AI-based Code Understanding and Modernisation. A Vendor Solution presented by Fresche Solutions

Steve Cast

Senior Vice President EMEA and APAC

info@freschesolutions.com

Agenda

AI-based Code Understanding and Modernisation.

1. Using X-Analysis AI for Analysis, Planning, & Enhancements
2. Integrating X-Analysis AI into Development Workflows
3. Using X-Analysis AI and X-Modernise AI for Code Modernization
4. Getting Started

Using X-Analysis AI for Analysis, Planning, & Enhancements

Analysis Requirements in the context of the IBM i

- Holistic, Connected View of All Elements
  - RPG, COBOL, Synon, CL, Query400, PFs, LFs, SQL/DDL, DSPF, PRTF, etc.
- Impact Analysis Across All Object Types (where used)
- Data Flows
- Screen Flows
- Parameters
- Business Rules (isolated and consolidated views)
- Documentation
- Deterministically Accurate

What is Application Understanding Used for?

- Impact Analysis
- Supporting Break / Fix Activities
- Test Planning
- Developer Documentation
- Code / Process Modernization (including refactoring)
- API Interfaces (creating / wrapping)
- Rewrites
- Mobile / Portal extensions
- BI / Analytics

Challenges with Gen AI for documentation (e.g. IBM Bob)

- Small context size (cannot process entire applications)
  - 1M tokens is current largest model, 1 token = 2/3 of a word (anything that could have a discreet meaning; eg. runn-ing = 2 tokens).
- Not deterministic
- AI processing for each request – Costly (especially on repeat queries)
- When AI runs out of room in its context, it summarizes and may ignore things, which creates a degree of inaccuracy.
- Size is not the only problem… can AI reason well with incredibly large context sizes.
  - You won't realistically know what it didn't do.

Deterministic vs. Probabilistic

	IBM Bob, Anthropic Claude, ChatGPT	X-Analysis AI
Pros	• Probabilistic • Can infer intent • Reasons about whether a comment still matches code • Can make a judgement about whether a refactor preserved the original meaning.	• Deterministic • 100% accurate (source of data) • Consistent and predictable results • Results are reproducible and auditable
Cons	• Unpredictable – same prompt can yield different answers or results. • It can confidently assert things that are not true. • You have to double check its work. • Doesn’t scale well. • Costs grow with token volume. • Quality of results degrades as the context window fills with repetitive structural detail. • You can end up acting on hallucinated data.	• Limited in functional scope and design • Cannot reason about intent • Eg. It doesn't know whether a function is well-named or whether a change violates unwritten architectural conventions.

The optimal pattern is to let the deterministic layer be the source of truth for facts about the code — what exists, what references what, what types flow where — and use AI as a reasoning layer on top of those facts.

AI-based tools for IBM i Developers

	IBM Bob	X-Analysis AI
Features	GenAI tool for IBM i built by IBM	Industry-leading, mature analysis and documentation tool for IBM i
Pros	• Purpose-built for IBM i (RPG, COBOL, CL) • Model trained by IBM • Integrated into VSCode • Pay per use consumption model	• Purpose built for IBM i (RPG, COBOL, CL, Synon, Query400, DDS, SQL, DSPF, PRTF, ...) • Fresche covers cost of AI tokens for first 6mo. • Deterministic model = 100% accuracy • AI consumption is for augmented analysis and explanations only. (less costly) • MCP interface to integrate with IBM i Bob • Fixed monthly cost
Cons	• Fork of Anthropic's LLM model (updated per IBM's schedule) • Fork of VSCode (may deviate from main VSCode branch. Updated by IBM.) • Becomes proprietary over time. • May not have latest and greatest LLM or VSCode • Each AI query/task consumes Bob Coins even if query was previously done • Unpredictable finances • Less accurate analysis when context size becomes too big	• More expensive when not in use • No specific development functionality. • Solely built for analysis and documentation • For testing, scope of use is limited to Test Planning and dependency analysis.
Best Fit	IBM i Development and Test Automation Work	IBM i Analysis and Documentation

Integrating X-Analysis AI into Development Workflows

 |

What do “Next-gen” AI Documentation Tools like X-Analysis AI give you?

