



Download Esri EGMP2201 Exam Dumps Free

Shared by Hurley on 17-06-2026

For More Free Questions and Preparation Resources

Check the Links on Last Page



Question 1

Question Type: MultipleChoice

A GIS administrator needs to facilitate the collaboration of two teams of GIS analysts in two different offices. Each office needs a copy of the data in its own enterprise geodatabase, and analysts in both offices will edit the same feature classes. Changes will be synchronized nightly.

The GIS administrator needs to set up the information infrastructure so that both teams can work together.

What should the administrator use to meet the requirements?

Options:

- A- Geodatabase replication
- B- Database replication
- C- Distributed collaboration

Answer:

A

Explanation:

To facilitate collaboration between two teams of GIS analysts located in different offices, each requiring a copy of the data in their own enterprise geodatabase with the ability to edit the same feature classes and synchronize changes nightly, geodatabase replication is the appropriate solution.

Understanding Geodatabase Replication:

Geodatabase replication is a data distribution method in ArcGIS that allows you to create copies of data across two or more geodatabases. This enables multiple users to work with the same datasets in different locations, with the ability to synchronize changes to ensure consistency.

ARCGIS PRO

Types of Geodatabase Replication:

There are three types of geodatabase replication:

One-Way Replication: Changes are sent in a single direction---from the parent to the child replica.

Two-Way Replication: Changes are synchronized in both directions between the parent and child replicas. This is suitable when multiple editors need to update the same datasets in different

locations.

Checkout/Check-in Replication: Data is checked out to a child replica for editing and then checked back in to the parent replica.

In this scenario, two-way replication is ideal, as it allows both teams to edit the same feature classes and synchronize changes nightly, ensuring that both geodatabases remain consistent.

ARCGIS PRO

Alternative Options:

Database Replication: This refers to replicating entire databases at the DBMS level. While it can synchronize data, it doesn't account for the geodatabase-specific behaviors, rules, and relationships managed by ArcGIS. Therefore, it may not be suitable for scenarios requiring synchronization of geodatabase-specific functionalities.

Distributed Collaboration: This is a framework in ArcGIS Enterprise that allows sharing of content, such as maps, layers, and apps, across multiple ArcGIS Enterprise deployments or between ArcGIS Enterprise and ArcGIS Online. However, it doesn't provide the fine-grained control over data editing and synchronization required in this scenario.

GEODATABASE RESOURCES

Therefore, to meet the requirements of both teams being able to edit the same feature classes in their respective enterprise geodatabases and synchronize changes nightly, geodatabase replication is the most appropriate solution.

Question 2

Question Type: MultipleChoice

Multiple editors in a web application need to collaboratively edit the same dataset using the following requirements:

- * Each editor works in isolation until they come to a good stopping point
- * The editor shares their edits with the other editors at this point
- * The editor again goes into isolation to continue editing until the next stopping point is reached
- * At any point, an editor can choose to see the edits that other editors have shared without sharing their own edits

How should the dataset be registered?

Options:

- A- Branch versioned
- B- Traditional versioned with editor tracking enabled
- C- Traditional versioned with the option to move edits to base

Answer:

A

Explanation:

Scenario Overview:

Multiple editors need to collaboratively edit the same dataset using a web application.

Requirements:

Editors work in isolation until they reach a stopping point.

Edits are shared with others at the stopping point.

Editors can view shared edits without sharing their own.

Branch Versioning in Enterprise Geodatabases:

Branch versioning is specifically designed for collaborative editing in web-based workflows.

It supports isolated editing by creating branches for each editor and allows users to reconcile and post changes at their discretion.

Editors can choose to reconcile shared changes without posting their own edits, fulfilling the requirement to view shared edits without sharing their own. (ArcGIS Documentation: Branch Versioning)

Key Features of Branch Versioning:

Supports Web Applications: Designed to work seamlessly with feature services, enabling real-time collaborative editing.

