123 S 9th Street
Lincoln, NE 68508

December 9, 2022

Stephen Wolfram, CEO
101 Wolf Way
Los Angeles, CA 30649

Dear Mr. Wolfram:

I am writing to introduce the recommendation report you requested.

This recommendation report recommends the most effective network topologies for your first-person shooter multiplayer video game, Wolf. First, I gathered knowledge of network topologies in video games. Then, I analyzed and evaluated these network topologies against rigorous criteria. Lastly, I recommended two of them to be used for your game, a dedicated game server and local/couch multiplayer.

I found that the best network topologies for multiplayer shooter games have low latency since players need to be able to react quickly to shoot at enemies. In addition, the best online network topologies have good security and can reach players worldwide.

I hope this report solves your problem. Please reach out if you require any more assistance.

Best wishes,

Kalen Wallin

Kalen Wallin
Wallin Labs

Attachment: Most Effective Network Topologies for a Shooter Multiplayer Game Recommendation Report

Most Effective Network Topologies for a Shooter Multiplayer Game

A Recommendation Report

by

Kalen Wallin

JGEN 200-006

Computer Science

College of Arts and Sciences

December 9, 2022

Stephen Wolfram, CEO

Wolfram Gaming

stephen.wolfram@wolf.com

Table of Contents

List of Tables

1 Executive Summary

This report recommends the most effective network topologies for Stephen Wolfram's game development company, Wolfram Gaming. Wolfram Gaming would like to develop a first-person shooter multiplayer video game called Wolf. The recommendation helps Wolfram decide how to connect players over a network for its multiplayer.

During my research, I found that some popular multiplayer games include a combination of network topologies, typically involving one or two offline and one online. After research, I analyzed the network topologies local/couch multiplayer, Local Area Network, Peer-to-Peer, listen server, and dedicated game server. These network topologies were evaluated according to latency, scale, security, cost, complexity, and reach. I found that the best network topologies for multiplayer shooter games have low latency since players need to be able to react quickly to shoot at enemies. In addition, the best online network topologies have good security and can reach players worldwide.

Thus, I recommend a dedicated game server for online multiplayer and local/couch for offline multiplayer. A Local Area Network multiplayer could also be used in specific use cases like eSports competitions.

2 Introduction

This report will recommend the most effective network topologies for a multiplayer shooter video game. First, the problem will be introduced along with its significance. Second, the objectives and proposed problem solutions will be stated. Third, the scope will be defined. Fourth, the background that motivated this report will be described. Lastly, the research will be described and analyzed.

3 Defining the problem

As a game developer, I know firsthand how difficult it is to build a multiplayer feature for a game. From my experience, the process of developing a multiplayer feature typically involves researching, designing, implementing, testing, and releasing. During the research and design phases, the developers must decide what gameplay they will provide their players, how that gameplay will be implemented, how players will connect through a network, and which services they will use or need to create themselves.

The current research is difficult to go through and can take up a lot of precious development time. By solving the problem of how players will connect through a network, amateur game developers will not need to do this research themselves, which will increase their development time. Since shooter video games were the top video game genre globally in 2021 (Clement, 2022), an influx of new developers will likely come into the genre and find this report useful.

3.1 Other work

Unity Technologies has done a tremendous job of gathering primary research to make the research on this topic easier to understand (Unity Technologies, 2022). They have identified the complexity problem in multiplayer game development, especially for new game developers, and worked to reduce it. Unreal Engine also has great documentation on multiplayer game development specifically for their game engine (Unreal Engine, n.d.). However, reading this documentation can improve one's knowledge of multiplayer development regardless of the game engine they are developing on.

4 Objectives

The first objective is to research network topologies and their ability to connect players for multiplayer shooter video games. The second objective is to analyze and evaluate these network topologies

according to criteria. The third and last objective is to recommend the most effective one of these network topologies for use in a multiplayer shooter video game.

4.1 Scope and Beyond the Scope

The scope is to look at gaming network topologies and how they can be useful for multiplayer shooter games. Beyond the scope is looking at any video game genre that is not a shooter, such as a sandbox, role-playing, sports, puzzles, etc. (Pavlovic, 2020). As well as the other network topologies not used in gaming, like ring, tree, and hybrid (Williams, 2022). Also beyond the scope is evaluating all other steps of the multiplayer development process described in Defining the problem. Perhaps those could be researched and solved in a series of future reports.