- AI is leveraging deep and accurate deterministic source of data (X-Analysis Repositories)
- Entire application ecosystem mapped and cross referenced in advance
  - Including Copybooks, external references, etc.
- Meta data as a source (not only source as a source)
- More accurate results
- More comprehensive results
- Faster

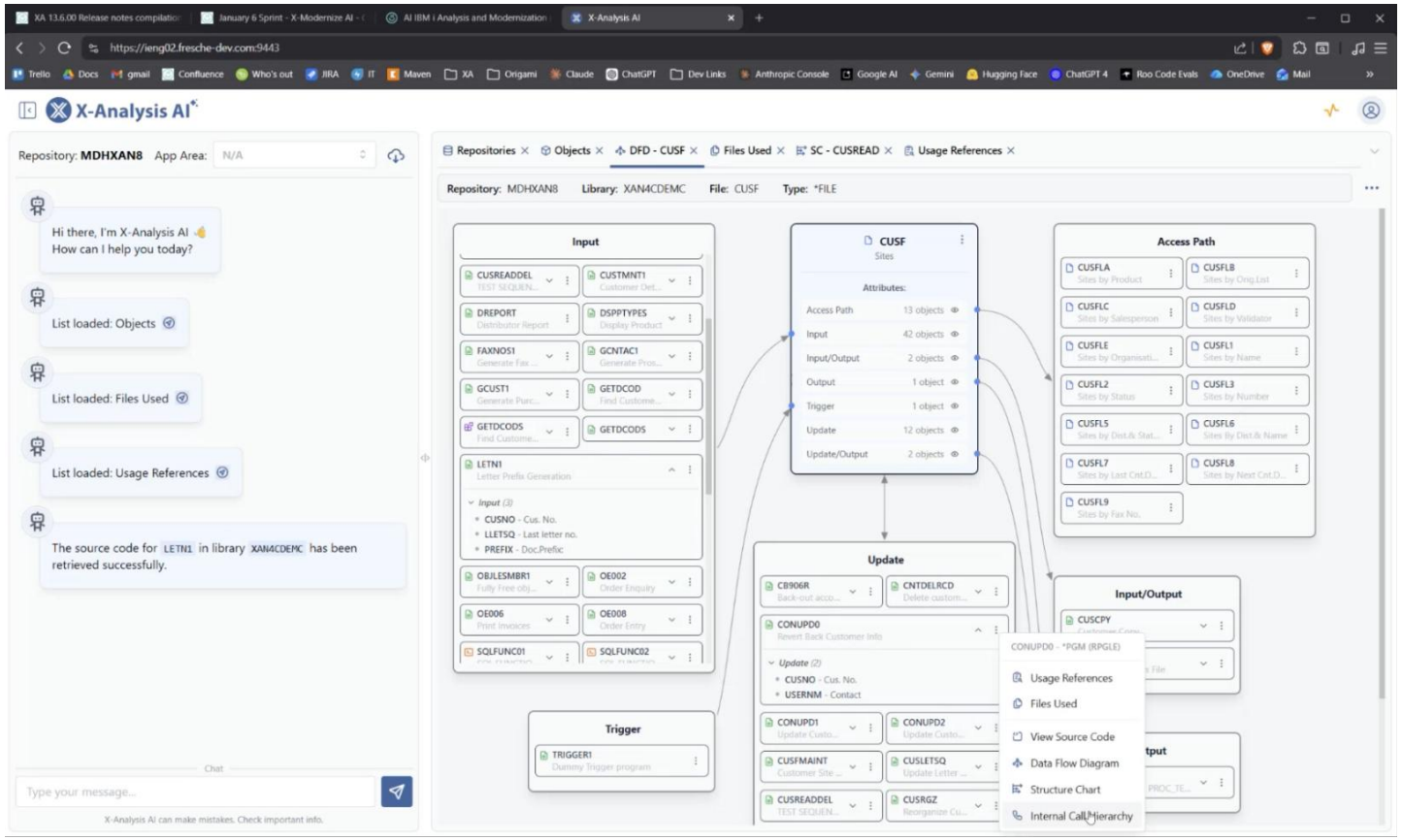
Productivity Impact of AI tooling accelerators

	Req. Gathering	Analysis	Design	Development	Testing	Deployment	Document.
New Dev with Java or .NET	10%	10%	15%	30-40%	20-25%	5-10%	5-10%
Changing Mature, Complex RPG / COBOL systems	5-10%	20-30%	10-15%	15-20%	30-40%	5-10%	5-10%
Changing Mature IBM i systems with AI tools	5-8%	**8-12%**	8-12%	10-15%	**15-20%**	5-8%	3-5% *(5-10x output)*

