

AI-supported Code Migration of Data Analysis Scripts (SAS/Python)

A Case Study from Banking using LegacyLift

ADV Data Excellence Konferenz
Wien, April 2024

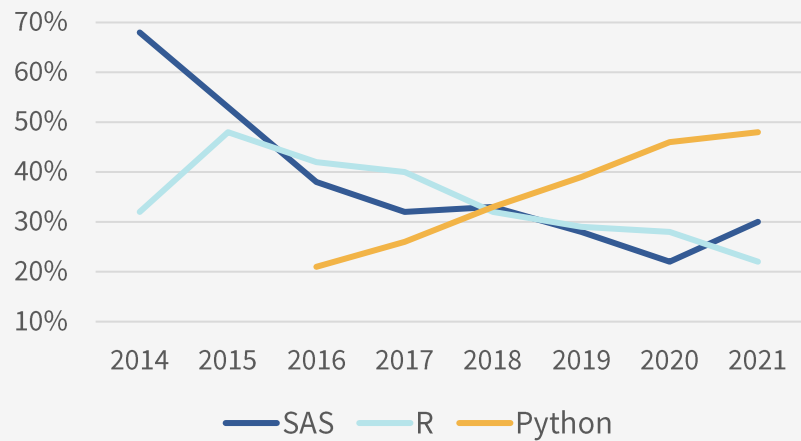
AI & Legacy Code Migration

SAS-Scripts to Python-Scripts?

Industry overview – Fight of SAS, R and Python in numbers

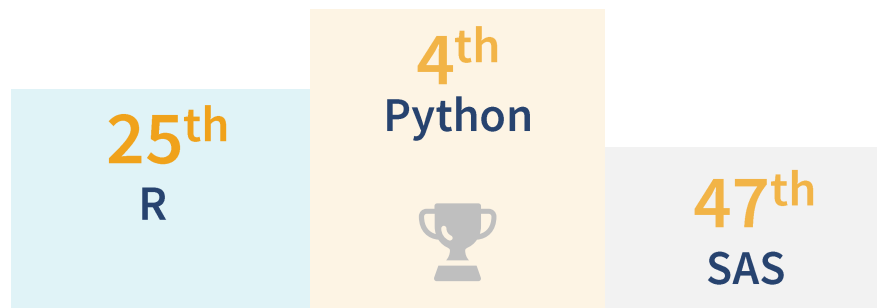
SAS loses its attractiveness by younger professionals, while Python has gained popularity.

SAS, R or Python: 8-Year Trend



Stack Overflow Most Popular Programming Languages

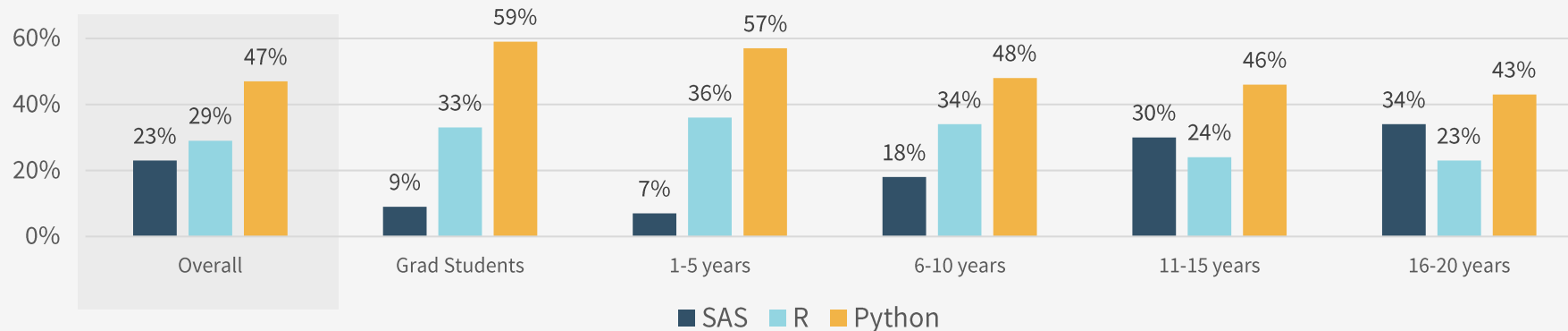
For several consecutive years Python is fighting to be in TOP 3 most popular programming languages, while SAS barely makes it to the ratings. R found its place in the middle.



