

Application Modernization: 5 Answers You Need to Know



The Challenge

Are you thinking about microservices? Are you wondering if the codebase is salvageable? Are you trying to figure out if the code can be untangled and modernized for the cloud? Would development time and resources be saved by rewriting it?

These questions can take development teams months of costly analysis to answer as they try to understand the interconnectivity and complexity that exist within a codebase.

Our Solution

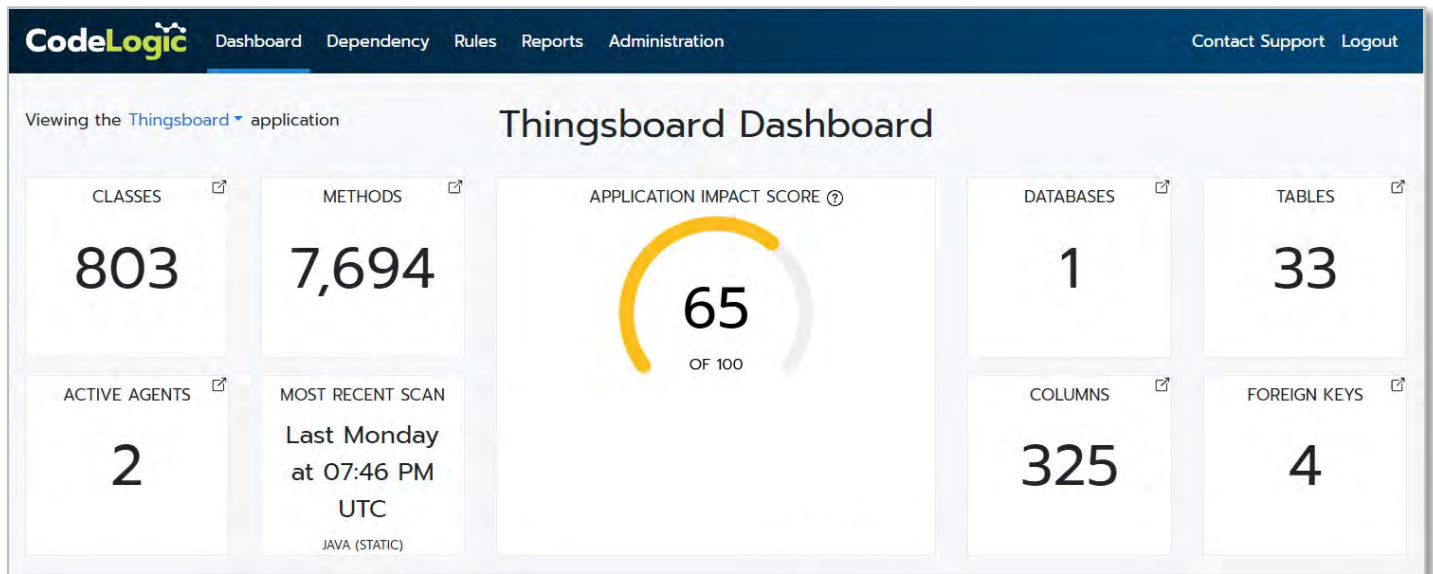
CodeLogic visualizations and dashboards make it easy to see every code dependency that exists within and across applications. With CodeLogic, application teams no longer must spend hours or even days crawling through code to identify and document every API-to-method-to-database connection and dependency. Using binary and runtime scanning, CodeLogic identifies connection points between an application and a database in minutes. Our solution illustrates these connections in an intelligent dashboard that surfaces application complexity and allows teams to quickly analyze dependencies in visualization maps. Once the team begins making changes, CodeLogic helps developers understand the impact of each change. With newfound visibility into application structure and technical debt, team leads can make sure the mistakes of the past are not repeated in the updated application.

Our Approach

CodeLogic provides the industry's most comprehensive, self-updating application dependency mapping and impact analysis tool available. Our approach delivers the application intelligence that teams need to be more productive, better understand code complexity, and make more informed decisions about everyday application and database changes. Teams using CodeLogic can map, document, visualize, and understand their application connections and dependencies from class to method to database – on schedule, on-demand, or after every build.

Question 1

How big is my application?

