Standard operating procedures for sweetpotato breeding data management

2020
SEPTEMBER

COP Breeding Data Management SweetGAINS

Standard operating procedures for sweetpotato breeding data management

COP Breeding Data Management SweetGAINS

September
2020

Standard operating procedures for sweetpotato breeding data management

COP Breeding Data Management SweetGAINS

ISBN: 978-92-9060-551-5
DOI: 10.4160/978929060-5515

International Potato Center
P.O. Box 1558, Lima 12, Peru
cip@cgiar.org • www.cipotato.org

Citation:
International Potato Center. (2020). *Standard operating procedures for sweetpotato breeding data management. COP Breeding Data Management SweetGAINS.* International Potato Center: Lima, Peru.

Design and Layout:
Communications Department

September 2020

CIP also thanks all donors and organizations that globally support its work through their contributions to the CGIAR Trust Fund: www.cgiar.org/funders

Acknowledgements

This work was undertaken as part of the CGIAR Research Program on Roots, Tubers and Bananas (RTB). Funding support for this work was provided by the Bill & Melinda Gates Foundation, investment OPP1213329, awarded to the International Potato Center (SweetGAINS). B. De Boeck was placed by the Centre for International Migration and Development (CIM), a joint operation of GIZ and the German Federal Employment Agency on behalf of the German federal government.

CIP thanks all donors and organizations that globally support its work through their contributions to the CGIAR Trust Fund: *www.cgiar.org/funders*

Table of contents

Introduction 1
1. Purpose 2
2. Scope 2
3. Responsibility 2
4. Procedures 3
SOP 1: Phenotyping data management 3
SOP 2: Crossing data management 10
SOP 3: Quality assessment data management 13
SOP 4: Germplasm inventory management 15
SOP 5: DNA sample management and tracking 18
References 21
Annex A. Sweetpotato naming conventions 22
1. Name structure 22
2. Abbreviations 22
  2.1. Country code 22
  2.2. Institution code 22
  2.3. Location code 22
  2.4. Trial type 22
  2.5. Cross experiment type 23
3. Examples 23
  3.4. Cross experiment example 23
  3.5. Cross unique id example 24
  3.6. Cross family name example 24
Annex B. Sweetpotato breeding data curation procedures 25
1. Install R and packages 25
  1.1. Download R 25
  1.3. Install the st4gi package 26
2. Check data 26
  2.1. Load the fieldbook data 26
  2.2. Check frequencies 26
  2.3. Check positions 27
  2.4. Check names 27
  2.5. Check consistency 27
    2.5.1. Rules 27
    2.5.2. Examples 28
    2.5.3. Change defaults for outliers detection 28
    2.5.4. A note on outliers' detection 28
3. Curate data 29
  3.1. Function setna 29
  3.2. Function setzero 29
  3.3. Function clean.data 30
  3.4. Function cdt 30
  3.5. Proposed steps for data curation 31
Annex C. How to Print Labels using the ZM400 Printer 32
Annex D. List of primary traits 34

Acronym list

CIP	International Potato Center
CSV	Comma-separated values
KSU	Kansas State University
NIRS	Near-infrared reflectance spectroscopy
SOP	Standard operating procedure
SPBase	Sweetpotatobase (https://sweetpotatobase.org/)

Standard operating procedures for sweetpotato breeding data management under the SweetGAINS project

Authors

Luka Wanjohi, l.wanjohi@cgiar.org;

Raúl Eyzaguirre, r.eyzaguirre@cgiar.org;

Bert De Boeck, b.deboeck@cgiar.org;

Contributors

Maria Andrade, m.andrade@cgiar.org;

Hugo Campos, h.campos@cgiar.org;

Edward Carey, e.carey@cgiar.org;

Doreen Chelangat, dmurenju@gmail.com;

Federico Diaz, f.diaz@cgiar.org;

Dorcus Gemenet, g.gemenet@cgiar.org;

Wolfgang Grüneberg, w.gruneberg@cgiar.org;

Fekadu Gurmu, fekadugb@gmail.com;

Simon Imoro, s.imoro@cgiar.org;

Some Koussao, koussao@hotmail.com;

Stanley Kwendani, skwendani@gmail.com;

Gonzaga Luis madroba@gmail.com;

Zakayo Machunde, zakayomachunde@gmail.com;

Godwill Makunde, g.makunde@cgiar.org;

Nwankwo Maxwell, nwankwomaxwell@yahoo.com;

Margaret McEwan, M.McEwan@cgiar.org;

Kiddo Mtunda, kjmtunda09@yahoo.co.uk;

Lukas Mueller, lam87@cornell.edu;

Denis Munyabarame, munybden13@gmail.com;

Robert Mwanga, r.mwanga@cgiar.org;

Abdul Naico, a.naico@cgiar.org;

Jean Ndirigwe, ndrick3@gmail.com;

Justin Njobvu, njobvujustin@yahoo.com;

Bonny Oloka, bonnymickael@gmail.com;

Guilherme Pereira, g.pereira@cgiar.org;

Jose Ricardo, j.ricardo1999@yahoo.com.br;

Damien Shumbusha, dshumbusha2@gmail.com;

Christiano Simões, ccs263@cornell.edu;

Ibrahim Somo, msomowork@gmail.com;

Reuben Ssali, r.ssali@cgiar.org;

Jolien Swanckaert, j.swanckaert@cgiar.org;

Titima Tantikanjana, tt15@cornell.edu;

Craig Yencho, yencho@ncsu.edu;

Introduction

Current modernization efforts of sweetpotato breeding operations in Africa establish a new mindset. A modern sweetpotato breeding program continuously generates vast amounts of data on which it depends for all decision making throughout the program. Without a proper systematization of efforts, it is likely that significant mistakes can be unwillingly made, which would impact in a negative manner both genetic gains and the adoption of new varieties by smallholders. This document describes standard operating procedures (SOPs) for implementing breeding data workflows to ensure that all necessary breeding data are recorded appropriately and made easily accessible. This document needs to be considered as an alive one, as through its ensuing iterations additional SOPs will be added, and the current ones would be modified to reflect the learnings acquired.

The data management SOPs in this volume cover the following key sweetpotato breeding data workflows: phenotyping, crossing, quality assessment, germplasm management, and DNA sample management. A relational database, SPBase (www.sweetpotatobase.org)[1], plays a central role as a breeding data management system across workflows. Several other digital tools have been developed to connect to SPBase to facilitate recording and uploading different types of data.

[1] Sweetpotatobase runs using software from Breedbase (www.breedbase.org), a comprehensive breeding management and analysis software.

1. Purpose

The purpose of this document is to describe the SOPs for data recording and data management in sweetpotato breeding activities. These activities include phenotyping, crossing, NIRS quality assessment, germplasm management and DNA sample management.

2. Scope

The SOPs apply to all breeding activities implemented under the SweetGAINS project and aim to reach the broader sweetpotato breeding community.

3. Responsibility

The breeding data management SOPs must be used by all staff involved in implementing breeding activities under SweetGAINS with no alteration unless approved exceptionally by the project's principle investigator. The table below shows the persons responsible for each individual SOP that comprise the program's entire approach to data collection, processing, and management.

SOP #	Procedure	Responsible
1	Phenotyping data management	Jolien Swanckaert; Godwill Makunde; Reuben Ssali; Doreen Chelangat; Fekadu Gurmu; Jose Ricardo; Kenneth Masamba; Zakayo Machunde; Denis Munyabarame; Justin Njobvu; Gonzaga Luis; Stanley Kwendani; Simon Imoro; Some Koussao; Nwankwo Maxwell; Abdul Naico; Federico Diaz; Luka Wanjohi; Raúl Eyzaguirre and Bert De Boeck
2	Crossing data management	Jolien Swanckaert; Godwill Makunde; Reuben Ssali; Doreen Chelangat; Fekadu Gurmu; Jose Ricardo; Kenneth Masamba; Zakayo Machunde; Denis Munyabarame; Justin Njobvu; Gonzaga Luis; Stanley Kwendani; Simon Imoro; Some Koussao; Nwankwo Maxwell, Abdul Naico; Federico Diaz; Luka Wanjohi; Raúl Eyzaguirre and Bert De Boeck
3	Quality assessment data management	Jolien Swanckaert; Godwill Makunde; Reuben Ssali; Simon Imoro; Abdul Naico and Luka Wanjohi
4	Germplasm inventory management	Luka Wanjohi and Rosemary Gatimu
5	Genotyping project management and tracking	Guilherme Pereira

4. Procedures

SOP 1: Phenotyping data management

1.1. Contacts for assistance

- Technical issues with SPBase: Christiano Simões, ccs263@cornell.edu.
- Technical issues with digital tools (Zebra printers, Fieldbook App), printing field labels and uploading data to Dataverse: Luka Wanjohi, l.wanjohi@cgiar.org, Abdul Naico, a.naico@cgiar.org, and Simon Imoro, s.imoro@cgiar.org.
- Experimental design, and uploading breeding program, trial location, trial design and trial data to SPBase, data curation and data analysis: Raúl Eyzaguirre, r.eyzaguirre@cgiar.org.
- Experimental design and data analysis: Bert De Boeck, b.deboeck@cgiar.org.

1.2. Applicability

The objective of this SOP is to describe the sweetpotato phenotyping data workflow. We have fixed procedures and the data workflow for conducting phenotypic trials and collection and storing the resulting phenotypic data. The workflow is built around www.sweetpotatobase.org.

1.3. Required tools

1.3.1. **SPBase** is a relational database that can manage many different types of breeding data and has been developed by the Boyce Thomson Institute (BTI) based at Cornell University. SPBase allows for instance to perform the following actions: define a germplasm list, design a field trial, select a trait list based on the standard sweetpotato ontology and create a field book compatible with Field Book App (see 1.3.2). SPBase generates a unique plot name for every trial plot created in the database. This plot name should be used as the unique identifier for all plot level data across a given breeding trial. This unique identifier should be available on field labels in the form of a 2D barcode, alongside other trial plot identification information (like accession name, plot number, replication number if applicable and certainly plot row and column coordinates).

1.3.2. **Field Book App**[2] is an Android-based mobile application for plant phenotyping. To setup data collection, a field layout file in CSV format and a traits list text file can be imported into the App. SPBase will automatically generate these files for all trials created on the platform. The app has been developed at KSU by POLAND LAB (https://wheatgenetics.k-state.edu/) and can be downloaded free of charge from the google play store: https://play.google.com/store/apps/details?id=com.fieldbook.tracker.