70%+ SAS programmers are **40+ y.o.**

Students choose to learn Python **57%**

SAS, R or Python Preference by Years of Experience



Why Python?

1. Versatility
2. Fast Growing Ecosystem of Libraries and Packages
3. Community Support
4. Ease of learning
5. Cost Effectiveness
6. High Job Market Availability

AI has the potential to provide substantial benefits when implemented in specific use cases in financial industry, according to McKinsey & others

25-40%

Productivity Increase

The direct impact of AI on software engineering productivity. Impact for banking sector is estimated to be the highest after high tech industry.

9-15%

Value Potential

As percent of operation profits for key use case, **legacy code conversion** being the biggest value driver

42%

Employees regularly using AI tools

While total of 83% percent of all respondents from financial services industry say they've had at least some exposure to AI tools

44%

Reported Cost Decrease

Across respondents in risk departments, 5% reported more than 20% cost decrease, 13% - a decrease from 20-29% and 26% a decrease of less than 10%

Beyond ChatGPT?

What to look for in a code translation and migration tool?



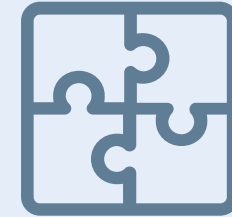
Ability to utilize the best AI models available

Ability to switch to a different and better AI model as soon as it becomes available, as better and more powerful models are likely to be available in the next months



Customized for your language pairs and data architecture

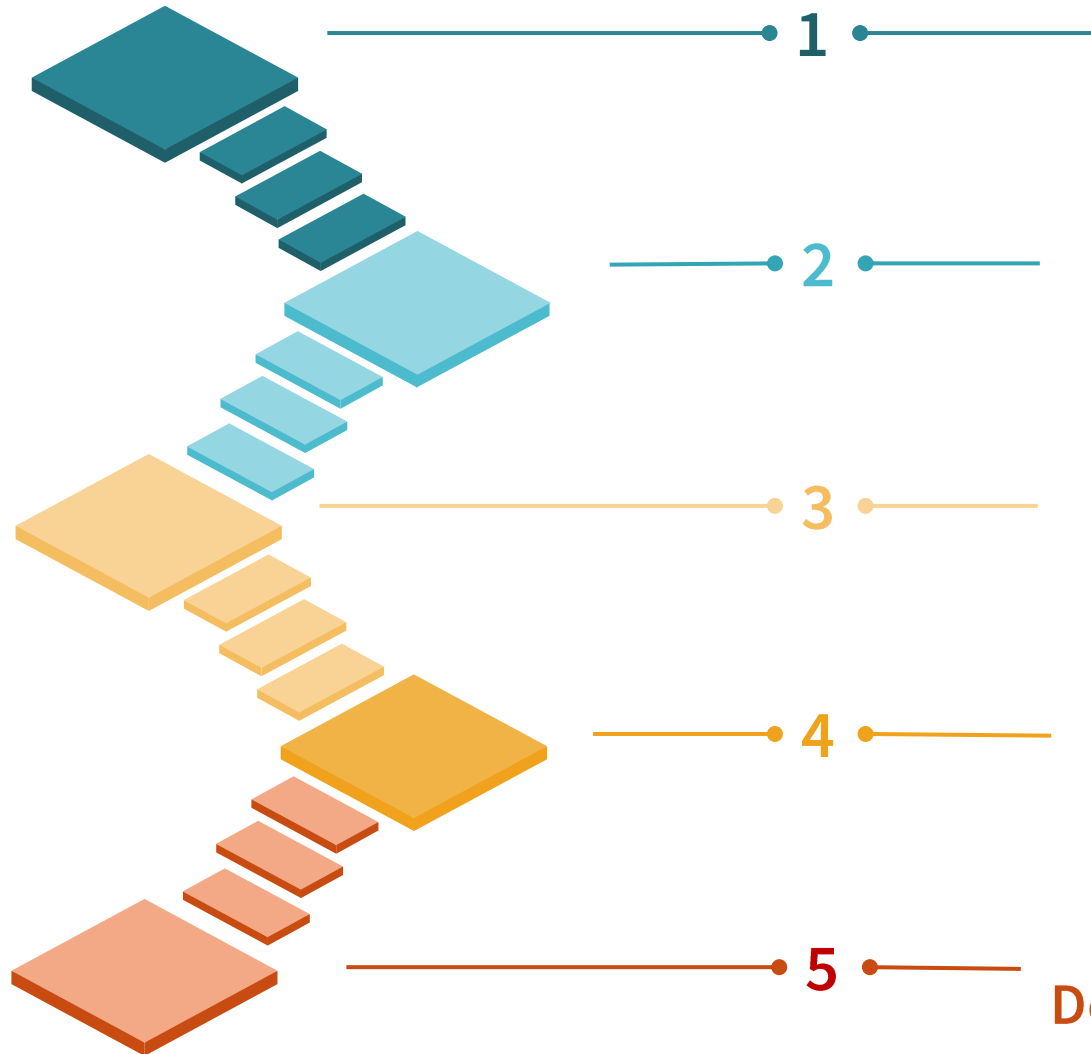
RAG-pattern code database filled with examples and adjusted to match the particular target architecture, coding standards and requirements coupled with configurable prompt engineering



