1. Selecting the appropriate EC2 instances for the Spacetech Galac5c Holodeck application involves considering the various components and their requirements. Based on the complex and resource-intensive nature of the application, it's important to balance performance with cost efficiency. Here are some instance types to consider:

a. Compute-Optimized Instances like C5, C5n. They are ideal for compute bound applications that benefit from high performance processors. Instances belonging to this category are well suited for high performance web servers, high performance computing (HPC), scientific modeling, dedicated gaming servers [1].

b. Accelerated Computing Instances like P3, G4, F1. Accelerated computing instances use hardware accelerators, or co-processors, to perform functions, such as graphics processing, or data pattern matching, more efficiently than is possible in software running on CPUs.

c. Storage-Optimized Instances like I4g, D2. Storage optimized instances are designed for workloads that require high, sequential read and write access to very large data sets on local storage. They are optimized to deliver tens of thousands of low-latencies, random I/O operations per second (IOPS) to applications [1].

d. Considering the network performance requirements, especially for the multiplayer mode and global connectivity. Instances from the High Network Performance family like m5n or r5n may be suitable [1].

2. Multi-AZ and Multi-Region Deployment: Implement a Multi-AZ and Multi-Region deployment to ensure high availability and fault tolerance.

To achieve high availability and fault tolerance for Spacetech Galac5c's Holodeck application, we should implement a Multi-Availability Zone (Multi-AZ) and Multi-Region deployment strategy. This will ensure that the application remains accessible even in the event of AZ or regional failures. Here is how we can do it:

Multi-Region Deployment:

a) Disaster Recovery Plan: Developing a disaster recovery plan that outlines how you would handle a regional failure, including data recovery and failover procedures.
b) Data Synchronization: Implementing data synchronization mechanisms to keep data consistent across regions. AWS offers services like AWS Database Migration Service, AWS DataSync, and AWS Data Pipeline to assist with data replication.
c) Global Load Balancing: Setting up multiple Elastic Load Balancers (ELBs) in different regions and by using routing policies to distribute traffic globally.

Multi-AZ Deployment:

a) Elastic Load Balancer (ELB): By Using an Elastic Load Balancer to distribute incoming traffic across EC2 instances in different AZs to ensure even distribution of traffic across Availability Zones. Configuring health checks to monitor the health of instances, and then allowing the ELB to route traffic only to healthy instances.

b) Data Replication and Synchronization: As Application involves data that needs to be synchronized across instances, using database replication or distributed data storage solutions to keep data consistent across different AZs.

3. EBS & EFS: Plan a strategy for data storage using Amazon EBS and/or EFS.

Selecting between Amazon Elastic Block Store (EBS) and Amazon Elastic File System (EFS) depends on specific data storage needs in the scenario of the Spacetech Galactic Holodeck application.

a) Using Amazon EBS for storing data that requires high performance and low-latency access such as for the 3D models of planets, spaceships, and extraterrestrial lifeforms, as well as the real-time physics engine, benefit from EBS for their performance requirements. Implementing EBS snapshots for backup and data recovery. This is essential for preserving the 3D models and simulation data, as well as for disaster recovery.

b) Amazon EFS for shared storage that can be accessed by multiple instances concurrently. Since application have a multiplayer mode where users from diverse geographical locations interact, EFS is suitable for storing shared resources and game state data. EFS automatically scales to accommodate growing storage needs, making it suitable for managing the vast amount of data collected from users and the multitude of static content required for VR experiences. This scalability is valuable for handling the dynamic demands of the application.

4. Spot instances can offer up to 90% discount but quickly get interrupted, reserved instances offer up to 72% decreased rates and high availability, but with long commitment periods [2]. However, there is a catch that we need to be careful about: AWS can terminate spot instances at any time and with a short termination notice. This happens when the capacity demand increases or your bid for the spot instance gets outbid. When AWS wants to reclaim a spot instance, it will send a two-minute warning through CloudWatch Events and instance metadata [2]. we can use these two minutes to save the application state, upload log files, or drain any presently running containers [2]. So, in our case we cannot proceed with Spot Instance type.

