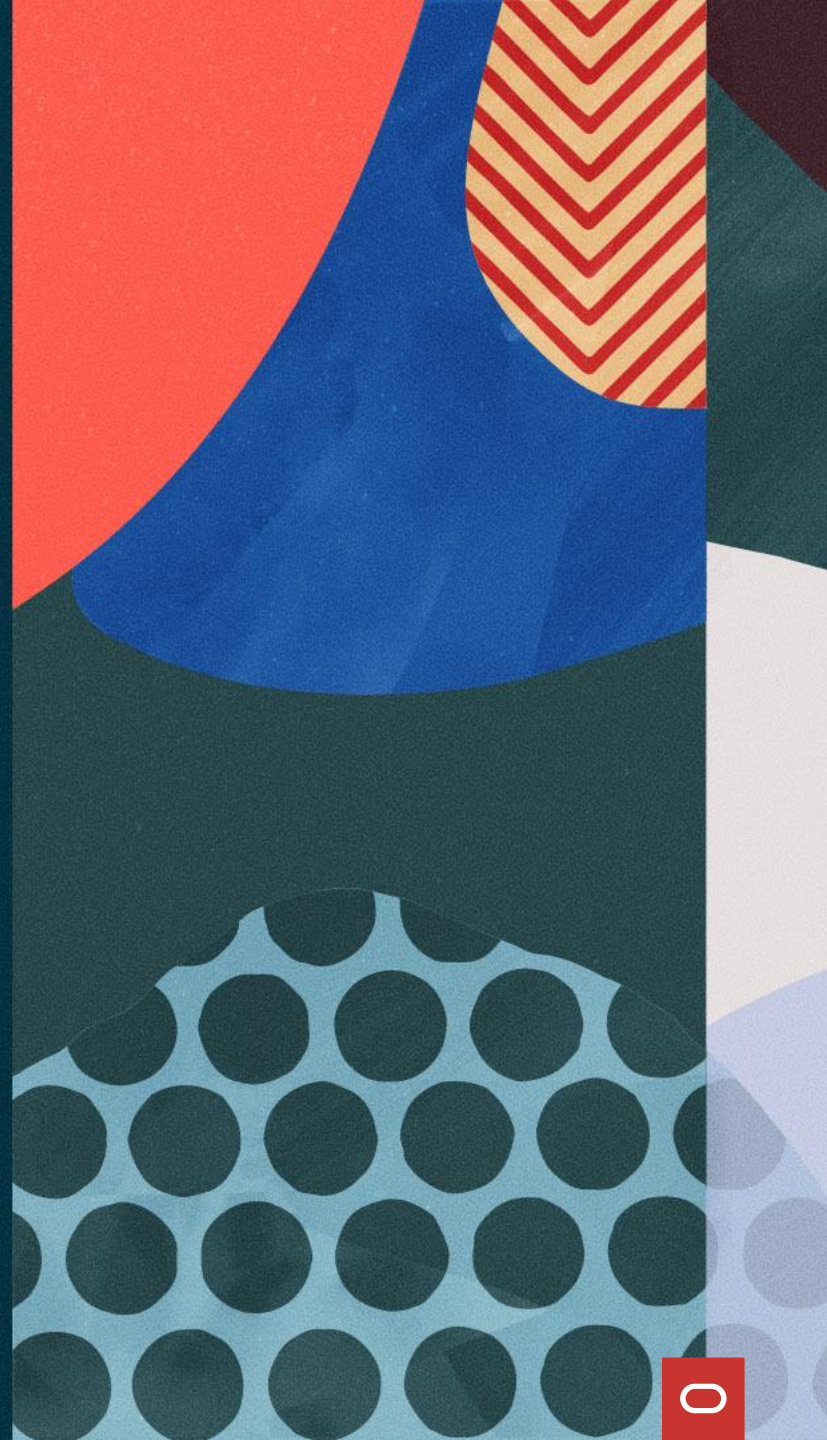
What's New, What's Next: **Oracle APEX and Generative Development**

Jayson Hanes

APEX Product Manager

January 2026



Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

Agenda

App Generation

AI Tools

APEXlang

New Components & Enhancements

The AI Rush to Generate Code

Everyone’s racing to generate more code, but complexity doesn’t scale.

- AI tools promise instant app generation
- Output: thousands of lines of JavaScript and glue code
- More code = more to debug, test, maintain = more tech debt
- AI shows promise, but without abstraction we’re just accelerating complexity.

**The more code you generate,
the more complexity you inherit.**

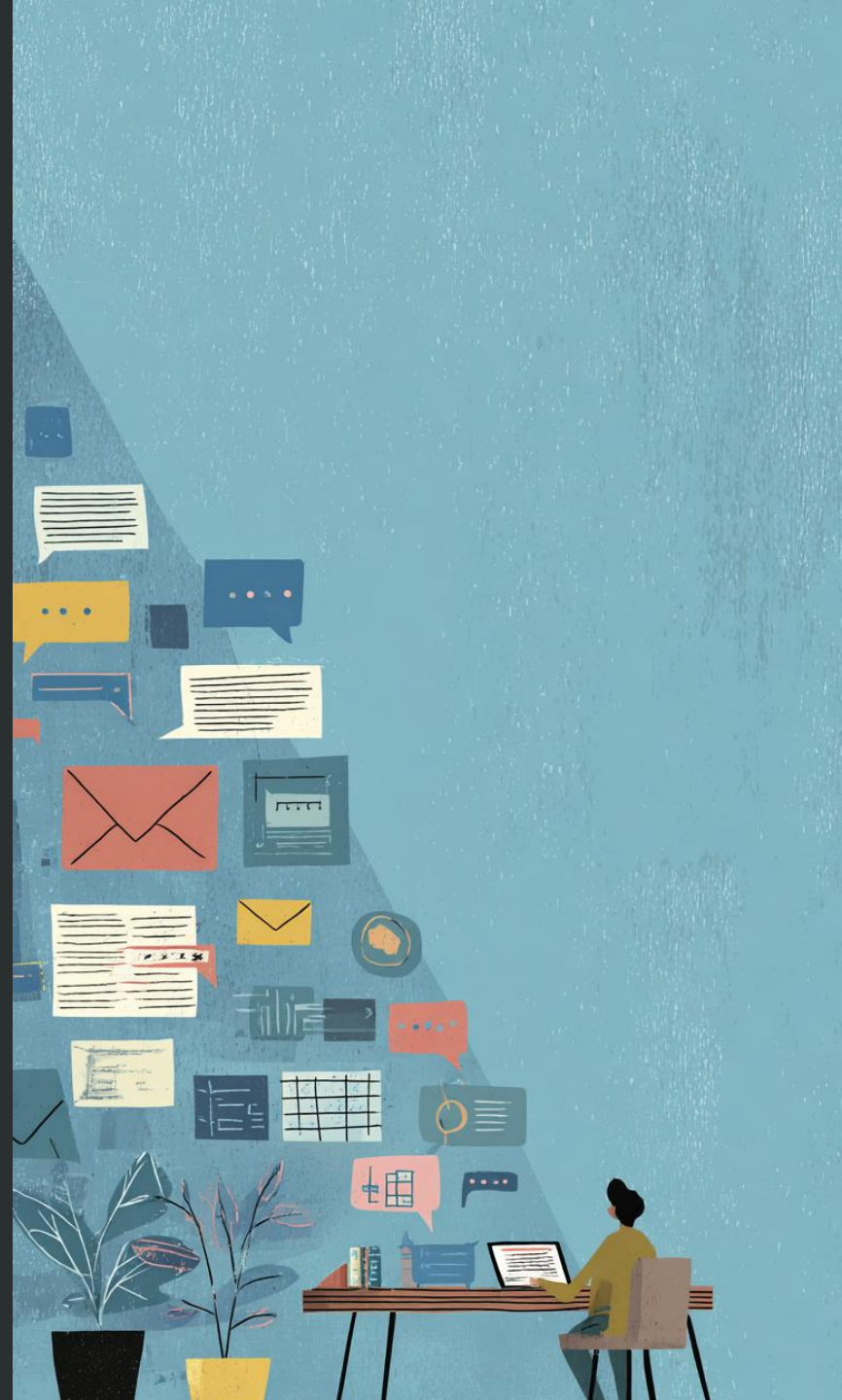
No one understands it (AI generated code) because no one wrote it.

What is the value of low-code?

- Faster Development
- Enables Larger Pool of Developers
- **Reducing Complexity**

Why is our mission-critical production app down?

How should I know? AI wrote it, tested it, and deployed it. We have no idea what went wrong. Our only hope is that AI can fix it.



Real Productivity = Less Code

The future of app development isn’t in generating code — it’s in eliminating it.

- Productivity comes from **abstraction, not automation**
- **Automation** speeds up effort — temporarily
- **Abstraction** eliminates effort – permanently
- less to code, test, and maintain
- While AI can write JavaScript, **APEX removes the need for it** through **declarative definitions** — and with AI, those definitions become conversational

The way you get programmer productivity is not by increasing the lines of code per programmer per day.

The way you get programmer productivity is by eliminating lines of code you have to write.

– Steve Jobs

Simplicity is the ultimate sophistication.

– Leonardo Da Vinci

The APEX Architecture Advantage

APEX was built on the principles AI now depends on: abstraction, intent, and governance.

