

GUIDO DIRR

SENIOR SOFTWARE ENGINEER

BASEL - SWITZERLAND

KEY AREAS OF EXPERTISE

INTERFACE PROGRAMMER

Integration of ServiceNow, Remedy, and SAP via REST/SOAP APIs, **SQL**, and Kafka Data Streaming. Development of interfaces to **automate** and ensure data transmission through monitoring and alerting systems.

C# DEVELOPMENT

Development of complex **multi-layer** applications in **C#** with a focus on **.NET Core** web and **API** development. Implementation of modern DevOps principles such as CI/CD pipelines and workflow automation.

MS SQL SERVER SPECIALIST

Optimization of MS SQL Server environments through query tuning, index optimization, and data process automation. Development of **high-performance** database **solutions** for analysis and reporting.

FULL STACK WEB DEVELOPMENT

Development of scalable web applications using **ASP.NET** and **Angular**. Focus on performance, user-friendliness, and modern architectures like MVC and Microservices.

AI ENTHUSIAST

Certified Microsoft Azure **AI Solution Designer**. Development of innovative AI-based solutions using **Python** and **C#**. **Prompt Engineering** professional.

PROJECTS OVERVIEW

Guido Dirr - Architecture and Implementation

SCHEDULER

WindowsService.NET / ASP.NET / WinForms

TRA – TRANSFER APPLICATION

WinService C# / ANGULAR

ASSET & INVENTORY SOLUTION

ASP.NET / ANGULAR / C# Agents / .NET API / SQL

KAFKA CONSUMER AND PRODUCER

Python / .NET API / SQL

SELENIUM ROBOTS

WinForms C# / Python / Selenium

SHAREAPP

WinForms / .NET API / SignalR

APIMAN

C# Agent / SQL / JSON / XML

DATABASE VARIANCE TOOL

SQL / ETL / ASP.NET

DEPLOYER

C# Agent / SQL

SCHEDULER

WindowsService.NET / ASP.NET / WinForms

Asynchronous Task Execution

Execute Windows commands, PowerShell scripts, SQL stored procedures, and specialized tools (e.g., data loads, report rendering) across multi-server environments.

Multi-Server Task Coordination

Sequentially execute tasks, e.g., run Command 1 on Server X, then Command 2 on Server Y.

Dependency System

Ensure proper workflow by triggering reports only after successful data loads.

Self-Healing Automation

Automatically resolve the entire task chain when a failed data load is successfully retried.

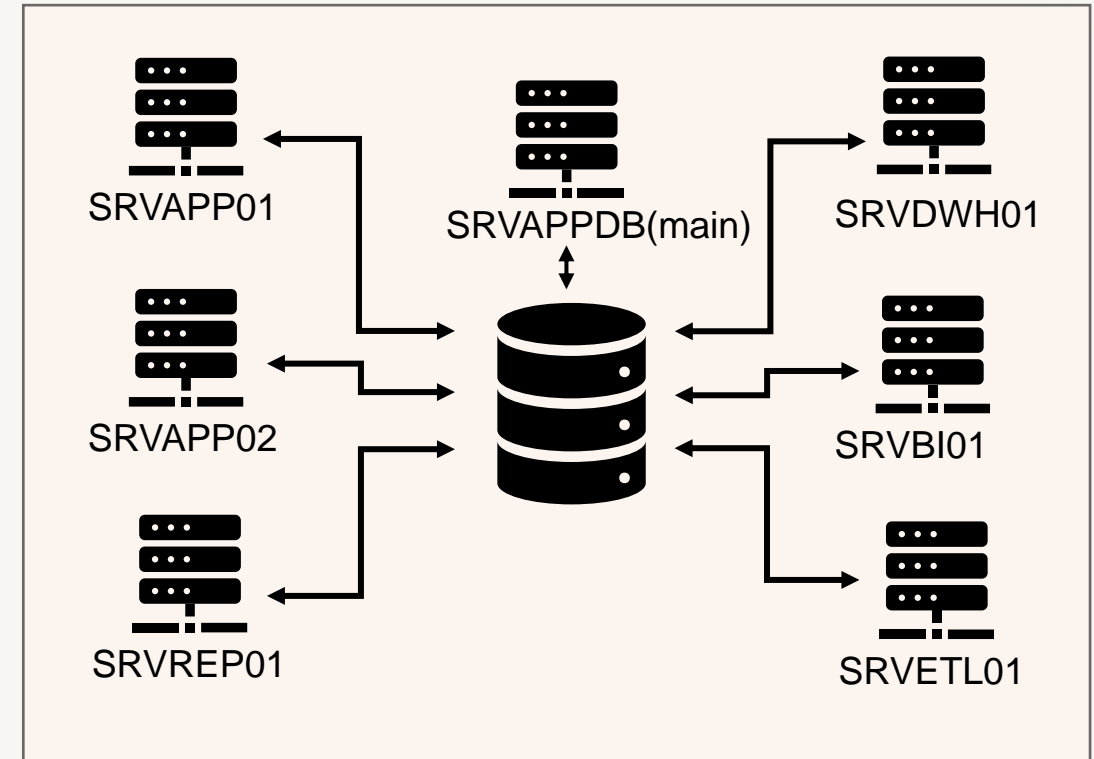
Comprehensive Monitoring & Assignment

Web-based dashboard to search, filter, resolve, and edit tasks. Add comments, mass-resolve issues, and assign tasks to teams or individuals.