Reserved instances allow us to reserve a specific amount of computing capacity for a fixed time period, which is usually one or three years. Reserving the capacity means we can launch the instances whenever required. They are more suitable for workloads with a steady and predictable demand. So, this is an ideal choice for our application.

5. Explain how you would use Amazon Machine Images (AMIs) to quickly deploy and replicate your application.

An Amazon Machine Image (AMI) is a template that contains a software configuration (for example, an operating system, an application server, and applications) [3]. All AMIs are categorized as either backed by Amazon EBS, which means that the root device for an instance launched from the AMI is an Amazon EBS volume, or backed by instance store, which means that the root device for an instance launched from the AMI is an instance store volume created from a template stored in Amazon S3 [3].

So, in our case, we can create our own AMI with the pre-determined configurations, software, settings, and data. Doing so enables us to quickly and easily start new instances that have everything we need.

Essentially, using Amazon Machine Images (AMIs) simplifies the process by eliminating the need to reinstall and configure software manually each time. This streamlines system deployment across different regions, ensuring efficiency and consistency in the setup.

6. Cost Management: Provide a rough estimate of the costs of running this infrastructure and discuss the strategies you would use to manage these costs.

As I am going to use Compute-Optimized Instances C5-4XLarge and the cost for the region US East (N.Virginia) is

The c5.4xlarge instance is in the compute optimized family with 16 vCPUs, 32.0 GiB of memory and up to 10 Gibps of bandwidth starting at $0.68 per hour.

$5956.80 - On Demand

$2003.41 – Spot

$3749.28 - 1 Yr Reserved

$2496.60 - 3 Yr Reserved

$208.05 per month (-58%) with Autopilot

By selecting the type as required for our use case we can go with reserved instance for 3 years as we can save rather than going for 1 year.

Storage-Optimized Instances I4g. The i4g.4xlarge instance is in the storage optimized family with 16 vCPUs, 128.0 GiB of memory and up to 25 Gibps of bandwidth starting at $1.23552 per hour.

$1.236 - On Demand

$0.3762 - Spot

$0.8018 - 1 Yr Reserved

$0.5715 - 3 Yr Reserved

7. EC2 Placement Groups: Design an architecture where the application has a need for low-latency, high throughput communication between instances.

EC2 Placement Groups are a feature of AWS that allow us to place instances within the same group in close physical proximity to each other. Use of a cluster placement group can span peered virtual private networks (VPCs) in the same Region. Instances in the same cluster placement group enjoy a higher per-flow throughput limit for TCP/IP traffic and are placed in the same high-bisection bandwidth segment of the network [4].

In our application we can make use of Cluster placement groups as that benefit from low network latency, high network throughput, or both. They are also recommended when the majority of the network traffic is between the instances in the group. To provide the lowest latency and the highest packet-per-second network performance for your placement group, choose an instance type that supports enhanced networking [4].

8. Instance Store: Design a scenario where temporary, high-IOPS storage is required and an instance store would be used.

Scenario where an instance store would be used:

One of the most thrilling experiences offered by the Holodeck is breathtaking interstellar journeys at high-speed space travel. To make these experience as realistic as possible, real-time physics calculations and process a large number of high-resolution textures and 3D models are required to simulate the warp spaceship thrusters, space-time distortion, and collision detection as well as users with a seamless experience. In this scenario, instance stores are used to provide temporary, high-IOPS storage for the real-time physics engine, which is an essential component of the Holodeck application.

9. Dedicated Hosts/Instances: Plan for scenarios where the application has to comply with strict licensing terms (BYOL) or meet dedicated hardware requirements.

