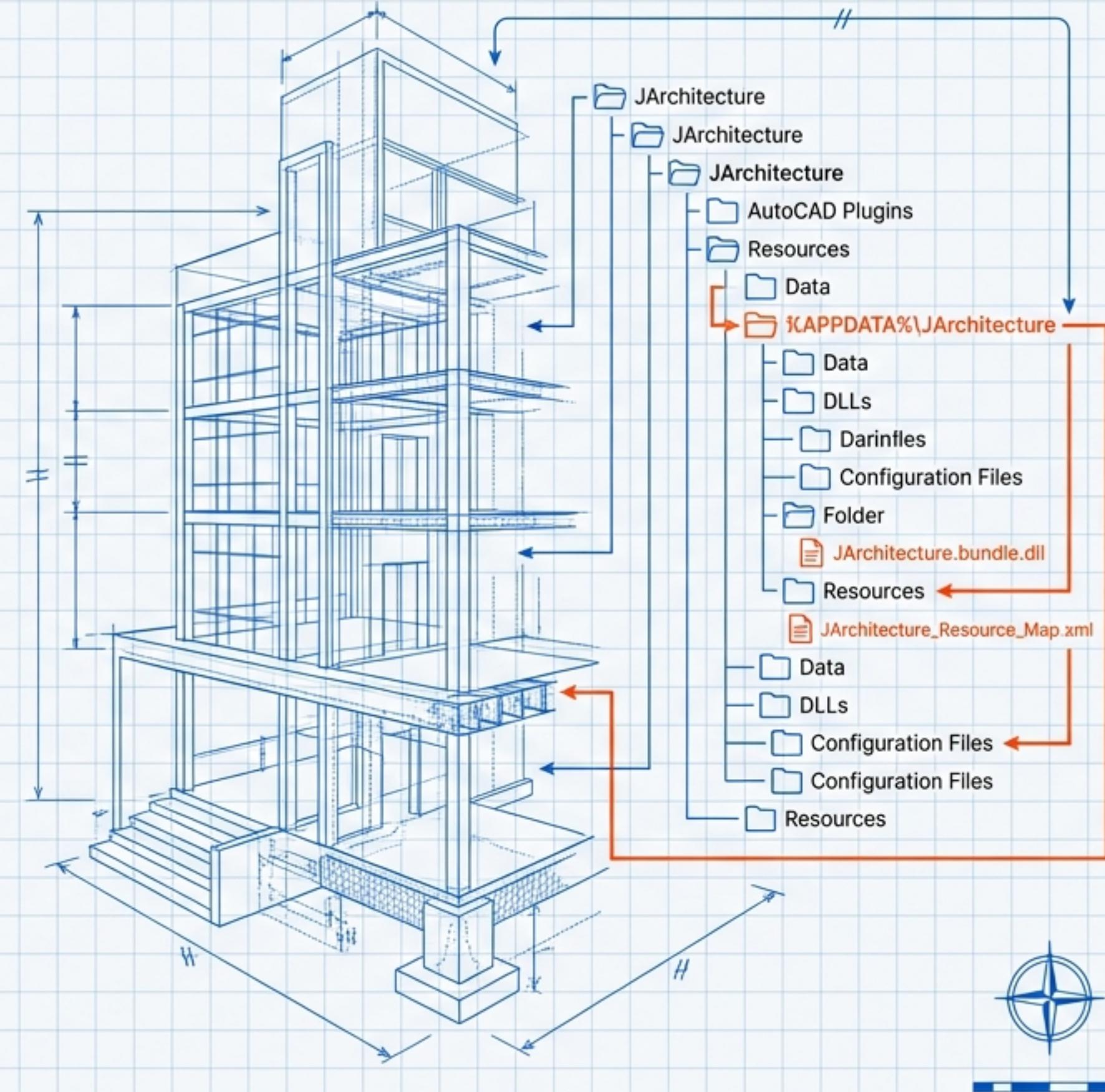


AutoCAD Addin Bundle: 설치 구조 및 리소스 접근 가이드

JArchitecture 배포를 위한
기술적 청사진 (Technical Blueprint)

본 문서는 AutoCAD 환경에서 JArchitecture 플러그인의
설치 경로, 파일 구조, 그리고 런타임 리소스 접근 방식을
다루는 기술 명세서입니다.