Isolation: Each editor can edit independently in their branch.

Reconciliation: Editors can reconcile and view changes made by others without posting their edits.

Flexible Sharing: Editors control when to post edits.

Alternative Options:

Option B: Traditional versioned with editor tracking enabled:

Traditional versioning supports isolated editing, but it does not provide the flexibility to view shared changes without posting your own edits.

It is also not as well-suited for web-based collaborative workflows.

Option C: Traditional versioned with the option to move edits to base:

Moving edits to base bypasses versioning workflows entirely and is not designed for collaborative editing.

Thus, branch versioning is the optimal solution for the requirements of this collaborative editing workflow in a web application.



Question 3

Question Type: MultipleChoice

A GIS data administrator receives a request to create a database view that meets the following criteria:

- * Data is combined from feature class and nonspatial table
- * Source feature class is versioned
- * Source is from a child version
- * Needs to be dynamically updated

How should the view be created?

Options:

- A- On the feature class and repository tables
- B- On the feature class and nonspatial table
- C- On the versioned view and nonspatial table

Answer:

C

Explanation:

Understanding the Scenario:

The request involves creating a view that dynamically combines data from a versioned feature class and a nonspatial table.

The data needs to be from a child version and updated dynamically.

Key Considerations:

Versioned Feature Class: Standard feature classes in versioned geodatabases store edits in delta tables (adds and deletes). Accessing data from a specific version requires using the versioned view, which includes these edits.

Dynamic Updates: Views created on the versioned view ensure that the data reflects the most current version edits.

Steps to Create the View:

Identify the versioned view for the feature class (created automatically during versioning).

Create a SQL query to join the versioned view and the nonspatial table on the appropriate key(s).

Save the SQL query as a database view.

Reference:

Esri Documentation: Versioned Views.

Creating Views with Versioned Data: Guidelines for joining versioned views with other tables in SQL.

Why the Correct Answer is C: Using the versioned view ensures that data reflects edits from the specified child version. Joining this view with the nonspatial table meets the requirement for dynamic updates. Options A and B would not provide data from the versioned child version dynamically.

Question 4

Question Type: MultipleChoice

A GIS administrator is investigating reports of slow performance in a map of complex polygons. The investigation finds the following:

- * The feature class is in an enterprise geodatabase
- * This data has been published as a map service in ArcGIS Enterprise
- * The feature class is not versioned
- * The map seems to perform slowly at several zoomed-out extents in the map

* Queries on the attributes perform quickly

Which recommendation should the GIS administrator make?

Options:

- A- Show cached data for the zoomed extents
- B- Run the Analyze Datasets tool
- C- Compress the enterprise geodatabase

Answer:

A



Explanation:

The slow performance at zoomed-out extents is likely caused by the rendering of complex polygon geometries in real time, which can be computationally intensive. Using cached data is the best approach to improve performance in this situation.

1. Reason for Slow Performance at Zoomed-Out Extents

Complex polygons require significant processing power to render at smaller scales due to the high number of vertices and complex geometry.

When users zoom out, the number of features being displayed increases, causing additional strain on the map service.

Attribute queries are not affected because they do not involve rendering the geometries.

2. Advantages of Using Cached Data

Cached tiles are pre-rendered images of the map at specific scales, stored on the server.

When cached data is used, the server simply retrieves and displays these images instead of dynamically rendering the features.

This drastically reduces server load and improves map performance at zoomed-out extents.

3. Why Not Other Options?

Run the Analyze Datasets Tool:

This tool checks for issues like invalid geometries, missing spatial indexes, or improper field types. While useful for general data health, it does not directly address rendering performance.

Compress the Enterprise Geodatabase:

Compression improves performance for versioned data by consolidating states and cleaning up the database. However, in this case, the feature class is not versioned, so compression would not resolve the rendering issues.

Steps to Implement Caching:

Open ArcGIS Server Manager and navigate to the map service configuration.

