##### **INS-CGN-PG-016 INSTRUCTIONS FOR UPDATING GENIS DATA ON GENIS-WEB**

## 

## Introduction

Updating GENIS data on the Internet has a number of aspects:

1. Creating the downloadable Excel-files with passport and phenotypic data and the HTML-files that arrange access to them.
2. Updating a MS Access database that is used to update the SQL-Server database that supports on-line searches with passport data.
3. Updating a MS Access database that is used to update the SQL-Server database that support on-line searches with phenotypic data.
4. Replacing files on the web server.
5. Adding pictures that are accessible via thumbnails on the ‘accession level’ via the

internet.

This document describes the procedures for these activities.

1

Procedure for creating the downloadable GENIS files 2

**Directory structure and required files** 2

**Element 1 - Creating the HTML files** 2

**Element 2 - Creating the downloadable passport files** 3

**Element 3 - Creating the downloadable trait files** 3

Procedure for creating the online searchable tables with passport and phenotypic data 3

**Directory structure and required files** 3

**General structure of the procedure** 4

**Element 1 – Creating the passport mirror tables** 4

**Element 2 – Refreshing the SQL Server database** 4

Procedures for replacing files on the web server 4

**Directory structure** 4

**General updating and back-up procedures** 5

**Specific updating procedures** 5

*Location of files* 5

**Checking integrity of pictures** 7

**Uploading pictures to the cloud** 7

## 

## Procedure for creating the downloadable GENIS files

### **Directory structure and required files**

The procedure for creating the complete set of files for downloading GENIS data via the Internet requires a particular directory structure as well as a number of files:

C:\Data\Genis-Internet\Download root directory with three SQL-programs: *MakeHtmls.sql*, *MakeTraitFiles.sql*, and *MakePasportFiles.sql*

C:\Data\Genis-Internet\Output output directory

C:\Data\Genis-Internet\Output\Download output sub-directory

C:\Data\Genis-Internet\Temp directory for all temporary files

C:\Data\Genis-Internet\Download\Sql directory with five SQL-programs: *HtmlDownload.sql*, *HtmlDownloadCrop.sql*, *ListAllPassport.sql*, *HtmlCropNum.sql*, and *ListAllScores.sql*

C:\Data\Genis-Internet\Download\Xls directory with two MS Excel templates: *DefaultPassport.xlsx* and *DefaultTrait.xlsx*

C:\Data\Genis-Internet\Download\VBA directory with two Macro-Enabled MS Excel workbooks: *CreateCGNfiles-pasp.xlsm* and *CreateCGNfiles-ce.xlsm* and a text-file CreateCGNfiles\_YesNo.txt.

**General structure of the procedure**

The procedure has three elements:

1. Creating all HTML files with links to each other and to the downloadable Excel-files.
2. Creating the downloadable Excel-files with the passport data for each crop.
3. Creating the downloadable Excel-files with the scores for phenotypic data for each crop/trait combination.

Before the procedure is started it must be checked if the directories C:\Data\Genis-Internet\Output, C:\Data\Genis-Internet\Output\Download and C:Data\Genis-Internet\Temp are empty.

Each element can be run separately and after finishing an element, the Temp-directory can be emptied.

While running the procedures, it is not allowed to open MS Excel in another application since this will disrupt the process.

### **Element 1 - Creating the HTML files**

When the file *MakeHtmls.sql* is started in SQL the following will happen:

* the program *HtmlDownload.sql* starts and creates the output file named *Download.htm*
* the temporary SQL-program *StartAllHtmls.sql* is created and started
* *StartAllHtmls.sql* starts the SQLprogram *HtmlDownloadCrop.sql* for each crop, with its crop number given as a parameter
* *HtmlDownloadCrop.sql* creates the output file *DownnloadCnrNN.lst*, in which the NN stands for the crop number given as parameter
* the output files are renamed from *\*.lst* to the extension *\*.htm*
* a file *CropNum.htm* is written which contains the number of accessions per crop

At the end of this part there will be the files *Download.htm*, *CropNum.htm* and for each crop number a *DownnloadCnrNN.htm* in the Output directory, and a program *StartAllHtmls.sql* in the Temp-directory.

Since the crops without any phenotypic data will have the formulation ‘The phenotypic data can be downloaded per trait in Excel format.’ in its HTML file, it is possible to edit the smallest files manually by removing this sentence and the empty table below.

### **Element 2 - Creating the downloadable passport files**

For this element a program, located in the root directory, should be started from the SQLplus-prompt: *MakePassportFiles.sql.* The program does the following:

* the temporary SQL-program *StartAllPassport.sql* is created and started
* *StartAllPassport.sql* starts the SQL-program *ListAllPassport.sql* for each crop, with its crop number given as a parameter.
* *ListAllPassport.sql* creates the output file *AllPassportCnrNN.lst*, in which the *NN* stands for the crop number given as parameter; this file contains all passport information per crop
* a temporary file named *ListPassportCrops.txt* is created; this file is used as input for the Macro-Enabled MS Excel workbook CreateCGNfiles-pasp.xlsm
* the Macro-Enabled MS Excel workbook CreateCGNfiles-pasp.xlsm is started and creates per crop a output file named *CnrNNPassport.xlsx* in which *NN* stands for the crop number; the macro uses the temporary file *ListPassportCrops.txt* for input.

