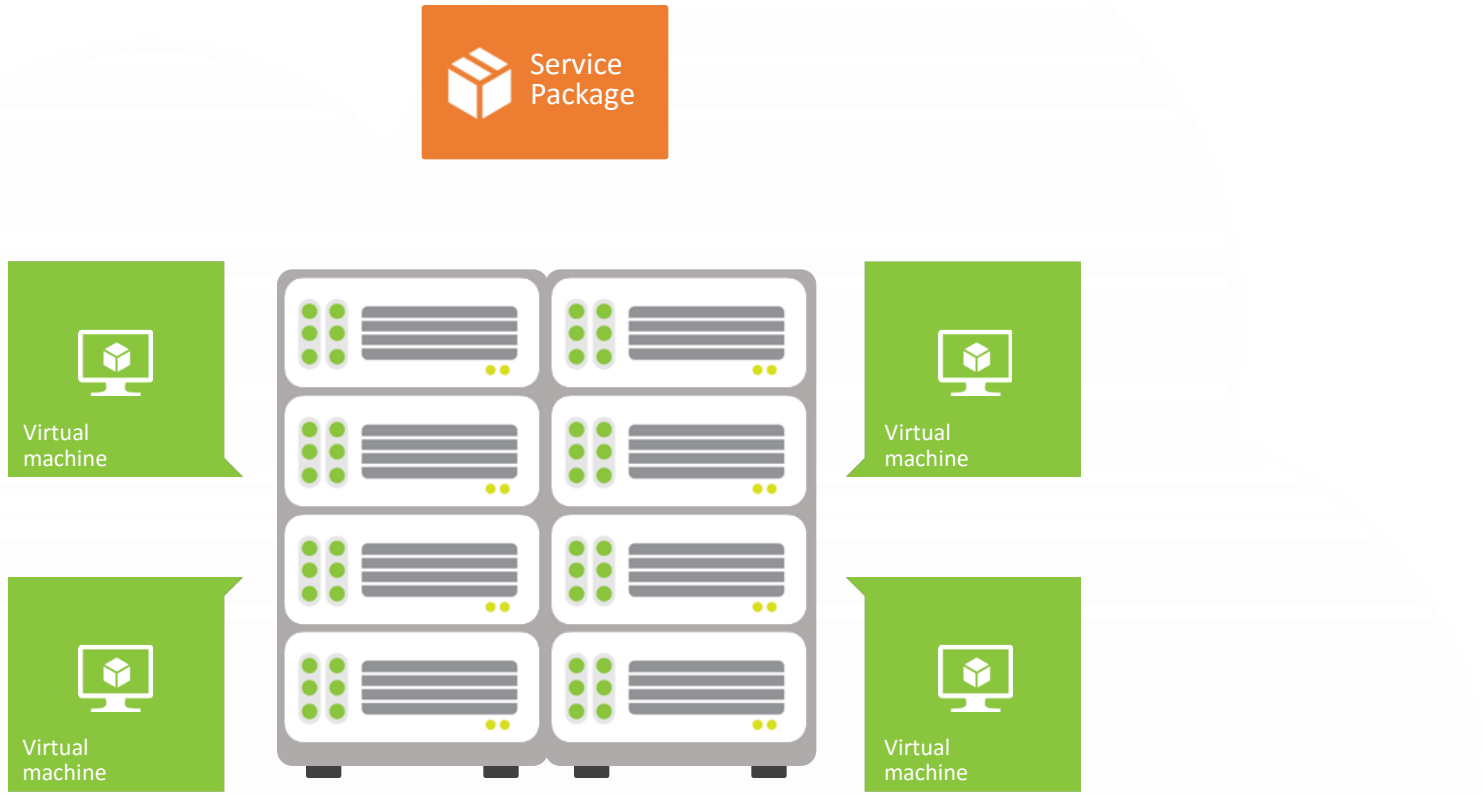
Automated application management


Service
Package
Service
Package

Provision Role Instances

Deploy App Code

Configure Network



Provision Role Instances

Deploy App Code

Configure Network

Provision Role Instances

Deploy App Code

Configure Network

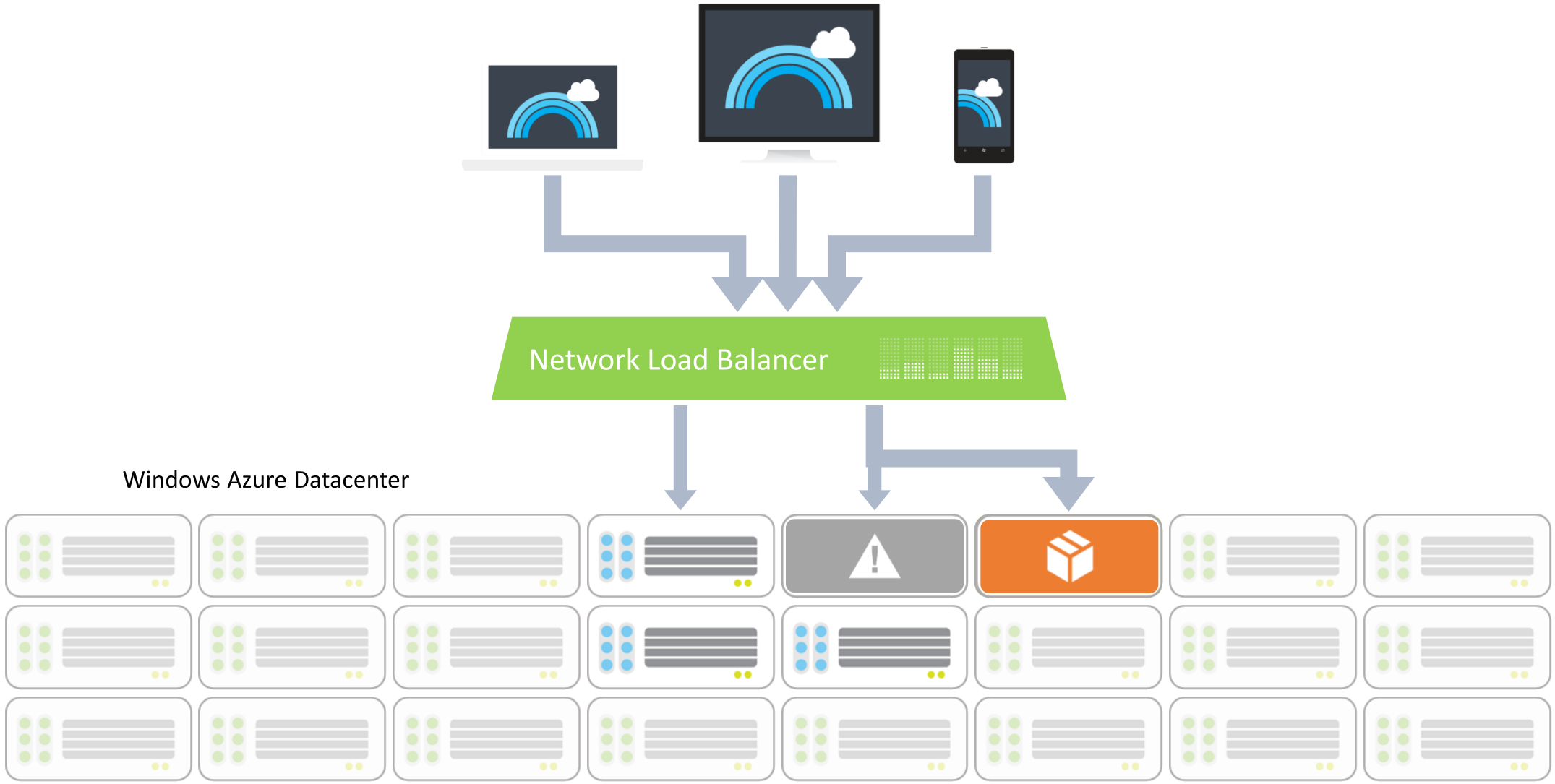
Provision Role Instances

Deploy App Code

Configure Network

← Network load-balancer configured for traffic

Windows Azure Datacenter


Network Load Balancer
Windows Azure Datacenter

Telco Grade Paas

Problems with the Paas

No telecom-ready PaaS

PaaS is coming from the IT (web) world

- Általában HTTP load balancer van csak,
  - egyéb protokollok (SIP, diameter, TCP session, stb.) jellemzően nem támogatottak.
- Belső állapotinformációk kiszervezése külső DB/cache
  - teljesítmény problémák
- Semmilyen QoS / válaszidő garancia nincs
- PaaS teljes mértékben elrejti a virtuális gépeket és hálózatokat
  - nem lehet közös VM-re tenni egymással sokat kommunikáló alkalmazásokat,
  - nem lehetséges hálózati optimalizációt (pl. Intel DPDK) kihasználni
- Nincs szabványos PaaS
  - a PaaS alkalmazásokat minden egyes operátor hálózatra fel kell(ene) készíteni

Requirements for a PaaS

- Actually Saas, PaaS,…
- http://12factor.net/

Req.s 1/2