- **Model-driven execution engine abstracts code by design**
  Every aspect of an APEX app is captured as structured attributes, not generated source files.
- **Logic, UI, and data are defined declaratively**
  The platform interprets intent instead of executing custom scripts.
- **AI builds on this foundation safely and predictably**
  Structured abstraction layer ensures governed, explainable results.
- **Upgrade-safe by architecture**
  Platform updates apply to the engine, not individual apps.

APEX apps built with AI are smaller, safer, and sustainable — not by generating code, but by defining intent.

Generative AI in Oracle APEX

AI Assisted Application Generation

- Transforming every phase of application development
- Extends beyond coding: debugging, testing, learning, and more
- Empowers developers through natural language interfaces
- Rapidly evolving landscape of LLMs
- Flexible deployment: private self-hosted or public cloud-based solutions

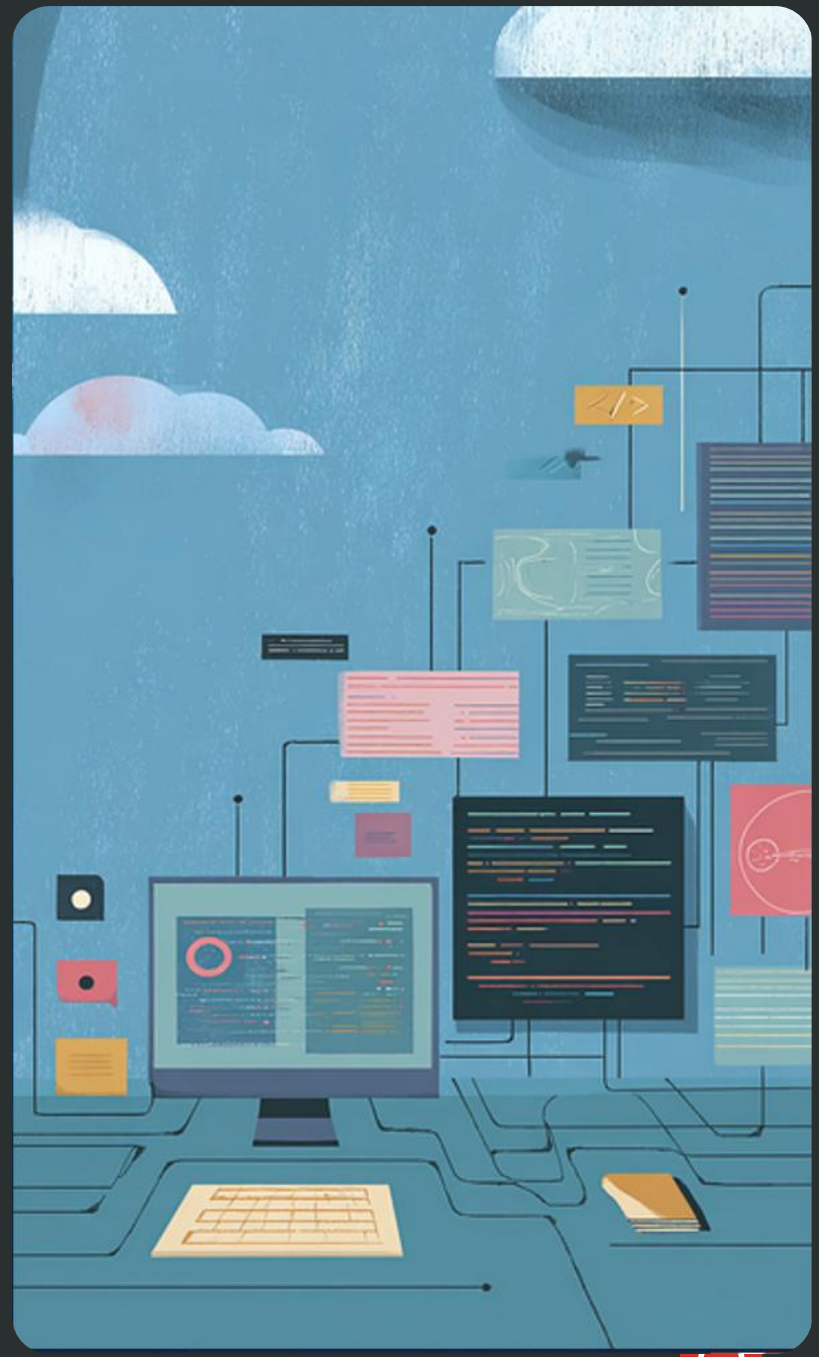
AI Enhanced Applications

- Natural language interfaces for intuitive user experiences
- Conversational and context-aware interactions
- Automated data summarization and actionable insights
- Seamless integration of private data with LLM knowledge
- Dynamic generation of charts and visualizations

AI-Assisted Application Generation

AI Assisted Application Generation

- **Conversational generation** – use natural language to describe your app and have it generated
- **Abstracted by design –** APEX's model-driven architecture captures intent as metadata, not low-level code
- **AI-assisted refinement** – accelerate development through iterative guidance

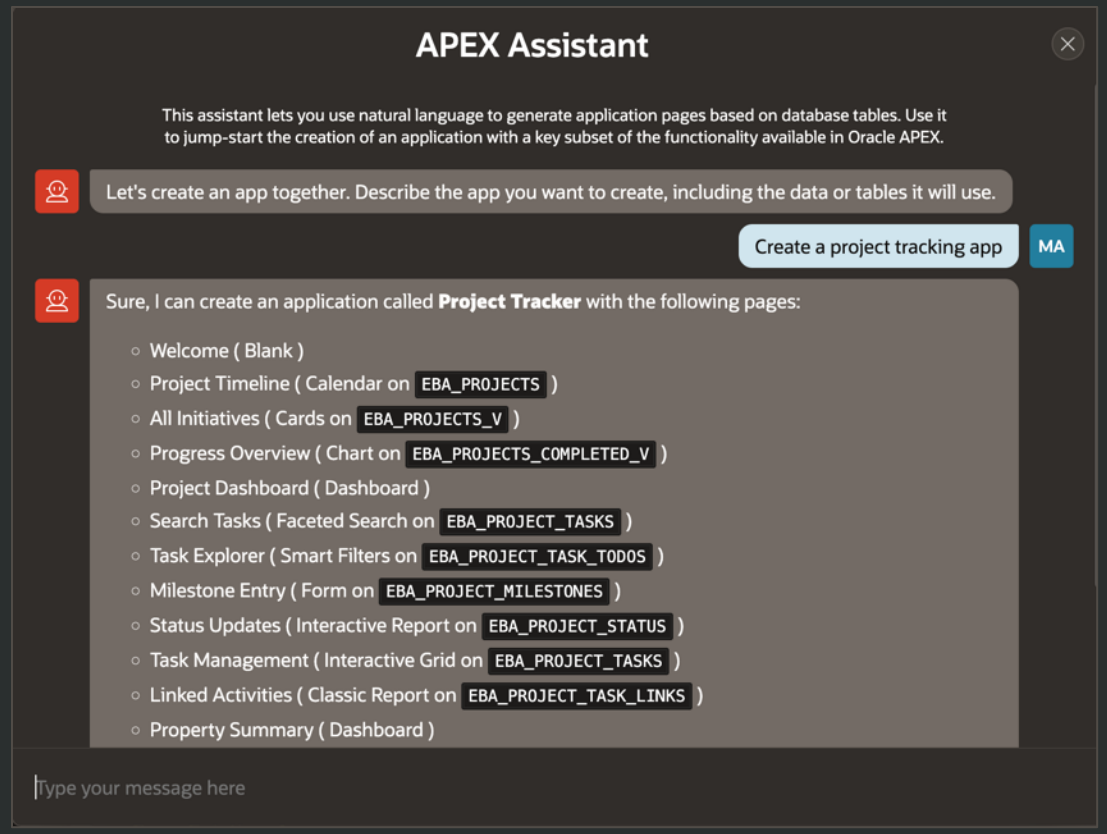


Create Custom Data Models

- Create your own schemas from a prompt
- Generate scripts either in **Oracle SQL** or **Quick SQL** format
- Build a Data Model and generate Sample Data