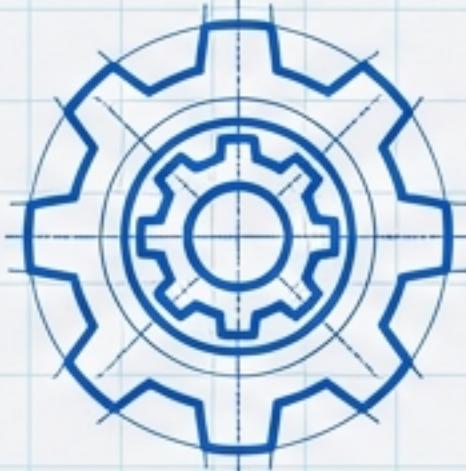


핵심 요약 (Executive Summary)



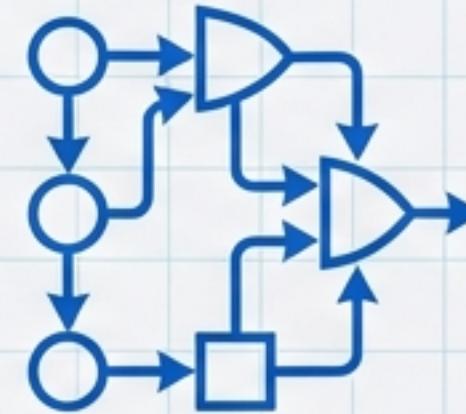
설치 타겟 (Target)

모든 사용자 공용 경로
(Public/Common Path).



구현 방식 (Mechanism)

Inno Setup의
{commonappdata}
변수 활용.



런타임 로직 (Runtime)

Reflection을 이용한
동적 경로 탐색
(하드코딩 배제).

