Best Practices

Performance Optimization

Limit the region: Restrict the lambda to the smallest area necessary to reduce compute costand time

Efficient algorithms: Use vectorized operations where possible for better performance

Memory management: Be mindful of memory usage when processing large datasets

Development Workflow

Test on a sample: Validate your code on a small subset before running it on the full dataset

Incremental development: Start with simple logic and gradually add complexity

Error handling: Include proper error handling for robust execution

Resource Management

Reuse templates: Start with VoxelSpace's built-in lambdas for common tasks; modify themrather than writing from scratch

Queue awareness: Plan your processing and lambda jobs to avoid long queues, especially onthe free tier where concurrency is limited

Monitor usage: Track your compute usage to stay within plan limits

Debugging and Validation

Use logs: Include log statements in your lambda to help diagnose issues and verifyintermediate results

Validate inputs: Check that your data has the expected properties before processing

Test edge cases: Consider what happens with empty regions, missing properties, or extremevalues

Revision #1
Created 21 October 2025 17:47:30 by admin
Updated 21 October 2025 17:48:47 by admin