At the end there will be a downloadable Excel-file *CnrNNPassport.xlsx* for each existing crop in the /Output/Download directory, and the temporary programs *StartAllPassport.sql* and the files *AllPassportCnrNN.lst* in the directory Temp.

### **Element 3 - Creating the downloadable trait files**

Similar to the previous element, a program should be started from the SQLplus-prompt. The program *MakeTraitFiles.sql* The program does the following:

* the temporary SQL program *StartAllScores.sql* is created and started
* *StartAllScores.sql* starts the SQL program *ListAllScores.sql* for each crop, with its crop number given as a parameter
* *ListAllScores.sql* creates the output file *AllScoresCnrNN.lst*, in which the *NN* stands for the crop number given as parameter; this file contains all scores and decoding information per crop
* a temporary file named *ListScoreCrops.txt* is created; this file is used as input for the Macro-Enabled MS Excel workbook CreateCGNfiles-ce.xlsm
* the Macro-Enabled MS Excel workbook CreateCGNfiles-ce.xlsm starts and creates per crop/trait combination an output file named *CnrNNTraitNNN.xlsx* in which *NN* stands for the crop number, and NNN stands for the trait number; the macro uses the temporary file *ListCrops.txt* for input

At the end there will be a downloadable Excel-file *CnrNNTraitNNN.xlsx* for each existing crop/trait combination in the \Output\Download directory, and the temporary programs *StartAllScores.sql* and the files *AllScoresCnrNN.lst* in the Temp-directory.

## Procedure for creating the online searchable tables with passport and phenotypic data

### **Directory structure and required files**

The procedure for updating the on-line searchable passport and phenotypic data requires a simple directory structure and a number of files:

C:\Data\Genis-Internet\On-Line root directory with a SQL-program: *CreateGenisWeb.sql*

C:\Data\Genis-Internet\Temp directory for all temporary files

The new ORACLE passport tables are generated on the ORACLE-server. These tables are ‘mirror- tables’ of some GENIS tables, in a adapted format for the search and order facility on the CGN website.

The ORACLE phenotypic data tables are maintained by Roel Hoekstra and are also available from the ORACLE server. These tables contain information about phenotypic data in order to present the data as one score per accession on the CGN website.

All ORACLE tables are imported via a MS ACCESS database application into the SQL Server database that is used by the web application.

### **General structure of the procedure**

The procedure has three elements:

1. Creating ORACLE passport mirror tables within SQLplus.
2. Importing the Oracle tables to the MS ACCESS database
3. Updating the SQL Server database by replacing the contents of the tables with passport data and phenotypic data in this database with the data in the tables in the MS ACCESS database.

The elements have to be executed after each other.

It is necessary to log on to GENIS via SQLplus as CGNDBA. The update of the SQL Server is fully automated and triggers when the MS ACCESS database is copied. To access the test and production environment Microsoft Azure Storage Explorer is needed. ICT contact persons for help with installing and setting rights are Ivy Visser and Henk Drent.

### **Element 1 – Creating the passport mirror tables**

* Start the program *CreateGenisWeb.sql*. The passport mirror tables will be generated on the ORACLE Server.

### **Element 2 – Refreshing the SQL Server database**

* Start the macro refresh in the MS ACCESS database..

**Element 3**

* Start Microsoft Azure Storage Explorer.
* Go to the test environment: sub-psg-wpr-so-genis-ot ([emailadress@wur.nl](mailto:emailadress@wur.nl)).
* Copy genis\_web.mdb from C:\Data\Genis-Internet\On-Line to: Storage Accounts > euwstgenis001t > Blob Containers > master-data.
* After one hour you can check the new data at https://euw-web-genis-t.azurewebsites.net by following the Checklist request service in the “KMS\_bijlagen” folder.
* If the results of the test are positive copy genis\_web.mdb also to: sub-psg-wpr-so-genis-pr (the production environment). Follow the path as above, euwstgenis001t is here euwstgenis001p.

## Procedures for replacing files on the web server

### **Directory structure**

The website of CGN runs within the Contents Management System of Wageningen-UR.

The downloadable files can be approached via this CMS and are downloaded from a CGN share on an additional server (SCOMP6089) which is available especially for the GENIS web extension. E.g. all downloads, thumbnails and pictures are placed there.

On the CGN share there is a directory “Website” which has the following directory structure:

* Appel Phenotypic data for apple.
* DataNotAvailableAccessions Historic passport data available as download.
* Downloads All downloadable passport, phenotypic data and the pages that give access to them.
* Pictures A back-up of the images of accessions that can be approached via the search and order facility. Name of the pictures can be checked in this location before upload to the cloud via Microsoft Azure Storage Explorer.

### **General updating and back-up procedures**

All online searchable and downloadable data are updated every two months. Since the online searchable and downloadable passport information should be identical, the new files and tables should be generated within one day.

