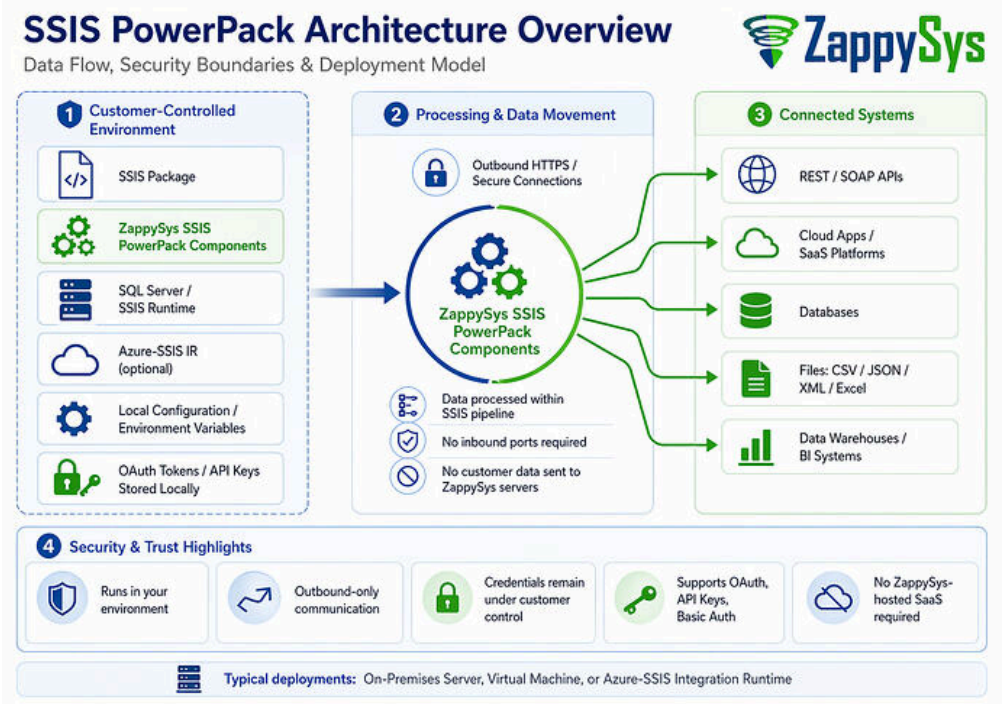
ZappySys Architecture Overview (SSIS PowerPack & ODBC PowerPack)

This document provides a high-level architecture overview of **ZappySys SSIS PowerPack** and **ODBC PowerPack**, focusing on **data flow, security boundaries, and deployment model** for compliance and security review purposes.

SSIS PowerPack Architecture



Key Components

- **SSIS Package (SQL Server Integration Services)**
  Runs inside your environment (on-premises VM or Azure-SSIS IR).
- **ZappySys SSIS PowerPack Components**
  Includes connectors for REST APIs, JSON/XML, cloud apps, and databases.

Skip to main content

- **Secure Credential Handling**

    - Credentials stored in SSIS configurations / environment variables
    - OAuth tokens stored locally within the execution environment
    - No credential storage on ZappySys servers

- **External Systems**
    - Cloud APIs (REST / SOAP)
    - SaaS platforms (e.g., CRM, Helpdesk, Marketing tools)
    - Databases (SQL Server, Oracle, etc.)
    - Files (CSV, JSON, XML, Excel)