Notes:

- On legacy systems, you're paying a "comprehension tax" and a "don't-break-anything tax" that together can double the effort of an equivalent-sized change on a modern codebase.
- 25-40% drop in effort by using AI tools, specifically when you pair AI with deterministic analysis tools.
- Neither AI tool alone cracks legacy work; together they attack exactly the two phases that dominate legacy effort: analysis and testing.
- AI alone hallucinates on legacy code. LLMs will confidently describe what a COBOL paragraph "does" and be subtly wrong. Combining with a deterministic layer/tool is what keeps the AI honest.

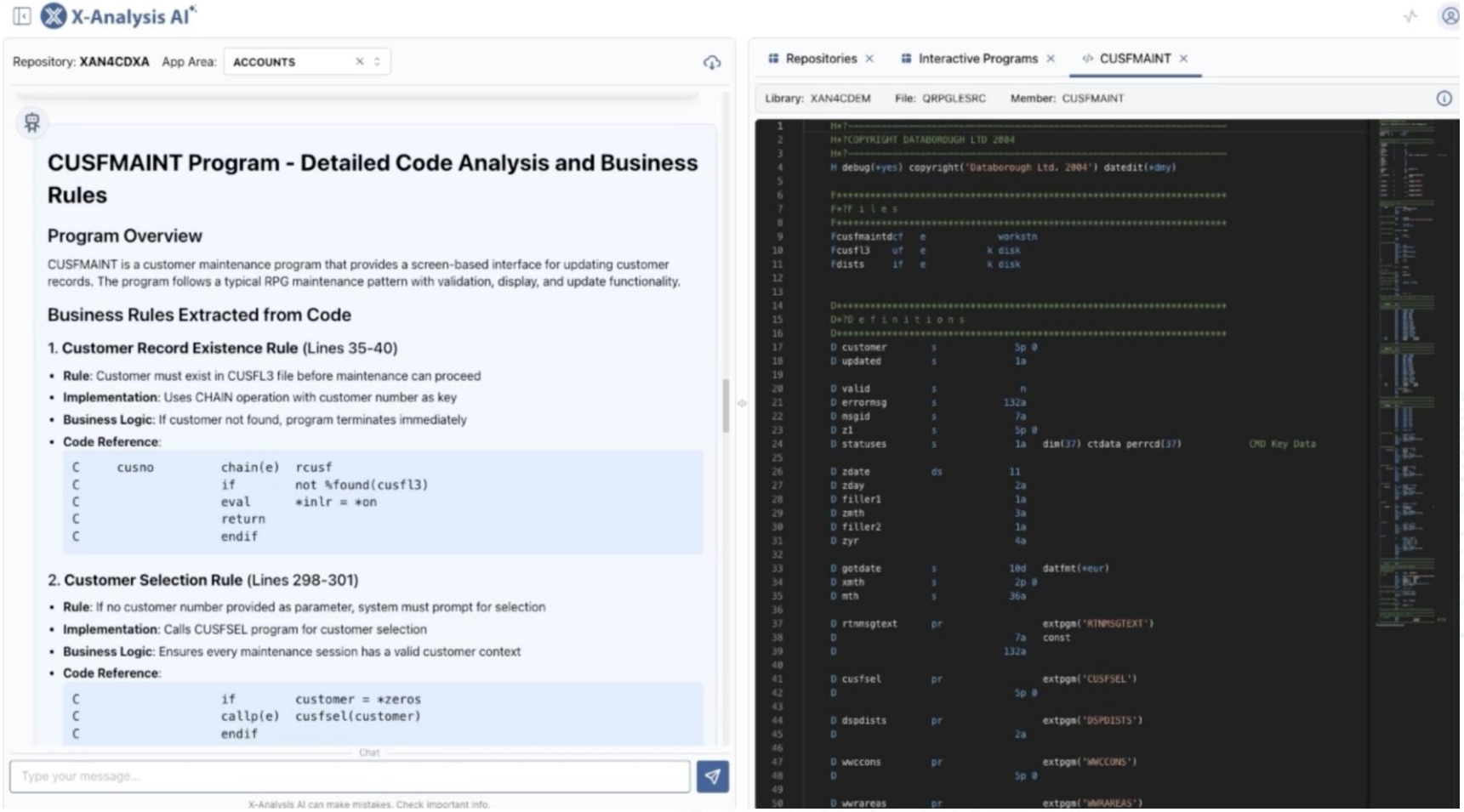
What About X-Analysis AI?