SCHEDULER

The screenshot shows the 'Edit Command' window for a task named 'ListCustomerSwitchAccountClosing'. The command is a Stored Procedure with the name 'ListCustomerSwitchAccountClosing'. The description states: 'check account closed dates of list customer switch. If closeddate is in the past the accountclosed flag goes to Y.' The sequence number is 2, and the server is 'SOSM28SNE.SNL.CH.IBM.COM'. The database name is 'AIS' and the stored procedure name is '[dbo].[P_ListCustomerSwitchAccountClosing]'. The 'Schedule History' grid below shows a list of tasks with columns for Action, Project, Task Name, Schedule, End Time, Status, Result, and Rebad. The grid is filtered for 'Successful Tasks' and shows a list of tasks with their respective schedules and results.

| Action | Project | Task Name | Schedule | End Time | Status | Result | Rebad |
|----------------------------|--|-------------|-------------|-----------|---------|-------------|-------|
| Q Dataloads | ERT_Filesare_Upload_ISM_Person_Data_SAP_COSTCENTER | 09-01 06:32 | 09-01 06:32 | Completed | Success | Action menu | |
| Q Dataloads | Job_SMY_Weekly_Monday_0815 | 09-01 06:32 | 09-01 06:34 | Completed | Success | Action menu | |
| Q Dataloads | IPCA_MAINFRAME_MEUSI_MONTHLY_EARLY_MASTER | 09-01 06:32 | 09-01 06:38 | Completed | Success | Action menu | |
| Q Dataloads | IPCA_MAINFRAME_MEUSI_FULLOAD_DAILY_MASTER | 09-01 06:32 | 09-01 06:33 | Completed | Success | Action menu | |
| Q Dataloads | IPCA_MAINFRAME_MEUSI_DELTALOAD_DAILY_MASTER | 09-01 06:32 | 09-01 06:39 | Completed | Success | Action menu | |
| Q Dataloads | GRF_Gdy_Loads | 09-01 06:32 | 09-01 06:46 | Completed | Success | Action menu | |
| Q Standard Reporting | ETL_XXX_ST2_Storage | 09-01 06:32 | 09-01 06:32 | Completed | Success | Action menu | |
| Q Shared_Layer | Check Common Storage Layer | 09-01 06:32 | 09-01 06:32 | Completed | Success | Action menu | |
| Q Dataloads | DB20_PROD_DELTALOAD_DAILY_MASTER | 09-01 06:01 | 09-01 08:36 | Completed | Success | Action menu | |
| Q ISM Load for SRT Persons | ISM_Load_for_SRT_Persons | 08-01 06:31 | 08-01 06:32 | Completed | Success | Action menu | |
| Q Dataloads | IPCA_MAINFRAME_MEUSI_0722_FULLOAD_MONTHLY_EARLY_MASTER | 08-01 06:31 | 08-01 06:32 | Completed | Success | Action menu | |
| Q Dataloads | IPCA_MAINFRAME_MEUSI_FULLOAD_WEEKLY_MASTER | 06-01 06:32 | 06-01 06:32 | Completed | Success | Action menu | |
| Q Dataloads | Julius Monthly Load | 03-01 06:31 | 03-01 06:32 | Completed | Success | Action menu | |
| Q Dataloads | Unload_BackupAndRecovery_Reporting_Monthly | 03-01 06:31 | 03-01 06:32 | Completed | Success | Action menu | |
| Q Standard Reporting | ReportSend_XXX_Monthly_21st | 24-12 06:31 | 24-12 06:32 | Completed | Success | Action menu | |
| Q HEN Reporting | HEN_Reporting_Monthly_17th_1100 | 20-12 08:38 | 20-12 08:38 | Completed | Success | Action menu | |
| Q HEN Reporting | HEN_Neues_Report_Monthly_18th_1000 | 20-12 08:38 | 20-12 08:38 | Completed | Success | Action menu | |
| Q ATL Reporting | ReportSend_ATL_Monthly_10th | 15-12 17:33 | 15-12 17:37 | Completed | Success | Action menu | |
| Q DWH Tasks | soem26_IAM_PerformanceOptimization | 09-01 14:33 | 09-01 14:33 | Completed | Success | Action menu | |
| Q Dataloads | EXPORT_RMKCOMP_MKT_OVERVIEW_SCHUTZERAND_ALL_SEB | 09-01 14:32 | 09-01 14:32 | Completed | Success | Action menu | |



TRA – TRANSFER APPLICATION

WinService C# / ANGULAR

Automated Data Transfers

Built with C# for seamless data exchange between customer and internal isolated networks.

Efficient File Handling

Files, regardless of size, are chunked, encrypted, and transferred securely.

The receiving TRA instance reassembles the file based on metadata and triggers follow-up tasks if configured.

Flexible Protocols

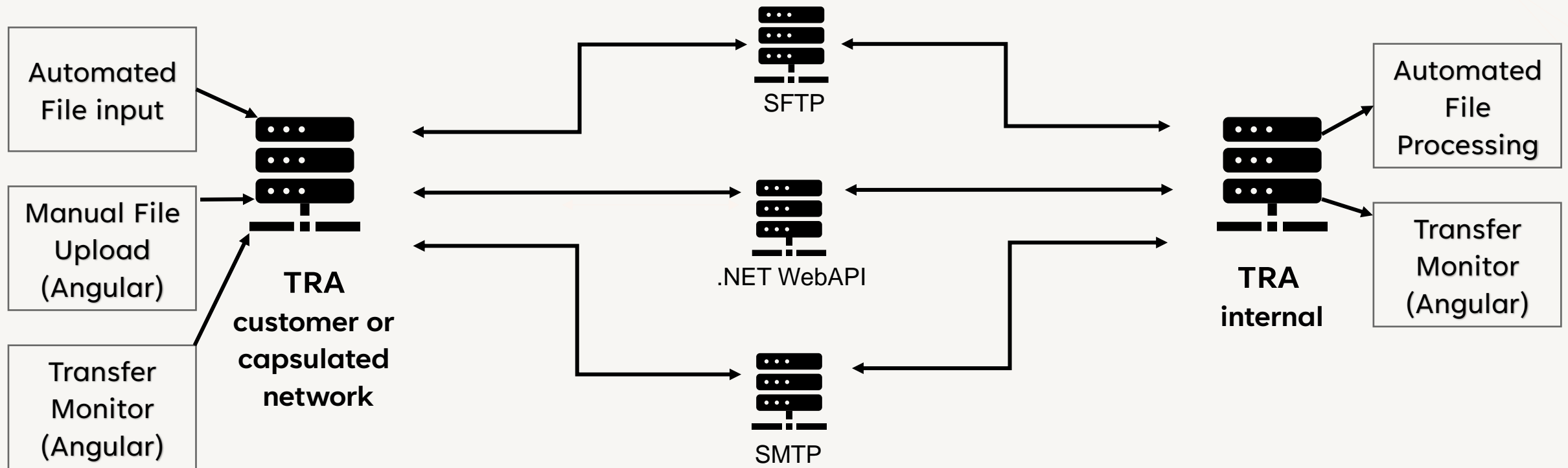
Supports SFTP, WebAPI, and SMTP for versatile file transfer.

