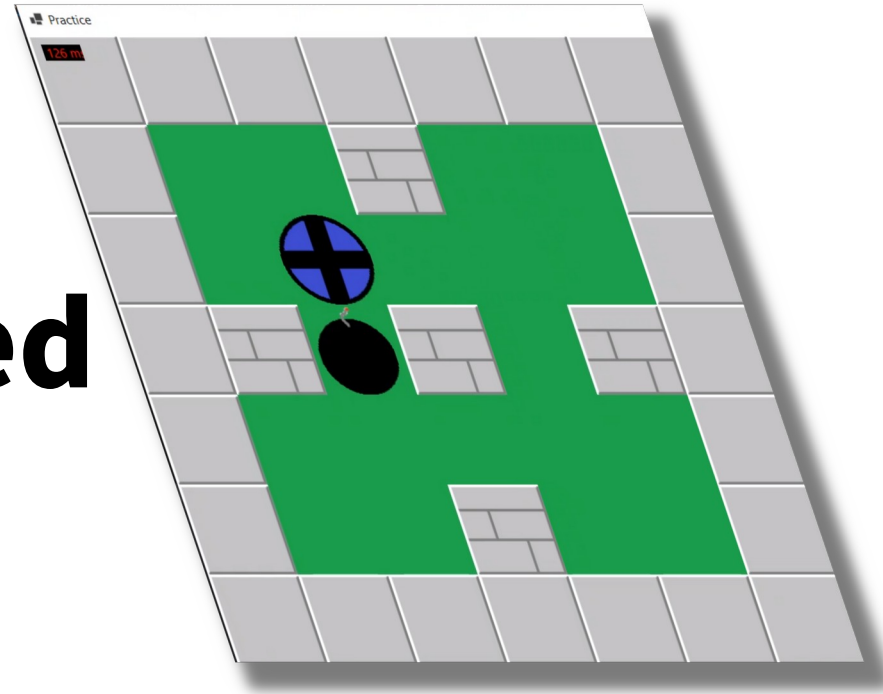


Serverless Bomberman: RTMPG PoC based on Durable Functions

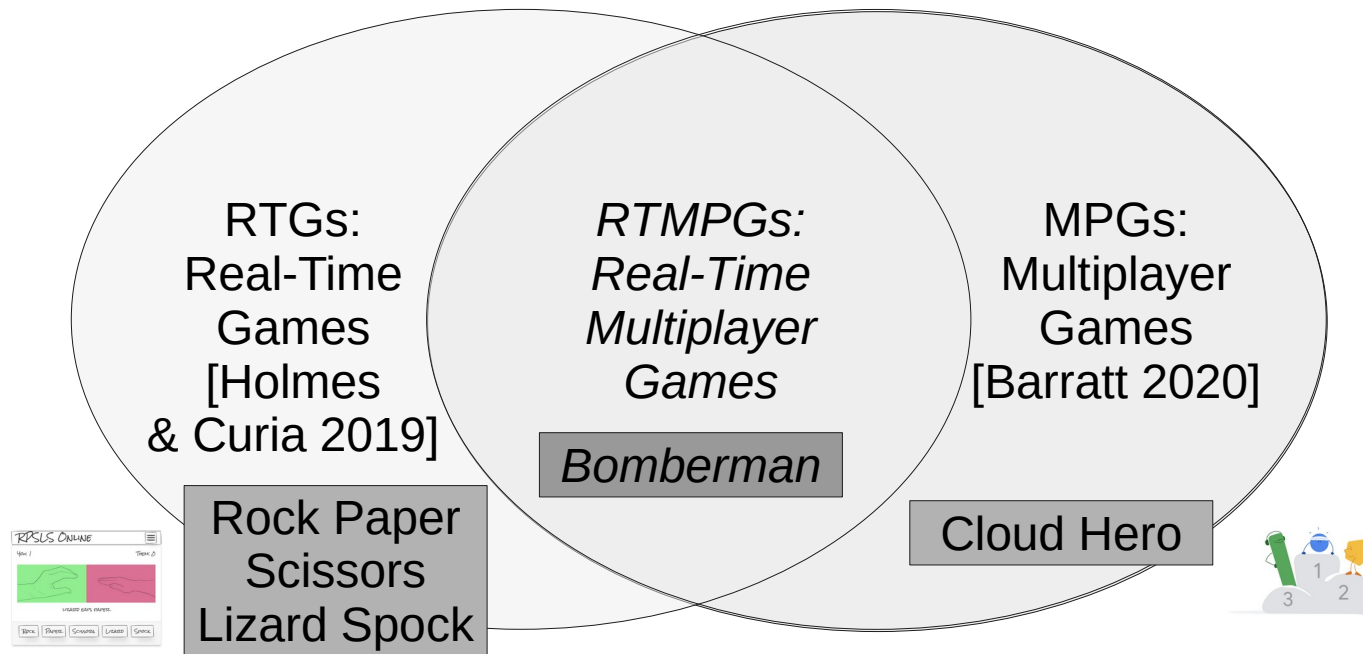


**Evan Hirschi¹, Rico Nachbur¹,
Josef Spillner¹, Jesse Donkervliet²**

2022 | WOSCx | Online

RTMPGs

A new frontier for serverless application design...