AI-Testing & -Documentation, Enterprise-grade Workflows

AI-automated testing and code documentation to support the process beyond the code translation. Ideally, enterprise-grade workflows and full audit trail.

Where to use AI in a code migration project?



1 AI Code & Architecture Analysis

Split the legacy code base in code complexity buckets, extract basic stats
Create architecture flows and ER diagrams

2 AI Code Commenting

Add in-line comments to legacy code
Both prior to translation and after translation

3 AI Code Translation

Translate code automatically based on the customized AI agent (target language libraries, styles, architecture examples)

4 AI Test Generation

Create automated tests
Decrease manual quality control efforts until code is production ready

5 AI Documentation

Generate human-readable summary of what each particular script/piece of code des

Case Study

Bank-wide SAS-to-Python Migration

Case Study: How to utilize AI to migrate SAS code to Python and Snowpark

1/2

Problem: A bank wants to migrate away from SAS and into Python to modernize its reporting and business analytics. Python with Jupyter and Snowflake/Snowpark are the tools of choice.

1. Human – Written SAS Code

- Own estimate **5700** PDs from own **business SAS** users
- SAS-transformations
- Calculations work
- Analytics & Reporting



2. Auto-Generated SAS Code

- **500.000 lines** of code
- ETL code
- Visualizations
- Monitoring Analytics



Challenges

- Very **high effort estimates** if done internally
- **Blocking** of internal resources, i.e. almost impossible to be done internally
- Which **options** are there to supercharge such a migration project with AI?
- Or is AI only good for generating images and LinkedIn posts?

Approach:

LegacyLift + SPG + minimal bank team

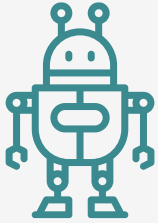
PoC Results:

Coming up at the end of presentation

Case Study: How to utilize AI to migrate SAS code to Python and Snowpark




LegacyLift

- SAS → Python AI Agent
- SAS → Snowpark AI Agent
- SAS → Python AI Agent that maximizes SQL





Project Team

External Team

		
Project Manager	Business Analyst	Data Engineer
0.5 FE	1 FE	2 FE

Client Side

	
SAS Expert	IT SAS person
0.2 FE	0.2 FE

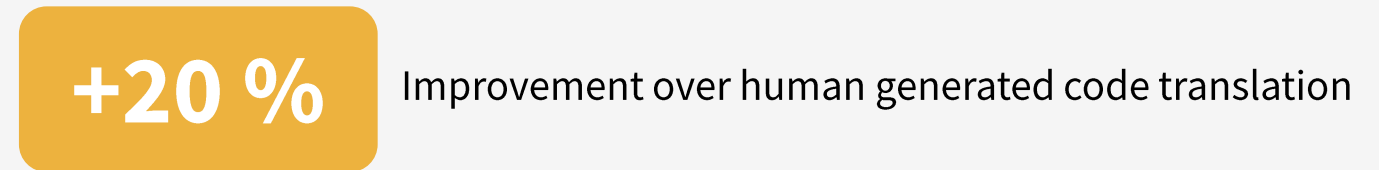
Client support is vital due to familiarity with the codebase making customization easier and resolving issues faster.