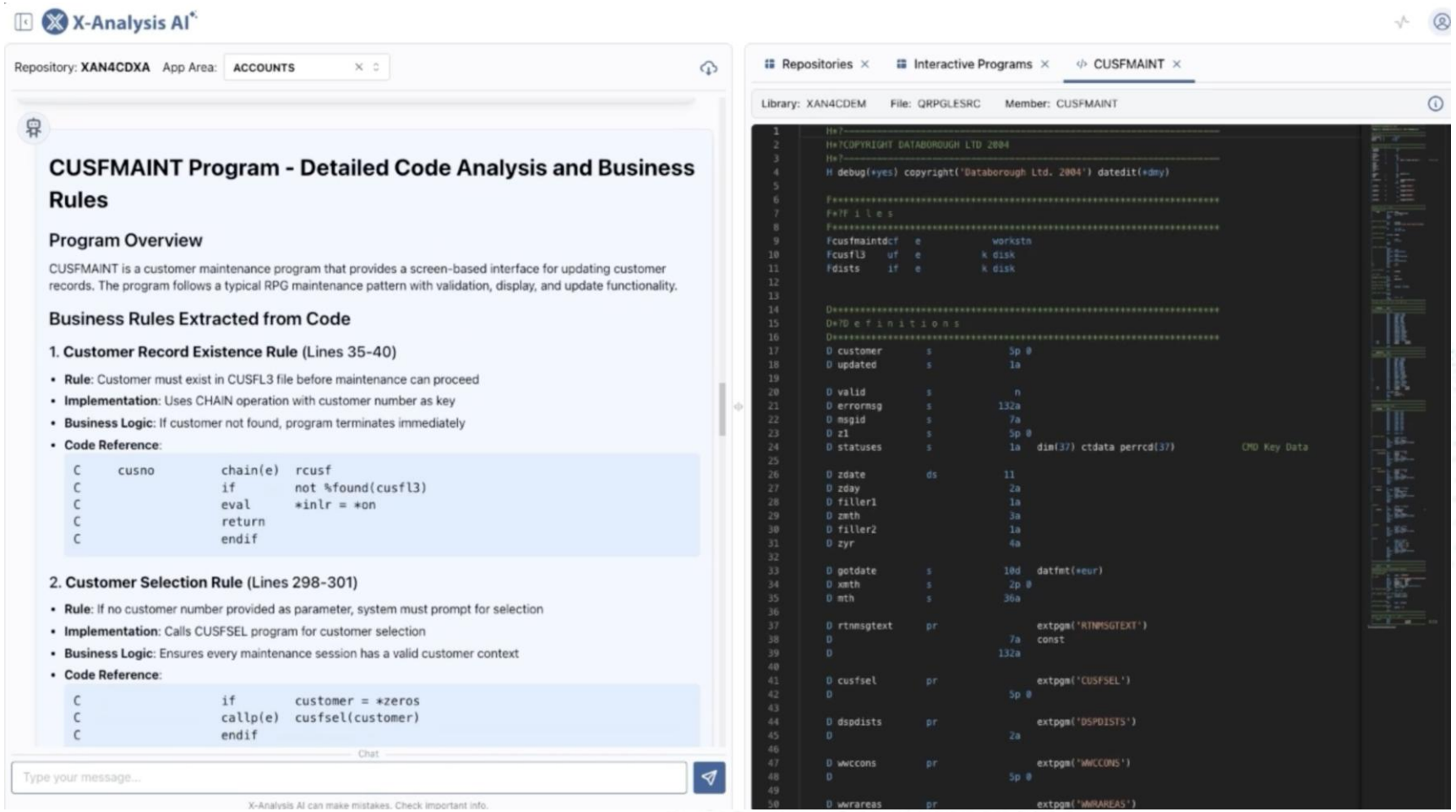
- AI-built code explanations based on the deterministic data
- Go anywhere from anywhere navigation
- Visualizations and flow diagrams
- NLP Interface (Natural Language Processing)
- Comprehensive descriptions of programs, logic blocks, business rules, etc.
- Faster and more accurate than other AI tools

