

“Designing with Microservices”

GnuLinuxMeeting

Palermo 11-12 Marzo 2016



Daniele Mondello



Chi sono e cosa faccio



GnuLinuxMeeting



General

- Management
- Team Leader
- Quality Manager
- Web Architect
- Healthcare IT Consultant
- UX Expert



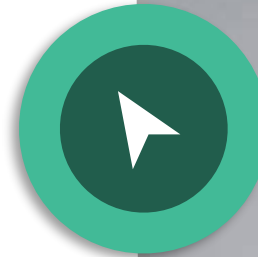
O.S.

- Ubuntu, CentOS
- Mac OSX
- Windows
- IOS - Android
- Embedded OS
- Raspai, Pidora

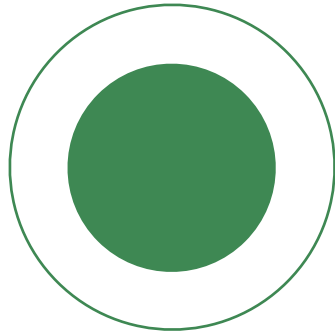


Developer

PHP (6 years)
Javascript (11 years)
ASP(13 years)
MySQL (6 years)
HTML, CSS3 (15 years)
Ruby, Rails (1 year)
Python (1 year)



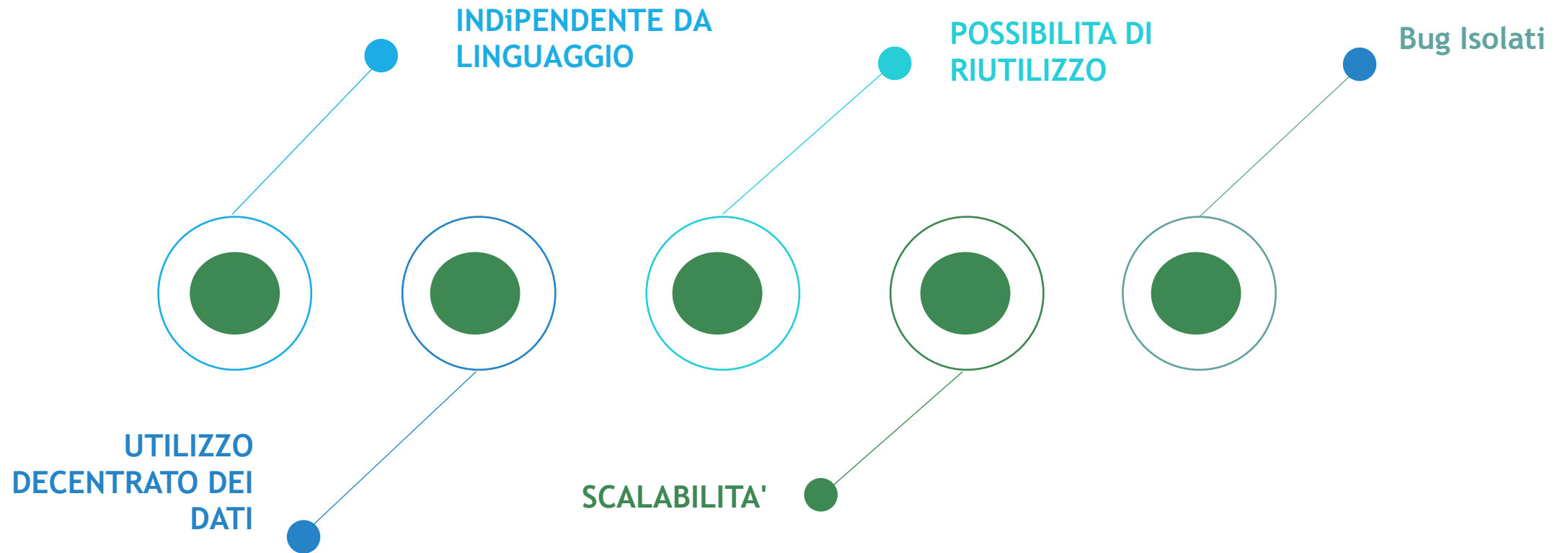
Cosa è un **Microservice**



Definizione di Micror-services

Sono dei servizi “micro” autonomi, che fanno una cosa sola (ma bene!) e che interagiscono tra di loro... in altre parole dei sistemi distribuiti-

Caratteristiche di un **Microservice**



L'obiettivo di un **Microservice**



Micro è l'obiettivo!

*Facci dunque uno principe di
vincere e mantenere lo stato:
e' mezzi
saranno sempre iudicati
onorevoli e da ciascuno laudati*



Architettura per **Microservices**



Microservice A



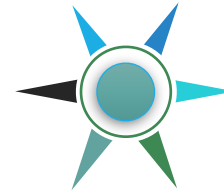
Microservice B



Microservice C



Microservice D



MicroserviceE

Docker



Cosa fa **DOCKER?**

Docker sviluppata da Docker inc di Salomon Hykes è un sistema di virtualizzazione basato su container. Mentre nella virtualizzazione classica sono virtualizzati programmi e kernel del sistema operativo nella virtualizzazione per containers sono virtualizzati software e librerie.