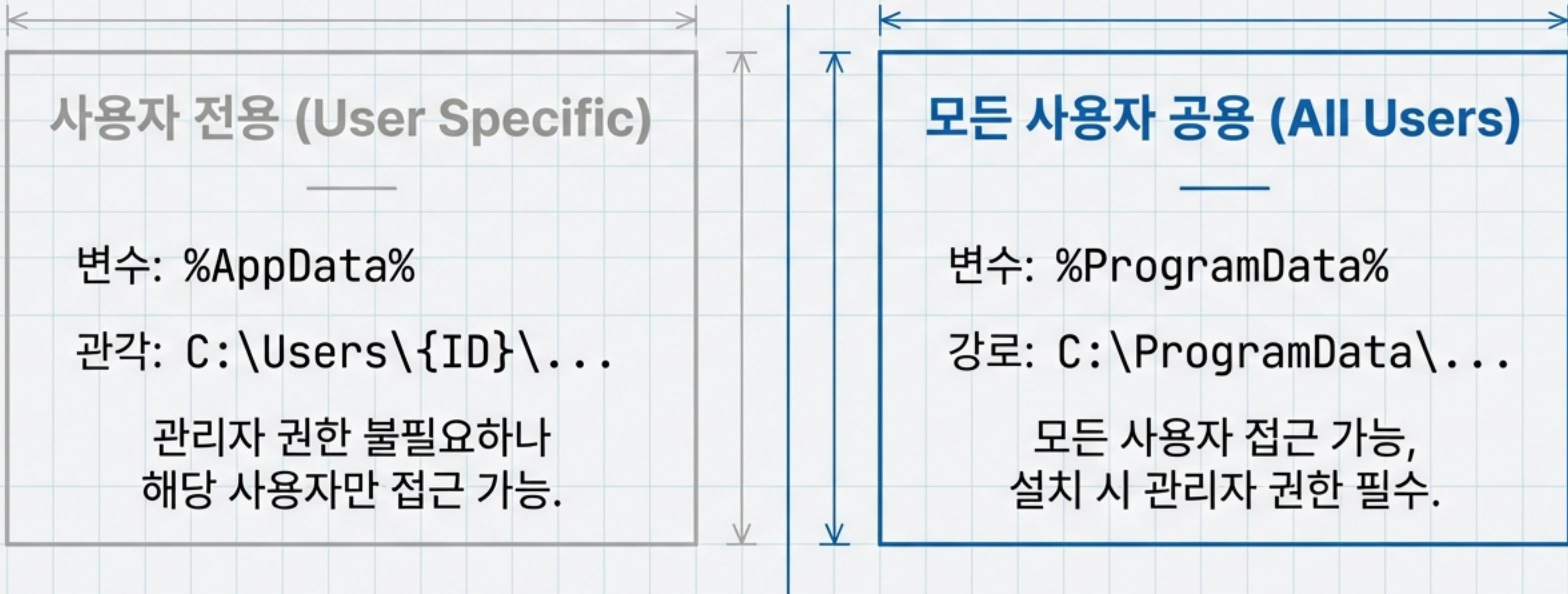


결과 (Result)

관리자 권한 기반의
안정적인 멀티 유저 지원.

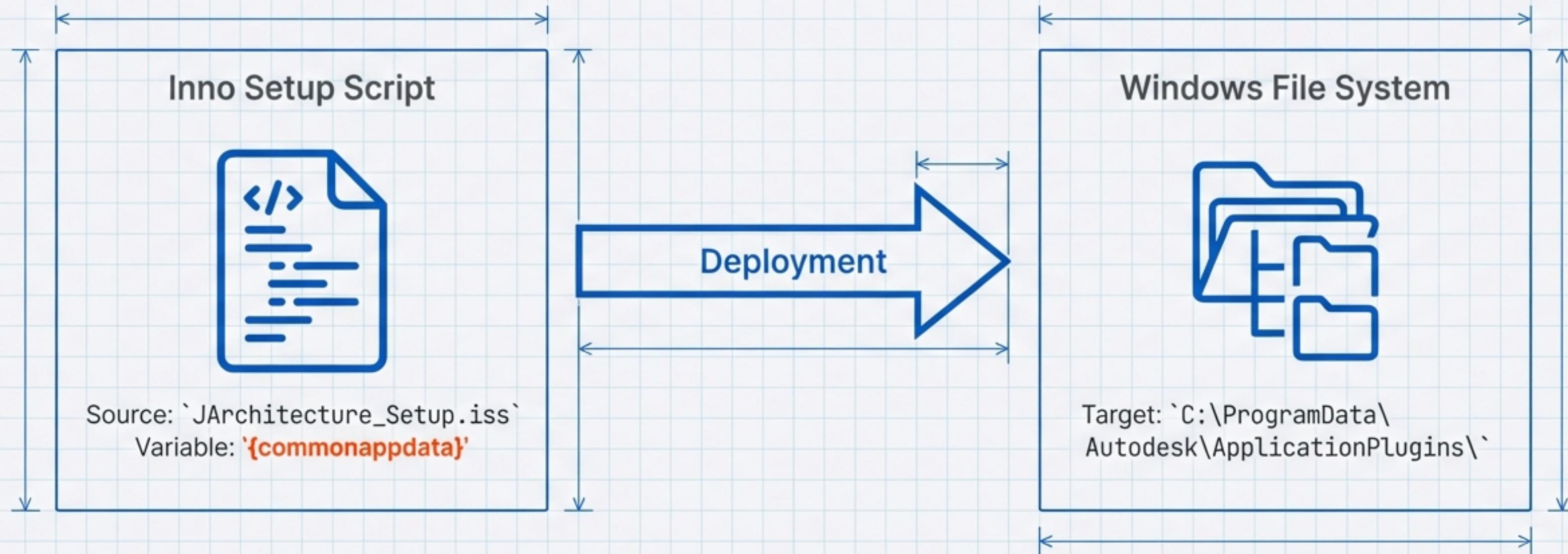
JArchitecture는 개별 사용자(%AppData%)가 아닌 시스템 전체(%ProgramData%)에 설치되어,
모든 사용자가 일관된 환경에서 플러그인을 사용할 수 있도록 설계되었습니다.

