COMPUTATIONAL TOOLS FOR THE ANALYSIS OF BIOLOGICAL NETWORKS IN PLANTS

A Thesis
Presented to
The Academic Faculty

by

Abhiram Das

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Biology

Georgia Institute of Technology
December 2015

COMPUTATIONAL TOOLS FOR THE ANALYSIS OF BIOLOGICAL NETWORKS IN PLANTS

Approved by:

Dr. Joshua S. Weitz, Advisor
School of Biology
Georgia Institute of Technology

Dr. I. King Jordan
School of Biology
Georgia Institute of Technology

Dr. Shamkant Navathe
School of Computer Science
Georgia Institute of Technology

Dr. Daniel I. Goldman
School of Physics
Georgia Institute of Technology

Dr. Charles Price
School of Plant Biology
University of Western Australia

Date Approved: August 11, 2015

Dedicated to my parents
Sabitri Das (late) and Balaram Das
For their love and courage

ACKNOWLEDGEMENTS

I am deeply grateful for the guidance and support of my advisor Prof. Joshua S. Weitz of Georgia Institute of Technology. Throughout my time at Georgia Tech, his insightful comments and prompt advice made my graduate-student life a fascinating and enriching experience. I thank him for his support, for sharing his endless supply of ideas, and for being my philosopher and guide. I could not have hoped for a better advisor. I am grateful to my committee, Prof. I. King Jordan, Prof. Shamkant Navathe, Prof. Daniel I. Goldman and Dr. Charles Price, for their valuable time, advice, expertise and good-natured support to better my work.

I would like to thank the following professors in the School of Biology, Prof. Jung Choi, Prof. Michael Goodisman and Dr. David Garton for their advice and support.

Many thanks to my family for all the love, support and courage they have given. I am especially thankful to my brothers, Ramani Ranjan Das and Saroj Ranjan Das, for their continued courage and support. I am proud of you as my family. Thanks to my friends, Sushanta Mohanty, Sagarika Barik, Sumeet Batra, Dr. Puspendu Deo, Dr. Jyoti R. Mohanty, Madhubrata Mohanty, Rajesh Dash, Morgan Dash and many others for their help and support. Particularly for being there with me from the beginning, when I was preparing to leave a job and enter into the graduate program, till the end of the graduate-student life. My sincerest thanks to Prof. Satchidananda Panda, Dr. M. Nalini and Burton Strauss III for writing recommendation letters for my M.S. graduate program applications.

The research presented in this thesis is the result of joint research with colleagues. Here I wish to thank each of these colleagues for their support. Especially I would like to thank, Dr. Charles Price for his work and support in Cleared Leaves Image Database project, and Dr. Alexander Bucksch for his work and support in Digital Imaging of Root Traits project. Many thanks to Troy Hilley and Wesley Emeneker, for their prompt support in setting up computational environment for all my work at Georgia Tech. I would like to thank Nirav Merchant, Andy Edmonds and others at iPlant Collaborative for their help and support.

I am extremely grateful for the financial support provided by the following funding agencies that awarded grants to Prof. Joshua S. Weitz: Burroughs Welcome Fund; Institute for Data and High Performance Computing, Center for Data Analytics, College of Science, Georgia Tech; NSF and the iPlant Collaborative.

Finally, I would like to thank the former and the current members of Weitz Group, who shared many good times with me during my stay at Georgia Tech and provided feedback and comments on all of my work: Dr. Yuriy Mileyko, Dr. Olga Symonova, Dr. Tae Lee, Dr. Lauren Childs, Dr. Michael Cortez, Dr. Andrey Kislyuk, Dr. Richard In-Ho Joh, Dr. Gabriel Mitchell, Dr. Cesar Flores, Kristen Knipe, Shimantika Sharma, Dr. Hayriye Gulbudak, Dr. Ceyhun Eksin, Dr. Joey Leung, Adrian Lawsin, Charles Wigington, Luis Jover and Bradford Taylor.

TABLE OF CONTENTS

Page

CHAPTER

LIST OF TABLES

LIST OF FIGURES

Page

SUMMARY

This thesis presents research associated to phenotyping of plants by applying informatics techniques which includes databases, web technologies, image processing and feature measurements of 2D and 3D images. The thesis presents two enabling bioinformatics tools related by a shared set of research objectives and distinct by the nature of their applications. The first project called ClearedLeavesDB, is a common platform for plant biologists to share data and metadata about cleared leaf images. This project resulted in an online interactive database of cleared leaf images. The second project called **D**igital **I**maging of **R**oot **T**raits (DIRT), is an application to store, manage, share and process root images as well as analyze root image traits with respect to different experiments. This application is deployed on iPlant's cyber-infrastructure and currently supports management of 2D root images and high-throughput processing and structural descriptor/trait estimation from root images. The application enables storage, management and sharing heterogeneous image data and metadata including dynamic environmental and descriptor data. In the final part of the thesis, I describe ongoing challenges in developing new methods to measure global and local descriptors from reconstructed 3D root images.

CHAPTER 1

INTRODUCTION

1.1 Background

The world is facing unprecedented agricultural challenges, as developing nations are on the verge of facing a major crisis in food security. Approximately 1 billion people are undernourished and the world's human population is expected to exceed 10 billion in the next decade [3]. Although many factors influence food insecurity in developing nations, low crop yield is a primary cause [4, 5]. Rich nations sustain productivity and crop yield by extensive use of fertilizers at the cost of significant environmental degradation [6]. Low crop yield and environmental stress to plants is expected to exacerbate in the next decade due to climate change [7]. Therefore there is a dire need for development of agricultural systems and crop plants that have high productivity with minimum resource utility.

Crop productivity depends on balanced growth of both roots and shoots systems and there is functional interdependence between the two. Shoots bearing leaves are the primary site of production in both land and crop plants. Leaves vary in shape, composition, color, as well as the structure of their venation networks. Carbon, nutrients, and water are transported through xylem and phloem contained within the vascular bundles that constitute veins. In addition, veins confer material strength to the leaf due to the higher stiffness of veins relative to other leaf tissue. The analysis of leaf venation networks is of increasing interest due to multiple hypotheses linking venation networks to changes in leaf physiological rates [8] and to the diversification of early land plants [9, 10]. For quantitative and qualitative analysis of vein structure, leaves are cleared [11] and digitized. Study of leaves over multiple decades have resulted in many independent collections of cleared leaf images and specimens held by individual researchers and institutions like the Smithsonian Institution, the Yale Peabody Museum.

Unlike shoots, roots grow in a complex and heterogeneous medium. They also exhibit a very wide range of association with biotic and abiotic factors of the soil system. To understand the diverse forms of root system and their functions it is important to study their morphological and anatomical features. Quantitative analysis of root systems is difficult because roots are underground, embedded in heterogeneous and opaque soil. The growth of root systems in soil is affected by a wide range of soil properties and the interface between the root and soil is complex and often an ill-defined boundary [12]. Water and nutrients required by plants are present in the soil and are heterogeneously distributed and/or are subject to local depletion [13]. In such circumstances the development and growth of root system may become highly asymmetric and the spatial arrangement of the root system will substantially determine the ability of a plant to secure those resources [14]. Such ideas have been investigated in a series of experiments and models using common bean [15, 16] and Arabidopsis [17].

Primary constraints facing the crop plants are water and nutrient availability. We need crop varieties that has optimum yield in low water and nutrient conditions. Roots have the unexplored potential for development of crop varieties with greater yield and productivity under low nutrient and water condition, thereby hastening the “second green revolution” [5]. Though study of plant roots and its network has made great advancements during past decades, it still has challenges including: (i) identification of descriptors or traits from 2D and 3D image data [18], (ii) visualization of structural properties of large and complex networks along with heterogeneous metadata [19, 20], (iii) use of common analysis frameworks for different networks associated to different problem domains [21] (iv) distribution of the raw data and results of spatial network analysis with the community, (v) identification of relation between the dynamics of the root system at the local (measurement of descriptors or features of individual roots and branches) scale and global (measurement of traits of the whole root system) scale.

Current plant phenotying platforms or applications to characterize agriculturally relevant traits have been primarily developed for use in controlled laboratory and/or greenhouse conditions. The effort with respect to root architectural traits is therefore limited to young plants grown in specialized containers and growth media. In recent years efforts undertaken to understand the crop root system architecture (RSA), has led to a number of imaging solutions [22-25]. These solutions have been primarily used to assess root structure at early seedling stage or 10-15 days after germination. Most of the laboratory phenotyping platforms are not imaging mature root systems. In fact, studying root systems under field conditions possess new challenges and opportunities [26]. To overcome some of the challenges, a field protocol called "Shovelomics" [27] has been proposed and used to study maize root architecture under field conditions. But it has its own limitations with respect to manual measurements and throughput. Shovelomics is a manual scoring system where ten root architectural traits are measured by use of protractors and calipers from mature root crowns excavated by removing a soil cylinder of 40 cm diameter and 25 cm depth with the plant base.

Hence, to mitigate these limitations and to complement “Shovelomics”, we developed an online application called **D**igital Imaging of **R**oot **T**raits (DIRT) to store, manage, share root images and to compute root descriptors by a pipeline [1] running on high-throughput grid-computing environment. As part of this application I have developed custom modules and scripts to integrate image processing pipeline on the grid environment. These modules enable users to process thousands of images through the pipeline with custom parameters and view processed results associated to the raw images. Thus, making high-throughput computational infrastructure projects and resources available to the users with no technical expertise.

Despite innovations, there are a number of challenges remaining. In this thesis, I describe steps to overcome the barrier limiting the use of computational resources and technologies in the service of the plant sciences. I do so through a series of projects. First, I describe ClearedLeavesDB, an online database with social network mechanism for a community of researchers to contribute, access and share cleared leaf images that also leverage resources of curated collections. Second, I describe DIRT, a high-throughput computational platform for data access, exchange and sharing, and field root phenotyping.

Both of these projects represent tools in the service of discovery in the plant sciences. They are already being utilized by the global community of plant researchers to understand variation in plant network structure and to link genotype to phenotype. Yet, there are many features of plant networks that could represent future targets for linking genotype to phenotype. Towards that aim, I describe my efforts to quantify a novel feature - the angle of entry of plant root systems as inferred from 3D reconstructed images of rice.

1.2 An online database of cleared plant leaf images

Distinct organs of a plant show different classifying features such as network structures, colors, and chemical concentrations. Analyzing and quantifying such features enables the functional interpretation of the plant organ. An example of such an organ is the leaf, which is often maintained as curated collections of images. These collections are not always easily accessible to researchers. However, there is a significant value in the digital analysis of collections, if made available to the larger scientific community.

Leaf vein networks are critical to both the structure and function of leaves. Moreover, a growing body of recent work has linked leaf vein network structure to the physiology, ecology and evolution of land plants. In the process, multiple institutions and individual researchers have assembled collections of cleared leaf specimens in which vascular bundles (veins) are rendered visible. In an effort to facilitate analysis and digitally preserve these specimens, high-resolution images are usually created either of entire leaves or of magnified leaf subsections. In a few cases, collections of digital images of cleared leaves are available for use online. However, these collections do not share a common platform nor is there a means to digitally archive cleared leaf images held by individual researchers (in addition to those held by institutions). Hence, there is a growing need for a digital archive that enables online viewing, sharing and disseminating of cleared leaf image collections held by both institutions and individual researchers.

We developed ClearedLeavesDB, a unique database of cleared leaf images that combines interactions between users and data via an intuitive web interface. The web interface allows storage of large collections and integrates with leaf image analysis applications via an open application programming interface (API). The open API allows upload of processed images and its output to the database, hence enabling distribution and documentation of analyzed data within the community. The initial database is seeded with nearly 19,000 cleared leaf images representing over 40 GB of image data. Extensible storage and growth of the database is enabled by using the data storage resources of the iPlant Discovery Environment. ClearedLeavesDB can be accessed at http://clearedleavesdb.org. The manuscript describing ClearedLeavesDB is available as "Das et al.,ClearedLeavesDB: an online database of cleared plant leaf images., Plant Methods 2014, 10:8".

1.3 DIRT - A high-throughput crop root phenomics platform

To meet the projected food demand of the global population, crop production must double by year 2050 [28]. Global climate change, unpredictable weather and crops dependence on rain water in most part of the world, might dawn 2050 little early on all of us. As for crop productivity, root system architecture (RSA) plays an important role under stress conditions [14] and root phenotyping is a major focus of the plant science community. Even though there is extensive genetic variation in root architecture, root traits have not been traditionally selected by plant breeders for improved yield or nutrient uptake efficiency, because of the difficulty in measuring and evaluating its quantitative mode of inheritance [29]. More than ever before, there is an urgent need to increase breeding efficiency. Next-generation inexpensive sequencing technologies, fast and high-throughput genomic data have paved the way for phenotyping by developing large mapping populations and diversity panel of thousands of recombinant inbred lines [30]. Therefore, advancement in phenotyping is essential to leverage the developments in plant breeding, thereby insuring future food demand [31]. To address the phenotypic challenges, many projects have been launched under different consortia such as the German Plant Phenotyping Network (http://www.dppn.de/), the European Plant Phenotyping Network (http://www.plant-phenotyping-network.eu/) and the International Plant Phenotyping Network (http://www. plantphenomics.com/).

All these projects have produced various automated and semi-automated high-throughput plant phenotyping platforms to study and examine plant function and performance under controlled conditions. These high-throughput phenotyping platforms have enabled quantitative trait loci (QTL) analysis and candidate gene identification by capturing detailed information of plant's life cycle in a controlled environment in a non-invasive manner. But generally, these results have not been translated to crop productivity in the field. It is very difficult to extrapolate the results of a controlled environment to the heterogeneous and uncontrolled field environment. In response, Shovelomics [27] has pioneered root phenotyping by enabling identification of some desirable root traits in the field. It has helped to shorten the path to incorporating field root traits in crop-breeding programs. Even though a high-throughput non-invasive technique to directly evaluate root system in the field does not exist yet [32], DIRT takes Shovelomics to a new level by integrating image processing pipeline [1], database, data storage and high-throughput computing under one platform and enabling its access over the internet via user-friendly graphical interface. DIRT not only overcomes the laborious, time consuming measurement processes of Shovelomics, but enables measurement of new traits that were not possible to estimate manually.

There are different software frameworks that can be used to extract and measure RSA traits from two-dimensional and three-dimensional digital images in various hierarchies of automation. A full list of available frameworks for root image analysis can be found at www.plant-image-analysis.org. These softwares range from manual (DART), semi-automatic (SmartRoot, RootReader) to automatic (GiaRoots, ARIA) frameworks. Some of them (ARIA) are even high-throughput, some (SmartRoot) are even integrated with a database to persist computed traits. But all of these frameworks needs to be installed on a researcher's (hence forth referred as user) workstation or server before use. This leads to a number of issues with compatibility and accessibility. In addition, none of these frameworks provides an integrated platform to associate root image with environmental and phenotypic data and share all these information with the community. DIRT addresses these phenotyping bottlenecks [33], by providing a single platform to meet the demand of data access, exchange and sharing, and field root phenotyping.

1.4 Quantifying gravitropism from 3D images of rice roots

Recent advances in phenotyping capabilities have increased attention on the study of RSA in rice and its relation to genetics. Previous studies [14, 34] have reported that root gravitropism is one of the important factors to determine root architecture, and it affects spatial distribution of root system in the soil and consequently the nutrient uptake efficiency. Root gravitropism is quantified via manual measurements of root tip angle [35]. Though much of the gravitropic mechanism have been understood with respect to molecular and physiological processes [36-40], considerable details are yet to be understood with regards to other important aspects such as dependence of gravitropic response to environmental conditions or developmental age. Earlier studies [16] have demonstrated that gravitropic response is highly plastic. Past investigations [41] have demonstrated that roots with high plasticity exhibit allometric relationships between different root orders. All previous studies have either manually measured the root initiation angle or used 2D images for automated measurement. As mentioned in a previous study [42], 3D information of root architecture can be used to enhance estimation of these traits; Here, I use 3D information of whole rice roots to measure and analyze these traits. As genetic factors responsible for gravitropism in rice have been identified [43], accurately quantifying these traits in different genotype may help identify responsible genes. My focus here is to quantify the root angle with respect to gravity in rice roots from 3D images and analyze it with respect to different genotypes.

CHAPTER 2

CLEAREDLEAVESDB: AN ONLINE DATABASE OF CLEARED PLANT LEAF IMAGES

This chapter is adopted from the publication Das A, Bucksch A, Price CA, Weitz JS; ClearedLeavesDB: an online database of cleared plant leaf images.; Plant Methods 2014, 10:8.

2.1 Background

The leaf organ is the site of the majority of primary production within land plants. Leaves vary in shape, composition, color, as well as the structure of their venation networks [44]. Here, we focus on efforts to describe and analyze the last of these characteristics: the venation network. A leaf venation network represents the interconnected set of veins present in a leaf. Carbon, nutrients, and water are transported through xylem and phloem contained within the vascular bundles that constitute veins [45]. In addition, veins confer material strength to the leaf due to the higher stiffness of veins relative to other leaf tissue [46, 47].

The analysis of leaf venation networks is of increasing interest due to multiple hypotheses linking venation networks to changes in leaf physiological rates [8, 48] and to the diversification of early land plants [9, 49, 50]. For example, increases in the density of leaf veins are hypothesized to increase physiological rates, such as net photosynthesis, because of the decreasing distance that water molecules must travel between the vein network, terminal sites of photosynthesis and stomates [10, 51, 52]. Similarly, the early adaptive radiation of land plants coincided with the appearance of increasingly reticulate (i.e. loopy) venation networks, as inferred from fossil leaves [53, 54]. Reticulate networks are known to confer added redundancy to leaves in the face of damage from herbivory or disease, providing a mechanism linking leaf venation structure to whole plant fitness [55].

In practice, the quantitative and qualitative analysis of venation networks requires the identification of veins as distinct from that of surrounding leaf tissue in areoles. Hence, a first step toward identifying a venation network is to “clear” a leaf [11, 56]. Multi-decadal efforts to collect cleared leaves have been undertaken by individual researchers and institutions. For example, in the USA there are now multiple, independent cleared leaf collections with thousands of specimens, e.g., held at the Smithsonian Institution [57], Yale Peabody Museum [58], New York Botanical Garden [59], and UC-Berkeley [60]. In each case, digital images have been taken of the specimens and in many instances, institutional websites offer free access to all or a portion of these curated collections. However, these sites do not necessarily provide functionality for individual researchers unaffiliated with these institutions to store, manage, search, and analyze images online – indeed, such functionality is outside of the scope of each individual collection. There are also other generalized repositories of images of plants and plant organs such as CalPhoto [61], Morphbank [62], USDA Plant database [63], Encyclopedia of Life records [64] and Bisque [65]. Bisque is notably positioned by its support for secure image storage, analysis and data management capabilities. It is also hosted on iPlant [66] environment giving users control over sharing of images and analysis results. While the above repositories are freely accessible, they are quite general in scope and does not target cleared leaf images. Moreover, individual collection sites are not linked, they do not share a common platform to share data, and they do not enable the integration of cleared leaf images with processed images and analysis. Hence there is a need for a complementary approach to enhance the utility of pre-existing digital-physical collections: a trans-institutional online database by which individual research groups and institutions can store, view and share their cleared leaf images with the community.

In order to address these issues, we have developed the Cleared Leaf Image Database (ClearedLeavesDB). ClearedLeavesDB enables researchers to contribute, access, and share cleared leaf images. The database houses existing curated collections, such as a major contribution from the Smithsonian Institution collection, and enables researchers to contribute small-scale curated collections for dissemination, viewing and re-use. The web interface of ClearedLeavesDB is built on open-source technologies, allowing users to: (i) create and manage leaf image collections; (ii) upload/download images and their metadata in single or batch mode; (iii) annotate images; (iv) search images; and (v) link analysis data with the original images. The database provides varying degree of access based on user privileges which enhances data quality and reliability. Broadening the accessibility of digital collections is of increasing importance across the biological sciences. As we describe below, we have developed ClearedLeavesDB with a similar intent: to accelerate basic science and public understanding of plant biology generally, and leaf veins specifically.

2.2 Construction and Methods

We describe a user-oriented view of the ClearedLeavesDB system, given that our focus in this paper is on the practical usage of the web application by and for plant biologists. Major sections of the application are described below, including how to create and manage image collections, search and browse images, manage and download marked lists of images, and user roles. For users interested in viewing and modifying the source code, we note that the application is developed using Drupal [67]; a content management system built upon Apache web server [68], MySQL [69] database server and PHP [70].

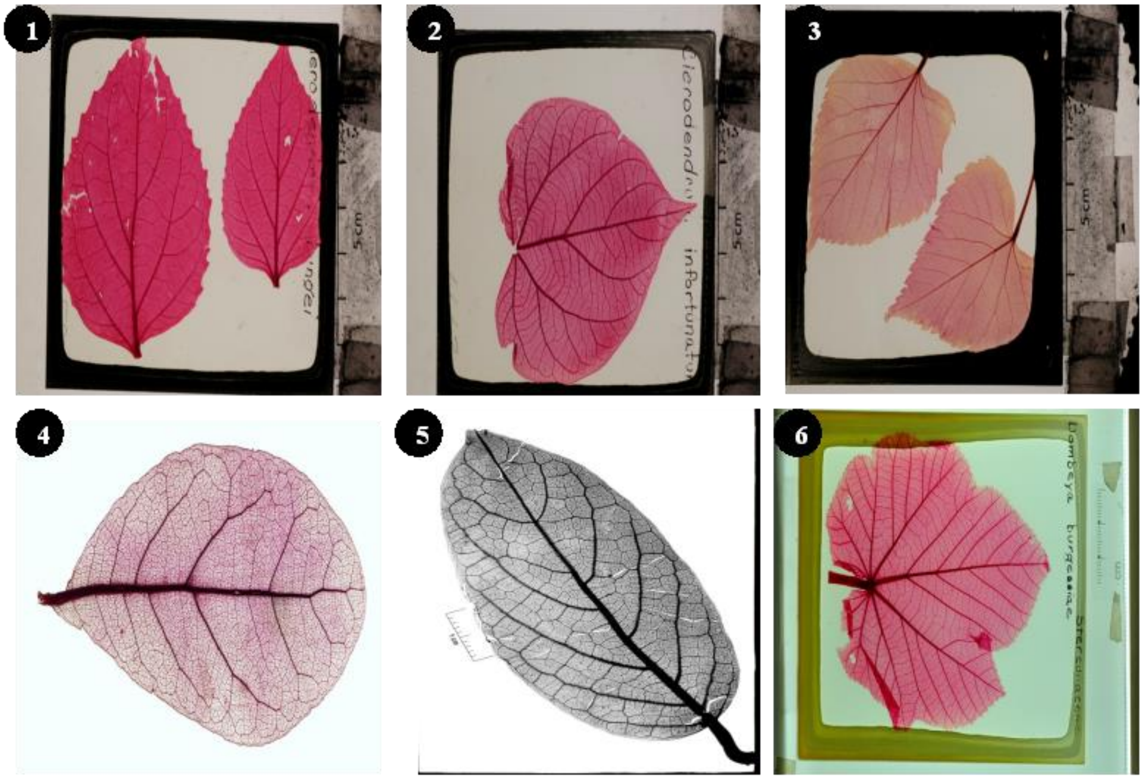


Figure 1: Representative images of cleared leaves in ClearedLeavesDB. 1. ***Clerodendrum bungei Steud.*** **2.** ***Clerodendrum infortunatum L.*** **3.** ***Tilia miqueliana Maxim.*** **4.** ***Quercus laurifolia*** **5.** ***Pseudocalyx africanus S. Moore*** **6.** ***Dombeya burgessiae***

Create and manage collections: Image collections are sets of uploaded images that share some common attributes (e.g. geographical location, experimental treatment, species etc.). These collections are curated and linked to the owner of the collection (but stored centrally). While creating a collection, the owner has to provide a name and brief description of the collection. The owner also has the option to mark a collection as private or public and the license under which the data is released. Public collections can be accessed by all users and visitors of the database whereas private collections can only be accessed by the owner and associated members. A personalized access module allows the owner to manage a collection membership for all registered users (see the user roles section below for details). Once a collection is created, images are uploaded via a simple user interface. Users with required privileges are permitted to annotate images. The collection owner can change all initial preferences at any time via the “Collections” menu. At the time of submission, ClearedLeavesDB includes nearly 19,000 images (see **Figure 1** for a sample of the images available) from over 144 countries (see **Figure 2** for the current geographical distributions of source specimens).

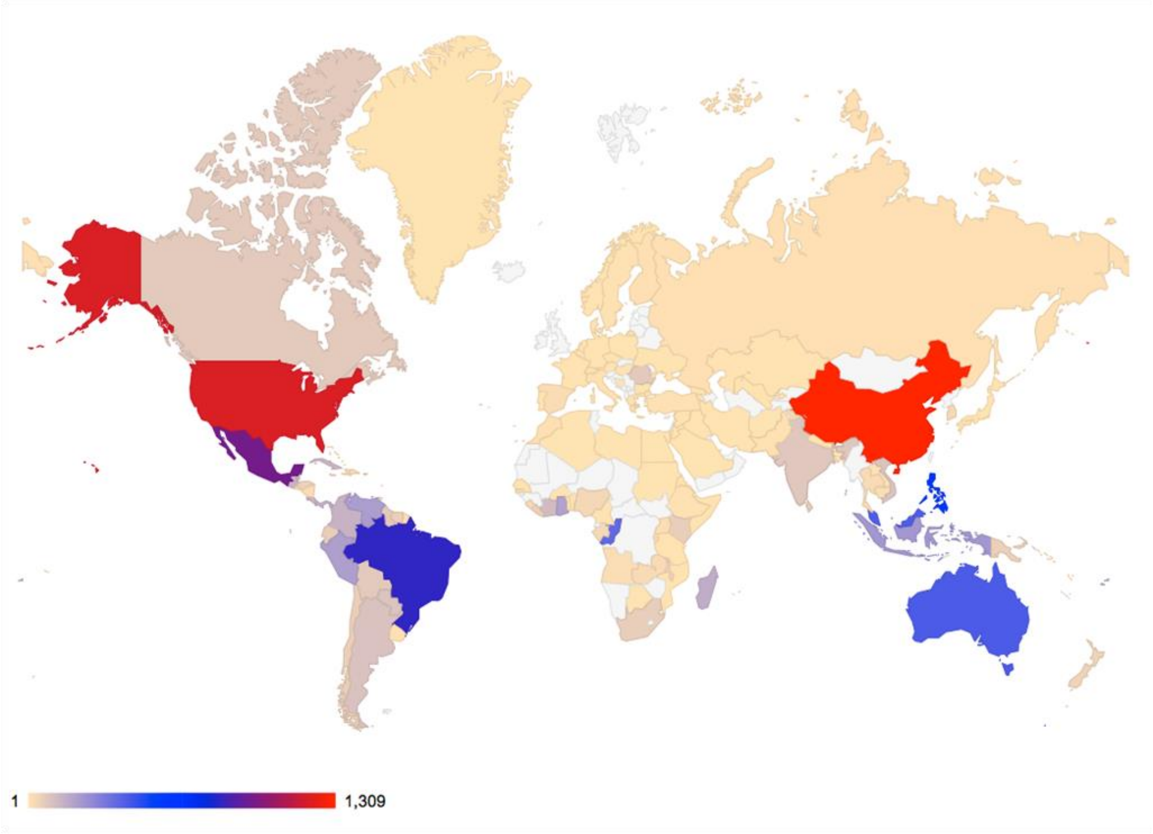
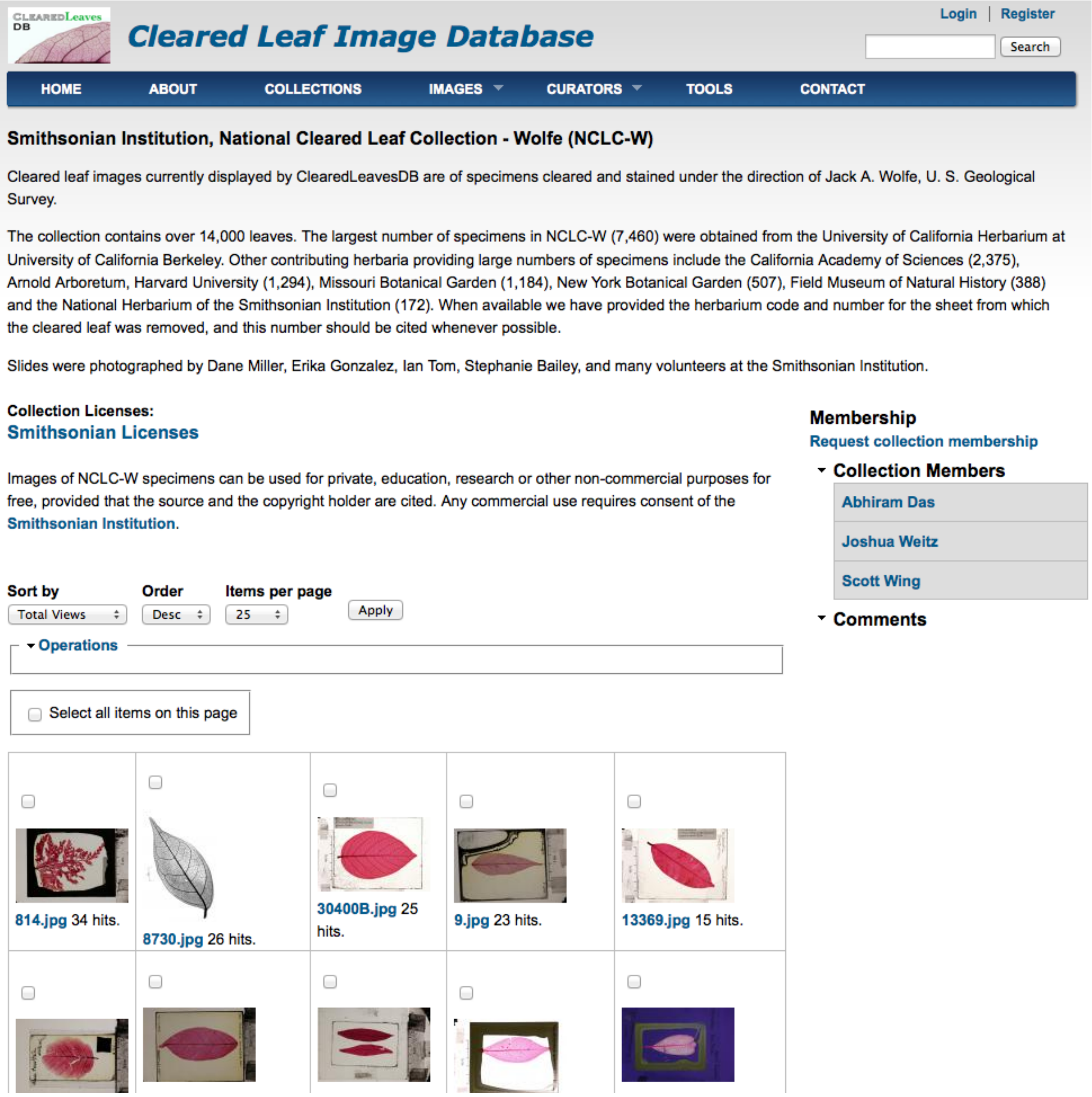


Figure 2: Geographic distribution of source specimens in ClearedLeavesDB. Each country's color denotes the number of specimens for which images are available in the database as of submission. The world map projection is via Google geo chart: https://code.google.com/apis/ajax/playground/?type=visualization#geo_chart.