Build



Ship



Run

Architettura per **Microservices**



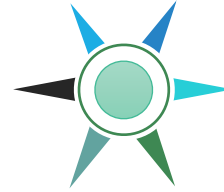
Microservice A



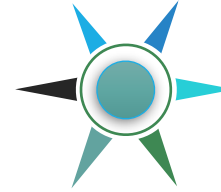
Microservice B



Microservice C



Microservice D



MicroserviceE

Platform Automation

Deployment Automation

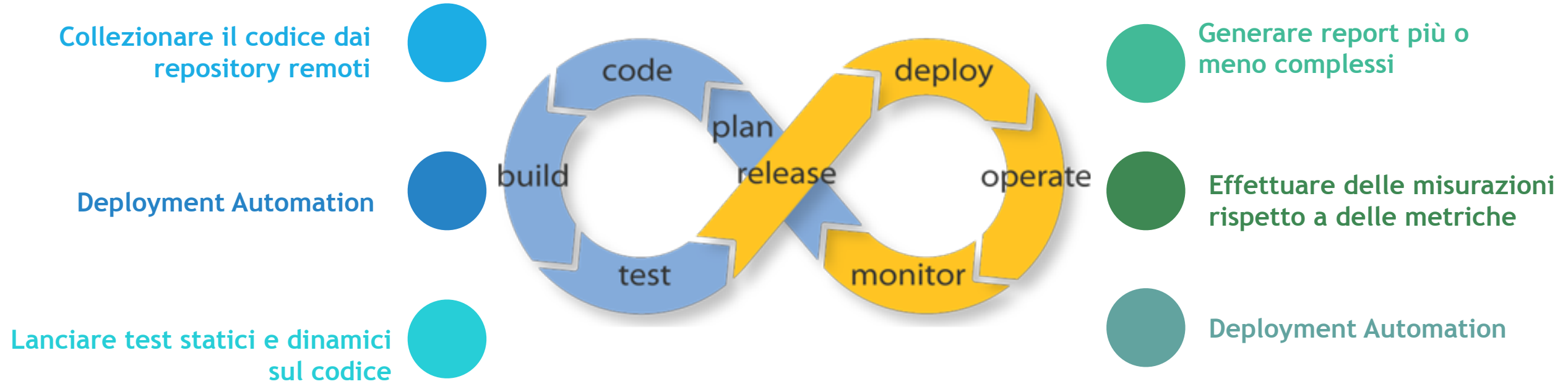
Build & Test Automation

Diagnostic & Instrumentation

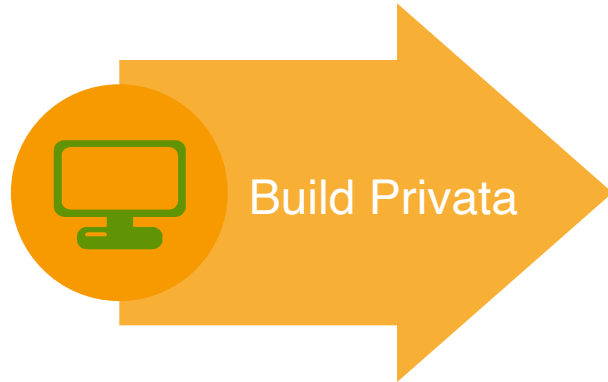
Monitoring & Alerting

AuthN/AuthZ

Continuous Delivery

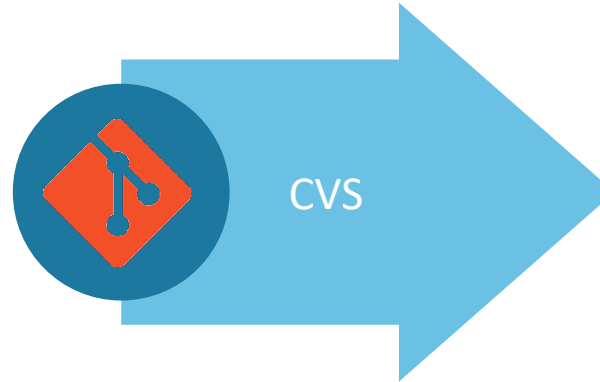


Flusso di Lavoro per un **Microservice**



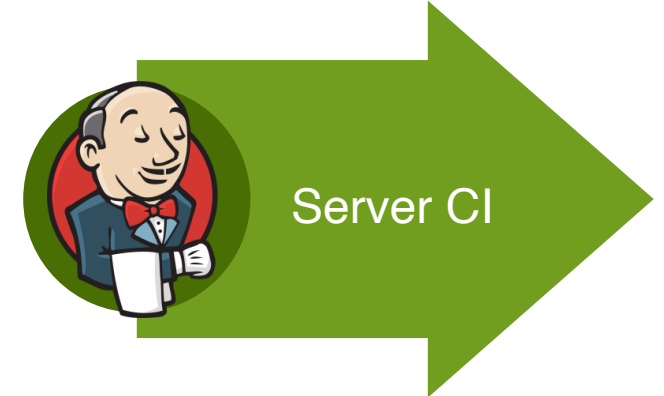
Lancio di una build privata

Lo sviluppatore effettua una Build Privata sulla propria postazione così da integrare eventuali cambiamenti effettuati dagli altri membri del team e verificato che l'integrazione dia un sistema funzionante.