User-Friendly Monitoring

Angular-based UI for real-time monitoring of file transfers.

Manual file uploads can be initiated via the Angular app and processed by the TRA service.

TRA – TRANSFER APPLICATION

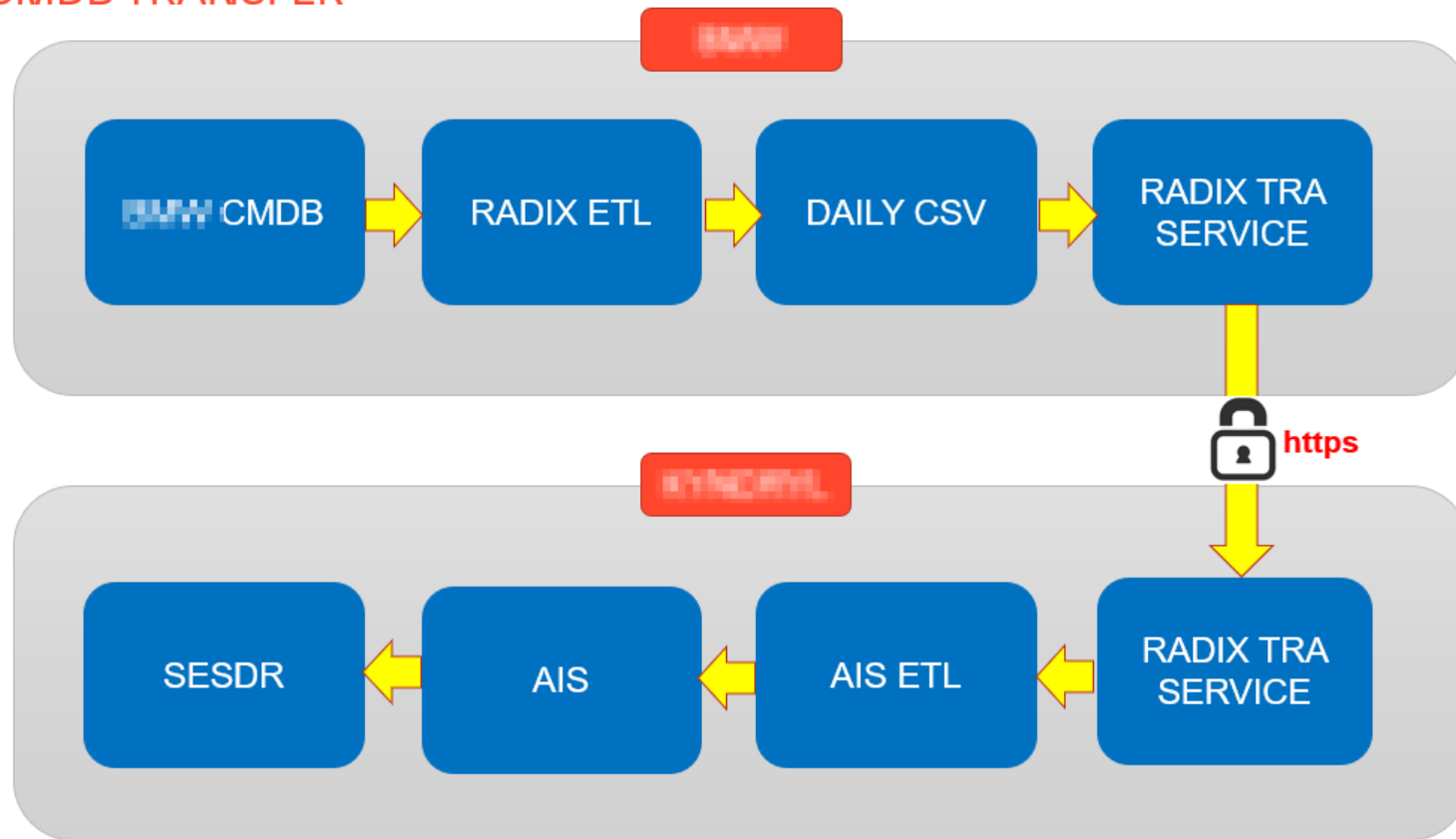


TRA EXAMPLE

-CUSTOMER CMDB PROCESSING -

AIS

CMDB TRANSFER



Radix ETL automatically fetches the newest CMDB data daily, creates a CSV file, hands it over to the TRA service, it is picked up and handed over to the internal TRA instance over the internet, when the transfer is done, a task is automatically triggered to process the file and import it into AIS where it is enriched from other systems and rules and from there automatically transferred to SESDR system.

AIS – ASSET & INVENTORY SOLUTION

ASP.NET / ANGULAR / C# Agents / .NET API / SQL

Customer Management

Manages over 300 customers with robust integration capabilities for seamless data inflow and outflow.