5 Background

5.1 A brief overview of background

We will identify the stakeholders, clients, and industry, identify the problem for the stakeholders, provide an overview of the industry, and lastly, share the credibility and research that meet the client's needs.

5.2 Stakeholders, Clients, Industry

The person requesting this report is Stephen Wolfram, Founder and CEO of Wolfram Research (Wolfram, n.d.). He is starting a new company called Wolfram Gaming to build multiplayer video games for gamers worldwide. The first game he would like to create is a first-person shooter called Wolf, and he would like to create everything himself. Only when he completely understands game development will he hire developers to continue his game development. Due to this stubbornness, Wolfram Gaming only has one employee, and that is Wolfram himself.

5.3 Problem for Stakeholder

Despite Wolfram's brilliance (Ferris, 2016), he's been having trouble wrapping his head around the concepts of multiplayer networking and needs help deciding which network topology to use in his first-person shooter game. Having realized this difficulty, Wolfram reached out to me, wanting to learn how I got out of this roadblock. Since I was in his shoes a few months ago, I could give him the best advice. It is best to solve this problem now since the chosen network topology will decide the game's direction and the multiplayer's implementation.

5.4 Industry Overview

Game engines such as Unreal Engine and Unity have recognized the complexity of game development and have created workflows and protocols to speed up game development through their software. They have documentation describing how to set up multiplayer through their engines, along with descriptions and explanations of the concepts. For example, Unity has a section on network topologies that was quite useful in the research for this report. (Unity Technologies, 2022) Likewise, Unreal Engine has a similar document that gives an in-depth overview of how networking concepts can be used for multiplayer video game development. (Unreal Engine, n.d.) In general, game engine companies have this role in the video game development industry of reducing the complexity for game developers.

5.5 Credibility and Research

My role in researching network topologies is to gather as many sources as necessary to provide a simple explanation of network topologies, describe the different types, evaluate them according to criteria suitable for a first-person shooter multiplayer video game, and recommend the most effective solution whether that be a single network topology or a combination since video games can have multiple

network topologies depending on their wants and needs. These findings will benefit Wolfram because he will know exactly what a network topology is, the types of network topologies, and which of them he should implement for his first-person shooter multiplayer video game, Wolf.

6 Research and Analysis

6.1 Introduction

The culmination of my research will be shared in two parts, research and analysis. The research will explore the five gaming network topologies from the Unity documentation. The analysis will analyze each network topology and evaluate them according to the criteria in Table 2.

6.2 Research

This section will explore the five gaming network topologies from the Unity documentation. First, the definition of network topology will be stated. Second, the source will be briefly described. Lastly, Unity's network topologies and their descriptions will be tabularized.

6.2.1 Unity

According to the Unity documentation, a "network topology defines the way a network is arranged, including how links and nodes relate to each other, clients and hosts, physical machines and virtual" (Unity, 2022). Unity's documentation on network topologies was well put together in terms of learning the concepts. They explicitly state each network topology, give a short description of it, and provide a figure that shows its configuration. However, they don't provide any implementations of the network topologies and state that "You can find implementations in the Netcode-community-contributions repository." (Coughlin & Unity, 2021) They don't include a link to the repository, though, which is a poor user experience. Thus, Unity documentation is useful for learning about network topologies but not useful for implementing them. The five network topologies and their descriptions are stated in Table 1.

Table 1: The five gaming network topologies (Unity Technologies, 2022)

Type	*Description*
Local or couch multiplayer	Local or couch multiplayer is a simple networking perspective where the players play on a single machine. It consists of a single game client, but the client processes multiple inputs (e.g., Call of Duty split screen). It actually doesn't involve any networking at all! But since it's the basis of multiplayer, it needs to be included. Most games that have an online multiplayer feature should also include local multiplayer for offline players.
Lan game (offline)	A LAN game connects multiple player devices through a network to a single device called a server. There are two implementations of the game: server and client. The server is made up of different code that can be run on the client or on a standalone device. The client is a player's game and sends requests to the server. The server

	answers these requests by running code and sending the results to each client. This method doesn't require an internet connection, but players must be connected to the same network.
Peer-to-Peer (P2P)	In P2P, the game is played on different devices connected directly to each other without needing a server. P2P connects multiple client devices to each other through an online network (e.g., the Internet). The game code is run on all devices, and results are synchronized without needing a server.
Client hosted (Listen server)	This method is like an online LAN game. The listen server connects multiple client devices through an online network to a server. Although, the server is hosted on one of the client's devices instead of a separate device. This client is known as the host. The host runs both the client and the server code in one process. While the other clients run just the client code. This requires the host to have both high internet speeds and computational ability.
Dedicated Game Server (DGS)	This method is like the client hosted (listen server) method, except the server is hosted on a device whose sole purpose is to run the server code, hence the name dedicated. The clients run the client code that connects to this server.