<u>Search and browse images</u>: A user searches for images or collections via a name or various metadata parameters. Current search fields include: class, order, family, genus, species, blade margin, blade shape, herbarium ID, identified by, specimen number, image resolution and if has processed images. A free text search is also available. When viewing a collection (**Figure 3**) a user sees thumbnails of all images. A mouse click on the thumbnail leads to a detail image view with metadata. The search functionality is accessible via a form available in the Images menu.



Smithsonian Institution, National Cleared Leaf Collection - Wolfe (NCLC-W)

Cleared leaf images currently displayed by ClearedLeavesDB are of specimens cleared and stained under the direction of Jack A. Wolfe, U. S. Geological Survey.

The collection contains over 14,000 leaves. The largest number of specimens in NCLC-W (7,460) were obtained from the University of California Herbarium at University of California Berkeley. Other contributing herbaria providing large numbers of specimens include the California Academy of Sciences (2,375), Arnold Arboretum, Harvard University (1,294), Missouri Botanical Garden (1,184), New York Botanical Garden (507), Field Museum of Natural History (388) and the National Herbarium of the Smithsonian Institution (172). When available we have provided the herbarium code and number for the sheet from which the cleared leaf was removed, and this number should be cited whenever possible.

Slides were photographed by Dane Miller, Erika Gonzalez, Ian Tom, Stephanie Bailey, and many volunteers at the Smithsonian Institution.

Collection Licenses:
Smithsonian Licenses

Images of NCLC-W specimens can be used for private, education, research or other non-commercial purposes for free, provided that the source and the copyright holder are cited. Any commercial use requires consent of the Smithsonian Institution.

Sort by: Total Views | Order: Desc | Items per page: 25 | Apply

Operations

□ Select all items on this page

□ 814.jpg 34 hits.	□ 8730.jpg 26 hits.	□ 30400B.jpg 25 hits.	□ 9.jpg 23 hits.	□ 13369.jpg 15 hits.
□	□	□	□	□

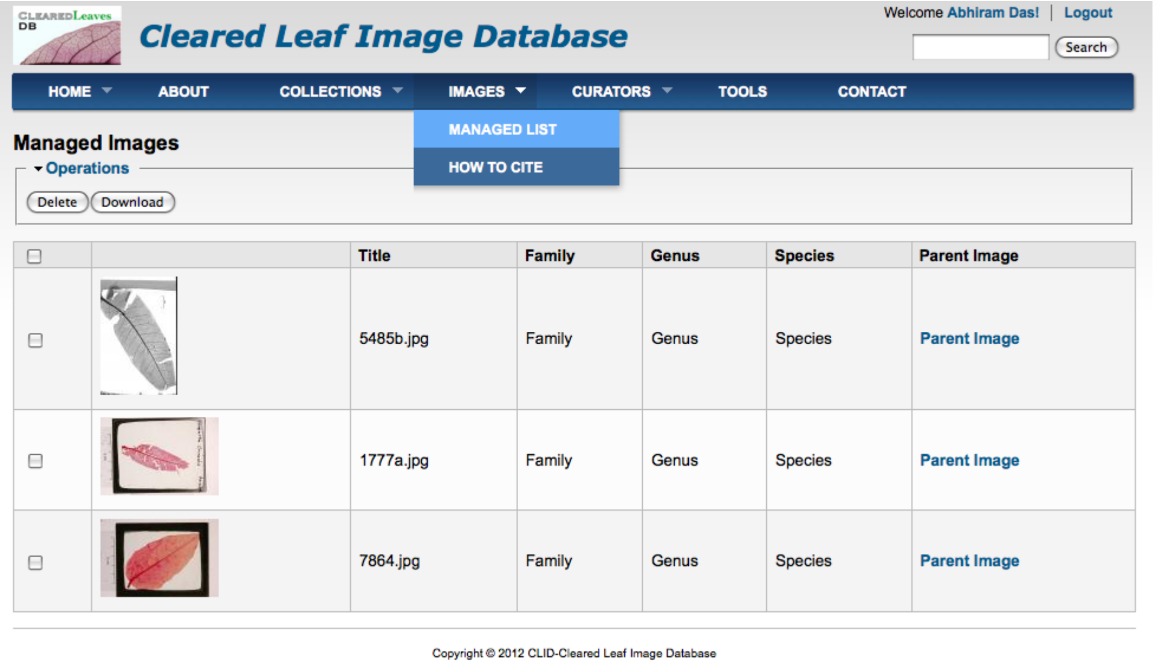
Membership
Request collection membership

Collection Members
- Abhiram Das
- Joshua Weitz
- Scott Wing

Comments

Figure 3: Collection view for curator user. Curator have right to add image to the marked list, post comments on the collection or manage collection membership.

Marked list: A marked list allows the user to maintain a list of images of interest as they search or browse. A user can add images to the marked list from the collection view module or via the search module. The marked list module also allows the user to download image groups (**Figure 4**) together with their associated metadata as a “zip” file. The downloaded zip file contains a comma-separated value (.csv) file containing the image name associated with the image metadata – such formats are compatible with all standard spreadsheet programs. Only registered users of the system have the privilege to maintain a marked list and batch download images. The metadata file contains a unique number i.e. the database identifier for each image downloaded. This number is required to associate processed images and their analysis data to the parent image in the database while uploading processed images through our desktop client interface. So, users interested in batch uploading processed images to the database are advised to use the batch download functionality to obtain image metadata. To facilitate this, we introduce a java tool for automated external access to the database without using the web interface (described next).



Managed Images

□		Title	Family	Genus	Species	Parent Image
□		5485b.jpg	Family	Genus	Species	Parent Image
□		1777a.jpg	Family	Genus	Species	Parent Image
□		7864.jpg	Family	Genus	Species	Parent Image

Copyright © 2012 CLID-Cleared Leaf Image Database

Figure 4: Marked List of images. User can bulk download images from the marked list with its metadata.

<u>Interface for linking processed data with original images</u>: ClearedLeavesDB is capable of storing original and processed images along with their metadata. The types of processed images might be binary images resulting from manual or automated segmentation of leaf veins from the leaf background as in [8, 48], skeletonized images of the approximate center-line of the leaf venation network [48], or aereole segmentation of the image [48]. Additional processed data might include the results of quantitative analysis of the

structure of the cleared leaf. In order to link post-processed images and data to the original image, we provide tools to upload processed images and its analysis data to the database from user's desktop or laptop through command line interface and application programming interface (API).

The command line interface is a Java client application that can be downloaded from ClearedLeavesDB website (http://www.clearedleavesdb.org/sites/default/files/clid.jar). The downloaded file "clid.jar" is an executable Java Archive (jar) file that contains the client application and an open API. This application and API uses Representational State Transfer (REST) Web services to connect and upload processed data to the database. It requires Java Runtime Environment (JRE) version 1.6 or above to be installed on the user's desktop or laptop. Instructions on how to launch the command line application is described in detail on the web site and can be accessed by going to the "API" tab of the "TOOLS" menu. Database users can also upload processed images and its analysis data through the web interface. To do so, users should navigate to the image view page from search, collection view or marked list page and click on "Add new processed image" link. The API is provided through a Java class called "UploadProcessedImages". It exposes an interface called "ToCLID()" that requires three parameters:

1. CSV file name (String data type)
2. Database user name (String data type)
3. Database user password (String data type)

The following Java code snippet can be used to invoke the API:

```
import clid.rest.client.UploadProcessedImages;
UploadProcessedImages.ToCLID(csvFile, username, password);
```

Please note that the CSV file should be compatible to the format as mentioned on the website. A sample file can be downloaded from ClearedLeavesDB website (http://www.clearedleavesdb.org/sites/default/files/sample_processed_upload.csv).

Analyzing images with third party software such as LeafGUI [71], LAMINA [72] and LIMANI [73] is enabled by the integrated batch download functionality. After analysis, processed images and their corresponding analysis data can be uploaded. These uploaded data are associated with the original image, hence heterogeneous and non-standard processing types can be uploaded, though the processed analysis data cannot yet be searched, a trade-off given the design decision to allow greater flexibility in allowing users to associate processed data with the original images.

User roles: The database provides multiple levels of user access to cater to different needs of the community. The database provides the following three levels of users (see **Table 1** for a list of user roles and associated functions): 1) guests, 2) registered users and 3) curators. A collection manager (i.e. a curator) can define custom roles and manage permissions for the collection irrespective of the global roles defined above. They can do so by clicking on the "Group" tab (**Figure 3**) of the collection.

Table 1: List of user roles and their associated permissions in ClearedLeavesDB.

Permissions	Guest	Registered User	Curator
Browse collections and images	√	√	√
View collections and its metadata	√	√	√
View images and its metadata	√	√	√
View high resolution original images	X	√	√
Create image collections	X	√	√
Upload images to the collection	X	√	√
Bulk upload images to the collection	X	X	√
Upload metadata to the collection	X	√	√
Comment on collections and images	X	√	√
Maintain a marked list of images	X	√	√

Table 1: continued

Download images and its metadata from the marked list	X	√	√
Image upload quota	0 KB	1 GB	20 GB
Add processed images and results	X	√	√

User scenarios

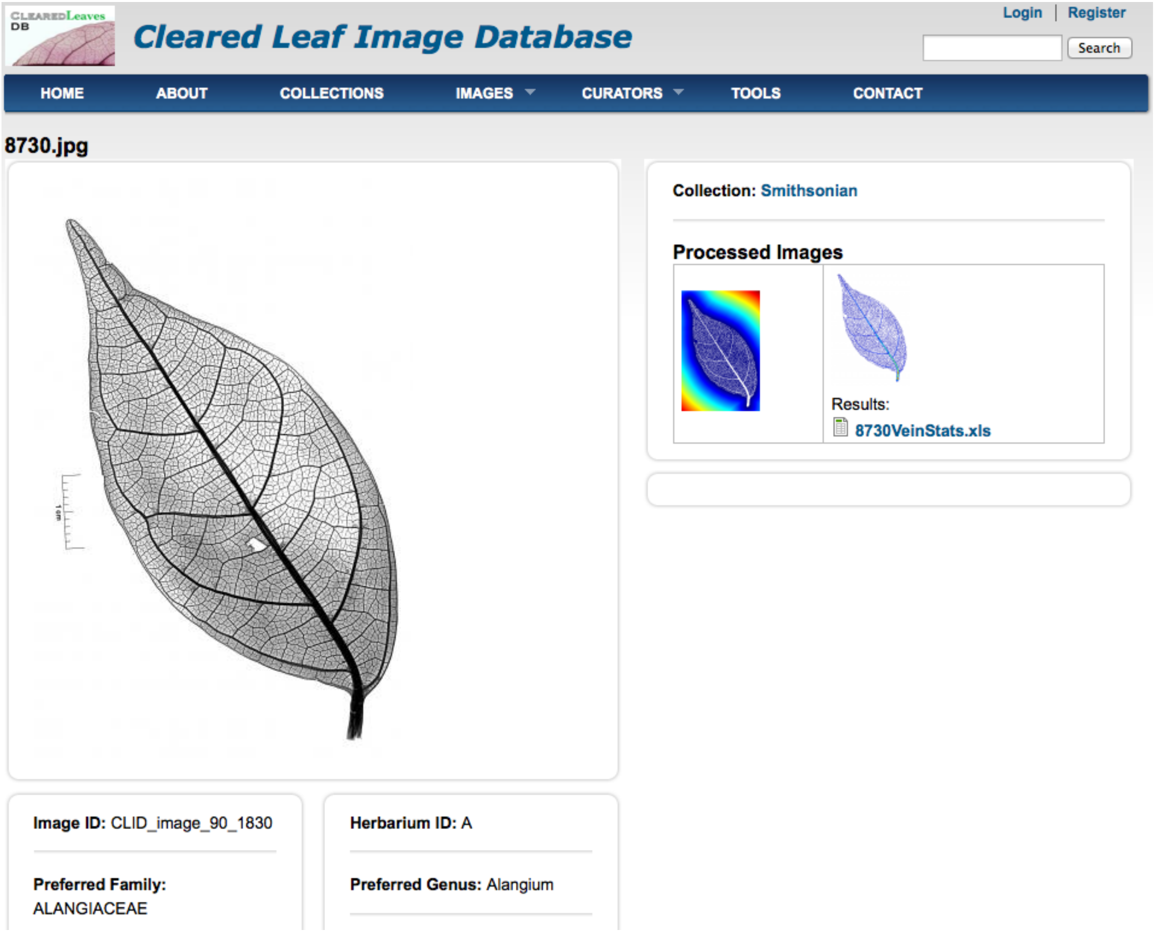
In this section we describe several examples to highlight the main use cases of our database: database browsing by a guest user; creating an image collection; managing collection membership, roles and permissions; and uploading processed images through REST Web Services client.

A guest browsing the database: A user accessing the database will first arrive on the home page displaying the welcome message and a list of public collections available on the database. The user can either select any one of the collections shown or click on the “Collections” menu to search for a public collection and then click on it to browse the collection. The collection view (**Figure 3**) displays thumbnail images in the collection in a grid format along with members of the collection and comments posted by different users. The guest user does not have rights to post comments, add images or request membership to the group. Guests can select an image and view its metadata but cannot view full size image, post comments on the image or edit metadata. Guests can click on the “Images” menu to search for images with certain filters. From the image search page user can also navigate to the image details view page as described earlier. Guests do not have access to marked list functionality i.e. adding images to the marked list from collection or search page.

Registered user and Curator creating image collection: Here we describe the difference in creating an image collection by an authenticated user and a curator. Both these user roles have the rights to create image collections but registered users do not have rights to bulk upload images to the collection. We demonstrate this functionality using a registered user and a curator. Both these users have access to the "Create Collection" sub-menu in "Collections" menu on the database but the registered user does not have access to "Bulk Upload Images" sub-menu. After creating a collection, the registered user is redirected to the single image upload page whereas the curator is redirected to the bulk upload page.

Managing collection membership, roles and permissions: The founding curator (i.e., "author") of a collection has all the rights to manage it members, roles and permissions. The author can visit the group tab (**Figure 3**) and click on the "Add People" link to add people to the collection and click on the "People" link to manage membership. Similarly "Roles" and "Permissions" link can be followed to manage roles and permissions of the collection.

Uploading processed images through REST Web Services client: Our client application can be used to upload processed images and its analysis data to the database. A registered user or a curator can add images to its marked list (**Figure 4**) and batch download the images with its metadata using the download feature. After the images are downloaded a user can use any image processing software to analyze its traits and produce processed images and analysis data files. These processed images and analysis data files can be uploaded to the database using the Web services client. The client application reads a csv file containing parent image id and corresponding processed image and data file. A sample csv file is available for download at the website in the FAQ section (i.e. CSV file format) of the 'About' page. The image name and ID in the csv file are obtained from the metadata file downloaded along with the images. In this case study, we downloaded three images from a user's marked list (**Figure 4**), processed them using Leaf GUI [71] and uploaded the analysis data using the client application. A user can view the processed images and their analysis data by navigating to the parent image's view page (**Figure 5**).



8730.jpg

Collection: Smithsonian

Processed Images

Results:
8730VeinStats.xls

Image ID: CLID_image_90_1830

Preferred Family:
ALANGIACEAE

Herbarium ID: A

Preferred Genus: Alangium

Figure 5: View of the image showing associated processed image and its results.

Datasets: ClearedLeavesDB is seeded with five distinct collections. The largest is from the National Cleared Leaf Collection housed at the Museum of Natural History, Smithsonian Institution, including specimens cleared and stained under the direction of Jack A. Wolfe, U. S. Geological Survey. The other collections are of (i) Arabidopsis thaliana specimens from a study of variation in venation network traits for which data include ecotypes, RILs, NILs, and vascular patterning mutants (Blonder et al., unpublished); (ii) Specimens from multiple Populus tremuloides clones from the Colorado Rocky Mountains [74]; (iii) specimens taken from the University of Arizona arboretum [75]; (iv) specimens taken from oak trees of different species on the campus of the Georgia Institute of Technology (Price et al., unpublished).

2.3 Discussion

ClearedLeavesDB provides a range of practical tools to store, manage and access cleared leaf images. The web interface for the database is built on open source technologies and is freely accessible online. At present, the database is seeded with over 40 GB of primary images of cleared leaves representing a total of 19,000 images. In doing so, ClearedLeavesDB provides a means to connect researchers, institutional repositories, and the public in accessing, sharing, and analyzing the biology of plant leaves. In this sense, ClearedLeavesDB is complementary to pre-existing websites that enable access to images of plants and plant organs, and specialized repositories of cleared leaf images. ClearedLeavesDB aims to bridge the gap between these two types of platforms by offering individuals and institutions a common platform, built on open source technology, to store, manage, share, view and analyze cleared leaf images. At present, ClearedLeavesDB leverages the infrastructure of iPlant [66] for flexible storage and access of third-party developed tools. In moving forward, we plan to extend the current system to integrate the database with iPlant's Data Store [66] built using integrated rule-oriented data-management system (iRODS) [76]. Hence, future efforts to analyze large-scale datasets of cleared leaves may benefit from bringing the software analysis tools to the data, rather than the other way around, as well as leveraging many of the other benefits of a scalable infrastructure.

The database is named as "ClearedLeavesDB" and is accessible at http://clearedleavesdb.org. The database can be accessed through any web browser; however it has been tested on Firefox (Version 15.0.1) and Chrome (Version 21.0.1180.90). A quick start guide is available on the website and is included here as **Appendix A**. The public datasets present in the database are freely accessible but carry copyright information defined by the collection author.

2.4 Acknowledgements

The authors thank Scott Wing and Benjamin Blonder for their contributions of datasets to the initial release of Cleared Leaves DB and also thank Troy Hilley for technical contributions to database testing. This project was supported, in part, by a seed grant from the Center for Data Analytics at the Georgia Institute of Technology (to AB and JSW). Joshua S Weitz, PhD, holds a Career Award at the Scientific Interface from the Burroughs Wellcome Fund. CAP is supported by a Discovery Early Career Research Award from the Australian Research Council.

CHAPTER 3

DIRT: A HIGH-THROUGHPUT COMPUTING AND COLLABORATION PLATFORM FOR FIELD-BASED PLANT PHENOMICS

This chapter is adopted from the publication Das A, Schneider H, Burridge J, Martinez Ascanio AK, Wojciechowski T, Topp CN, Lynch JP, Weitz JS, Bucksch A;DIRT: a high-throughput computing and collaboration platform for field-based plant phenomics; 2015, In Review.

3.1 Background

Automated high-throughput phenotyping methods are increasingly used in laboratory-based efforts to link plant genotype with phenotype, whereas similar field-based studies remain predominantly manual and low-throughput. Here, we present an open-source phenomics platform "DIRT" (**D**igital **I**maging of **R**oot **T**raits), as a means to integrate scalable supercomputing architectures into field experiments and analysis pipelines. DIRT is an online platform that enables researchers to store images of plant roots, measure dicot and monocot root traits under field conditions, and share data and results within collaborative teams and the broader community. The DIRT platform seamlessly connects end-users with large-scale compute "commons" enabling the estimation and analysis of root phenotypes from field experiments of unprecedented size.

Global food demand is projected to double by the year 2050 [28]. Meeting this increased demand requires significant improvements in crop yield and the development of crop plants adapted to water-stress [77] and low fertility soils [78]. Breeding more efficient roots is increasingly recognized as a high-priority target to achieve yield improvements because roots are essential for nutrient and water uptake [12]. Yet, little is known regarding the relationship between root system architecture (RSA) and crop function with few examples linking root phenotype to genotype and/or to phenotypic advantages under given field conditions [14, 29].

Developing new crop varieties includes both, laboratory- and field-based studies. Especially field studies involve laborious manual tasks, which limits the achievable sample size to characterize RSA of mature field-grown crops. Extending field-based studies and sample sizes is a widely shared goal for future phenotyping scenarios [79]. Indeed, phenotyping rather than genotyping is recognized as the bottleneck limiting advances, given inexpensive next-generation sequencing technologies that have paved the way for characterizing the genotypes of diversity panels of thousands of recombinant inbred lines [30]. In response, a number of national and international efforts, including the International Plant Phenotyping Network, have established “plant phenomics” centers [79, 80] to quantify plant phenotypes and identify their genetic origin.

Despite some successes, there are relatively few publicly available root phenotyping datasets [81]. Available large datasets are pre-dominantly derived from laboratory-based root phenotyping platforms. Laboratory studies benefit from increased levels of control and, at least in a few cases, have identified loci with candidate genes underlying RSA in early root development [82]. However, growth containers used in these studies, filled with real or artificial soil [18, 42, 83], limit observations spatially and temporally to small or immature root systems.

Establishing a link between RSA and genotype requires the measurement of root phenotypes, often derived from automatic analysis of two-dimensional and three-dimensional digital images. A comprehensive overview of existing software for root image analysis is maintained at the site: www.plant-image-analysis.org [84]. The scope of this software collection is impressive, in that individual tools provide different degrees of computational automation, ranging from manual, semi-automatic to fully automatic. However, none of these provide an integrated platform that can (a) associate root images with environmental and phenotypic meta-data, (b) provide seamless access to scalable, supercomputing resources for non-technical users and (c) share information within a collaborative team and the plant science community [81].

3.2 Construction and Methods

DIRT addresses these phenotyping bottlenecks within the computational plant sciences, by providing a single platform to meet the demands of data access and storage, exchange and sharing, and image-based high-throughput root phenotyping [1]. DIRT is a multi-tiered integrated online platform developed using the Drupal framework [85]. As depicted in **Figure 6**, it utilizes a three tier architecture where the client tier represents the user interfaces on the web browser, the processing tier represents the Drupal modules and the image processing pipeline, and the storage tier represents the database and file systems. This makes it a unique root phenotyping platform, accessible via an interactive web-based interface. **Figure 7** shows a high-level overview of the DIRT workflows. The algorithms deployed on DIRT have been specifically designed and tested on two-dimensional images taken of root systems in the field (**Figure 9**). By focusing on crop root traits, DIRT also overcomes the time consuming manual measurement processes involved in Shovelomics [27], while enabling measurements of manually inaccessible traits such as Dominant Root Tissue Angle [1].

The RSA trait computation pipeline (developed in python and configurable to run on any high-throughput grid computing environment) (**Figure 8**) available in DIRT is fully automated (by user of parameters listed in **Table 2**) and includes automatic estimation of 78 traits in total, categorized into common traits (**Table 3**) for all root system architectures, dicot traits (**Table 4**), monocot traits (**Table 5**) and traits for excised root samples (**Table 6**). We provide a separate, optional threshold calibration tool that allows the researcher to select a representative image from the marked collection and compute masked images using different segmentation threshold values. As a last step in this calibration workflow, the user selects the most appropriate value by visually checking the binary masked image.

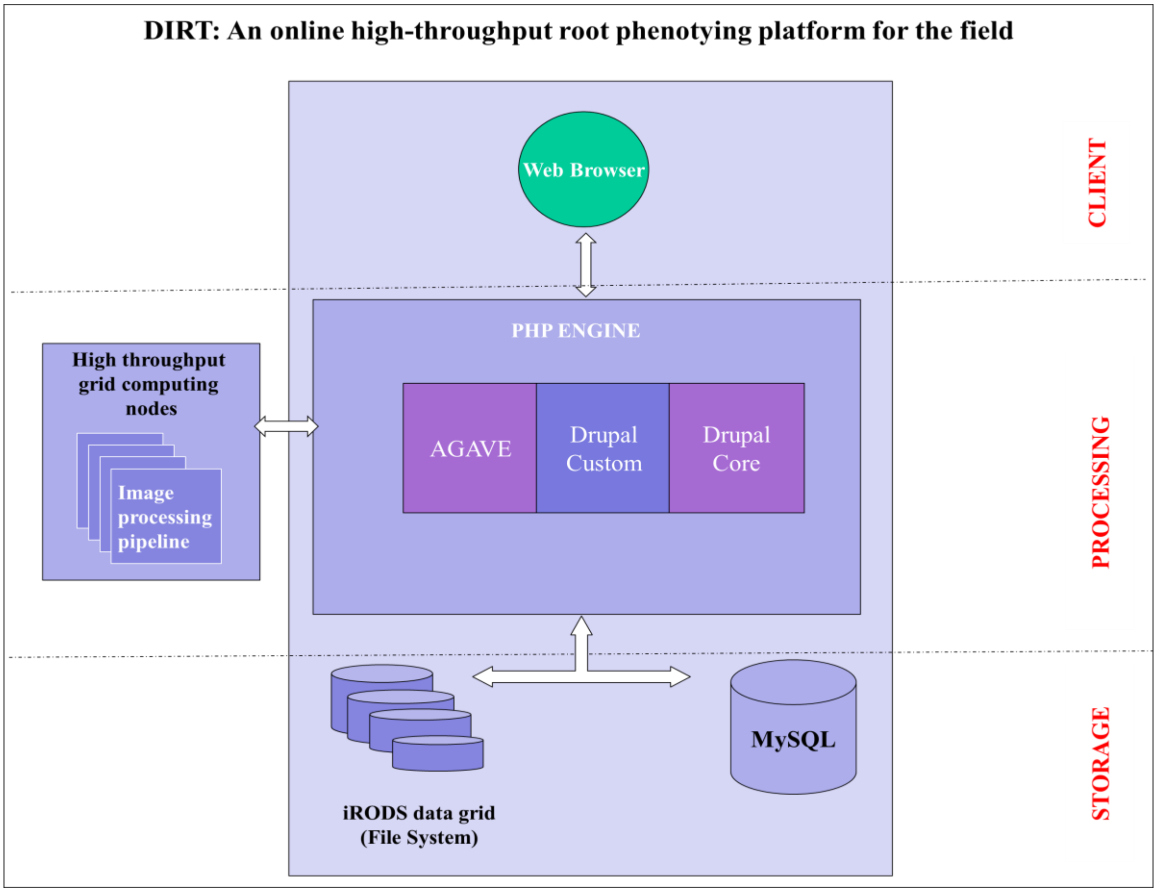

DIRT: An online high-throughput root phenotying platform for the field
Web Browser
CLIENT
PHP ENGINE
AGAVE
Drupal Custom
Drupal Core
High throughput grid computing nodes
Image processing pipeline
PROCESSING
iRODS data grid (File System)
MySQL
STORAGE

Figure 6: DIRT Architecture. DIRT is programed within the framework of the Drupal content management system and can be configured to interact with any high-throughput grid computing environment. The iPlant installation uses the Agave API [86-88] to communicate with the high-throughput computing environment. The Agave API [86-88] is utilized to transfer images from the iPlant data store [66, 89] to the computing platform and execute the computation. Metadata is organized in a MySQL database. All customized Drupal modules are open source and provide the functionality for configuration and communication with remote high-throughput grid computing platform.

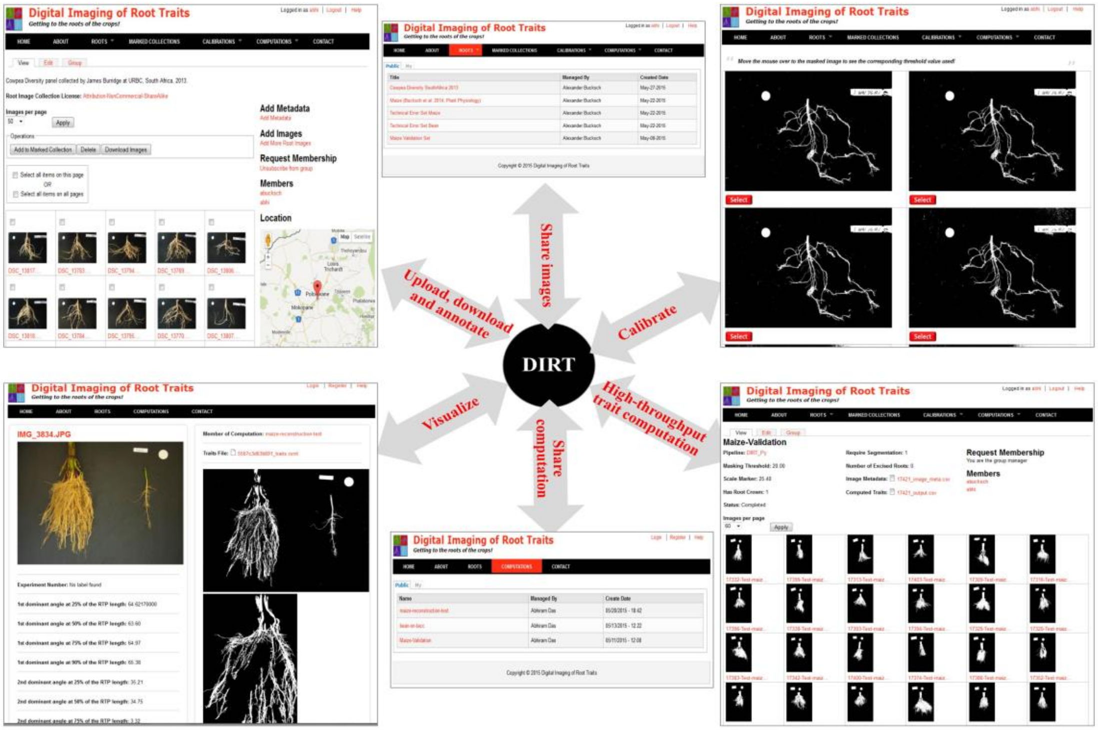

Digital Imaging of Root Traits
Getting to the roots of the crops!
Upload, download and annotate
Share images
Calibrate
DIRT
Visualize
Share computation
High-throughput trait computation

Figure 7: Major workflows performed within DIRT. It enables storage of root images as a collection, upload and download of images to/from the collection, annotation of collection and images with metadata, sharing of collections with the community, calibration of segmentation threshold of images, computation of RSA traits in high-throughput, sharing of computed RSA traits with the community, and visualization via simple web interfaces.