AutoCAD Bundle 경로 표준: 사용자 전용 vs 공용



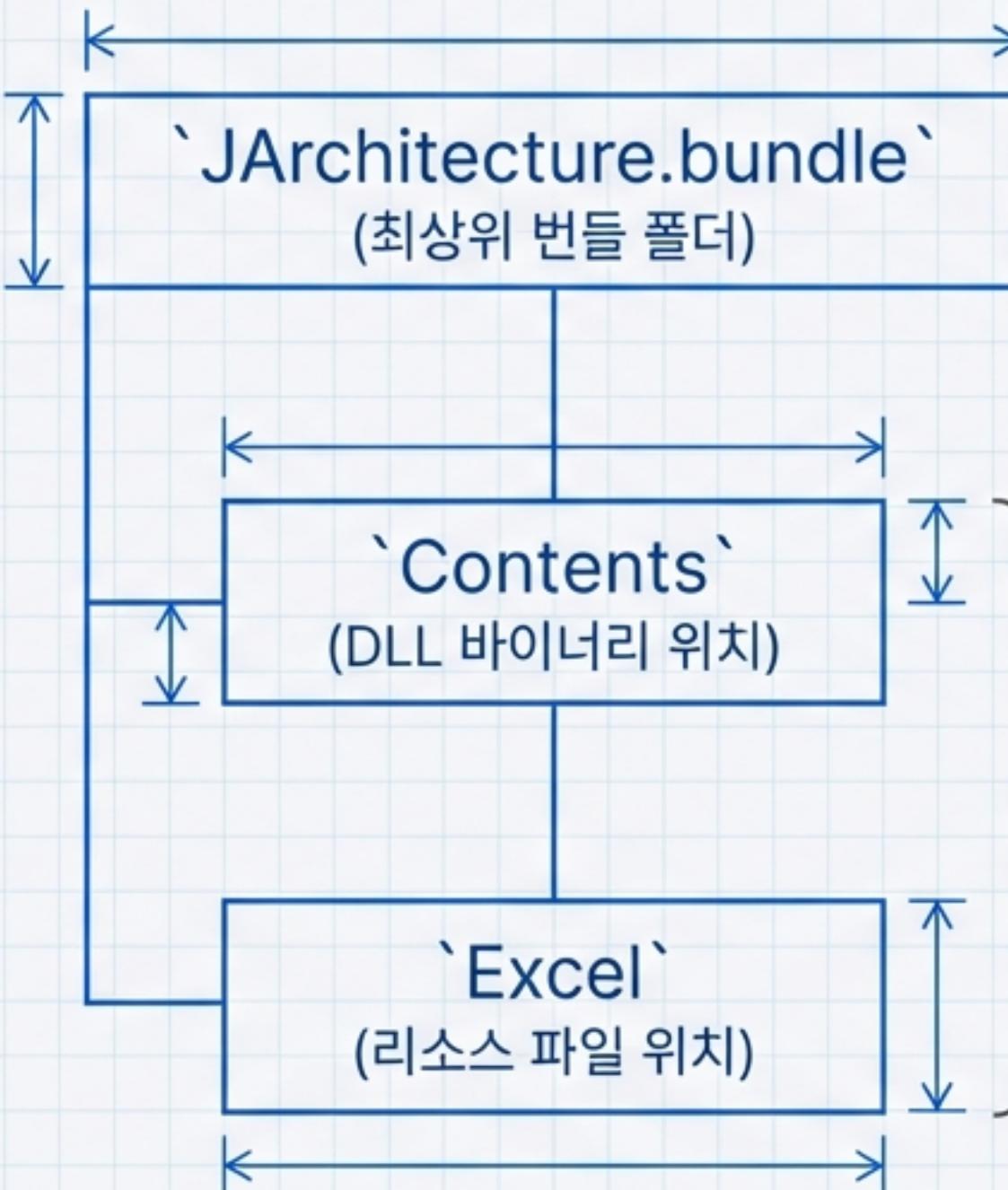
Critical Note: PowerShell 사용 시 '%AppData%' 구문은 동작하지 않습니다.
반드시 '\$env:AppData' 또는 '\$env:ProgramData' 형식을 사용해야 합니다.

