Open Source Project „Rapid Fire”

Do you have files with millions or even billions of records? Imagine you have to change such a file e.g. a new field has to be added. Possibly you have the time to install this change at the weekend. But what if your shop works 24 hours a day and 7 days a week? In this case it is absolutely necessary to reduce the time needed for the installation to avoid a long shutdown of the system.

This is the point Rapid Fire comes in. With Rapid Fire it is possible to reduce the downtime of your system to a minimum. Instead of shutting down the system for hours or even days you can reduce this with Rapid Fire to minutes.

First let`s have a look how such a task will be realized today. There are several ways.

- If you have a DDS described physical file you can use command CHGPF to apply the changes.
- If you have a SQL Table you can use the command ALTER TABLE to apply the changes.
- You can create the new file by using the commands CRTPF or CREATE TABLE and then use command CPYF to copy the records from the old file to the new file.

All this ways have in common, that the users have to leave the application from the moment you start the conversion to the end. Depending on the amount of records in the file this can be a very long time.

Now let`s have a look how such a task will be realized by Rapid Fire.

- A special library – the shadow library – will be created.
- The file with the new structure will be created in the shadow library.
- A RPG program will be generated which copies the records from the file in the production library to the file in the shadow library. Copying the records by an RPG program is much faster than doing this job by the CPYF command.
- If the file in the production library has not already journaled, a journal will be created in the shadow library and the file in the production library will be journaled in the created journal.
- Now the copy process starts which copies the records from the file in the production library to the file in the shadow library. This process

runs in background and the users don`t have to leave the application and can continue their work.
- The copy process can take hours or even days. A separate Rapid Fire job receives the changes in the file of the production library. Rapid Fire receives this changes from the journal the file is journaled. These changes will be applied periodically to the file in the shadow library. Now the file in the production library and the file in the shadow library are in synchronization.
- When all records are copied, all changes are applied and you have reached the time frame for the installation, you have to ensure the users leave the application. After this you can stop the Rapid Fire job and the last changes will be applied. The last thing you have to do is to promote the file in the shadow library to the production library manually.

There are just a few features which are not provided by the open source version of Rapid Fire. These features will be provided, if you use Rapid Fire together with our Change Management System CMOne. These features are.

- Applying the last number of an identity column in the file of the production library to the corresponding file of the shadow library.
- Applying triggers.
- Applying referential constraints.
- Promoting the file in the shadow library to the production library.

Rapid Fire needs IBM i V7R1M0 to run.

Install Rapid Fire

Download Rapid Fire, unzip the save file and transfer the save file to your system e.g. with FTP.

Restore the Rapid Fire library by entering the following command in the command line of a 5250 session.

RSTLIB SAVLIB(RAPIDFIRE) DEV(*SAVF) SAVF(MYLIBRARY/MYSAVEFILE)

Start Rapid Fire

To start Rapid Fire enter in the command line of a 5250 session the following commands.

CHGCURLIB CURLIB(RAPIDFIRE)

RF