* Barratt, J., 2020. Building a Multiplayer Game with API Gateway+Websockets, Go and DynamoDB. [Online]
Available at: <https://serialized.net/2020/09/multiplayer/>

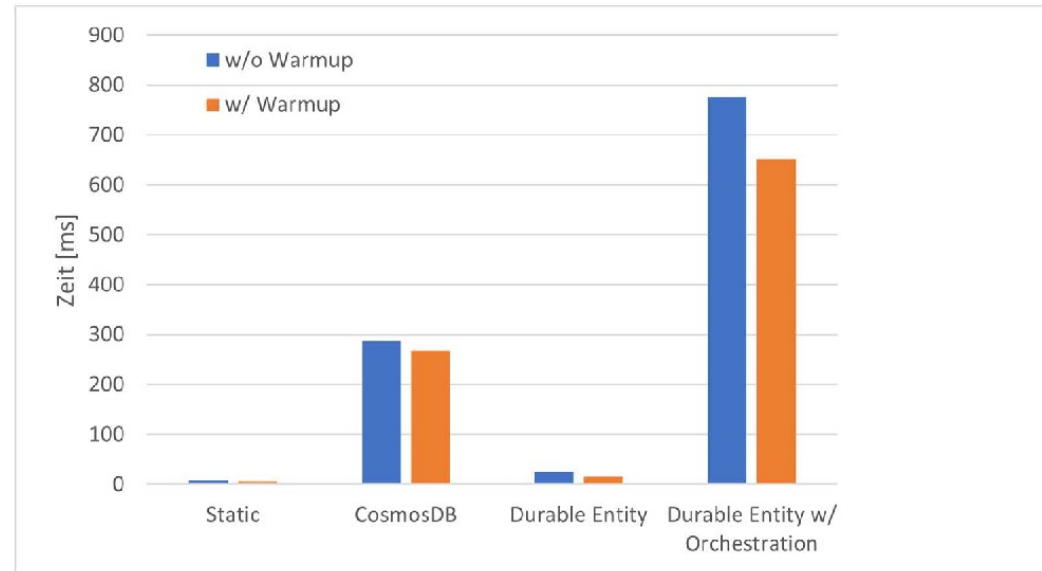
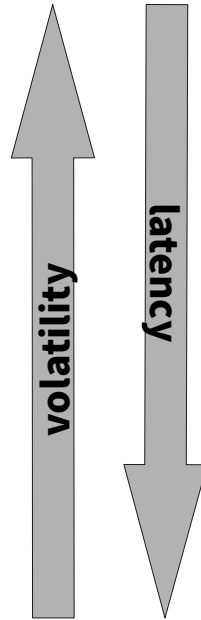
Holmes, P. & Curia, N., 2019. Building a serverless online game: Cloud Hero on Google Cloud Platform. [Online]
Available at: <https://cloud.google.com/blog/products/application-development/building-a-serverless-online-game-cloud-hero-on-google-cloud-platform>

Statelessness of Functions

**In memory
(static class)**

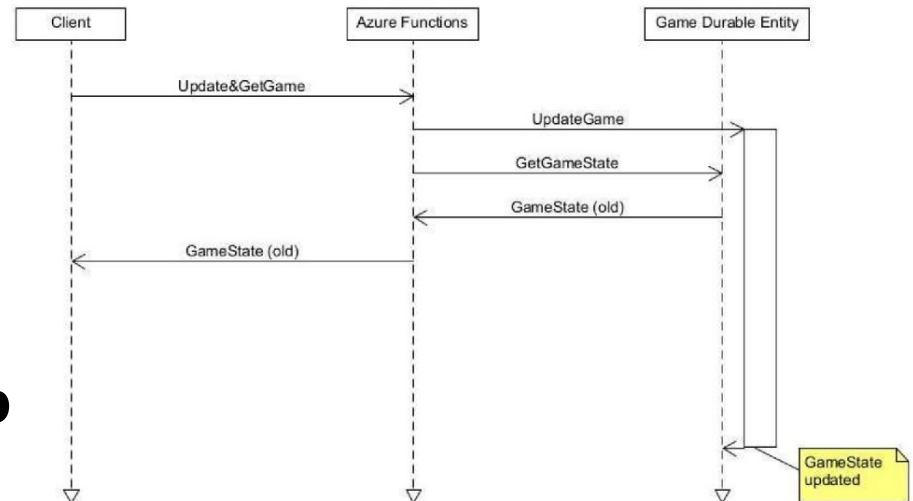
**On local disk
(probabilistic)**

**In backend
(BaaS: DB,
orchestration)**



Azure durable entities:

- **Cost-effective**
(almost no extra pricing)
- **Appropriate size limits**
(64 kB → Tables, else → Blob)
- **Eventual consistency**
(polling required)