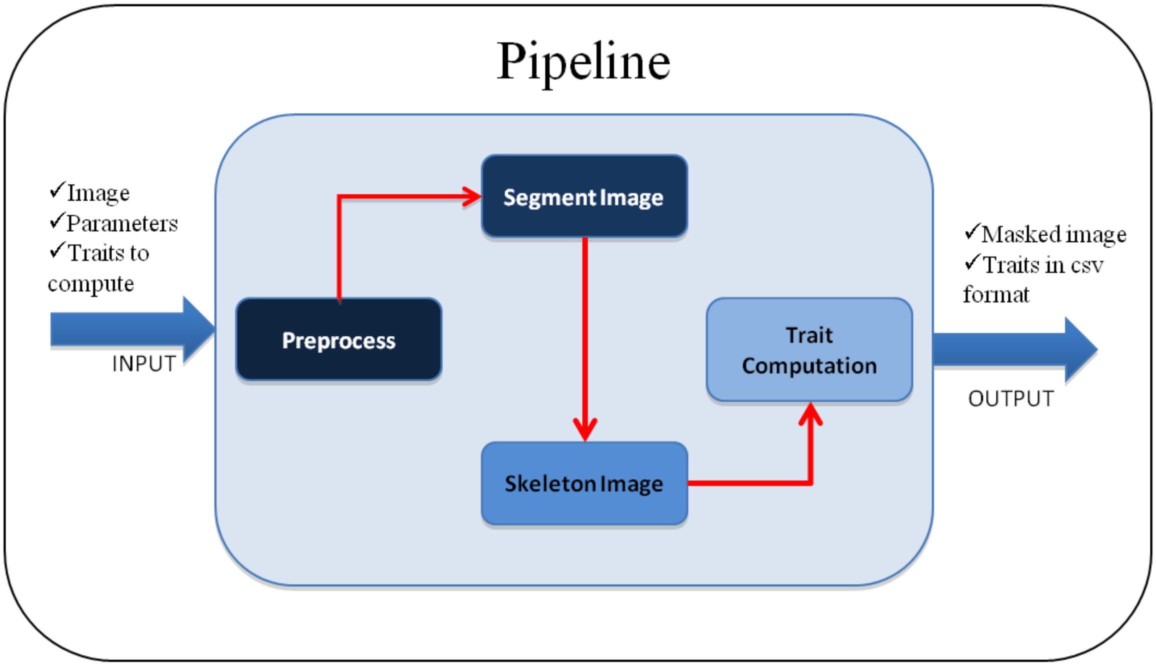
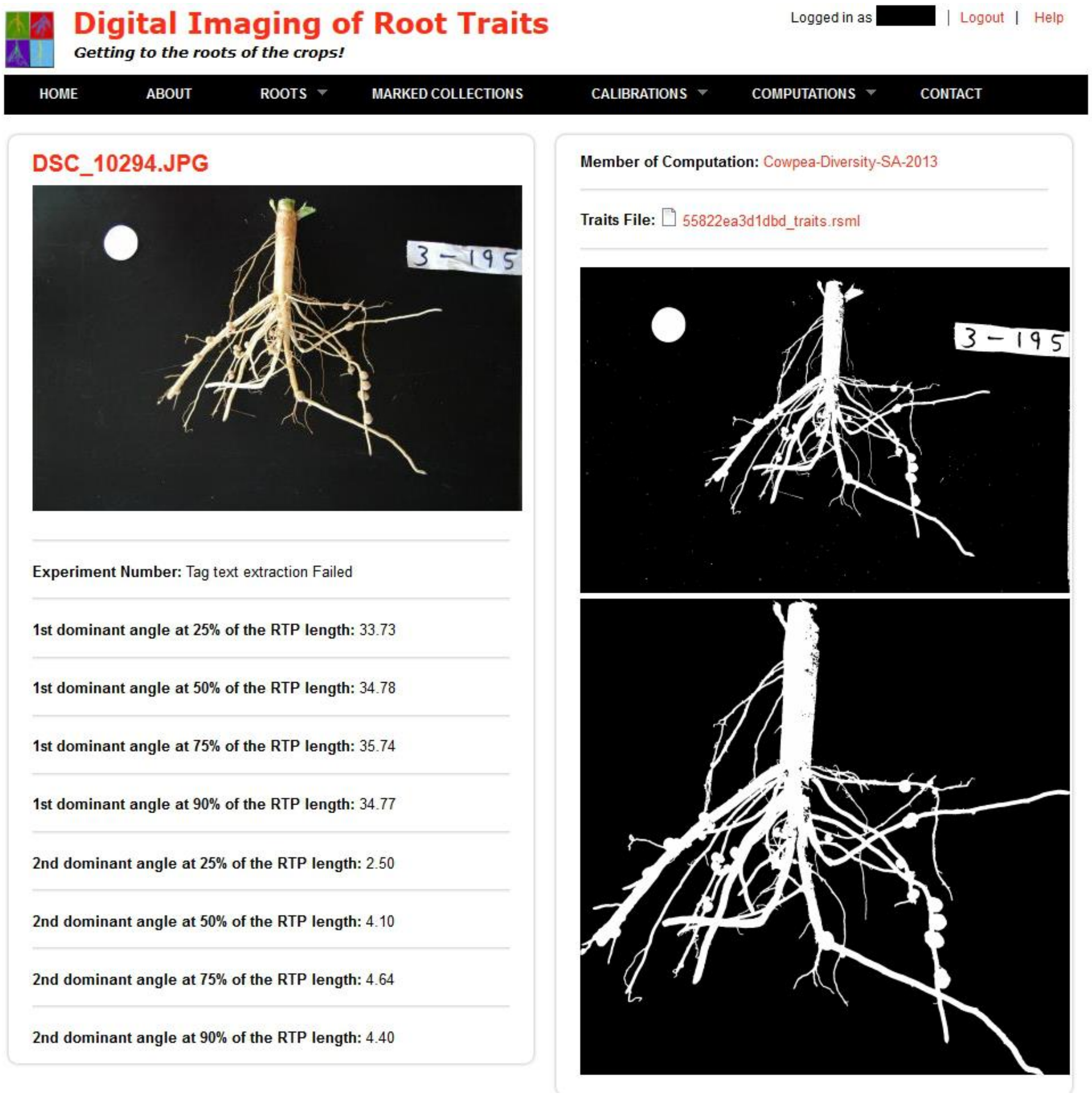


Figure 8: A schematic showing different steps of the DIRT RSA trait computation pipeline.

Table 2: Input parameters to DIRT RSA trait computation pipeline.

Name	Requirement	Description
Masking Threshold	Mandatory	A real value that is used to segment the root structure from the image background.
Scale Marker	Mandatory	The known scale marker diameter allows correction of camera tilting and transforming image coordinates into metric units. For example check the image in **Figure 10**. A value of 0.0 indicates that no scale marker is present, but the traits will be shown in pixels.
Stem Reconstruction	Optional	Turns on an optional reconstruction of the stem. Maize genotypes occasionally have dark stem parts that have to be reconstructed because they cannot be differentiated from the dark background. We suggest careful use and visual inspection of this feature.
Require Segmentation	Optional	Whether segmentation of foreground and background is required for the image.
Has Root Crown	Optional	Whether the root in the image has crown roots.
Number of Excised Roots	Optional	Number of excised roots present in the image. For example the image in **Figure 10** has one excised root.



Logged in as [redacted]

DSC_10294.JPG

Member of Computation: Cowpea-Diversity-SA-2013

Traits File: 55822ea3d1dbd_traits.rsml

Experiment Number: Tag text extraction Failed

1st dominant angle at 25% of the RTP length: 33.73

1st dominant angle at 50% of the RTP length: 34.78

1st dominant angle at 75% of the RTP length: 35.74

1st dominant angle at 90% of the RTP length: 34.77

2nd dominant angle at 25% of the RTP length: 2.50

2nd dominant angle at 50% of the RTP length: 4.10

2nd dominant angle at 75% of the RTP length: 4.64

2nd dominant angle at 90% of the RTP length: 4.40

Figure 9: The Dirt interface provides visual quality control for individual root images. The original image of a cowpea root is shown on the left containing a round scale marker of known diameter, the excavated root on a black yet diffuse reflecting background and a tag for documentation. The extracted binary mask on the top right and the automatically detected root on the bottom right are visual guides to control for potential outliers. The computed trait values are shown on the bottom right.

The DIRT platform provides a number of major functionalities that enable researchers to: (a) manage root image collections and metadata; (b) interactively calibrate measurement pipelines; (c) compute crop root traits on scalable compute platforms; (d) analyze the results of computations. Here, a root image collection contains a set of root images that typically represents one field experiment. A marked collection is a set of images that the researcher has selected/marked from one or more root image collections for trait computation. The inclusion of “marked” collections enables the creation of virtual experiments combined over accessible data sets. A user can maintain multiple marked collections on DIRT, each being either a previously uploaded dataset or a selection of images from all image collections accessible to a user.

Table 3: Common root traits computed by the DIRT RSA trait computation pipeline. (* annotates traits requested by the user community)

Trait Name	Trait Code	Trait Description
Stem Diameter	DIA_STM	Stem Diameter derived from the medial axis
Simple Stem Diameter*	DIA_STM_SIMPLE	Simple Stem Diameter as calculated in Root Estimator for Shovelomics Traits (REST) from the ETH Zürich [90]
Projected Root Area*	AREA	Number of foreground pixels belonging to the root system. Previously defined in GiA Roots [25]
Average Root Density	AVG_DENSITY	Ratio of foreground to background pixels with in the root shape
Median Tip Diameter	TD_MED	Median Tip Diameter estimated from the medial circle at the tips
Mean Tip Diameter	TD_AVG	Mean Tip Diameter estimated from the medial circle over all detected tips
Median width of root system	WIDTH_MED	Median width of root system measured horizontally from the first to the last foreground pixel
Maximum width of root system	WIDTH_MAX	Maximum with of root system measured horizontally from the first to the last foreground pixel
Accumulated width over 10%-90% percent depth (D-values)	D10-D90	Percentage of width accumulation at 10%-90% depth
Slope of the graph of D-values	DS10 – DS90	Slope at the D10-D90 value that represents the rate of accumulation
Spatial Root Distribution X	RDISTR_X	Spatial distribution of the root shape in x-axis. This is the x component of the vector pointing from the center of the bounding box of the root shape to the center of mass of the root shape
Spatial Root Distribution Y	RDISTR_Y	Spatial distribution of the root shape in y-axis. This is the y

Table 3: continued

		component of the vector pointing from the center of the bounding box of the root shape to the center of mass of the root shape
Rooting depth skeleton*	SKL_DEPTH	Rooting depth calculated from the medial axis of the root system. Previously used in GiA Roots [25]
Skeleton Width*	SKL_WIDTH	Width calculated from the medial axis of the root system. Previously used in GiA Roots [25]
Number of Root Tip Paths (RTPs)	RTP_COUNT	Corresponds to the overall number of tips detected in the image

Table 4: Dicot root traits computed by the DIRT RSA trait computation pipeline. (* annotates traits requested by the user community)

Trait Name	Trait Code	Trait Description
Soil Tissue Angle Range (STA)	STA_RANGE	Range of STA angles present in the root
First Dominant Soil Tissue Angle	STA_DOM_I	Average of the 1st significant peak bin in the histogram of calculated soil tissue angles
Second Dominant Soil Tissue Angle	STA_DOM_II	Average of the 2nd significant peak bin in the histogram of calculated soil tissue angles
Soil Tissue Angle x% 1	STA_25_I, STA_50_I, STA_75_I, STA_90_I	1st dominant angle at 25%, 50%, 75%, 90% of the RTP length
Soil Tissue Angle x% 2	STA_25_II, STA_50_II, STA_75_II, STA_90_II	2nd dominant angle at 25%, 50%, 75%, 90% of the RTP length
Dominant Root Tissue Angle 1	RTA_DOM_I	Average of the 1st significant peak in the histogram of calculated root tissue angles binned in 1 degree steps
Dominant Root Tissue Angle 2	RTA_DOM_II	Average of the 2nd significant peak in the histogram of calculated root tissue angles binned in 1 degree steps
Minimum Soil Tissue Angle	STA_MIN	Minimum Soil Tissue Angle measured over all RTPs
Maximum Soil Tissue Angle	STA_MAX	Maximum Soil Tissue Angle measured over all RTPs

Table 4: continued

Median Soil Tissue Angle	STA_MED	Median Soil Tissue Angle measured over all RTPs
Root Tissue Angle Range	RTA_RANGE	Range of RTA angles present in the root
Minimum Root Tissue Angle	RTA_MIN	Minimum Root Tissue Angle measured over all RTPs
Maximum Root Tissue Angle	RTA_MAX	Maximum Root Tissue Angle measured over all RTPs
Median Root Tissue Angle	RTA_MED	Median Root Tissue Angle measured over all RTPs
Roots Seg 1*	NR_RTP_SEG_I	Number of RTPs emerging from the Hypocotyl (Root seg 1)
Roots Seg 2*	NR_RTP_SEG_II	Number of RTPs emerging from the taproot (Root seg 2)
Number of adventitious roots*	ADVT_COUNT	Number of adventitious roots estimated as RTP bundles emerging from root seg 1
Number of basal roots*	BASAL_COUNT	Number of basal roots estimated as emerging RTP bundles from root seg 2
Adventitious root angles*	ADVT_ANG	Adventitious root angel estimated from the paths detected in the number of adventitious roots
Basal root angles*	BASAL_ANG	Basal root angles estimated from the paths detected in the number of basal roots
Hypocotyl Diameter*	HYP_DIA	Hypocotyl Diameter estimated over the detected hypocotyl region as the average of diameters of medial circles
Tap root Diameter*	TAP_DIA	Tap root diameter estimated over the detected taproot region as the average of diameters of medial circles
Maximum diameter at 90-100 percent depth	MAX_DIA_90	Maximum diameter found in the interval of 90-100 percent rooting depth
50 percent drop*	DROP_50	Depth value were 50% of the RTPs emerged from the central path (hypocotyl+taproot)
Central path diameter at 25 percent depth	CP_DIA25, CP_DIA50, CP_DIA75, CP_DIA90	Approximation of the tap root diameter at 25%,50%,75%,90% of the rooting depth

Table 5: Monocot root traits computed by the DIRT RSA trait computation pipeline. (* annotates traits requested by the user community)

Trait Name	Trait Code	Trait Description
Root Top Angle*	ANG_TOP	Root Top Angle measured between the Random Sample Consensus [91] (RANSAC) fit line at depth of the D10 value and the horizontal soil line.
Root Bottom Angle*	ANG_BTM	Root Bottom Angle measured between the RANSAC fit line at depth of the D80 value and the horizontal soil line.

Table 6: Excised root traits computed by the DIRT RSA trait computation pipeline.

Trait Name	Trait Code	Trait Description
Average lateral root length	NODAL_LEN	Average length of lateral roots along the central path
Nodal root path length	NODAL_AVG_DIA	Length of the central path along the excised root
Lateral branching frequency	LT_BRA_FRQ	Lateral branching frequency
Mean nodal root diameter	LT_AVG_LEN	Mean nodal root diameter measured along the medial axis of the excised root sample
Lateral mean angle	LT_AVG_ANG	Mean angle of all lateral roots emerging from the excised root sample
Lateral angular range	LT_ANG_RANGE	Range of angles of the lateral root sample
Lateral minimum angle	LT_MIN_ANG	Minimal lateral angle present in all measurements of the excised root sample
Lateral maximum angle	LT_MAX_ANG	Maximal lateral angle present in all measurements of the excised root sample
Distance to first lateral	LT_DIST_FIRST	Distance to first lateral along the medial axis of the excised root
Median diameter of lateral roots	LT_MED_DIA	Median diameter of lateral roots estimated from the medial axis
Mean diameter of lateral roots	LT_AVG_DIA	Mean diameter of lateral roots

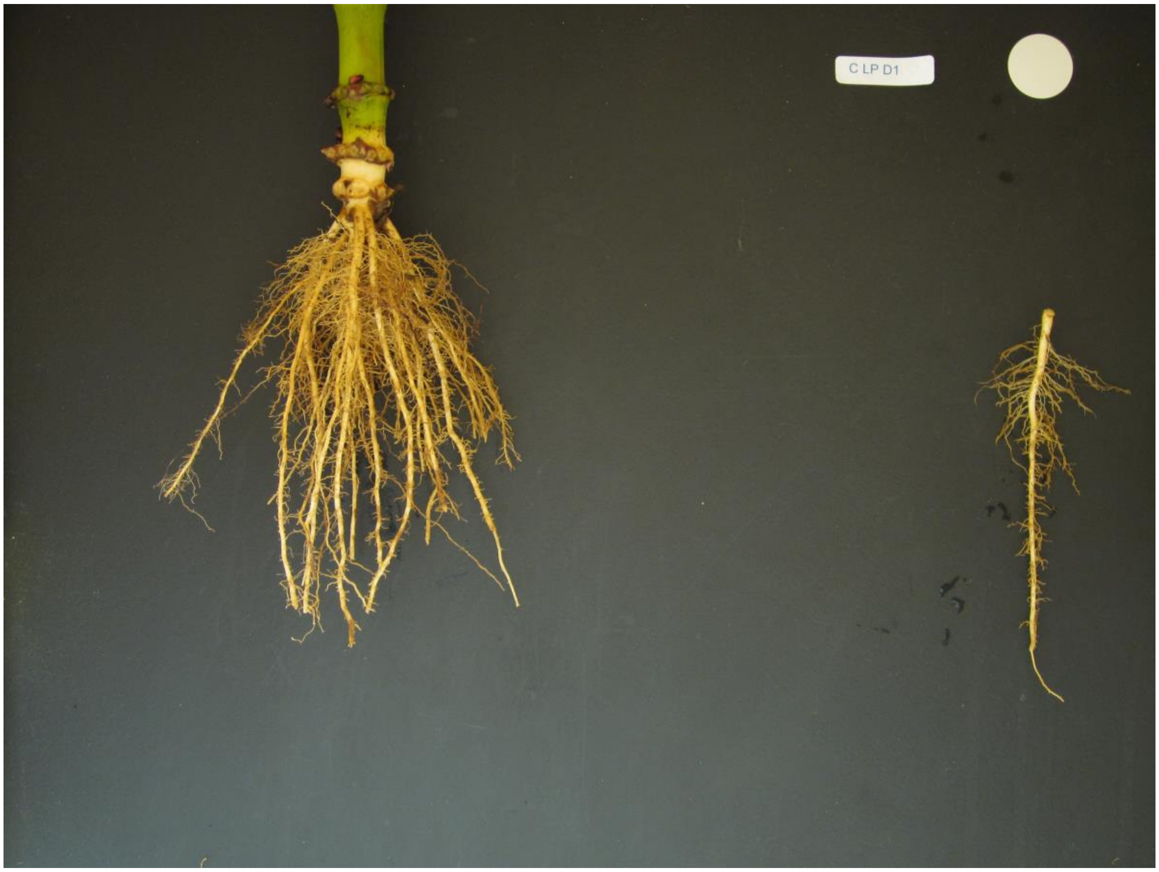


Figure 10: An example image that uses all DIRT image acquisition features [1]. The round scale marker is needed to recalculate pixels into units. The rectangular experiment tag is automatically detected and its content (letters or barcode) is stored in the result file. The root crown and the excised roots are separately analyzed.

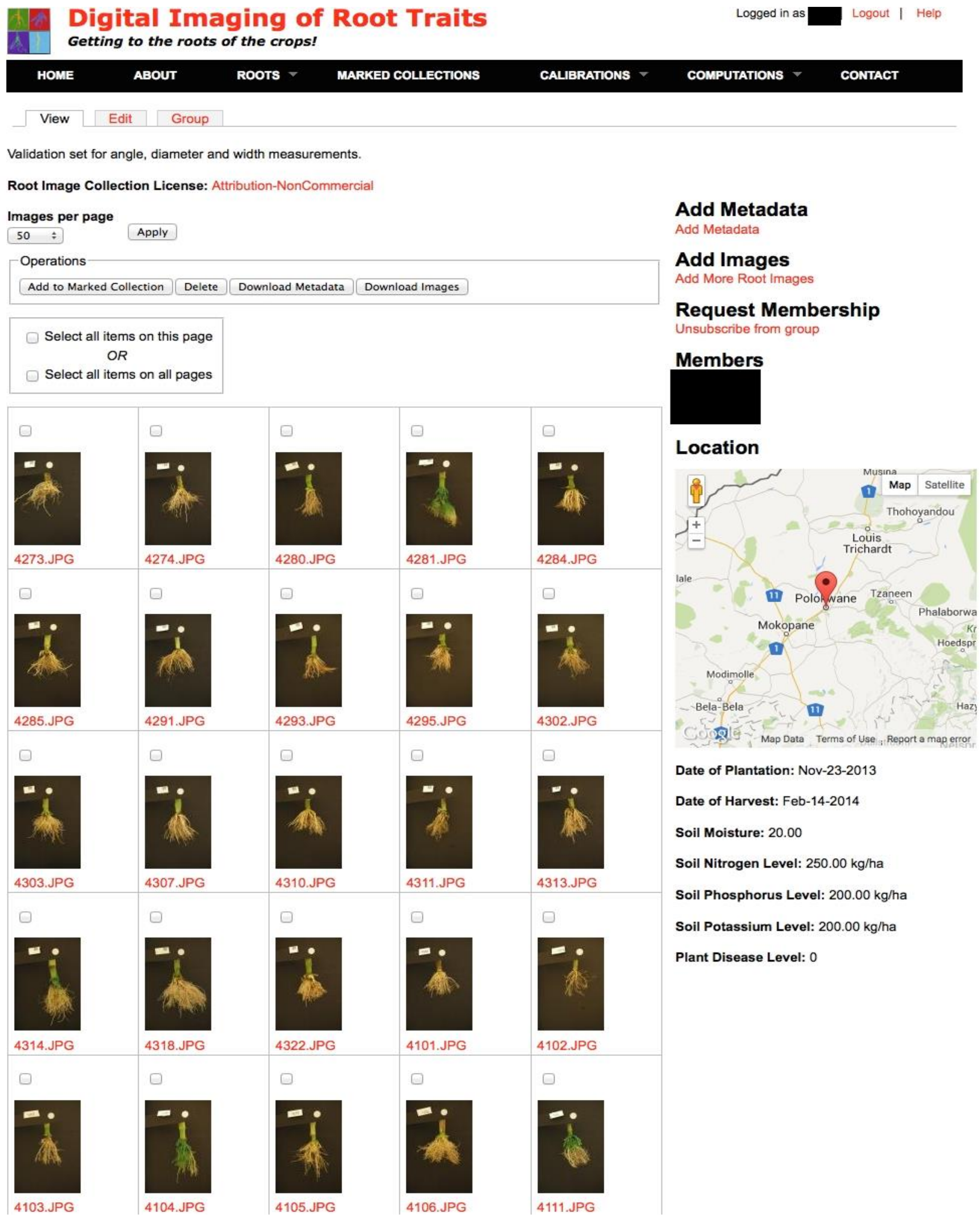
The core trait computation pipeline in DIRT extends algorithms originally developed for the analysis of crop root traits, as applied to both monocot and dicots [1]. The current pipeline includes previously unpublished algorithms to measure traits such as top and bottom angle in monocots (see **Table 5**), which were requested by the user community. This pipeline follows a protocol to process 2D root images. In brief, the washed root is imaged against a dark diffuse reflecting background that contains a light colored circle with known diameter (shown in **Figure 10**). Additionally, a barcode, QR-code or simple text can be placed above the root for automatic identification to be associated with trait computations. On completion of the computation, masked images, traits, and corresponding CSV and RSML files [92] populate the computation view tab.

The trait computation pipeline is abstracted from the computational resources and from the aggregation and sharing of data and images. Hence, it is possible to extend DIRT by incorporating new pipelines adapted to distinct imaging and field conditions (see **Appendix B**).

The design of DIRT is intended to help researchers either individually or collaboratively to take full control of their data including root images, metadata and computations. Each newly created collection is designated to be private by default, but it can also be made public via a Creative Commons License. Moreover, data and computed results can be shared privately with one or many collaborators via the platform's web-interface. DIRT enables different functions based on user access rights. The owners of data can edit, upload, download and delete images and corresponding metadata. Metadata for whole experiment conditions per data set to document experiment conditions (e.g. FAO soil type, GPS location, soil moisture content) can be entered per data set. On top of suggested standard experiment parameters a dynamic form allows the documentation of non-standard parameters such as nitrogen content per depth level. Each root image can be annotated manually or by uploading a pre-formatted CSV file with specific metadata (e.g. genotype, dry biomass) and may contain RSML files of manual measurement to annotate the image, e.g. from RootNav [93] (see **Appendix B**).

3.3 Discussion

DIRT is hosted publically on the iPlant cyber-infrastructure [89] leveraging its cloud data storage and the Advanced Agave API to communicate with the Texas Advanced Computing Center (TACC) for high-throughput computation of stored root images. It is built as a multi-tiered application of web server, database server, iPlant's data store, middleware and grid computing (see **Figure 6**). The core middleware components are the PHP modules interfacing the database, iPlant data store and grid-computing environment. DIRT's web interface is developed using the widely adopted open source content management system Drupal (http://www.drupal.org). **Figure 11** presents views of the DIRT interface in a web browser. Root images and its metadata are organized in MySQL database and stored in iPlant's data store. The image-processing pipeline is developed in python and runs on TACC. The DIRT source code and installation instructions are available for download from the DIRT website (see **Appendix B**) to facilitate use of private supercomputing resources for the plant science community. As a proof of concept we have also released an installation of DIRT at Georgia Tech (http://www.dirt.biology.gatech.edu) that uses Georgia Tech's high performance computing environment; instructions for a local installation of DIRT on proprietary computing resources are described in **Appendix B**.


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
View
Edit
Group

Validation set for angle, diameter and width measurements.

Root Image Collection License: Attribution-NonCommercial

Images per page: 50

[redacted]

Date of Plantation: Nov-23-2013

Date of Harvest: Feb-14-2014

Soil Moisture: 20.00

Soil Nitrogen Level: 250.00 kg/ha

Soil Phosphorus Level: 200.00 kg/ha

Soil Potassium Level: 200.00 kg/ha

Plant Disease Level: 0

Figure 11: Screenshot from the DIRT web-application. The screenshot shows the root collection overview tab for a maize validation data set collected in Ukulima at the Ukulima Root Biology Center, South Africa. On the top the main menu is visible that contains all functionality to manage root images, create marked collections, run computations and perform the threshold calibration. Individual root images are shown below, along with an informal description of the dataset, an accompanied Creative Commons license and the location of the root excavation.

We provide 10 public data sets (**Table 7**) for newly registered users to test the platform. These initial data sets contain 4894 root images of field-grown roots excavated with the Shovelomics [27] technique. In **Figure 12** we show maize traits that were requested and validated by the DIRT user community. All traits were measured manually and compared against DIRT measurements. The validation data set is publicly available at http://dirt.iplantcollaborative.org/content/maize-validation-set. Note, only traits with manually accessible counter parts can be validated. However, DIRT can compute traits that are manually inaccessible. In the spirit of open-source development, we have hosted DIRT on iPlant's cyber infrastructure, which is open to the public, and have made all related source code accessible via the DIRT GitHub repository (https://github.com/abucksch). Our efforts to make DIRT an open-source, transparent and freely accessible tool will enable further development and adaptation of the platform to related research needs [81]. DIRT is freely accessible and usable at http://dirt.iplantcollaborative.org.

Table 7: Data sets publicly available on DIRT at http://dirt.iplantcollaborative.org

Name	Previously used in publications	Number of Images	Description
Cowpea Diversity SouthAfrica 2013	[1]	1500	Cowpea Diversity panel collected by James Burridge at URBC, South Africa, 2013
Maize Wisconsin Diversity Panel	[1]	85	Subset of the Wisconsin Diversity Panel collected by Eric Nord, Penn State University and Scott Stelpflug, University of Wisconsin-Madison at URBC, South Africa 2013. (Images taken by Tsi'tso Mokoena).
Technical Error Set Maize	[1]	50	Test set to determine the technical errors in imaging. The maize roots were placed with a supporting structure on the board to minimize placing errors.
Technical Error Set Bean	[1]	50	Test set to determine the technical error in imaging. The bean roots were placed without supporting

Table 7: continued

			structures on the board.
Rice grown in gellan gum	[1, 18, 25]	2406	Data produced and used by Iyer-Pascuzzi et al. 2010; Imaging and Analysis Platform for Automatic Phenotyping and Trait Ranking of Plant Root Systems;Plant Physiology. 152:1148-1157
Maize Validation Set	Unpublished	99	Maize validation data set collected in Ukulima at the Ukulima Root Biology Center, South Africa 2014.
Barley Diversity Panel	Unpublished	52	Barley Diversity Panel (192 accessions)
German and Australian barley varieties	[94]	64	A comparison study of root system of German and Australian barley varieties. The plants were grown in a semi-hydroponic phenotyping system (Ying L. Chen et al., 2011)
Tomato NILs	Unpublished	568	Solanum lycopersicum x pennellii introgression lines collected summer 2014 from Bradford Research station Columbia, Missouri
Maize Cadriano	Unpublished	20	F4 families of maize

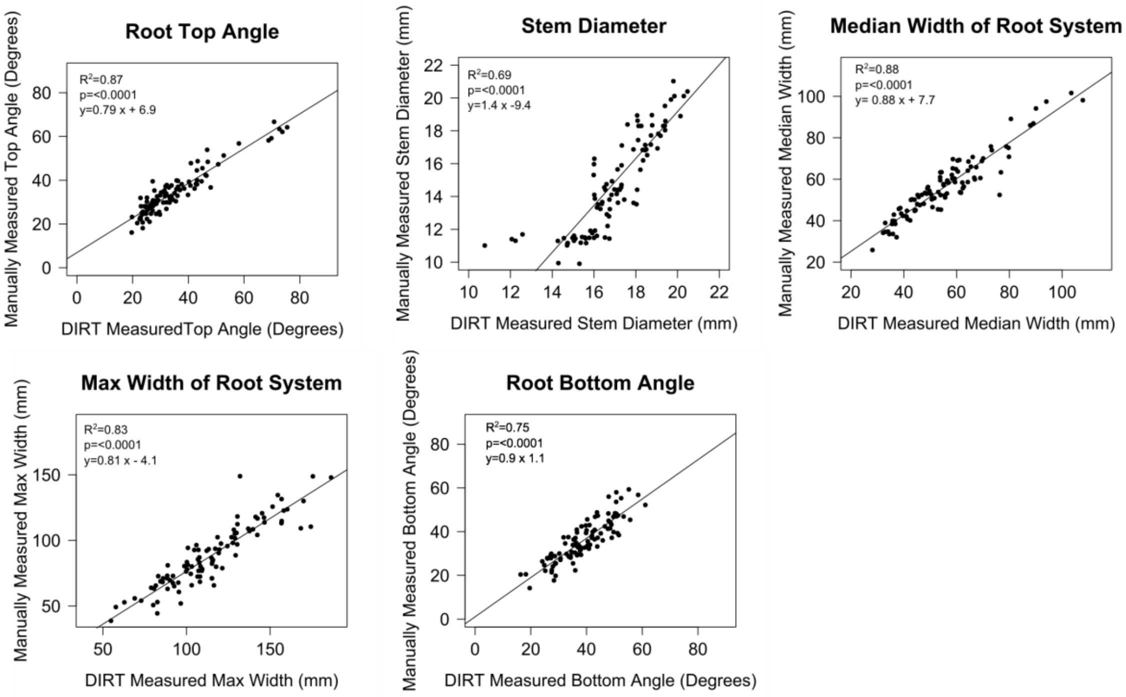


Figure 12: Validation of the DIRT root top angle, bottom angle, stem diameter, median width and maximum width for the public data set accessible at http://dirt.iplantcollaborative.org/content/maize-validation-set. The data set comprises of maize roots.