Modern User Interface

Dynamic ASP.NET DevExpress UI as well as Angular components for a responsive and user-friendly experience.

Advanced Data Validation

Rule-based Data Validation Engine enabling automatic corrections or manual interventions through team assignments.

Flexible System Integration

.NET Web API for smooth interaction with systems like Ansible, supporting both data producers and consumers.

Dynamic Data Loading

Fully automated XLS/X-to-SQL web mass loader with ASP.NET, complete with post-load procedure triggers.

Automated Console Agents

Scheduled C# Console Agents for efficient data integration and validation processes.

AIS – ASSET & INVENTORY SOLUTION

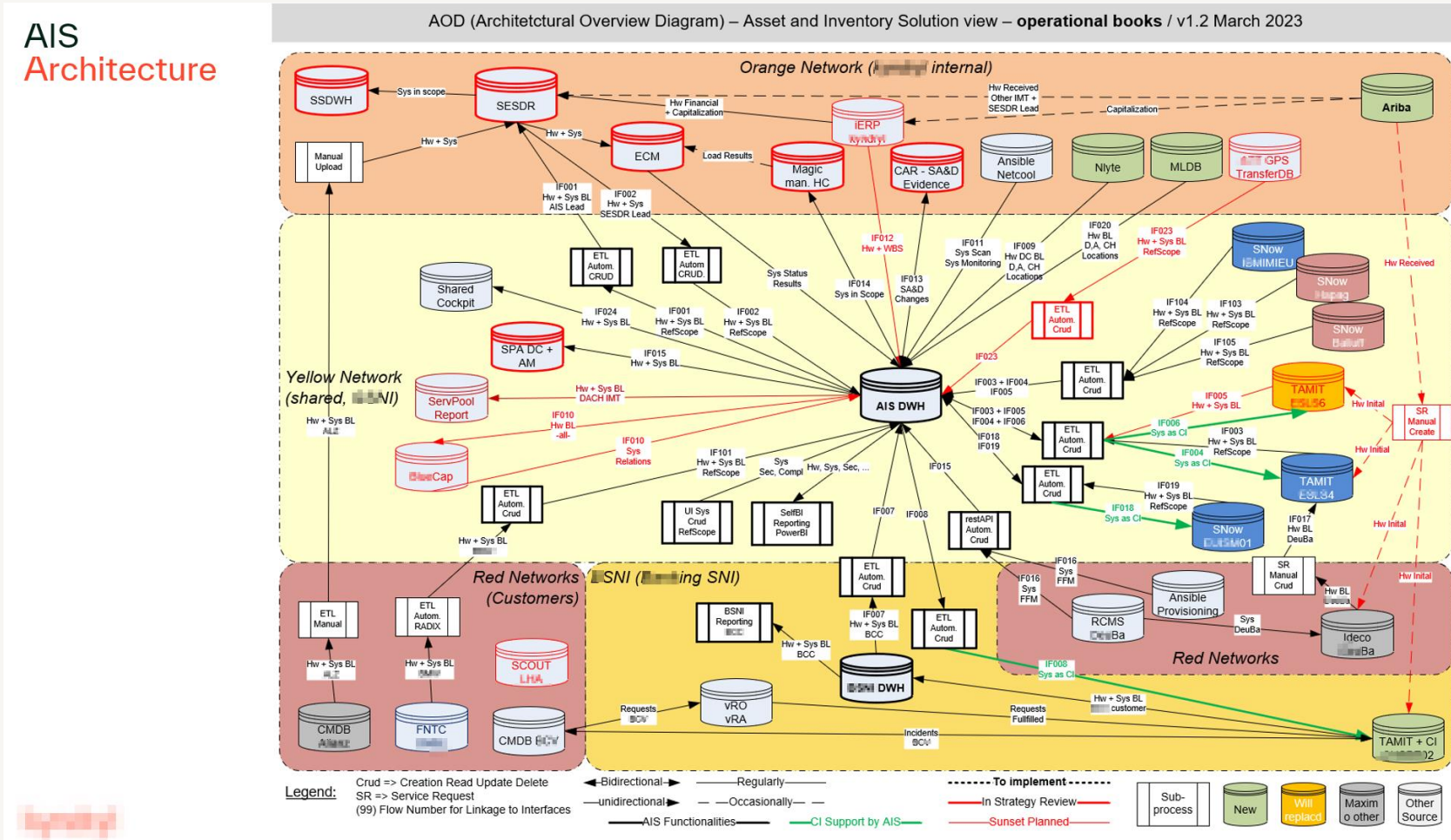
ASP.NET DevExpress

The screenshot displays the AIS ASP.NET DevExpress web application. The top section shows a table of assets with columns for AIS ID, Ref-to-Entry, Hostname, Serial-No, Operating System, Invent. Status, Oper. Status, Record Source, HW, and Parent S. Below the table, a detailed view for asset ID AIS-00815218 is shown, including fields for Customer, BP_ID, BAM ID, Customer Type, NDA Required, AIOPS Short ID, Owner ID, Company, Acc ID, ISM Customer, AIS HW Cat, MBC Category, Item ID, Manufacturers, Type, Model, Serial-No, Item Descr, Location ID, Country, Site, City, Building, Floor, Room, Rack, Location-ID, SPA Room Key, Coord.X, Coord.Y, Postal-Address, HW, Virtual, AIOPS Scope, Status ID, AIS Status, AIOPS IS Status, Invent. Status, Oper. Status, Hostname, Inventory Purpose, HW description, SAP V4H, SAP V4H INVR, DEPCH, Request #, SOCA, Installation Date, Delivery-Date, Created, Modified, Modified by, Last Status, and Change Date.