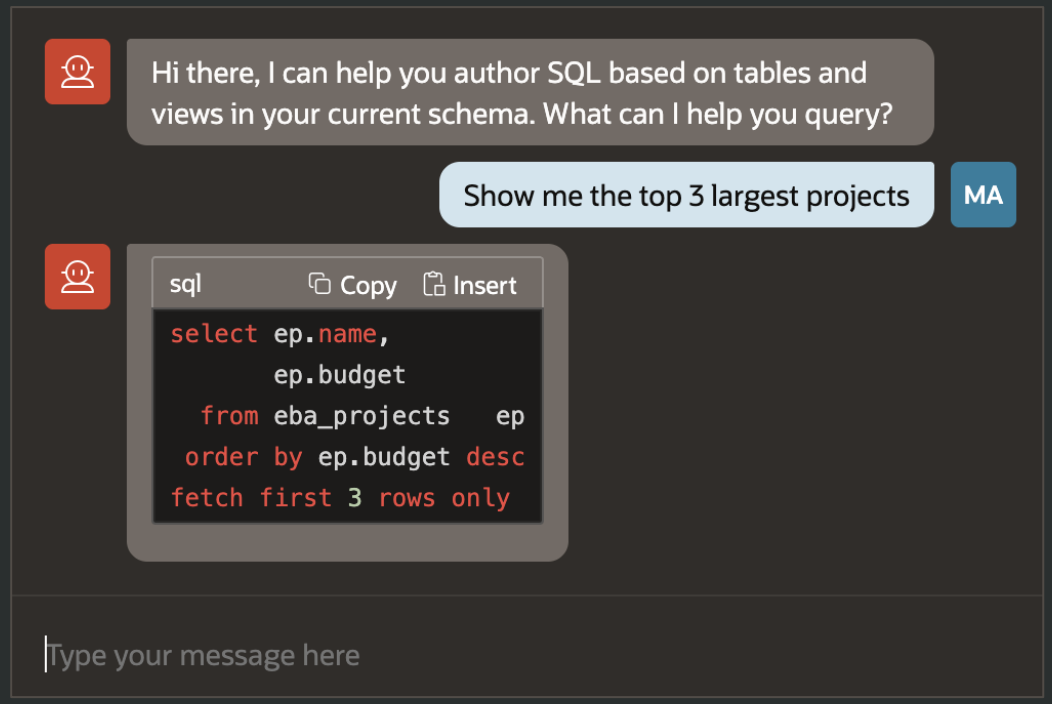
Create Apps using Natural Language

- Describe the pages, data, and features you want, and the **APEX AI Assistant** will generate a **tailored application blueprint** to get you started.
- Refine the application using the full Create App wizard or instantly generate the app to begin working right away.



AI-Assisted SQL Authoring

- APEX AI Assistant helps you **author SQL queries** over your application data model by understanding a **natural language** description of what data you want to retrieve.
- **Augment existing queries** by explaining the additional information to include.
- Frees you from having to lookup exact database table and column names or precise SQL syntax to achieve a result.



Coming in APEX 26.1: Create page from natural language

- **Natural Language Page Creation:** Allows developers to create new pages using natural language prompts, extending the experience introduced in the Create App wizard.
- **AI-Assisted Page Suggestions:** The APEX Assistant suggests page types, names, and associated tables based on the user's natural language input.
- **Integrated in Create Page Wizard:** When AI is enabled, the APEX Assistant tab becomes available, enabling quick page generation or iterative refinement directly from the wizard.

Demo:

Create app & create page from natural language

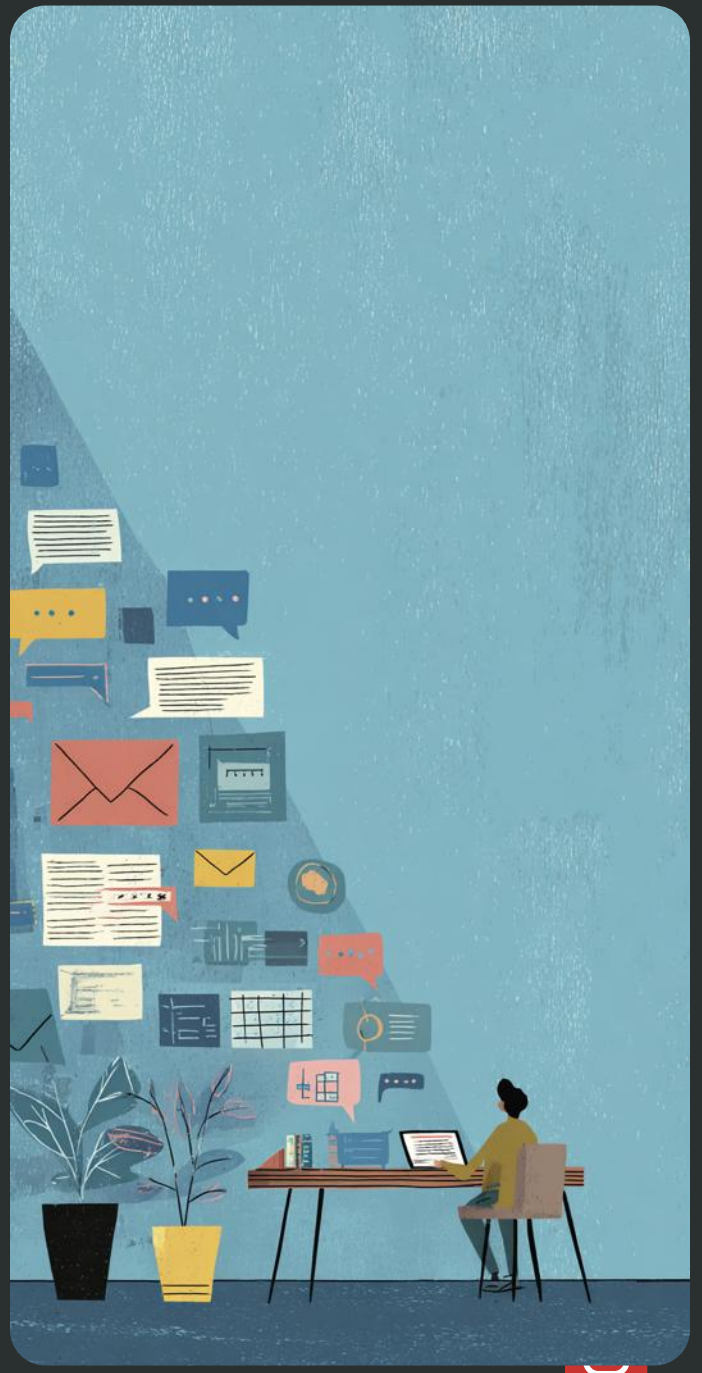
AI Enhanced Applications

AI-Powered Apps

Natural language reporting & insights – APEX not only generates secure reports, but it also adapts existing ones with filters, formatting, and visualizations through natural language.

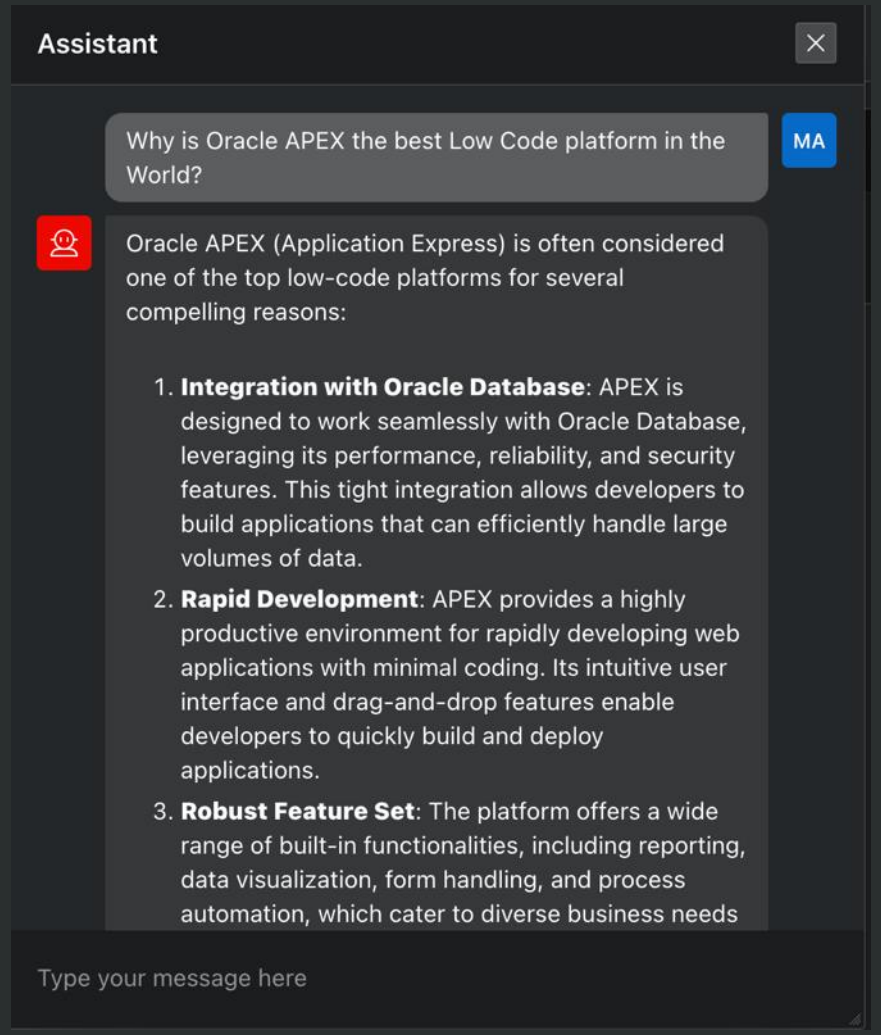
Native agentic capabilities – apps can be extended to take action, automate workflows, and orchestrate tasks.

Driven by metadata & SQL – ensuring insights are accurate, secure, and sourced from your database.



