
Article

[Sergey Mikhailenko](#) · Jan 25, 2021 2m read

[Open Exchange](#)

ZAPM is a shell - extends the ZPM shell and adds any other commands.

Hello, developers!

I would like to share the history of the project - the ZAPM shell.

As soon as ZPM was implemented, I immediately began to think about how best to use it.

And immediately faced with the desire to move more quickly between namespaces, especially when there are more than 20 of them.

I had to leave the ZPM, move to the desired namespace and re-enter the shell.

I suggested an improvement - a new "namespace" command for easier navigation.

I didn't wait - I did it myself. And so it went. If you need it, get ready to do it yourself.

But more and more my wishes for the development of the functionality went beyond the functionality of the package manager.

And I had to make my own shell for the aggregation of specific commands and functionality that I needed.

And then I got tired of jumping from one shell to another and I combined them.

My ZAPM checks if the command entered is a ZPM command, then sends the execution to the ZPM shell.

Then I wanted to improve the color commands and expand the functionality.

And now my ZAPM has survived to the first release and can add any command I need.

The list of available commands can be displayed with a description, both in full and in context.

The command can be executed immediately by entering its number according to the list or the assigned short name.

Commands are divided into immediately executed (I), executed with a question (A), and non-executed (N) (stored for reference, usually having parameters).

The command can be executed not only from the shell, but also from the command line of the terminal.

Commands are stored in a stored class [Object Model](#) and accessed via SQL [Relational Model](#).

All executed commands are stored in the global. The history of the command execution can also be displayed both in its entirety and in context.

An earlier executed command can be re-executed.

Command history is stored in globals [Key-Value model](#).

I plan to improve this project with new features and new commands (add command, delete command).

I will definitely add history management (cleaning and editing).

There is also an idea to add a context for command execution (scope, parameters, roles, usernames).

I hope this ZAPM wrapper is useful to you. I accept any criticism and wishes.

[#System Administration](#) [#Terminal](#) [#Tools](#) [#InterSystems IRIS](#) [#InterSystems IRIS for Health](#)
[Check the related application on InterSystems Open Exchange](#)

Source

ZAPM is a shell - extends the ZPM shell and adds any other commands.

Published on InterSystems Developer Community (<https://community.intersystems.com>)

URL: <https://community.intersystems.com/post/zapm-shell-extends-zpm-shell-and-adds-any-other-commands>