PoC Results

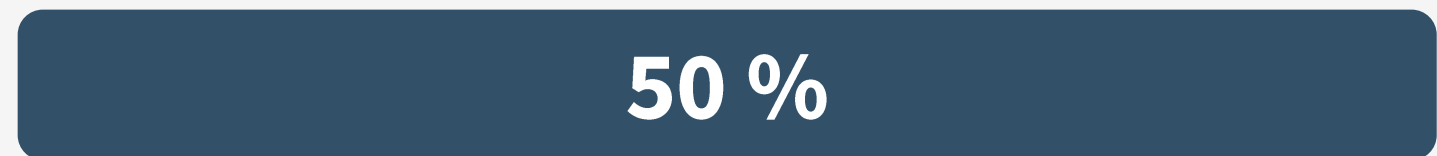
Human-generated SAS code



SAS-generated ETL:



Overall project savings vs fully human code migration



Demo: Legacy Code Migration in Action using




Thanks for your attention! Questions?



Karl Ivo Sokolov
Managing Partner


 ivo.sokolov@specific-group.com

 Tel: +43 664 8337312



Daria Kovalchuk
Product Owner

 daria.kovalchuk@specific-group.com

 Tel: +43 677 64373288



Daniel N. Huber
Principal Data Scientist

 daniel.huber@specific-group.com

 Tel: +43 1 489 25 28

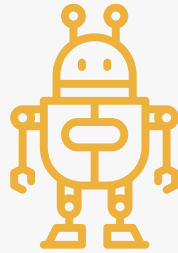
Backup Slides

Information on LegacyLift

Distinct Features

We provide a range of model choices, all optimized with carefully crafted prompts, to ensure the best performance and results

Utilize the power of AI Agents



- Choose between multiple secure public and private LLMs
- Select the language pair you need
- Use specific libraries based on your target system architecture
- Maximize or minimize SQL usage
- Adhere to any other company standard by creating new agents

