API Versioning and Deprecation Policy

Overview

This document outlines our guidelines for managing API versions and the deprecation process. The goal is to ensure backwards compatibility while allowing the API to evolve over time and giving developers sufficient time to migrate.

Versioning

Version Format

API versions follow the format: `v{MAJOR}` (e.g., `v1`, `v2`).

Versioning in Endpoints

Versioning is indicated in the URL path.

```
GET /v1/users/{id}
```

Version Lifecycle

- **Current Version:** Actively supported and maintained
- **Previous Versions:** Supported for deprecation period
- **Deprecated Versions:** Scheduled for removal

Deprecation Policy

When a version is deprecated, users will be notified at least **6 months** in advance before its removal.

Deprecation Timeline Example

Event	Description
Deprecation Notice	API version marked as deprecated; announcement sent to users
Grace Period	Deprecated API remains functional for at least 6 months
End of Life	Deprecated API version is removed and no longer accessible

Change Guidelines

- Non-breaking changes (e.g., adding optional parameters) may be introduced within an API version.
- Breaking changes (e.g., removing fields, changing response formats) require a new major version.

Best Practices for Clients

- Always specify the API version in requests.
- Monitor deprecation announcements.
- Plan updates to support new versions before old versions are removed.

Contact

For questions regarding API versioning or deprecation, contact our support team at `api-support@example.com`