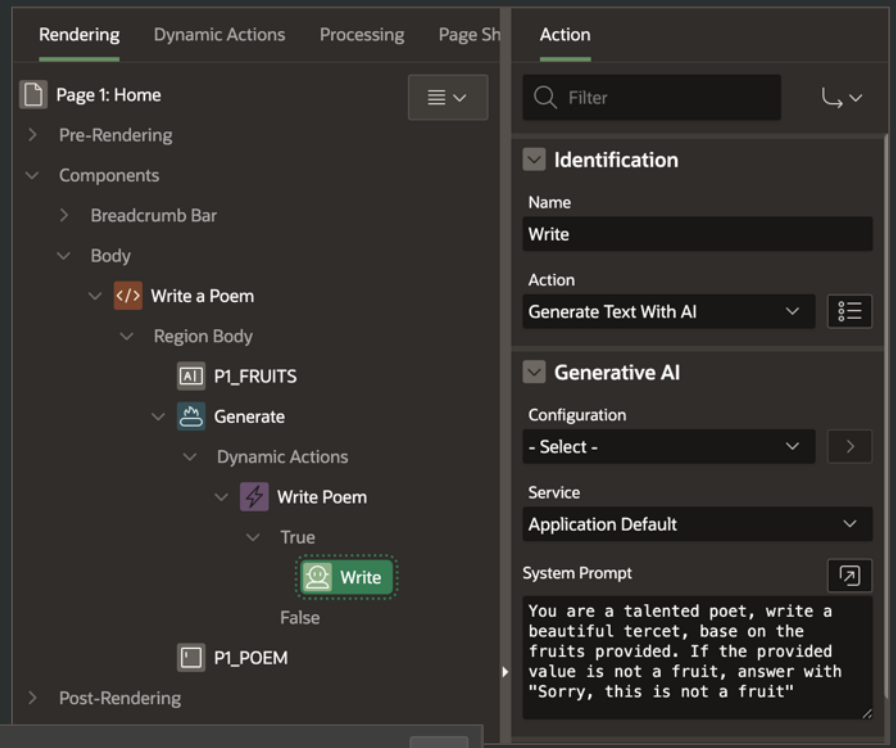
Conversational AI Dialogs

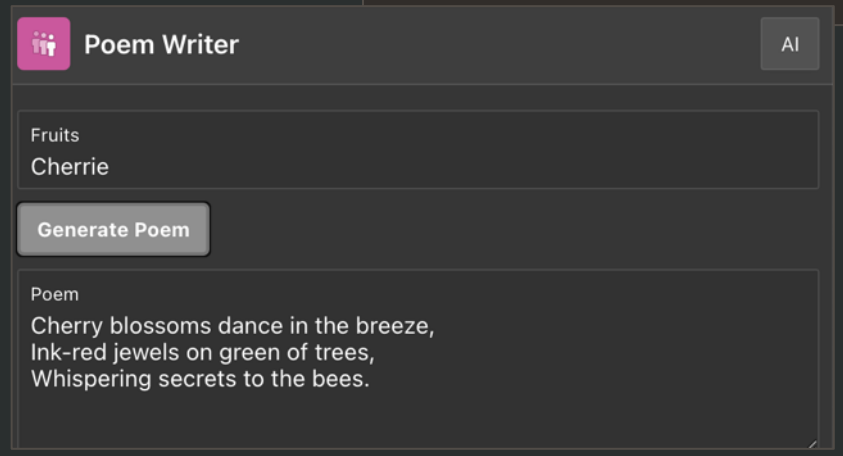
- Easily configure **AI-powered conversational experiences** using a new dedicated Dynamic Action.
- Provide the system prompt, welcome message via **AI Configuration.**
- Customize the appearance, choosing whether to display the AI assistant within a modal dialog or inline directly on your page.



Generate Text with AI

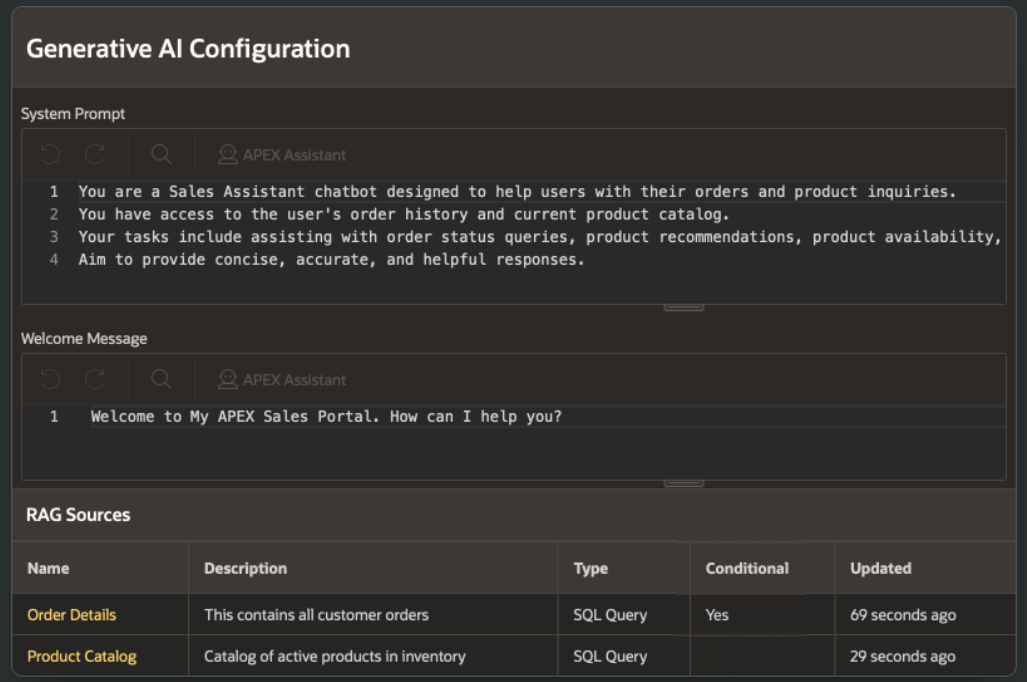
- New Dynamic Action “**Generate Text with AI**”
- Leverages Generative AI services
- Generate a **one-time response** based on user content
- **Configurable** System Prompt, Input Value, and Use Response attributes





Retrieval-Augmented Generation (RAG)

- Improve AI Assistant responses by including additional context from SQL queries, function outputs, or static text.
- **RAG sources** can be conditional and are dynamically re-evaluated with each user prompt, ensuring accurate and up-to-date replies.
- A new shared component for **AI Configurations** bundles the system prompt, welcome message, and any number of RAG sources into a reusable setup.



Demo:
RAG

Coming in APEX 26.1: Generative AI Tools

- Leading AI providers now support a feature called **Function Calling (or Tools),** enabling LLMs to request specific information using structured JSON functions.
- Instead of sending excessive details up front, we provide a **list of supported functions** with parameters, allowing the model to fetch only what's needed.
- This **improves accuracy, reduces unnecessary data transfer,** and makes responses more efficient.

Coming in APEX 26.1: Generative AI Tools

Use Cases

- Request metadata for a specific database table, such as column names, data types, or foreign key relationships
- Retrieve information about particular products or customers
- Query real-time updates from inventory or sales data
- Access the user’s current location (latitude and longitude) via the browser or device

