**Sourcetrail:**

**Website:** https://www.sourcetrail.com/

1. **Code Comprehension Support:**

Source trail's interactive dependency graph provides strong support for code comprehension. Users are able to efficiently browse the code base, visualize code relationships, and comprehend complicated inter dependencies.

1. **Code Translation Support:**

Translation is not Source trail's primary goal; rather, it is code understanding. It doesn't offer direct code translation features, but it does let users investigate code bases in-depth.

1. **Support for Extracting Design Artifacts**:

With its ability to visualize inheritance structures, class hierarchies, and method invocations, Sourcetrail is an excellent tool for extracting design artifacts from code. This aids in comprehending the patterns and design concepts applied throughout the program.

**Advantages:**

* Dependency graph that is interactive and intuitive for simple code exploration.
* supports a variety of programming languages, such as Python, C/C++, Java, and so on.
* Large codebase indexing that is quick and lightweight. Frequent updates and receptive community assistance.

**Disadvantages:**

* Restricted assistance in reworking or translating code.
* A premium subscription may be needed for certain functionality, such as live code synchronization.
* May occasionally have problems with specific IDEs or codebases in terms of compatibility.

**2**. Understand

**Website:** *<https://scitools.com/>*

**Code Comprehension Support:**

With its powerful code analysis features, Understand offers thorough support for code comprehension. To help in understanding complex code bases, it provides cross-referencing, visualization tools, and extensive analytic.

**Code Translation Support:**

Understands primary functions are code analysis and comprehension, but because to its refactoring capabilities, it also offers some assistance with code translation. The tool allows users to carry out simple code rearrangement activities.

**Support for Extracting Design Artifacts:**

Through the creation of intricate dependency matrices, call graphs, and class diagrams, Understand excels in removing design artifacts from code. This aids in the analysis of software architecture and the discovery of design errors.

**Advantages:**

* Capabilities for deep code analysis with intricate metrics and visualization.
* Supports a large number of programming languages, such as Python, Java, C/C++, and others.
* Scripting and plugins allow for customization and expansion.Suitable for both small and large code bases.

**Disadvantages:**

* Steeper learning curve than that of certain other instruments.
* Comparatively more expensive enterprise licenses.
* Occasionally, very huge code bases might cause performance concerns.