6.3 Analysis

Each network topology will be analyzed and evaluated according to the criteria in Table 2. Each criterium will receive a ✅ if its needs are met, a 🔄 if some, but not all, needs are met, and an ❌ if the needs are not met. Each network topology will have an accompanying description explaining its results. The description will explain the pros, mixed results (if any), and cons.

Table 2: Gaming network topology evaluation criteria (Unity, 2018)	
Name	*Brief description*
Latency	Responsiveness of the game
Scale	Number of players online at the same time
Security	Ability to withstand hacks and protect players' systems
Cost	Pricing of services needed
Complexity	Time and difficulty to implement
Reach	Number of players supported and distance of those players

Table 3: Analysis of the first four gaming network topologies (Unity, 2018)							
Local or couch multiplayer		**Local area network (LAN) game (offline)**		**Peer-to-Peer (P2P)**		**Client hosted (Listen server)**	
Latency	✅	Latency	✅	Latency	🔄	Latency	🔄
Scale	❌	Scale	✅	Scale	❌	Scale	❌
Security	✅	Security	✅	Security	❌	Security	❌
Cost	✅	Cost	✅	Cost	✅	Cost	✅
Complexity	✅	Complexity	✅	Complexity	🔄	Complexity	❌
Reach	❌	Reach	❌	Reach	🔄	Reach	🔄

6.3.1 Local or couch multiplayer

The pros of local or couch multiplayer are latency, security, cost, and complexity. Since there is no networking involved, there is literally zero latency which results in ideal conditions for responsive gameplay. Since the game code is run on the client, players can cheat, but it is of little risk since the only players affected by the cheating are one to three other local players. Cost is free because it doesn't cost anything to run code on the client. Complexity is also quite low due to no networking challenges.

The cons of local or couch multiplayer are scale and reach. Since no networking is involved, players can only play with each other on the same device. This typically involves four or fewer players. However, some video games have unique gameplay scenarios which allow up to eight players on the same device (e.g., Brawlhalla, Gang Beasts). There is a player limit, and players must be in person to play together.

6.3.2 Local area network (LAN) game (offline)

The pros of LAN games are latency, scale, security, cost, and complexity. Since the players are on the same network, the latency is low, which creates responsive conditions similar to that of a local multiplayer game. The scale for LAN games is large because they can support 100+ players. The maximum number of players depends on the local server, but typical implementations can support around 100 players simultaneously. The security of a LAN game depends on the security of the local server but typically involves low risks. If the network is password protected and trust is maintained between the players, then there should be no problems. Since the game isn't connected to the internet, only onsite players connected to the network will be able to perform a hack. Cost is free because the players host the servers themselves. Complexity is higher than local or couch multiplayer because you need to develop a server version of your game in addition to the client. However, the complexity is still low compared to the next three network topology implementations.

The only con of LAN games is reach. Since the network topology is limited to the local area network, players can only play if they are on the same network, which is usually only within the same building.

6.3.3 Peer-to-peer (P2P)

The only pro of P2P is cost. Since there is no server, there is no cost.

The mixed results of P2P are latency, complexity, and reach. Latency is low if players are close but high if players are far, resulting in inconsistent gameplay. This can be fixed by using a matchmaking service that only allows players nearby to connect to each other. However, a service like this usually costs money, nullifying the free aspect of P2P. Complexity is mixed because it can be difficult to handle client synchronization. However, the developers only need to develop a client game version since there is no server. Reach is mixed because P2P can reach anyone worldwide through an internet connection, but only if the clients' network-address-translation (NAT) types are compatible (Coughlin, 2021).