Angular

The screenshot displays the AIS Angular web application. The top section shows a table of assets with columns for AIS ID, Ref-to-Entry, Hostname, Serial-No, Operating System, Invent. Status, Oper. Status, Record Source, HW, and Parent S. Below the table, a detailed view for asset ID AIS-00815218 is shown, including fields for Customer, BP_ID, BAM ID, Customer Type, NDA Required, AIOPS Short ID, Owner ID, Company, Acc ID, ISM Customer, AIS HW Cat, MBC Category, Item ID, Manufacturers, Type, Model, Serial-No, Item Descr, Location ID, Country, Site, City, Building, Floor, Room, Rack, Location-ID, SPA Room Key, Coord.X, Coord.Y, Postal-Address, HW, Virtual, AIOPS Scope, Status ID, AIS Status, AIOPS IS Status, Invent. Status, Oper. Status, Hostname, Inventory Purpose, HW description, SAP V4H, SAP V4H INVR, DEPCH, Request #, SOCA, Installation Date, Delivery-Date, Created, Modified, Modified by, Last Status, and Change Date.

Massive Data Integration Layer for integration with Systems like ServiceNow and IBM Control Desk.



KAFKA CONSUMER AND PRODUCER

Python / .NET API / SQL



Python-Based Integration

Developed two Python applications (consumer and producer) to interface with the AIOps system using Kafka streams for seamless data input and output.

High Availability and Reliability

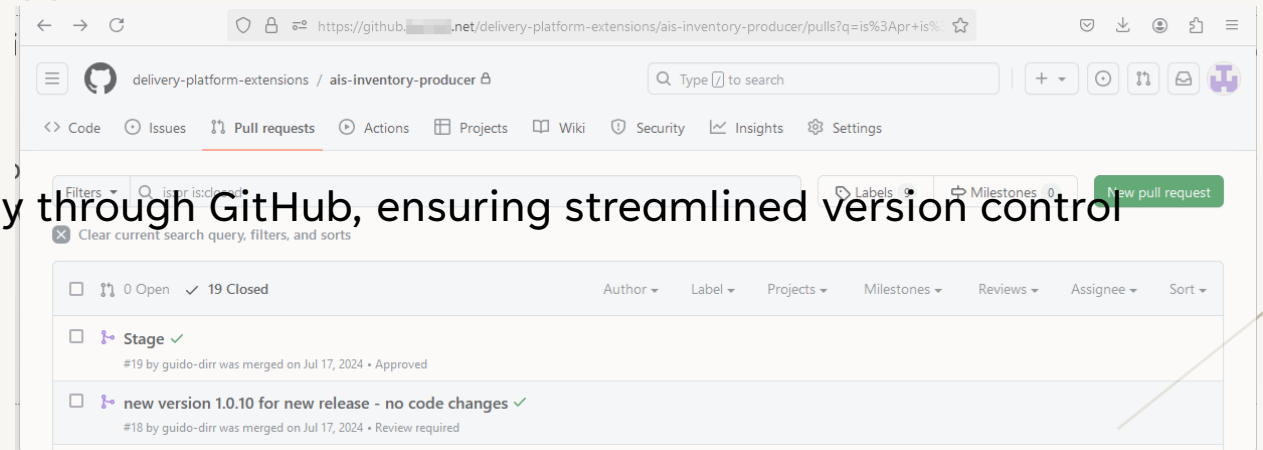
CE-certified Python applications run as Kubernetes containers 24/7 on a remote system, ensuring error-free operation and continuous data consumption for over 6 months.

.NET Web API

Facilitates seamless integration with the source application for enhanced data exchange and interoperability.

