

Optimizing API Usage:

Adverity's strategies for Managing Quotas and Enhancing Efficiency

Modern data integration platforms rely heavily on APIs to extract data from various sources. However, API rate limits, quotas, and data constraints present significant challenges in ensuring seamless data ingestion and processing. To address these limitations, the Adverity platform employs a combination of strategic optimizations and intelligent request management techniques.

This document outlines several key approaches used within Adverity to maximize data retrieval efficiency while adhering to API constraints.

API Maintenance Continuity

- ✓ Regular audits ensure redundancy in API calls is minimized.
- ✓ High-investment, ongoing process to enhance connector reliability
- ✓ Breaking changes, ad-hoc change management

Adverity ensures API reliability through regular audits and proactive maintenance. By minimizing redundant calls, the platform optimizes request efficiency and reduces quota usage. Continuous investment in connector stability helps adapt to breaking changes, ensuring seamless data flow. An ad-hoc change management approach allows for quick updates when API providers modify endpoints or authentication methods, preventing disruptions in data retrieval.

Vendor Partnerships for Special Quotas

- ✓ Collaborate with API vendors for access to extended or exclusive quotas.
- ✓ Example: Partnership with IAS includes non-standard quota allocations.

Adverity collaborates with API vendors to secure extended or exclusive quotas, ensuring uninterrupted data access. By negotiating tailored agreements, the platform can bypass standard rate limits and enhance data throughput. For example, a partnership with IAS grants non-standard quota allocations, allowing for more efficient data retrieval. These partnerships help optimize API performance and maintain seamless integrations.

Intelligent Throttling Management

- ✓ Auto-Retry with "Exponential Backoff"
- ✓ Standard algorithm tailored per connector to find the optimal retry rate (e.g., per second, minute).
- ✓ Continuous improvement to expand auto-retry across connectors

Adverity optimizes API request handling through intelligent throttling and automated retries. Using exponential backoff, the system dynamically adjusts retry intervals to prevent exceeding rate limits. A tailored algorithm ensures each connector operates at the optimal retry rate, whether per second or minute. Continuous improvements enhance auto-retry efficiency across all connectors, maximizing data retrieval while minimizing failures.

Chunking

- ✓ Reduces API call volumes by batching requests efficiently without compromising data integrity.
- ✓ Examples: Pinterest and TikTok APIs utilize chunking to handle large datasets effectively.

Adverity enhances API efficiency by batching multiple requests into larger units, reducing overall call volume while maintaining data integrity. This technique is essential for APIs that impose strict rate limits or return large datasets. By leveraging chunking, platforms like Pinterest and TikTok process extensive data more effectively. Adverity applies similar strategies to optimize data retrieval, ensuring seamless integration without exceeding API quotas.

Filtering

- ✓ Uses hidden filters to eliminate redundant calls while ensuring necessary data output.
- ✓ Helps minimize request volume; clear documentation guarantees transparency in results.

To minimize unnecessary API calls, Adverity employs hidden filters that eliminate redundant data retrieval while maintaining accuracy. By refining queries before they reach the API, filtering reduces request volume, improving efficiency without affecting results. Additionally, clear documentation ensures users understand filtering logic, maintaining transparency in the data output.

Sliding Windows

- ✓ Manages API payloads using date-based filtering for more precise data retrieval.
- ✓ Smart Scheduling: Automatically spreads API requests over time to avoid rate-limit thresholds (e.g., per second, per day).

Adverity applies sliding window techniques to segment data retrieval over time, preventing large payloads from overwhelming an API. This method is particularly useful for APIs with strict per-second or per-day rate limits. Smart Scheduling further optimizes this approach by strategically distributing requests, ensuring consistent data flow without exceeding quota constraints.

Poll for Reports

- ✓ Leverages vendor-generated reports to reduce direct API calls.
- ✓ Example: Previous SA360 connector utilized this approach; current connectors compute multiple endpoints for improved performance.

Instead of relying on frequent direct API requests, Adverity utilizes vendor-generated reports where available, significantly reducing API load. This approach improves efficiency while maintaining data completeness. Previously, the SA360 connector used this strategy to minimize API interactions, and modern connectors now enhance it by computing data across multiple endpoints for optimized performance.

Conclusion

Adverity's approach to API management ensures seamless data integration, even when working within strict rate limits. By combining intelligent throttling, optimized request handling, and proactive maintenance, the platform maximizes data retrieval while minimizing failures and disruptions.

These strategies provide greater efficiency, reliability, and scalability, allowing users to access data without worrying about rate limits or API changes. Advanced techniques like smart scheduling and vendor partnerships give Adverity a distinct advantage, ensuring uninterrupted data flow while optimizing performance.

By automating retries, reducing redundant calls, and leveraging vendor-generated reports, Adverity helps businesses save time, reduce API costs, and maintain data accuracy. These solutions make Adverity a highly resilient and future-proof platform for data integration.

