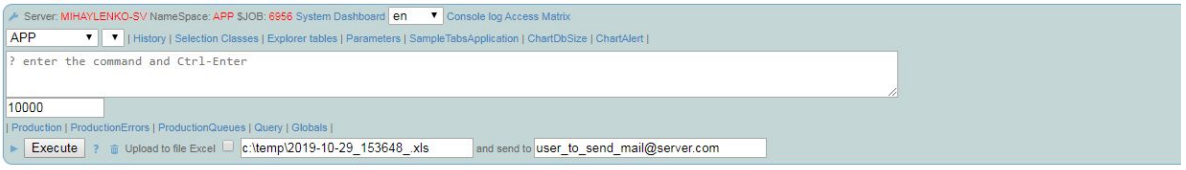
Documentation AppTools

A panel of additional tools for monitoring and investigating application errors and integration solutions on the InterSystems IRIS data platform, the Ensemble integration platform and the caché DBMS.

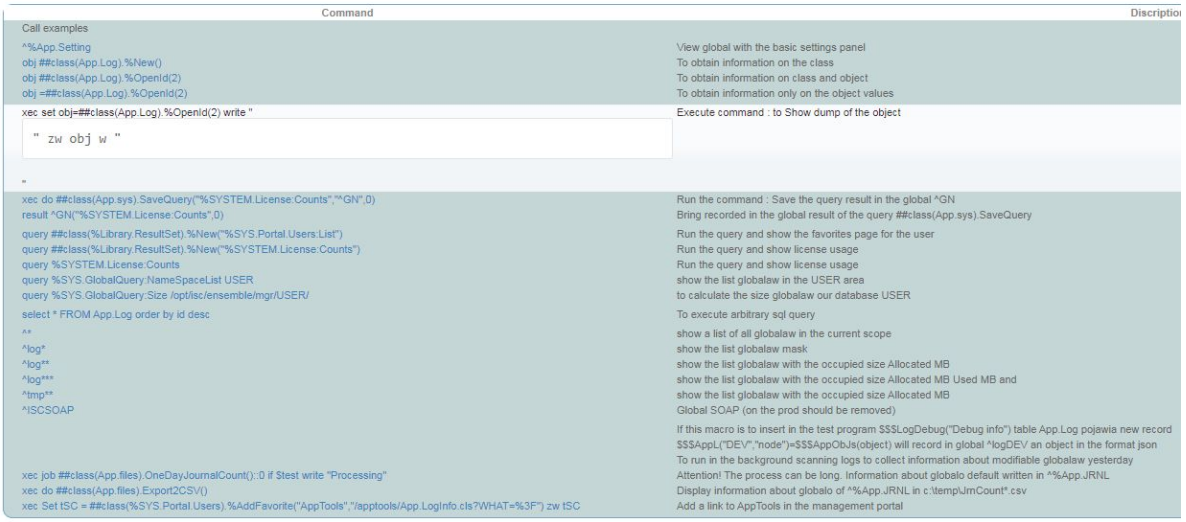
In this article, I want to talk about the application that, along with the standard administration tools, I use every day when monitoring applications and integration solutions on the InterSystems IRIS platform and finding errors when they occur.
The solution includes viewing and editing global arrays, executing queries (including JDBC / ODBC), sending search results via email as archived XLS files. View class objects with the ability to edit. A few simple graphs on the system protocols.

