# D.2.1. Public databases

# The NewProt SSP portal revolves around data and software brought together in the SSP (Self Service Portal). In Deliverable 2.1 the access to publicly accessible data is arranged. The CMBI has for many years been working on the MRS project, and an old seminar on this topic is added as an appendix to this deliverable. This seminar is old, but the MRS concepts haven’t changed.

# The CMBI was deeply involved in the FP6 NoE called EMBRACE that aimed at European interoperability of software and databases in the biosciences. The EMBRACE project was very successful, and actually, it seems unlikely that NewProt could even have been proposed if EMBRACE had never taken place. The interoperability standards that came as recommendations from EMBRACE have been followed in many CMBI software activities, including MRS. This has allowed fluidOps to easily incorporate MRS in the SSP.

# The SSP and MRS work totally independently and are connected through a simple script that only has to follow the XML definitions set-up for the project. Consequently, MRS development (and of course the weekly updates of its ~50 databases) can continue without the need for frequent communication between the MRS and SSP software teams.

# MRS is also one of the core packages of the EMBnet activities for developing bioinformatics in Africa, but that is neither a part of NewProt project, nor crucial for this deliverable.

# The core of D.2.1 is that access to all databases is warranted. As a simple execution of the SSP shows, this goal has been achieved and Deliverable D.3.3 (actually due in month 18, but completed already) shows that the user can have access to the data through the SSP.**MRSan introduction**

## **M.L. Hekkelman, G. Vriend CMBI 2006**

# **Three parts**

## Automatic data gathering and updating

## Indexing of data. Mostly, but not limited to, flat file and XML

## Searching the data and presentation of results

# **Design goals**

## Avoid errors by keeping it simple

## Fault tolerant, show must go on

## Platform independent

## Minimal hardware requirements

# **Data updating**

## Based on makefiles

## Minimal configuration

## Continues after an error

## Clear reporting

# **Indexing**

## Creates one big file containing data and indices

### No more out of sync situation

### Very clear if indexing succeeded or failed

### Atomic replacement of production file

### File is completely standalone without any dependency

## Creates in one go, or by splitting work over cluster nodes and merge results

## Can deal with update databanks and join databanks

## Uses Perl plug-in scripts to parse the flat files

# **Searching**

## Fast with lots of headroom to become even faster

## Boolean queries with optional operators and wildcards

## Future version will contain ranked queries

## Results are pretty printed by Perl plug-in scripts

## Blast built in, blast against the results of a Boolean query

# **Wrap up**

## MRS is a library with several front-ends (Perl/Java/C++)

## Flexible, fast and with lots of potential

## Has been in production for several years at the CMBI

# **Future developments**

## Create a distribution with documentation

## Extend the Web Services interface using gSoap

## Distributed data and queries