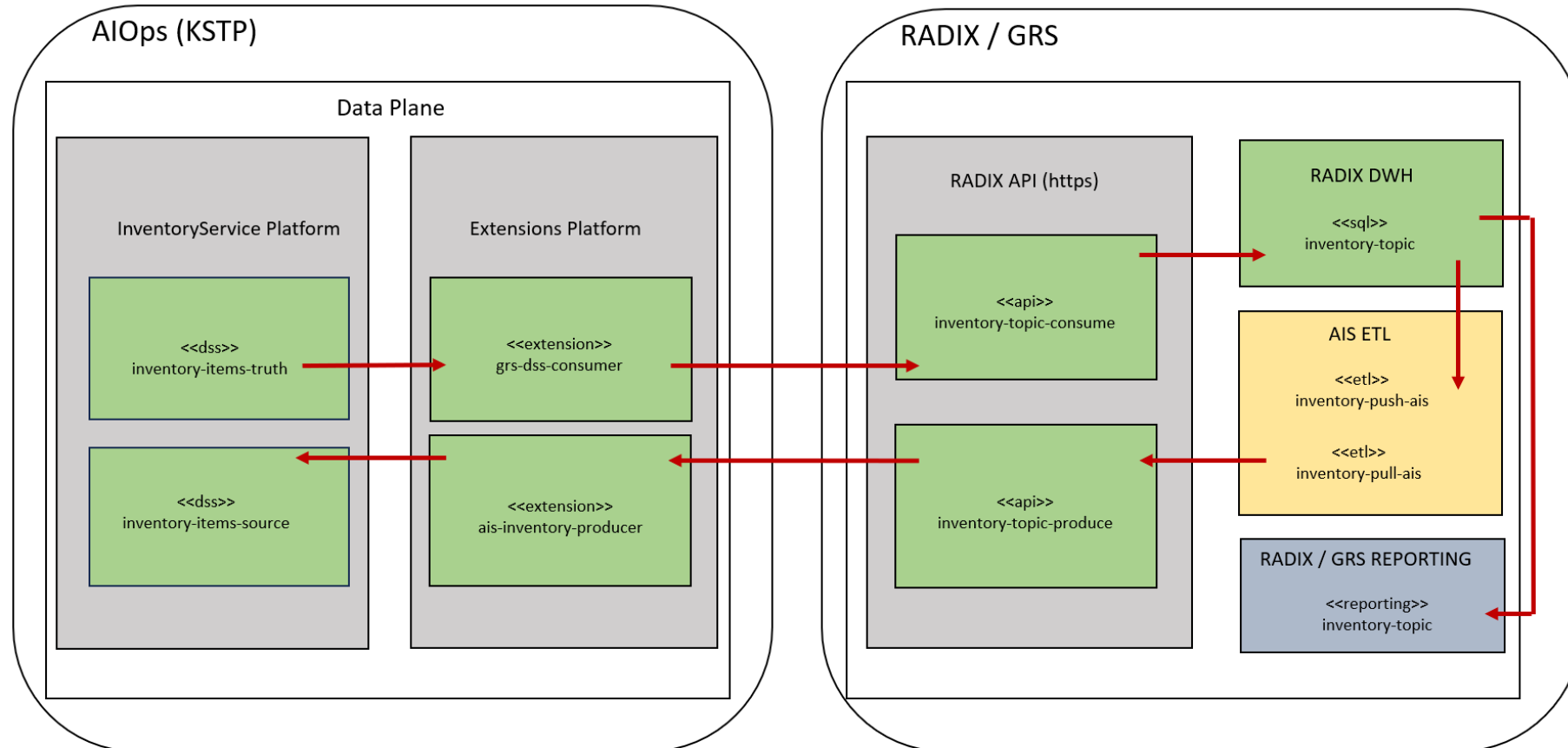
DevOps Best Practices

Source code and deployment managed efficiently through GitHub, ensuring streamlined version control and CI/CD pipelines.




KAFKA CONSUMER AND PRODUCER

RADIX/GRS API / AIS <-> InventoryService – AIOps Extension





KAFKA CONSUMER AND PRODUCER

A comment from the main architect of the integrated system AI Ops.

Jonathan [redacted]

To: ● Guido Dirr; ● DI [redacted]

Cc: ● P [redacted]

☀️ 😊 ↩️ Reply ↩️ Reply all ➡️ Forward  ⌵  ⋮


Thu 11/7/2024 4:46 PM

Start reply with:

Very helpful. Thank you!

Very nice! Thank you!

Thanks for sharing!


 Feedback

Hi, to put some context to this,
Guido implemented the first set of extensions for Inventory Service completed outside of the inventory team which are in production, and are an excellent example of a consumer and producer performing round-trip processing with an external cmdb, adhering to the preferred platform integration patterns

Jonathan [redacted] CEng MIET

Principal Architect

Open Group Certified Distinguished Architect

 · [Integrated AI Ops](#)

Phone: +[redacted]

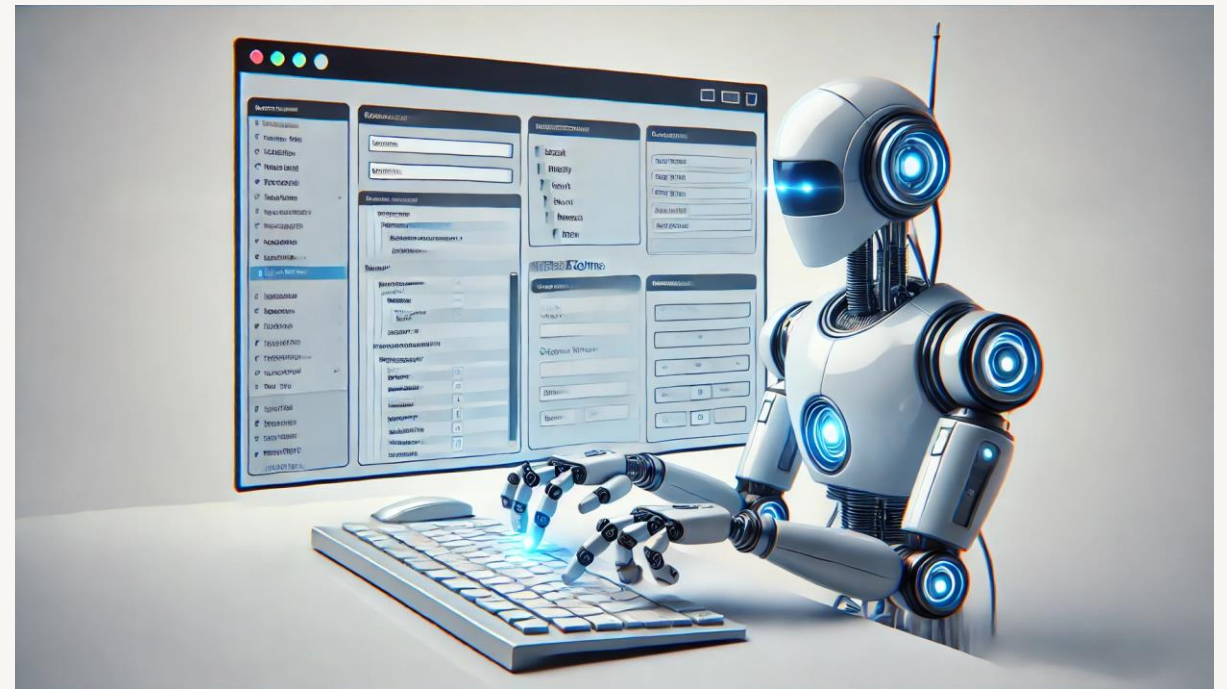
E-mail: [jonathan.\[redacted\]](#)

SELENIUM ROBOTS

WinForms C# / Python / Selenium

Developed advanced **Selenium** robots for automating:

- Application **Integration** Testing
- Application and SharePoint **Manipulation**
- Data Feed Automation via **complex web forms**
- System **Warming and Preloading** to ensure seamless operations



SHARE APP

WinForms / .NET API / SignalR

Challenge:

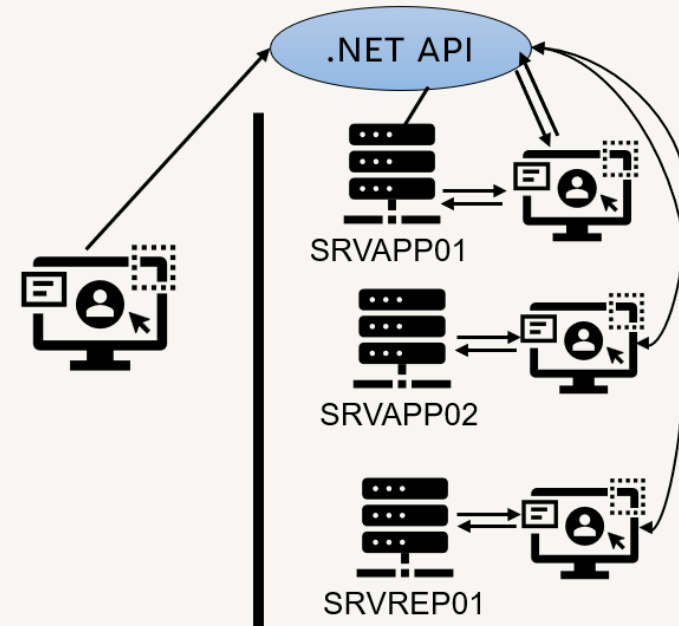
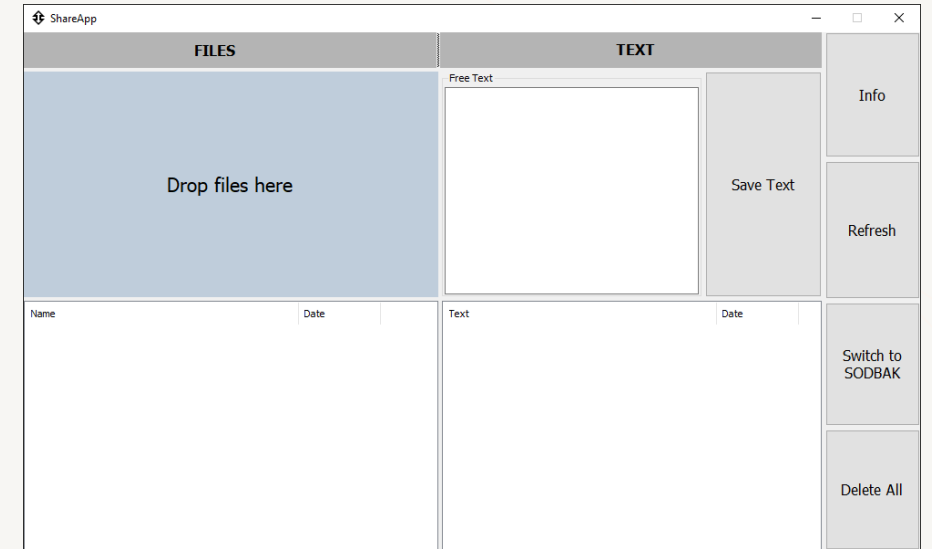
With the file copy option via RDP removed by policy, our team of ~20 developers required an efficient method to transfer files between servers.

Solution:

Developed a WinForms.NET client integrated with a .NET API for seamless file transfers.

Real-Time Updates:

Leveraged SignalR to provide live progress updates and notifications during file transfers.



APIMAN

C# Agent / SQL / JSON / XML

Automated Data Integration Framework

C# .NET Agent

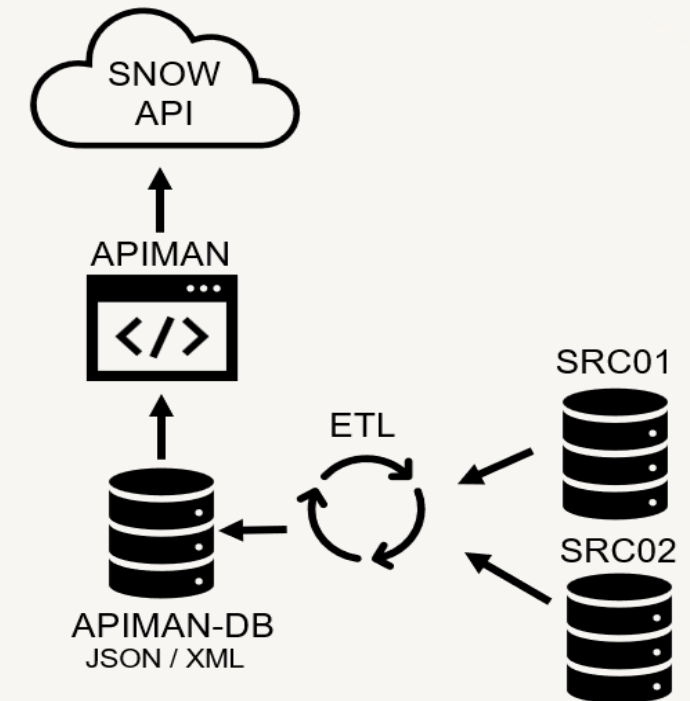
Interacts with external systems like ServiceNow, uploading data via APIs using JSON or XML formats.

Scheduled SQL ETL Jobs

Prepares data entities for the target system, managing operations such as insert, update, and delete.

APIMAN

- Processes and uploads prepared entities
- Tracks upload states and performs error handling
- Monitors interface activity and notifies teams via email or ticketing in case of issues



DATABASE VARIANCE TOOL

SQL / ETL / ASP.NET

Purpose:

Compare data across 5 different asset management systems that host varying data for the same entities.