1.3.3. Zebra Designer Pro 2 Software

1.3.4. Zebra ZM/ZT series Printer

1.3.5. Premium Wax ribbons (110mm x 450M OW Wax AWR)

1.3.6. Z-Perform 1000T - 76171Plain PP White self-adhesive film labels special layout 3 Across 33 x 35 mm

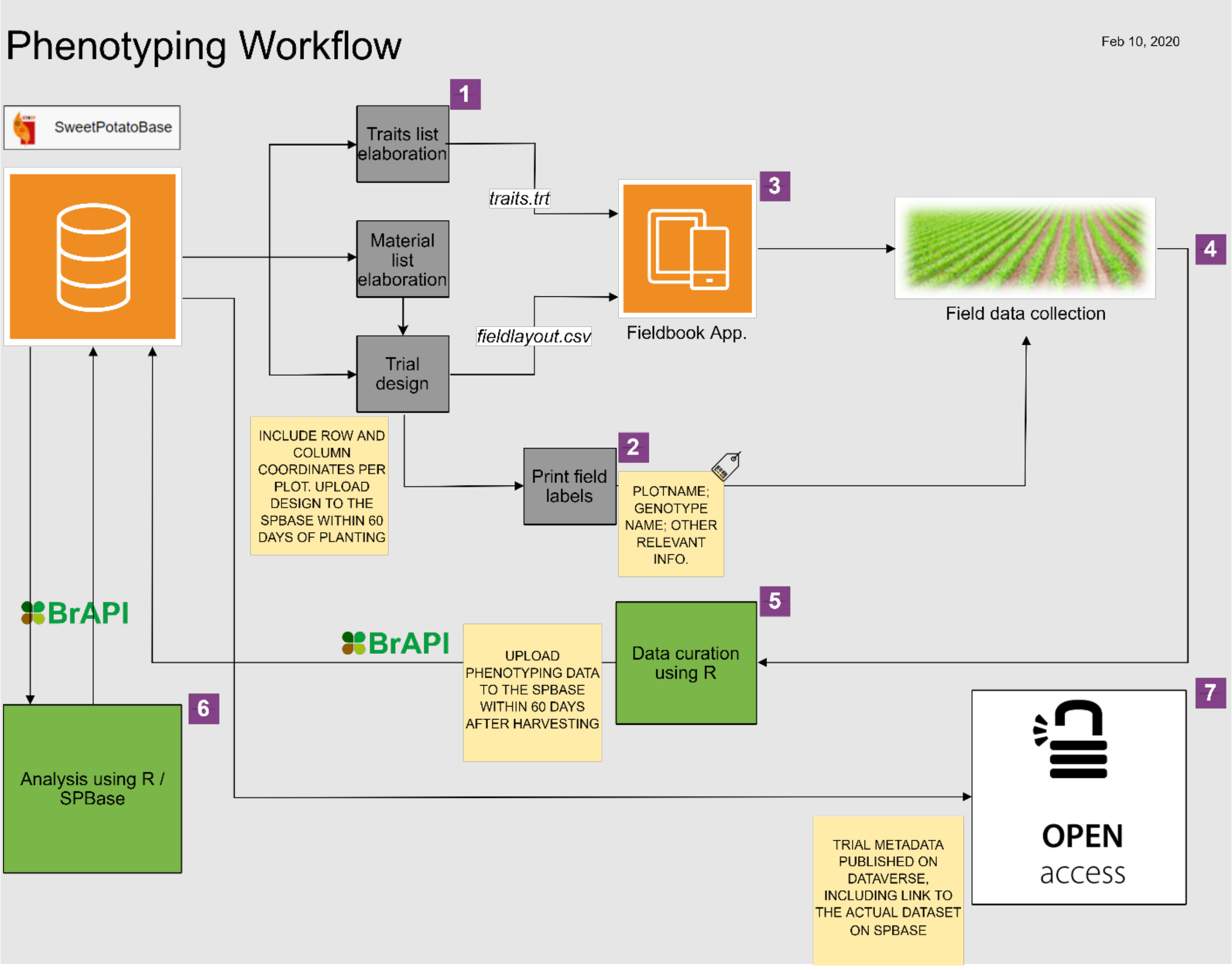
1.3.7. Self-tie labels (polyplas material, 150 microns, 4 across)

1.3.8. Android tablet

[2] Field Book is produced by PhenoApps (www.phenoapps.org).

1.4. Procedures:

Below you find the typical phenotyping workflow using SPBase and the Field Book App.



Phenotyping Workflow

Feb 10, 2020

INCLUDE ROW AND COLUMN COORDINATES PER PLOT. UPLOAD DESIGN TO THE SPBASE WITHIN 60 DAYS OF PLANTING

PLOTNAME; GENOTYPE NAME; OTHER RELEVANT INFO.

UPLOAD PHENOTYPING DATA TO THE SPBASE WITHIN 60 DAYS AFTER HARVESTING

TRIAL METADATA PUBLISHED ON DATAVERSE, INCLUDING LINK TO THE ACTUAL DATASET ON SPBASE

To help us serve all the breeding programs under SweetGAINS better, we require notifying Luka Wanjohi (l.wanjohi@cgiar.org), Raúl Eyzaguirre (r.eyzaguirre@cgiar.org) and Bert De Boeck (b.deboeck@cgiar.org) of any field trials during the planning phase by e-mail.

General information and procedures for sweetpotato field trials can be found in the "Procedures for the evaluation of sweetpotato trials" manual. The SOPs described below focus on the data management aspects and give an update about how to select the design of an experiment.

1.4.1. Experimental design selection for field trial

By an experimental design we refer to the way in which the different planting materials are allocated to the available plot positions in the field. Depending on the type of trial (mostly on the number of genotypes or treatments) there will be a more adequate experimental design. In principle there are three main types of experiments; their characteristics and the typical experimental design for each of them are described below:

- Observational trial: 1 single row plot with 3 planting positions per genotype; no replications except checks and parents; 2 locations; more than 500 genotypes; used for population improvement (many families and 4 to 12 genotypes per family) and elite crossings (few families and > 200 genotypes per family).

typical design: **augmented row-column design.**

- Preliminary yield trial: 2-3 row plots; each row has 10 to 15 planting positions; in total 4 replications planted across 3 locations (1.33 replications per location); 100 to 500 genotypes.

  typical design: **augmented p-rep design.**

- Advanced yield trial: 2-6 row plots; each row has 10 to 15 planting positions; 3 replications in at least 5 locations; less than 100 genotypes.

  typical design: **resolvable row column design**.

In all three cases listed above, consider a row distance of 100 cm and a planting distance within rows of 30 cm as a recommendation. Also, the standard use of check clones is recommended, ideally by including the global check clones Cemsa and Dagga (with accession names "Cemsa_74-228" and "CIP199062.1" in SPBase) and local varieties that are aimed to replaced by the product profile of the breeding program.

These are guidelines for design type depending on the type of experiment. In case of doubt, verification of the design type can be requested when sending out the notification that a field trial is being planned.

Before going to the next step, it is crucial that there is clarity about the design type, the available planting material and the exact field dimensions (number of plot rows and plot columns known); otherwise it is not possible to generate an adequate experimental design.

1.4.2. Create new trial in SPBase

Detailed step-by-step instructions on how to create a trial in SPBase are available here: https://solgenomics.github.io/sgn/03_managing_breeding_data/03_07.html.

1.4.2.1. Field trial can be designed in SPBase

Log into SPBase to add a new trial. The typical steps followed are:

- Navigate to *Manage Trials.*
- Begin the "Design new trial" workflow by clicking on *design new trial.*
- Enter all relevant meta data in "Trial Information" (e.g. trial name using naming convention, location, plot and field dimensions, trial type using mapping in naming convention, design type (see 1.4.1)).
- Enter "Design Information" (e.g. list of genotypes in trial (list should already be defined in SPBase)).
- Enter "Trial Linkage" information if applicable.
- Enter "Field Map Information" (keep *field map display* checked and enter the number of plot rows in the field).
- Custom Plot Naming (leave *plot prefix* empty, choose *plot start number* 1 and *plot number increment* 1).
- Review Designed Trial (check that the design layout corresponds with the situation of the field).
- Add Field Management Factors to your design (Optional by clicking the button below). This is required when treatment factors are applied to this experiment. The *Field Management Factor Name* is most important and will appear later in the fieldbook. Use for this factor name standard

treatment names like: “treatment_with_NPK”, “treatment_without_NPK”, “early-season-drought”, “mid-season-drought”, “late-season-drought” and “irrigation”.

- Save new trial in the database.

The accessions to be used in the new trial must be selected as a defined list of accessions (a new list of accessions can for instance be defined by navigating to *Search Accessions and Plots*, searching the intended accessions and clicking *add to new list*). The accessions themselves also already must exist in the database. This means that they have to be uploaded directly to the database (navigate to *Manage Accessions*) or, preferably, defined as progeny of a cross (navigate to *Manage Crosses* and follow the entire crossing data workflow; see SOP 2: Crossing data management). Care must be taken to ensure that the new accessions follow the **accession naming convention** in Annex A. Sweetpotato naming conventions. Similarly, the trial name should comply with the **trial naming convention** for new trials in Annex A. Sweetpotato naming conventions.

1.4.2.2. Type of design is not available (yet) in SPBase

Sometimes, it might not be possible to design your trial using SPBase due to limited trial design type options. When this is the case, the trial design can be generated outside of SPBase and then must be uploaded to the database. This can be done by navigating to *Manage Trials* and clicking on *Upload Existing Trial(s)* and running through the wizard (we recommend uploading every trial separately by choosing *Single Trial Design*; the required .xls file format template can be obtained).

In SweetGAINS it is a required to upload the plot row and column coordinates of each plot in the field trial. The variable fields *row_number* and *col_number* are thus required in the .xls file used to upload the trial. If there are checks (control accessions) the variable field *is_a_control* should also be used.

If necessary, assistance for the generation of an experimental design outside SPBase can be provided by Bert De Boeck (b.deboeck@cgiar.org) or Raúl Eyzaguirre (r.eyzaguirre@cgiar.org).

A commitment of SweetGAINS is to **upload all field trials to SPBase within 60 days after planting**. Assistance for this upload can be provided by Raúl Eyzaguirre (r.eyzaguirre@cgiar.org).

1.4.3. Print field labels

The typical steps followed when printing field labels are:

1.4.3.1. Creating an Excel or CSV field layout file based on the design of field trial
1.4.3.2. Designing your field label using Zebra Designer Pro 2 software based on the layout file
1.4.3.3. Printing your field labels using your Zebra ZM / ZT series printer
1.4.3.4. Labeling your plots in the field

To export the field layout, navigate to the trial detail page on SPBase and scroll down to the “Experimental Design” section. Click on the first tab labeled “Download Layout” and follow the wizard to download your file. The exported file will contain amongst other fields the plot name, plot ID, accession name, plot number, block number, replication number, row number and column number. The Zebra Designer Pro 2 software will help you select which information to include in your label from the downloaded file per plot. Each label should include at least the plot name as a 2D barcode, plot number, accession number and replication number. Other additional information could be planting date and location. The recommended labels for field labeling are Self-tie labels (polyplas material, 150 microns, 4 across). See

Annex C. How to Print Labels using the ZM400 Printer for detailed instructions on how to print field labels using the Zebra Designer Pro 2 software and the Zebra ZT/ZM series of printers.

Whereas it is possible to design and print labels using SPBase directly, we recommend printing of field labels using the Zebra suite of tools because of the diverse types of labels that can be printed using this method, including labels for other breeding activities outside of SPBase.

1.4.4. Field data collection using the Field Book App

The list of traits being recorded depends on what the phenotyping experiment is targeting, and in general we expect all traits present in the breeding program's product profiles to be of interest. For reasons of data comparability, a list of **primary traits** that must be recorded in each phenotyping experiment was established in Annex D. List of primary traits. More traits can be recorded if there are sound reasons to do so, but the list of **primary traits** is an absolute minimum set of traits that must be recorded.

Steps:

1.4.4.1. Creating a field layout file based on the design of the field trial

- Navigate to *Manage Trials* and select the desired trial to open the "trial detail page" for the desired trial by browsing to and double clicking on it.
- Scroll down the page and expand the *Upload Data Files* section.
- On the *Android Field Book Layout* row click on *Create Field Book* to generate the layout file.
- A new dialogue box will appear with the trial details and options to select spreadsheet format and desired data level.
- Click S*ubmit* and click on the link popping up to download layout file.

1.4.4.2. Creating a trait file from the list of traits

- Navigate to menu item *Search Traits* to open the "trait search page".
- Search desired traits by ID, name or description.
- Select desired trait in the results.
- Add selected results to a trait list (*This could be an existing trait list or a new trait list. It is possible to create a new trait list on the fly).*
- Navigate to menu item "Manage" then "Field Book App".
- Find the heading "Trait Files" and click on *new* to create your file.
- A new dialogue box will open. Select the *traits list* that you created from the drop-down list.
- Check the box titled "Include Notes Trait" if you would also like to record and upload general plot notes in the field.
- Type in an appropriate file name and click on *Submit* to download your traits file.

1.4.4.3. Downloading the field layout file and trait file from the database to your computer

Your field layout and trait files are now downloaded to your computer. The next step is to copy these to your tablet.