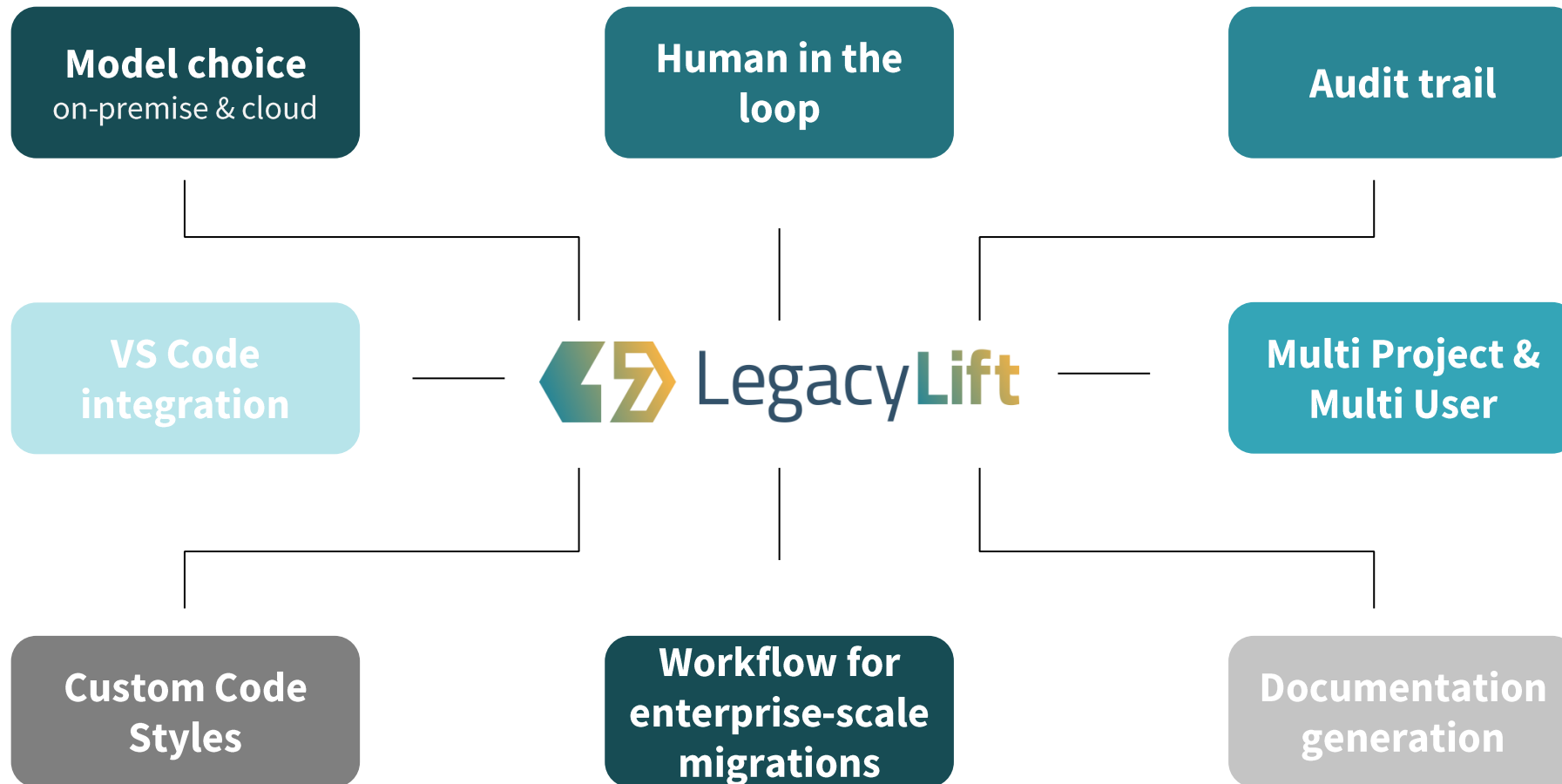
End – 2 – End Migration Workflow

- Human-in-the-loop
- Fits into your modern system architecture
- Testing & Validation
- An experienced development/consulting team at hand

Available Languages

SAS		Python / also optimized for Snowpark
COBOL	→	SQL / also optimized for Snowflake
PL/1		Java

Legacy Lift offers all features needed for enterprise scale code translation projects



This is how it works:

Upload the code example for AI analysis, which splits the code into functional blocks and selects the best example from the RAG database. This optimized example, along with tailored prompts, is sent to the LLM model, with the process iteratively refined through multiple feedback loops for accuracy and efficiency.

Effortlessly track and manage your translation projects

Projects

Alexandra Abraham
Dark mode:

- Dashboard
- Projects
- User Management
- Role Management
- Code Converter
- Agents
- Analytics

+ Add Project

Project Name	Description	Progress	Status	Action
CoreBank Data Migration Project	CoreBank Data Migration Project	56 %	Active	Dev Tool File Management Edit Delete
CoreBank Integration Platform	CoreBank Integration Platform	65 %	Active	Dev Tool File Management Edit Delete
Regulatory Compliance System	Regulatory Compliance System	10 %	Active	Dev Tool File Management Edit Delete
Payment Processing Implementation	Payment Processing Implementation	40 %	Active	Dev Tool File Management Edit Delete
Fraud Prevention Enhancement	Fraud Prevention Enhancement	54 %	Active	Dev Tool File Management Edit Delete

+

Efficiently administer user access and permissions, ensuring secure and organized process

The screenshot displays the LegacyLift User Management interface. At the top, there is a search bar with filters for 'Project' and 'Role', and a '+ Add User' button. The main content is a table with columns for Employee Name, Projects, Roles, Username, Email Address, and Action. The table lists five users: John Doe, Sam Smith, Liam Holmes, Harry Kane, and Kate Hudson, all with the role of 'Admin' and associated with the 'Fraud Prevention Enhancement' project. Each user row includes 'Edit' and 'Delete' buttons. A dashed box with a plus sign is at the bottom of the table area. The left sidebar contains navigation options: Dashboard, Projects, User Management (highlighted), Role Management, Code Converter, Agents, and Analytics. The top right shows the user profile 'Alexandra Abraham' and a 'Dark mode' toggle. The SPG Specific Group logo is in the bottom left corner.