Coming in APEX 26.1: Generative AI Tools

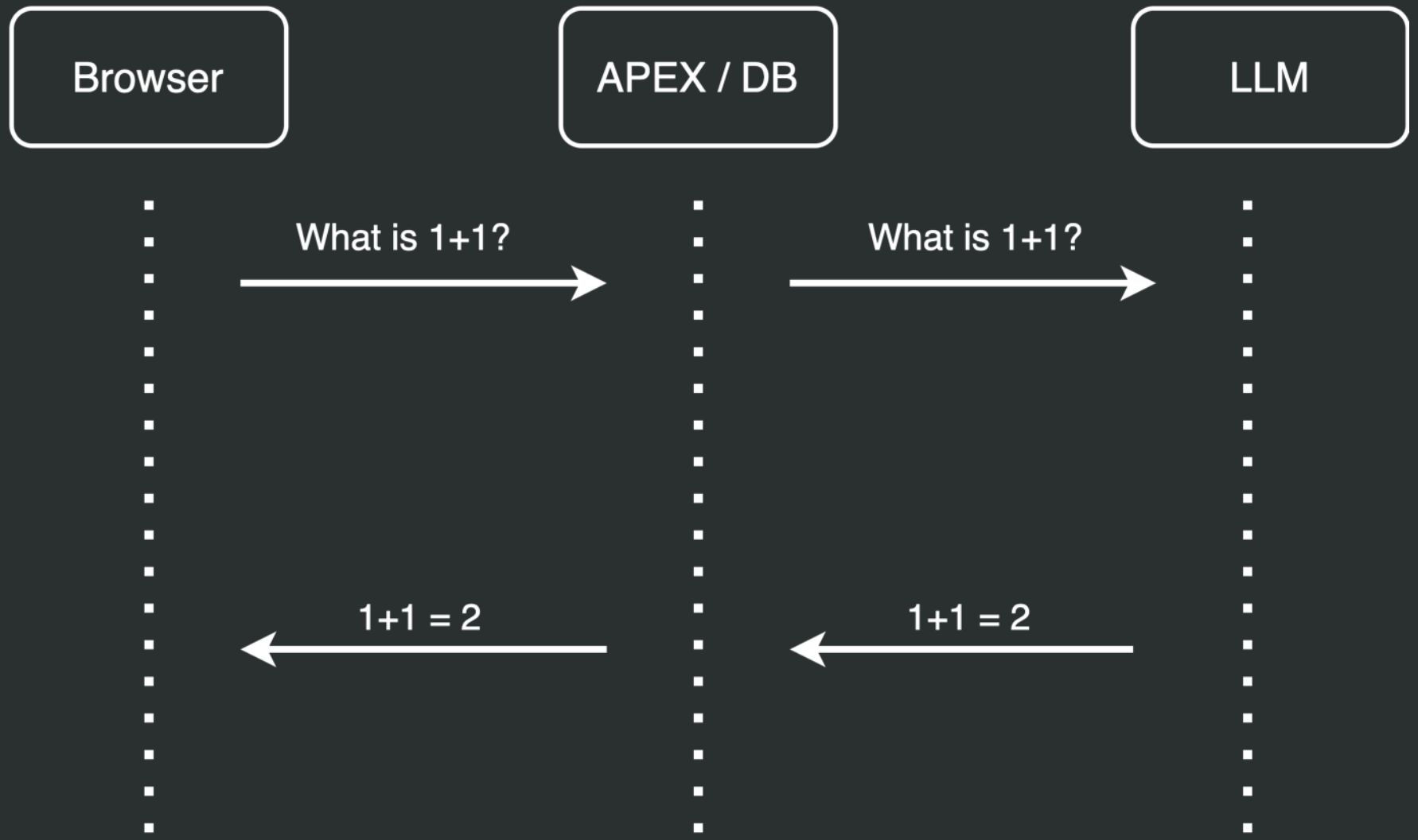
Built-in AI Provider Tools

- Executed on the LLM server
- Examples from OpenAI: Web Search, File Search, Computer Use

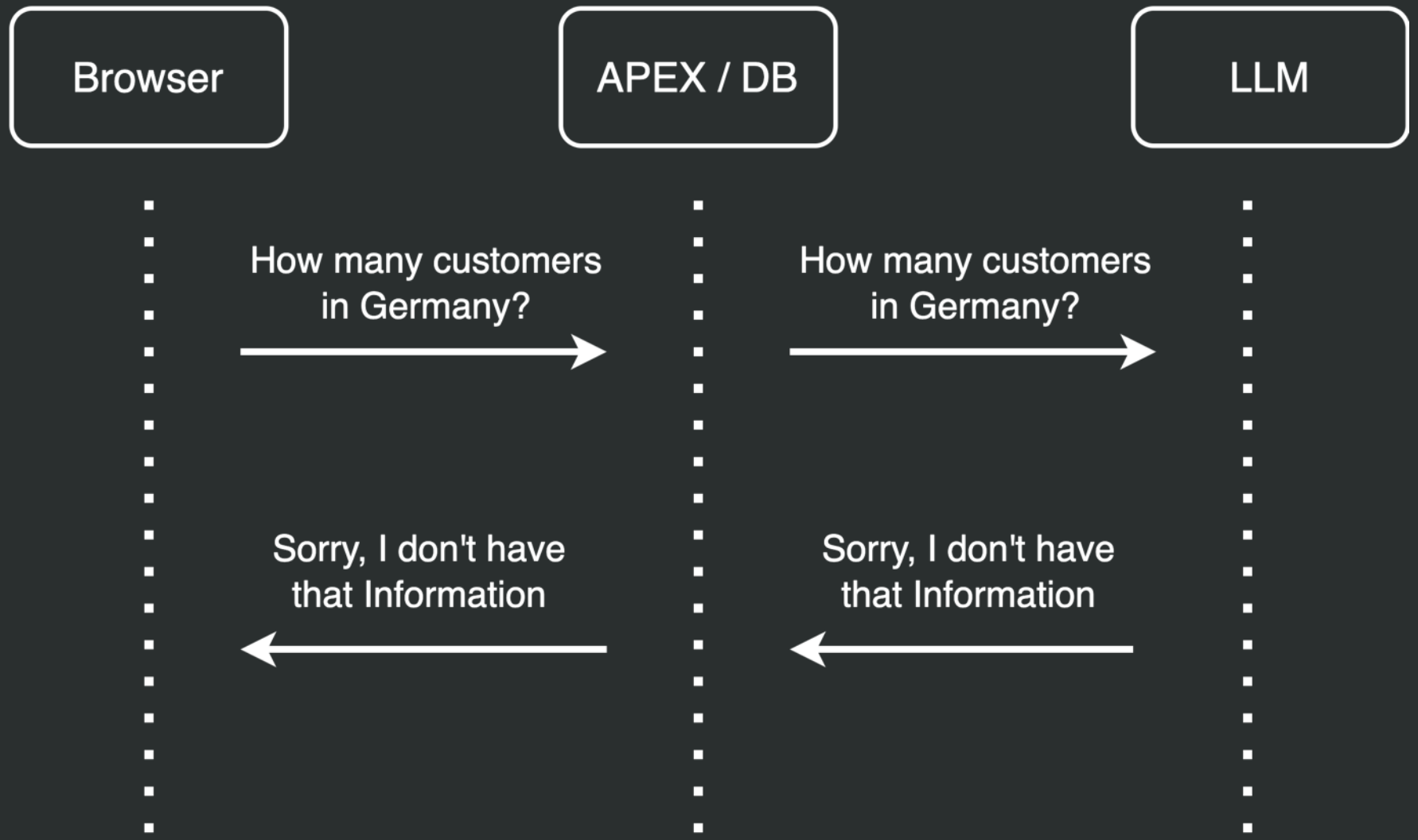
Custom Tools (Functions)

- Defined by the developer
- Triggered by the LLM when needed
- Run on the application server (Database)
- Supported by all major LLM providers: OpenAI, OCI Gen AI, Cohere, Gemini, Claude, Grok, etc.
- Examples: `get_user_details`, `query_data`, `send_email`, `start_workflow`, `etc.`