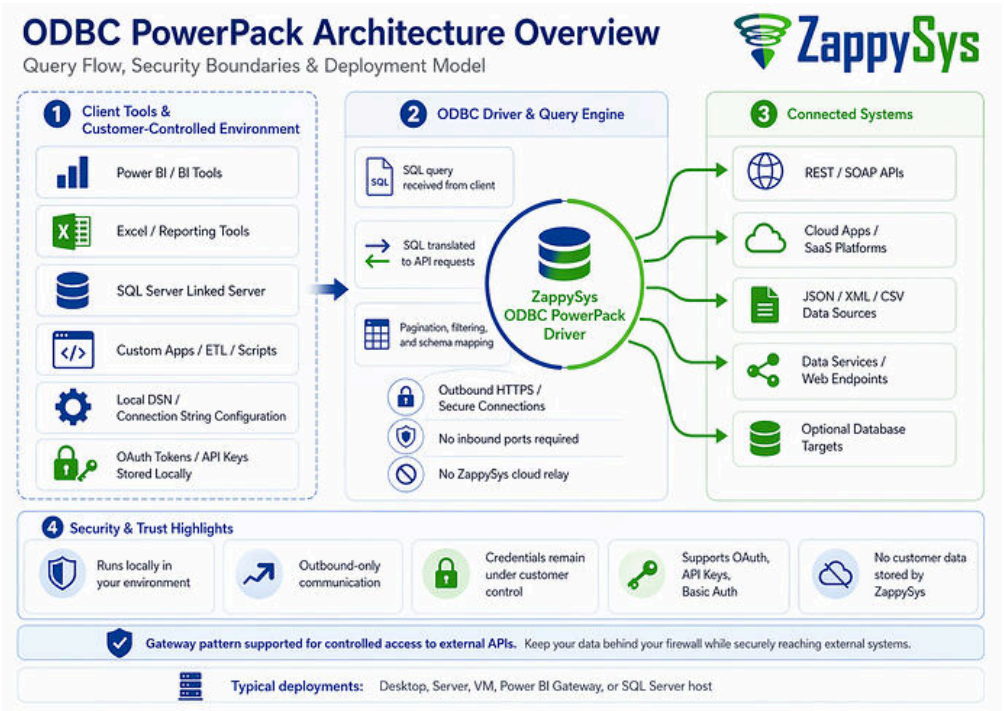
Data Flow Summary

1. SSIS package executes inside customer-controlled environment
2. ZappySys connectors make outbound API/database calls
3. Data is processed in-memory within SSIS pipeline
4. Data is written to destination systems (DB, file, warehouse)

Security Notes

- ✅ **No inbound connections required**
- ✅ **All communication is outbound (HTTPS)**
- ✅ **Data stays within customer-controlled infrastructure**
- ✅ **Supports secure token-based authentication (OAuth, API keys, etc.)**

ODBC PowerPack Architecture



Key Components

- **ODBC Driver (ZappySys API Driver)**
  Installed on client machine or server.
- **Client Applications**
  - Power BI (DirectQuery / Import)
  - Excel
  - SQL Server Linked Server
  - Any ODBC-compliant tool
- **ZappySys Query Engine**
  - Translates SQL queries into API requests
  - Handles pagination, filtering, joins (where supported)
- **External Systems**
  - REST APIs / SaaS platforms
  - Cloud data services

Data Flow Summary

Skip to main content SQL query via ODBC
2. ZappySys driver translates SQL → API calls
3. Data fetched via HTTPS from external systems

4. Results streamed back to client tool

Security Notes

- ✅ **Driver runs locally (no cloud dependency)**
- ✅ **No data stored or persisted by ZappySys**
- ✅ **All API calls are outbound over HTTPS**
- ✅ **Credentials stored locally (DSN or connection string)**

Data Gateway & Linked Server Security Overview

In some enterprise scenarios, a **Data Gateway pattern** is used to securely bridge internal systems with external APIs or cloud services. This is common when using:

- SQL Server **Linked Server (via ODBC Driver)**
- On-premise reporting tools (Power BI, Excel, SSRS)
- Restricted environments behind firewalls

🧩 How It Works

- The **ODBC Driver (ZappySys PowerPack)** is installed inside your secure network
- All API calls are executed **locally from within your environment**
- External systems are accessed via **outbound HTTPS only**
- No inbound ports or external callbacks are required

🛡️ Security Characteristics

- ✅ Runs entirely **inside your network boundary**
- ✅ **No data persistence** outside your system
- ✅ **No middleware or cloud relay by ZappySys**
- ✅ Works with existing **firewalls, proxies, and network controls**
- ✅ Credentials remain **local (DSN / secure config / vault integration)**

⚠️ Special Note (SQL Server Linked Server Stability)

When using ODBC drivers with SQL Server Linked Server, there are known stability considerations depending on driver configuration and environment.