This is a CSP application, based on jQuery-UI, chart.js, jsgrid.js

http://localhost:57772/apptools/App.LogInfo.cls?NSP=APP&WHAT=%3F

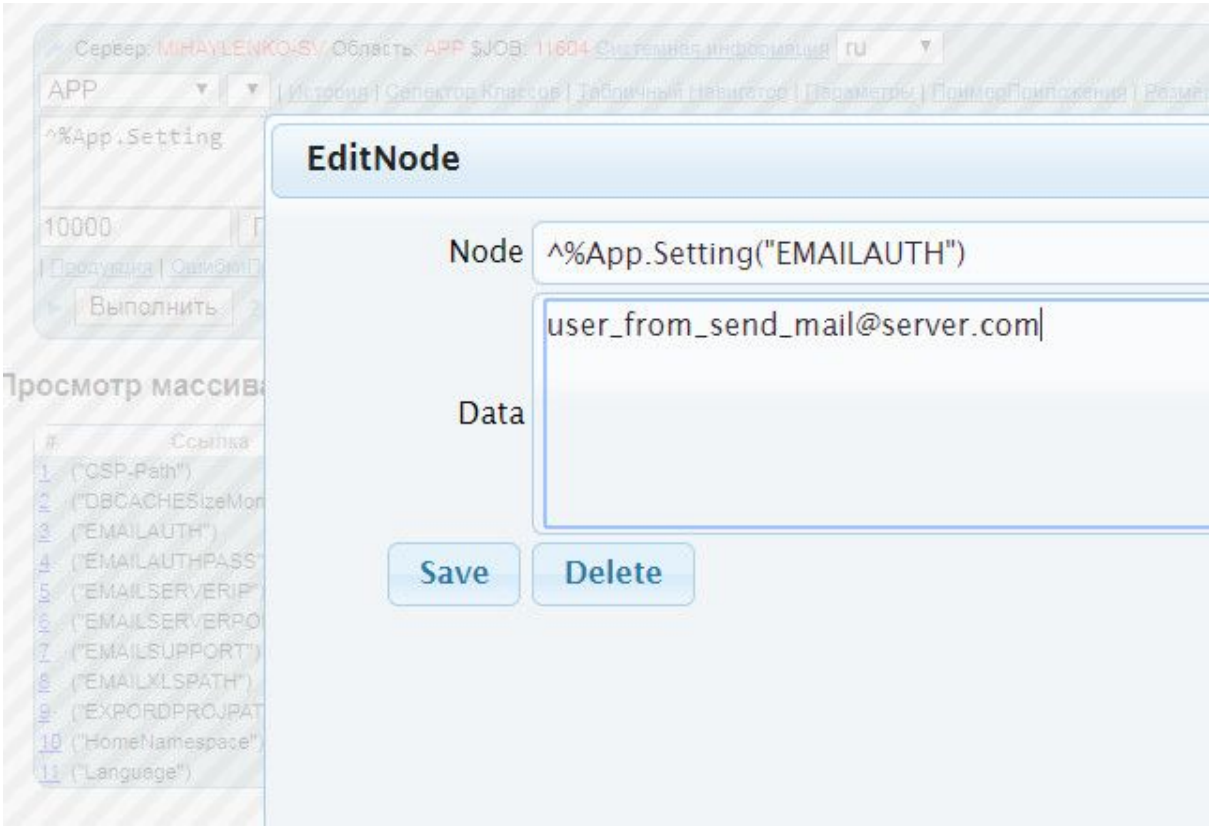


help command



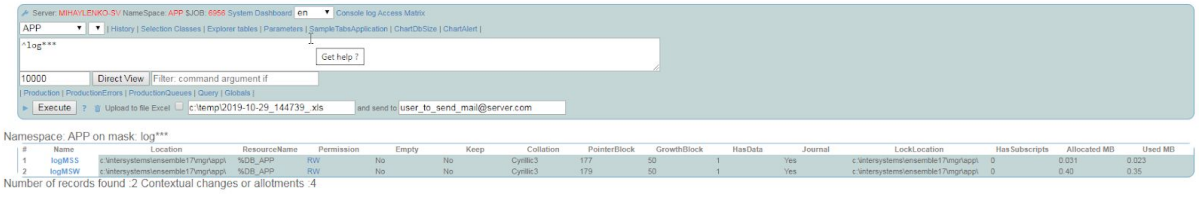
Globals

The most frequent my team — viewing global. As a rule, this is a global Protocol when debugging your own or someone else's project. It can be viewed in reverse order, as well as by applying a filter on both the link and the data. Found nodes can be edited and deleted:



You can delete the entire global by typing in the command after the name minus ^logMSW- But it is possible to remove so only globals beginning on ^log (Protocol globals), i.e. restriction from casual removal is implemented.