3.4 Acknowledgements

The authors of DIRT want to acknowledge and thank, Wesley Emeneker and Troy Hilley for providing the access and configuration for our test environment at the Georgia Institute of Technology. Furthermore we want to thank Andy Edmonds, Nirav Merchant and Martha Narro of the iPlant Collaborative for their computing support. This work was supported by the National Science Foundation (NSF) Plant Genome Research Program (grant nos. NSF0820624 to J.P.L. and J.S.W.), the Howard G. Buffett Foundation, and the Center for Data Analytics, Georgia Institute of Technology, Spatial Networks in Biology: Organizing and Analyzing the Structure of Distributed Biological Systems (to A.B. and J.S.W.). DIRT is powered, in part, by iPlant.

CHAPTER 4

SYSTEM SPECIFICATIONS AND DESIGN OF DIRT

This Chapter provides a high-level view of the system with the intent to give insights into the rationale behind the design of DIRT and to enable extensibility and sustainability of the platform. The purpose of this Chapter is to provide information to developers and users of DIRT to understand the basic components of the system, how they interface, how they can be extended, how a pipeline can be designed *de* novo, and how the system communicates with the high performance computing resources.

4.1 Background

Plant root systems are key drivers of plant function and yield [14]. They are also under-explored targets to meet global food and energy demands [33]. Many new technologies have been developed to characterize crop root system architecture (CRSA). These technologies have the potential to accelerate the progress in understanding the genetic control and environmental response of CRSA. To realize this potential, requires new methods and algorithms for analysis of image data of crop root systems (e.g. [23, 25, 27, 42]). Most prior approaches have focused on the estimation of root traits from images. Yet, no integrated platform exists that allows easy and intuitive access to trait extraction and analysis methods from images combined with storage solutions linked to heterogeneous metadata.

DIRT is a platform specifically designed to help estimate root traits from images of roots taken under field conditions. The analysis platform DIRT – described in Chapter

3 - enables field biologists to store, process and share their root image collections through a single web interface. Using this platform a user can organize images into data sets per experiment, run image processing algorithms on the datasets with different sets of parameters, view processed images and their estimated trait values. The user also has the option to download processed results for further analysis.

4.2 Requirement and design specifications

One of the important goals of DIRT is to provide a platform that does not involve any software installation or maintenance on the end users laptop or desktop computer. The platform should be available freely on public infrastructure for the science community. DIRT is designed and implemented to meet the following functional and systems specifications:

1. Storage of high volume of root images and its metadata.

   A key goal of the DIRT platform is to allow and enable plant biologists to store and manage root image data and its metadata without worrying about storage and computation requirements. The users should have full control of their data. The platform should have an intuitive user interface to organize and manage the root image data as collections. Overall the platform should address the following requirement specifications with respect to the end user:

   a. Computational storage requirements should be transparent to the user.
   b. Upload and store large (hundreds of images) number of images per batch.
   c. Associate and store metadata about each image.
   d. Organize root image as collections.
   e. Associate metadata to each collection.
   f. Add new metadata to both root images and collections.
   g. Control visibility and accessibility of the root image collections.
   h. Associate copyright and licensing information to a collection.

i. Manage user membership and their access rights on the root image collections.
j. Add new root images to collections.
k. Bulk add/update metadata of the images in a collection.
l. Download images from a collection.
m. Download metadata of images in a collection.
n. All the above mentioned functions should be available via easy to use user-interfaces.
o. The user should be able to perform all the above functions without installing software on their local computers.

2. The platform should allow the user to create private virtual collections by selecting root images from different physical collections. This functionality would allow the user to define an experiment by selecting root images taken from different field conditions.
3. Root system architecture (RSA) trait estimation is the core functionality of DIRT. RSA trait estimation is fully automated in DIRT by the underlying root image processing and trait estimation pipeline's parameters. One step in this pipeline is the image thresholding to clearly separate the pixels of the roots (subject of interest) from the image background. The current pipeline available on DIRT requires an input from the user as a value to this threshold parameter for automation. As there is no single standard threshold value that can be applied to all images, the platform should provide the functionality to segment one representative image (at a time) of a collection using a range of threshold values, so that the user can select the most appropriate one after visual inspection.
4. The DIRT platform should allow the users to compute, store and manage RSA traits from large number of images in high-throughput. Specifically the DIRT

platform should address the following specifications with respect to the RSA trait estimation and computation:

a. Full automation of RSA trait estimation.

b. High-throughput (100s and 1000s of images) RSA trait estimation.

c. Estimation of RSA traits of choice. The user should have option to choose RSA traits of interest.

d. Full control i.e., start and stop of RSA trait computation.

e. Email notification to the user on completion of the computation.

f. Storage and management of the output of RSA trait computation.

g. Export of the RSA traits in comma separated value (CSV) format and other standard formats like RSML.

h. Control accessibility and visibility of the computed output.

i. Manage user membership and their access rights on the computed output.

j. All the above functions should be performed via easy-to-use user interfaces without installation of any software on the user's local computers.

5. The DIRT platform should allow storage of different types of image data.
6. The DIRT platform should be extensible to incorporate new RSA trait computation pipelines.
7. The DIRT platform should be scalable within reasonable limits with respect to storage (10s to 100s TB of data) and computation (allocation of 1000s of processors at a time).
8. The source code of the DIRT platform should be made available to the science community.
9. The DIRT platform should be developed using an open-source and non-proprietary software stack.

10. The DIRT platform should be well documented for maintenance and future enhancements.

4.3 Software stack

Based on the DIRT platforms requirement and design specifications, we choose the following software stack to develop and build the application:

1. For the RSA trait estimation we choose the pipeline developed in Python [95] (described in Chapter 3) and ported it to grid computing infrastructure for high-throughput computation.
2. For user interfaces, user management, access control, data management, application workflow, user task scheduling and system's configuration, we choose Drupal [67], an open-source content management system (CMS).
3. For the high-throughput computation we choose STAMPEDE at TACC [96].
4. For scalable storage and public infrastructure we choose iPlant's cyber-infrastructure.
5. For communication between web application and grid computing we choose secure shell (SSH) and AGAVE API [86].

<u>Software requirements for the Python RSA trait computation pipeline</u>

The Python RSA trait computation pipeline is dependent on the following python modules. These modules need to be installed and made available to the system's user running web application on the web server machine as well as grid computing nodes:

1. Python version 2.7.x [95]

   Most of the Unix version of the operating systems comes with Python 2.x out of the box. If the system does not have Python, we suggest administrator to refer to the online version of the Python documentation at https://docs.python.org/2.7/using/unix.html#getting-and-installing-the-latestversion-of-python. Here we represent the steps to be taken to build and install Python 2.7.x from the source. Login to the system as a root user and follow these steps:

1. Download the source tarball as per your systems operating system from https://www.python.org/downloads/release/python-2710/ using following command

   ```
   wget https://www.python.org/ftp/python/2.7.10/Python-2.7.10.tgz
   ```

2. Extract the content of the tarball to a temp directory

   ```
   tar zxvf Python-2.7.10.tgz
   ```

3. Change directory to the extracted directory
4. Execute the following commands in sequence

   ```
   ./configure
   make
   make install
   ```

2. NumPy [97]

```
apt-get install python-numpy
```

3. SciPy [98]

```
apt-get install python-scipy
```

4. Mahotas [99]

```
easy_install install mahotas
                    OR
pip mahotas
```

5. PIL [100]

```
easy_install install pil

easy_install install pillow
```

6. Leptonica [101]

1. Download source
   ```
   wget http://www.leptonica.org/source/leptonica-1.72.tar.gz
   ```
2. Extract the source tarball
3. Go the extracted directory and run following command in sequence
   ```
   ./configure
   make
   make install
   ```

7. Tesseract [102]

   To install tesseract follow the instruction on the wiki at https://code.google.com/p/tesseract-ocr/wiki/Compiling. It depends on Leptonica besides other libraries. Make sure that the dependent libraries are installed. Even though it can be installed using apt-get, we suggest administrator to build and install it from the source. Follow these steps:

   1. Download source
wget https://tesseract-ocr.googlecode.com/files/tesseract-ocr-3.02.02.tar.gz
   2. Extract the source tarball
   3. Go the extracted directory and run following command in sequence
./autogen.sh
./configure
make
make install
ldconfig
   4. Install and configure the language data
      a. Download langugage data file (e.g. 'wget http://tesseract-ocr.googlecode.com/files/tesseract-ocr-3.01.eng.tar.gz' for 3.01 version)
      b. Decompress it ('tar xf tesseract-ocr-3.01.eng.tar.gz')
      c. Move it to installation of tessdata (e.g. 'mv tesseract-ocr/tessdata $TESSDATA_PREFIX' if defined TESSDATA_PREFIX)

8. ZBar [103]

   To install ZBar refer to the documentation at http://zbar.sourceforge.net/index.html. It depends on Imagemagick; make sure that it is installed on the system. We suggest administrator to build and install ZBar from the source. Similar to other library installations, download the source, extract the content, configure, build and install.

9. Graph Tool [104]

   To install graph-tool libraries refer to the documentation at https://graph-tool.skewed.de/download. Login to the system as a root user and run the following command.

   apt-get install python-graph-tool

Software requirements for Drupal

We choose Drupal version 7.x to develop the integrated DIRT platform to meet the systems design and requirement specifications. The detail documentation on Drupal 7's system requirements can be found on the Drupal website at https://www.drupal.org/requirements. The software includes:

1. Web server: Apache 2.x [68]
2. Database: MySQL 5.0.15 or higher with PDO [69]
3. PHP 5.2.5 or higher (5.4 or higher recommended) [70]

What is Drupal?

According to the book, Using Drupal [2], "Drupal is an open source content management system (CMS) being used by hundreds of thousands of organizations and individuals to build engaging, content-rich web-applications and websites. Once a web-application is built using Drupal, it can be maintained with online forms, and without having to change code manually. Drupal is free to use; it has an enormous library of constantly evolving tools that can be used to make a rich web-application."

How does Drupal work?

Drupal not only is a content management system but is also a content management framework made up of a software stack that looks like **Figure 13**. It behaves as a middleware that connects different software components of an enterprise application. Considering DIRT as an enterprise application, the Drupal stack connects and integrates the high-throughput computing environment, network file system, database server and the RSA trait computation pipeline. At a conceptual level it consists of components or modules developed in PHP language to access database and build dynamic content, manage access control, manage control flow between modules and render user interfaces. The modular framework of Drupal allows enabling and disabling modules on the fly to build a rich or lean web application. It being a web application, user requests are routed through the web server, which in-turn forwards it to the Drupal bootstrap process. The Drupal modular framework makes use of the *inversion of control* design pattern, thereby allowing the framework to call the appropriate module at the appropriate time. This is achieved by the *hooks* of the modules. These module *hooks* can be considered as *callbacks* or Drupal *events*. The *hooks* are created by function naming conventions rather than registering with a *listener*, so technically they are not truly called back. In other words the hooks allow a module to "hook onto" the things happening in the rest of Drupal. Another important aspect of Drupal is that it separates the data from the markup of the data, by the use of themes. There are two main steps involved while rendering a web page to the end user; (i) assembling the appropriate data, and (ii) marking up the data for the web browser. It is the theme layer in Drupal that marks up the data or creates the HTML to be presented to the web browser. The most important aspect of the Drupal framework is *node*. Every content presented by Drupal is derived from a single base type called *node*. This design feature of the framework allows it to be extended by the module developer to add any additional feature. This also allows the administrators to mix and match different nodes to build and new content type appropriate for their business requirements.

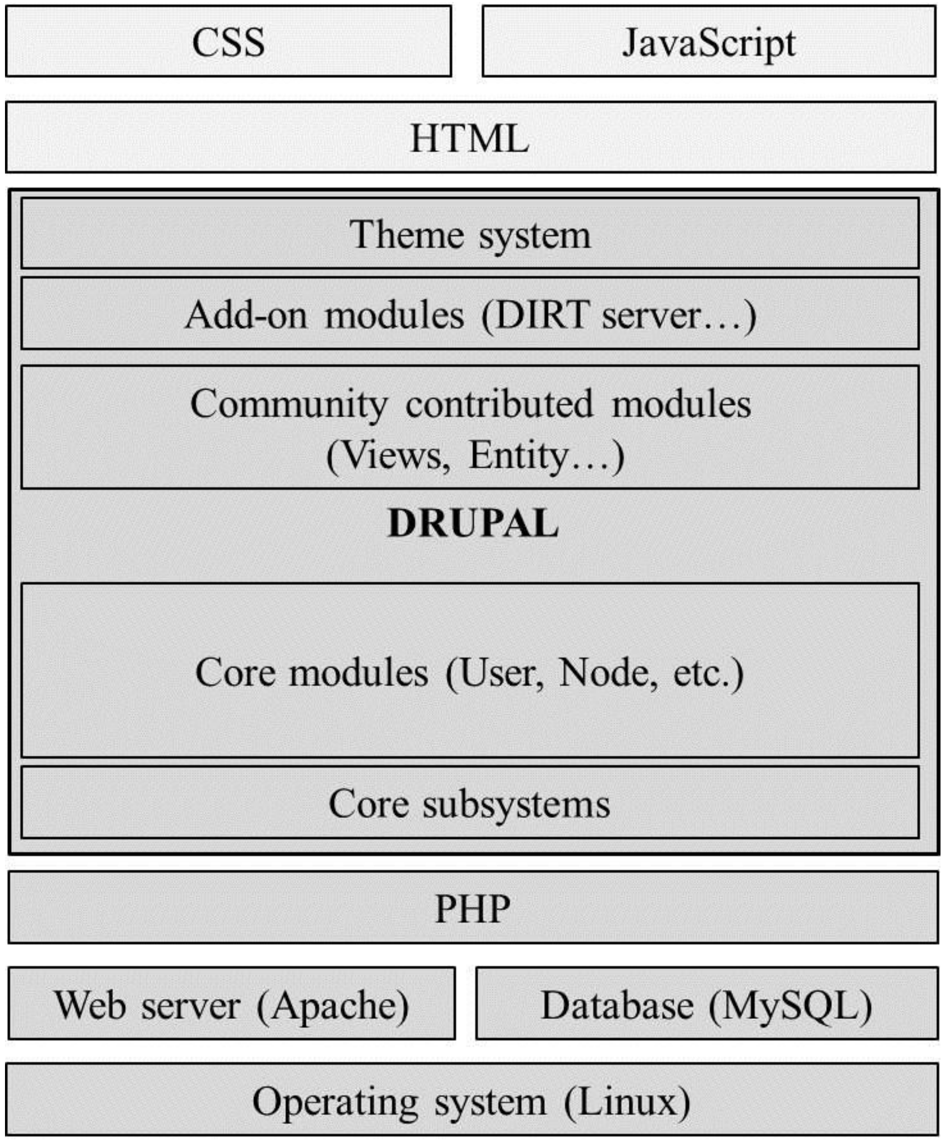


Figure 13: Software stack of Drupal content management system. The figure is adapted from the book "Using Drupal" [2].

Why did we choose Drupal?

We choose Drupal because it provides a number of features that can be easily adopted to meet DIRT platforms system's requirements. These include:

1. Modular framework

   Modules are the extensions or plugins that modify, extend, or enhance the functionality of Drupal. A module consists of a number of text files written in PHP scripting language. At the bare minimum a module requires two basic files.

The first file is the *.info* file that contains the name, description, version number, core compatibility and dependency with other modules. The second file is the *.module* file that contains the PHP code. Because of the modular framework we developed custom modules and plugged these in to meet our system requirements.

2. Theming system

   The theming system allows the administrator to modify the site's layout, color scheme and HTML markup. This also allows the separation of the handling of data from the presentation of the data. We modified the theming system to meet our system requirements.

3. Content structure

   Drupal handles its contents as nodes or content types. Because of this design it allows significant flexibility towards extending the functionality and creating new content types. This feature was very helpful in designing DIRT contents and its functionalities.

4. Search engine

   Drupal's core by default provides a search module, which can be easily configured. The search module automatically indexes its content types and users thereby making them automatically available to the search engine. The administrator can selectively include or exclude content types and users from the search engine.

5. Role-based access permissions

   Drupal's core provides a robust user management system that allows the administrator to define and configure content access permissions to easily manage a member-only web application. This system allows the administrator to create different user role types and grant permissions to roles based on the system requirements.

6. Triggers, Actions, Rules and Workflows

Drupal core and contributed modules provide trigger and action framework to build rules and workflows to meet system requirements. DIRT uses this feature to define custom rules and workflows to meet system requirements.

4.4 System design

The DIRT platform broadly consists of three tiers including storage, processing and user interface. For an overview of DIRT architecture refer to **Figure 6** in **Chapter 3**. In this section we present more fine-grained view of content model, their relationship, and design of DIRT workflows and DIRT add-on modules to achieve the platforms requirement specifications. Besides we also discuss the modules or scripts that reside outside of Drupal framework both on the Web server and grid computing environment to facilitate the high-throughput RSA trait computation.

Content model

This is the core of the DIRT platform that builds up the structure and framework to handle different attributes of the functional requirements including root images, collections, virtual collections, metadata, etc. The content model is designed using Drupal core and many community contributed modules. The content model also presents the relationship between different contents. With respect to object oriented design, it can be considered as the class association model. Because of the Drupal framework this model can be tweaked to meet future requirements using on-line forms and with minimal to no changes in code. Only the webmaster or the administrator of the DIRT platform has access rights to make changes to the content model. **Figure 14** represents the content model diagram of DIRT. This diagram only presents major attributes or fields of contents.

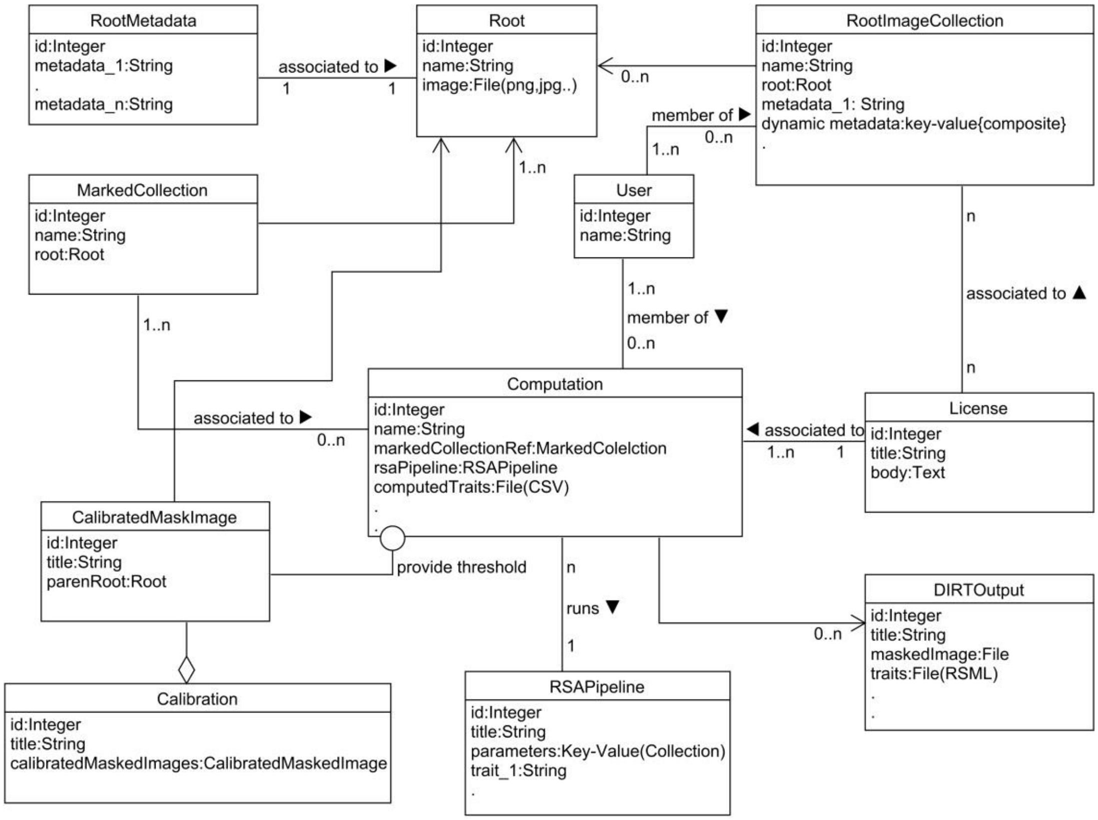


Figure 14: Content model or class association diagram of DIRT. This figure shows the custom content types and their relationship within DIRT. The entire functionality within DIRT is achieved by the use of these content types.

Each content type in Drupal has a set of common attributes which includes:

- NID (Node ID): Every node or content in the Drupal system has a unique ID irrespective of the content type.
- Title: Every node or content in the system is required to have a title.
- UID: Every node or content in the system is explicitly tied to its creator i.e. the user of the system who created it.
- Status: Every node or content in the system is always in either of the two states i.e. Published or Unpublished. This feature allows a node or content to be created and kept offline, until decided by a system workflow or user to take it online.

- Created and Changed: A timestamp that allows to record, when the content or node was created and updated.
- VID (Version ID): Every node or content in the system can maintain its version information. If enabled, all changes to a content or node is recorded and maintained.

Besides the common attributes, the DIRT custom content types have specific attributes to meet the system requirements. Here we briefly describe each of these content types:

- **Calibrated Mask Images**: This content type is designed to handle calibration functionality of DIRT. It has an image attribute to hold multiple masked images created during the calibration of a raw root image. It also has an attribute that refers to the raw root image in the system.
- **Computation**: As the name suggests, it defines a computation, e.g., RSA trait estimation performed on a collection of images via a pipeline. Therefore a computation has references to a Marked Collection, RSA trait computation pipeline, pipeline parameters and a suite of traits available in a pipeline. It has an attribute to define visibility of the computation, to mark it either private or public. A computation also has an attribute of type file that links to a CSV file containing computed RSA traits of all the images of the referred Marked Collection.
- **DIRT Output**: This content type defines the output produced for each raw root image by the RSA trait computation pipeline. It has attributes to refer to a computation and to a raw root image that has been processed. It contains an image attribute that holds the masked images. It has other attributes to hold the RSA trait values and the output RSML file.
- **Image Processing Pipeline**: It defines a RSA trait computation pipeline. It contains attributes that defines the parameters of the pipeline and the traits computed by the pipeline.

- **License**: This content type defines the Creative Commons license or any other license provided by the DIRT platform. It is associated to Computation and Root Image Collection content types.
- **Marked Collection**: As the name suggests, it defines a Marked Collection and has an attribute that refers to a list of Root Images.
- **Metadata**: It defines metadata of images in a collection. Therefore, it has an attribute that refers to a Root Image Collection and an attribute of type file that holds a pre-formatted CSV file.
- **Root**: This content type defines a root image that belongs to a Root Image Collection. Hence it has attributes to hold a root image and refer to a Root Image Collection. Besides it contains many other attributes to hold metadata related to the root image.
- **Root Image Collection**: This content type defines and represents a root image collection. It has attributes to define collection visibility, collection membership, collection license and different collection metadata.

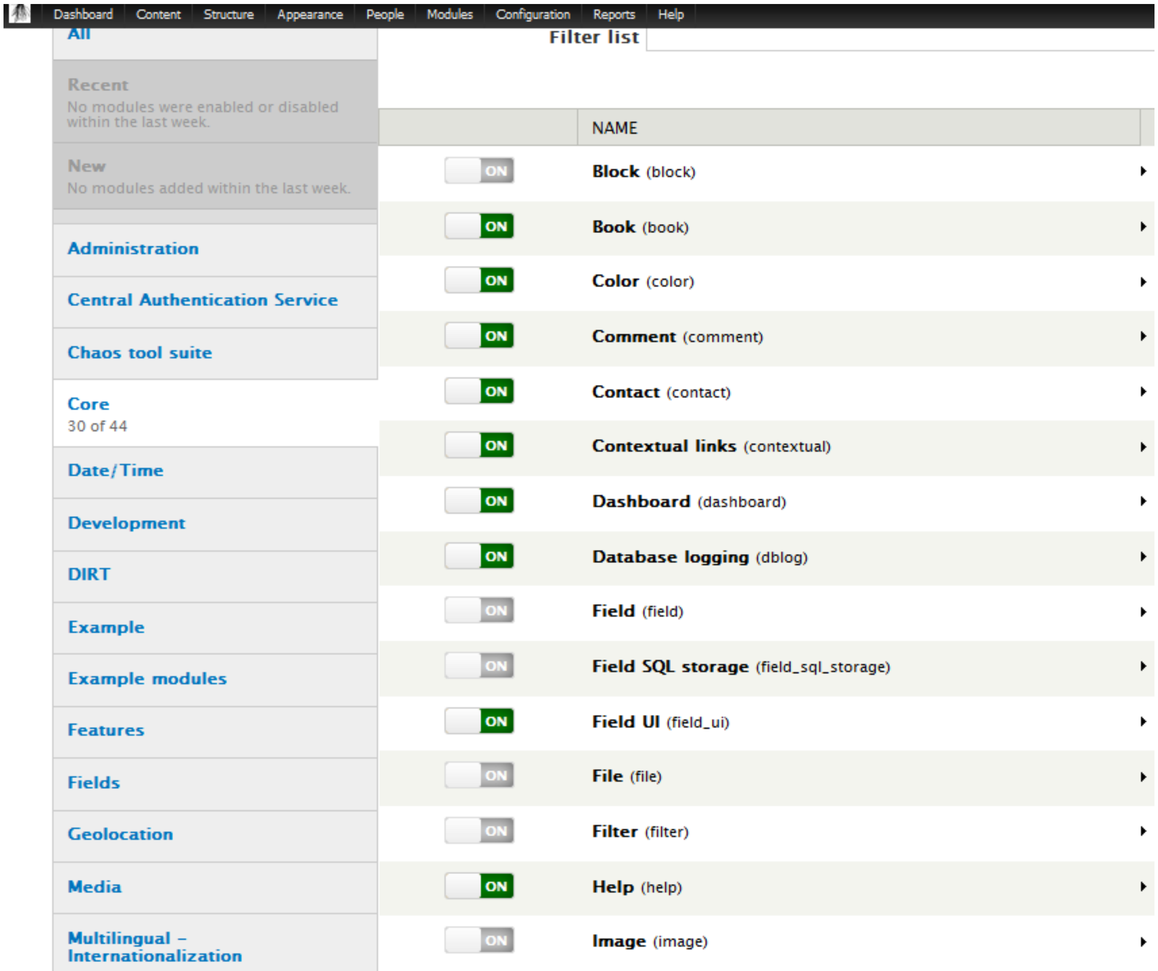
Each of the above custom content type in DIRT is themed individually to present the desired functionality defined in the systems specification.

Component model

The DIRT platform broadly consists of three major components:

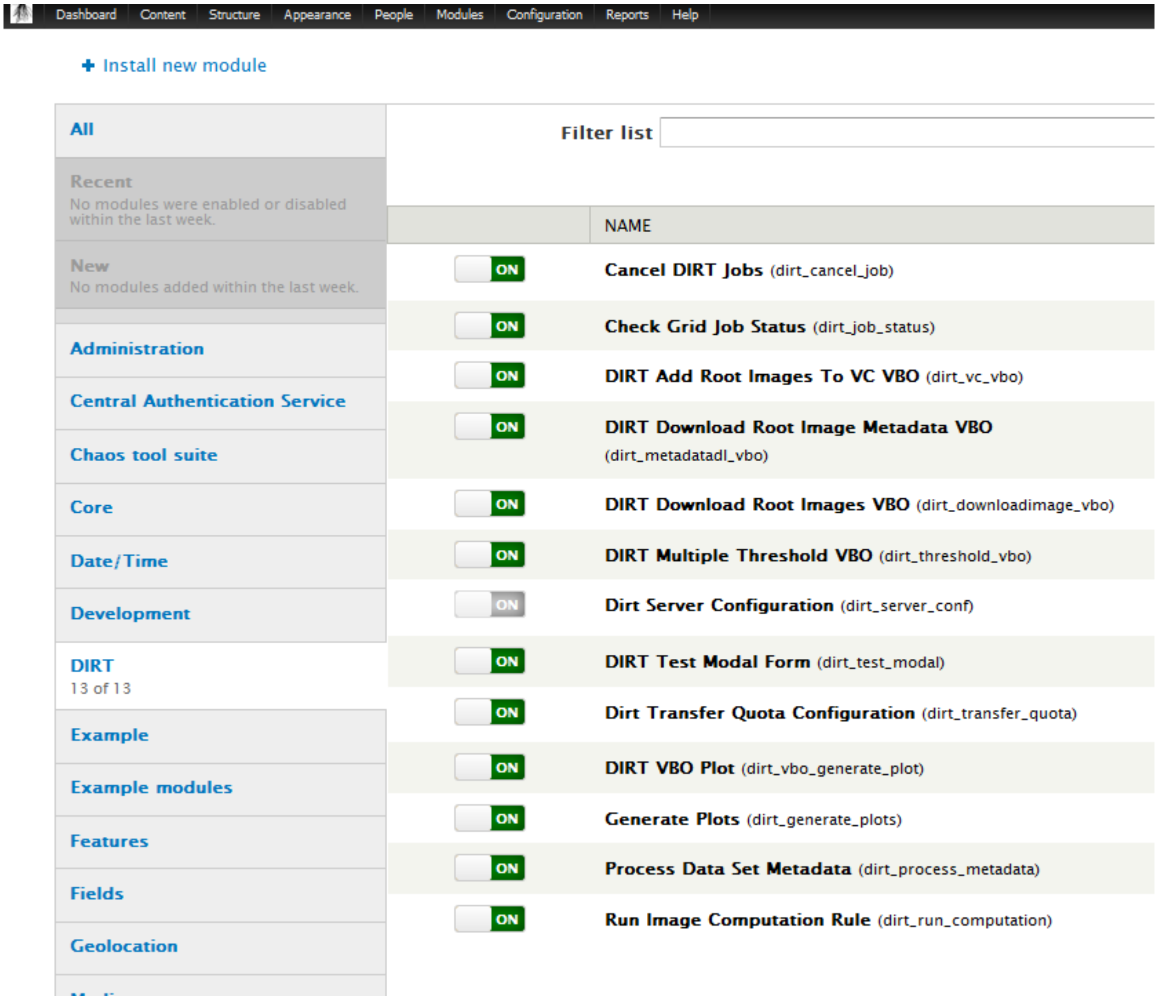
1. Web server components

   These are the Drupal components including core, community contributed and custom DIRT modules that orchestrates the whole platform in cohort. The content model described in the previous section is designed and implemented using these modules. **Figure 15** shows some of the core modules and **Figure 16** shows the list of DIRT modules that are deployed and enabled to build this platform along with many others.