1.4.4.4. Copy the field layout file and trait file to the tablet with the Field Book App

To copy the field layout and trait file to your android tablet:

- Connect your android tablet to your computer using via a USB cable.
- Copy the field layout file into the *fieldBook>field_import* folder in the tablet.
- Copy the traits list files into the *fieldBook>trait* folder.
- Open the field Book app to import the new field layout and Traits files.

1.4.4.5. Collecting phenotypes

An important aspect when collecting phenotypic data at harvest is dealing with missing values. Ideally, rules for that must be included in the data collection device, like Field Book App, but if not the case, consider the following guidelines.

Missing values occur when:

- All the plants died in a plot due to external reasons (not related with the quality of the genotype).
- For some reason it was not possible to record a value in a random plot (again not related with the genotype assigned to that plot; e.g. yield of a plot that simply was forgotten to be recorded by a technician).
- There were no roots or vines to evaluate a trait that needs roots or vines. For example, roots are needed to evaluate dry matter, so if there are no roots, dry matter is a missing value.

A frequent error is to write zero for a missing value or the other way around. **A zero and a missing value are different** because in the first case a value is observed for the trait, while in the second it is indicated that the value is unknown. This makes a big difference for data analysis and is not a detail!

Two important but very different cases are the following:

- If all the plants die in a plot because the genotype was not strong enough, set zero for the number of harvested plants and all the yield traits observed at harvest. Post-harvest traits like the quality traits (e.g. minerals, sugars) that need some roots for evaluation, however, should be recorded as missing value when there are no roots available to get the evaluations.
- If for some reason the results of a random plot were impossible to observe (e.g. all the plants died because of some external reason like flooding but independent from the genotype that was assigned to the plot, the plot was destroyed by a truck going off the road, …), a missing value must be assigned to the number of harvested plants and all the traits that would be observed at harvest and post-harvest.

1.4.4.6. Exporting phenotypes from Field Book App to your computer

Step-by-step instructions on how to achieve the above are available here.

The Field Book App is available on Google Play for free. To download the app, you will need to be connected to the internet. Internet connection is not necessary during data collection. The App allows you to disable moving to the next field during data collection if no data has been collected. This can be enabled in the advanced settings after installation. Also GPS location data can be recorded with Field Book App, but much more important is to have the trial location defined in SPBase when defining the trial in step 1.4.2.

Most Android phones and tablets can be used for field data collection. A typical 7” to 8” android tablet will be enough for the data collection needs of most breeding programs.

1.4.5. Data Curation

After the collection of phenotypic data with Field Book App, the phenotype data files can be uploaded to your computer. Typically, this will be done partially on different occasions since data recording in the field is often done by several users on several days. We recommend taking regular back-ups on the computers, and even better on a cloud server. Ideally the data on every tablet should be exported at least every day.

These partial phenotype data files can be uploaded also partially to SPBase but it is important to not forget to curate the data by checking it for inconsistencies and correcting it if necessary. In the future, data curation will be made possible within SPBase so that the partial uploads would be recommended, and phenotypic data of a trial will be signaled to be curated and complete, or not.

For now, however, the data can best be merged on your computer so that it can be curated and then be uploaded clean to SPBase (please keep in mind that merging is always risky: If you are going to merge files with information for different sets of plots, make sure of correctly aligning the column names for all the files; in the case of merging files with different traits, make sure of correctly aligning the plot ids). In particular, the collected data should be checked for accuracy of accession (and other) names and consistency of the phenotyping data before uploading. Guidelines and R based tools for data curation are described in

Annex B. Sweetpotato breeding data curation procedures.
Raúl Eyzaguirre (r.eyzaguirre@cgiar.org) is in charge of all data curation for SweetGAINS and **should be notified whenever phenotypic data is to be uploaded** to SPBase, so the quality of the data can be checked.

1.4.6. Upload collected data to SPBase.

A commitment of SweetGAINS is to **upload all field trial data to SPBase within 60 days after data collection**.

1.4.7. Data analysis

Trial data can be analyzed by downloading the data so statistical software like R can be applied to it. Soon some standard analyses will be available directly in SPBase as well.

SOP 2: Crossing data management

2.1. Contacts for assistance:

Luka Wanjohi, l.wanjohi@cgiar.org; Bert De Boeck, b.deboeck@cgiar.org; Raúl Eyzaguirre, r.eyzaguirre@cgiar.org; Abdul Naico, a.naico@cgiar.org; Jolien Swanckaert, j.swanckaert@cgiar.org; Godwill Makunde, g.makunde@cgiar.org;

2.2. Applicability:

The objective of this SOP is to describe the sweetpotato crossing and pedigree data workflows.

2.3. Required tools:

2.3.1. SPBase: Besides management of breeding field trials, SPBbase allows a breeder to manage crossing experiments. Breeders can define a crossing experiment specifying a name, the breeding program, the location, the year and a description of the crossing experiment. Individual crosses are issued with globally unique cross ID. Supported cross types include biparental, self, open and polycross. Each individual cross on plant, plot or accession level (choice of the breeder) in a crossing experiment will be associated with a *cross_unique_id*, which is unique in the database. To each individual cross or *cross_unique_id* a *cross_type*, parental information, *Number of Flowers, Number of Fruits*, *Number of Seeds* and other information can be added. Crosses (determined by a *cross_unique_id)* can be grouped together as a cross family (which we define as a genetic family per crossing experiment). Progeny from a cross can be named easily automatically becoming accessions in SPBase with known pedigree.

2.3.2. Intercross App: The Intercross app is a generic cross tracking application. The application has been designed by KSU to run on the mobile Android platform. Breeders can, by scanning labels of the parents or the cross itself, easily record cross information through the mobile application. New crossing data are stored on the local database of the application and later uploaded to SPBase.

2.3.3. Zebra Designer Pro 2 Software

2.3.4. Zebra ZM/ZT series Printer

2.3.5. Zebra Mobile Printer

2.3.6. 2D Barcode Handgun Scanner

2.3.7. Premium Wax ribbons (110mm x 450M OW Wax AWR)

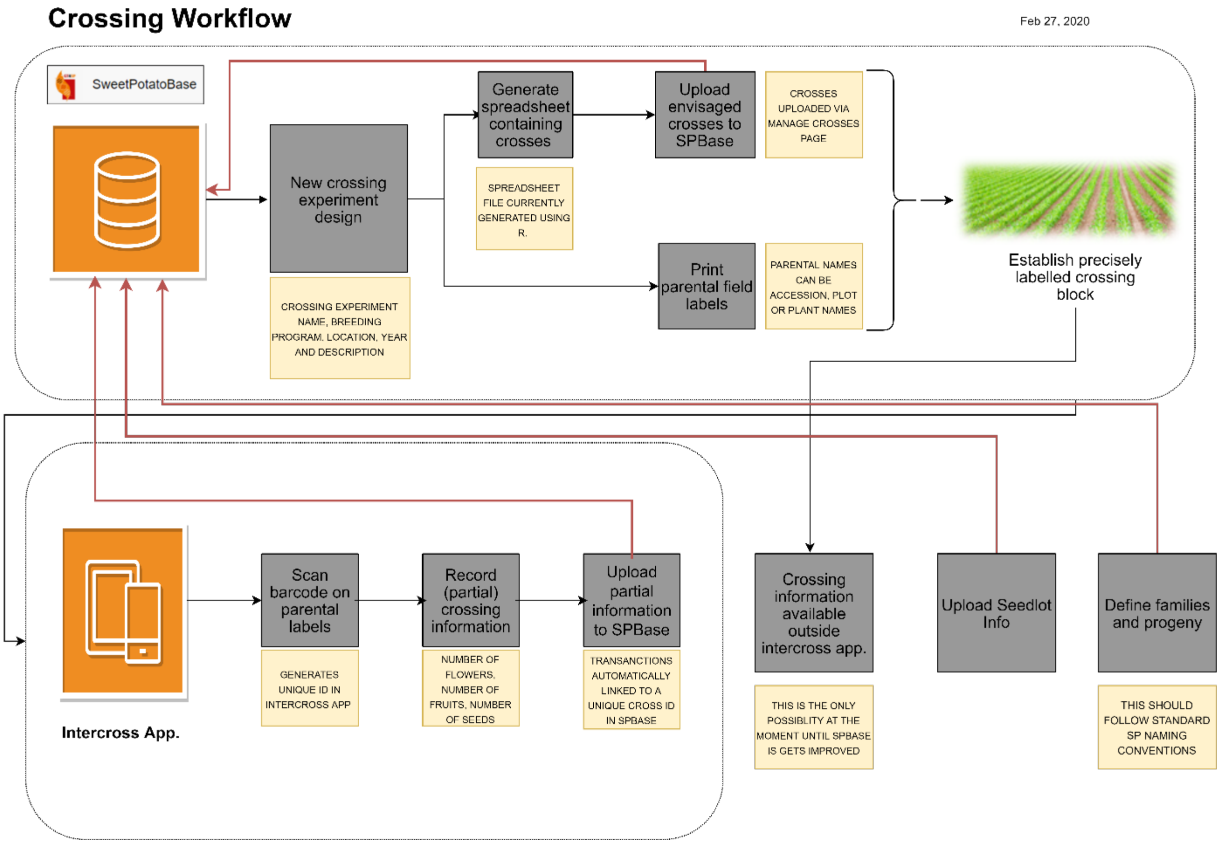
2.3.8. Z-Perform 1000T - 76171Plain PP White self-adhesive film labels special layout 3 Across 33 x 35 mm

2.3.9. Self-tie labels (polyplas material, 150 microns, 4 across)

2.3.10. Android Tablet

2.4. Procedures:

Below you find the typical crossing workflow using SPBase and Intercross App.


Crossing Workflow
Feb 27, 2020
SweetPotatoBase
New crossing experiment design
CROSSING EXPERIMENT NAME, BREEDING PROGRAM, LOCATION, YEAR AND DESCRIPTION
Generate spreadsheet containing crosses
SPREADSHEET FILE CURRENTLY GENERATED USING R.
Upload envisaged crosses to SPBase
CROSSES UPLOADED VIA MANAGE CROSSES PAGE
Print parental field labels
PARENTAL NAMES CAN BE ACCESSION, PLOT OR PLANT NAMES
Establish precisely labelled crossing block
Intercross App.
Scan barcode on parental labels
GENERATES UNIQUE ID IN INTERCROSS APP
Record (partial) crossing information
NUMBER OF FLOWERS, NUMBER OF FRUITS, NUMBER OF SEEDS
Upload partial information to SPBase
TRANSACTIONS AUTOMATICALLY LINKED TO A UNIQUE CROSS ID IN SPBASE
Crossing information available outside intercross app.
THIS IS THE ONLY POSSIBLITY AT THE MOMENT UNTIL SPBASE IS GETS IMPROVED
Upload Seedlot Info
Define families and progeny
THIS SHOULD FOLLOW STANDARD SP NAMING CONVENTIONS

Note that the sweetpotato naming conventions specified in Annex A. Sweetpotato naming conventions should be used to name crossing experiment, cross unique id and cross family. Also, for the cross_combination field we suggest using all the same format "*female-male*", so it determines the genetic family.

2.4.1. Establish Crossing Experiment

The typical steps when adding a new crossing experiment to SPBase are:

2.4.1.1. Navigate to *Manage Crosses*.

2.4.1.2. Click on *Add Crossing Experiment*.

2.4.1.3. Enter experiment details: Crossing experiment name (naming convention!), breeding program, location, year of start crossing experiment and description.

2.4.1.4. Upload all the envisaged crosses to SPBase on the *Manage Crosses* page by clicking on *Upload Crosses*. This can be on accession level, but also on plot or plant level. In the latter case a field trial with those plots or plants already must exist in SPBase. An excel containing all the crosses has to be created. For now, this cannot be done in SPBase so we recommend to use R to create this .xls file.

2.4.1.5. Print barcode labels with parental names, which can be accession, plot or plant names. If you use the plant or plot level to define crosses, the crossing experiment will already have to exist as field trial in SPBase. In this particular case the trial detail page can be used to generate the labels, but more generally and easy it is possible to generate the labels of the parents (on the accession, plot or plant level) directly on the crossing experiment page (navigate to and double click on the crossing experiment name on the *Manage Crosses* page, and click on *Generate Barcode Labels* under *Crosses in this experiment*).