If you enter " * " after the name, we get a list of globals with additional characteristics. The second " * "- will add a new field "Allocated MB", and another asterisk " "Used MB" this is the Union of the two reports and the division into "asterisks" is done to divide the often long-formed report on the occupied blocks of large globals.



From this tab, you can follow the active links to view the global itself or to view/edit it in the standard way from the management portal by clicking on R or W in the Permission field.

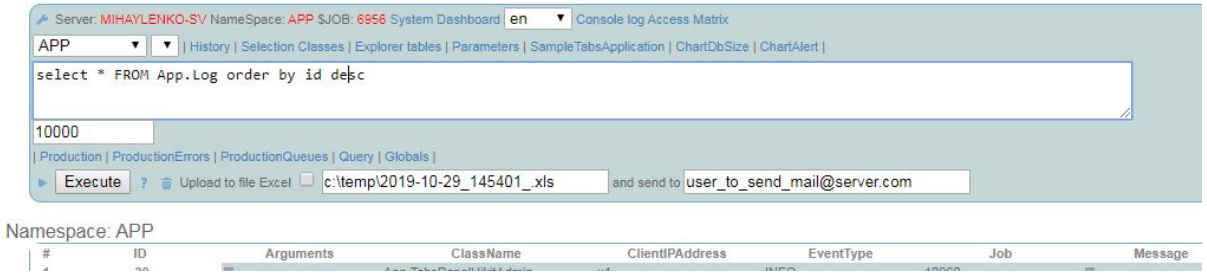
Query

Convert report to Excel format

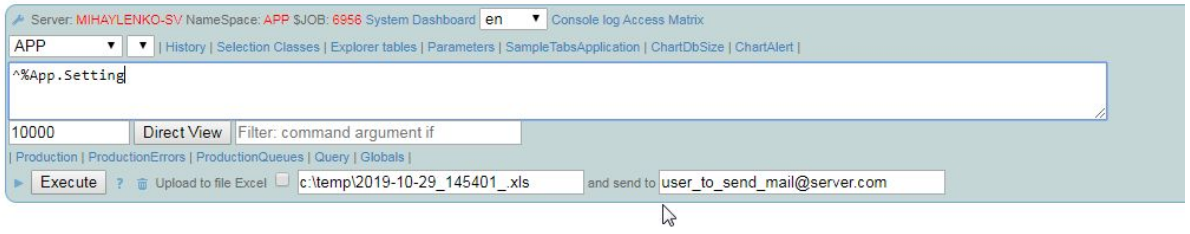
The second function, in terms of frequency of use, is query execution. To do this, enter the sql statement as a command.

The main thing that I lacked in the standard management Portal of the system, it is the execution of queries configured in the database JDBC- / ODBC-sources and output the results in XLS format, archiving and sending the file to the mail. To do this, in my tool, before executing the command, you need to enable the checkbox "Upload to Excel file".

This feature saves me a lot of time in my daily routine, and I successfully integrate ready-made modules into new applications and integration solutions.



But to do this, you first need to configure the path of creating files on the server and user credentials and mail server, for this in turn, you need to edit the nodes of the global program settings ^%App.Setting.



