From OS/2 to macOS
- a travel in time & space

Presenter: P.O. Jonsson (oorexx@jonases.se)

30th International Rexx Language Symposium,
Hursley Park, UK, September 2019

Presenter P.O. Jonsson, oorexx@jonases.se

Session title: "From OS/2 to macOS - a travel in time & space"

Session abstract:
Showcased is a rexx project that started as a simple data entry script in 2000, a project that eventually evolved into a fully automated data mining ooRexx program in 2019.

The presentation is, I think, proof of the resilience of Rexx in moving between operating systems and platforms.

Credentials:
P.O. Jonsson have a MSc in engineering physics and worked as a development engineer before taking up a post as a patent examiner at the European Patent Office (EPO) some 30 years ago. At the EPO he came into contact with Rexx as it was used to automate search procedures. In his spare time he used Rexx and later ooRexx to create in-house search tools that are now part of the toolbox for all examiners at the EPO.
I have worked for almost 30 years as a patent examiner and to understand my project you need to get familiar with some patent basics.

I will explain with the help of one example:

I will use a European Patent application rather than this one.
Here is how a European, or EP patent application looks like (using a coder/decoder system as an example)

The Applicant is known to us (IBM)

We also know the inventor, I think? (Mike Cowlishaw)

And the representatives location is also familiar (Hursley Part).

Now, for a patent examiner the work starts by reading the application, trying to understand the inventive idea, and what distinguish it from the prior art.

When reading the application you will typically see something like this:
1st line of page 4 contains a cited document

I will enlarge the piece I am interested in (pointing to the US document in line 1).

Now, since the inventor indicates that the invention builds on ideas presented in this document we need to make a detour and find that document.

THIS is where my little project kicks in. It might seem to be a trivial task but believe me it is not, at least not every time.

Once the application is understood, a novelty search is performed and a search report drawn up
Here the examiner cited four documents, the second one being the one we just saw.

The „D“ in the search report indicates that this document come from the description itself

It is thus a „D-document“ in the lingua of patent examiners, and also the name of my project, Ddoc.
After an examination phase a patent is eventually handed down by the EPO.

The EP patent can be validated in the Designated countries.

My project DDOC thus aims at retrieving the documents cited by the Applicant.

Please note that US3618047, the document cited by the Applicant, is cited in three different ways in this single document, which makes it a non-obvious task to extract such documents in general.
Year 2000

- Version 1 of the program
- Under OS/2
- Using IBM REXX (or IBM Object REXX?)

The original data entry script contained around 300 lines of code

Another script was used by the examiners

The system was used by around 40 examiners
This is a representative script for this time, just to give an idea of the complexity (joke)
Early Year 2003

- Version 2 of the program
- Still using OS/2 (possibly OS2/Warp)
- Still using IBM REXX (or IBM Object REXX)

Used in 5 directorates ~140 Users

6500 files were processed between 2000 and 2003

Corresponding to 528 000 pages of paper not printed

A saving of 4.5 Pallets of paper, or 2.6 tons.
This is from a presentation I held in 2003
(probably the last time I held a presentation 😊)
DDOC System Layout I:

- Clerk enter the references and DDOC generates a crossfile on the T: drive

\[ T:\general\ddoc\pj30332\12345678.xft \]
DDOC System Layout II:

- Crossfile is automatically downloaded to the Examiners workstation for direct use in EPOQUE (Jviewer, Internal, Caesar etc)

T:\general\ddoc\pj30332\12345678.xft
Please note the (familiar) IBM OS/2 desktop
Editor “E” was used for data input
The examiner’s view of the entered documents
Late Year 2003

- Version 3 of the program
- Ported to Windows 2000
- Using IBM Object REXX V2.1 (most likely)

IBM Object Rexx was apparently delivered with W2k in 2003

In general the version number was bumped with one whenever the OS was changed
Stats on use

The decreasing use 2006-2007 and the possibility of obtaining the fulltext made me ponder over an automated tool.
Year 2008

- Ddoc version 4
- Windows XP replaces Windows 2000
- Still using IBM Object REXX V2.1

Ddoc is now using automatic extraction of data from fulltext

First tests with BSF4Rexx

First tests with ooRexx (4.0) in parallel to IBM Object Rexx
The use of ooRexx + BSF4ooRexx made it possible to establish an interface between rexx and JAVA

Rexx used in the frontend by examiners to script database queries

and with the backend with middleware written in JAVA

A first ooRexx and BSF Workshop was held with Rony Flatscher 2011
Year 2014

- Ddoc version 6
- Windows 7
- Still ooRexx (4.2.0) and BSF4ooRexx

An entirely new database structure requires rewrite

Windows 7 triggers a new main version
Year 2016

- Development frozen
- Ddoc handed over to the EPO
- migrated to JAVA (by the EPO)

Ddoc Migrated to JAVA 2016 by the EPO

I supported the 1:1 port from ooRexx to JAVA of 15 000 lines of code.

The port was so good that even bugs (as I detected later) had been ported!
The statistics on use show a peak of half a million uses of Ddoc in 2015, or 2000 uses per day or 2-3 times per minute.

The reason that 2016 show lower number is because the JAVA version have taken over part of the work and stores the data differently than the ooRexx version.

What is seen here is for the ooRexx version only

After 2016 no further statistics on use was collected (by me)
Year 2018

• I leave the EPO
• Ddoc running in Pre-Search on every file
• maintained by the EPO

In 2018 the JAVA version have taken over all tasks and the ooRexx version is only kept as a control
In 2019 I wanted to revive my project Ddoc and make it available to the public as a service.

In order to migrate Ddoc to macOS I needed to replace the checking of document numbers.

Where previously it was an inhouse database query it is now a https call over the internet to the EPOs API called OPS, Open Patent Service.

Less than 10% of the code was rewritten, the client for OPS, the rest was ported to MacOS without any major changes needed to the code.
Thank you for your attention!

P.O. Jonsson (oorexx@jonases.se)
A long time ago in a program far, far away...