Commit sul CVS del codice

Lo sviluppatore conclusa l'attività effettua un commit sul repository del sistema di controllo di versione. 5



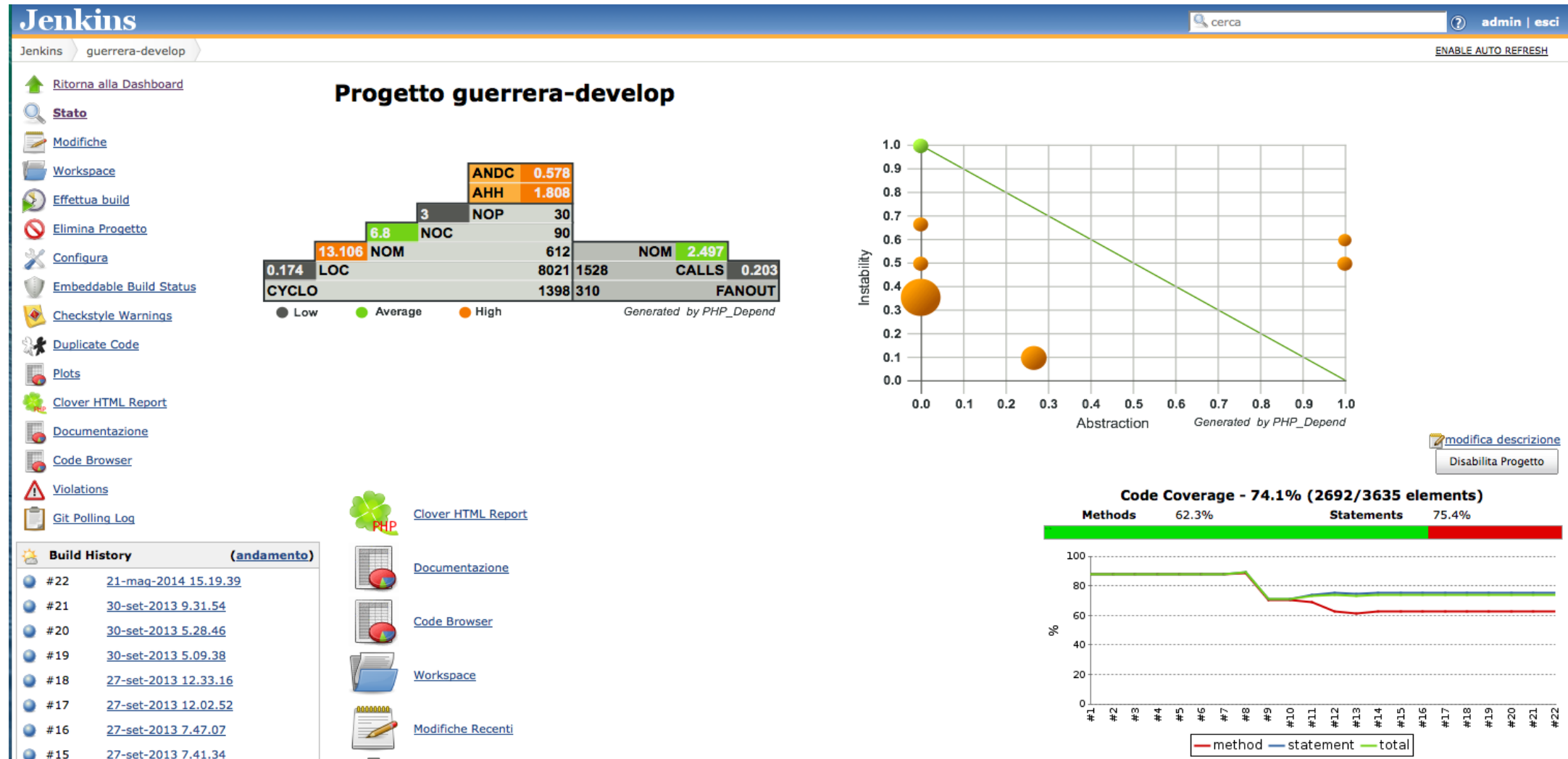
Lancio SCRIPT su server CI

Il server di CI accortosi del cambiamento scarica il codice aggiornato e lancia uno script di build così da integrare questi cambiamenti e da ricostruire il sistema e ritestarlo ed a seguito del risultato della build, genera un feedback (es: email) visibile al team

GnuLinuxMeeting

#	Stato
1	Inattivo
2	Inattivo

Un progetto in Jenkins 1/2



Un progetto in Jenkins 2/2

#15	27-set-2013 7.41.34
#14	26-set-2013 8.26.23
#13	26-set-2013 8.21.06
#12	25-set-2013 12.48.28
#11	31-ago-2013 2.58.42
#10	26-ago-2013 19.44.22
#9	26-ago-2013 17.42.35
#8	11-lug-2013 9.42.02
#7	8-lug-2013 12.41.55
#6	5-lug-2013 17.22.25
#5	5-lug-2013 17.20.55
#4	5-lug-2013 17.19.00
#3	30-giu-2013 18.05.49
#2	28-giu-2013 10.37.33
#1	18-giu-2013 15.53.16

