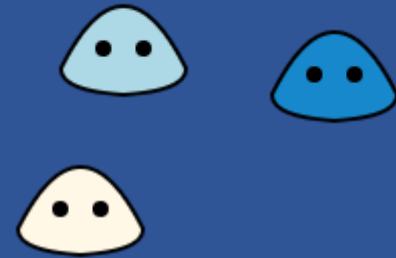


BUILDING AN ONLINE MULTIPLAYER GAME ON SERVICE FABRIC

Antonio Menarde



ABOUT ME

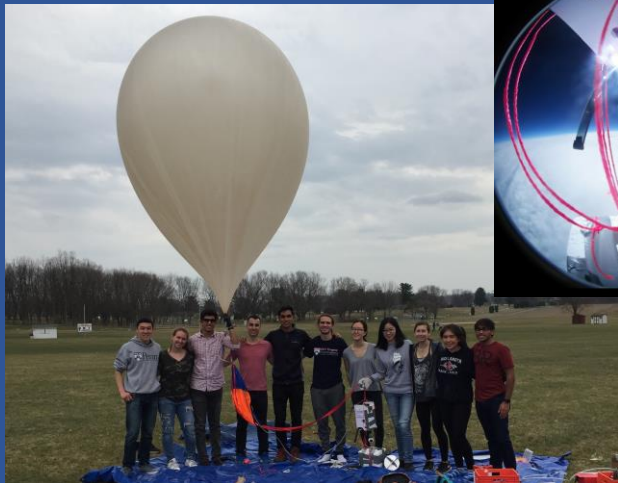
University of Pennsylvania

BSE in Systems Engineering

BS in Business – Decision Processes



Penn Aerospace Club



- Photography
- Cinema
- Furniture Design
- Woodblock Printing
- Bikes
- Coffee
- Arduino / RP
- Volunteering (BBBS)
- Hiking



WHAT I DIDN'T KNOW WHEN I STARTED

PM ROLE

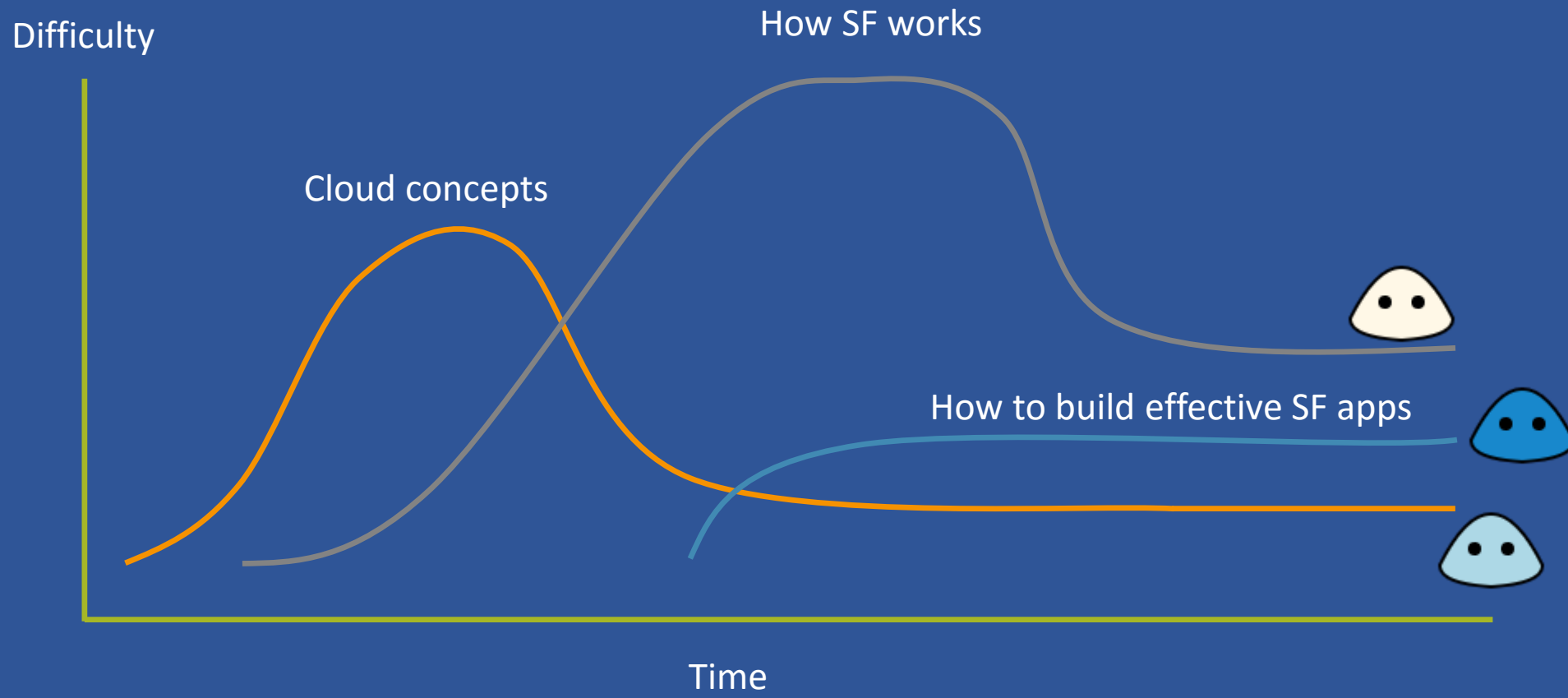
- How to write a spec
- Choose features from the sea of options
- Develop KPIs over those features
- Service Fabric in context
- Azure in context
- What its like to work at Microsoft