	NAME
ON	**Block** (block)
ON	**Book** (book)
ON	**Color** (color)
ON	**Comment** (comment)
ON	**Contact** (contact)
ON	**Contextual links** (contextual)
ON	**Dashboard** (dashboard)
ON	**Database logging** (dblog)
ON	**Field** (field)
ON	**Field SQL storage** (field_sql_storage)
ON	**Field UI** (field_ui)
ON	**File** (file)
ON	**Filter** (filter)
ON	**Help** (help)
ON	**Image** (image)

Figure 15: Screenshot of DIRT module administration page showing a portion of the Drupal core modules.



	NAME
ON	**Cancel DIRT Jobs** (dirt_cancel_job)
ON	**Check Grid Job Status** (dirt_job_status)
ON	**DIRT Add Root Images To VC VBO** (dirt_vc_vbo)
ON	**DIRT Download Root Image Metadata VBO** (dirt_metadatadl_vbo)
ON	**DIRT Download Root Images VBO** (dirt_downloadimage_vbo)
ON	**DIRT Multiple Threshold VBO** (dirt_threshold_vbo)
ON	**Dirt Server Configuration** (dirt_server_conf)
ON	**DIRT Test Modal Form** (dirt_test_modal)
ON	**Dirt Transfer Quota Configuration** (dirt_transfer_quota)
ON	**DIRT VBO Plot** (dirt_vbo_generate_plot)
ON	**Generate Plots** (dirt_generate_plots)
ON	**Process Data Set Metadata** (dirt_process_metadata)
ON	**Run Image Computation Rule** (dirt_run_computation)

Figure 16: Screenshot of DIRT module administration page showing the list of custom DIRT modules.

2. RSA trait computation components

   These are the trait computation pipeline Python scripts that are deployed both on the web server and grid computing node to meet the calibration and trait computation system specifications respectively.

3. Interface components

   These are the shell scripts that reside on the web server and grid computing node to interface with the grid job scheduler.

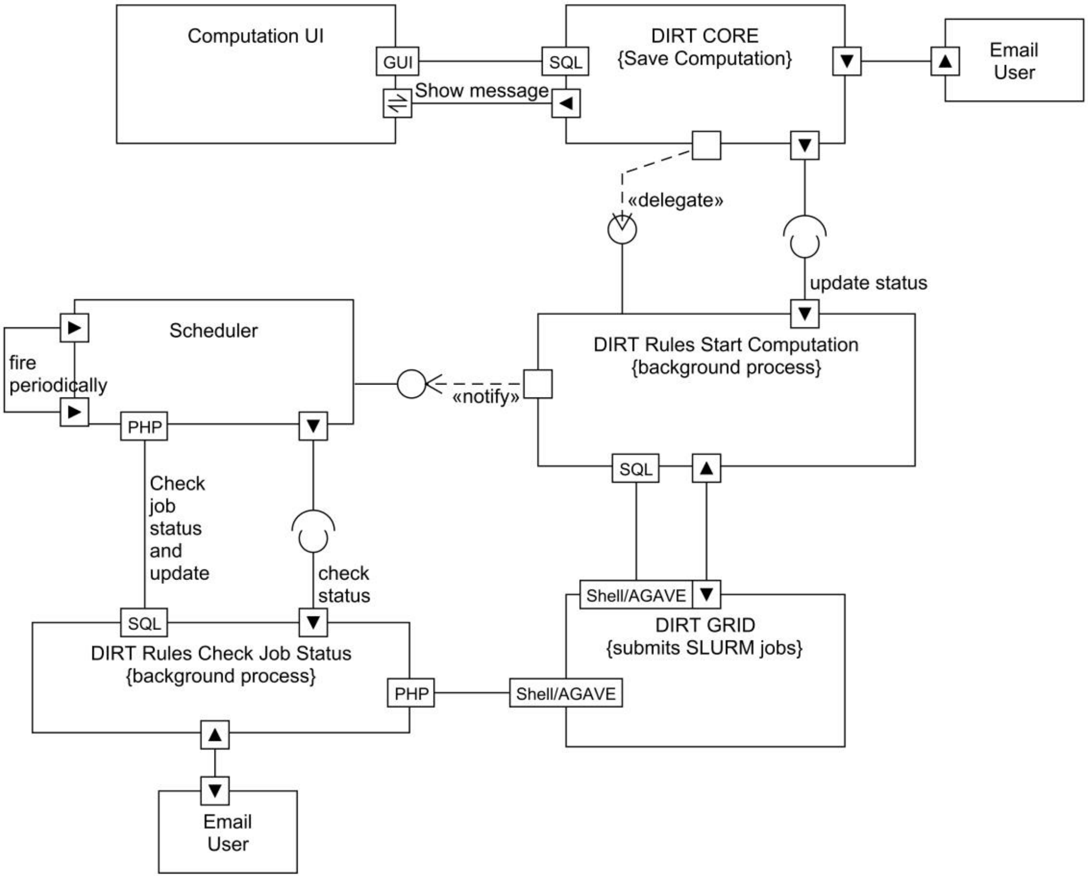


Figure 17: Component diagram showing the components involved in RSA trait computation process on the DIRT platform

By virtue of Drupal architecture, DIRT is modular and every process in DIRT involves many components or modules. In **Figure 17** we present the components and their interactions in DIRT for the RSA trait computation process. The computation process in DIRT involves the user interface component, rules component, workflow component, custom DIRT components and core components. The processes starts with the user saving a new content of type “Computation”, wherein the user provides a computation name, selects a “Marked Collection”, selects a RSA trait computation pipeline, provides pipeline parameters and selects traits. On save of the computation the rules engine is notified to trigger two DIRT workflows; (i) start DIRT job submission module in background to run the RSA pipeline on the grid-computing environment, get the grid job details, update the database, change computation status and notify the user. (ii) schedule the DIRT job status check module to run in background in every 10 minutes (until job completion or termination) to ping the grid for job status, check job status, get computed output to the web server, update the database, create DIRT output contents and notify the user. Each step in these workflows in turn uses or invokes many other sub-modules or components located across different nodes of the platform to complete the process.

Deployment model

The deployment model depicts a static view of the run-time configuration of the processing nodes and all the components that run on these nodes. In this section (**Figure 18**) we present the distribution and location of all components of DIRT across different physical/virtual nodes.

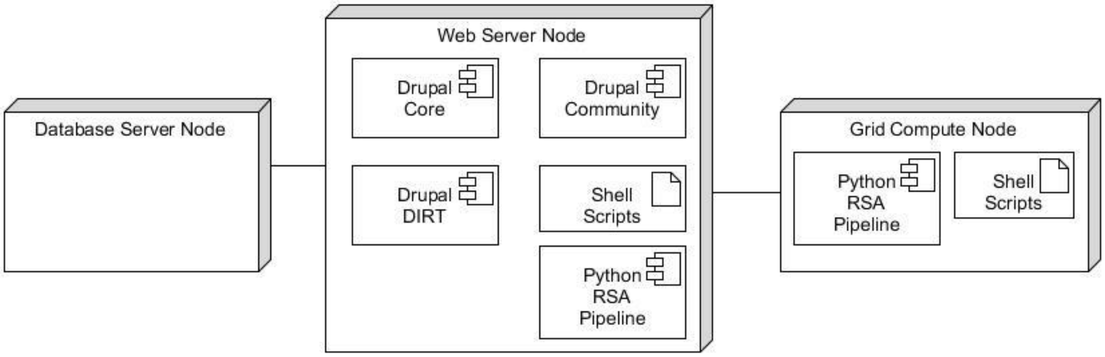


Figure 18: Deployment diagram of nodes and components of DIRT platform

4.5 Implementation

In this section we describe in detail how the design specifications were implemented to achieve the functional requirement specifications listed in previous section. As part of the implementation process we describe the configuration of different nodes, deployment of components onto these nodes, function of custom DIRT modules, how to extend DIRT to use new trait computation pipeline and how to configure and administer DIRT using Drupal administration interfaces.

Configuration of Database server node

MySQL database was installed and the appropriate user (i.e. database user) and database were created for DIRT. We recommend that the system administrator follow MySQL and Drupal installation instructions/guides to configure a database.

Configuration of Web server node

The node hosting web server is configured with PHP, Apache and Python to run Drupal and trait computation pipeline. Besides all PHP libraries (i.e. bz2, gmp, zlib, gd, libjpeg, libtiff, libpng3, libxml, t1lib5, libmcrypt, libmhash, wget, bzip2, ming, pdflib, php-bcmath, date, dba, dom, json, ssh, scp, mysql etc.) File_Archive libraries are also installed. After PHP installation, following configuration were updated in php.ini file:

- max_execution_time=120
- memory_limit=512M
- post_max_size=1024M
- upload_max_filesize = 10242M
- max_file_uploads = 20

After installation of Python and other dependent python modules (listed in earlier), make sure that these libraries are accessible to all the users of the system.

Directory structure as shown in **Figure 19** is created to host the contents of DIRT modules. Write permission is granted to all on the ‘temp’ directory. The ‘logs’ directory will be used to house web server access and error log files. The ‘public_html’ directory will be used to house all the web server contents including all Drupal contents. The ‘calibrate’ directory under ‘scripts’ will house trait computation python scripts. The ‘job_status’ directory will house shell scripts to process trait computation outputs.

Drush [105] libraries are also installed on this node to automate the Drupal backup and upgrade process.

Apache web server is configured with an appropriate virtual host to serve content from the ‘public_html’ directory and write log to the ‘logs’ directory.

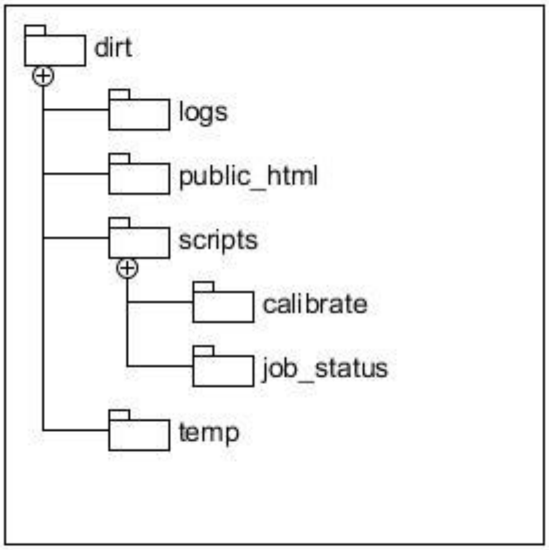


Figure 19: Directory structure on DIRT web server node

Configuration of Grid computing node

As mentioned earlier besides other requirements DIRT also computes RSA traits in high-throughput. After getting an account on the TACC’s STAMPEDE environment we configured it with required python libraries to run the pipeline. The directory structure as shown in **Figure 20** is created on the grid node to house python and shell scripts. The ‘data’ and ‘out’ directory under $WORK (working directory) will house the raw image data and output data from the pipeline during and after trait computation.

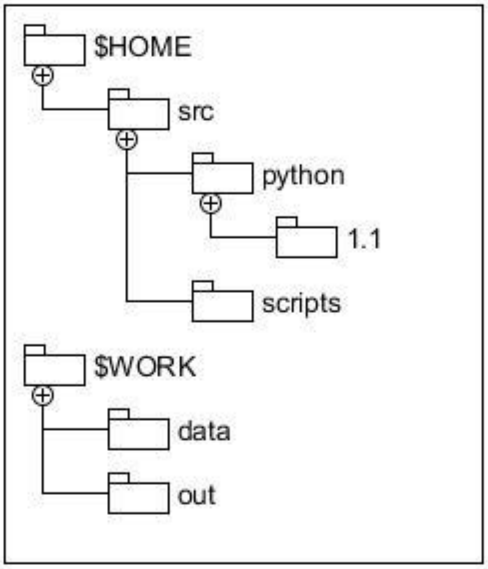


Figure 20: Directory structure on DIRT grid node

Deployment of database

For the current DIRT installation, the database is seeded and populated during configuration and module installation. But this instruction can be used to install and setup and instance of DIRT on local/private proprietary environments. To deploy the database; (a) download the DIRT source archive from the DIRT website at http://dirt.iplantcollaborative.org/sites/default/dirt_files/dirt-src.tar to the database server node, (b) extract the archive and (c) load the dirt.sql file to the database created earlier.

Deployment on Web server

For current DIRT installation, web server components were developed de novo as part of the configuration process. The following instructions are for users wishing to install and setup an instance of DIRT on their local/private proprietary environment. Download the DIRT source archive from the DIRT website at http://dirt.iplantcollaborative.org/sites/default/dirt_files/dirt-src.tar to the web server node, (b) extract the archive to the 'public_html' directory created earlier and (c) edit the ‘settings.php’ file under ‘public_html/sites/default’ directory and update the database configuration setting parameter appropriate to your installation.

Download and extract the web server scripts from the site at http://dirt.iplantcollaborative.org/sites/default/dirt_files/dirt-web-scripts.tar to the ‘scripts’ directory created earlier. Make sure that these scripts have execution rights and ‘calibrate’ directory contains python scripts and ‘job_status’ directory contains following shell scripts:

- dirt_clean.sh
- dirt_extract.sh
- dirt_read_log.sh

DIRT web server schedules the ‘dirt_job_status’ rules modules via Cron to check the status of the active computations on the grid every 10 minutes. The web server administrator should setup a Cron job on the web server to trigger every 10 minutes with the following command:

```
10 * * * * /usr/bin/wget -O - -q -t 1
http://<your_domain_url>/cron.php?cron_key=<your_cron_key>
```

Make sure to change the domain URL and cron key as per your environment.

Deployment on grid node

Download and extract the archive file from the website at http://dirt.iplantcollaborative.org/sites/default/dirt_files/dirt-server-scripts.tar to the grid node’s home directory. It will create and copy the files to appropriate directories as mentioned in the configuration section. Make sure that the ‘src/python/1.1’ directory contains the python scripts and ‘src/scripts’ directory contains following shell scripts:

- cancel_jobs.sh
- clean_output.sh
- prepare_data.sh

- check_job_status.sh
- dirt_job.slurm
- prepare_output.sh
- clean_all.sh
- submit_dirt_jobs.sh

Modify these scripts to meet your grid job scheduler's commands.

Configuration and administration of DIRT

After successful configuration and deployment modules on respective nodes, start your database and Apache server. Access your instance's URL (e.g., Apache virtual hosts name) in a browser and login with your administrator credentials. As you have setup the instance from the existing DIRT source, all the required modules, contents, rules, workflows, themes, views, etc. should have already been enabled for you. Even though it has already been setup, in this section we will describe in detail the functioning of all custom DIRT modules and some important community contributed modules whose source code were updated to meet DIRT specifications. But we recommend you to familiarize yourself with all the enabled Drupal modules in DIRT.

DIRT theme

The DIRT platform has adopted and used a Drupal community contributed module called 'Danland' to theme its user interfaces. Detailed information about this module can be found on the Drupal website at https://www.drupal.org/project/danland. We have tweaked its source code and customized it to provide the current user interfaces. This module can be administered from the DIRT's 'Appearance' administration menu. This module resides on the web server node under 'public_html/sites/all/themes/danland/' directory. Its CSS and page template files have been modified for the current look and feel. We have also added a custom JavaScript file called ‘jquery.dirt.js’ to alter validation on computation form. Therefore, this module should not be auto updated to a new version by the webmaster or the administrator.

DIRT content model

After basic core setup and before any custom module development and configuration, the content model for the DIRT platform (as mentioned earlier) must be defined and configured to handle system requirement specifications. In Drupal terms, the ‘Content Types’ (analogous to a class in object oriented concept) must be defined. The DIRT content model depends on many Drupal modules (like *Entity, Entity Reference, Organic Groups, etc.*). These modules must be deployed and configured before defining new custom content types. All these modules reside on the web server node at ‘public_html/sites/all/modules’ directory and they can be accessed via administrative ‘Modules’ menu item. All custom DIRT content types can be accessed and managed via administrative ‘Structure > Content Types’ menu item.

Batch upload of root image to a collection

To enable batch upload of images to a collection DIRT has used and adopted a community contributed module called ‘bulk_media_upload’. Detailed information about this module can be found on the Drupal website. This module resides on the web server node under ‘public_html/sites/all/modules/bulk_media_upload’ directory. The source code of this module has been modified to meet DIRT specifications. This module should not be auto upgraded. This module can be configured by the administrator by visiting the ‘Media/Bulk Media Upload Settings’ page under ‘Configurations’ menu.

DIRT grid server configuration

This is a custom DIRT administration module that resides on the web server node at ‘public_html/sites/all/modules/dirt_server_conf’ directory. This enables storage of grid computing server details and web server location details to be used by other modules like grid job submission module across the system. In other words, other modules of the platform depend on this and dependent modules cannot be deployed and/or enabled without this. Therefore the administrator should first configure and save server configuration details by accessing the ‘Configuration > DIRT Server Configuration’ menu item.

DIRT user quota configuration

Similar to the previous module, this is also a custom DIRT administration module that resides on the web server node at ‘public_html/sites/all/modules/dirt_transfer_quota’ directory. As the name suggests it allows storing and parameterizing the daily file transfer quota for a user from web server node to the grid computing node. It can be accessed by the administrator via ‘Configuration > DIRT Quota Configuration’ menu item. Besides daily transfer quota DIRT also has option to configure total file storage quota for a user. This quota can be set per each user role by visiting the ‘People > Disk Quota’ administration menu item.

DIRT grid job submission

This is a custom DIRT rules module that depends on ‘dirt_server_conf’ and ‘rules’ module to handle RSA trait estimation and computation on the configured grid node as an asynchronous process. This module is available on the web server node at ‘public_html/sites/all/modules/dirt_run_computation’ directory. This module is used to define a DIRT workflow rule, that is fired on creation or save of a new content of type ‘Computation’. On start of the workflow rule, this module starts a background process, that reads details from the computation object, prepares an archive of the image data along with pipeline parameters and selected traits, estimates the average grid computation wall time and processor requirements based on the image size and count, receives a secure connection with the grid node, transfers the content to the grid node, executes scripts on the grid node to extract and prepare the data for computation and executes scheduler job submission script, wait for it to be submitted, gets the grid job identifier, updates the database and notifies the user. DIRT employs many workflow rules to meet its requirement specification. All these workflow rules can be accessed and configured by visiting the ‘Configuration > Workflow > Rules’ administrative menu item.

DIRT grid job status check

Similar to the previous module, it is also a custom DIRT rules module that depends on ‘dirt_server_conf’ and ‘rules’ module. This module is responsible for tracking a grid job and bringing the trait output from the grid node to the web server node, updating the database and notifying the user on job status. This workflow rule is also triggered on creation of a new content of type ‘Computation’. Unlike the previous module, it is configured and scheduled to run every 10 minutes until the grid job completion. This module uses scripts on web server nodes and grid server nodes to perform its tasks. This module resides on the web server node at ‘public_html/sites/all/modules/dirt_job_status’ directory. Similar to the previous module it can be accessed and configured via ‘Rules’ administrative menu item.

DIRT calibrate threshold

This is a custom DIRT module that enables image threshold calibration. It depends on ‘dirt_server_conf’ and ‘views_bulk_operation’ (a community contributed module). It resides on the web server node at ‘public_html/sites/all/modules/dirt_threshold_vbo’ directory and uses RSA trait estimation pipeline components on the web server node.

DIRT marked collection

This module is responsible for creating/adding images to a ‘Marked Collection’ from a ‘Collection’. It resides on the web server node at ‘public_html/sites/all/modules/dirt_vc_vbo’ directory and depends on ‘views_bulk_operation’ module.

DIRT metadata upload and download

Metadata upload to a ‘Collection’ and download from a ‘Collection’ is handled by two different modules called ‘dirt_process_metadata’ and ‘dirt_metadatadl_vbo’ respectively. Both these modules reside on the web server node. Metadata upload module is a rules module that resides at ‘public_html/sites/all/modules/dirt_process_metadata’ directory and depends on ‘rules’ and ‘background_process’ community contributed module. This module is triggered on upload of a metadata file to a collection. Metadata download module resides at ‘public_html/sites/all/modules/dirt_metadatadl_vbo’ directory and depends on ‘views_bulk_operation’ module.

DIRT grid job cancellation

This module is responsible for cancelling a submitted or a running job on the grid environment. It resides on the web server node at ‘public_html/sites/all/modules/dirt_cancel_job’ directory and depends on ‘dirt_server_conf’ module.

DIRT bulk image download

Like metadata download module, this is a custom DIRT module responsible for downloading images as an archive (zip) file from the ‘Collection’. It depends on ‘views_bulk_operation’ module and resides at ‘public_html/sites/all/modules/dirt_download_image_vbo’ directory on the web server node.

Extending DIRT and adding a new pipeline

The DIRT platform design enables it to be extended to add new RSA trait computation pipelines for existing traits or new traits. Eligible pipelines for inclusion must be fully automated; that is they do not require any human intervention during its execution. But this extension cannot be done via user interfaces alone, rather it requires updates to the source code for the following nodes of the DIRT platform:

- Grid node
    1. The new pipeline must be manually installed and configured on the grid computing environment.
    2. New scheduler specific job submission scripts must be created to handle its parameters.
- Web server node
    1. A new content of type ‘Image Processing Pipeline’ must be added via the administrative ‘Content > Add content > Image Processing Pipeline’ menu item.
    2. If the new pipeline introduces traits, these traits must be added to the ‘Computation’ and ‘DIRT Output’ content types.
    3. If the pipeline requires any specific parameters, those fields must be added to the ‘Computation’ content type.
    4. User interface validation scripts must be added to ‘Computation’ forms to show parameter fields with respect to a pipeline.
    5. The ‘dirt_run_computation’ and dirt_job_status’ modules must be updated to handle new pipeline, traits and pipeline parameter fields.
    6. If the pipeline generates output in any proprietary format, ‘dirt_job_status’ module needs to be updated to process the pipeline output file types.

4.6 Discussion

The DIRT platform is designed and built using a popular, robust, modular and open-source content management system/framework. Its modular framework allows us to adapt it to new or change functional requirements with minor structural changes. The platform is deployed and uses state of the art public infrastructure (iPlant cyber infrastructure) and high-performance computing (STAMPEDE on TACC) resources available in the country. The platform can easily be scaled to high volume of storage and high-throughput computation.

CHAPTER 5

CONCLUSION: COMPUTATIONAL TOOLS ENABLING DISCOVERY IN THE PLANT SCIENCES

This thesis presents a set of tools developed to enable the analysis of plant networks, including venation networks of leaves and root systems. Specifically, the tools help the plant science community to store, manage, share and collaborate, and analyze digital images of leaves and roots. First, we have developed an online database of cleared leaf images for the community, interested in studying and analyzing vein network structure of the plant leaves. Next, we have also developed an online high-throughput computational platform for the plant science community interested in studying root system architecture of crop plants. These tools have culminated in the following journal publications and manuscripts:

1. Das A, Bucksch A, Price CA, Weitz JS; ClearedLeavesDB: an online database of cleared plant leaf images.; Plant Methods 2014, 10:8.

2. Bucksch A, Burridge J, York LM, Das A, Noord E, Weitz JS, Lynch JP; Image-based high-throughput field phenotyping of crop roots.; Plant Physiology, 2014; 166:470-486.

3. Das A, Schneider H, Burridge J, Martinez Ascanio AK, Wojciechowski T, Topp CN, Lynch JP, Weitz JS, Bucksch A; DIRT: a high-throughput computing and collaboration platform for field-based plant phenomics.; 2015, In Review.

5.1 Summary of the tools developed

ClearedLeavesDB

ClearedLeavesDB is an online database of cleared leaf images hosted on iPlant cyber-infrastructure and is accessible at http://www.clearedleavesdb.org. It enables plant scientists around the world to store, annotate, search and share cleared leaf image collections through a single web interface. Its unique features can be categorized into two groups: community features and technological features. The community features allow its users to (i) mark an image collection - private or public,(ii) share image collections with a trusted user community, (iii) post comments on images and its meta-data, (iv) flag images for review, and (v) enables non-scientists with important data to contribute to the scientific community via a common trusted platform. The technological feature allows its users to (i) store and manage cleared leaf images in a database, (ii) manage access control of the collections via simple user interfaces, (iii) bulk upload and download images from the collection, (iv) search for images in the collection with specific metadata, and (v) use an open application programming interface (API) to upload processed images and its output from any proprietary image processing platform.

DIRT

Digital Imaging of Root Traits (DIRT) is an online automated high-throughput computing and collaboration platform for field based crop root phenomics. It is also hosted on iPlant cyber-infrastructure, uses TACC's high-throughput grid computing platform and is accessible at http://dirt.iplantcollaborative.org/. It is a high volume central depository and high-throughput RSA trait computation platform for the plant scientists working on crop plants. It enables scientists to store, manage and share crop root image collections with metadata and compute RSA traits from thousands of images in parallel. It makes high-throughput RSA trait computation available to the community just a few button clicks away. It enables plant scientists to spend more time on science rather than on technology. It makes a large variety of crop root image data, its metadata and corresponding RSA trait outputs easily accessible to the public and broader scientific community. We hope this easy accessibility of data will attract new tool developers, spur creative data usage that may even be applied to other fields of science.

5.2 Summary of the usage of the tools

ClearedLeavesDB

Since its release on March 2014, it has been widely used by plant scientists, students, researchers around the world. The article was the editors pick for the publication on Plant Methods and has been denoted as a “Highly Accessed” article. The following table provides some statistics of the usage of the database as of July 30, 2015.

Table 8: Statistics on ClearedLeavesDB as of July 30, 2015

Feature name	Number
Registered users	37
Total visitors	1,218,039
Unique visitors	19,195
Accessed from different countries	81
Total number of images downloaded	> 50,000
Total number of public images	20,528
Images from different countries	145
Number of plant families	256
Total size of images	43 GB

DIRT

DIRT was hosted on the public platform i.e. iPlant cyber-infrastructure two months ago and the related article is in review. Nonetheless, it has been in development and use in Georgia Tech's infrastructure since mid of 2014. In fact it was used for our Plant Physiology paper listed above to show that the new and previously inaccessible RSA traits computed by the integrated pipeline are heritable (**Figure 21**).

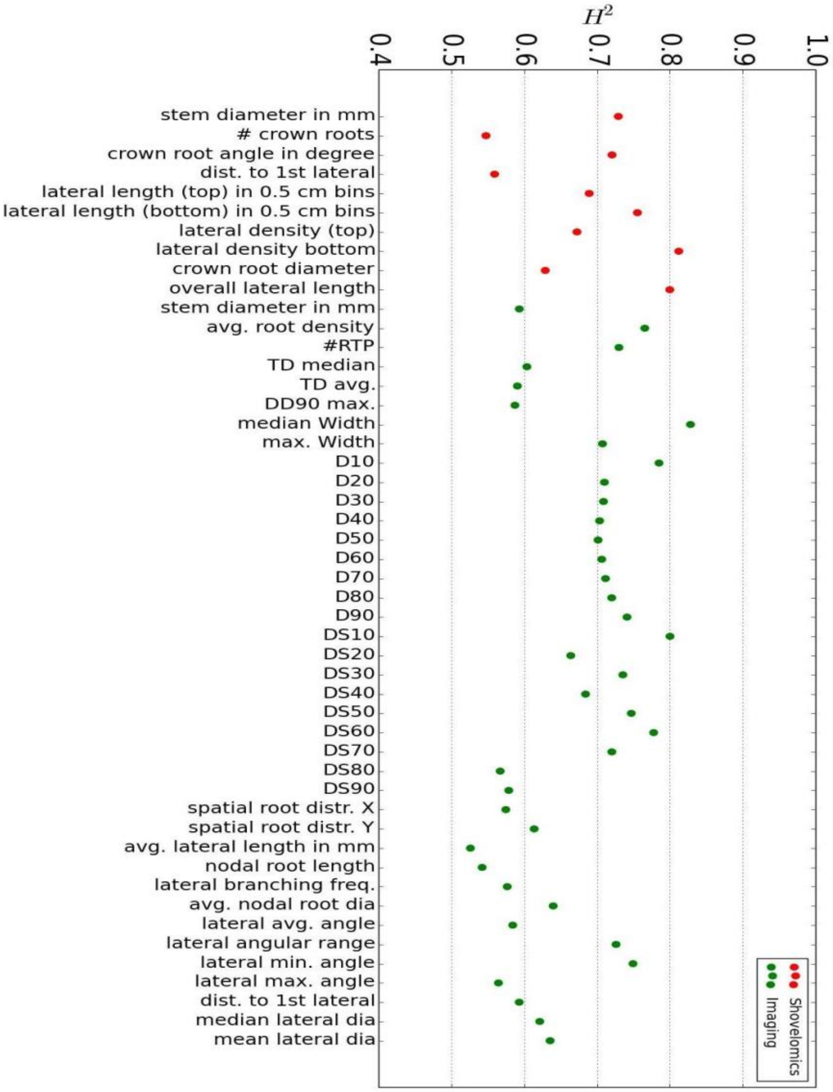
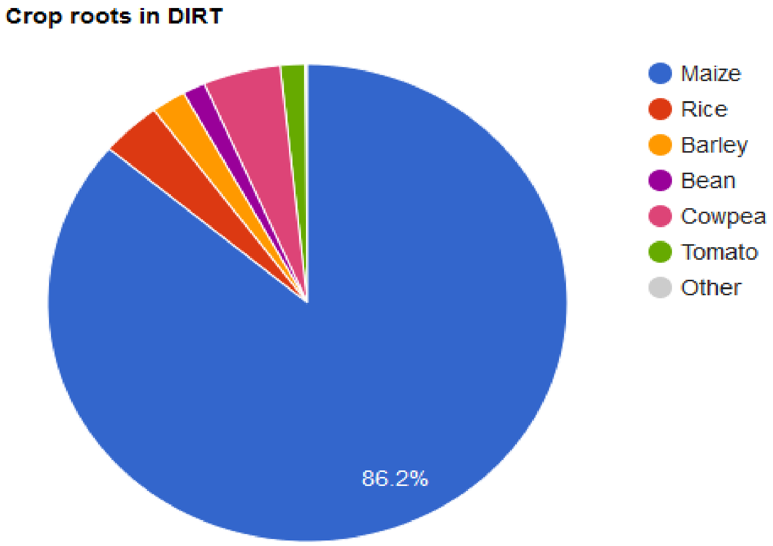


Figure 21: Broad sense heritability of maize traits derived from the crown root images. The figure is reprinted from "Bucksch A, Burridge J, York LM, Das A, Noord E, Weitz JS, Lynch JP; Image-based high-throughput field phenotyping of crop roots.; Plant Physiology, 2014; 166:470-486".