View array : ^%App.Setting in namespace APP

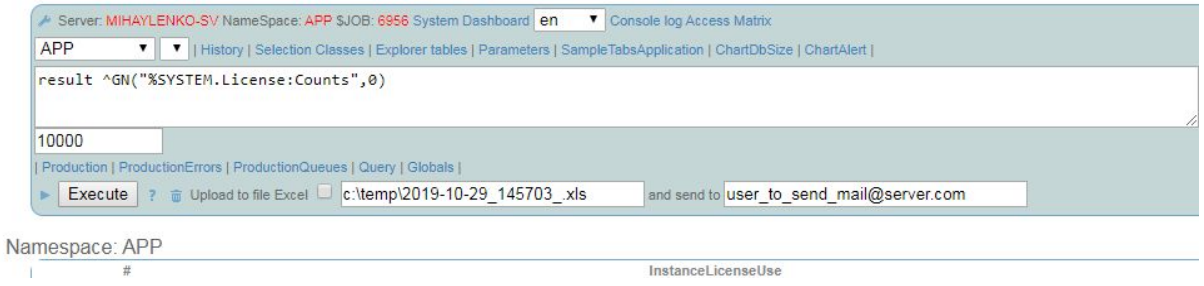
#	reference	Data
1	("CSP-Path")	C:\InterSystems\Ensemble17\CSP\app
2	("DBCACHESizeMon")	CACHESYS,CACHEAUDIT
3	("EMAILAUTH")	user_from_send_mail@server.com
4	("EMAILAUTHPASS")	12345
5	("EMAILSERVERIP")	127.0.0.1
6	("EMAILSERVERPORT")	25
7	("EMAILSUPPORT")	user_to_send_mail@server.com
8	("EMAILXLSPATH")	c:\temp\
9	("EXPORDPROJPATH")	c:\temp\source\
10	("HomeNamespace")	APP
11	("Language")	en
12	("PATHCSP")	/apptools/

Saving reports globally

Very often, you want to save the results of the report execution to the global. For this I used the procedures:

Для JDBC:	##class(App.sys).SqlToDSN
Для ODBC:	##class(App.sys).SaveGateway
Для SQL выражений:	##class(App.sys).SaveSQL
Для Query:	##class(App.sys).SaveQuery

For example, if the command
exec do ##class(App.sys).Save Query("%SYSTEM.License:Count", "^GN", 0)
let's save the result of the license usage calculation query in the ^GN array, and you can see what you saved in the panel with the command: result ^GN("%SYSTEM.License: Counts", 0)



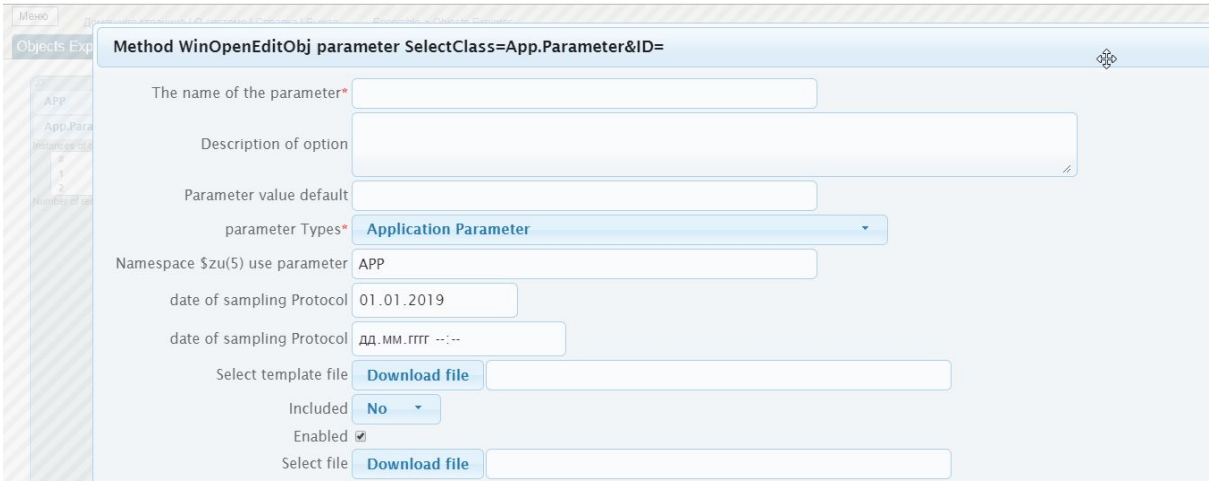
Enhanced functionality modules

And the second improvement that greatly simplified and automated my work is the implementation of the ability to execute specially written modules when forming each query

string. This way I can embed new functionality into the report on the fly in one pass, for example, active links for additional operations on data.