Employee Name	Projects, Roles	Username	Email Address	Action
John Doe	Fraud Prevention Enhancement: Admin	johndoe	john@example.com	Edit Delete
Sam Smith	Fraud Prevention Enhancement: Admin	Sam_Smith	Sam_Smith@mail.com	Edit Delete
Liam Holmes	Fraud Prevention Enhancement: Admin	Liam_Holmes	Liam_Holmes@mail.com	Edit Delete
Harry Kane	Fraud Prevention Enhancement: Admin	Harry_Kane	Harry_Kane@mail.com	Edit Delete
Kate Hudson	Fraud Prevention Enhancement: Admin	Kate_Hudson	Kate_Hudson@mail.com	Edit Delete

Defining access roles within the system to manage permissions and responsibilities

The screenshot displays the LegacyLift Role Management interface. At the top, there is a search bar and an '+ Add Role' button. The main content is a table with columns for Role Name, Permissions, Description, and Action. The roles listed are QA, PM, Developer, and Admin. The Admin role is described as 'Administrator role'. The bottom of the table has a dashed border with a plus sign, indicating a new role can be added.

Role Name	Permissions	Description	Action
QA	Project: read Role: read User: read		Edit Delete
PM	Project: delete read create update		Edit Delete
Developer	Project: update read		Edit Delete
Admin	Role: update delete read create User: create read update delete Project: create update read delete	Administrator role	Edit Delete

Easily create and adjust agents to your company requirements

The screenshot displays the LegacyLift Agent Management interface. On the left is a sidebar with navigation options: Dashboard, Projects, User Management, Role Management, Code Converter, Agents (highlighted), and Analytics. The main area shows a grid of agent cards. Each card includes an icon, name, model, projects, and language pair. A '+ Add Agent' button is in the top right. An 'Edit Agent' modal is open, showing fields for Agent Name, Key, Model, Input Language, Output Language, Libraries, and a Customization text area. A 'Maximize SQL' toggle and an 'Icon' selection row are also visible. A 'Cancel' and 'Save' button are at the bottom right.

