CDC SHARE IT Act Automation – Execution Architecture

# ✅ Solution Execution Model

The automation is designed to run from a secure server (internal or cloud-hosted) and use remote API access to collect metadata from multiple code repository systems maintained at CDC.

# 🔁 How It Works

1. Scheduled or Triggered Execution:  
 - A script, cron job, or CI/CD pipeline runs periodically (e.g., daily or weekly).  
  
2. Environment Loads Config & Credentials:  
 - Loads `.env` and `config.yaml` at runtime for API tokens and settings.  
  
3. Each Client Connects via API:  
 - GitHub, GitLab, Bitbucket, and TFS are queried using their official REST APIs.  
  
4. Metadata Normalization:  
 - Data is extracted and transformed into a common structure (name, URL, usageType, etc.).  
  
5. code.json Generation:  
 - All repository metadata is aggregated into a single `code.json` per the code.gov schema.  
  
6. Publishing:  
 - The resulting file is copied to the public-facing web server at CDC (e.g., `/var/www/html/code.json`).

# 🛡️ Security & Network Assumptions

- The execution server must have network access to the GitHub, GitLab, Bitbucket, and TFS endpoints.  
- Authentication is done using Personal Access Tokens (PATs), loaded securely from environment variables.  
- The solution requires read-only API access and never modifies source code.  
- Secrets and credentials are not stored in code — `.env` and Azure DevOps secrets are used.

# 🧩 Suggested Execution Environments

|  |  |
| --- | --- |
| Environment | Benefit |
| Internal CDC Linux Server | Secure, controlled, direct network access to on-prem systems. |
| Azure-hosted VM | Good for cloud-based scheduling and integration with Azure DevOps. |
| GitHub Actions or Bitbucket Pipelines | Only feasible if repositories are publicly accessible or credentials are scoped for API access across systems. |