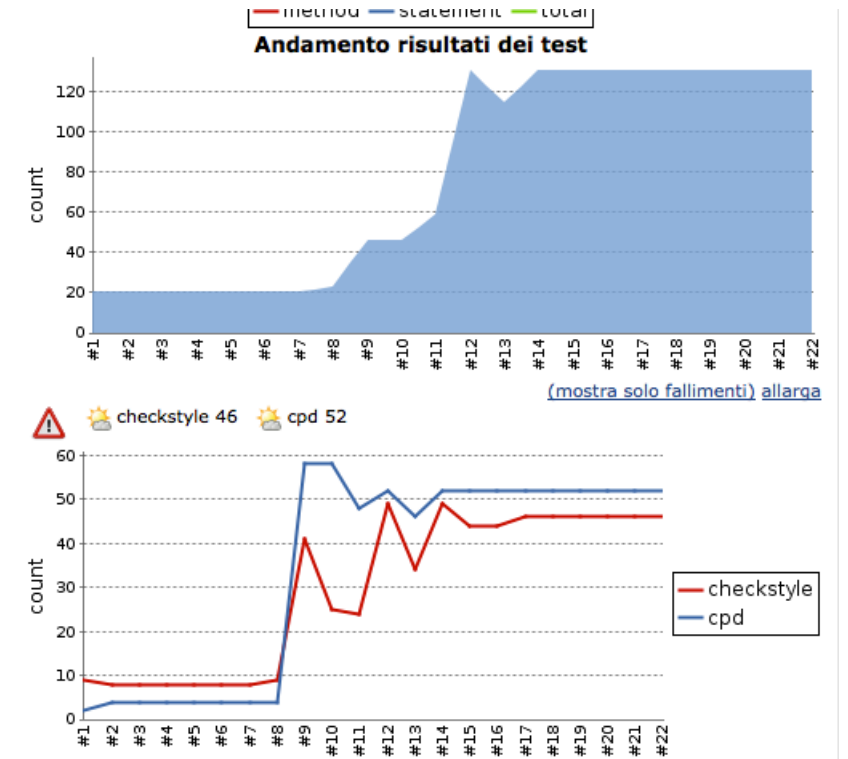
 [RSS per tutto](#)  [RSS per i fallimenti](#)



[Latest Test Result](#) (nessun errore)

Permalink

- [Last build \(#22\), 4 mo 29 days fa](#)
- [Last stable build \(#22\), 4 mo 29 days fa](#)
- [Last successful build \(#22\), 4 mo 29 days fa](#)



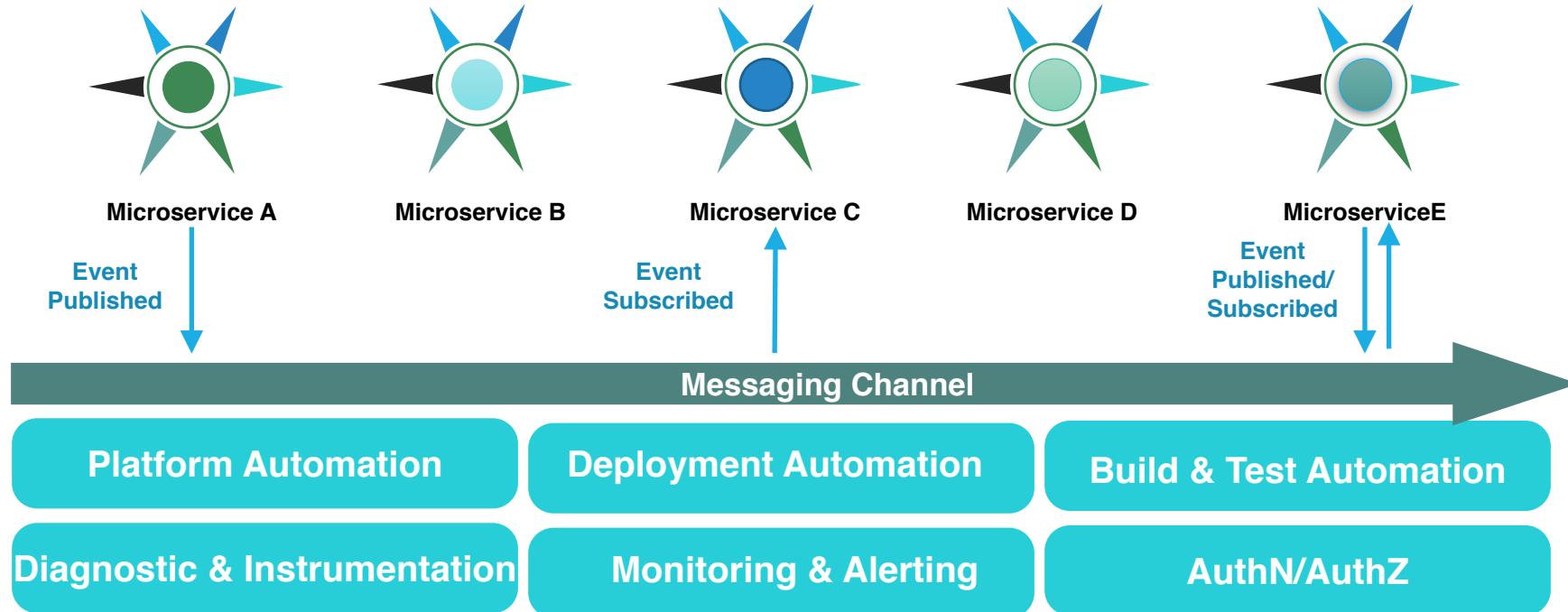
 [Aiutaci a tradurre questa pagina](#)

Pagina generata il: 18-ott-2014 2.27.15

[REST API](#)

[Jenkins ver. 1.505](#)

Architettura per **Microservices**



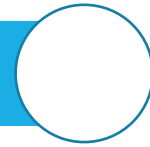
Messaging Channel



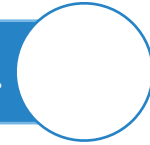
Robusta soluzione per messaggistica tra applicazioni con supporto a diverse piattaforme di sviluppo



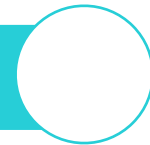
Easy to Use!