Usage statistics of the first version of DIRT on Georgia Tech's infrastructure by our collaborators are as follows:

- Total number of datasets: 200

- Total number of root images: 67,500
- Total volume of images: 150 GB
- Crop types:



- Total number of images or jobs processed: 250,000
- Maximum number of images or jobs processed in parallel: 6000

After installation of the new version of DIRT on iPlant cyber-infrastructure on May 2015, its usage has also gone up quite fast. As of July 30, 2015, it has more than 10,000 root images amounting more than 30 GB. It has been actively used by members from the following institutions:

- Penn State, USA
- Danforth Plant Science Center, St.Louis, USA
- Forschungszentrum Juelich IBG-2 Pflanzenwissenschaften, Germany
- University of Bologna, Italy
- Madihol University, Thailand

- Duke University, USA
- Greenville College, USA
- University of Bonn, Germany
- Federal University of Goias , Brazil
- The University of Western Australia, Australia
- Cinvestav, Mexico

5.3 Conclusion and future direction

In this study we presented two computational tools to manage and analyze plant leaf images and crop root images. This study advances the availability of computational tools in the field of plant phenomics. It also makes public datasets available to the plant science community on a single integrated platform via simple web interfaces. In particular, DIRT enables access to high-throughput computation to estimate RSA traits from 2D root images taken from field conditions. As of this release, DIRT handles heterogeneous 2D image data, but it can be extended to handle 3D image data and new 3D image processing pipeline can as well be plugged in to estimate RSA traits. Even though 3D imaging systems are not yet widely accessible to the plant phenomics community, the platform designers and developers should make provision to extend their platform to handle 3D data, because correct mapping of phenotype to genotype will require the precision provided by 3D imaging. Many studies [42, 82] have been done to measure traits from 3D images. One study [82] even provides evidence of QTL on rice genome controlling growth of RSA. The authors of this study have used many 3D traits in their analysis, but have not directly measured 3D root angle. But we have seen in our previous study [1] that root angle traits are heritable.

Root angle is considered to reflect the extent to which roots exhibit gravitropism. Gravitropism in roots has fascinated many scientists, including Charles Darwin [106]. Studies on gravitropism over more than last 100 years have made great strides in accumulating information and our understanding of the process [107]. Previous studies [14, 35] have reported that root gravitropism is one of the important factors that determine root architecture, and in turn affects spatial distribution of root system in the soil and consequently the nutrient uptake efficiency. Root gravitropism can be quantified via the measurement of root tip angle [35]. Though much of the gravitropic mechanism have been understood with respect to molecular and physiological processes [36-39], considerable details are yet to be understood with regards to other important aspects such as dependence of gravitropic response to environmental conditions or developmental age. Earlier studies [17] have demonstrated that gravitropic response is highly plastic. Previous studies have either manually measured the root initiation angle or used 2D images for automated measurement. As mentioned in previous work [42], 3D information of root architecture can be used to enhance estimation of these traits; we are using 3D information of whole reconstructed rice roots grown in gel medium [82], to measure root angle and analyze it with respect to different genotypes. As some genetic factors responsible for gravitropism in rice have been identified [43], accurately quantifying these traits in different genotype may help identify responsible genes.

As future direction, we have developed a preliminary method (**Appendix D**) to estimate root angles from 3D root images. The key idea of our method is to find the center of distinct roots within projected slices of the 3D image data. We then connect these centers to form centerlines and estimate the angles in 3D.

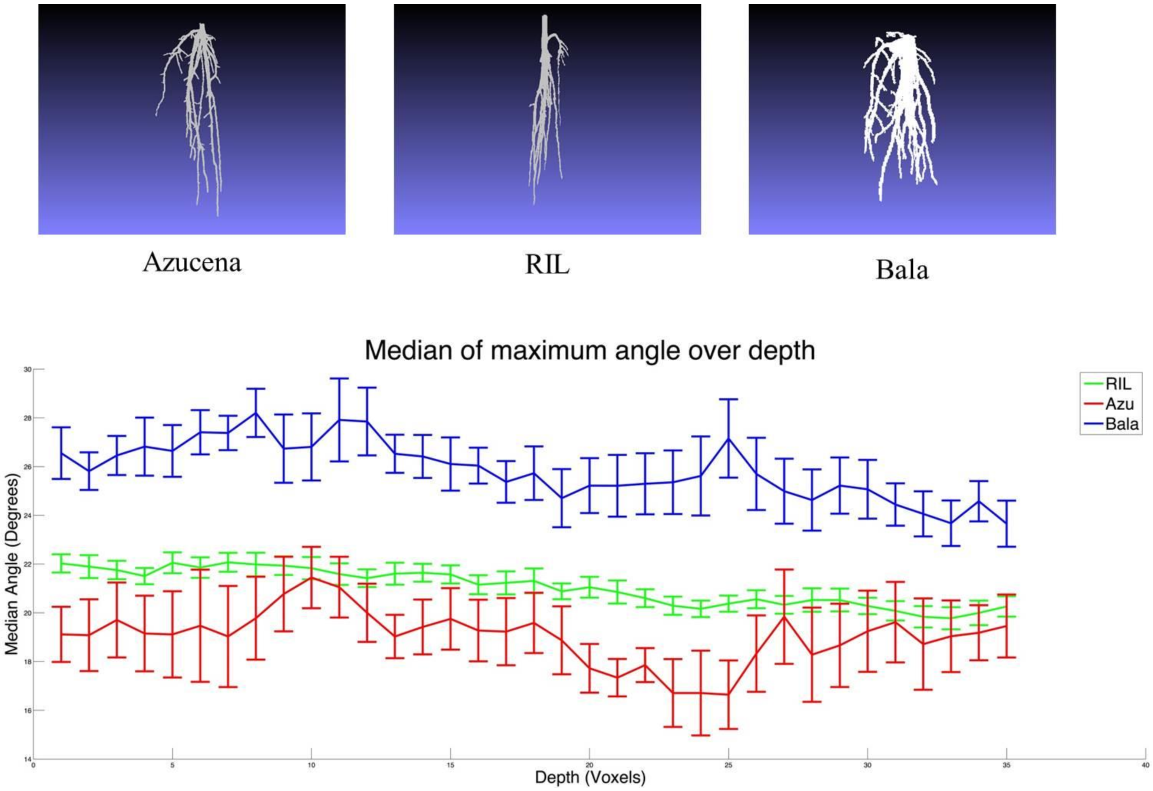


Figure 22: Top panel shows reconstructed 3D images of Azucena, RIL and Bala rice varieties. Bottom panel shows the median angle of maximum angle computed from the entire dataset for 35 voxels from the top of the image.

We have applied the method to estimate root angles from the reconstructed 3D rice roots. **Figure 22** shows the comparison of this angle between the two parental rice varieties (i.e. Azucena and Bala) and their recombinant inbred line (RIL). It can be seen that the root angles (in the top 35 voxels) of RIL varieties falls in between their parental varieties. The median root angle is an average from estimates taken from 105 images for Azucena, 1232 for RIL and 86 for Bala. This initial result suggests that automated phenotyping of 3D root images holds the potential for identifying heritable traits inaccessible via analysis of 2D images. We expect that the next-generation of DIRT will enable 3D analysis as a core component of its feature set.

APPENDIX A

A QUICK START GUIDE TO CLEARED LEAVES DATABASE

This appendix is adopted from the supplementary material of the publication Das A, Bucksch A, Price CA, Weitz JS; ClearedLeavesDB: an online database of cleared plant leaf images.; Plant Methods 2014, 10:8.

It describes the major functions that can be performed on the Cleared Leaves Database. A copy of this document can also be found on the website's home page accessible at http://www.clearedleavesdb.org/sites/default/files/QuickStart_ClearedLeavesDB.pdf.

Access the Database

To experience full functionality of the database, specifically bulk image upload, we encourage scientists and end users to use a HTML5 compliant browser to access the database. To use the database access the following url in a browser:
http://www.clearedleavesdb.org

User Registration

Please register to create and manage your own image collections. To register, please click on the 'Register' link on the top right corner of the database web site. An email will be send to your account with login information. Please check your email account after registration for an email from admin@clearedleavesdb.org. If you have email spam filters setup, the email might land in your 'trash' folder. Follow the instructions on the email.

Login

If you are a registered user of the database, use the ‘Login’ link on the database web site to login to the database.

Become a Curator

A registered user has full access to the database except bulk upload of images. So, to bulk upload images become a Curator. Please access the “How to be a curator” sub-menu under the “Curators” menu and fill out a form by clicking on the ‘form’ link on the page. User will be notified via email on approval of the request. Note that applications for curator status may take 1-2 week to be processed. Our objective is to personally review each request for a curator.

Create Image Collection

To create an image collection, please select “Create Collection” sub-menu under “Collections” menu. To create a private/public image collection, select collection visibility option appropriately. After saving the collection, user will be directed to collection view page, where he/she can click on the “Leaf Image” in “Add to Collection” section to add a leaf image to the newly created collection. If user is a curator, he/she can access the “Bulk Upload Images” sub-menu under the “Collections” menu to upload more than one image at a time.

Manage Collection Membership

A collection administrator can manage user and their roles in a collection. By default collection administrator can add new members to the group and manage group membership requests, but cannot edit roles and permissions.

To add new member to the collection:

- Go to “My Collections” page by selecting the “My Collections” sub-menu under “Collections” menu.

- Select the collection name to which you want to add a member.
- Click on “Group” tab
- Click “Add people” link.

A registered user can even request membership for a collection by clicking on the “Request collection membership” link of the collection view page. All membership requests can be managed by collection administrator by taking following steps:

- Go to “My Collections” page by selecting the “My Collections” sub-menu under “Collections” menu.
- Select the collection name to which you want to add a member.
- Click on “Group” tab
- Click on “People” link

To create custom user roles and permissions:

- Go to “My Collections” page by selecting the “My Collections” sub-menu under “Collections” menu.
- Select the collection name for which you want to override roles and permissions.
- Click on the ‘Edit” tab
- Select “Override default roles and permissions” option and click save.
- Click on the “Group” tab.
- Click on “Roles” and “Permissions” link to manage roles and permissions respectively.

Add Images to the Managed List

Images can be added to the managed list either from the collection view page or from search page. To add images from the collection view page follow these steps:

- Select one collection from the “Collections” page (i.e. click on the “Collections” menu) to go to the view page.

- Select images by clicking on the checkboxes next to it and add them to “Managed List” by clicking on the “Add to Managed List” button.

Similarly images can be added to the managed list from the search page (i.e. click on the “Images” menu).

Bulk Download Images

To bulk download images go the “Managed List” page by clicking on the “Managed List” sub-menu under “Images” menu, select some or all images and click on the “Download” button. It will prompt you to save a “.zip” file containing all images and its metadata.

Upload Processed Images

A registered user can add processed images to the database. Processed images are associated to an image. You can add processed images in either of following ways:

- From the image view page: Select an image from anywhere (Collection view page, image search page) in the site to go to the view page. Click on the 'Add new processed image' link to add processed image and its results to the database.
- Batch upload, processed images and its output files using our client application.

Please follow the directions mentioned in the following page on the web site.
http://www.clearedleavesdb.org/tools?qt-tools_quicktabs=2#qt-tools_quicktabs

Upload Image Metadata

A registered user and creator of a collection can upload an excel file (.xls) file containing metadata information for the images in the collection. Go to the collection view page by selecting a collection name from your collections listed on “MY COLLECTIONS” page. From the collections view page, click on the “Upload Metadata” link to upload an excel file containing the metadata. Please see “How to upload metadata for images in a collection?” section on FAQ page on the web site i.e.

http://www.clearedleavesdb.org/?q=about-clid&qt-about_clid=3#qt-about_clid

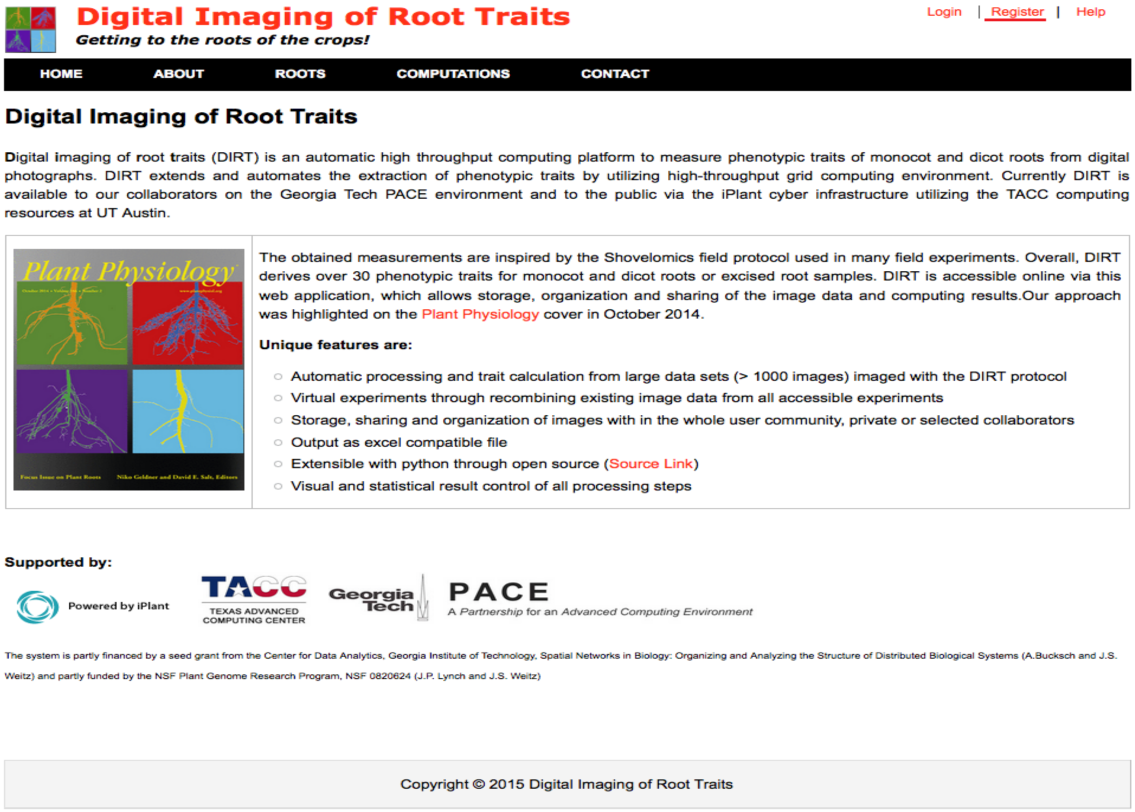
APPENDIX B

A QUICK START GUIDE TO DIRT

This appendix is adopted from the supplementary material of the publication Das A, Schneider H, Burridge J, Martinez Ascanio AK, Wojciechowski T, Topp CN, Lynch JP, Weitz JS, Bucksch A;DIRT: a high-throughput computing and collaboration platform for field-based plant phenomics; 2015,In Review.

Access DIRT

The public DIRT installation on the iPlant cyberinfrastructure is accessible at http://dirt.iplantcollaborative.org. The link accesses the DIRT home page (**Figure 23**).


Digital Imaging of Root Traits
Getting to the roots of the crops!
Login
Register
Help
HOME
ABOUT
ROOTS
COMPUTATIONS
CONTACT
Plant Physiology
Focus Issue on Plant Roots
Niko Geldner and David E. Salt, Editors
Powered by iPlant
TACC
TEXAS ADVANCED COMPUTING CENTER
Georgia Tech
PACE
A Partnership for an Advanced Computing Environment

Digital Imaging of Root Traits

Digital imaging of root traits (DIRT) is an automatic high throughput computing platform to measure phenotypic traits of monocot and dicot roots from digital photographs. DIRT extends and automates the extraction of phenotypic traits by utilizing high-throughput grid computing environment. Currently DIRT is available to our collaborators on the Georgia Tech PACE environment and to the public via the iPlant cyber infrastructure utilizing the TACC computing resources at UT Austin.

The obtained measurements are inspired by the Shovelomics field protocol used in many field experiments. Overall, DIRT derives over 30 phenotypic traits for monocot and dicot roots or excised root samples. DIRT is accessible online via this web application, which allows storage, organization and sharing of the image data and computing results.Our approach was highlighted on the Plant Physiology cover in October 2014.

Unique features are:

- Automatic processing and trait calculation from large data sets (> 1000 images) imaged with the DIRT protocol
- Virtual experiments through recombining existing image data from all accessible experiments
- Storage, sharing and organization of images with in the whole user community, private or selected collaborators
- Output as excel compatible file
- Extensible with python through open source (Source Link)
- Visual and statistical result control of all processing steps

Supported by:

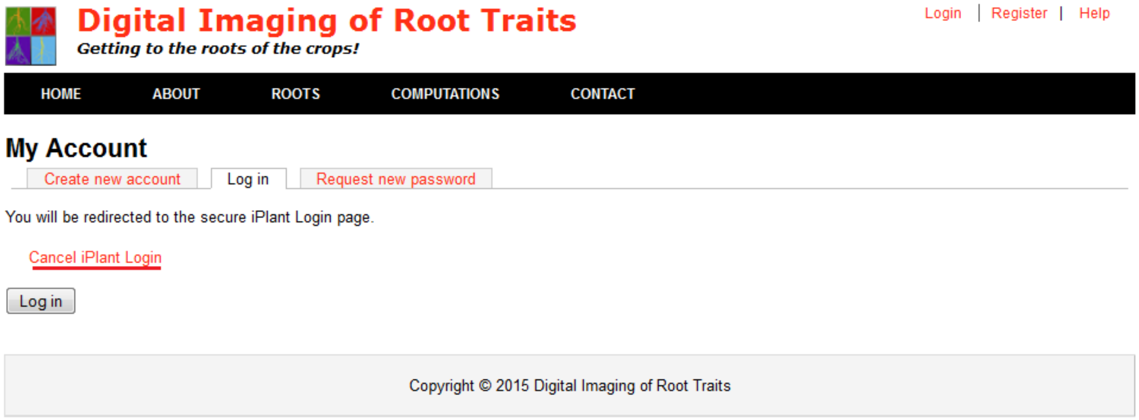
The system is partly financed by a seed grant from the Center for Data Analytics, Georgia Institute of Technology, Spatial Networks in Biology: Organizing and Analyzing the Structure of Distributed Biological Systems (A.Bucksch and J.S. Weitz) and partly funded by the NSF Plant Genome Research Program, NSF 0820624 (J.P. Lynch and J.S. Weitz)

Copyright © 2015 Digital Imaging of Root Traits

Figure 23: Screenshot of the DIRT home page.

Login to DIRT

Click on the ‘Login’ link on the home page (as show in **Figure 23**) to log into DIRT. DIRT uses iPlant’s Central Authentication Service (CAS) to authenticate its users. Therefore, it’s available to all iPlant’s registered users. By clicking on the ‘Log in’ button (**Figure 24**), will direct the user to the iPlant authentication services interface. Alternatively, using the ‘Cancel iPlant Login’ link (**Figure 24**) will direct the user to the DIRT authentication services interface. The iPlant authentication interface also provides link to sign up for an iPlant account.



You will be redirected to the secure iPlant Login page.

Copyright © 2015 Digital Imaging of Root Traits

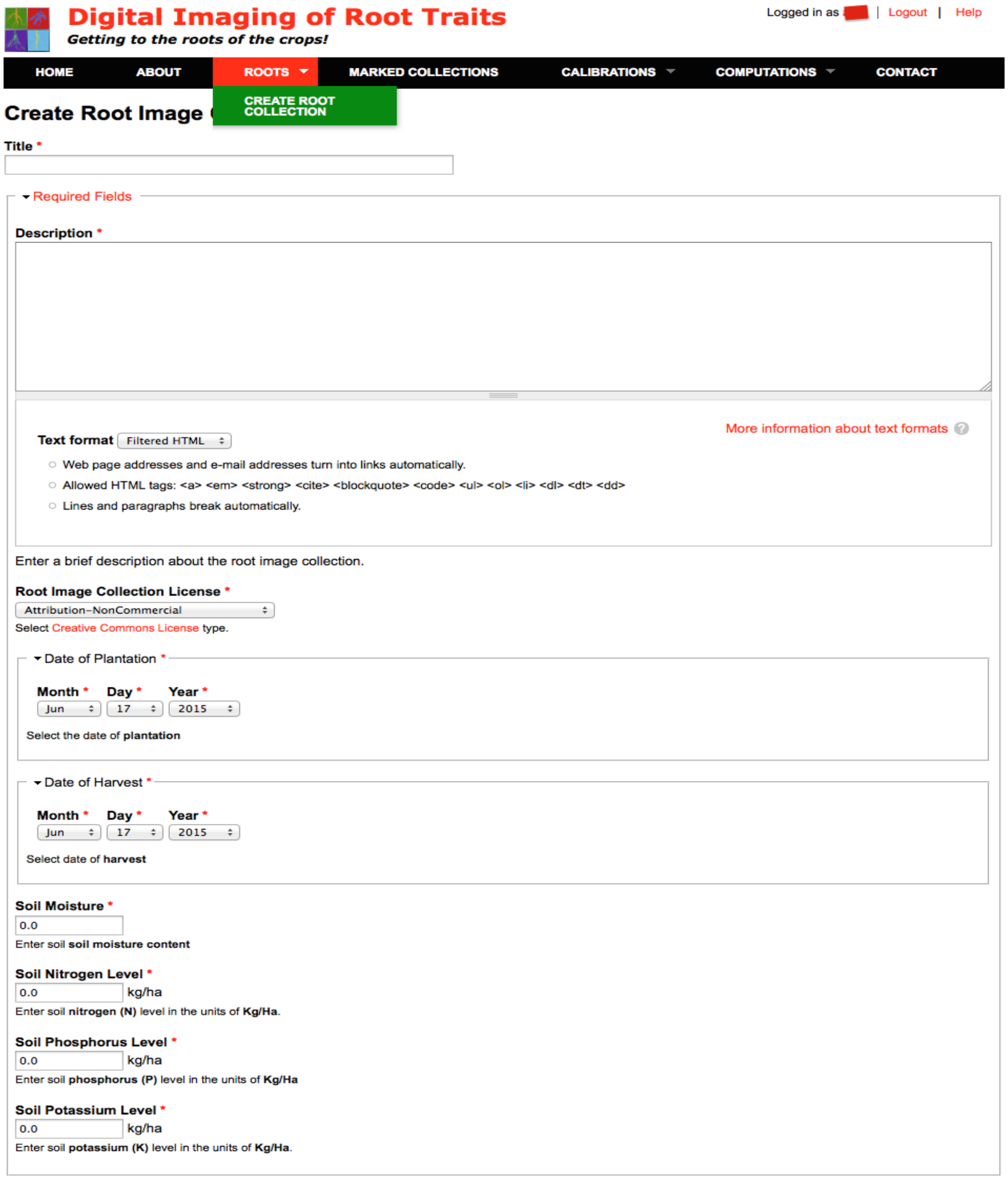
Figure 24: Screenshot of the DIRT user’s login page. This is shown after clicking on the ‘Login’ link available at the top right corner of the DIRT home page.

Root image collection

A root image collection (aka collection) is a container that holds a set of root images belonging to an experiment and its metadata. Each root image in the collection can also be tagged with its own metadata. All functionalities (mentioned in following subsections) associated to root image collection(s) are available to a logged in user via ‘ROOTS’ menu and its sub-menu. An anonymous user can only view publicly available collections, its metadata and associated root images.

Create root image collection

A user first creates a root image collection, in order to store and manage root images. To create a root image collection, select ‘CREATE ROOT COLLECTION’ sub-menu as shown in **Figure 25** and fill in values for the mandatory metadata fields. On successful submission of this form, the user will be navigated to the image upload screen as described in the next section.


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
CREATE ROOT COLLECTION

Create Root Image

Title *

Required Fields

Description *

Text format Filtered HTML

More information about text formats

- Web page addresses and e-mail addresses turn into links automatically.
- Allowed HTML tags: <a> <em> <strong> <cite> <blockquote> <code> <ul> <ol> <li> <dl> <dt> <dd>
- Lines and paragraphs break automatically.

Enter a brief description about the root image collection.

Root Image Collection License *
Attribution-NonCommercial
Select Creative Commons License type.

Date of Plantation *

Month *	Day *	Year *
Jun	17	2015

Select the date of **plantation**

Date of Harvest *

Month *	Day *	Year *
Jun	17	2015

Select date of **harvest**

Soil Moisture *
0.0
Enter soil **soil moisture content**

Soil Nitrogen Level *
0.0 kg/ha
Enter soil **nitrogen (N)** level in the units of **Kg/Ha**.

Soil Phosphorus Level *
0.0 kg/ha
Enter soil **phosphorus (P)** level in the units of **Kg/Ha**

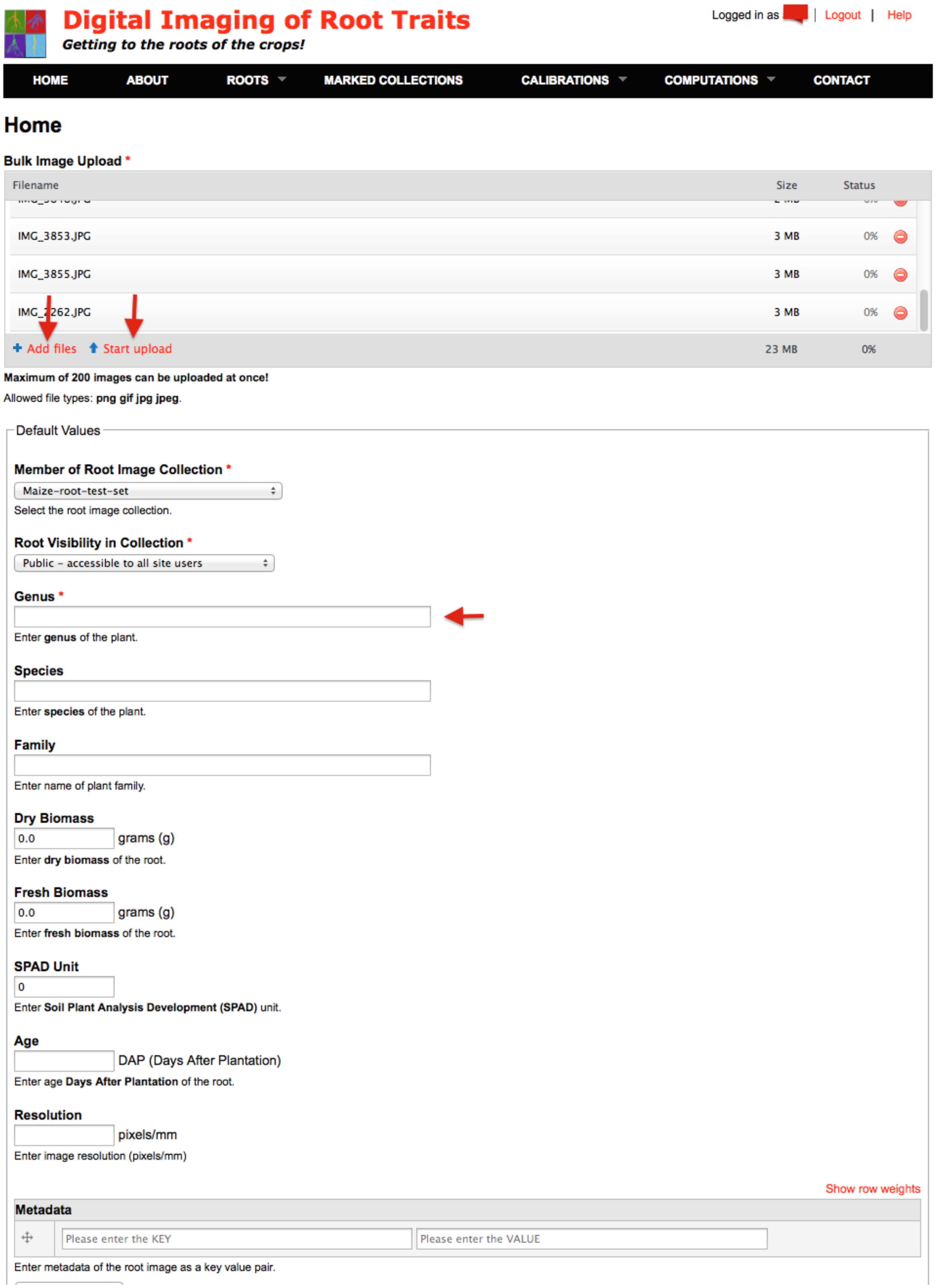
Soil Potassium Level *
0.0 kg/ha
Enter soil **potassium (K)** level in the units of **Kg/Ha.**

Figure 25: Screenshot of the DIRT's user interface to create a root image collection. It shows a portion (i.e. required field's) of the entire interface. All field names shown here are self-explanatory. All fields have default values; with the exception of 'Title' and 'Description' fields, edit them as appropriate for the collection.