- I. Codebase
- **One codebase tracked in revision control, many deploys**
- II. Dependencies
- **Explicitly declare and isolate dependencies**
- III. Config
- **Store config in the environment**
- IV. Backing Services
- **Treat backing services as attached resources**
- V. Build, release, run
- **Strictly separate build and run stages**
- VI. Processes
- **Execute the app as one or more stateless processes**

Req.s 2/2

- VII. Port binding
- **Export services via port binding**
- VIII. Concurrency
- **Scale out via the process model**
- IX. Disposability
- **Maximize robustness with fast startup and graceful shutdown**
- X. Dev/prod parity
- **Keep development, staging, and production as similar as possible**
- XI. Logs
- **Treat logs as event streams**
- XII. Admin processes
- **Run admin/management tasks as one-off processes**

Microservices architecture

- http://martinfowler.com/articles/microservices.html

TelcoGrade Paas – as it is on the markeet today

- FeedHenry – Mbaas
- http://www.feedhenry.com/mobile-application-platform/mbaas/
  - (Twillio)
  - https://www.twilio.com/customers

Summary

- Instead lof a summary…
- … select your own PaaS of choice

- http://www.paasify.it/vendors

Docker előnye – Virtualizáció teljesítménye

- Density = virtuális gép száma / gazdagép