Runs on all major O.S.



Open Source



Utilizzo Rabbit MQ

1 "Hello World!"

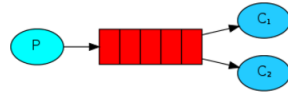
The simplest thing that does *something*



- › Python
- › Java
- › Ruby
- › PHP
- › C#
- › Javascript
- › Go
- › Elixir

2 Work queues

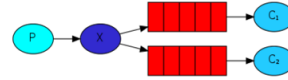
Distributing tasks among workers



- › Python
- › Java
- › Ruby
- › PHP
- › C#
- › Javascript
- › Go
- › Elixir

3 Publish/Subscribe

Sending messages to many consumers at once



- › Python
- › Java
- › Ruby
- › PHP
- › C#
- › Javascript
- › Go
- › Elixir

4 Routing

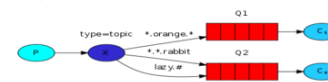
Receiving messages selectively



- › Python
- › Java
- › Ruby
- › PHP
- › C#
- › Javascript
- › Go
- › Elixir

5 Topics

Receiving messages based on a pattern



- › Python
- › Java
- › Ruby
- › PHP
- › C#
- › Javascript
- › Go
- › Elixir

6 RPC

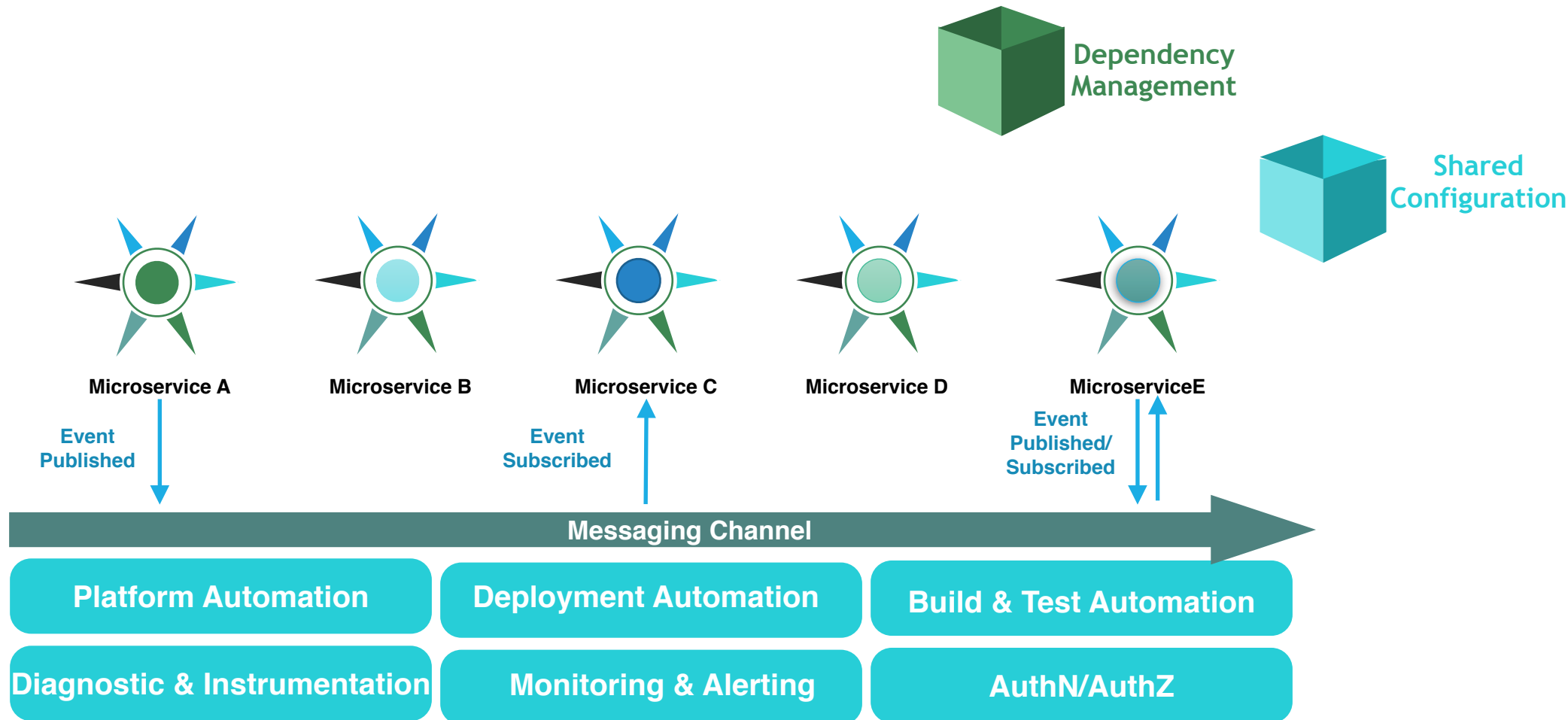