2.4.1.6. Establish the precisely labelled crossing block.

2.4.2. Add (partial) crossing information

2.4.2.1. Each transaction that is done in the field (pollination or harvest) has to be recorded by using Intercross App. When scanning the male and female parent (accession, plot or plant) this generates an ID in the Intercross App, corresponding to a transaction ID, and typical crossing information (*Number of Flowers, Number of Fruits*, *Number of Seeds*) can be recorded.

2.4.2.2. It will be made possible to upload this crossing information in SPBase, for available groups of transactions. Since the male and female parent define the cross of an experiment uniquely, every transaction will be automatically connected to a *cross_unique_id*. Summary information for each *cross_unique_id* will be available since 1 *cross_unique_id* can correspond to various transactions.

2.4.2.3. The advantage of being able to upload crossing information partially is that an intermediate score of the different crosses can be pulled from SPBase, making a more targeted crossing approach possible. It is good practice to generate a wish list for crosses you still want to make. This can be done in R, or in the future possibly in SPBase.

2.4.2.4. If the complete crossing information is already available, without making use of Intercross App, it is also possible to directly upload this to SPBase, by using the right template. **Note that for now this is the only possibility because the above approach is not yet in production.** For this go again to the crossing experiment page and click on *Upload Cross Info of Existing Crosses* under *Cross Info* and upload the crossing information by using the proposed template.

2.4.3. Upload Seedlot Info

2.4.4. Define families and progeny using the naming convention described in Annex A.

2.4.4.1. Go to the crossing experiment page and click on *Upload family names* under *Progenies and Family Names* and upload the family information by using the proposed template. Note that each *family_name* corresponds to 1 or multiple *cross_unique_id*. A *cross_unique_id* can only correspond to 1 single *family_name.*

2.4.4.2. Go to the crossing experiment page and click on *Upload progeny names that are NOT in database* under *Progenies and Family Names* and upload new progenies by using the proposed template.

SOP 3: Quality assessment data management

3.1. Contacts for assistance:

Luka Wanjohi, l.wanjohi@cgiar.org; Abdul Naico, a.naico@cgiar.org; Godwill Makunde, g.makunde@cgiar.org and Jolien Swanckaert, j.swanckaert@cgiar.org

3.2. Applicability:

Harvested sweetpotato roots are sent to the laboratory for assessment of quality traits (for example dry matter, beta-carotene). The samples could either be roots from all the harvested plots in a trial or roots from randomly selected plots in the trial. It is imperative that the data management in the laboratory is made efficient so that it is easy to clearly identify samples, link these samples back to their respective plots in the trial and report on the status of the analysis of samples delivered to the laboratory.

The objective of this module therefore is to describe the proposed electronic data management workflow for sweetpotato breeding trials root samples in the laboratory.

3.3. Required tools:

3.3.1. Sweetpotatobase

3.3.2. Zebra Designer Pro 2 Software

3.3.3. Zebra ZM/ZT series Printer

3.3.4. Zebra Mobile Printer

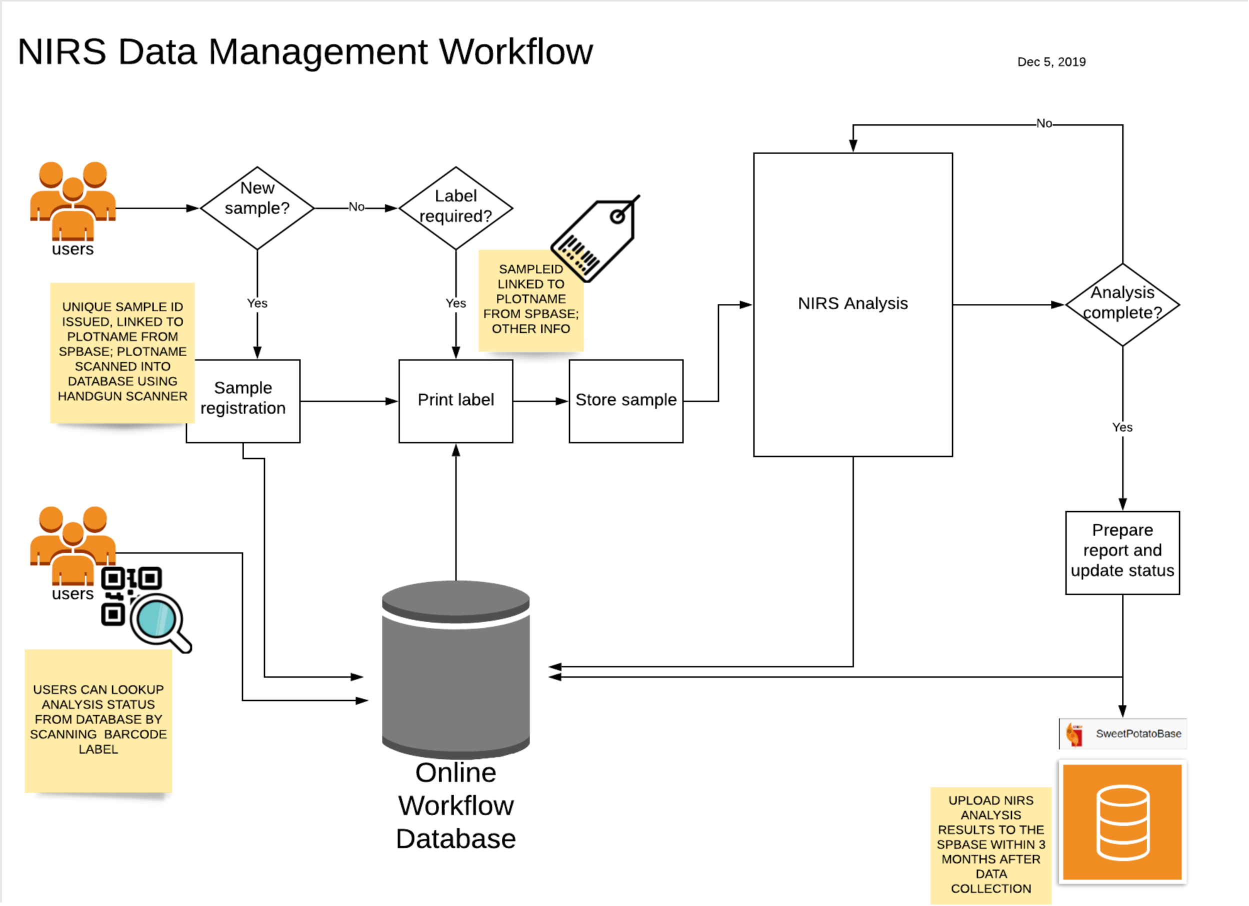
3.3.5. 2D Barcode Handgun Scanner

3.3.6. Premium Wax ribbons (110mm x 450M OW Wax AWR)

3.3.7. Z-Perform 1000T - 76171Plain PP White self-adhesive film labels special layout 3 Across 33 x 35 mm

3.4. Procedures

Below is a typical workflow.


NIRS Data Management Workflow
Dec 5, 2019
users
New sample?
No
Label required?
Yes
Yes
UNIQUE SAMPLE ID ISSUED, LINKED TO PLOTNAME FROM SPBASE; PLOTNAME SCANNED INTO DATABASE USING HANDGUN SCANNER
SAMPLEID LINKED TO PLOTNAME FROM SPBASE; OTHER INFO
Sample registration
Print label
Store sample
NIRS Analysis
No
Analysis complete?
Yes
Prepare report and update status
users
USERS CAN LOOKUP ANALYSIS STATUS FROM DATABASE BY SCANNING BARCODE LABEL
Online Workflow Database
SweetPotatoBase
UPLOAD NIRS ANALYSIS RESULTS TO THE SPBASE WITHIN 3 MONTHS AFTER DATA COLLECTION

3.4.1. Roots Harvesting

Prepare samples to be sent to the NIRS laboratory during trial harvest. The samples should be packed in bags with a clear barcode label. For breeding trials the label should contain: a barcode linked to the plot name generated by SPBase, and possibly useful information (e.g. plot name, genotype, row and column coordinates, replication block). Barcode labels can be printed ahead of time before going to the field or in the field using the Field Book app and the mobile Bluetooth Zebra label printer. See Annex C for detailed instructions on how to print labels.

3.4.2. Sample Reception

Roots samples are received at the laboratory. Using a handgun barcode scanner to read the label on the samples as they come in from the field will output the plot name as text. These can be read out into an excel worksheet. During registration, the following data should be recorded per sample: plot name from SPBase, date sample received, storage location in laboratory and a brief description of general status of the sample

3.4.3. Generate a new label for the samples after receiving these in the NIRS laboratory. Label should include new sample ID issued in laboratory.

3.4.4. Perform analysis

3.4.5. Upload results to Sweetpotatobase once analysis is finalized. All results must be uploaded to Sweetpotatobase within three months since the data was collected (when analysis results are shared).

SOP 4: Germplasm inventory management

4.1. Contact:

Luka Wanjohi, l.wanjohi@cgiar.org; Bert De Boeck, b.deboeck@cgiar.org; Margaret McEwan, m.mcewan@cgiar.org; Maria Andrade, m.andrade@cgiar.org,abdul Naico, a.naico@cgiar.org and Godwill Makunde, g.makunde@cgiar.org.

4.2. Applicability:

Breeding programs maintain sweetpotato germplasm for use within the program and for sharing the same with partners. Germplasm is maintained either in the form of tissue culture, as potted plants with double protection in the screen house or both as tissue culture and potted plant. Virus clean-up is done on all germplasm before conservation or distribution. Efficient data management workflows are required to support these activities of germplasm conservation, multiplication and distribution.

The objective of this SOP therefore is to describe the proposed electronic data management workflow for sweetpotato germplasm management in the screenhouse and the tissue culture laboratory, and seed system and breeding germplasm distribution.