DEV ROLE

- Locks
- Checkpoints
- Working through a code review
- Asynchronous processes
- Actors
- C#
- JavaScript
- .NET
- Distributed System Concepts



LEARNING SERVICE FABRIC



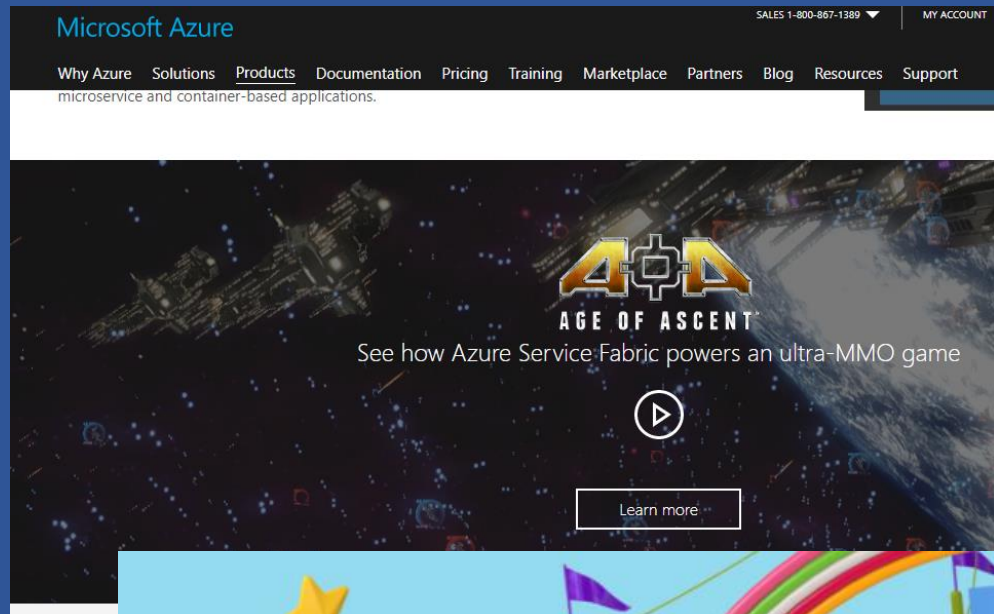


WHAT I MADE

[AKA.MS/WORKBUD](https://aka.ms/workbud)