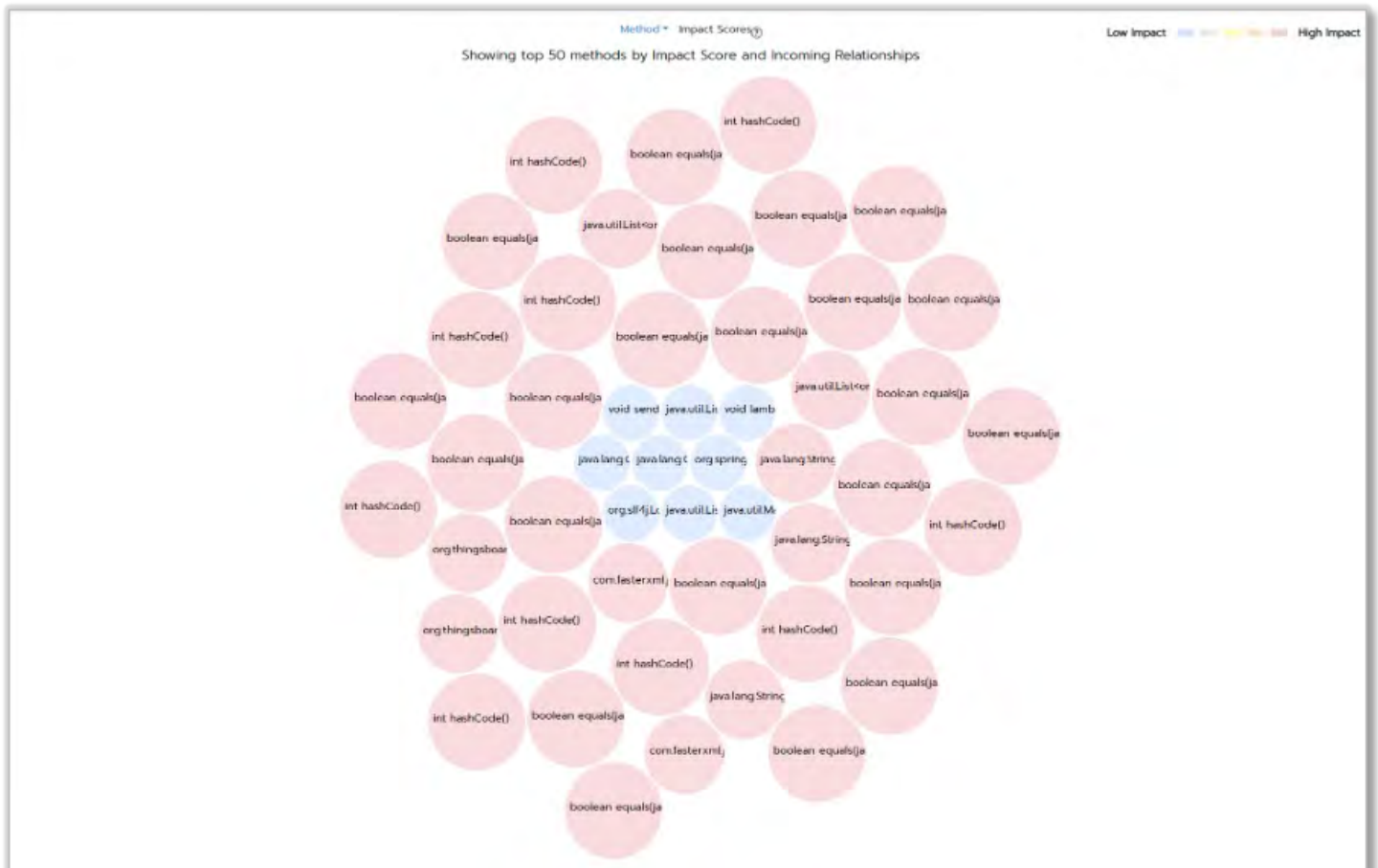


CodeLogic Answer

The CodeLogic Application Dashboard provides clear sizing of the application across multiple services and database.

Question 2

Where is the most critical technical debt?