JArchitecture 배포 전략 (Deployment Strategy)



JArchitecture는 다중 사용자 환경 지원을 위해 **{commonappdata}** 변수를 사용하여 공용 경로인 **ProgramData**에 번들을 배포합니다.

번들 폴더 구조 (Bundle Anatomy)



DLL 파일은 'Contents' 폴더에 위치하며,
참조용 Excel 리소스는 'Contents\Excel'
하위 폴더에 격리되어 관리됩니다.

ISS 변수와 실제 경로 매핑 (Mapping Table)

ISS 변수 (Variable)	실제 경로 (Physical Path)
{app}	C:\ProgramData\...\JArchitecture.bundle
{app}\Contents	...\JArchitecture.bundle\Contents\
{app}\Contents\Excel	...\JArchitecture.bundle\Contents\Excel\

{app} 변수는 번들의 루트 디렉토리를 가리키며, 모든 하위 리소스는 이 경로를 기준으로 상대적으로 배치됩니다.

런타임 경로 접근의 과제



절대 경로(Hardcoding)의 위험성

- 사용자마다 드라이브 문자나 사용자명(Users\{ID})이 다를 수 있음.
- 설치 경로가 %AppData%에서 %ProgramData%로 변경될 경우 코드가 깨짐.

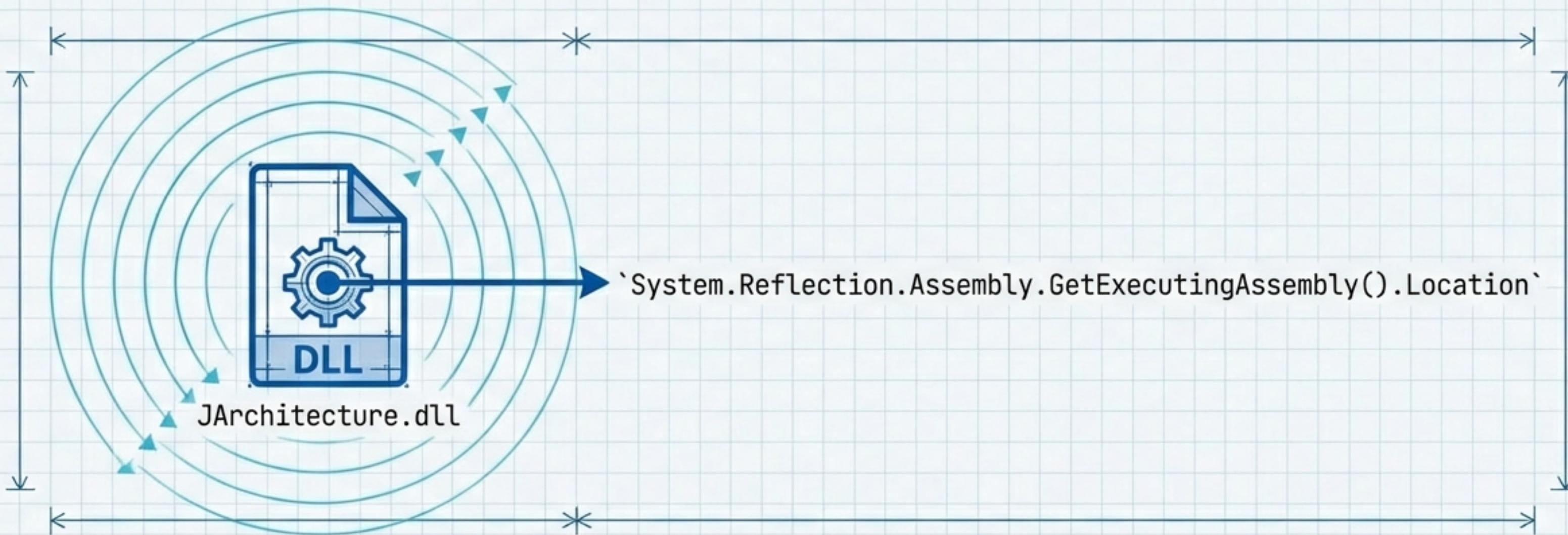


실행 중인 어셈블리(DLL)의 위치를 기준으로 한 '상대 경로' 계산 방식이 필요합니다.