We strongly recommend reviewing the following guide for best practices, fixes, and secure

Skip to main content

👉 **Read More:**
https://zappysys.com/blog/sql-server-linked-server-odbc-crash-fix/

This guide covers:

- Common crash scenarios and root causes
- Secure and stable configuration patterns
- Best practices for production deployments

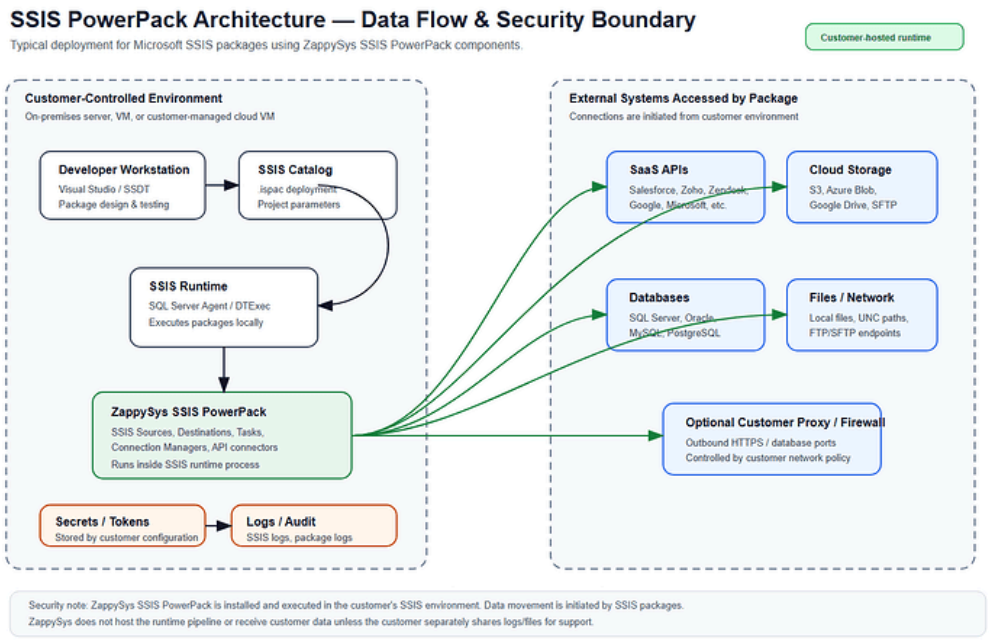
📌 When to Use This Pattern

Use Data Gateway / Linked Server architecture when:

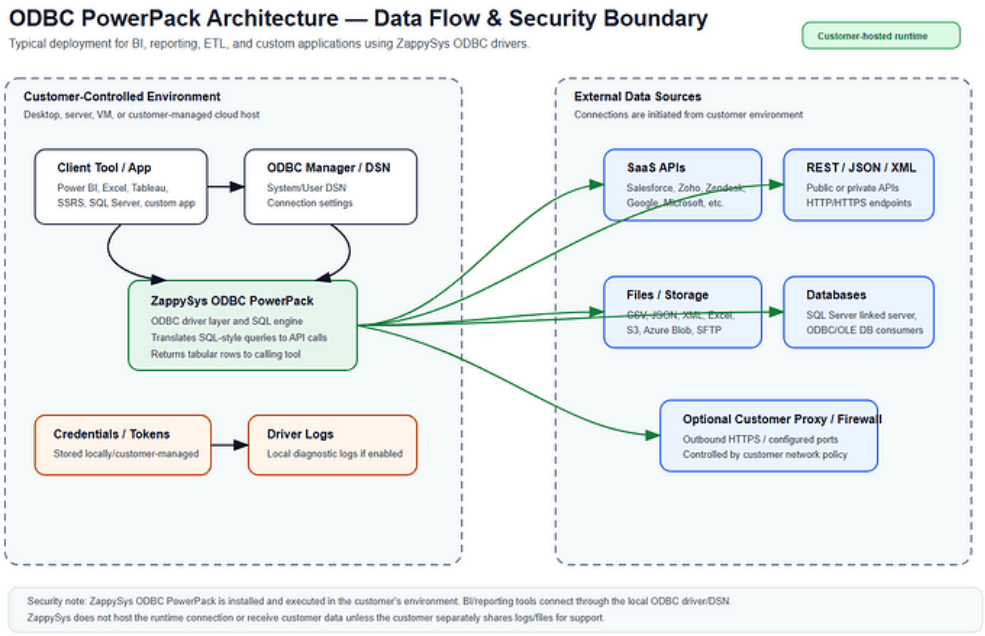
- Direct cloud connectivity is restricted
- You need **centralized access via SQL Server**
- Multiple users/tools must query external APIs through a controlled layer
- Security policies require **no direct client-to-API access**

Additional Security Boundary Diagrams

The diagrams below provide a simplified security-boundary-only view for compliance reviewers.



Skip to main content



Skip to main content