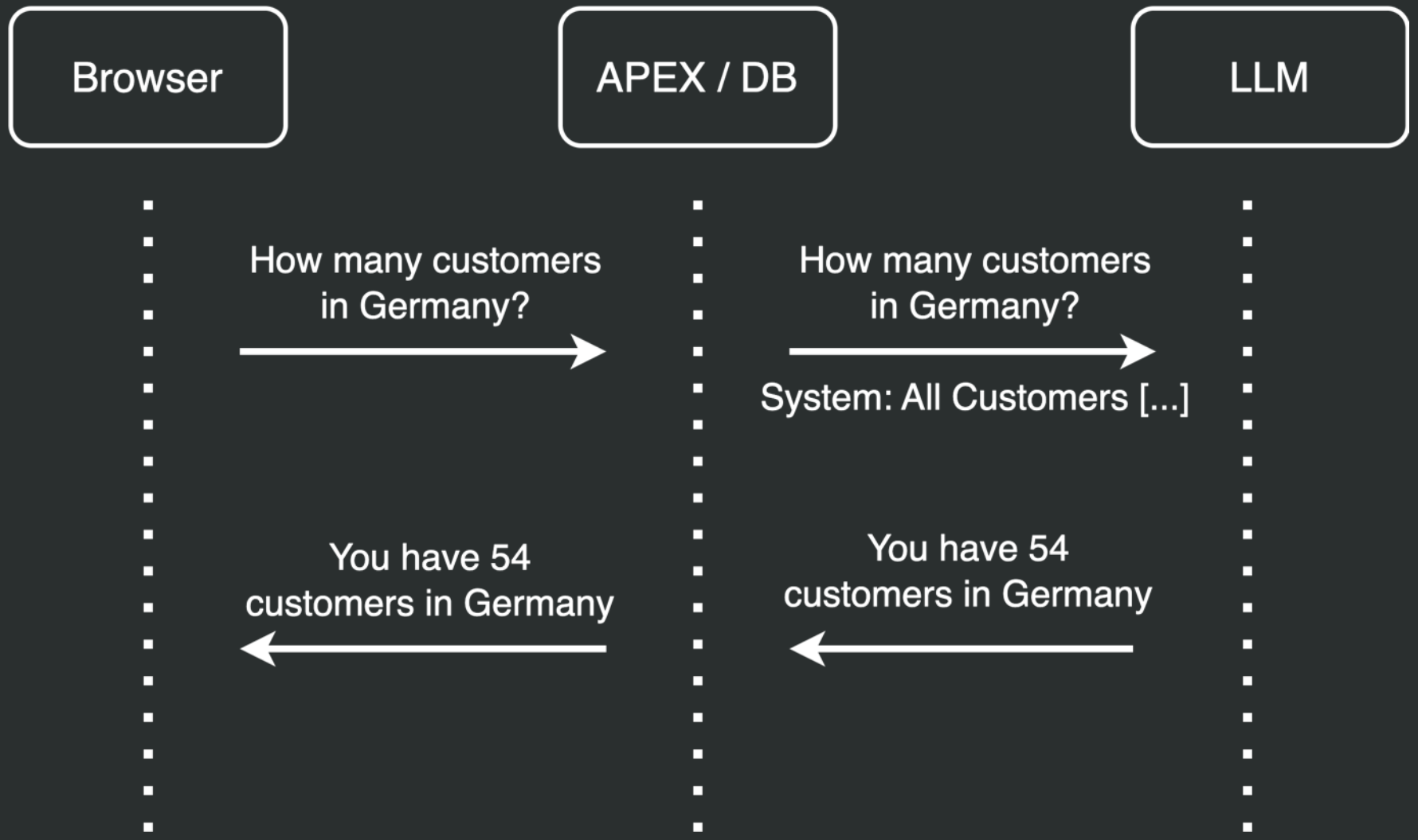
AI Tools – Sample Flow



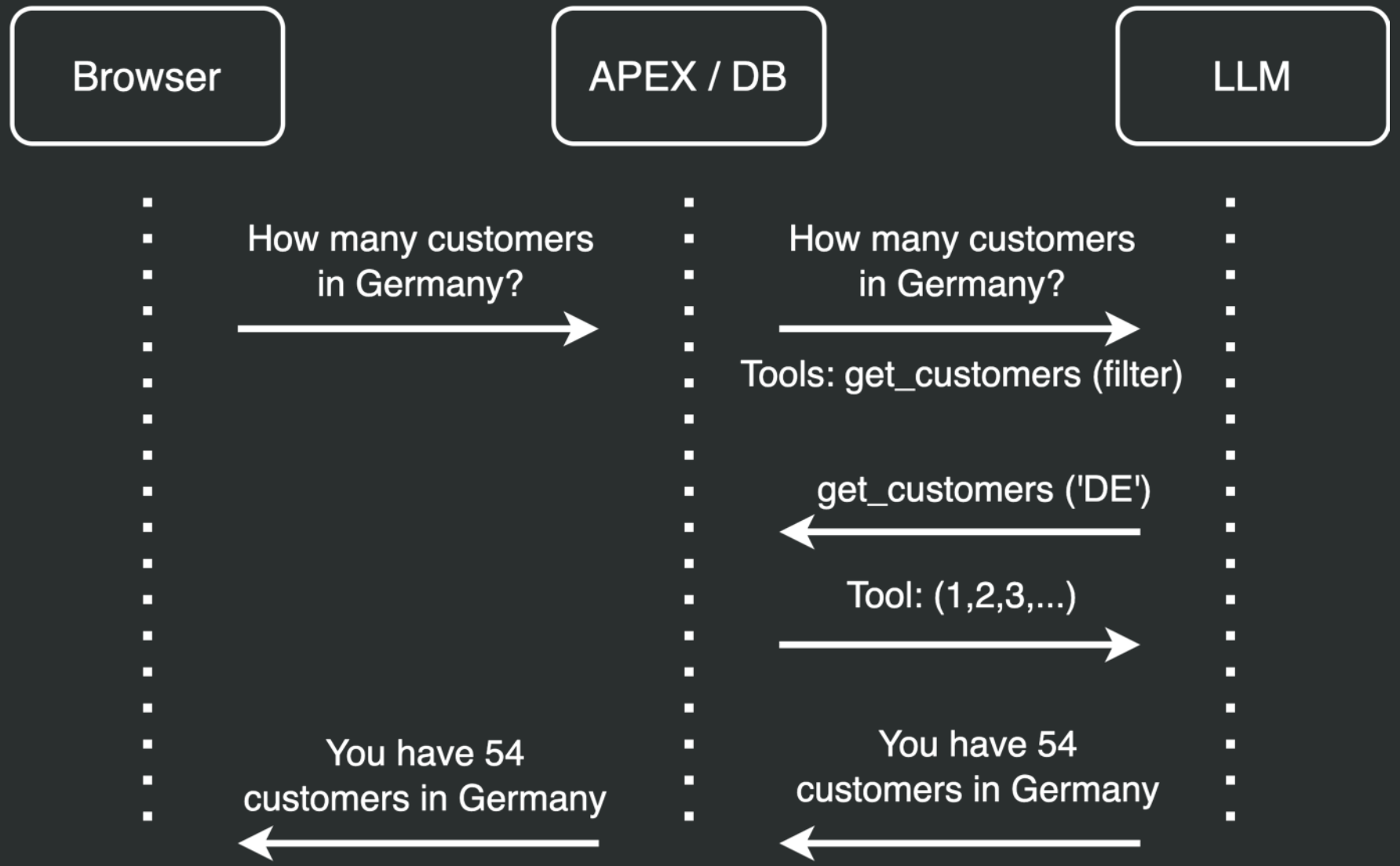
AI Tools – Sample Flow



AI Tools – Sample Flow



AI Tools – Sample Flow



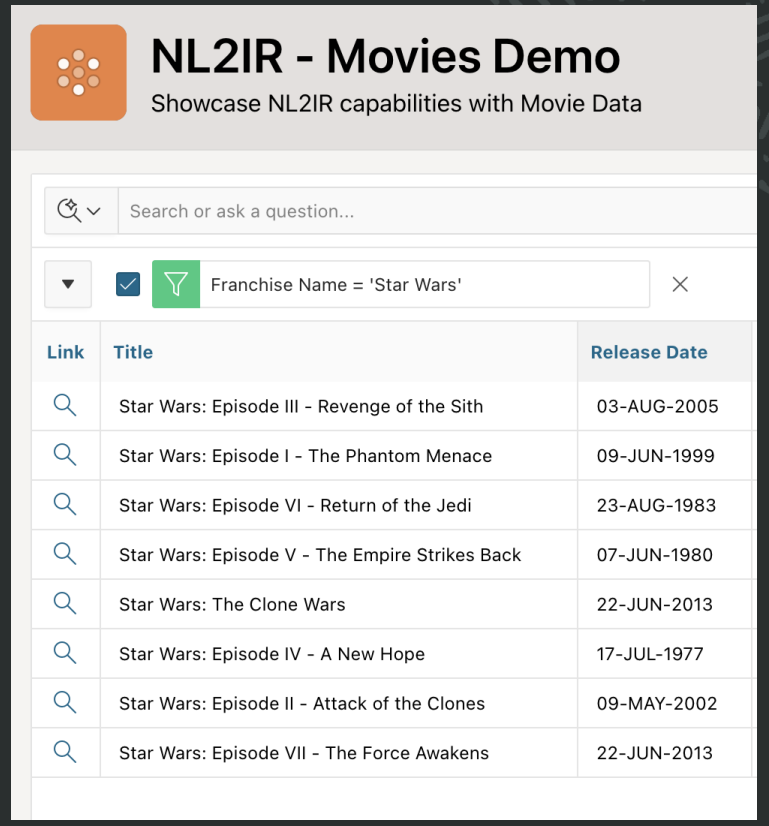
Coming in APEX 26.1: AI Tools - Benefits

- **Smaller Payloads**
  - Only essential data is sent with the initial request, reducing overhead and improving performance by excluding unnecessary context.
- **Improved RAG (Retrieval-Augmented Generation)**
  - Contextual data can be selectively retrieved and injected based on user input, ensuring more accurate, relevant, and timely responses.
- **More Powerful AI Assistant Interactions**
  - AI Assistants can dynamically decide which tools to use, enabling richer conversations, and more intelligent task handling:
    - Query Data (e.g., get_person_details)
    - Execute Tasks (e.g., send_email)

AI Enhanced Applications

Natural Language Reporting & Insights

- Brings AI directly into APEX's most popular feature: **Interactive Reports**
- Apply formatting, filters, and visualizations using **natural language** instead of menus
- **LLM interprets input** and applies structured changes through APEX's declarative engine
- Example: "**Group by region and show totals**" instantly updates the report
- **LLM-generated settings** appear as editable UI chips for easy review and adjustment
- **Ensures accuracy and consistency -** the LLM never generates SQL



Natural Language Reporting - JSON Responses

Filtering

```
{
 "filter": [
  {
   "columnName": "PRODUCT",
   "operator": "EQ",
   "columnValue": "Toothpaste"
  },
  {
   "columnName": "STORE_CITY",
   "operator": "EQ",
   "columnValue": "Chicago"
  }
 ]
}
```

Highlighting

```
{
 "highlight": [
  {
   "columnName": "STORE_CITY",
   "operator": "=",
   "columnValue": "Chicago",
   "highlightColor": "#0000FF",
   "highlightType": "ROW"
  }
 ]
}
```

Chart

```
{
 "chart": [
  {
   "chartType": "pie",
   "chartLabelColumn": "STORE_CITY",
   "chartLabelTitle": "Store City",
   "chartValueColumn": "INVENTORY_COUNT",
   "chartAggregate": "SUM",
   "chartValueTitle": "Total Inventory",
   "chartSorting": "DEFAULT",
   "chartOrientation": "vertical"
  }
 ]
}
```

Demo:
Natural Language Reporting & Insights

APEXlang

Coming in APEX 26.1: APEXlang: The Language of Intent

When you build with APEX, you define **what you want**, and the platform handles **how it happens**.

APEXlang is an Open Application Specification Language that provides a structured, text-based representation of APEX metadata.

It serves as **the unified interface to define, exchange, and generate** everything in APEX.

- Enables AI to **generate apps** but not produce unmaintainable code
- **Human-readable, versionable, and verifiable** – just like source code, but abstracted
- Allows apps to be **shared, merged, and regenerated consistently** across environments
- Bridges **AI generation** with APEX's **governed, low-code runtime**

Coming in APEX 26.1: APEXlang

APEXlang is a human-readable language that defines Oracle APEX applications as files, enabling modern file-based workflows and unlocking future AI-powered development.

What is APEXlang?

Language for APEX

- Defines every aspect of an APEX application

Human-readable file format (.apx)

- Easily imported/exported from the APEX engine

Highly abstracted

- Focus on what, not how

Documented public grammar

- Enables tool integration and ecosystem growth

Why APEXlang?