솔루션 원리를 실행 중인 어셈블리(DLL)의 위치를 기준으로 암시적 경로하는 시작에 다치 코로가 방과도 미Win32 API 어셈블리택을 겸장을 인으로고 '상대야 풀'e' 계산을 필요합니다.



해결책: System.Reflection 활용



이 메서드는 현재 실행 중인 DLL의 **실제 로드 경로**를 반환합니다. 설치 위치가 사용자 전용이든 공용이든 관계없이 항상 올바른 기준 경로를 제공합니다.

C# 코드 구현 (Implementation)

```
// 1. 현재 실행 중인 DLL의 경로 획득  
string assemblyPath = System.Reflection.Assembly.GetExecutingAssembly().Location;  
  
// 2. DLL이 위치한 폴더(Contents) 경로 추출  
string contentsPath = Path.GetDirectoryName(assemblyPath);  
  
// 3. Excel 리소스 폴더 결합 (상대 경로)  
string excelPath = Path.Combine(contentsPath, "Excel");
```

Best Practice

문자열 단순 결합 대신
Path.Combine을 사용하여
OS별 경로 구분자(\ 또는 /)
오류를 방지해야 합니다.

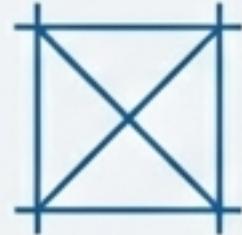
설치 검증 (Verification via CLI)

```
PS C:\Users\Admin>  
dir "$env:ProgramData\Autodesk\ApplicationPlugins" # JArchitecture.bundle 폴더 존재 확인
```

```
PS C:\Users\Admin>  
dir "$env:ProgramData\Autodesk\ApplicationPlugins\JArchitecture.bundle\Contents\Excel"  
# 리소스 파일의 정상 배포 확인
```

참고: CMD가 아닌 PowerShell 환경 기준 명령어입니다.

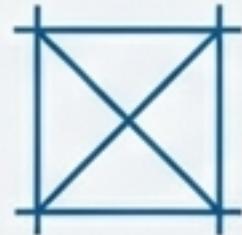
주요 주의사항 (Critical Checkpoints)



권한 (Admin Rights)

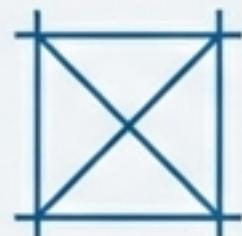
'ProgramData' 경로는 쓰기 작업 시 관리자 권한이 필수입니다.

(Inno Setup: `PrivilegesRequired=admin`)



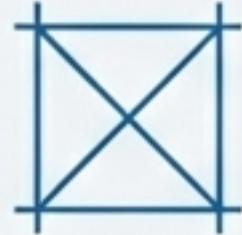
구문 (Syntax)

PowerShell에서는 `%ProgramData%` 대신 `$env:ProgramData`를 사용해야 합니다.



무결성 (Integrity)

새로운 Excel 파일을 추가할 경우, 반드시 ISS 스크립트의 `[Files]` 섹션도 함께 업데이트해야 합니다.



경로 (Path)

DLL은 `Contents`에, 리소스는 하위 폴더에 격리하는 구조를 준수하십시오.

최종 기술 요약 (Architecture Review)

Path: "Common / Public (`C:\ProgramData`)"

Installer Variable: "{commonappdata}"

Access Method:

`Assembly.GetExecutingAssembly().Location` + `Path.Combine`



이 아키텍처는 사용자 환경에 구애받지 않는 일관성(Consistency)과 확장성(Scalability)을 보장합니다.