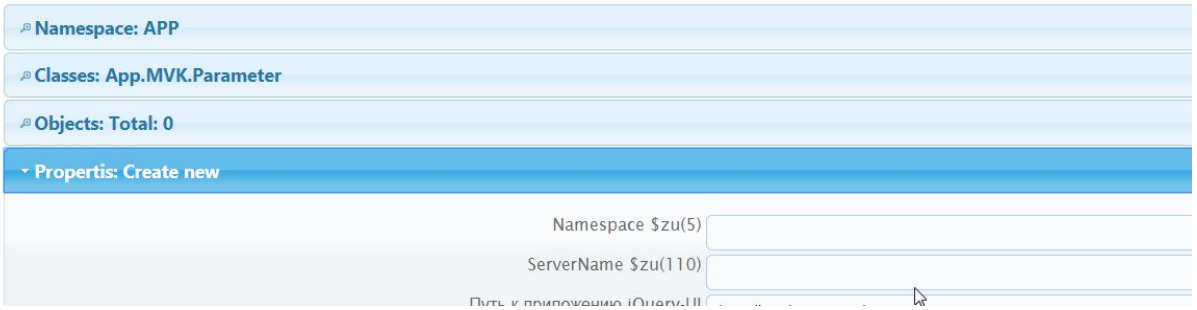
Example 1: working with the App class.Parameter

http://localhost:57772/apptools/App.FormExp.cls?NSP=APP&SelClass=App.Parameter

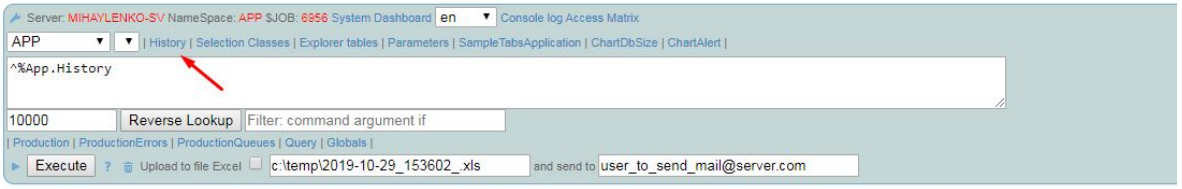


Example 2: To create a parameter via the "Table Navigator»

http://localhost:57772/apptools/App.FormExp.cls?panel=AccordionExp&NSP=APP



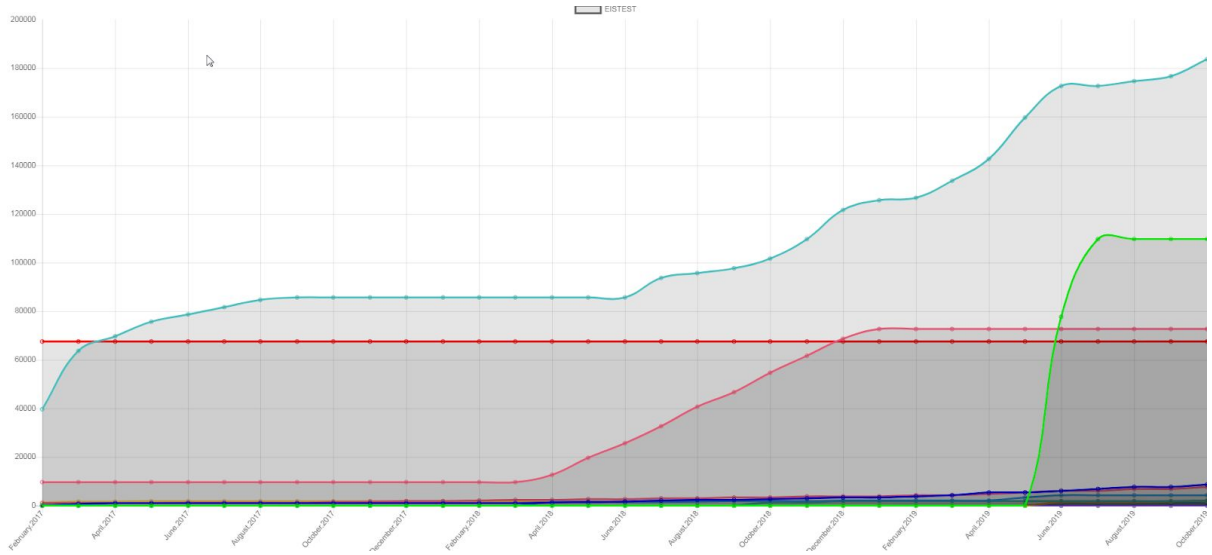
Example 3: View global via History link»