Upload images to a collection

On creation of a new root image collection the user is navigated to the image upload interface shown in **Figure 26**. A maximum of 200 image files of type png, gif, jpg and jpeg can be uploaded at once. This restriction is in account of the limitations of browser's support for the http POST data transfer protocol. Steps to upload images are:

1. Click on 'Add files' link on this interface to add images from your local computer.
2. Click on 'Start upload'.
3. Enter values for the mandatory fields (i.e. with *) and optional fields as applicable.
4. Scroll down and click on 'Add to collection' button (not seen in **Figure 26**).
5. While images are being added to the collection by a background process, the user is navigated to the root image collection's view interface with a message i.e. '# uploaded images are being added to this collection'. The collection's view interface can be refreshed manually to see the recently added images or can be visited at a later time.



Home

Bulk Image Upload *

Filename	Size	Status
IMG_3853.JPG	3 MB	0%
IMG_3855.JPG	3 MB	0%
IMG_3262.JPG	3 MB	0%
+ Add files ⬆ Start upload	23 MB	0%

Maximum of 200 images can be uploaded at once!

Allowed file types: **png gif jpg jpeg**.

Default Values

Member of Root Image Collection *

Maize-root-test-set

Select the root image collection.

Root Visibility in Collection *

Public - accessible to all site users

Genus *

Enter **genus** of the plant.

Species

Enter **species** of the plant.

Family

Enter name of plant family.

Dry Biomass

0.0 grams (g)

Enter **dry biomass** of the root.

Fresh Biomass

0.0 grams (g)

Enter **fresh biomass** of the root.

SPAD Unit

0

Enter **Soil Plant Analysis Development (SPAD)** unit.

Age

DAP (Days After Plantation)

Enter age **Days After Plantation** of the root.

Resolution

pixels/mm

Enter image resolution (pixels/mm)

Metadata

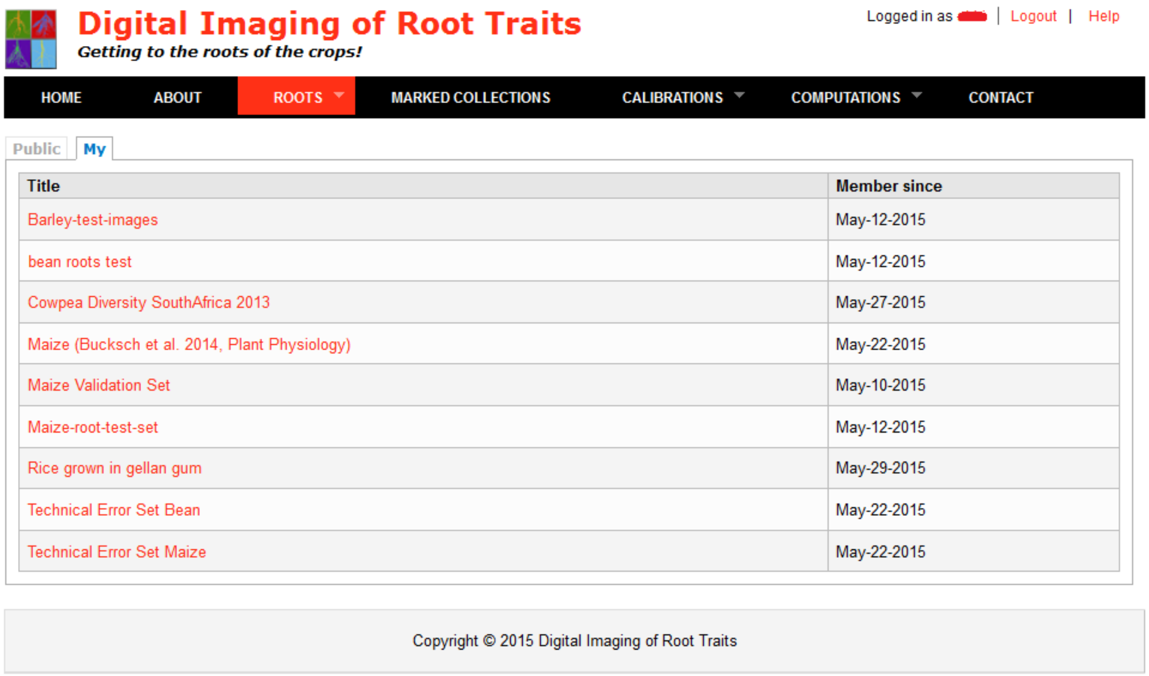
Please enter the KEY | Please enter the VALUE

Enter metadata of the root image as a key value pair.

Figure 26: Screenshot of the DIRT's user interface to upload and add images to a newly created or an existing collection. It shows a portion of the whole interface. This interface shows metadata fields associated to each image being uploaded to the collection. The red arrows indicate the links and fields that the use has to click and fill in respectively.

View root image collections

All collections managed by a user are accessible by visiting the 'ROOTS' menu and then clicking on the 'My' tab (**Figure 27**).


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
Public
My

Title	Member since
Barley-test-images	May-12-2015
bean roots test	May-12-2015
Cowpea Diversity SouthAfrica 2013	May-27-2015
Maize (Bucksch et al. 2014, Plant Physiology)	May-22-2015
Maize Validation Set	May-10-2015
Maize-root-test-set	May-12-2015
Rice grown in gellan gum	May-29-2015
Technical Error Set Bean	May-22-2015
Technical Error Set Maize	May-22-2015

Copyright © 2015 Digital Imaging of Root Traits

Figure 27: Screenshot of the DIRT's user interface that shows the list of root image collections managed by the logged in user.

Share root image collection

Root image collections managed by a user can be shared publicly or privately with collaborators.

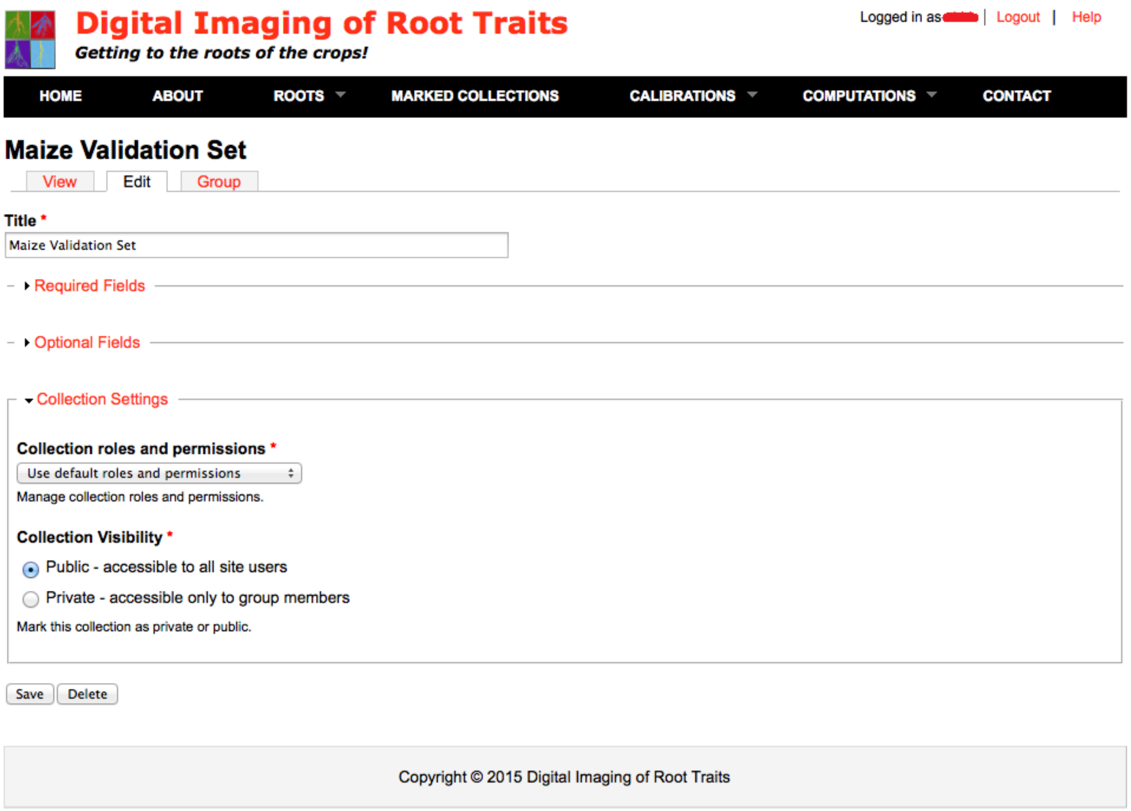
Share a collection with public

Root image collections managed by a user can be shared publicly or privately with collaborators.

1. Click on the 'ROOTS' menu item.
2. Select the collection from the 'My' tab (**Figure 27**).
3. Click on the 'Edit' tab (**Figure 28**).
4. Scroll down and expand the 'Collections settings' section.

5. Choose public under collection visibility section and click save.

Public collections are visible to all the visitors of the site including non-registered or anonymous users.



Maize Validation Set

Title *
Maize Validation Set

Collection roles and permissions *
Use default roles and permissions
Manage collection roles and permissions.

Collection Visibility *
● Public - accessible to all site users
○ Private - accessible only to group members
Mark this collection as private or public.

Copyright © 2015 Digital Imaging of Root Traits

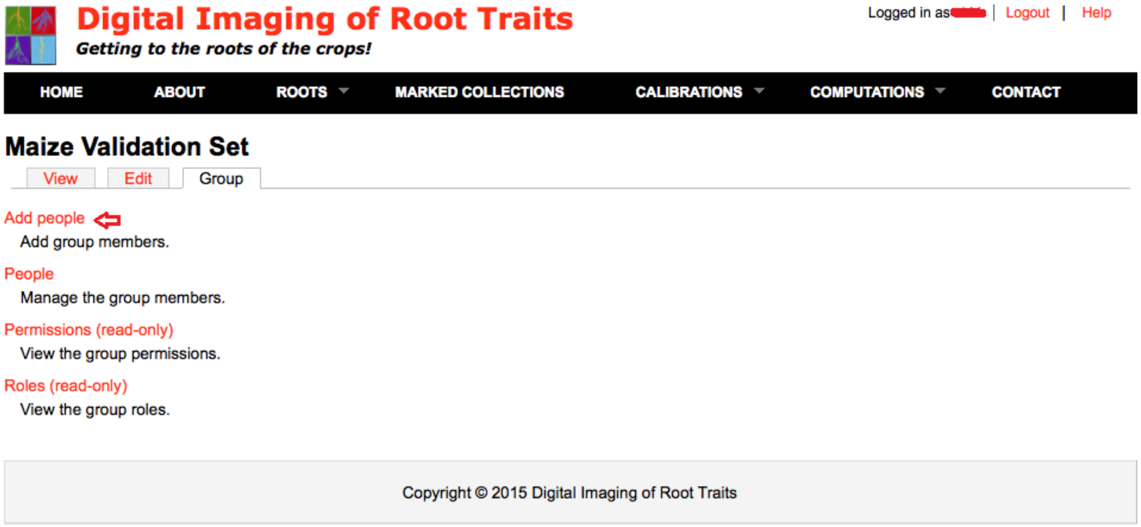
Figure 28: Screenshot of the DIRT's user interface to edit a root image collection and change its collection setting to 'Public'.

Share the collection with collaborators

To share a root image collection privately with collaborators (i.e. other registered users of DIRT):

1. Select a collection from the 'My' tab by visiting the 'ROOTS' menu.
2. From root image collections view page click on the 'Group' tab (**Figure 29**).
3. Click on the 'Add people' link.

The 'Add people' link will bring up an interface to search for a user and add the user to the collection. To manage membership of the existing users, click on the 'People' link as shown in **Figure 29**. User name of all registered users can be accessed by visiting the 'ABOUT' menu.


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as[redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
View
Edit
Group

Maize Validation Set

Add people

Add group members.

People

Manage the group members.

Permissions (read-only)

View the group permissions.

Roles (read-only)

View the group roles.

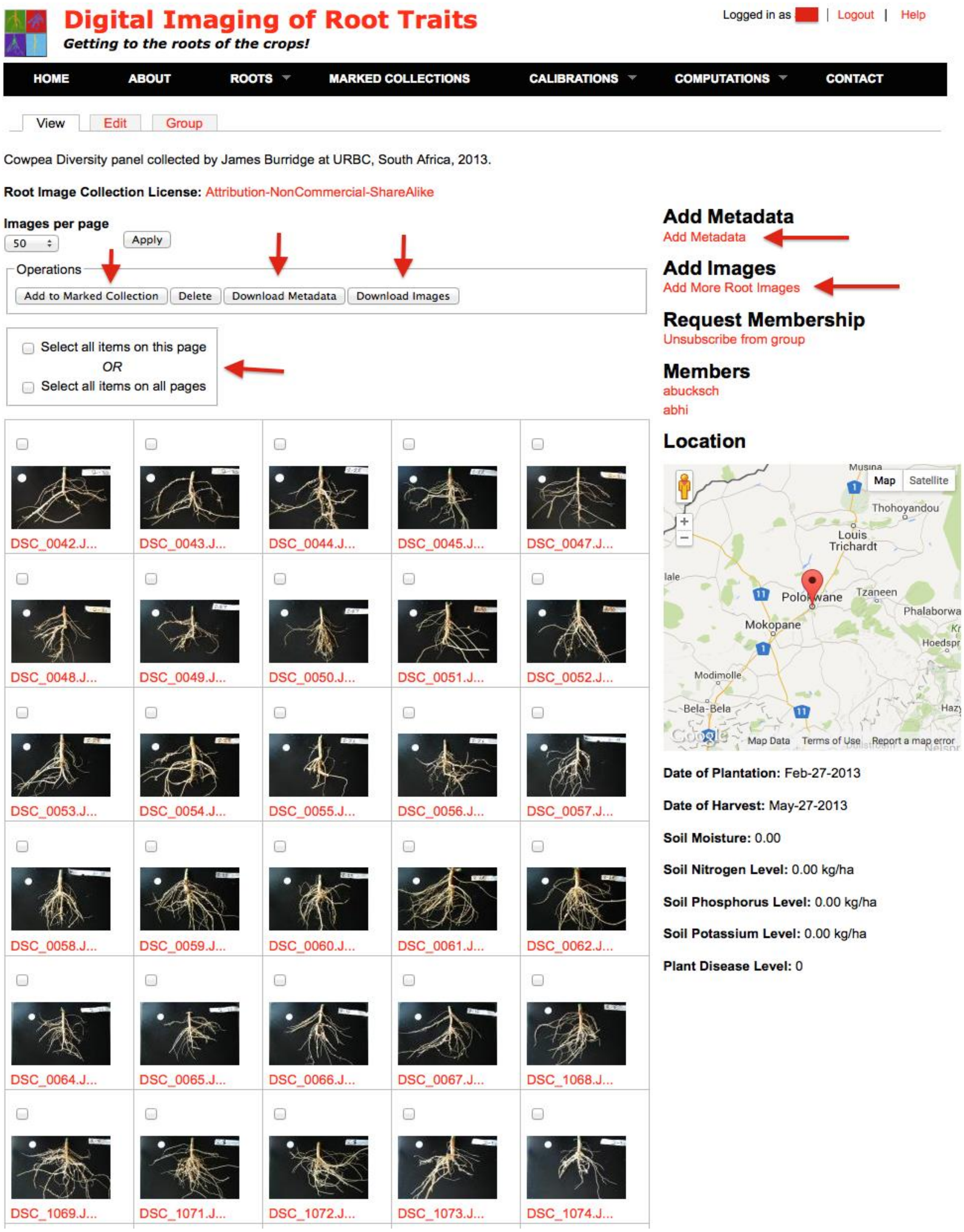
Copyright © 2015 Digital Imaging of Root Traits

Figure 29: Screenshot of the DIRT's user interface that enables to add and manage members of a root image collection. The red arrow indicated the link the user has to click in order add people/member to the collection.

Download images from a collection

A registered user can download root images from a public collection or from a collection he/she is a member of. To download images:

1. Select a collection from the 'Public'/'My' tab by visiting the 'ROOTS' menu.
2. Selecting a collection will bring up the collection view page (**Figure 30**).
3. Select images of choice and click on the 'Download Images' button (**Figure 30**).
4. Selected images will be downloaded as a 'ZIP' file. Based on the user's browser settings it will either open up a window asking user to save the zip file or it will save the zip file to the download directory.



Cowpea Diversity panel collected by James Burridge at URBC, South Africa, 2013.

Root Image Collection License: Attribution-NonCommercial-ShareAlike

Date of Plantation: Feb-27-2013

Date of Harvest: May-27-2013

Soil Moisture: 0.00

Soil Nitrogen Level: 0.00 kg/ha

Soil Phosphorus Level: 0.00 kg/ha

Soil Potassium Level: 0.00 kg/ha

Plant Disease Level: 0

Figure 30: Screenshot of the DIRT's root image collection view interface. A user can perform all functions associated to a collection from this interface. Functions are (a) Edit collection metadata by clicking on the 'Edit' tab, (b) Add/manage people or member of the collection by clicking on the 'Group' tab, (c) Add images to a 'Marked Collection', (d) Delete selected image(s), (e) Download image metadata, (f) Download images, (g) Add image metadata and (h) Add more root images to the collection. The red arrow indicates the major operations/actions that can be performed on a collection.

Download image metadata of a collection

Metadata associated with images of a collection can be downloaded from the collection view interface (**Figure 30**). The metadata of the selected images will be downloaded in CSV file format. The following lines shows contents of a sample metadata CSV file that can be edited via an ASCII text editor such as Notepad or Microsoft Excel.

Image ID,Image Name,Genus,Species,Family,Dry Biomass,Fresh Biomass,SPAD,Age, Resolution(pixels/mm)

17951,DSC_0042.JPG,Vigna,unguiculata,Fabaceae,0,0,0,0,0

17952,DSC_0043.JPG,Vigna,unguiculata,Fabaceae,0,0,0,0,0

First line of the CSV file represents the field labels and subsequent lines represent the metadata values of the images. This file can also be used to 'Add metadata' to the images of a collection.

To download metadata of images of a collection:

1. Select a collection from the 'Public'/'My' tab by visiting the 'ROOTS' menu.
2. Selecting a collection will bring up the collection view page (**Figure 30**).
3. Select images of choice and click on the 'Download Metadata' button (**Figure 30**).

Add metadata of images in a collection

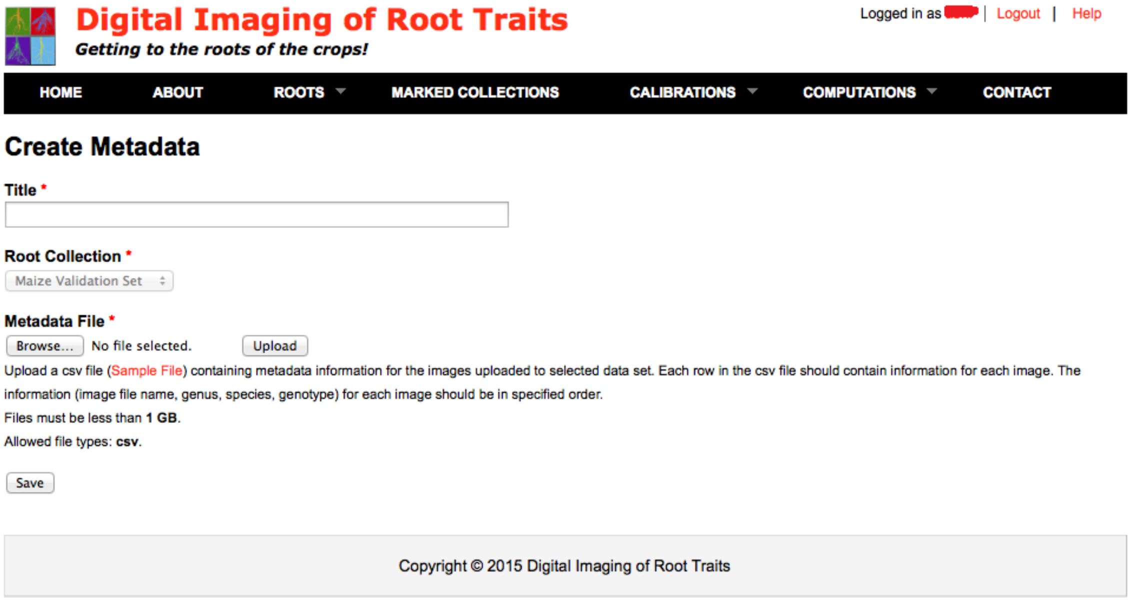
Each image in DIRT has its own metadata. Image metadata can be added either by editing each image or in batch by uploading a CSV file per collection containing metadata information about each image. To add/update/edit metadata of each image:

1. Go the collection view page as shown in **Figure 30.**
2. Select an image by clicking on the link shown below the image.
3. On image view page click on the 'Edit' tab to add/update its metadata.

Metadata batch upload for multiple images of a collection:

1. Download metadata file of the selected images or all images in a collection.

2. Edit and update the downloaded CSV file with appropriate metadata values. Optionally, you can add additional key names at the end of the header and fill in the corresponding values.
3. Click on the 'Add Metadata' on the collection view interface (**Figure 30**).
4. It will bring up the metadata upload interface as shown in **Figure 31.**
5. Enter a title, select the updated metadata CSV file by clicking on the 'Browse' button and then click on 'Upload' and finally click 'Save'.


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
Browse...
Upload
Save

Create Metadata

Title *

Root Collection *
Maize Validation Set

Metadata File *
No file selected.

Upload a csv file (Sample File) containing metadata information for the images uploaded to selected data set. Each row in the csv file should contain information for each image. The information (image file name, genus, species, genotype) for each image should be in specified order.
Files must be less than **1 GB**.
Allowed file types: **csv**.

Copyright © 2015 Digital Imaging of Root Traits

Figure 31: Screenshot of the DIRT's interface to add/update metadata of images in a collection.

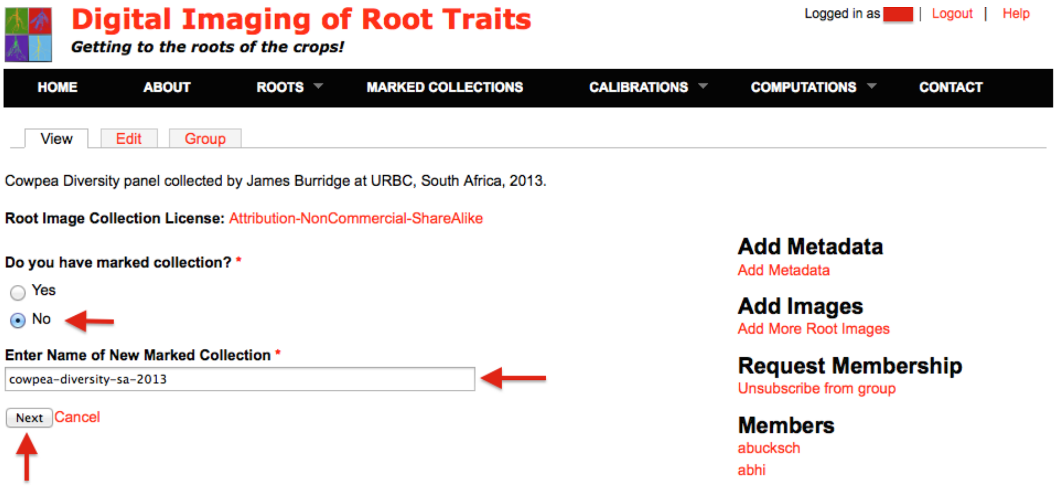
Marked image collection

Marked image collection (aka Marked Collection) is a functionality that allows users to create virtual image collection by mixing images from different physical root image collections. This is also a required functionality for the calibration and RSA trait computation. The marked collection functionality is only privately available to a registered user. A user can access all of his/her marked collections by visiting the 'MARKED COLLECTIONS' menu as shown in **Figure 34**. Marked collection functionality is also available on the collections view page as shown in **Figure 30**, where a user can add selected images to a new or existing marked collection.

Create a marked collection

To create a new marked collection:

1. Go to the collection view page by selecting a collection from your list (**Figure 27**).
2. Select (all or some) images from the collection view page (**Figure 30**).
3. Click on ‘Add to Marked Collection’ button.
4. The click will open the marked collection interface as shown in **Figure 32**, select option ‘No’ and enter a name for the new marked collection and click on the ‘Next’ button.


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
View
Edit
Group

Cowpea Diversity panel collected by James Burridge at URBC, South Africa, 2013.

Root Image Collection License: Attribution-NonCommercial-ShareAlike

Do you have marked collection? *

○ Yes

● No

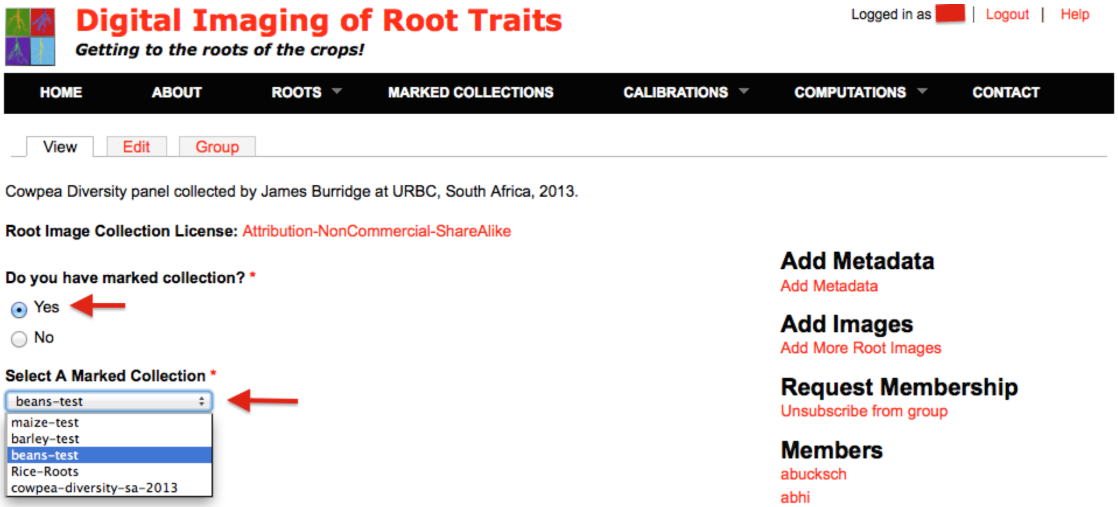
Enter Name of New Marked Collection *

cowpea-diversity-sa-2013

Figure 32: Screenshot of the DIRT’s interface to create a new marked collection and add images to it. The red arrow indicates the fields that need user’s attention.

Add images to an existing marked collection

To add images to an existing marked collection, follow the same procedure as mentioned in the previous section, in step 4 instead on selecting option ‘No’, select ‘Yes’ as shown in **Figure 33** and then choose a marked collection from the select list. If the select list has zero items, change your option to ‘No’ to create a new marked collection.


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
View
Edit
Group

Cowpea Diversity panel collected by James Burridge at URBC, South Africa, 2013.

Root Image Collection License: Attribution-NonCommercial-ShareAlike

Do you have marked collection? *

● Yes

○ No

Select A Marked Collection *

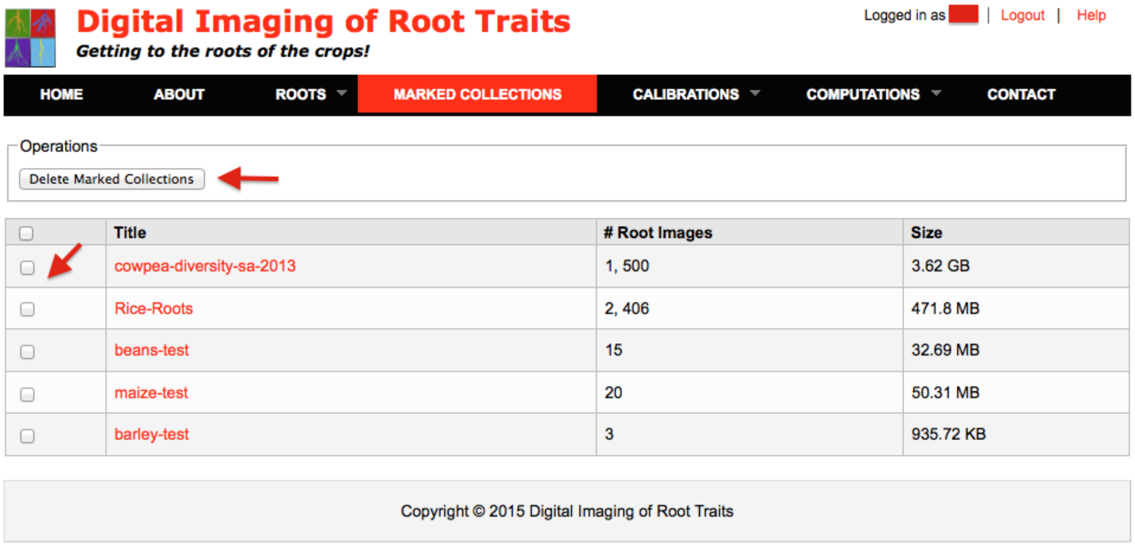
beans-test

- maize-test
- barley-test
- beans-test
- Rice-Roots
- cowpea-diversity-sa-2013

Figure 33: Screenshot of the DIRT's interface to add images to an existing marked collection. The red arrow indicates the fields that need user's attention.

View marked collections

To view the list of available marked collections, click on the 'MARKED COLLECTIONS' menu item as show in **Figure 34**. The user can delete an existing marked collection by selecting a collection from the list and clicking on the 'Delete Marked Collections' button.


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
Operations
Delete Marked Collections

□	Title	# Root Images	Size
□	cowpea-diversity-sa-2013	1, 500	3.62 GB
□	Rice-Roots	2, 406	471.8 MB
□	beans-test	15	32.69 MB
□	maize-test	20	50.31 MB
□	barley-test	3	935.72 KB

Figure 34: Screenshot of the DIRT's interface that lists the marked collections of the logged in user. The red arrow indicates the fields that need user's attention/action to delete a marked collection.