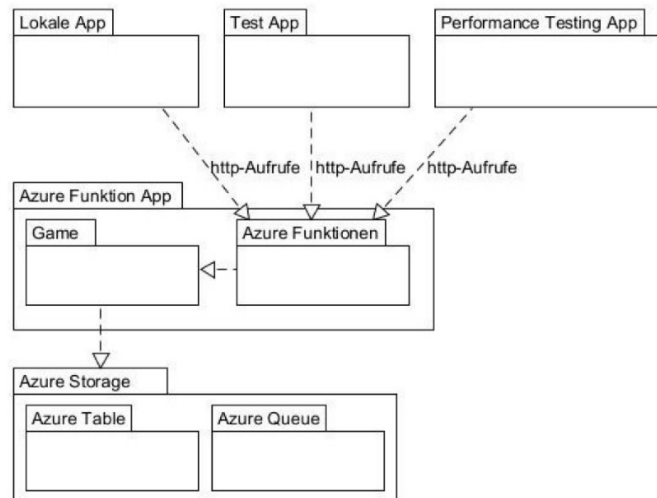


Architecture

Functions:

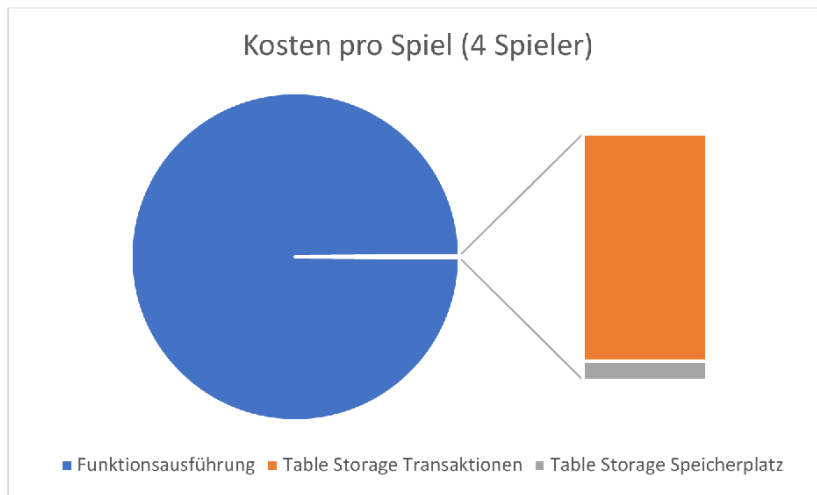
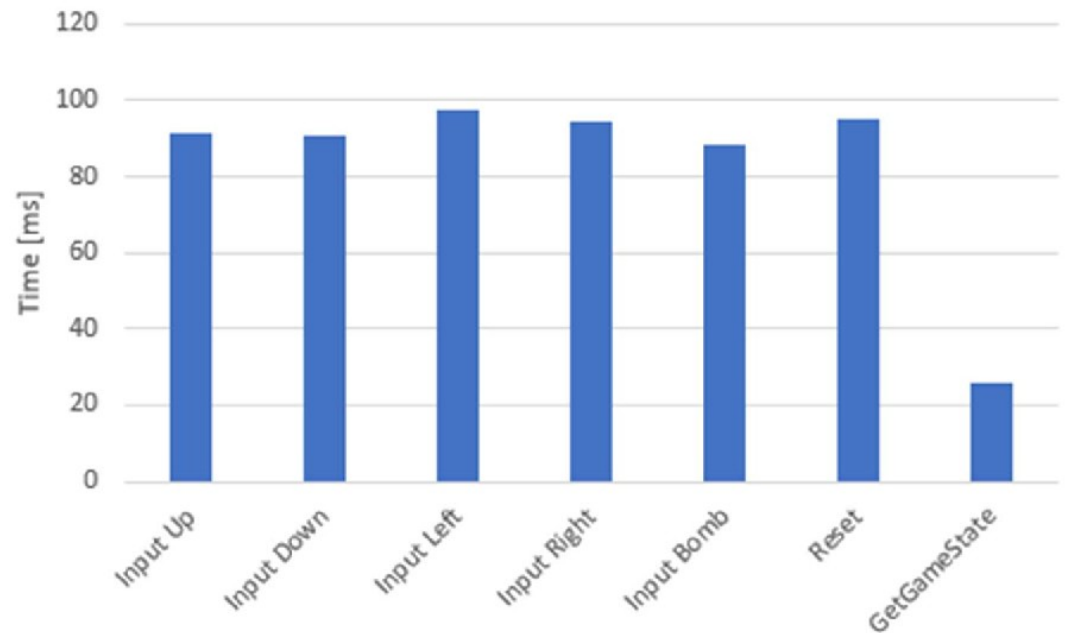
GameInputFunction	api/input/{gameKey}	← Input
GameResetFunction	api/reset/{gameKey}	
RemovePlayerFunction	api/remove/{gameKey}	
GetGameStateFunction	api/getgamestate/{gameKey}	→ Game

Game function (entity-triggered)



Results

**Consistently < 100ms
per function response
→ RT feasible**



**3 minute game:
GameInputFunction 1x/s
GetGameStateFunction 100x/s
= 18180 events
= 0.0036 US\$ per player**

**Storage 3 kB (field-dependent)
= 180 transactions
= 0.0000024 US\$ per player**

Optimisation: delta transfers