Distributed Sweep Management System

This system implements a robust and scalable solution for generating, uploading, and running Weights & Biases (W&B) sweeps across multiple clients. It consists of three main components: a server-side sweep generation and upload script, a client-side sweep execution script, and a cleanup process. The system utilizes W&B's artifact system to create a distributed locking mechanism, ensuring efficient and fault-tolerant sweep management.

Server-Side Sweep Generation and Upload

The server-side component is responsible for generating sweep configurations and uploading them to W&B. Key features include:

1. Generation of sweep configurations based on predefined parameters and hyperparameter ranges.
2. Tagging of sweeps with batch identifiers and status tags ('unclaimed', 'unfinished').
3. Inclusion of GPU resource requirements in sweep metadata.
4. Robust error handling and retry logic for sweep uploads.
5. A completion check mechanism that monitors the progress of all sweeps in a batch.

The `upload_sweeps` function handles the creation and uploading of sweeps, while the `run_completion_code` function allows for actions to be taken when all sweeps in a batch are finished.

Client-Side Sweep Execution

The client script finds unclaimed sweeps, claims them, and runs the associated experiments. Notable features include:

1. A distributed locking mechanism using W&B artifacts to claim sweeps.
2. GPU availability checking to ensure efficient resource utilization.
3. A heartbeat mechanism that continuously updates lock timestamps for long-running sweeps.
4. Comprehensive error handling with retry logic and exponential backoff for network operations.
5. Automatic release of locks in case of errors or completion of sweeps.

The `client_main` function orchestrates the process of finding, claiming, and running sweeps on each client machine.

Cleanup Process

An independent cleanup process identifies and releases stale locks, ensuring system resilience. Key aspects include:

1. Periodic checking of all sweeps in the project for stale locks.
2. Verification of sweep status before releasing locks to prevent interruption of active runs.
3. Handling of edge cases, such as sweeps marked as claimed but lacking a lock artifact.

The `cleanup_stale_locks` function is central to this process, carefully managing the release of stale locks while respecting active sweeps.

System Interaction

The three components work together to create a flexible and fault-tolerant sweep management system:

1. The server generates and uploads sweeps, making them available for execution.
2. Multiple clients can simultaneously search for and claim unclaimed sweeps, distributing the workload.
3. The cleanup process ensures that locks are released if clients crash or lose connection, allowing other clients to pick up unfinished sweeps.
4. The heartbeat mechanism and status checking prevent premature release of locks for long-running sweeps.

This system allows for efficient distribution of W&B sweeps across multiple machines, handles network issues and client crashes gracefully, and supports long-running sweeps without interruption. It's designed to be scalable, robust, and adaptable to various experimental setups and computational resources.