

Supercharge Enterprise Software Development with Automated Static Analysis

October 18th, 2017

Question and Answer Transcript

1. Question: You mentioned payback in the presentation. Do you have any specific examples?

[Roger Hammer] Here's one example: One of our customers wanted us to help with a special analysis related to understanding how some data flowed into other data elements. We were able to do an impact analysis for them using our code slice technology to trace 6000 pathways. From that, we found 16 paths were relevant to their needs. The customer had planned to do the analysis with 32 teams, and they were planning to spend about a man-year to get this information. We found the 16 paths in 50 hours, with zero false positives or negatives, which were verified by manual analysis. In this case, we were able to save them 97%, which may be more than a typical situation, but does clearly illustrate the power of automated static analysis.

2. Question: What is the cost of your product?

[John Rhodes] Obviously, this webinar is not the place for a detailed discussion, but I can give you some basic information. The price of our product is based on a server license, where you pay for a server and that cost is tiered to the size of the codebase. So, if you have a large code base, it's obviously more expensive; if you have a smaller code base, the cost is less. In general, it is quite affordable. We don't charge for each developer seat, so you can get your whole team involved including all of your business analysts, developers and management without having to worry about incremental costs.

Our customers have told us that payback is usually measured in weeks, so it's can be a pretty easy thing justify a purchase, so you can start using it.

3. Question: I've seen some other products that scan COBOL code. What is different about your product?

[Roger Hammer] We discussed some differentiators in the webinar earlier. We parse all of the source code and build a meta model database that is using single element of code. As part of this, we are also identifying all the control flow and all the data flow, in much the same way that a compiler would parse the code. This provides specific and factual information from the details of that source code. So you're going to answer much more detailed questions than you may get from other products.

4. Question: How are your customers using code quality metrics?

[John Rhodes] That's an interesting question. There are a lot of ways people use the quality metric information that you saw on the dashboard earlier. One specific example I can think of was with a help desk customer who wanted to reduce maintenance costs. We helped them monitor the help desk requests that came in and then use CM evolveIT to determine which subsystems contained the defect. In practice, you would see the same subsystem or programs coming up again and again, and typically these would have a very high complexity count. So they used CM evolveIT to focus first on those programs in the attempt to do remediation. Using CM evolveIT in combination with the help desk can be one way to reduce the costs of determining where you can get the biggest bang for the buck.

5. Question: We investigated some other tools that can't handle all the source constructs we have. How does your product deal with these?

[Roger Hammer] We natively handle the code in the most common constructs - COBOL, Copybooks, batch JCL, DB2, IMS, etc. We cover all the basics and most extensions. Many times, customers have additional component types. And as you said, a lot of tools can't really deal with that. We're able to bring those other component types into CM evolveIT and provide the ability to use regular expression scanning and impact analysis with a capability we call a Correlation. Correlations allow us to actually link together these other component types with the COBOL and the database information where it makes sense and provide a full complete view for your developers and analysts.

6. Question: We are considering an IMS to DB2 conversion. How can this tool help?

[John Rhodes] That is a good question. We do analyze IMS databases and DB2 databases. CM evolveIT doesn't specifically help with code translations. We do have a partner, Syncsort. They use our tool within their automated DB2 to IMS conversion service. If you're interested in that, get in contact with us so we can help you reach the right folks at Syncsort. If you want to do this on your own, that's also something you can use our tool for. You can focus on the IMS calls and use those to analyze how the code is flowing around those calls and then do the changes manually. That's something else we can discuss. Obviously that's very technical and a difficult conversion to undertake, but I can see that we could help in that situation with our services offering.

7. Question: Does your product have any limitations as far as the number of programs or lines of code it can parse?

[Roger Hammer] Good question. I guess the quick answer is no - there's no built-in limitations to the size and scale of a particular application set. We have application sets as small as a few hundred programs up to many thousands of programs. CM evolveIT is built to handle scale, and we want to make sure customers understand that. For us, number of programs and lines of code are really the same thing. As we parse all the code for different programs, we provide full database information for each of those programs and all the related functionality. All the copybooks and different connected components flow into our database and becomes available in the Inventory Viewer. When it is loaded, I can work with tens of thousands of components, or I can scale back and work with only the few that I'm interested in, by filtering in and understanding a subset or code slice.

8. Question: We are being asked to move part of our COBOL application to Java. What would your approach be? Do you automatically migrate code?

[John Rhodes, Roger Hammer] This is a big question so maybe we'll both answer just part of it for now. Our product does not automatically migrate code. And the reason for that is that it's very difficult to have a one-size-fits-all approach around automated code transformation, because typically folks are coming from a particular architecture and have specific idea of where they want to go which is unique to each organization. If you have the right volumes of code and you have the right target architecture in mind, we can help with our machinery. But that would be a separate conversation. Roger, do you have any general comments on folks that are moving COBOL to Java. I know we've had several customers that have kind of done that kind of thing.

Another way that we've helped customers with this is our ability to find and understand business rules. We can extract pure business as well as build specifications from the existing code information. We've done this work with customers to identify all their requirements and specifications from the data perspective, and help them identify how that data gets created and generated within the application. And then we're able to output that information in a structured format, providing the developers who are building the new system with a blueprint. In one case we helped a customer move from COBOL to modern systems with a Java back-end and a Sencha ExtJS Javascript front-end. In that case, we were providing the specs and logic flows to the Java back-end developers and UX front-end developer to rebuild the logic of the application.

9. Our organization has a large number of Assembly language programs. Can evolveIT analyze Assembler? We have a great deal of PL/1 code; do you process PL/1?

[John Rhodes, Roger Hammer] Out of the box, CM evolveIT can do an analysis of Assembly language source via its Correlation facility which will enable regular expression matching and basic impact analysis. For more sophisticated analysis requirements, we have a plug-in architecture where an advanced ASM parser could be utilized. We typically tune the advanced parser to the Macros and other constructs in use a particular code base. PL/1 is basically the same story. We should have a detailed technical discussion of your requirements to see if we can support what you need.

10. What DB structures can be used for the repository?

[John Rhodes, Roger Hammer] If the question is which storage engine do we use for the repository itself, we current use a Windows-based graph or "NO SQL" database. SQL-based database are not used because they are too slow for data and control flow analysis. There is an API available to extract information out of the database. The current implementation requires Windows Server storage but we have a Linux version on the roadmap.

11. Is your interface Eclipse-based? If not, is that on the roadmap?

[John Rhodes, Roger Hammer] The interface is not currently Eclipse-based. We have a roadmap to deploy a browser based interface by the end of 2017, and anticipate an Eclipse plug-in interface in 2018.

12. How much effort goes into maintaining the repository (size, indexes, etc.)?

[John Rhodes, Roger Hammer] After initial setup, there is only a minimal effort that must go into maintaining the repository. There is no ongoing effort required around indexes or tablespaces, since we are using graph “NO SQL” datastore that automatically indexes, as required. For very large repositories, a clustering mechanism is recommended. CM evolveIT provides full automation for a “Discovery Refresh” to update the database with only changed components (and dependent components) which CM evolveIT identifies automatically.

13. Does the product span distributed apps as well to follow data enterprise-wide? Natural, SAS, assembler? Quickjob, easytrieve?

[John Rhodes, Roger Hammer] Good question. We have a Java capability currently available. For other languages, we have two options – use our Correlation feature to bring in source for regular expression type searching and analysis. For more complex data flow analysis, we offer a plug-in architecture that can be used for any language. Once the parsers are developed and in place, data lineage can be done enterprise wide.