4.3. Required tools:

4.3.1. Sweetpotatobase

4.3.2. Zebra Designer Pro 2 Software

4.3.3. Zebra ZM/ZT series Printer

4.3.4. Zebra Mobile Printer

4.3.5. 2D Barcode Handgun Scanner

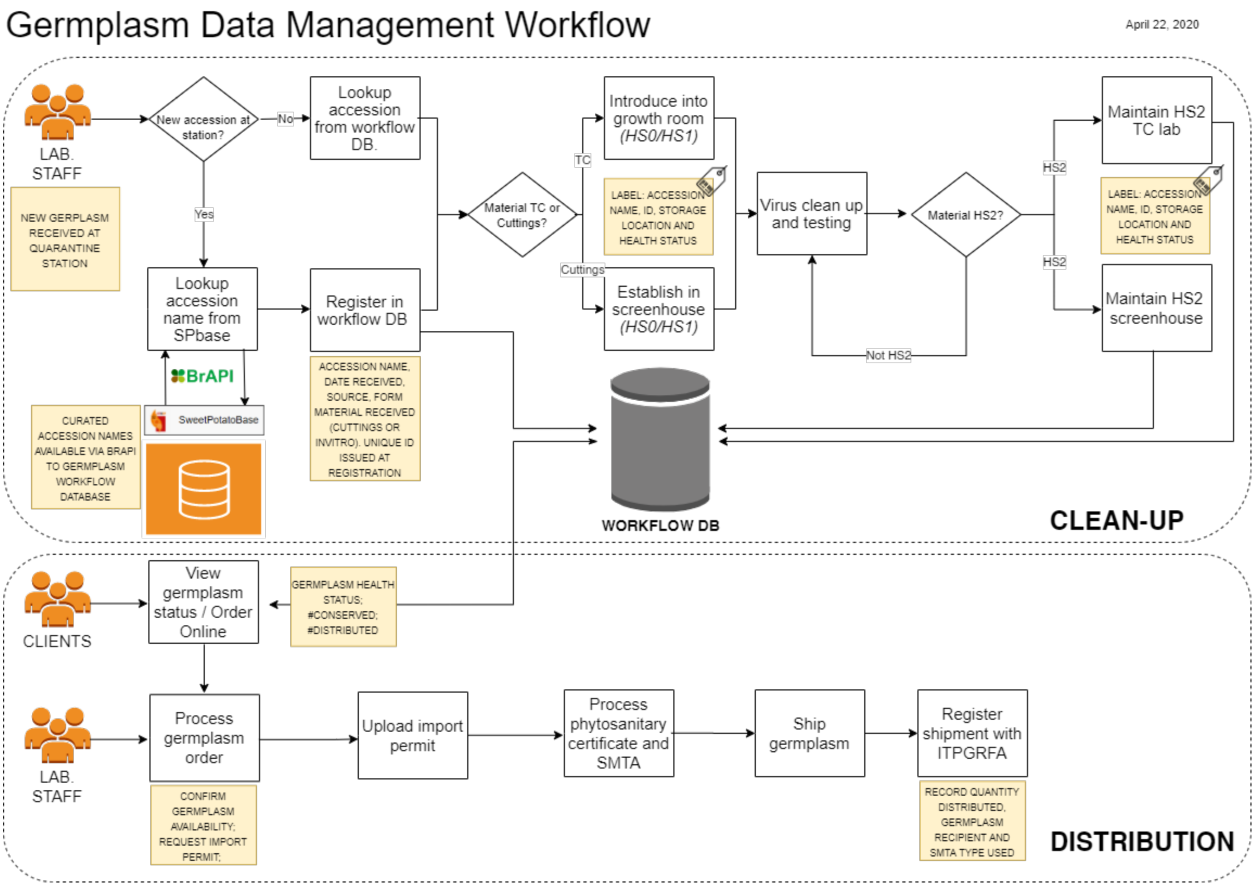
4.3.6. Premium Wax ribbons (110mm x 450M OW Wax AWR)

4.3.7. Z-Perform 1000T - 76171Plain PP White self-adhesive film labels special layout 3 Across 33 x 35 mm

4.3.8. Self-tie labels (polyplas material, 150 microns, 4 across)

4.4. Procedures:

Below is a typical workflow.



Germplasm Data Management Workflow

April 22, 2020

LAB. STAFF
New accession at station?
No
Yes
Lookup accession from workflow DB.
NEW GERPLASM RECEIVED AT QUARANTINE STATION
Lookup accession name from SPbase
Register in workflow DB
BrAPI
SweetPotatoBase
CURATED ACCESSION NAMES AVAILABLE VIA BRAPI TO GERMPLASM WORKFLOW DATABASE
ACCESSION NAME, DATE RECEIVED, SOURCE, FORM MATERIAL RECEIVED (CUTTINGS OR INVITRO). UNIQUE ID ISSUED AT REGISTRATION
Material TC or Cuttings?
TC
Cuttings
Introduce into growth room (HS0/HS1)
LABEL: ACCESSION NAME, ID, STORAGE LOCATION AND HEALTH STATUS
Establish in screenhouse (HS0/HS1)
Virus clean up and testing
Material HS2?
HS2
HS2
Not HS2
Maintain HS2 TC lab
LABEL: ACCESSION NAME, ID, STORAGE LOCATION AND HEALTH STATUS
Maintain HS2 screenhouse
WORKFLOW DB
CLEAN-UP
CLIENTS
View germplasm status / Order Online
GERMPLASM HEALTH STATUS; #CONSERVED; #DISTRIBUTED
LAB. STAFF
Process germplasm order
CONFIRM GERMPLASM AVAILABILITY; REQUEST IMPORT PERMIT;
Upload import permit
Process phytosanitary certificate and SMTA
Ship germplasm
Register shipment with ITPGRFA
RECORD QUANTITY DISTRIBUTED, GERMPLASM RECIPIENT AND SMTA TYPE USED
DISTRIBUTION

4.4.1. Germplasm sample is received at the station. Key details are registered such as the accession name, the form of received germplasm (cuttings or in-vitro), source country, purpose germplasm has been sent to station (acquisition, conservation or pathogen elimination), date germplasm received and details of the person sending the germplasm. Accession names are provided by the person sending the germplasm. A unique serialized sample ID is issued by the electronic system (in the format: SIDYEARSTATIONCODESERIALCODE, E.G. SID20201334)

4.4.2. The health status of germplasm samples is assumed to be virus positive (HS0) upon acquisition.

4.4.3. Germplasm samples in tissue culture form are introduced into the HS0 growth rooms for multiplication in preparation for virus cleanup and testing.

4.4.4. Germplasm samples in cuttings form are established in a HS0 screenhouse for multiplication. After multiplication, copies are introduced into the HS0 growth rooms in preparation for virus cleanup and testing.

4.4.5. The HS0 copies of the germplasm sample from 4.4.3 or 4.4.4 are subjected to thermotherapy treatment, in tissue culture form. In the online workflow, the start and end dates of thermotherapy treatment per germplasm sample are recorded.

4.4.6. After thermotherapy treatment, meristem-tip culture is performed. At least three copies of meristem tips per germplasm sample are prepared. The date the meristem tip culture is performed is recorded in the online workflow.

4.4.7. To perform virus testing, a copy of the excised meristem in 4.4.6 above is grafted onto an indicator plant. The commonly used indicator plant is *Ipomoea setosa*. The grafted *Ipomoea setosa* plant is then hardened and established in a greenhouse.

4.4.8. The remainder meristem tip copies are maintained in the TC labs and screenhouses. In the online workflow, these are assigned health status HS1. This is to indicate that the samples have undergone thermotherapy treatment and are now awaiting virus testing.

4.4.9. The grafted *Ipomoea setosa* in 4.4.7 is observed for Sweetpotato Virus Disease (SPVD) symptoms. These are recorded in the online workflow.

4.4.10. Virus detection and identification are then performed on the grafted *Ipomoea setosa* by either serological (NCM-ELISA) or molecular (PCR) tests. The date of testing and the test results are recorded in the online workflow. Test results are recorded simply as either positive or negative.

4.4.11. If the test results in 4.4.10 above are negative, the health status of the germplasm in the online workflow will be automatically updated to HS2 (virus negative). The backup HS1 copies in 4.4.8 will be multiplied and conserved as HS2 copies of the germplasm sample. Conservation is done both in the tissue culture laboratory HS2 growth room and in the HS2 screenhouse.

4.4.12. If the test results in 4.4.10 turn positive, new samples are prepared for thermotherapy treatment using the HS0 backup copies from 4.4.3 or 4.4.4. The thermotherapy and virus detection and identification processes are then repeated as detailed above. If the grafted samples keep returning a positive result after several cleaning and testing cycles, all the material from that germplasm sample are discarded and fresh copies ordered from the source. The status of the germplasm sample is then updated as ELIMINATED.

4.4.13. The physical location of all materials in tissue culture laboratories and screenhouses are always recorded and updated. Location information for lab material: Growth room number, shelf number, shelf row number and shelf column number. Location information for screenhouse material: Screenhouse name and bench number. All materials must always be labeled with an appropriate label indicating the: Sample ID, accession name, health status and the reason for being maintained / conserved.

4.4.14. The health status of all germplasm is available to members of the public via the online germplasm workflow. This allows anyone sending germplasm for cleanup to easily follow the status. Germplasm orders can be placed directly on the system via an online web interface. Once the processing of a given order commences, the person ordering germplasm will receive an acknowledgement and be requested to upload an import permit online.

4.4.15. All international distributions of sweetpotato germplasm must be accompanied with a duly completed standard material transfer agreement (SMTA). In compliance with the International Treaty on Plant Genetic Resources for Food and Agriculture (ITPGRFA), all SMTAs concluded in accordance with the instructions made by the governing body of the international treaty must be reported back. To facilitate compliance with this requirement, all information required for reporting purposes will be recorded in the germplasm workflow management database each time a distribution is made.

SOP 5: DNA sample management and tracking

5.1. Contact:

Guilherme Pereira, g.pereira@cgiar.org; Luka Wanjohi, l.wanjohi@cgiar.org; Bert De Boeck, b.deboeck@cgiar.org and Dorca Ndege (d.ndege@cgiar.org).

5.2. Applicability:

The objective of this SOP is to describe the proposed electronic data management workflow for sweetpotato genotyping project management and tracking. Seed from breeding trials is germinated, tracked, and performance evaluated in tissue culture and open field. DNA samples are obtained from leaves of germinated seed and shipped to North Carolina State University for analysis. DNA analysis results are uploaded back to corresponding plot name in SPBase.

5.3. Required tools:

5.3.1. Sweetpotatobase

5.3.2. Coordinate App

5.3.3. CIPTCL (CIP Tissue Culture Lab. Management System)

5.3.4. Zebra Designer Pro 2 Software

5.3.5. Zebra ZM/ZT series Printer

5.3.6. Zebra Mobile Printer

5.3.7. 2D Barcode Handgun Scanner

5.3.8. Premium Wax ribbons (110mm x 450M OW Wax AWR)

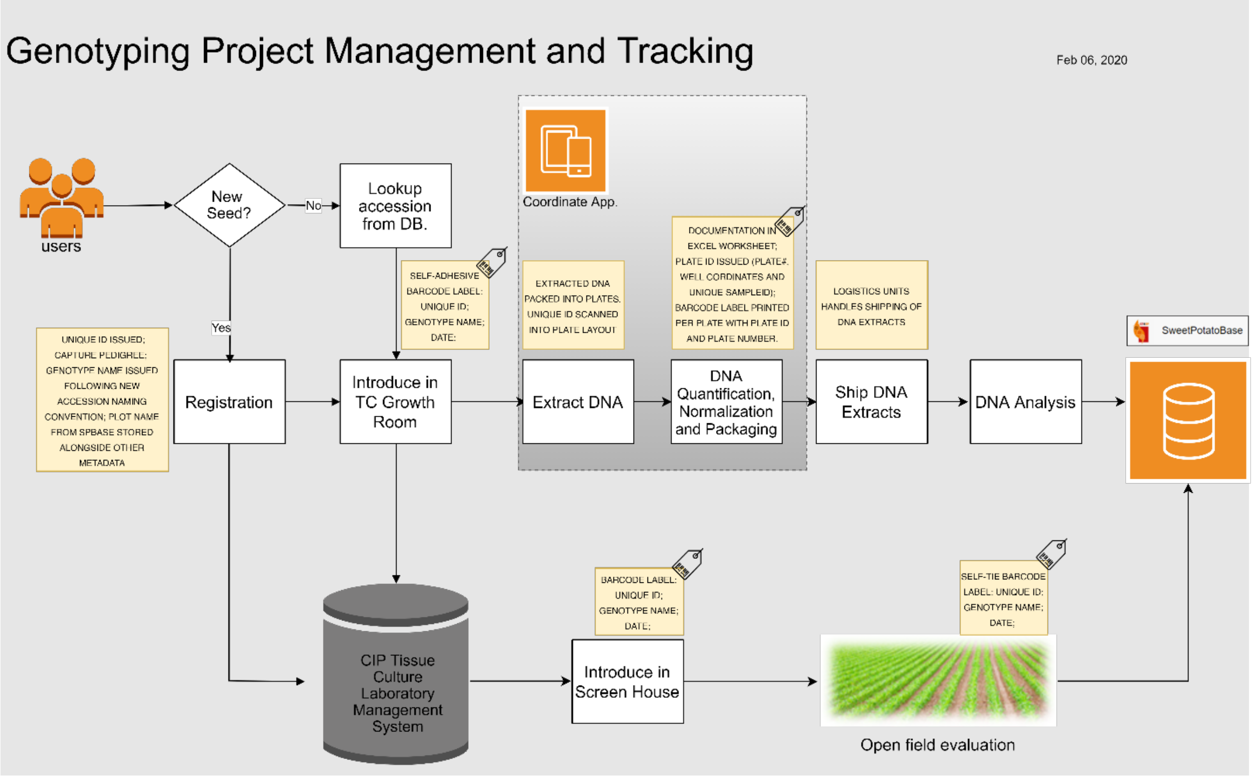
5.3.9. Z-Perform 1000T - 76171Plain PP White self-adhesive film labels special layout 3 Across 33 x 35 mm

5.3.10. Self-tie labels (polyplas material, 150 microns, 4 across)

5.3.11. Android Tablet

5.4. Procedures:

Below is a typical workflow.


