Google Cloud Platform (GCP) is gelanceerd in April 2008. Het draaide des tijds in dezelfde datacenters als ‘google search’, ‘youtube’ en ‘gmail’. Het was slechts in 2011 dat GCP beschikbaar was voor het brede publiek. GCP is een onderdeel van de Google Cloud. Google Cloud biedt een enorme serie van producten aan waarvan GCP maar een klein onderdeel is. GCP specifiek, biedt ‘infrastructure as a service’ (IaaS), ‘platform as a service’ (PaaS) en ‘serverless computing’ aan.

Omdat het doel van dit onderzoek specifiek de support voor CI/CD pijpleidingen vergelijken is, bekijken we een specifiek onderdeel van GCP. De ‘Cloud Developper Tools’. Onder deze categorie vallen er een aantal zeer interessante tools. Hier vindt men onder andere ‘Cloud Build’, ‘Cloud-SDK’, ‘Tools voor Powershell’, ‘Tools voor Visual Studio’, enz.

Cloud Build is Google zijn antwoord op een volledige geautomatiseerde CI/CD pijpleiding. Het is dan ook volledig mede met de moderne vereisten. Met Google Cloud Build (GCB) is een organisatie instaat om snel en gemakkelijk een volledige pijpleiding te configureren. Het gelijkt dan ook op Azure DeVops.

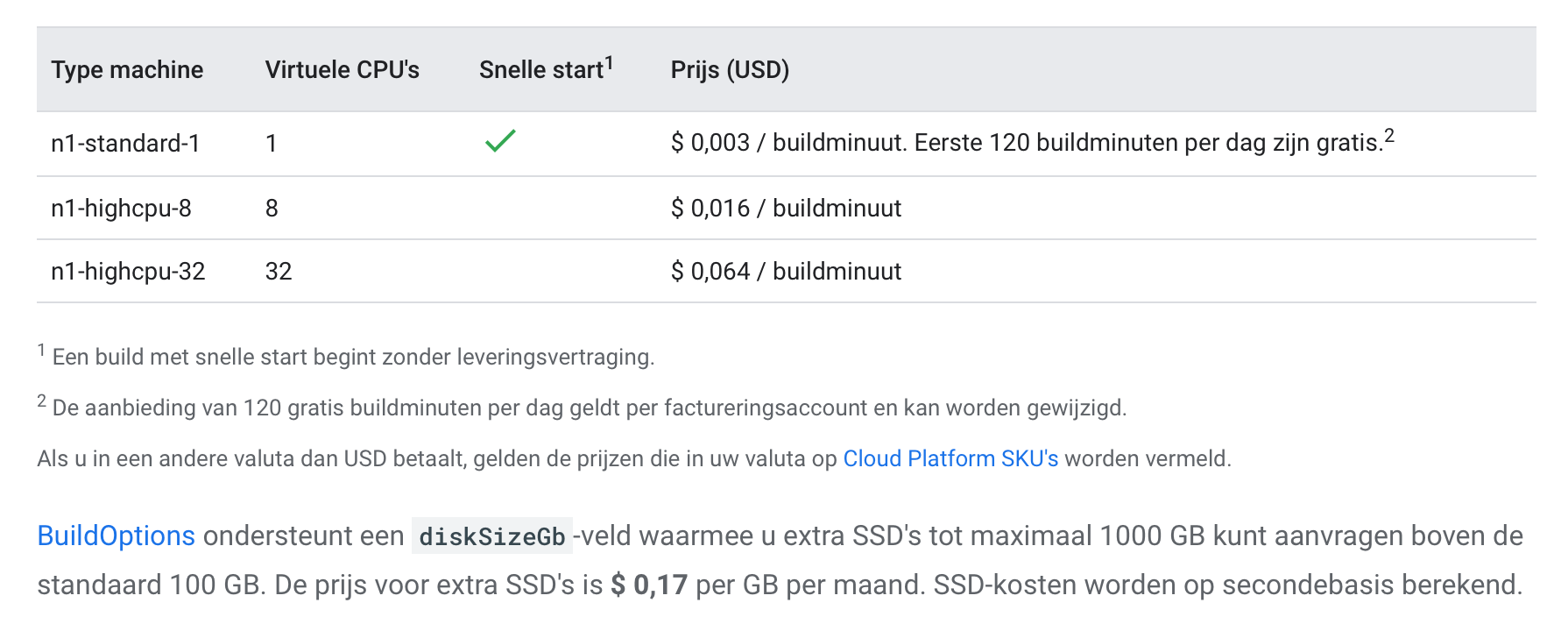
GCB werkt hoofdzakelijk met Git en GitHub om een pijpleiding te bouwen. Google heeft ook zijn eigen ‘Cloud repository service’ die naadloos integreert met GCB, maar deze is helaas betalend. Daarom is het gebruik van Git met GitHub een beter en goedkoper alternatief. Aangezien deze ook perfect integreren met GCB en omdat Git wijdverspreid en simpel in gebruik is. Een organisatie kan dan configureren op GCB dat bij het moment van een code update op GitHub, automatisch een compileer pijpleiding wordt gestart. Om GCB te laten weten wat er specifiek moet uitgevoerd worden, moet er op de GitHub repository een YML-file voorzien worden waarin regels gedefinieerd moeten worden. Dit maakt het gemakkelijk om snel aanpassingen te maken.

