

Building Secure AI Access to SQL Server with MCP

AT A GLANCE

Challenges

- User-level authentication for data access
- Security risks from natural-language-to-SQL
- Inconsistent MCP architectures
- Testing non-deterministic AI agent behavior

Methods

- MCP with .NET Aspire and Azure API Management
- Entra ID and On-Behalf-Of authentication
- .NET MCP server templates and SQL access patterns
- Software, agent behavior, and security testing practices

OBJECTIVES

A large enterprise organization wanted to make critical SQL Server data more accessible through Microsoft Copilot Studio agents while maintaining enterprise security and governance. The organization needed to ensure that users could access data according to their existing permissions, while preventing unrestricted SQL generation and maintaining control over what data and operations AI agents could access.

IntelliTect's objective was to establish a secure, scalable architecture for connecting SQL Server to Copilot Studio through Model Context Protocol (MCP), while creating reusable development patterns that multiple teams could apply to future AI initiatives. The engagement also addressed a growing challenge with AI-enabled applications: traditional software tests can verify that tools work correctly, but they do not necessarily determine whether an AI agent can discover, understand, and use those tools effectively.

SOLUTIONS

IntelliTect developed a standardized architecture covering infrastructure, authentication, governed data access, custom MCP development, testing, and security. The solution used .NET Aspire with Azure API Management for infrastructure and observability, while Microsoft Entra ID and the On-Behalf-Of flow preserved individual user identities and permissions throughout the request chain. Data API Builder provided a governed, configuration-driven approach for exposing approved SQL Server data without allowing arbitrary SQL generation. For more complex scenarios, IntelliTect created reusable .NET MCP server templates using the official C# MCP SDK, Entity Framework Core, and parameterized SQL. IntelliTect also established testing practices covering both deterministic MCP tool behavior and AI agent interactions, while conducting security reviews of existing MCP implementations across multiple teams.

BENEFITS



Governed AI Data Access

Established secure, repeatable patterns for connecting agents to SQL Server without exposing unrestricted database access.



Reusable MCP Architecture

Standardized patterns that multiple teams can apply to future AI initiatives.



AI Security and Reliability

Improved security while introducing testing for both MCP functionality and agent behavior.

