Discussion
[Dmitry Maslennikov](#) · Sep 4, 2020

Format of stored files in Source Code Control systems

The use of Source Code Control systems in development is important. And there are a few systems known in the world, like GIT, SVN (Subversion), Perforce, Mercurial. Where the most popular nowadays is the git. Using it is very useful in many cases, but mostly it depends on the code as text, which can be compared between commits, branches, or versions of releases and so on.

And InterSystems ObjectScript is not an exception, for many years it was possible to use SourceControl class in Studio, which helps to work with any Source Code Control System, and some companies already used it for many years. But the issue with ObjectScript was, in the format how it was usually stored. It was not easy to store classes as it looked in Studio, and the easiest way was to save it in XML format. And the issue in storing classes as XML, in the way how it's interpreted by Source Control Code systems, and it does not care about XML, so in some cases, like merging branches, it can be easily broken.

And nowadays, when InterSystems Cache (since 2016.2) and IRIS supports easy export of classes in the way how it's visible in Studio. And in conjunction with VSCode, where you can work with your source code as just files, which can be compiled on the server. And still using XML makes it more difficult to use.

Using plain Class code instead of XML, does not mean you only have to work in VSCode, you still may use Studio, just change a bit your Source Code class.

To solve the issue with XML files stored in the Source Code control system, I see two ways:

- Start a new repository from scratch, and use source code in a new non-XML way. The easiest way, but the issue here in losing the history of changes. Even if you keep the repository, but change the format of files, you will not be able to compare code in between formats.
- Convert the entire repository with the history, to be supposed like it was always in a newer format. In this case, you will keep the history of changes and can compare changes during the whole life of the repository. As this suppose to change comments back up to the first one, most of the commits will be changed, and if you would use some links to commits somewhere, like pull requests, you may break it, as well as signatures on commits.

As a company [CaretDev](#), we would like to offer you a service of conversion your repositories from XML to a modern format.

If you have any thoughts I would like to hear from you.

And a pool, just to see the numbers.

[#CaretDev](#) [#Development Environment](#) [#VSCode](#) [#Caché](#) [#InterSystems IRIS](#)