De pijpleiding op GCB werkt op basis van Docker images. Deze worden in de cloudbuild.yml gedefinieerd. Ook wordt er per Docker container gedefinieerd wat er moet uitgevoerd worden in de vorm van commando’s, script, enz. Dit maakt het mogelijk dat iedere stap in het CI gedeelte van de pijpleiding volledig aangepast kan worden naar de noden van de organisatie. Zo kan de organisatie beslissen om voor gemaakte containers te gebruiken van de ‘DockerHub’ pagina. Ook kan de organisatie zelf container maken met aangepaste scripts om bijvoorbeeld in lokale omgevingen testen uit te voeren. GCB kan dus in een hybride opstelling geïmplementeerd worden. Wat ook de bedoeling zou zijn aangezien dit een use case van Aucxis is. Daarnaast biedt GCB ook de mogelijkheid om de gecompileerde code rechtstreeks vanuit Google Cloud beschikbaar te maken voor verdere verdeling.

Met behulp van deze containers kunnen dan functionele testen uitgevoerd worden op de gecompileerde code. Het programma kan dan op basis van de uitkomst van deze uitgevoerde taak, naar de volgende stap zijn wachtrij worden geplaatst. Hier kan dan de volgende taak starten. GCB genereert rapporten en statistieken van de uitgevoerde taken zodanig dat de gebruiker van het platform inzicht kan krijgen in de uitgevoerde taken.

De compilatie en uitvoer-tijden van GCB zijn zeer goed aangezien het platform automatisch schaalt naarmate er meer rekwesten tegelijk verstuurd worden. Dit maakt mogelijk dat verschillende programmeurs tegelijk aan hetzelfde project werken of aan meerdere projecten tegelijk. Ook biedt GCB de mogelijkheid om redundantie te voorzien. GCB maakt het mogelijk om naar andere Cloudplatformen uit te rollen of zelfs om de werklast te verdelen over verschillende Cloud platformen. Dit is mogelijk door gebruik te maken van Tekton. Tekton is een open-source framework voor Kubernetes. Dit maakt het mogelijk dat een organisatie over verschillende Cloud platformen heen kan werken. Kubernetes is een clustering hypervisor voor Docker containers. Tekton zou een goede oplossing kunnen zijn voor CI/CD pijpleidingen in de Cloud maar valt hier buiten beschouwen vermits het doel de verschillende aanbiedingen van de Cloud platformen vergelijken is.

De prijzen in Google Cloud worden berekend zoals bij ieder moderne Cloud aanbieder. De gebruiker betaald wat hij verbruikt. De volgende tabel moet dienen om een beeld te vormen over wat men kan verwachten te betalen voor het gebruik van GCB. Dit zonder netwerk kosten voor het transfereren van gegevens. Er wordt ook niks in rekening gebracht voor zaken die in een wachtrij staan of pijpleidingen die niet gebruikt worden. De Google Cloud Developper Tools hebben een voordeel dat een groot deel van de tools voor ondersteuning met het platform gratis zijn. Er zijn ook weinig tools van derden nodig om de gewilde functionaliteit te bereiken.



Met andere woorden is GCB dus een volwaardig alternatief voor Azure DeVops. Er is ondersteuning voor verschillende tools, biedt uitgebreide functionaliteiten aan en bovendien adverteert Google dat het gemakkelijk samenwerkt met andere Cloudplatformen. Het zei door het gebruik van Tekton. Daarnaast is het ook relatief simpel in gebruik door dat GCB gebruikmaakt van Docker containers voor te compileren.