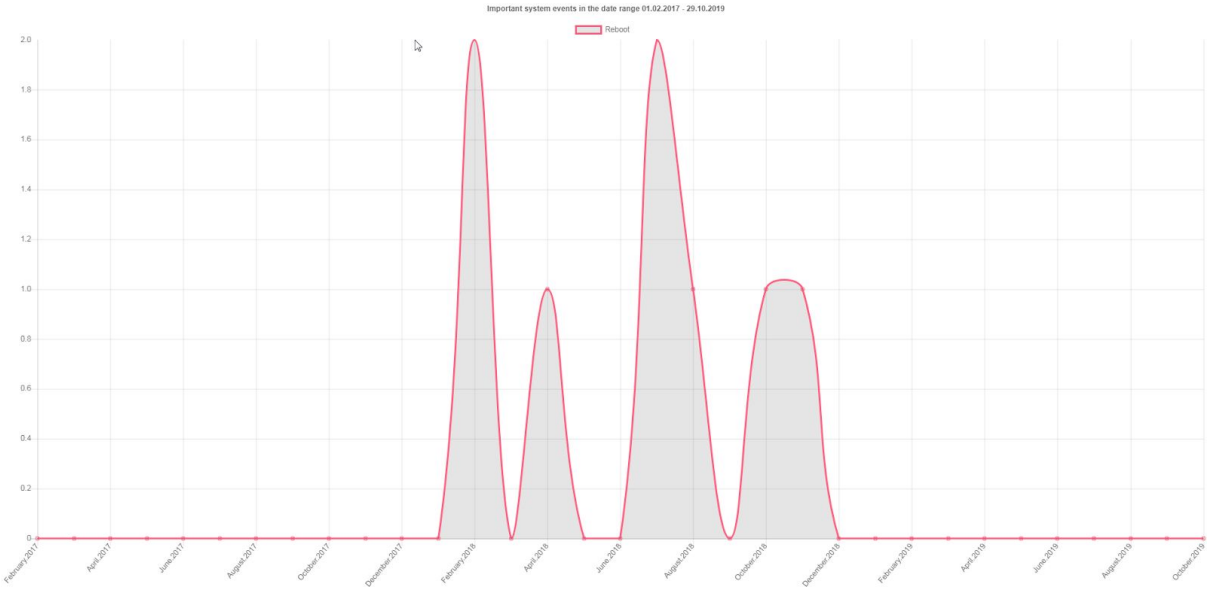


Graphics

Visualization of database growth a page has been created that displays a monthly graph of the database size created from the iris file.log (cconsole.log) on records "Expand" retrospectively from the current day.