The cons of P2P are scale and security. Since the game is synchronized between each client, the more players, the more difficult it is to synchronize the game. Typically, the max number of players that can play at one time is 12. P2P is insecure because "the data the player sends to others can be manipulated, making cheating easier." (Roxl, 2021)

6.3.4 Client hosted (Listen server)

The only pro of a listen server is cost. Since the server is run by the client, there is no cost.

The mixed results of a listen server are latency and reach. Like in P2P, latency is low if players are close to the host but high if players are far from the host, resulting in inconsistent gameplay. Also, the host doesn't have any latency since they are running the server code, which gives them a significant advantage over other players. For example, if a player has a latency of 100ms, then the player lags behind the host by 100ms. Multiplayer shooter games rely on reaction time, so every ms counts. The other mixed result is reach. Like P2P, client hosted listen servers can reach anyone worldwide but only if the players' NAT types are compatible, resulting in inconsistent matchmaking (Coughlin, 2021).

The cons of a listen server are scale, security, and complexity. Scale depends upon the host's internet connection quality and computation speeds. Typically, only 10-20 players can join a client-hosted listen server. Security is bad as well, like P2P, because "the data the player sends to others can be manipulated, making cheating easier." (Roxl, 2021) Also, the IP addresses of all the clients are exposed. With the right tools, you can find someone's location using their IP address. This is a severe invasion of privacy. Complexity is bad because developers must build a server into the client game version, which can be difficult. Also, if a host quits the game, then the server shuts down, and the multiplayer game is also lost. There is a way around this through a process called host migration. But it requires the developers to set up an always-on server which can be costly, thus nullifying the only pro of a client hosted listen server.

Table 4: Analysis of the fifth and final gaming network topology (Unity, 2018)

Dedicated Game Server (DGS)	
Latency	✅
Scale	✅
Security	✅
Cost	❌
Complexity	❌
Reach	✅

6.3.5 Dedicated Game Server (DGS)

The pros of DGS are latency, scale, security, and reach. Latency is low if players are close to the servers. It is also consistent because all players have equal latency. Typically, DGSs have a commercial internet connection which has high connection speeds. The scale is high because DGSs tend to have high computation speeds and don't have to worry about client synchronization because the code is run once, and results are sent to each client. Typically, there are 100+ players connected to a DGS. Security is also great since the code runs on the server, not the client. Thus, cheaters can't alter data sent to other players. IP addresses aren't exposed either, allowing players to game in privacy. A DGS can reach anyone worldwide regardless of their NAT type. And since the server is the host, players leaving the match have no effect on the game state, like in a client hosted game.

Despite all these pros, there are still some cons of DGS: cost and complexity. Cost can get quite high because it requires server hosting and services. Running a server 24/7 is expensive; you also need a cool place to store the computer. And if you want fast gameplay worldwide, you will need a server in each continent. Thus, developers tend to rely on major service companies such as Amazon Web Services or Microsoft to host their DGSs worldwide. These services have a pay-as-you-go price model, which is certainly cheaper than setting up everything yourself. However, it is still more expensive than any of the previous network topologies. Complexity is also high because the developers have to know the entire multiplayer development stack, including both client and server game versions, multiplayer services such as matchmaking, player identity, and persistent data, as well as server management services like hosting, scaling, and updates. Once again, major service companies can reduce some of this complexity, though it is still more complex than any of the previous network topologies.

6.4 Conclusion

The culmination of my research was shared in two parts, research and analysis. The research explored the five gaming network topologies from the Unity documentation. The analysis examined each network topology and evaluated them according to the criteria in Table 2.

7 Conclusion of Findings

I found that the best network topologies for multiplayer shooter games have low latency since players need to be able to react quickly to shoot at enemies. In addition, the best online network topologies have good security and can reach players worldwide. It is also important to have both online and offline multiplayer when designing a multiplayer game. That way, you account for players with poor or no internet connection. So it is best to break up the finding into two sections: offline and online

7.1 Offline

Based solely on the pros and cons from my analysis, LAN appears to be the best, and local/couch comes in second. However, these two offline network topologies serve different purposes and cannot be compared to choose one or the other. It is best to consider each of their use cases and decide whether or not to implement them from there. If you want to have multiplayer on one device, then use local/couch. And if you want to have multiplayer on multiple devices, then use LAN.

7.2 Online

Based on the pros and cons of each network topology, the dedicated game server is the best for a multiplayer shooter game due to its worldwide reach, low latency, and good security. P2P and Listen servers don't even come close. However, these are still used if the developers don't have the money to pay for a dedicated game server.