The online searchable phenotypic data differ from the downloadable. They are the result of extensive analysis summarizing the available (downloadable) data to the level of one score per accession per trait. Furthermore, only a selection of traits is available for online search.

The contents of the directory C:\Data\Genis-Internet\, except for the generated data, is saved as a back-up on the network directory N:\Documentatie\CGN\_Databases\Genis Archief\Genis-Internet.

### **Specific updating procedures**

### *Location of files*

For the downloadable files, the HTML output (CropNum.htm, *Download.htm*, and for each crop number a *DownloadCnrNN.htm*) should be copied into the \Downloads directory.

The Excel-files (for each crop number a *CnrNNPassport.xlsx* and for each crop/trait combination a *CnrNNTraitNNN.xlsx* file) must be copied into the \Downloads\download directory.

#### *Adding a new crop*

If there are passport data for a new crop in the database *Genis\_Web.mdb*, this is indicated in the table “GNS\_WEB\_CROP”. The latter table is used by the SQL-server for (automatically) making adjustments in the search and order facility, specifically for new crop. Besides adjustment in the search and order facility, a picture of the crop of sizes 100x100 pixels, and named *NN.jpg* in which NN is the crop number is used at the crop-specific search page. This picture is placed in the system by FB-ICT, which maintains the search and order facility for CGN.

#### *Adding phenotypic data*

If the phenotypic data for an additional crop can be searched on line, the table WEB\_GNS\_CROP should contain a ‘1’ in the field EVALUATION for that crop. This can be updated by adding the crop number in the SQL program in the file *CreateGenisWeb.sql*, before running it.

update GNS\_WEB\_CROP

set EVALUATION = '1'

where CNR in (6,41)

#### *Adding a core selector*

If the core selector can be used for an additional crop, the table WEB\_GNS\_CROP should contain a ‘1’ in the field CORE for that crop. This can be updated by adding the crop number in the SQL-program in the file *CreateGenisWeb.sql*, before running it.

update GNS\_WEB\_CROP

set CORE = '1'

where CNR in (6)

#### *Adding a new crop with a substantial number of pictures*

If there are enough pictures for a crop to let the users make selections for only accessions that contains pictures, the table WEB\_GNS\_CROP should contain a ‘1’ in the field PICTURES for that crop. This must be added in the program that fills this table: the crop number should be added in the SQL-program in the file *CreateGenisWeb.sql*.

update GNS\_WEB\_CROP

set PICTURES = '1'

where CNR in (6,41)

#### *Setting availability for a crop or an accession*

If, e.g. for phytosanitary reasons, a crop is not available for some time we can exclude this crop from requesting: the crop number should be modified in the SQL-program in the file *CreateGenisWeb.sql*.

update GNS\_WEB\_CROP

set REQUESTABLE = '0'

where CNR in

(37,38);

To exclude an accession from requesting:

update GNS\_WEB\_PASSPORT

set REQUESTABLE = '0'

where RNR in (

'922838',

'922889',

'923685',

'922573',

'922162',

'923502'

);

A reason for not being available of a crop can be given in the field ADDTEXT in GNS\_WEB\_CROP:

update GNS\_WEB\_CROP

set ADDTEXT = 'BlaBla'

where CNR in (38);

***Procedure for adding pictures on the web server***

Pictures are provided by the curators to the documentation group.

Format elements of the filenames of the pictures, separated by underscores:

* Receipt number (instead of the two first digits that represent the receipt year, four digits must be used)
* Crop number (two characters)
* Description (underscores instead of spaces, place en date between brackets)

*e.g.*: 19962501\_41\_overview\_plot\_(Elst\_2001).jpg

20010001\_02\_hairy\_leaf\_(Wageningen\_2002).jpg

The size of the pictures must be checked due to response time of the web site to display the pictures. The maximum size should not exceed about 400 Kb.

Thumbnails are generated by the documentation group. The size of the thumbnails must be approximately 100 pixels wide. Height is dependent on shape of the picture.

The pictures and the thumbnails must be copied to the appropriate directory on the CGN share, ..\Website\pictures\pictures\cnrXX and

..\Website\pictures\thumbnails\cnrXX, where XX is the crop number.

### **Checking integrity of pictures**

The integrity of the (name of the) pictures can be validated with the Excel Workbook “pictures.xls” located in the C:\Data\Genis-Internet\Download\pictures on the web folder. Open the Workbook and run the macro “check all”. Results are displayed on the sheet “log”. Before running the macro you must refresh the data on sheet “RNR’s”.

### **Uploading pictures to the cloud**

After the pictures are validated they must be copied to the test and production environment in the cloud with Microsoft Azure Storage Explorer.

For test:

* Go to the test environment: sub-psg-wpr-so-genis-ot ([emailadress@wur.nl](mailto:emailadress@wur.nl)).
* Copy the new pictures and thumbnails to: Storage Accounts > euwstgenis002t > Blob Containers > $web.

For production:

* Go to the production environment: sub-psg-wpr-so-genis-pr ([emailadress@wur.nl](mailto:emailadress@wur.nl)).
* Copy the new pictures and thumbnails to: Storage Accounts > euwstgenis002p > Blob Containers > $web.