https://freschesolutions.com/products/x-analysis/

X-Analysis AI Demonstration – Part 1 of 2



X-Analysis AI Demonstration – Part 2 of 2



https://freschesolutions.com/products/x-analysis/

Using X-Analysis AI and X-Modernize AI for Code Modernization

 |

Challenges of Gen AI for Code Modernization

Skipping What It Doesn't Understand

Manual verification and changes required

Hallucinations

Adding things that weren't there. Manual review and removal required

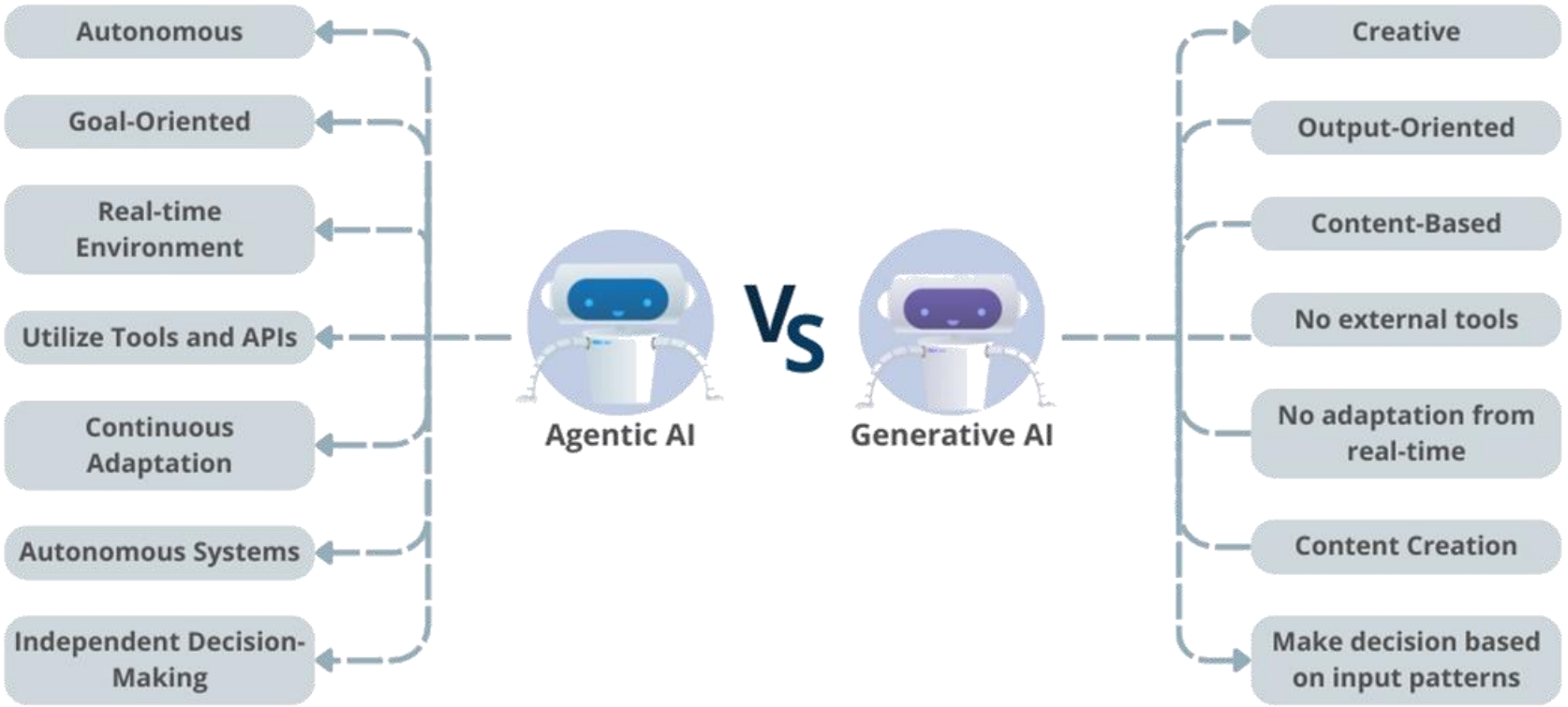
Inconsistent Generation

Generating the same programs multiple times generates different results. Especially when LLMs change/update.