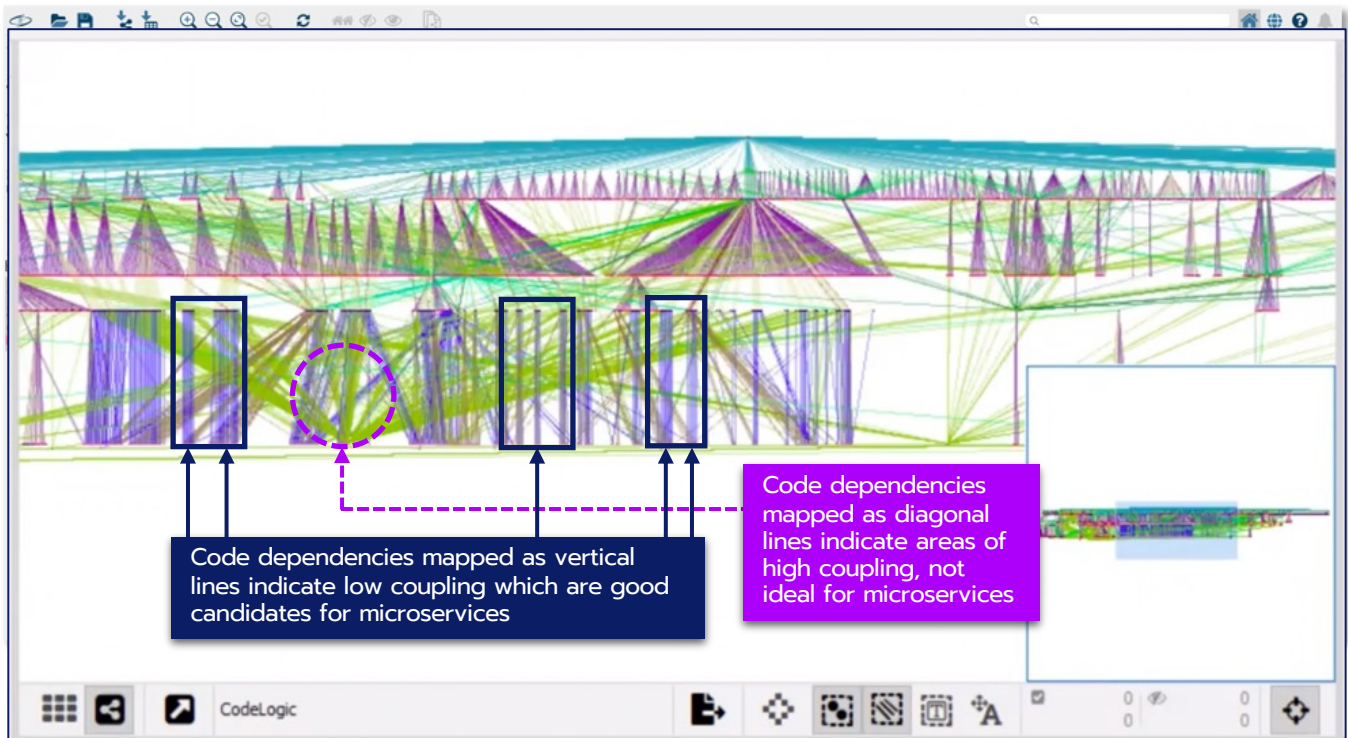


CodeLogic Answer

The Impact Analysis section in the Application Dashboard illustrates the top 50 methods and classes that have high complexity (debt) and high utilization by other parts of the application.

Question 4

What parts of my application are good candidates for microservices?

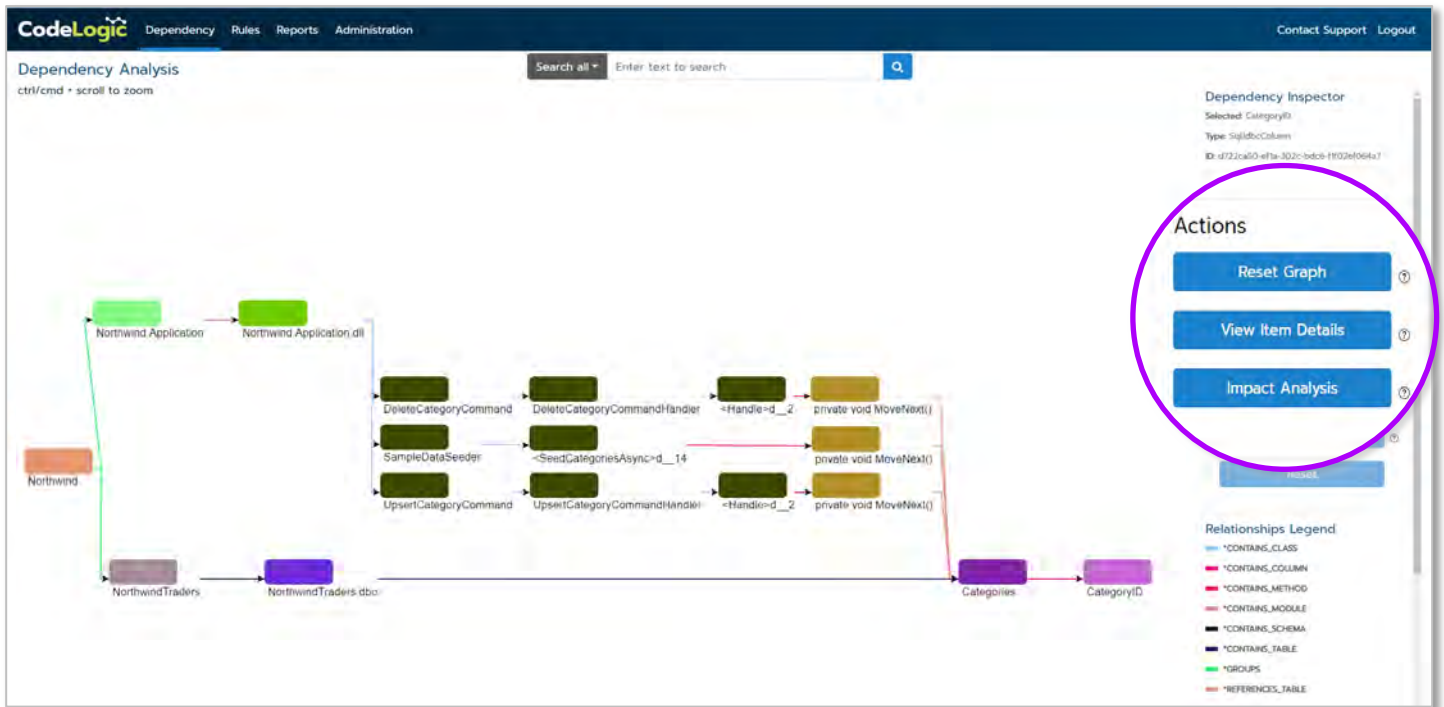


CodeLogic Answer

A macro view of application code to database connections highlights which parts of the code base are ideally structured to be candidates for microservices.

Question 5

What is the impact if I am going to make a change?



CodeLogic Answer

Use Impact Analysis to see all the code items impacted by a proposed change.