Enable tile caching and define the scales at which tiles should be created (include the problematic zoomed-out extents).

Generate the cache using the 'Manage Map Server Cache Tiles' tool.

Test the map to ensure performance improvements.

Reference from Esri Documentation and Learning Resources:

Map Caching in ArcGIS Enterprise

Best Practices for Map Services

Conclusion:

The GIS administrator should recommend enabling cached data for the zoomed extents to significantly improve map performance.

Question 5

Question Type: MultipleChoice

A GIS data manager needs to set up one-way parent-to-child replication to provide read-only copies of data to regional offices. The replication must be set up so that the parent geodatabase can be fully compressed, even if there are unacknowledged data change messages.

How should the replication be configured?

Options:

- A- Replicate only the base tables
- B- Use the Full replica access type
- C- Use the Archiving option

Answer:

A

Explanation:

To ensure that the parent geodatabase can be fully compressed, even with unacknowledged data change messages, the replication must be configured to replicate only the base tables.

1. What Happens with Unacknowledged Messages?

In traditional one-way replication, unacknowledged data change messages in the delta tables (Adds and Deletes) prevent full compression of the parent geodatabase.

By replicating only the base tables, the replication avoids using delta tables entirely, allowing the geodatabase to be fully compressed.

2. Why Replicate Only the Base Tables?

No Dependency on Delta Tables: This configuration ensures that the replication is based directly on the base table contents. As changes are not recorded in delta tables for replication, the parent geodatabase can be fully compressed without any impact.

Read-Only Copies: The replicated data in the child geodatabase will be read-only, which aligns with the requirement for regional offices.

3. Why Not Other Options?

Use the Full Replica Access Type:

Full replica access allows editing in the child geodatabase, which is unnecessary for read-only requirements. It also uses delta tables, preventing full compression.

Use the Archiving Option:

Archiving tracks historical edits and is unrelated to the replication or compression process. It does not solve the problem of unacknowledged messages blocking compression.

Steps to Set Up One-Way Replication with Base Tables:

Open ArcGIS Pro and connect to the parent geodatabase.

Use the Create Replica tool and select One-Way Replication.

Choose the option to replicate base tables only during the configuration process.

Define the datasets to replicate and complete the replication setup.

Reference from Esri Documentation and Learning Resources:

One-Way Replication Overview

Compressing Enterprise Geodatabases

Conclusion:

Configuring replication to replicate only the base tables ensures that the parent geodatabase can be fully compressed, even with unacknowledged data change messages.

Question 6

Question Type: MultipleChoice

A GIS administrator needs to convert an existing database into a geodatabase. The new database has been created, but the repository owner has not been created. The GIS administrator has been given the database administrator credentials for this conversion to a geodatabase.

Which tool should the GIS administrator use?

Options:

- A- Create Enterprise Geodatabase
- B- Enable Geodatabase
- C- Create Workflow Database

Answer:

A

Explanation:

Understanding the Scenario:

A database exists but lacks the repository owner required for a geodatabase.

The GIS administrator has database administrator credentials to set up the geodatabase.

Options Overview:

Create Enterprise Geodatabase: This tool creates a geodatabase in a relational database. It sets up the repository owner and adds the required system tables, schema, and configurations.

Enable Geodatabase: This tool enables geodatabase functionality in an existing database that already has the repository owner and structure. Since the repository owner is missing, this tool cannot be used.

Create Workflow Database: This tool is unrelated to creating or enabling geodatabases. It is used

for managing workflows in a production environment.

Steps to Use Create Enterprise Geodatabase:

Open the Create Enterprise Geodatabase tool in ArcGIS Pro.

Provide the database connection information, including administrator credentials.

Specify the authorization file for geodatabase licensing.

Run the tool to create the geodatabase.

Reference:

Esri Documentation: Create Enterprise Geodatabase.