Dedicated Instances are Amazon EC2 instances that run in a VPC on hardware that's dedicated to a single customer. Your Dedicated instances are physically isolated at the host hardware level from instances that belong to other AWS accounts [5]. Dedicated instances may share hardware with other instances from the same AWS account that are not Dedicated instances. Dedicated Hosts give you additional visibility and control over how instances are placed on a physical server, and you can reliably use the same physical server over time [5].

10. EC2 Metadata and User Data: Describe how you would use EC2 metadata and user data to handle configuration tasks and pass information to instances at launch time.

Instance metadata is data about our instance that we can use to configure or manage the running instance. Instance metadata is divided into categories like host name, events, and security groups [6].

EC2 metadata provides a way to retrieve instance-specific information, such as the instance ID, IP address, availability zone, and more. We can use metadata to dynamically configure our instances. For example, we can retrieve metadata to determine which role an instance should assume or which environment it's running in (e.g., production, staging, or development). And also, instance type, CPU, memory, and storage through metadata. This can be helpful for optimizing resource utilization and capacity planning.

User data allows us to run custom scripts or commands when an instance is launched. This can be used to perform various configuration tasks, software installations, and application setup during the instance's bootstrapping phase, installing software, setting environment variables, and performing post-launch configuration. While user data can be used for configuration, be cautious about passing sensitive information like database passwords. User can use AWS Systems Manager Parameter Store or AWS

Secrets Manager can be used to securely store and retrieve secrets, and then we can reference them in our user data scripts

11. Optimization and Performance: Describe how to use tools like AWS Compute Optimizer and Trusted Advisor for identifying optimal EC2 instance types and for maintaining cost efficiency

AWS Compute Optimizer is a service that analyzes your EC2 instances and makes recommendations for optimizing performance and cost-efficiency. It can be instrumental for enhancing the performance and cost efficiency of the Spacetech Galac5c application. Once activated, Compute Optimizer performs automated analyses of our EC2 instances. Given the variable user traffic and resource demands of the Holodeck application, Compute Optimizer takes into account historical usage patterns and performance metrics to formulate recommendations. These recommendations encompass various aspects, such as suggested instance types and purchase options like reserved instances.

AWS Trusted Advisor plays a vital role in optimizing the Spacetech Galac5c application's performance and cost efficiency. Trusted Advisor provides recommendations specific to EC2, including insights on resizing instances, purchasing reserved instances, or optimizing storage. Given the stringent requirements for low latency and the application's need to handle surges in demand seamlessly, it's crucial to implement these recommendations as they align with your EC2 instances. Regularly checking Trusted Advisor for new recommendations is an excellent practice to ensure the Spacetech Galac5c application remains cost-efficient and continues to meet its evolving operational needs. This process should be integrated into your routine to maintain peak performance at minimal cost.

References

[1] *Amazon.com*. [Online]. Available: https://aws.amazon.com/ec2/instance-types/#:~:text=Amazon%20EC2%20M7a%20instances%2C%20powered,50%20Gbps%20of%20networking%20bandwidth. [Accessed: 21-Oct-2023].

[2] nOps, “Spot Instances vs Reserved Instances: Which is the right EC2 pricing model?,” *nOps*, 10-Jun-2023. [Online]. Available: https://www.nops.io/blog/spot-instances-vs-reserved-instances/. [Accessed: 21-Oct-2023].

[3] *Amazon.com*. [Online]. Available: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instances-and-amis.html. [Accessed: 21-Oct-2023].

[4] *Amazon.com*. [Online]. Available: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-groups.html. [Accessed: 21-Oct-2023].

[5] *Amazon.com*. [Online]. Available: https://aws.amazon.com/ec2/pricing/dedicated-instances/#:~:text=Dedicated%20Instances%20are%20Amazon%20EC2,belong%20to%20other%20AWS%20accounts. [Accessed: 21-Oct-2023].

[6] *Amazon.com*. [Online]. Available: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instance-metadata.html. [Accessed: 21-Oct-2023].