Get an overview of available agents and create your own

Select language pairs, libraries, LLM Models. Add customized prompts

Connect to GIT to see project files overview and their progress and complexity

File Management

Status Start date End date Complexity + Connect to GIT

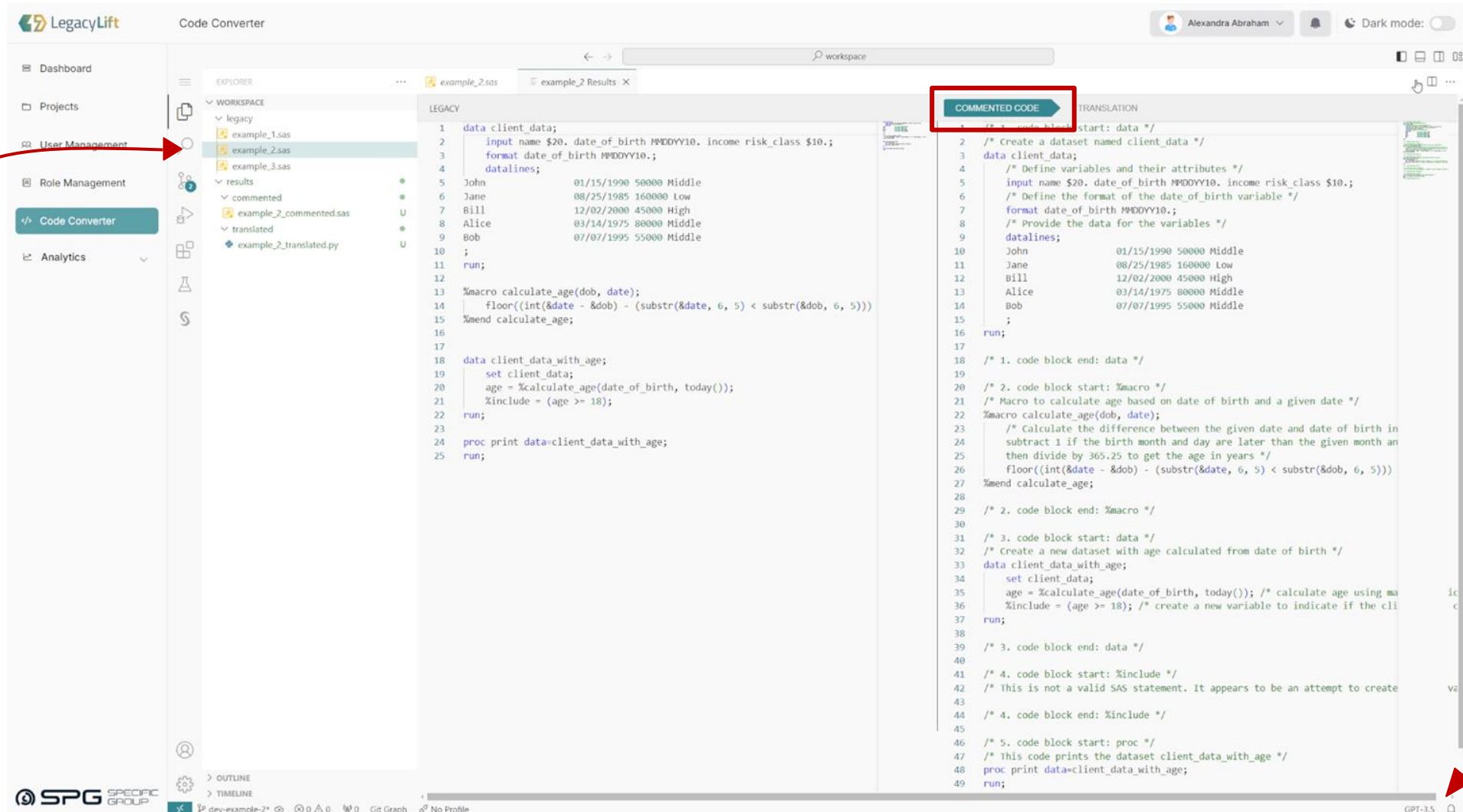
File Name	Status	Last Update Time	Responsible	Agent	Complexity	Lines	Action
customer_data_integration.sas	In Progress	2-20-2024 10:43	Zain Saris	Python2SAS	MEDIUM	113	Edit
mortgage_underwriting_analysis.sas	In Progress	2-18-2024 11:23	Lydia Rosser	Python2SAS	LOW	56	Edit
funds_transfer_validation.sas	Done	2-14-2024 14:34	Hannah Calzoni	Python2SAS	MEDIUM	146	Edit
regulatory_reporting_compliance.sas	In Progress	2-12-2024 09:12	Alfredo Donin	Python2SAS	HIGH	234	Edit
anti-money_laundering_monitoring.sas	In Progress	2-05-2024 15:11	Talan Franci	Python2SAS	HIGH	1335	Edit
loan_portfolio_management.sas	Done	2-01-2024 13:34	Paul Davis	Python2SAS	MEDIUM	56	Edit
wealth_management_analysis.sas	Done	2-05-2024 15:43	Martin Vtrovos	Python2SAS	LOW	554	Edit
credit_risk_assessment.sas	In Progress	1-25-2024 14:22	Aram Shafir	Python2SAS	HIGH	768	Edit
treasury_operations_optimization.sas	In Progress	1-18-2024 12:55	Giana Curtis	Python2SAS	MEDIUM	223	Edit
Basel_III_reporting.sas	Done	1-17-2024 11:51	Ryan Garcia	Python2SAS	HIGH	453	Edit
credit_scoring.sas	In Progress	1-15-2024 16:13	Christian Thomas	Python2SAS	MEDIUM	879	Edit