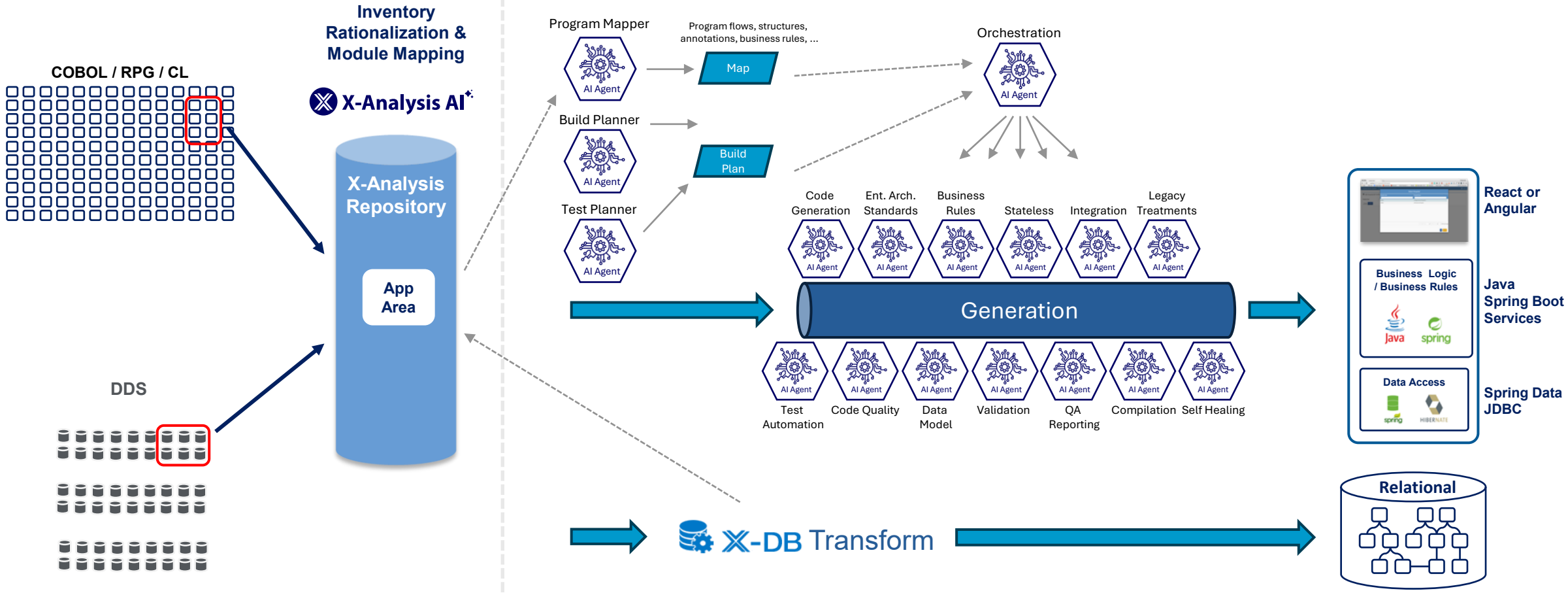
Single Program vs. Enterprise App

Additional structure and coding required for enterprise features, like security, global variables, user-specific variables, context specific shared data, naming conventions, code repetitions in every program (eg. date routines), etc.