Why the Correct Answer is A: The Create Enterprise Geodatabase tool is specifically designed for initializing a geodatabase, including creating the repository owner and required system components. The other options are either not applicable or insufficient for the given scenario.

Question 7

Question Type: MultipleChoice

A GIS administrator needs to simultaneously grant access for multiple users of a branch versioned feature class in an enterprise geodatabase.

How should permissions be granted?

Options:

- A- To a role
- B- To a list
- C- To a group

Answer:

A

Explanation:

Understanding the Scenario: The GIS administrator wants to simultaneously grant access to multiple users for a branch-versioned feature class in an enterprise geodatabase. Managing permissions for individual users can be inefficient, especially for larger teams, making roles a

better solution.

Why Grant Permissions to a Role:

Role-Based Access Control (RBAC): In enterprise geodatabases, assigning permissions to a role allows administrators to define access levels once and apply them to multiple users who are members of that role. This simplifies management and ensures consistent permission levels across users.

Branch Versioning Considerations: In branch versioned datasets, role-based permissions ensure that all users working with the feature class can access, edit, or view data as needed without individual customization.

Steps to Assign Permissions to a Role:

Create a role in the database or use an existing one.

Assign permissions for the branch versioned feature class to the role (e.g., read, edit, or administrative permissions).

Add users to the role.

All users in the role inherit the permissions automatically.

Reference:

Esri Documentation: Best practices for managing permissions in enterprise geodatabases: Managing Users and Roles.

Branch Versioning Overview: Information on the unique requirements for branch versioned data: Branch Versioning.

Why the Correct Answer is A: Using roles allows the GIS administrator to efficiently grant and manage access for multiple users at once. Options like lists or groups are not applicable in the context of enterprise geodatabases, as roles are the standard mechanism for assigning permissions.

Question 8

Question Type: MultipleChoice

A GIS data administrator needs to load a large amount of data into a version, verify its quality, and then reconcile and post this version to default. The data administrator needs to create the fewest number of rows in the database.

Which versioning method should be used?

Options:

- A- Traditional versioning without the archiving option
- B- Traditional versioning with the archiving option
- C- Branch versioning

Answer:

A

Explanation:

To minimize the number of rows created in the database while performing versioning workflows (loading, quality checking, reconciling, and posting), Traditional versioning without the archiving option is the best choice.

1. Traditional Versioning Without Archiving

This method stores edits in delta tables (Adds and Deletes) rather than directly in the base table.

Without the archiving option, the system does not create additional rows to track historical changes, which helps reduce the number of rows.

2. Why It's Ideal for This Workflow

Load Data: Data is directly inserted into the delta tables, keeping base tables untouched.

Quality Verification: Edits can be reviewed and adjusted without additional overhead.

Reconcile and Post: Only the changes made during the session are pushed to the default version, and unnecessary rows are avoided.

3. Why Not Other Options?

Traditional Versioning with Archiving Option:

Archiving tracks historical changes, creating additional rows for each edit in the archive tables. This increases storage and processing overhead.

Branch Versioning:

Branch versioning stores all changes in a single table and is designed for web services workflows. It may not minimize row creation compared to traditional versioning.

Steps for the Workflow:

Enable Traditional Versioning for the target dataset without enabling archiving.

Load the large dataset into a new version created for this purpose.

Verify the data quality by querying and editing the version.

Reconcile the version with the default version, resolve conflicts, and post changes to default.

Reference from Esri Documentation and Learning Resources:

Understanding Traditional Versioning

Archiving in Enterprise Geodatabases

Branch Versioning vs. Traditional Versioning

Conclusion:

Using Traditional versioning without the archiving option ensures the creation of the fewest number of rows while maintaining data integrity and supporting the described workflow.



To Get Premium Files for EGMP2201 Visit

<https://www.p2pexams.com/products/egmp2201>

For More Free Questions Visit

<https://www.p2pexams.com/esri/pdf/egmp2201>

20%
DISCOUNT

P2P
exams