Genotyping Project Management and Tracking
Feb 06, 2020
users
New Seed?
No
Yes
Lookup accession from DB.
Coordinate App.
SELF-ADHESIVE BARCODE LABEL: UNIQUE ID; GENOTYPE NAME; DATE:
EXTRACTED DNA PACKED INTO PLATES. UNIQUE ID SCANNED INTO PLATE LAYOUT
DOCUMENTATION IN EXCEL WORKSHEET; PLATE ID ISSUED (PLATE#. WELL CORDINATES AND UNIQUE SAMPLEID); BARCODE LABEL PRINTED PER PLATE WITH PLATE ID AND PLATE NUMBER.
LOGISTICS UNITS HANDLES SHIPPING OF DNA EXTRACTS
SweetPotatoBase
UNIQUE ID ISSUED; CAPTURE PEDIGREE; GENOTYPE NAME ISSUED FOLLOWING NEW ACCESSION NAMING CONVENTION; PLOT NAME FROM SPBASE STORED ALONGSIDE OTHER METADATA
Registration
Introduce in TC Growth Room
Extract DNA
DNA Quantification, Normalization and Packaging
Ship DNA Extracts
DNA Analysis
CIP Tissue Culture Laboratory Management System
BARCODE LABEL: UNIQUE ID; GENOTYPE NAME; DATE;
Introduce in Screen House
SELF-TIE BARCODE LABEL: UNIQUE ID: GENOTYPE NAME; DATE;
Open field evaluation

5.4.1. Introduce new seed

The typical steps when introducing a new accession are:

5.4.1.1. Receive seed from breeding program

5.4.1.2. Register new seed into CIP Tissue Culture Laboratory Management System (CIPTCL)

5.4.1.3. Upon registration: Capture parental names, unique ID automatically issued per seed, new accession name given to seed based on new accession naming conventions. See appendix A.

5.4.2. Introduce in tissue culture

5.4.2.1. Introduce new accession into tissue culture and print a barcode label with unique ID, accession name, pedigree and printing date.

5.4.2.2. Introduced material stored in growth rooms.

5.4.3. Extract DNA

5.4.3.1. DNA extracted when optimal growth has been achieved. Extracted DNA stored in DNA plates.

5.4.3.2. During DNA extraction: Scan barcode on test tube using Coordinate app. This reads unique ID into the corresponding well on the DNA plate layout.

5.4.3.3. Label DNA plates. Plate ID generated by a combining the Plate Number with all well coordinates and unique sample IDs in the plate.

5.4.4. Ship DNA

5.4.4.1. Barcode label with Plate ID and Plate name printed and attached to all plates before shipping.

5.4.4.2. Logistics unit ships DNA to North Carolina State University (NCSU), USA

5.4.5. Upload analysis results to Sweetpotatobase

5.4.5.1. Germinated seed also evaluated in open field trials and data uploaded to Sweetpotatobase

5.4.5.2. DNA analysis results uploaded to corresponding accessions / plot names in Sweetpotatobase.

References

Courtney, C., & Neilsen, M. (2019, September). Intercross: a Breeding Application for High-throughput Phenotyping. In *Proceedings of 32nd International Conference on* (Vol. 63, pp. 72-79).

Grüneberg, W. J., Eyzaguirre, R., Diaz, F., Boeck, B. D., Espinoza, J., Swanckaert, J., ... & Ndingo-Chipungu, F. P. (2019). Procedures for the evaluation of sweetpotato trials. Manual. International Potato Center: Lima, Peru. https://cgspace.cgiar.org/handle/10568/105875

Rife, T. W., & Poland, J. A. (2014). Field book: an open-source application for field data collection on android. *Crop Science*, *54*(4), 1624-1627.

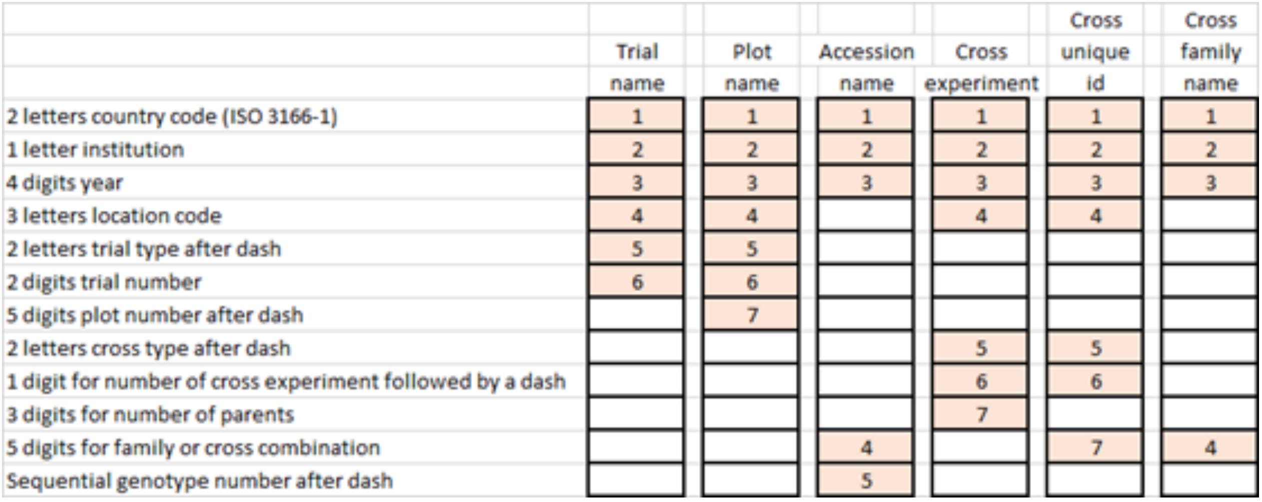
Annex A. Sweetpotato naming conventions

Sweetpotato naming conventions

Sweetpotato breeding community

1. Name structure

Trials, plots, accessions, cross experiments, crossings, and families need a standard naming structure. These names are constructed by concatenating the letters, numbers and symbols listed in the first column of the following table in the order described by the numbers in the corresponding column for each item.



					Cross	Cross
	Trial	Plot	Accession	Cross	unique	family
	name	name	name	experiment	id	name
2 letters country code (ISO 3166-1)	1	1	1	1	1	1
1 letter institution	2	2	2	2	2	2
4 digits year	3	3	3	3	3	3
3 letters location code	4	4		4	4	
2 letters trial type after dash	5	5				
2 digits trial number	6	6				
5 digits plot number after dash		7				
2 letters cross type after dash				5	5	
1 digit for number of cross experiment followed by a dash				6	6	
3 digits for number of parents				7		
5 digits for family or cross combination			4		7	4
Sequential genotype number after dash			5			

2. Abbreviations

2.1. Country code

ISO 3166-1 standard, link here.

2.2. Institution code

P for CIP, link here.

2.3. Location code

Each platform must work on a list of locations, link here.

2.4. Trial type

- SN: Seedling nursery.
- OT: Observational yield trial.

- PT: Preliminary yield trial.
- AT: Advanced yield trial.
- FT: on-Farm trial.
- CT: crossing trial.
- ST: Special trial (e.g. Storage, genetic gain, drought, bio assay)

To map these trial types into sweetpotatobase, use the following correspondences (sometimes there is more than one option in sweetpotatobase that matches with one of our trial types):

- SN corresponds to Seedling Nursery.
- OT corresponds to phenotyping_trial or Clonal Evaluation.
- PT corresponds to Preliminary Yield Trial.
- AT corresponds to Advanced Yield Trial, Uniform Yield Trial, or Variety Release Trial.
- CT corresponds to crossing_block_trial
- FT corresponds to On-Farm Trial.
- ST corresponds to All other options.

2.5. Cross experiment type

- OP: Open pollinated crosses from non-isolated field trials or multiplication blocks.
- PC: Polycross (open pollinated seed from parents in an isolated crossing block).
- BC: Biparental controlled crosses.

3. Examples

3.1. Trial name example

Ghana CIP trial, planted in 2019, location Fumesua, observational trial number 1: GHP2019FMS-OT01.

3.2. Plot name example

Ghana CIP trial, planted in 2019, location Fumesua, observational trial number 1, plot 1, 2, 3...: GHP2019FMS-OT01-00001, GHP2019FMS-OT01-00002, GHP2019FMS-OT01-00003...

3.3. Accession name example

Genotype germinated in Uganda by NaCCRI in 2020, germination family number 1 for NaCCRI in 2020, first clonal selection: UGN202000001-1. Note that the germination was done by NaCCRI (UGN country + institution code) in 2020, and it doesn’t matter if the crossing was done by another platform in another year.

3.4. Cross experiment example

Ghana CIP cross experiment started in 2019, location Fumesua, biparental controlled crossing block 1, 80 different parents: GHP2019FMS-BC1-080.

3.5. Cross unique id example

Ghana CIP cross experiment started in 2019, location Fumesua, biparental controlled crossing block 1, first cross: GHP2019FMS-BC1-00001. Note that a cross - with a unique cross unique id - can be defined on accession, plot or plant level for the parents, or a combination of those levels. Different crossing events in the same cross experiment with the same parents (on the chosen level) have the same cross unique id.

3.6. Cross family name example

Ghana CIP cross experiment started in 2019, first **cross family**: GHP201900001. Note that Ghana CIP is the **program of crossing** and 2019 is the **year of crossing**, even in case the germination is done in another program in another year. A unique cross family name corresponds to a genetic family for a given program and year of crossing, and can correspond to multiple cross unique ids of a crossing experiment.

Annex B. Sweetpotato breeding data curation procedures

Sweetpotato data check

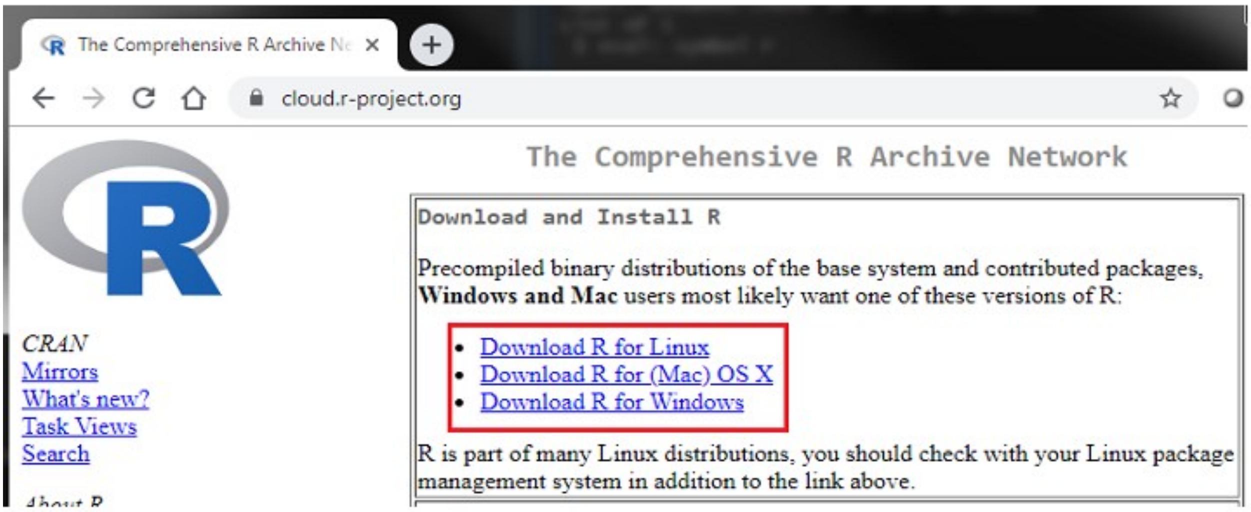
CIP Sweetpotato breeding community

27/03/2020

1. Install R and packages

1.1. Download R

Tools for data checking are implemented in the st4gi R package. R is a free software environment for statistical computing and graphics. To download R go to https://cloud.r-project.org and click on the suitable version for your operating system. Make sure to have the last version installed (there is at least a couple of new releases per year).