Remote procedure call implementation



- › Python
- › Java
- › Ruby
- › PHP
- › C#
- › Javascript
- › Go
- › Elixir



Architettura per **Microservices**



Dipendenze e Configurazioni



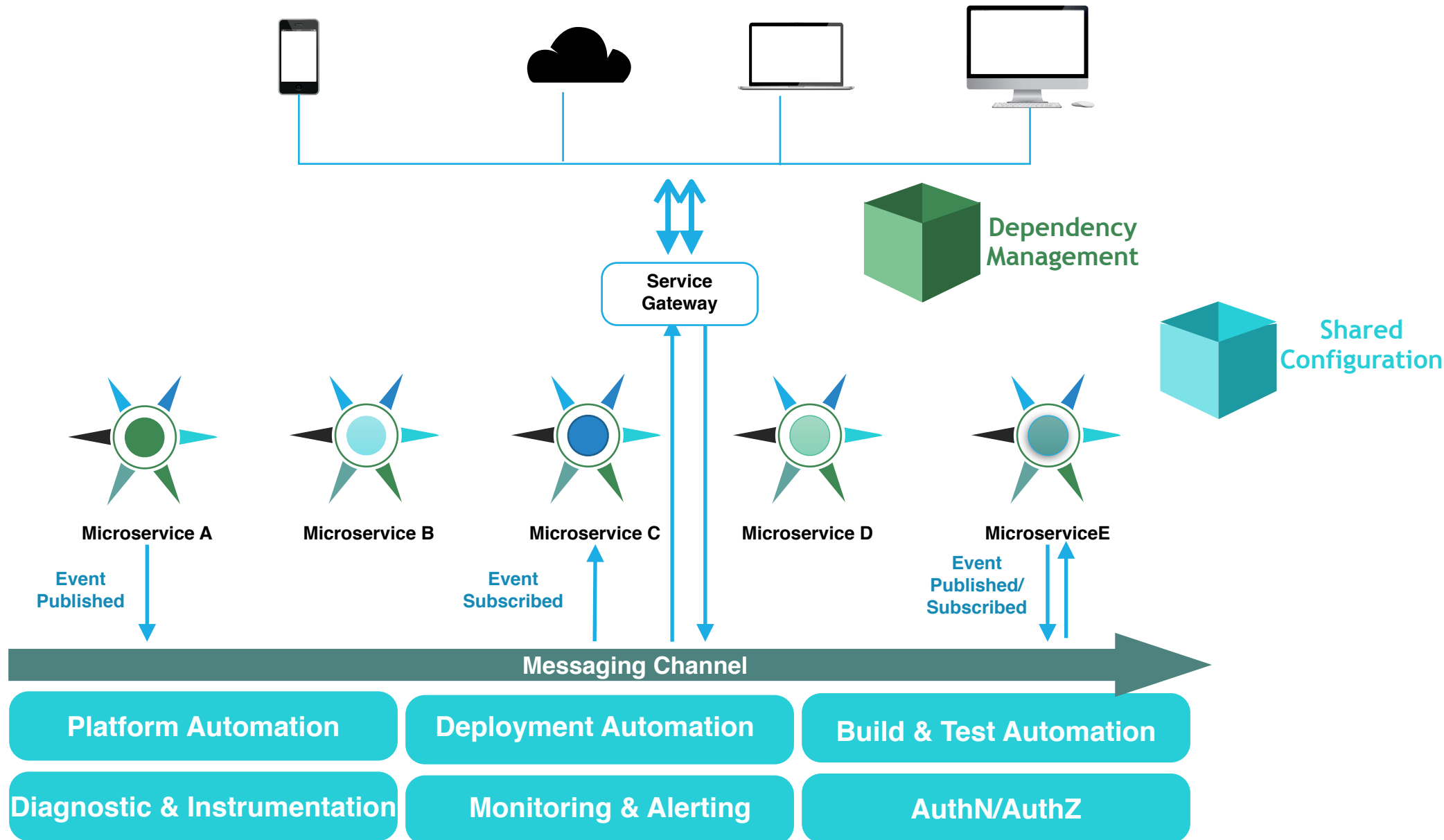
Configurazioni



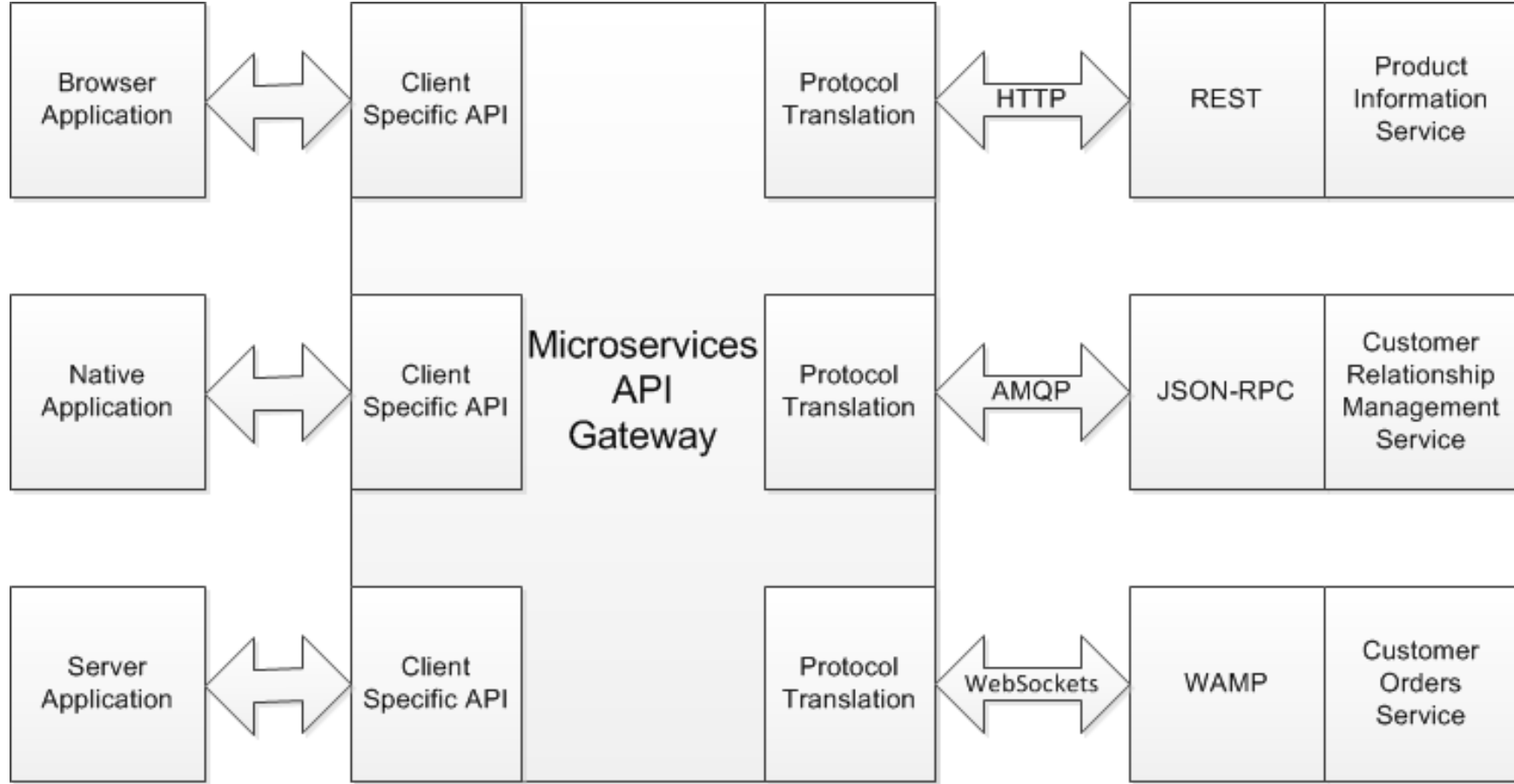
VS



Architettura per **Microservices**

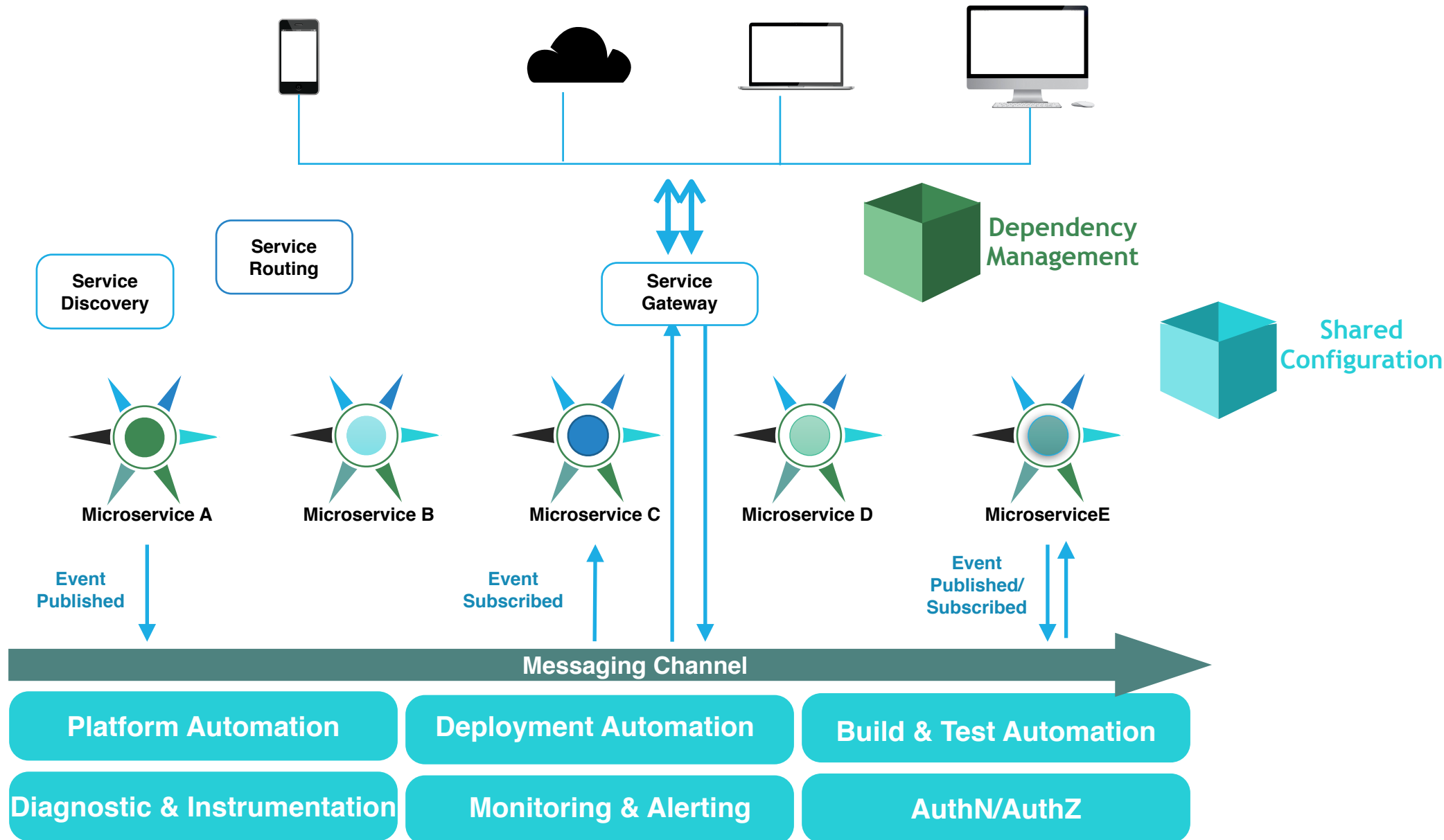


Service Gateway

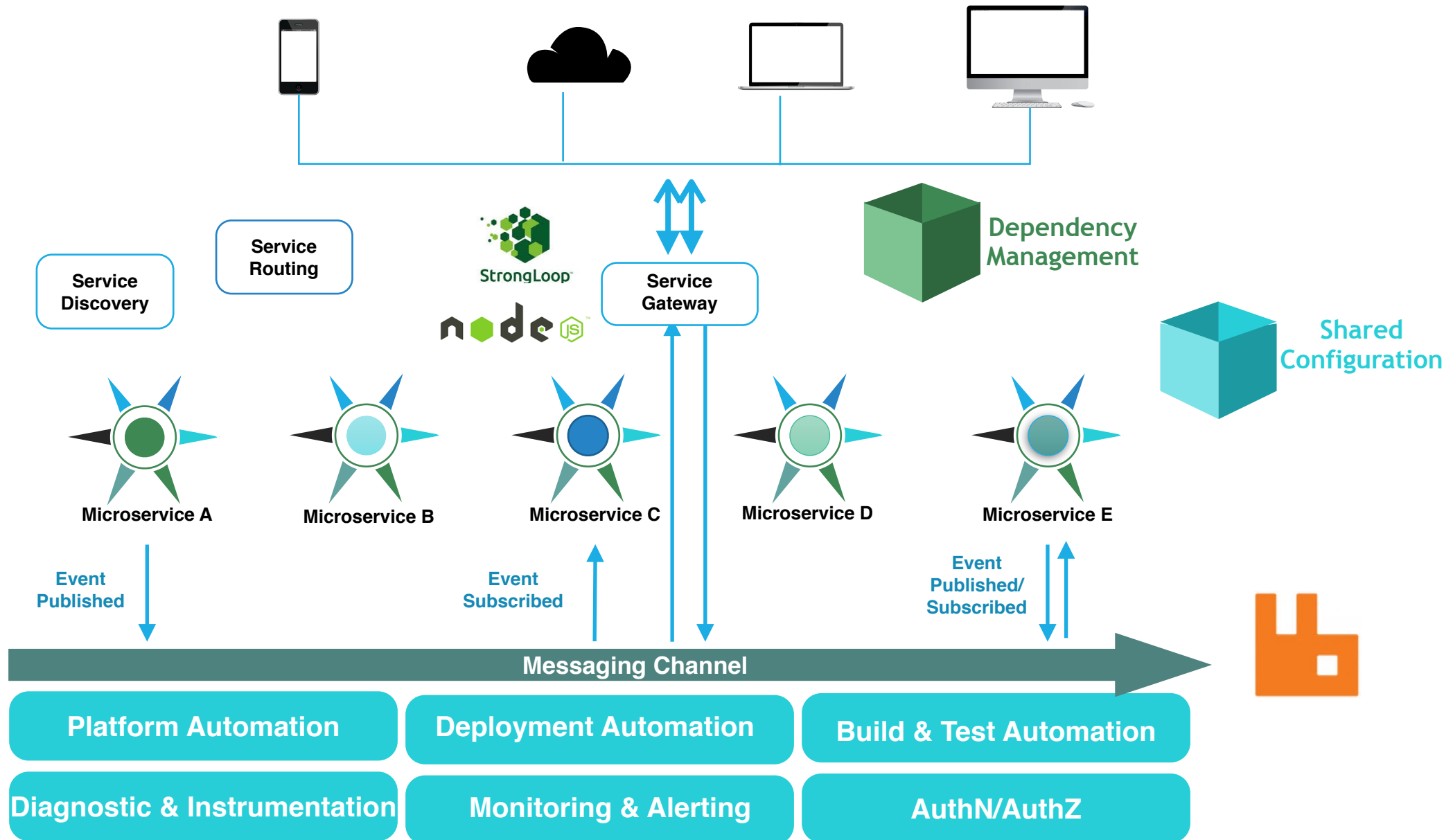


L'API Gateway incapsula l'architettura interna dei microservicese fornisce delle API a misura di client. Può avere anche responsabilità come authentication, monitoring, load balancing, caching, request shaping, static response handling (StrongLoop,NodeJS)

Architettura per **Microservices**



Architettura per **Microservices**





Daniele Mondello

GRAZIE PER L'ATTENZIONE



info@danielemondello.it



Danielemondello



@dmondello