SPG SPECIFIC GROUP

Analyze complexity with AI

Track project progress on files level

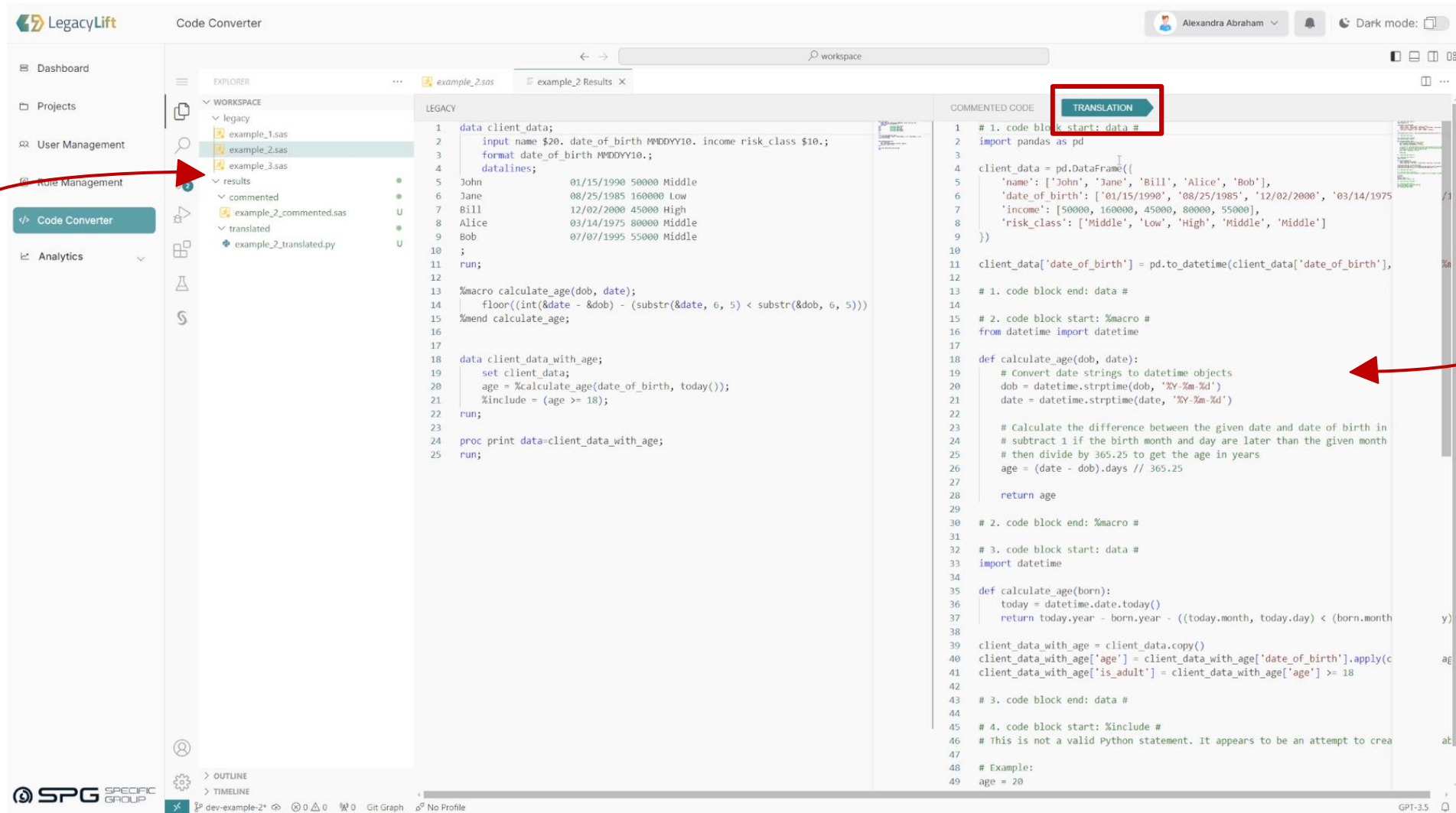
Clear and concise comments within your legacy code for easy understanding and reference



Use the entire VS Code functionality right inside Legacy Lift, such as git integration, automatic unit testing and more

Change AI Agent with few clicks

Seamlessly converted legacy code into modern languages, ensuring functionality and efficiency



Easily navigate, organize, and manage your code repository

Benefit from convenient syntax highlighting

We still need capable human teams to complete the entire migration process. Human-in-the-loop + AI agents + workflow tooling fit for the enterprise