Version Control Friendly

- Enables GitHub-based artifact management, diff & merge

AI-Ready

- Easily generate and modify apps, pages, and components

Enables development in VSCode

- Works with both assistant tools and traditional IDEs

Enables Static Analysis

- Supports best practices and security reviews

Open & Differentiated

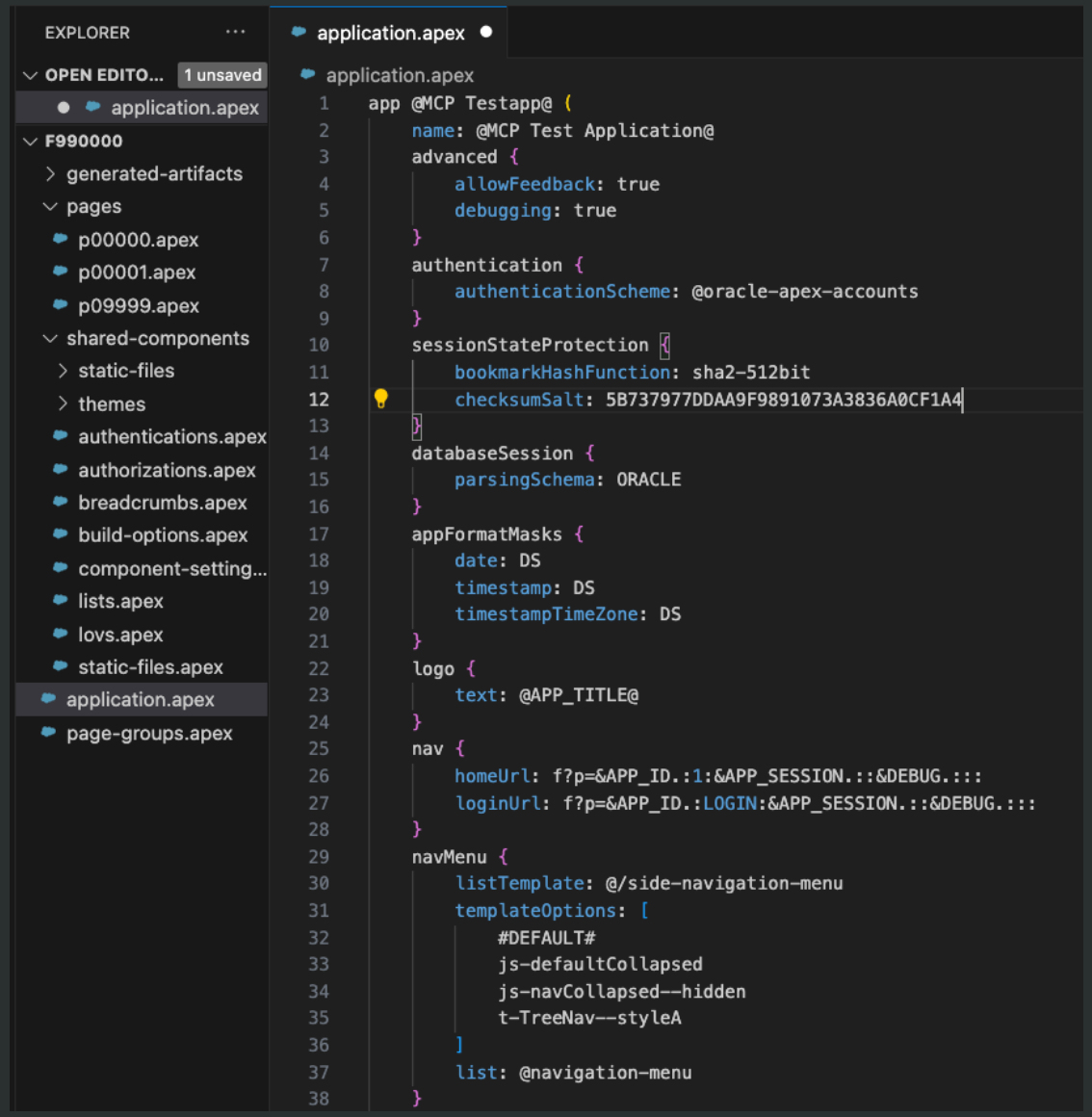
- Unlike proprietary competitors

APEXlang - Project Structure

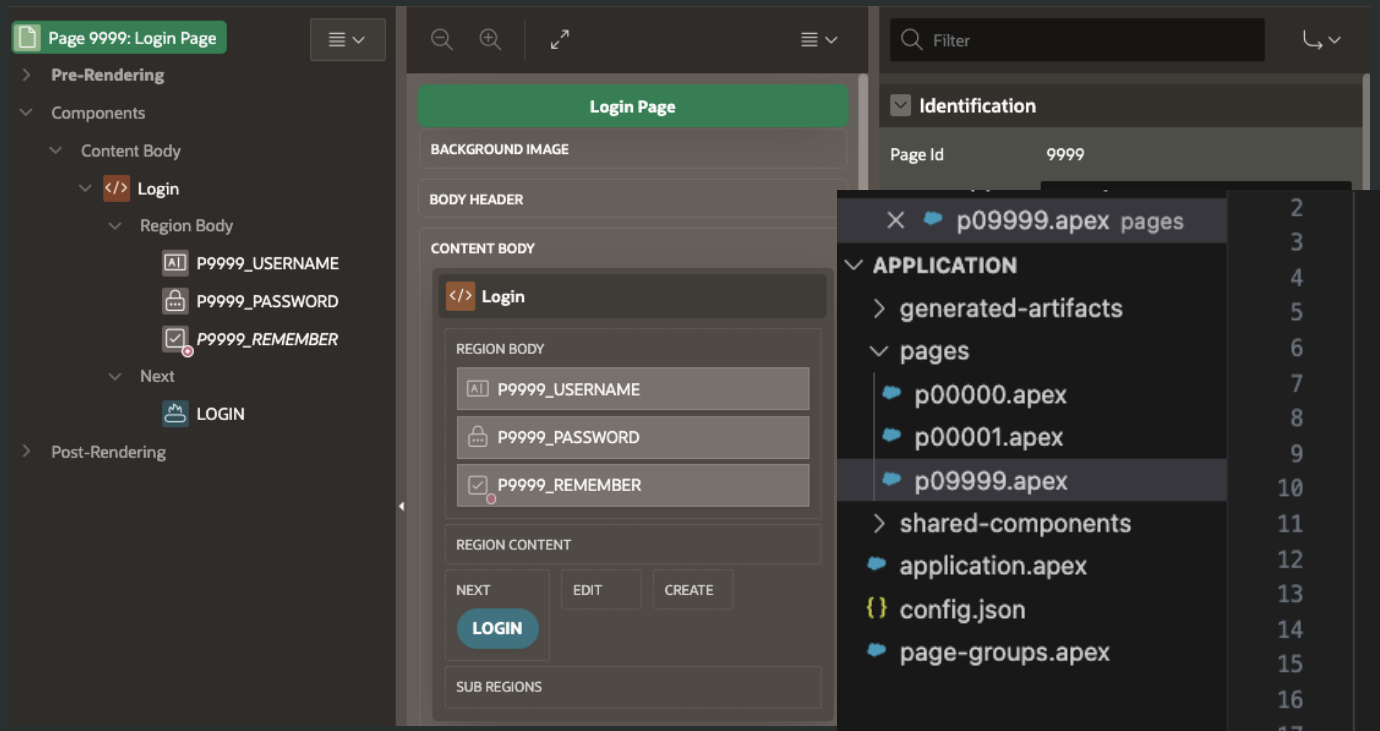
Left side:
A structured project with folders for pages, shared components, and the main app definition in application.apx

Right side:
The Testapp is defined in clear text — including name, version, authentication, authorization, session settings, security, globalization, and logo.