8 Recommendations

My recommendation for the most effective network topologies for multiplayer shooter games are a combination of two important network topologies, one online and one offline:

1. Dedicated Game Server (DGS)
2. Local or couch multiplayer

I recommend a DGS for your first-person shooter multiplayer game Wolf because a DGS can reach gamers worldwide, has low latency, is easily scalable, and is secure. The costs are high, but if your game reaches the masses, your players will thank you for setting up a good online experience. And you can

add ways to make money off the game, such as microtransactions or advertisements to pay for the server costs. The only drawback left is complexity. The complexity of a dedicated game server is high and will certainly be no easy task to set up. Keep this in mind before you get started.

I also recommend adding local or couch multiplayer as an alternative to online multiplayer, so offline players can still play with their friends. Local multiplayer is quite easy to set up, so you'll still have plenty of time to set up the DGS.

8.1 Optional Recommendation

LAN multiplayer serves a different purpose than local multiplayer does and is valuable in specific use cases such as eSports competitions. So only implement it if you need it. It is important to note that LAN does not replace local multiplayer.

9 Bibliography

Bajpai, P. (2021, February 24). *How Cloud Computing is Changing the World of Games and Gaming*. Nasdaq. Retrieved October 24, 2022, from https://www.nasdaq.com/articles/how-cloud-computing-is-changing-the-world-of-games-and-gaming-2021-02-24

In this informative article, Bajpai discusses how cloud computing is changing the gaming world by providing an overview of cloud gaming and how it's evolving. She explains that cloud gaming is like "Netflix for games." Then, she lists major companies that have released cloud gaming services for consumers and developers.

The information is reliable because it is on NASDAQ, a reputable website, and the author includes many references. This article benefits my research by pointing out game cloud services for developers.

Coughlin, B. (2021, October 12). *Create a game with a listen server / host architecture*. Unity Multiplayer Networking. Retrieved October 30, 2022, from https://docs-multiplayer.unity3d.com/netcode/current/learn/listen-server-host-architecture

In this informative article, Coughlin explains the listen server and host architecture, disadvantages and advantages, use cases, connection strategies, and recommendation. This is useful for my research because it will help me explain how the client hosted listen server network topology works and how to analyze it.

Ferris, R. -. (2016, April 7). *Stephen Wolfram: Why this brilliant physicist ditched his job*. CNBC. Retrieved October 24, 2022, from https://www.cnbc.com/2016/04/07/stephen-wolfram-why-this-brilliant-physicist-ditched-his-job.html

This article explains the brilliance of Stephen Wolfram. This source is useful for my story about him as the stakeholder.

Game Backend Services. (2021, May 12). Flentas Technologies. Retrieved October 24, 2022, from https://www.flentas.com/key-backend-services-for-games

In this informative article, Khatri breaks down what mobile game backend services are and lists the types of services found in successful games. She comes up with two types of services, functional and operational, to better organize these services. The player utilizes functional services, while the game developer/studio utilizes operational services.

The information in this article doesn't seem as reliable as the other sources because it doesn't include any references. However, the site, Flentas, has a partnership with Amazon Web Services, which increases its credibility. This information will benefit my research because it shares a list of features I need to consider when comparing backend services.

Game Development Software: Build a Multiplayer Game. (n.d.). Unity. Retrieved October 24, 2022, from https://unity.com/solutions/build-backend

On this webpage on Unity's website, the company shows off its product Unity Game Services (UGS), with features like multiplayer, a user account system, and live game management.

This information is reliable because it comes straight from the source. This page will be helpful for my research when comparing cloud service providers as it contains most of the information or links to helpful information.

House, B., & Unity Technologies. (n.d.). *Choosing the right netcode for your game*. Unity Blog. Retrieved October 24, 2022, from https://blog.unity.com/technology/choosing-the-right-netcode-for-your-game

In this informative article based on the research of the Unity team, House attempts to provide an "in-depth overview" of choosing a netcode framework for building a multiplayer game in Unity. Using data from a survey of 200+ Unity users, 20 in-depth interviews, and learnings from prototypes, House's team created tools to help guide readers through the process of choosing a framework. These tools consider customer feedback based on six aspects: stability/support, ease of use, performance, scalability, feature breadth, and cost.