The Comprehensive R Archive Network

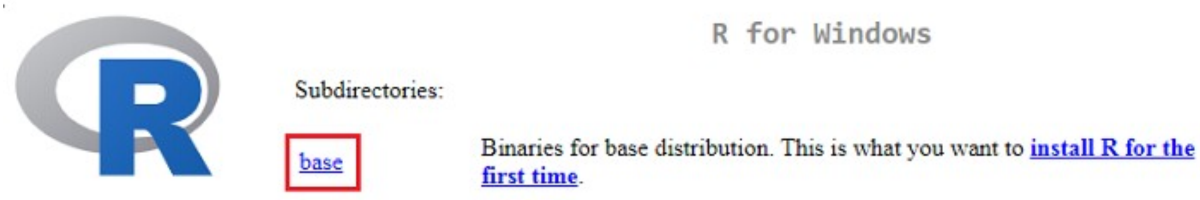
Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- Download R for Linux
- Download R for (Mac) OS X
- Download R for Windows

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

If in Windows, click on base

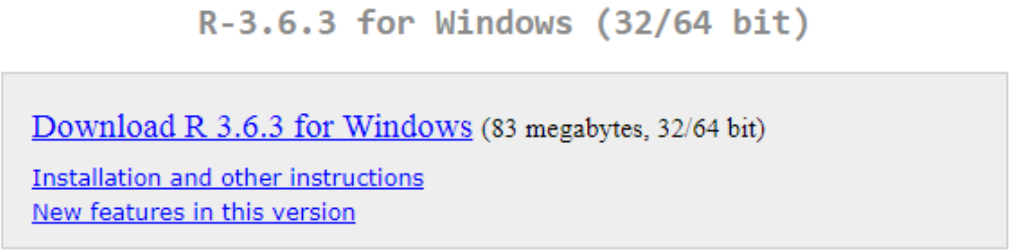


R for Windows

Subdirectories:

base Binaries for base distribution. This is what you want to **install R for the first time**.

and then on Download R 3.6.3 (last version as of today) for Windows



R-3.6.3 for Windows (32/64 bit)

Download R 3.6.3 for Windows (83 megabytes, 32/64 bit)

Installation and other instructions

New features in this version

1.2. Install R

Installing R is straightforward. Just double click on the installer file and accept all the default options.

1.3. Install the st4gi package

R is command driven software. To install the st4gi package open R and type the following commands in the R console. Make also sure to have an updated version of the package (update the package at least each 6 months).

```
install.packages("devtools")
devtools::install_github("reyzaguirre/st4gi")
```

2. Check data

2.1. Load the fieldbook data

One of the easiest ways of loading data into R is by reading csv files. Save your fieldbook file as a csv file and use the read.csv command to load the data. In the following example the file PEP2019HUA-ST01.csv is loaded.

```
mydata <- read.csv("D:/spdata/PEP2019HUA-ST01.csv")
```

Have in mind the following considerations:

- R is an object-oriented language. For any object that you want to load in memory you need to assign a name. In this example, I assign the name mydata to the data object that is loaded using the read.csv command. The symbol <- is used to do the assignment.
- You need to specify the full path to your file. In this example the file PEP2019HUA-ST01.csv is in the folder spdata located in drive D.
- Note that R uses / instead of \ to separate folders and subfolders.

In addition, consider that read.csv starts reading data from the first row of the file, and that this first row is for the labels of the columns. If you have any number of rows with other information at the beginning, then you need to skip those rows. For instance, if the fieldbook starts with the row of labels in the second row, you skip the first row by typing skip = 1.

```
mydata <- read.csv("D:/spdata/PEP2019HUA-ST01.csv", skip = 1)
```

2.2. Check frequencies

Most of the time (exceptions would be the p-reps design and designs with checks replicated several times like with the Westcott layout) we work with balanced data in complete replications, that is, all the genotypes are present once in each replication. A typical mistake is to incorrectly type the replication number for some plot, and therefore the data will look as unbalanced. You can use the check.freq command to verify this. The syntax for this command is check.freq(trait, genotype, environment, replication, data). If data comes from only one environment, then type NULL for environments. You don’t need to run this for all traits if you only want to check the frequency structure of the data set, if that is the case just pick one trait at random. In the following example the frequencies are checked using trait crw and since this data is for only one environment, NULL is typed for environments.

```
check.freq("crw", "cipno", NULL, "rep", mydata)
```

2.3. Check positions

With this command we check that there is only one plot in a given row and column position on the field for each block or replication. Syntax is of the form check.pos(row, column, replication, data). Below you can see an example.

```
check.pos("row", "col", "rep", mydata)
```

2.4. Check names

The command check.names.sp checks if the fieldbook column names correspond with the short labels defined in the Sweetpotato protocol. These short names are synonyms for the long names defined in the Crop Ontology and used in Sweetpotato Base. It returns a new data frame with all the names in lowercase and a warning with the list of names that are not in the protocol. Type ?check.names.sp to go to the help page and see the list of labels. Below you can see example.

```
mydata <- check.names.sp(mydata)
```

It is not necessary to run this function since all the other functions for checking the data will call this function as part of their routine.

2.5. Check consistency

2.5.1. Rules

There are several data consistency checks in the command check.data. The following rules are considered:

1. nops cannot be 0 or missing value.
2. nops > nope > noph > nopr.
3. If nope = 0 then should not be data for pre harvest evaluations (virus, Alternaria, vine vigor) and vice versa.
4. If noph = 0 then there should not be yield data different from 0 for vines and roots, and if there is a vine value then noph should not be 0.
5. If vine weigth is 0, then should not be data for dry matter evaluation of vines.
6. Dry weight must be < fresh weight for dry matter determination (for vines and roots).
7. If nopr = 0 then there should not be yield data for roots and vice versa.
8. If there are no roots (nocr or nonc) then there should not be root weight (crw or ncrw) and vice versa.
9. If there are no roots (nocr and nonc) then should not be data for evaluations on roots like fcol, scol, rs, rf, damr, rspr, wed, dm, and quality traits (ca, fe, zn, sucr, malt, etc).
10. Check extreme values for all traits. This is based on number of interquartile ranges. By default, any number out of the range [Q1 – 5IQR; Q3 + 5IQR] is detected as extreme, where Q1 is the first quartile, Q3 the third quartile, and IQR is the interquartile range. You can change this default with the argument f.
11. Out of range values, where the valid ranges are:
    - Integer 1 to 9 for quality traits measured with the 1-9 scale or 1 to 30 for flesh color measured with RHS color charts.
    - Integer non-negative values for discrete traits.
    - 0-100 for percentages.
    - Non-Negative values for quantitative traits.

12. Outliers detection based on residuals. It only works for rcbd design with one or several environments. Values with absolute residuals higher than 4 by default are detected. You can change this default with argument out.max.

2.5.2. Examples

An example using default values is shown below.

```
check.data(mydata)
```

With large data sets the output of this procedure can be very large, so a good idea could be to send the output to a file. You can do it using the command sink().

```
sink('outputfile.txt')
check.data(mydata) sink()
```

If there are numeric traits that are not defined in the check.names.sp function, you can add those traits for outliers detection using argument add. Below you can see an example where traits with names trait1 and trait2 are added.

```
check.data(mydata, add = c('trait1', 'trait2'))
```

2.5.3. Change defaults for outliers detection

By default, command check.data looks for outliers based on the quartiles method only, and with this method, by default, any value out of 5 times the IQR is detected. You can change this with the f argument. With very large data sets, this method can produce a lot of outliers, so it could be a good idea to change the default value to a more conservative one. In the example below I change the f value to 10.

```
check.data(mydata, f = 10)
```

To detect outliers based on residuals under some linear model, the out.mod argument must be set to rcbd or met (so far, the only two options). By default, it uses a threshold of 4 for the absolute value of the residuals but you can change that with the out.max argument. In the example below I check the mydata data that follows a RCBD with a value of 5 as residuals threshold.

```
check.data(mydata, out.mod = "rcbd", out.max = 5)
```

A value of 4 is quite popular for outlier's detection. Why? Because under the assumption of a normal distribution for the residuals, the probability of being out the [−4,+4] interval is very small (and maybe because the statistical tables for normal probabilities that were used in the past used to show values from -4 to 4). In the next table you can see some probabilities under normality:

Residuals threshold	Probability to be out of the interval	Number of values out of the interval
3	0.00269980	1 in 1482
4	0.00006334	1 in 63149
5	0.00000057	1 in 6977112

2.5.4. A note on outliers' detection

With any method, consider an outlier as a warning that something can be wrong. For instance, if you have a plot yield of 10 kilograms that is recorded by mistake as 100 kilograms, this value will surely appear as an outlier with any method. But there is always the chance that a value detected as an outlier is just a real extreme value. I would recommend the following rules:

1. If there are hard copies, check that the value detected as outlier has been correctly typed and correct accordingly.
2. If the value is clearly a mistake (e.g. one sweetpotato root of 5 kilograms) and there is no way to get the correct value, delete the value (change to missing value). But be very conservative with this way of action.
3. If the value is not clearly a mistake (perhaps something difficult to believe but possible), just leave the value as it is.

3. Curate data

3.1. Function setna

Function setna will try to identify all the impossible values and then will set them to missing value. The conditions to set a value to missing value are:

1. Continuous non-negative traits with negative values.
2. Continuous positive traits with non-positive values.
3. Percentage non-negative traits with values out of the [0, 100] interval.
4. Percentage positive traits with values out of the (0, 100] interval.
5. Discrete non-negative traits with negative and non-integer values.
6. Categorical traits with out of scale values.
7. Beta carotene values determined by RHS color charts with values different from the possible values in the RHS color chart.
8. Extreme clearly impossible low and high values.
9. If nope is 0 and there is some data for any trait, then nope is set to missing value.
10. If noph is 0 and there is some data for any non-pre-harvest trait, then noph is set to missing value.
11. If nopr is 0 and there is some data for any trait evaluated with roots, then nopr is set to missing value.
12. If noph is greater than 0 and nocr, nonc, crw, ncrw, and vw are all 0, then vw is set to missing value.
13. If nopr is greater than 0 and nocr, nonc, crw, and ncrw are all 0, then nonc and ncrw are both set to missing value.
14. If nocr is 0 and crw is greater than 0, then nocr is set to missing value.
15. If nocr is greater than 0 and crw is 0, then crw is set to missing value.
16. If nonc is 0 and ncrw is greater than 0, then nonc is set to missing value.
17. If nonc is greater than 0 and ncrw is 0, then ncrw is set to missing value.

When running this function, you will get a list of warnings for all the cells that have been modified. To use this function, you just type setna(mydata).

3.2. Function setzero

Missing values occur when:

1. All the plants died in a plot due to external reasons (not related with the quality of the genotype).
2. Because for some reason it was not possible to record a datum in a given plot.

3. There were no roots or vines to evaluate a trait that needs roots or vines. For example, roots are needed to evaluate dry matter, so if there are no roots, dry matter is a missing value, never a zero.

A frequent problem is confusion between missing value and zero, with a tendency to put missing values instead of zeros more frequently that the other way around for traits observed at harvest (vw, nocr, nonc, crw, and ncrw); the reason is that it is always easier just to leave a cell empty instead of typing a zero. To try to fix this issue there is function setzero.