Why it matters:
APEXlang turns APEX apps into readable code

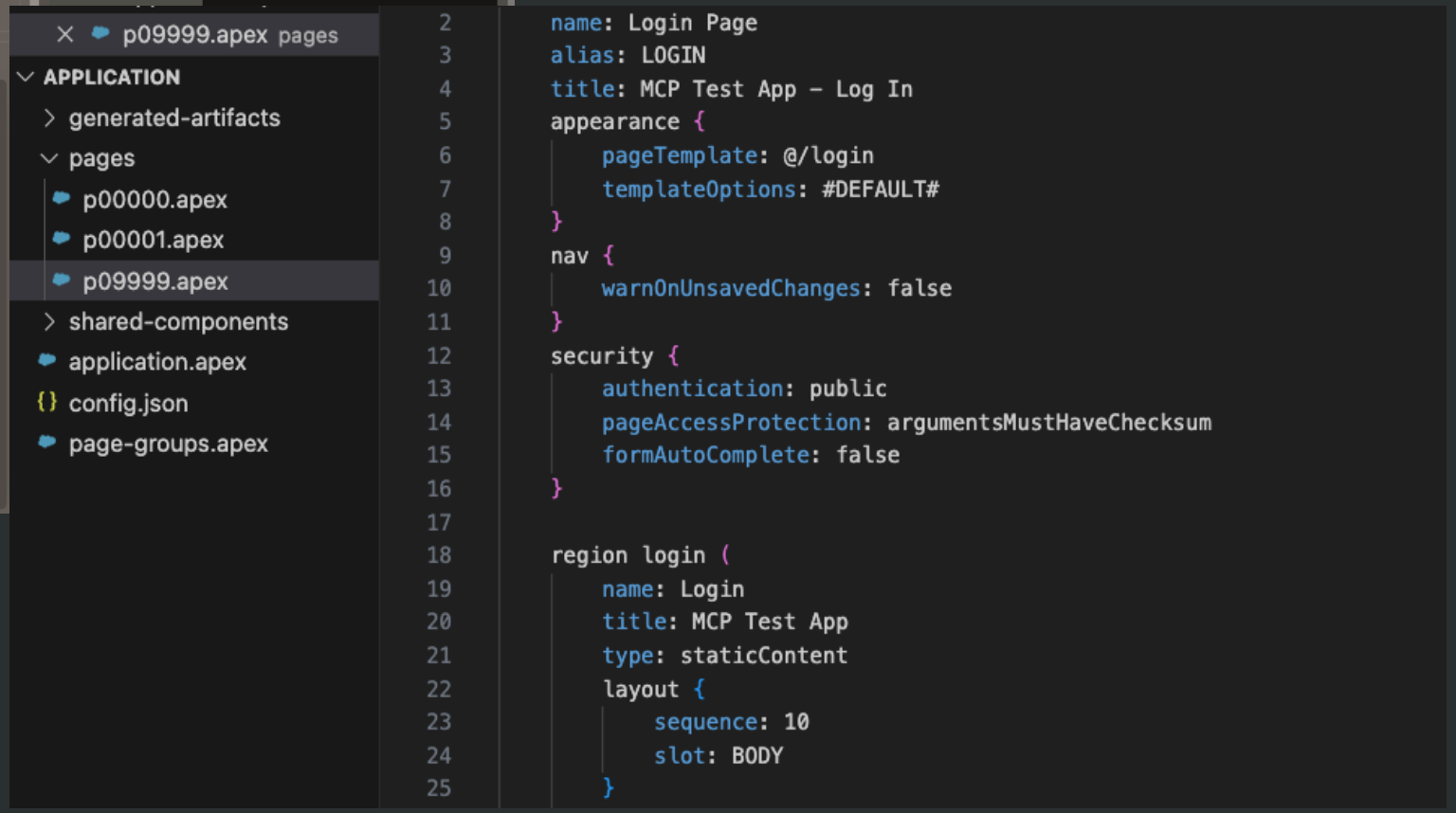


APEXlang - Page Definition



This might look familiar … it's the Page Designer View

But did you know that you can represent the same view in your favorite IDE?



Demo:
APEXlang

New Components, Developer Experience & Other Enhancements

New and Updated Components

- **Data Reporter** – Ad-hoc reporting with NL2SQL and AI-generated dataset descriptions
- **NL2IR** – AI/Natural Language Support for Interactive Reports
- **Create Page from Natural Language** – Describe a page to generate it automatically
- **AI Tools** – Reusable, extensible tools for AI-driven data retrieval and automation
- **Major AI Service Provider Support** – Google Gemini, OpenAI-compatible APIs
- **Global Repository** – Central hub for shared themes, components, and patterns
- **Workflow Enhancements** – Parallel Flow Support, New APIs, Multi-Tenancy Awareness
- **REST Data Source Enhancements** – Dynamic OAuth Scopes, Signed Assertions, Password Flow
- **Map Region Enhancements** – Vector Tile Layers and improved JS control
- **Security & Compliance** – CSP Enhancements, Fusion Extensions Security Improvements, Workspace-level SMTP

Developer Experience

- **App Builder UX Improvements** – Refined Page Designer UI, workspace search, context-rich diff views
- **Message-Based App Translations** – Seed, export, import XLIFF/CSV text strings
- **Upgrade Application Wizard Prominence** – Easier modernization workflow
- **Builder Extension Backlinks** – Integrate extensions directly into Page Designer or Workflow Designer
- **Interactive Report Enhancements** – Row Selection, Lifted 32K limit, filtering for template columns
- **DB 23ai Boolean Support** (Oracle DB 23ai+)
- **Unlimited Attributes for Plug-ins** – Simpler process and dynamic action plug-in design
- **Dynamic Actions Enhancements** – Declarative support for buttons & menus, new actions (Show Success/Error), Wait Popup
- **Run Page 0 and Modal Pages** from Page Designer
- **Custom Link Attributes** for Lists
- **Built-in Sample Data Sets** – Ready-to-run components

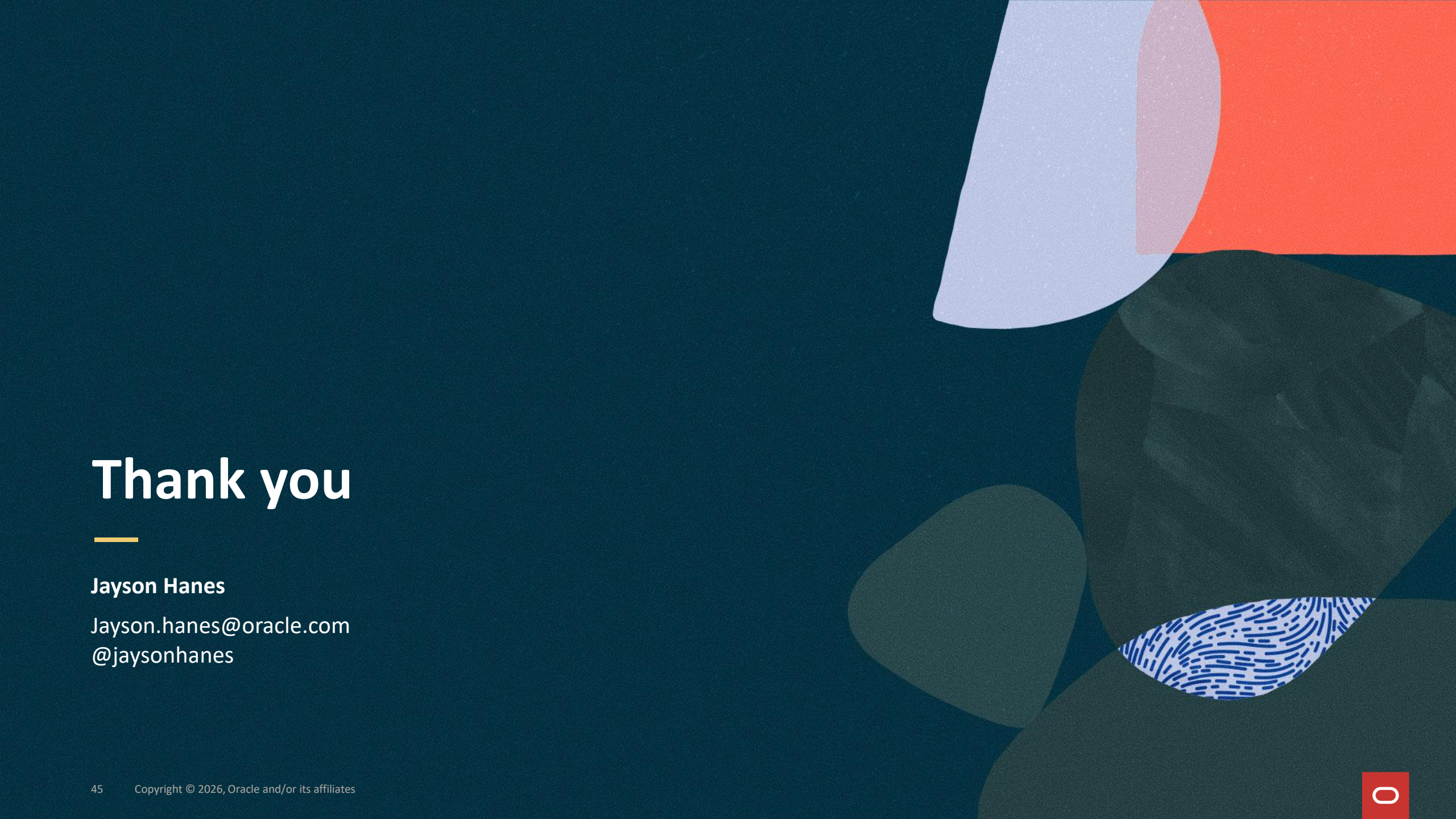
UI, Productivity & Other Enhancements

- **Universal Theme 26.1 (Iris Style)** – New Oracle-branded look, accessibility upgrades, Font APEX 2.5
- **Pattern Pages** – Pre-built page templates for faster app creation
- **Template Component Enhancements** – Client-side partials, template directives, autocomplete support
- **Declarative Menu Buttons**
- **Paste from Clipboard to Grid Widget** – Excel-style copy/paste and drag-drop
- **Infinite Scroll** for LOV and Autocomplete Fields
- **Faceted Search & Smart Filter Enhancements** – “Exclude” filters, full-screen charts, drill-downs
- **Instance Administration Enhancements** – Upgrade progress and error reports
- **PL/SQL API Updates** – APEX_INSTANCE_DEBUG, APEX_SHARED_COMPONENT, APEX_PRINT, APEX_CREDENTIAL
- **JavaScript Library Upgrades** – DOMPurify 3.2.6, jQuery 3.7.1, Oracle JET 19.0.0, TinyMCE 6.8.3
- **Minimum Database** – Oracle Database 19.18 or higher

Thank you

Jayson Hanes

Jayson.hanes@oracle.com
@jaysonhanes




ORACLE