Key Features:

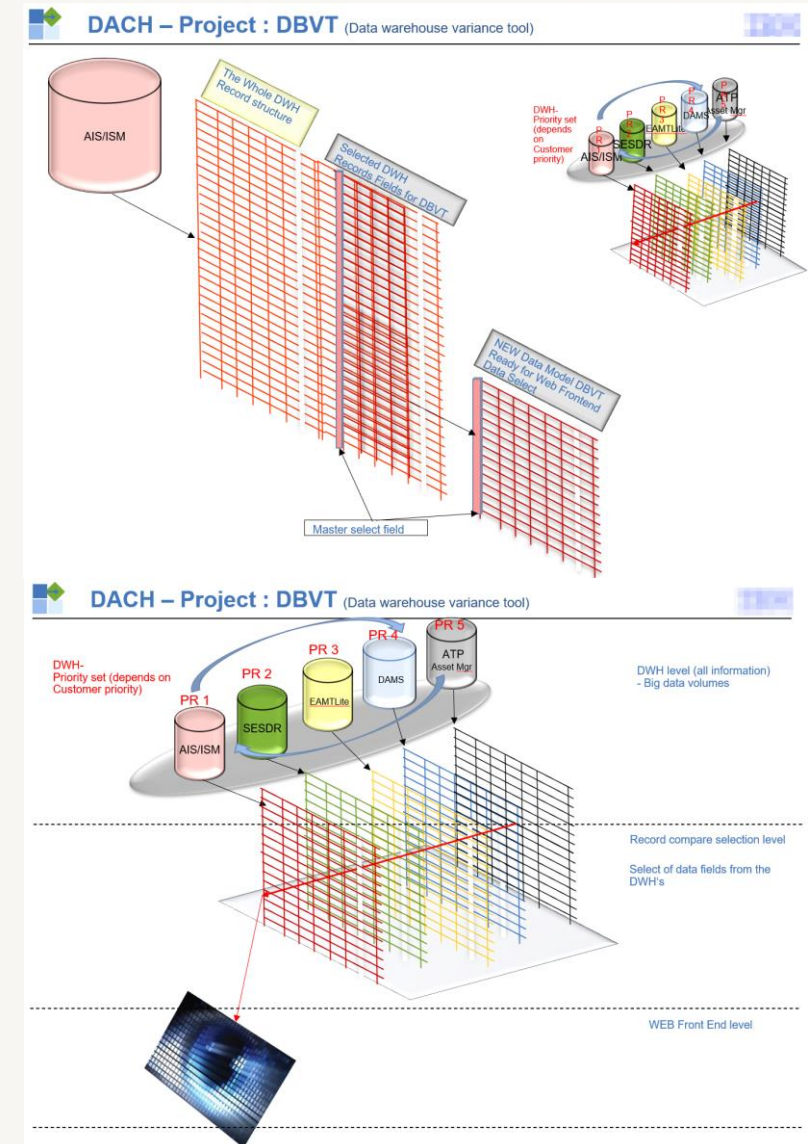
Map and compare fields across data sources to identify discrepancies.

Trigger automated corrections based on identified variances.

User-Friendly Interface:

ASP.NET Wizard: Easily select master and slave systems for detailed analysis.

Powerful filtering capabilities for precise data comparisons.



DEPLOYER

C# Agent / SQL

Centralized Job Management

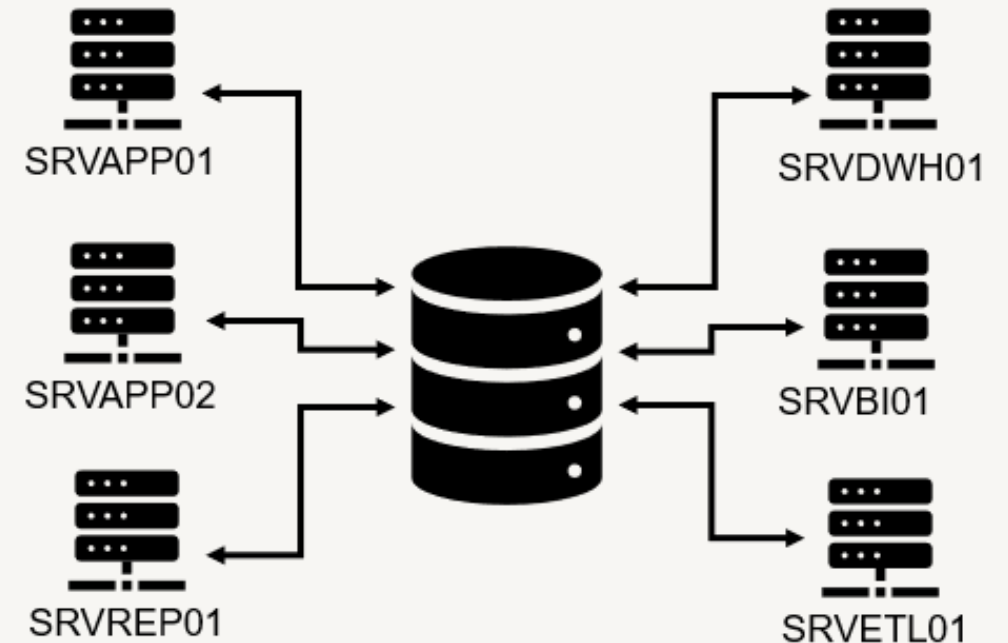
A C# Agent is installed on every server and polls a central database for jobs to execute, ensuring seamless task automation across the network.

Key Capabilities:

- Enable, disable, start, and stop services
- Deploy files using SQL binary streams
- Execute SQL and Windows commands in a sequential chain

Automation Use Cases:

- Automatically manage and update service user passwords
- Schedule automatic environment shutdowns on Sundays for server patching purposes



AI ENTHUSIAST

AI ENGINEERING / PROMPT ENGINEERING / MACHINE LEARNING FINANCIAL DATA PREDICTION PYTHON