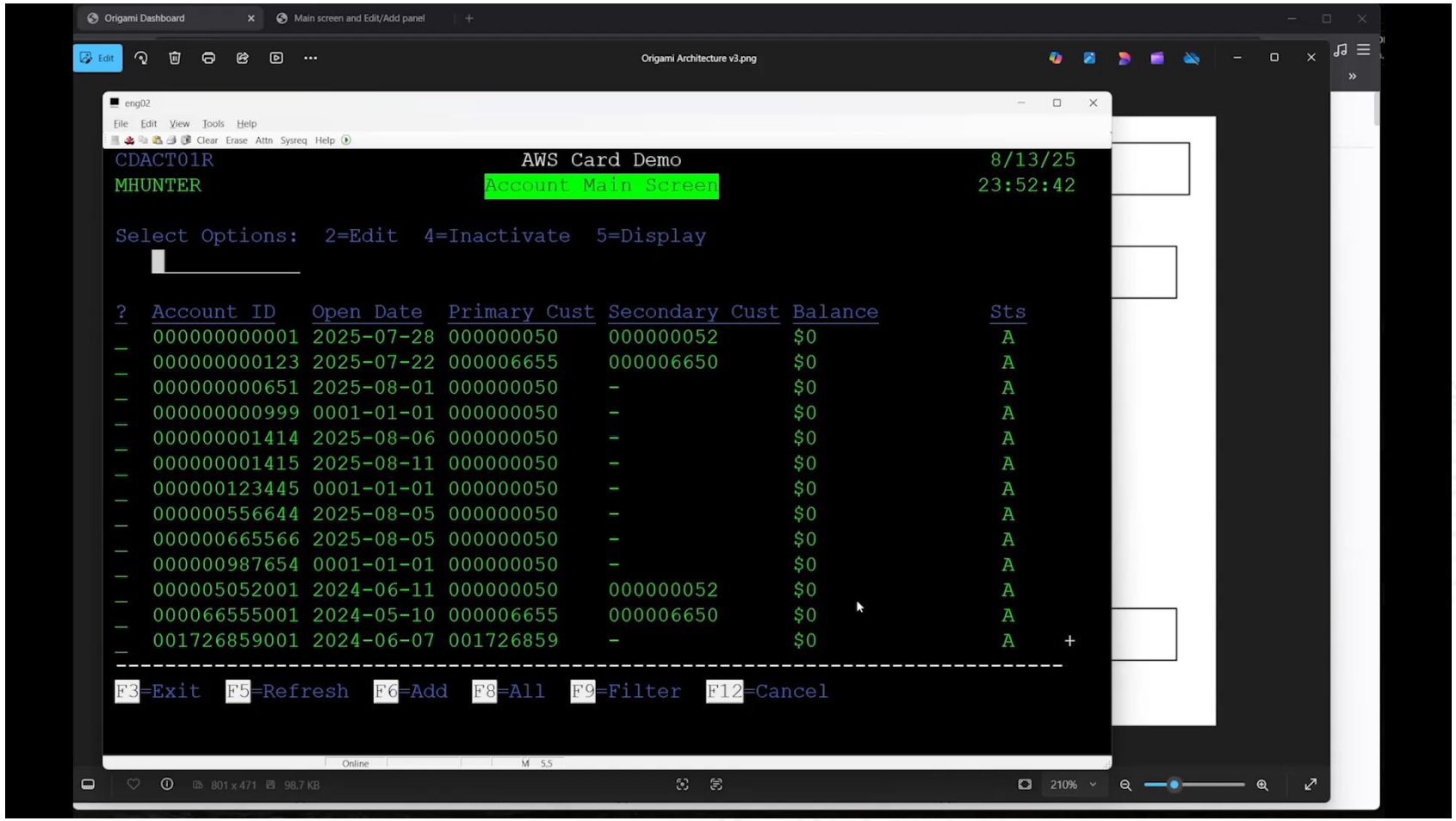
Agentic AI vs. Generative AI



X-Modernize AI Process Flow



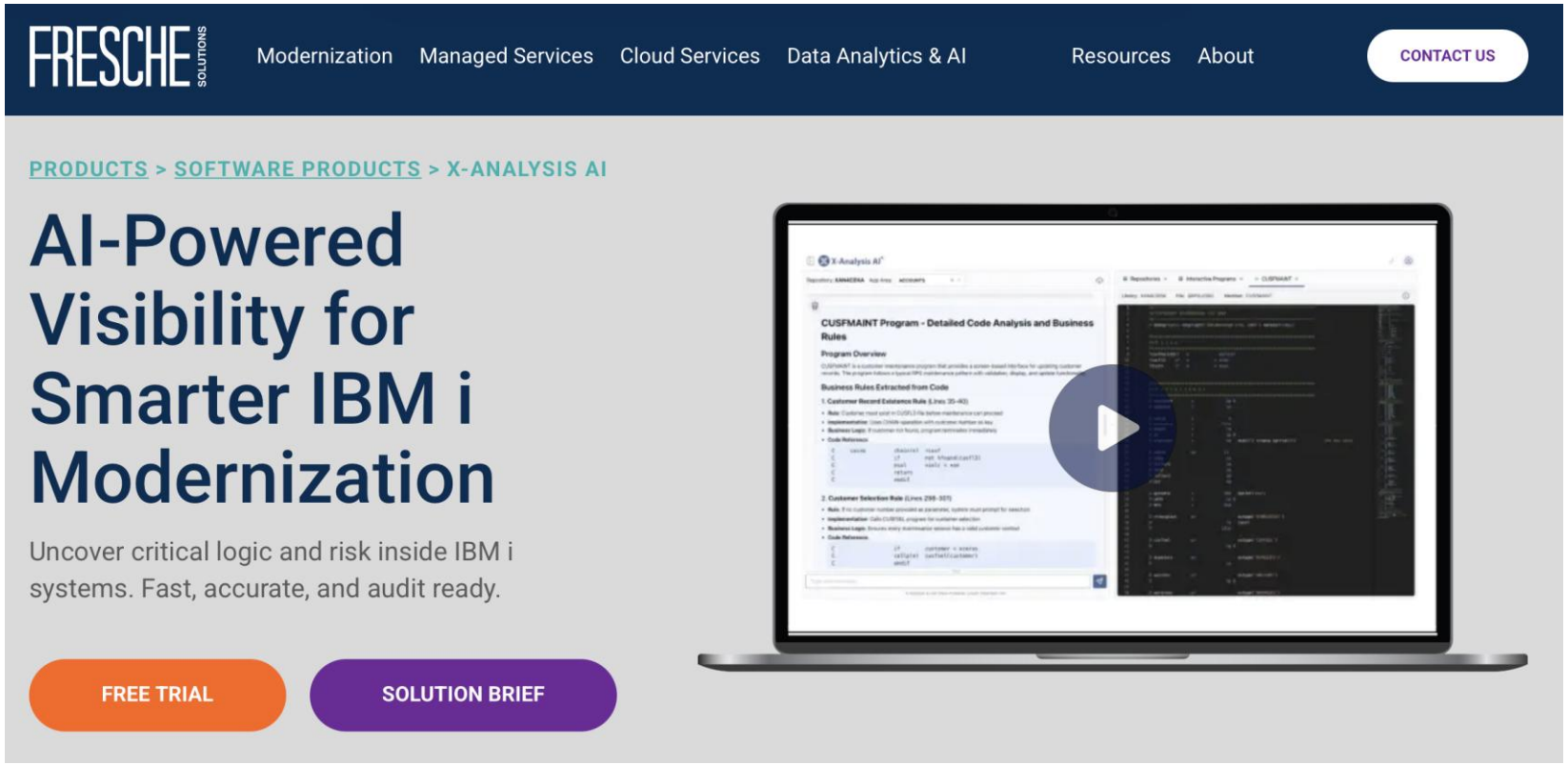
X-Modernize AI Demonstration



Getting Started

 |

Getting Started with X-Analysis AI



- Contact Fresche
- Get X-Analysis AI Trial
  - Comes with API key for LLM model (during trial)

https://freschesolutions.com/products/x-analysis/

Getting Started with X-Modernize AI

- Contact Fresche for a presentation and POC

SAMPLE CODE CONVERSION PROGRAM

Customers can submit a code sample of:

- Their selected application (or a slice of an app)
- All its dependent objects & compliable source code

We select a subset of the submission and modernize using **X-Modernize AI**

FREE POCs:

(+) Teams will ensure the RAW output is demo able (*minor tweaks*)
(+) Customer demo
(+) Converted code is for customer to keep with no obligations
A very small number of programs/lines of code are selected for conversion
No Last-Mile effort (*Only one interface may work*)

Ideal for:

- ***Output quality evaluation***
- ***Gauging efficiency/accuracy of X-Modernize AI.***

PAID POCs:

(+) Teams will ensure the output is fully demo able
(+) Includes last-mile work
(+) Customer demo & discussions around efforts needed/invested
(+) Converted code is for customer to keep with no obligations
(+) A much wider selection of programs/lines of code for conversion
(+) Predicted timeline commitment

Ideal for:

- ***Output quality evaluation***
- ***Gauging ROI & Effort estimates requirement***
- ***Efficiency/accuracy of X-Modernize AI with-in a short period***

Questions?

AI-based Code Understanding and Modernisation. A Vendor Solution presented by Fresche Solutions

Steve Cast

Senior Vice President EMEA and APAC

info@freschesolutions.com

SC