The setzero function will change missing values by zeros for noph and nopr and traits vw, nocr, nonc, crw, and ncrw according to the following rules:

1. If noph is 0, then all traits are set to 0.
2. If all traits are 0, then noph is set to 0.
3. If nopr is 0, then all traits with exception of vw are set to 0.
4. If all traits with exception of vw are 0, then nopr is set to 0.
5. If nocr is 0, then crw is set to 0.
6. If crw is 0, then nocr is set to 0.
7. If nonc is 0, then ncrw is set to 0.
8. If ncrw is 0, then nonc is set to 0.

As with function setna, this function will also return a set of warnings for all the cells that have been modified. To use this function, you just type setna(mydata). To avoid incongruencies or circular issues, run function setzero always after running function setna or, even better, just run function clean.data.

3.3. Function clean.data

Function clean.data runs functions setna and setzero in that order. To use this function, you just type clean.data(mydata).

3.4. Function cdt

Another function that can help you curate your data is function cdt. This function computes all the traits that can be computed from the observed traits (like rytha from crw and ncrw). To use this function, you just type cdt(mydata), but if you want to get the traits defined in tons per hectare, you need to specify a method and a value to extrapolate from the plot to the hectare. There are two ways to do this:

- Using the number of plants (method = "np"): Consider a plot with 0.3 m between plants and 1 m between ridges. With these distances you can allocate 10000 / (1 * 0.3) = 33333 plants in a full hectare. Then, in the cdt function you must specify method "np" with a value of 33333. Example: cdt(mydata, method = "np", value = 33333).
- Using the plot size (method = "ps"). Consider a plot with 10 plants and 0.3 m between plants and 1 m between ridges. The plot size is 10 * 0.3 * 1 = 3 squared meters. Then, in the cdt function you must specify method "ps" with a value of 3. Example: cdt(mydata, method = 'ps', value = 3).

It is recommended running this function before checking the data with function check.data because there are several issues that can be detected with computed traits. Most typical examples are:

- Average root weight. Roots with very high weights (several kilograms) can appear when checking traits acrw, ancrw, and atrw that can remain undetected checking only number of roots and weight of roots per plot.

- A big number of roots per plant that can remain undetected just looking to number of roots per plot.
- High values for all totals (trw, tnr, biom) that can remain undetected just looking to their single components.
- Too high or low values for dry matter that can remain undetected just looking to the fresh and dry weights.

3.5. Proposed steps for data curation

1. Load the data. Start only with observed data (no calculated traits).
2. Check frequencies with function check.freq.
3. Check positions with function check.pos if there is row and column information.
4. If there are problems detected in steps 2 and 3, fix them in the data file, save the file, and load the data again.
5. Run the cdt function to compute all possible traits and run check.data.
6. Fix all the issues that can be fixed (mostly typos that you can verify with the hard copies of the data), save the file and load the data again.
7. Run the clean.data function to detect and delete all impossible values and set missing values to zero when they match the rules. Be very careful with this operation and have a look to the warnings (type warnings() to see them all) for all the cells that have been modified. We don't want to spoil our data.
8. When the data is ready, compute all traits again with cdt and save a copy with a new name.

Below you can see an example of these steps with R code:

```
# Load the st4gi package library(st4gi)

# Load the data mydata <- read.csv("D:/spdata/PEP2019HUA-
ST01.csv")

# Check frequencies with any trait check.freq("crw", "cipno",
NULL, "rep", mydata)

# Check positions check.pos("row", "col", "rep",
mydata)

# Fix all the detected problems in the data file and load the data again if necessary mydata <-
read.csv("D:/spdata/PEP2019HUA-ST01.csv")

# Compute all possible traits mydata <- cdt(mydata, method =
"np", value = 33333)

# Check data check.data(mydata)

# Fix all the detected problems in the data file and load the data again mydata <-
read.csv("D:/spdata/PEP2019HUA-ST01.csv")

# Run clean.data mydata <-
clean.data(mydata)

# Compute all possible traits mydata <- cdt(mydata, method =
"np", value = 33333)

# Save the output with a new name

write.csv(mydata, "newname.csv")
```

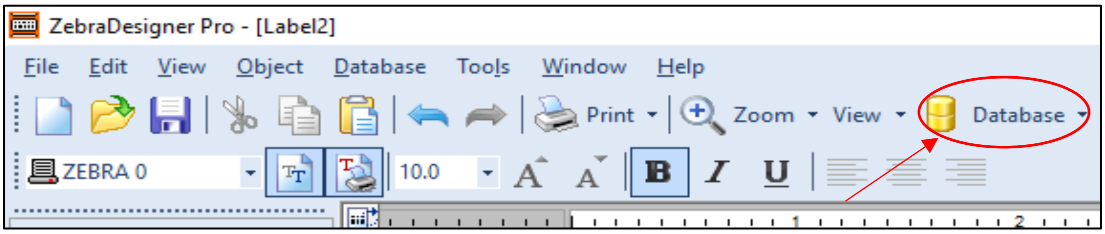
Annex C. How to Print Labels using the ZM400 Printer

CIP is using the ZM series of bar code printers by Zebra for the printing of labels for germplasm being maintained in tissue culture form, in screen houses or in the open field. All bar code labels are printed using a software application called ***Zebra Designer Pro 2***. This guide will explain how to print barcode labels using this application. ***Zebra Designer Pro 2*** is a commercial software application that can be purchased online on the Zebra online shop.

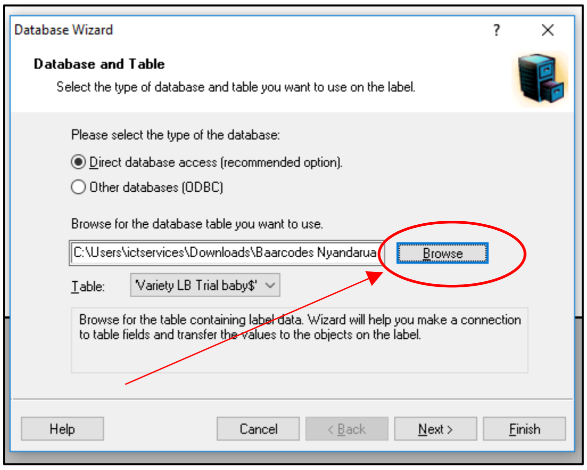
1. Launch ***Zebra Designer Pro 2*** Software. The software can be launched directly by opening any of these label template files provided here: https://cgiar-my.sharepoint.com/:f:/g/personal/l_wanjohi_cgiar_org/Eor8DJNZeDRAnpmF689LtIcBz3Vom84Bcilcm4qe1QC4wA?e=wtbRy7. There are different template files based on your label size. Most of our field labels are the four-column type. Using a label template file saves you time since you do not have to define the label dimensions afresh.

   Zebra Designer Pro 2 allows you to print labels from a database file. The information to be printed on the labels should be organized into separate columns for example the genotype name, the plot name, site name, date of planting, etc. The database file should be in MS Excel, CSV or text formats.

2. Create a connection to your database file by clicking on the database icon



3. Browse to your database file as shown below and then click next



Database and Table

Select the type of database and table you want to use on the label.

Please select the type of the database:

● Direct database access (recommended option).

○ Other databases (ODBC)

Browse for the database table you want to use.

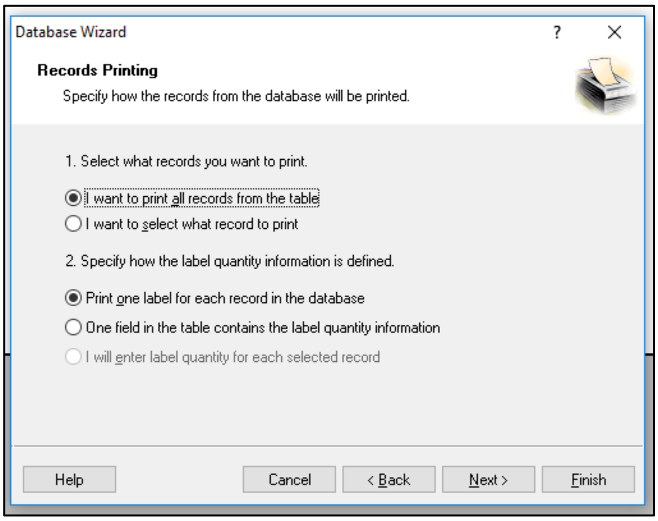
C:\Users\ictservices\Downloads\Baarcodes Nyandarua

Table: 'Variety LB Trial baby$'

Browse for the table containing label data. Wizard will help you make a connection to table fields and transfer the values to the objects on the label.

4. Set your preferred printing options on the next dialogue box. We recommend choosing the second option "*I want to select what record to print*" when you are designing your label. This allows you to print a few records to test print. You can change this setting back to "*I want to print all records*" when you are ready to print your final labels. Choosing this option will print labels for all the records in your database at once.

   On this dialogue box, it is also possible to specify printing of more than label per record in the database. When you are done, click Finish.



Database Wizard

Records Printing

Specify how the records from the database will be printed.

1. Select what records you want to print.

● I want to print all records from the table

○ I want to select what record to print

2. Specify how the label quantity information is defined.

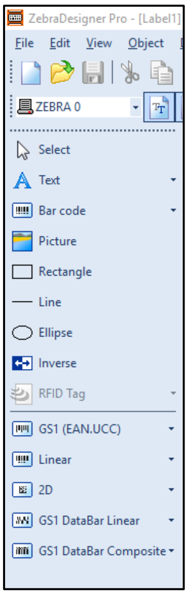
● Print one label for each record in the database

○ One field in the table contains the label quantity information

○ I will enter label quantity for each selected record

5. You are now ready to add the fields from your database to the label. These are added as text fields or bar code fields. Select the appropriate field from the available options. Both text field and bar code fields will let you choose the appropriate column to link into the database, as well as the option for having fixed text where necessary.

   The bar code field should be linked to the column that uniquely identifies the material being labelled. For breeding trials, this should be the plot name. We recommend using a 2D barcode type e.g. QR.



6. When you are done adding the field you want printed to your label, you can print preview or test print to see how your labels will look like. Print your labels when you are satisfied with the design.

Annex D. List of primary traits

Recording time	trait variable name in SPBase	synonym
At planting	Number of plants planted per NET plot|CO_331:0000678	nops
1 month after planting	Number of plants established per NET plot|CO_331:0000192	nope
1 month before harvest	Vine vigor estimating 1-9|CO_331:0000197	vv
At harvest	Number of plants harvested per NET plot|CO_331:0000679	noph
	Number of commercial storage roots per NET plot|CO_331:0000214	nocr
	Number of non-commercial storage roots per NET plot|CO_331:0000217	nonc
	Weight of commercial storage roots per NET plot in kg|CO_331:0000220	crw
	Weight of non-commercial storage roots per NET plot in kg|CO_331:0000223	ncrw
	Weight of vines per NET plot in kg|CO_331:0000227	vw
	Storage root predominant Flesh color estimating 1-9|CO_331:0000178	fcol
	Storage root skin predominant color estimating 1-9 CIP|CO_331:0000175	scol
	Storage root size estimating 1-9|CO_331:0000184	rs
	Storage root appearance estimating 1-9 CIP|CO_331:0000202	rf
	Storage root damage estimating 1-9|CO_331:0000206	damr

WWW.CIPOTATO.ORG

CIP is a research-for-development organization with a focus on potato, sweetpotato and Andean roots and tubers. It delivers innovative science-based solutions to enhance access to affordable nutritious food, foster inclusive sustainable business and employment growth, and drive the climate resilience of root and tuber agri-food systems. Headquartered in Lima, Peru, CIP has a research presence in more than 20 countries in Africa, Asia and Latin America.
www.cipotato.org

CIP is a CGIAR research center
CGIAR is a global research partnership for a food-secure future. Its science is carried out by 15 research centers in close collaboration with hundreds of partners across the globe.
www.cgiar.org

For more information, please contact CIP Headquarters. Av. La Molina 1895, La Molina. Apartado 1558, Lima 12, Peru.

☎ *+51 1 3496017* ✉ *cip-cpad@cgiar.org* *www.cipotato.org* | *@cipotato* *@Cipotato* *@cip_cipotato*