The information is reliable because the Unity Technologies team presents it. The source is credible because it is a source of primary information based on research they conducted. They provide their criteria for grading frameworks and are unbiased by not picking a "best in category" option. This source will be helpful for my research because it provides a high-level guide to evaluate which multiplayer implementation is right for my game type.

Madhav, S., & Glazer, J. (2016). Network Topologies. In *Multiplayer Game Programming: Architecting Networked Games* (1st ed., p. 384). Addison-Wesley.

In this book chapter, Madhav and Glazer explain the use of network topologies in multiplayer game programming. They provide explanations of the concepts and programming implementations. This is useful for my research because it provides a new perspective on my research topic.

NotoriousAJ. (2018, October 12). *Split Screen Gameplay - Call of Duty: Black Ops 4*. YouTube. Retrieved October 26, 2022, from https://www.youtube.com/watch?v=InQh7d2YwFA

An example of local or couch multiplayer gameplay in the hit series Call of Duty: Black Ops's split screen mode. This is useful for my research on the local or couch multiplayer network topology.

Pavlovic, D. (2020, July 23). Video Game Genres Everything You Need To Know. HP. Retrieved October 30, 2022, from https://www.hp.com/us-en/shop/tech-takes/video-game-genres

In this informative article on HP's website, Pavlovic explains everything you need to know about video game genres. This is useful for defining the scope of this project.

Roxl, R. (2021, December 10). *What is Peer-to-Peer Gaming, and How Does it Work?* VGKAMI. Retrieved October 29, 2022, from https://vgkami.com/what-is-peer-to-peer-gaming-and-how-does-it-work/

In this informative article, Roxl explains what peer-to-peer gaming is and how it works. This is useful for my research and analysis of the peer-to-peer network topology.

Unity. (2018, November 19). *Connected Games: Building real-time multiplayer games with Unity and Google - Unite LA*. YouTube. Retrieved October 29, 2022, from https://www.youtube.com/watch?v=CuQF7hXlVyk

In this video, employees from Unity and Google explain how real-time multiplayer games are built and the challenges that come along with it. This video is useful for my research and analysis because it explains each of the five gaming network topologies and analyzes them. Which is what I'll be doing specifically for multiplayer shooter games.

Unity. (2022, May 4). *Network Topologies*. Unity Multiplayer Networking. Retrieved October 24, 2022, from https://docs-multiplayer.unity3d.com/netcode/current/reference/glossary/network-topologies/index.html

This webpage is from Unity's documentation on Multiplayer Networks. It defines network topology and provides information on the five configurations of network topologies in gaming. This information is reliable because it is a part of the documentation for Unity, a well-respected game engine. To change the documentation, a person must propose their changes via GitHub and receive a peer review by a Unity Technologies employee. This source will be helpful for my research because it provides definitions of network topology configurations, which I'm trying to determine and recommend.

Unreal Engine. (n.d.). *Networking Overview for Unreal Engine | Unreal Engine 5.0 Documentation*. Unreal Engine 5 Documentation. Retrieved October 24, 2022, from https://docs.unrealengine.com/5.0/en-US/networking-overview-for-unreal-engine/

On this website, we find Epic Games' documentation for Unreal Engine 5, the rival of Unity Game Engine. The documentation explains the multiplayer concept for video games with greater in-depth information for games built on their engine. It is refreshing to look at a source other than Unity. This site will prove useful in my research by giving me a new perspective on the same topic.

Williams, L. (2022, October 15). *Types of Network Topology: Bus, Ring, Star, Mesh, Tree Diagram*. Guru99. Retrieved October 24, 2022, from https://www.guru99.com/type-of-network-topology.html

In this informative article, Williams explains what topology is, the types of network topology: physical and logical, and the seven types of logical network topologies. With each logical network topology, Williams includes its advantages and disadvantages. He also discusses some important considerations for selecting the best topology to create a network in an organization.

This article will contribute to my research by providing a high-level overview of network topologies and how they work. I can use this knowledge to understand better the multiplayer network topologies described by the Unity Multiplayer Networking documentation.

Wolfram, S. (n.d.). *Stephen Wolfram*. Stephen Wolfram. Retrieved October 24, 2022, from https://www.stephenwolfram.com/

On this website, Stephen Wolfram describes his roles, companies, things he has done, and other things about himself. This source is useful for my story about him as the stakeholder.