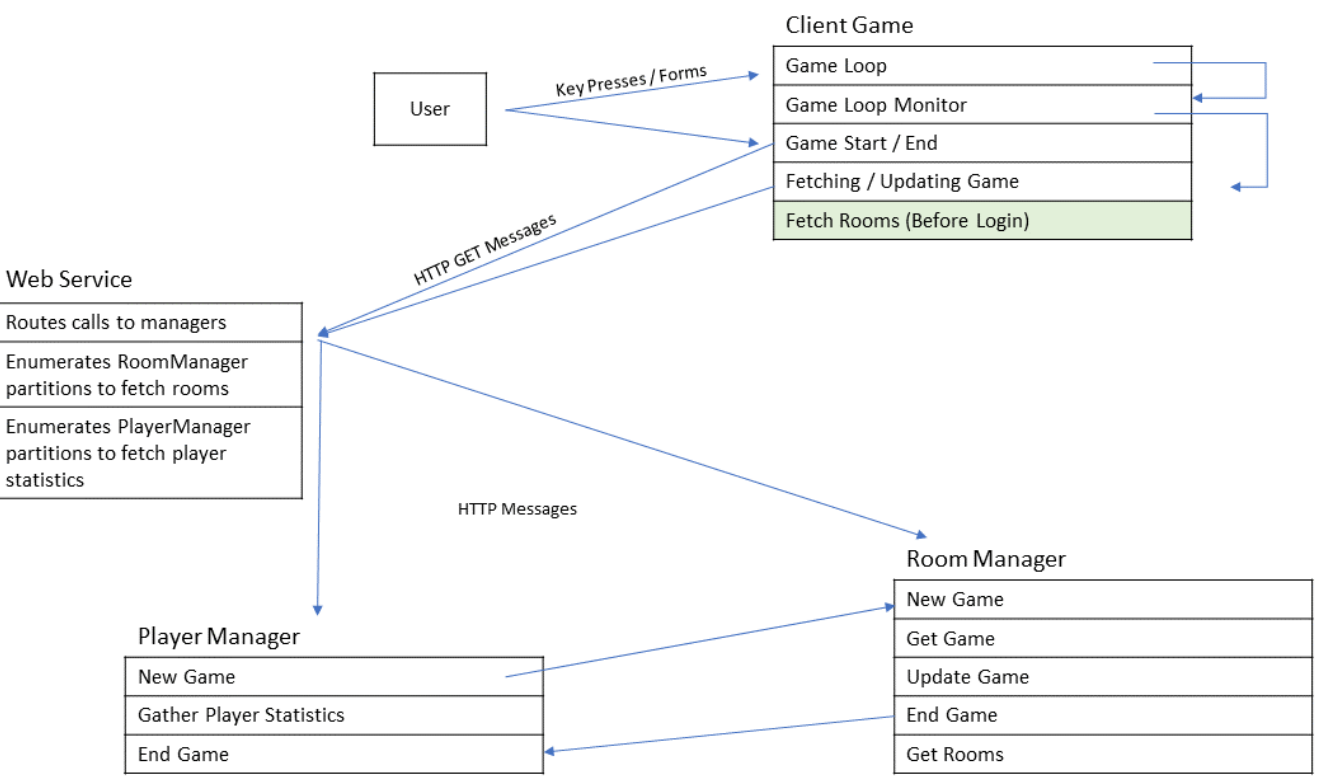


WHY?



Tencent Games and Microsoft
build the cloud game solution

HOW?



Player-Routed Calls

Action	User	Web Client	Web Service	Player Manager	Room Manager
New Game	Enters username into form and chooses a room or creates a room	Parses entry and sends new game HTTP request to webservice	Routes to correct player manager partition by player id	Finds or creates player data, sends that data to correct room manager partition by room id	Finds or creates a new room dictionary, puts player data in room, returns success code
Get Player Statistics	Requests player data	Sends request	Asks each partition in player manager to send their statistics, concatenates them	Iterates through player dictionary to gather statistics for partition	

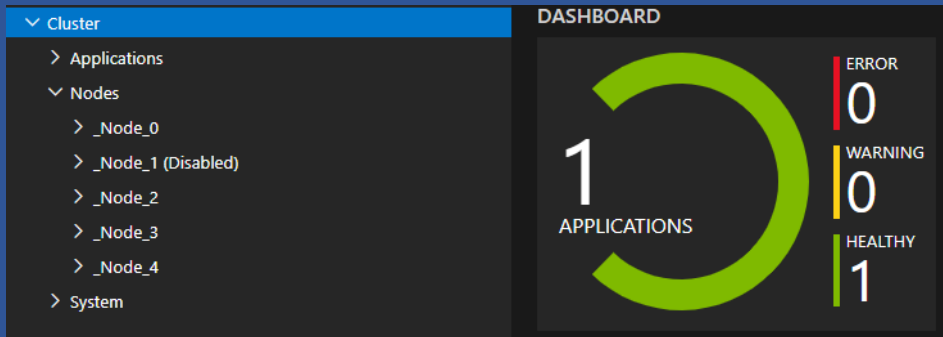
Room-Routed Calls

Action	User	Web Client	Web Service	Room Manager	Player Manager
Get Rooms	Loads webpage	Sends request	Asks each partition in room manager to send their room info	Iterates through room data to gather room information	
Get Game		Sends fetch request at regular intervals during game	Routes to correct room manager partition by room id	Finds correct room dictionary, iterates through room dictionary and returns game state	
Update Game	Updates a form or does keypress	Updates client game, a monitor finds this change and sends an update game request	Routes to correct room manager partition by room id	Finds correct room dictionary and	
End Game	Presses logout or closes window	Sends request	Routes to correct room manager partition by room id	Sends player data to correct player manager partition by player id / then deletes data	Updates the player data and marks them as logged out, returns success code



BEST MOMENTS OF THE SUMMER

WORK



Killing a node the first time I deployed app to the cloud and having no downtime in-game



PERSONAL



Waking up on a beach in San Juan Island





AZURE-SAMPLES
AKA.MS / WORKBUD



THANKS