For example, a graph of events in InterSystems IRIS was created, which is also formed by a Protocol file:

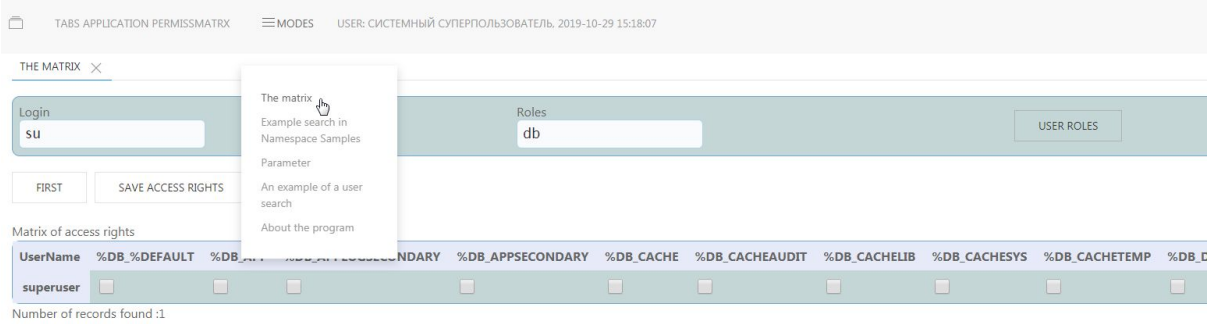




Access rights matrix

When there are several thousand users in the database, it becomes long and inconvenient to assign rights to them through the standard IRIS interface. This application was created to automate this process.

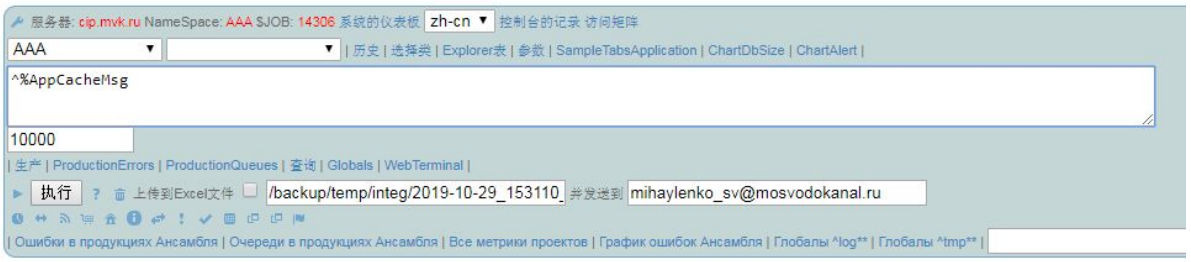
You can use the permissions Matrix application to assign and modify roles for users by selecting them by context
http://localhost:57772/apptools/App.TabsPanelUikitPermissMatrx.cls?autoload=Matrix

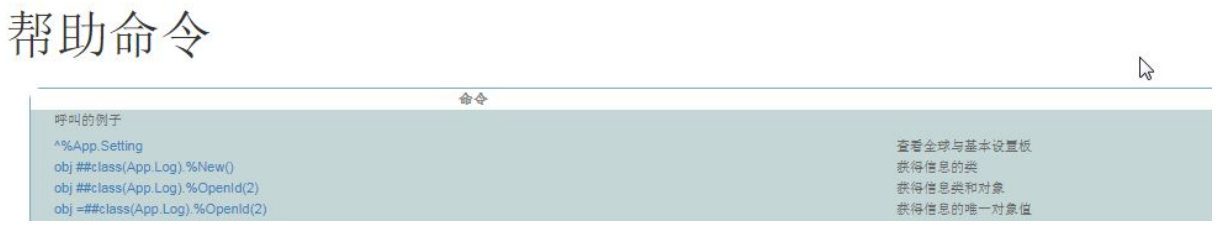


This application can be used as an example of quickly creating simple applications. The description of the menu items is in the method GetAllApps.

Translation

To translate the application menu items you need to import into the global system ^%AppCacheMsg
Which can be found in the installation directory of the project apptools\src\glb\appcachemsg.xml





The $$$aText macro is used to prepare a multilingual interface.