Image Threshold Calibration

The RSA trait computation pipeline available on DIRT, requires the user to provide segmentation threshold value as one of the parameters. A user can determine the threshold value by using image threshold calibration (aka Calibration) functionality. Calibration enables DIRT users to visually inspect calculated masks of a selected root image for a set of threshold values of 1, 3, 5, 10, 15 and 20. Hence, users without technical background can choose appropriate threshold values for the computation. A user can access this functionality by clicking on the 'CALIBRATIONS' menu and its sub-menu 'CALIBRATE'. DIRT enables calibration of one image at a time. If a user wants to calibrate the thresholds for multiple images, he/she has to calibrate several images individually and compare their results visually. The calibration functionality can also be accessed from the 'Create Computation' interface defined in future sections of this appendix.

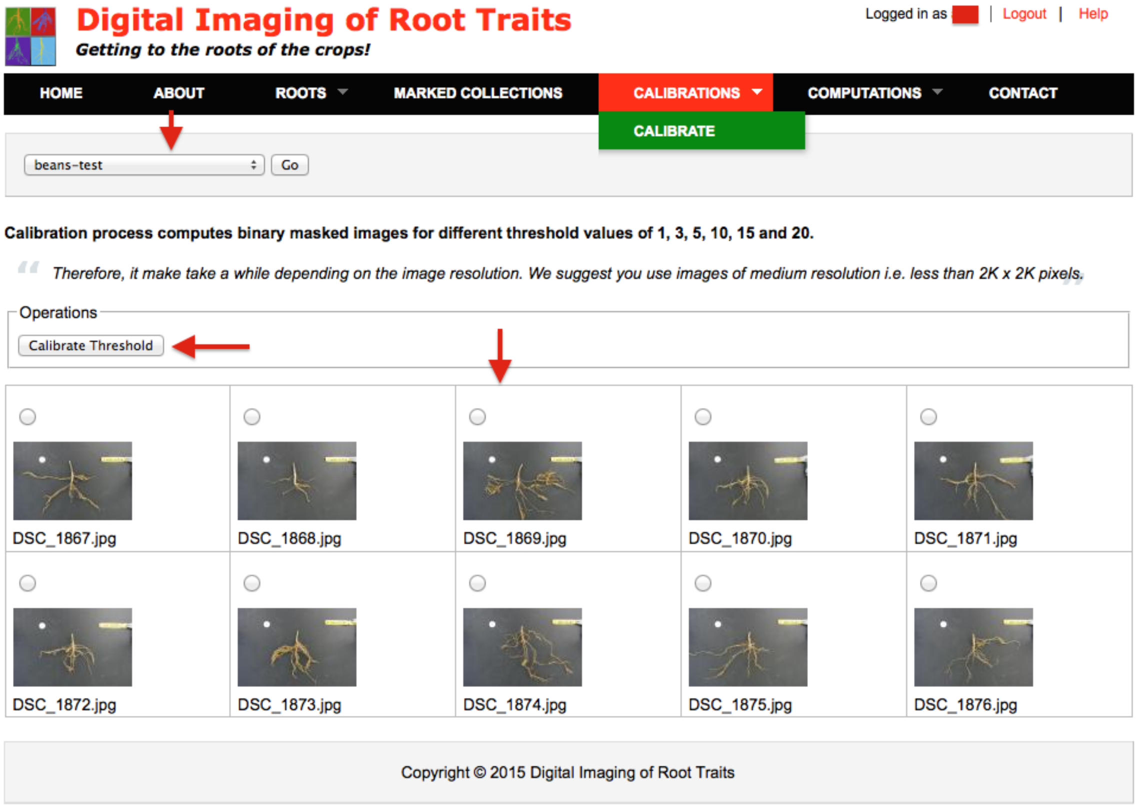
Calibrate threshold value of an image

To calibrate an image, click on the 'CALIBRATE' sub-menu as shown in **Figure 35**.

It works in two steps:

1. Select a marked collection whose image needs calibration and click 'Go' as shown in **Figure 35**.
2. Choose an image and click on 'Calibrate Threshold'. Depending on the image resolution this process may take several minutes (may be more than 15 minutes). During this computation the user will see the interface as shown in **Figure 36**. Do not close the browser window while it computes. Moving the focus away from the browser, won't show the message as seen in **Figure 36**, but the computation is still running for the selected image and the page will refresh with masked images after its completion. Closing the browser window will terminate the computation.

On completion of the calibration, the user will be presented with binary masked images and their corresponding threshold values (**Figure 37**). The user can select the appropriate masked image and navigated to the 'Create computation' page with corresponding threshold value.



Digital Imaging of Root Traits

Getting to the roots of the crops!

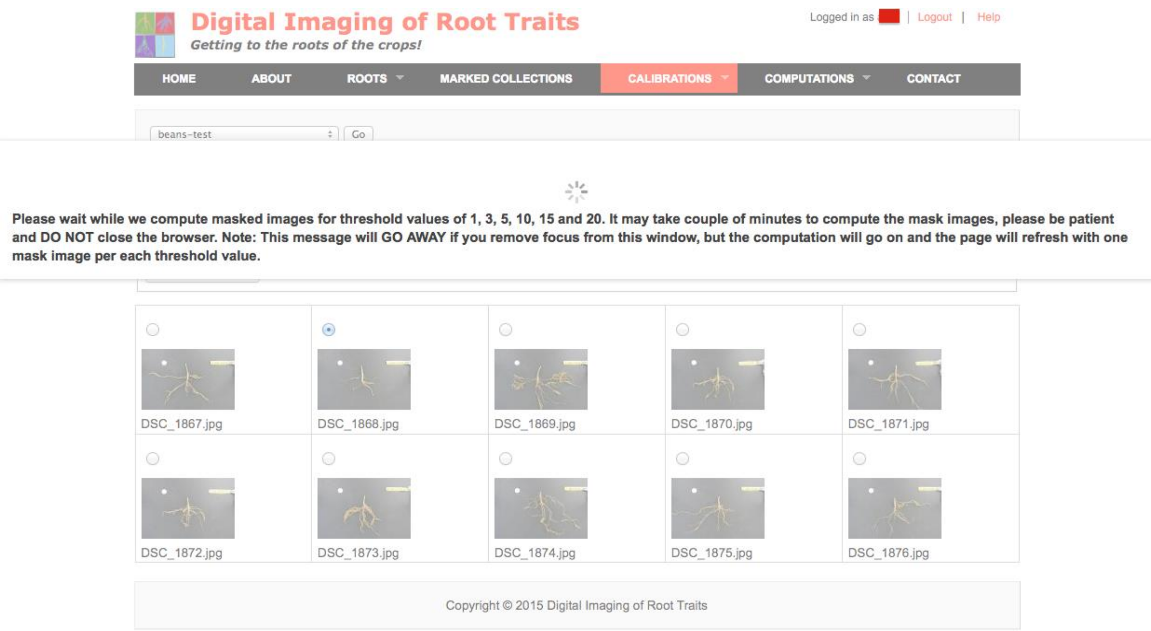
Logged in as [redacted] | Logout | Help

Calibration process computes binary masked images for different threshold values of 1, 3, 5, 10, 15 and 20.

> *Therefore, it make take a while depending on the image resolution. We suggest you use images of medium resolution i.e. less than 2K x 2K pixels.*

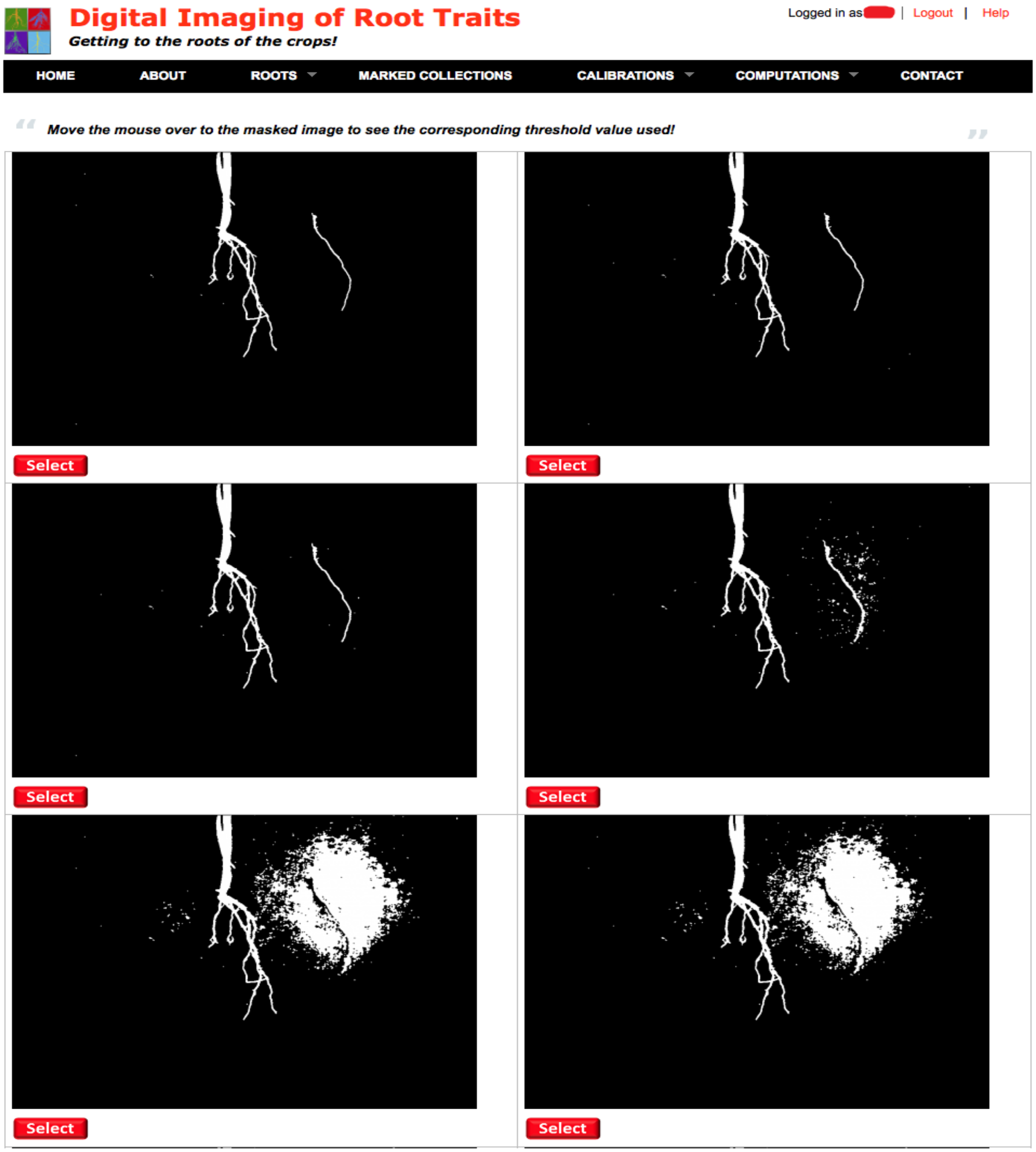
Copyright © 2015 Digital Imaging of Root Traits

Figure 35: Screenshot of the DIRT's interface to calibrate segmentation threshold value for an image in the marked collection. It works in two steps (a) Select a marked collection from the list as seen in the above figure and click 'Go', this will load the images of the marked collection, and (b) select an image and click on 'Calibrate Threshold' button. The red arrow indicates the fields/buttons that need user's attention/action for threshold calibration.



Please wait while we compute masked images for threshold values of 1, 3, 5, 10, 15 and 20. It may take couple of minutes to compute the mask images, please be patient and DO NOT close the browser. Note: This message will GO AWAY if you remove focus from this window, but the computation will go on and the page will refresh with one mask image per each threshold value.

Figure 36: Screenshot of the DIRT's interface that shows up while an image is being calibrated by a user.



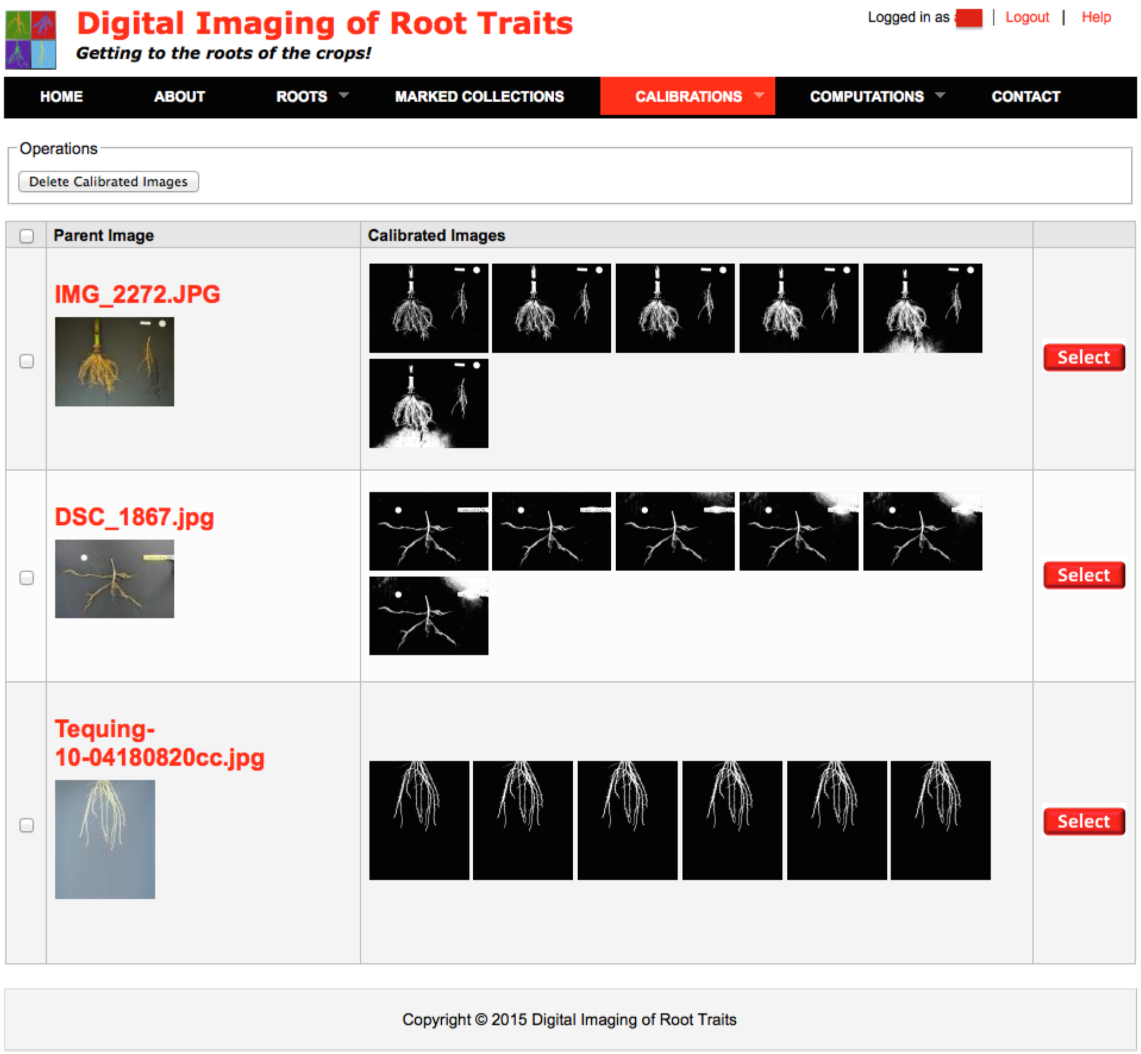
Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
Select
Select
Select
Select
Select
Select

"Move the mouse over to the masked image to see the corresponding threshold value used!"

Figure 37: Screenshot of the DIRT's interface that shows up on completion of the calibration. All masked images with threshold values are presented to the user for inspection and selection.

View calibrations

A user can access all past calibrations (**Figure 38**) by clicking on the 'CALIBRATIONS' menu. The calibration list interface also allows the user to delete past calibrations. To delete a calibration, select the appropriate row and click on the 'Delete Calibrated Images' button as shown in **Figure 38**.


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
Operations
Delete Calibrated Images

□	Parent Image	Calibrated Images	
□	IMG_2272.JPG		Select
□	DSC_1867.jpg		Select
□	Tequing-10-04180820cc.jpg		Select

Copyright © 2015 Digital Imaging of Root Traits

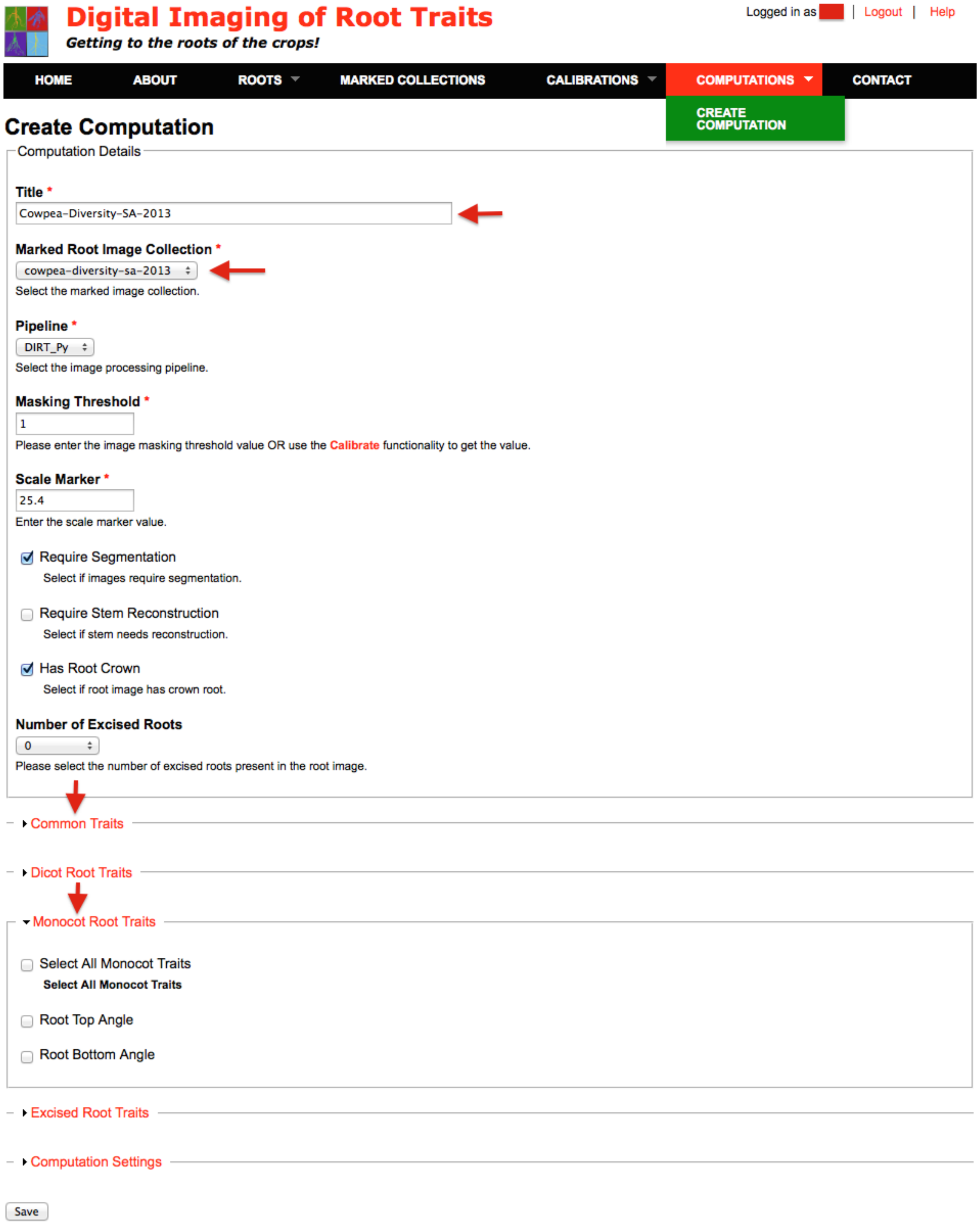
Figure 38: Screenshot of the DIRT's interface that shows the list of calibrations performed by a user.

RSA Trait Computation

RSA trait computation (aka Computation) enables the user to compute the whole or sub-set of traits defined in **Chapter 3**, for all the images in a marked collection. A user can access this functionality by clicking on the 'COMPUTATIONS' menu and its 'CREATE COMPUTATION' sub-menu.

Create a computation

The create computation interface enables the user to transfer the images of a marked collection to the high-throughput computing system and run the DIRT RSA trait computation pipeline on the images of the marked collection. DIRT on the iPlant cyber infrastructure submits images to TACC’s stampede environment (https://www.tacc.utexas.edu/systems/stampede) to compute multiple images at once. In order to initiate the transfer, a user has to navigate to the create computation interface. This interface is accessible as the ‘CREATE COMPUTATION’ item in the ‘COMPUTATIONS’ menu (**Figure 39**). For detailed description about the pipeline parameters, we refer to chapter 3. For an appropriate masking threshold value the user can access the calibration functionality, from this interface by clicking on the ‘Calibrate’ link as shown in **Figure 39**. By default all RSA traits are marked for computation. The user can change this setting by expanding the appropriate sections (i.e. Common Traits, Dicot Root Traits, etc. as shown **Figure 39**) and then checking and unchecking the required traits. On successful submission the user will be navigated to computations list page with a message i.e. ‘Submitting jobs to the grid! Computation <name> has been created’. Note, that also images can be computed with no scale marker (e.g. Barley Diversity panel). However, the output will be in pixels.



Create Computation

Computation Details

Title *
Cowpea-Diversity-SA-2013

Marked Root Image Collection *
cowpea-diversity-sa-2013
Select the marked image collection.

Pipeline *
DIRT_Py
Select the image processing pipeline.

Masking Threshold *
1
Please enter the image masking threshold value OR use the Calibrate functionality to get the value.

Scale Marker *
25.4
Enter the scale marker value.

☑ Require Segmentation
Select if images require segmentation.

□ Require Stem Reconstruction
Select if stem needs reconstruction.

☑ Has Root Crown
Select if root image has crown root.

Number of Excised Roots
0
Please select the number of excised roots present in the root image.

Monocot Root Traits

□ Select All Monocot Traits
Select All Monocot Traits

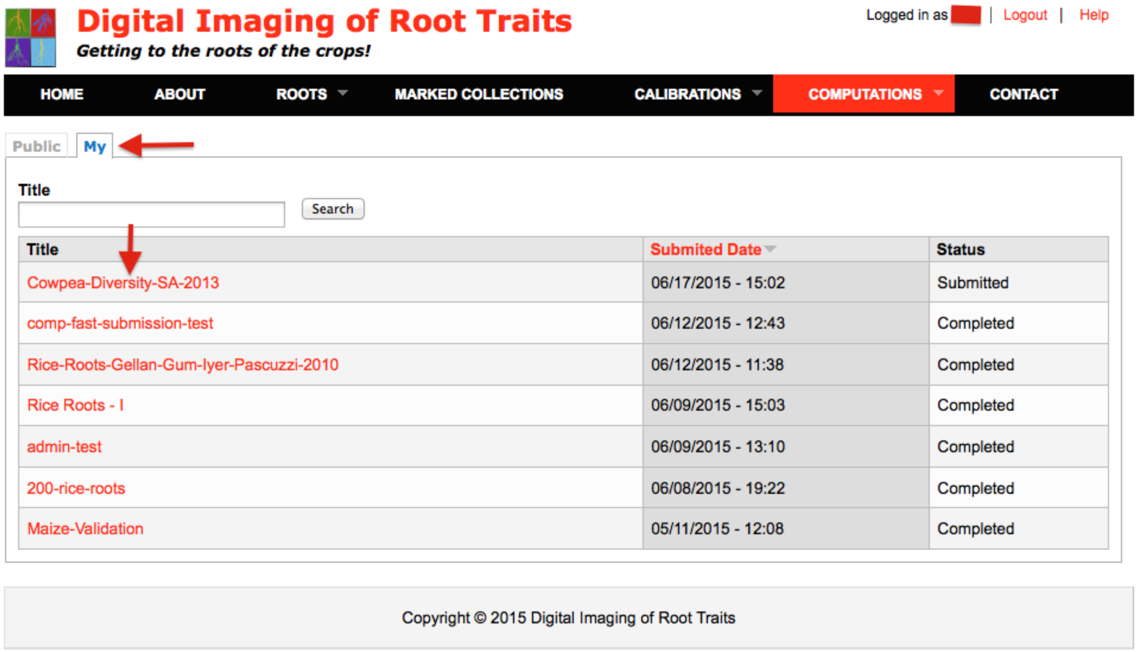
□ Root Top Angle

□ Root Bottom Angle

Figure 39: Screenshot of the DIRT's interface to create a new computation. This interface enables the user to give a title to the computation, select the marked collection whose images are to be used for trait computation, select the trait computation pipeline, provide pipeline parameters and select traits of interest. The red arrow indicates the fields that need user's attention.

View computations

A user can view his/her computations by clicking on the ‘COMPUTATIONS’ menu and then selecting ‘My’ tab interface as shown in **Figure 40**.


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
Public
My
Title
Search
Copyright © 2015 Digital Imaging of Root Traits

Title	Submitted Date	Status
Cowpea-Diversity-SA-2013	06/17/2015 - 15:02	Submitted
comp-fast-submission-test	06/12/2015 - 12:43	Completed
Rice-Roots-Gellan-Gum-Iyer-Pascuzzi-2010	06/12/2015 - 11:38	Completed
Rice Roots - I	06/09/2015 - 15:03	Completed
admin-test	06/09/2015 - 13:10	Completed
200-rice-roots	06/08/2015 - 19:22	Completed
Maize-Validation	05/11/2015 - 12:08	Completed

Figure 40: Screenshot of the DIRT’s interface that shows the list of computations managed/associated to the logged in user. The red arrow indicates the fields that need user’s attention/action.

Share a computation

Similar to a collection, a computation can be shared publicly with all or privately with trusted collaborators.

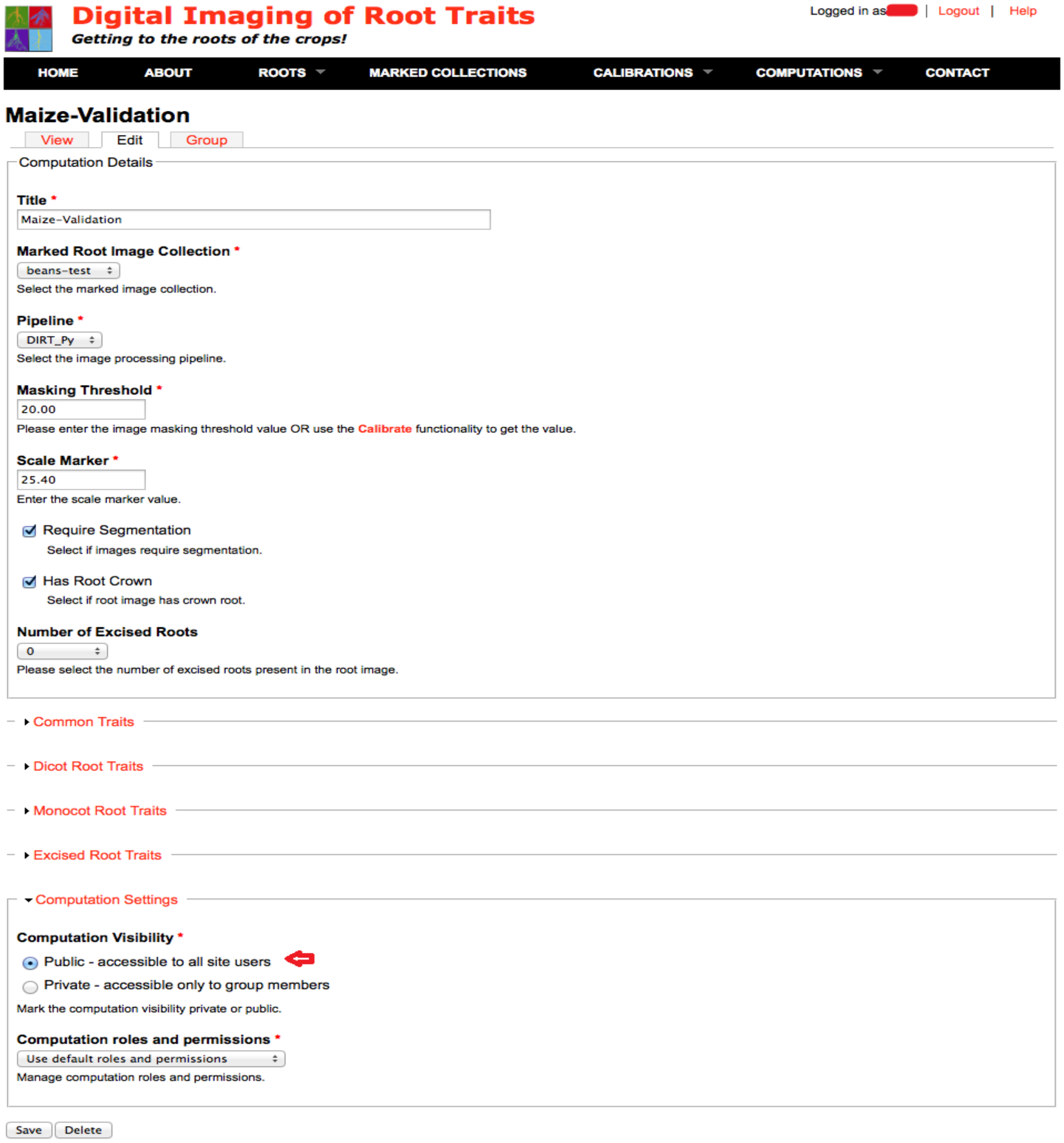
Share with public

Similar to root image collections, computations are marked private by default. The owner of the computation can declare it public by editing the computation as shown in Figure 31 and changing its ‘Computation Settings’ to public. To mark a computation public:

1. Click on the ‘COMPUTATIONS’ menu item.
2. Select the computation from the ‘My’ tab (**Figure 40**).

3. Click on the ‘Edit’ tab (**Figure 41**).
4. Scroll down and expand the ‘Computation settings’ section.
5. Choose public under collection visibility section and click save.

Public computations are visible to all the visitors of the site including non-registered or anonymous users.



Maize-Validation

Computation Details

Title *
Maize-Validation

Marked Root Image Collection *
beans-test
Select the marked image collection.

Pipeline *
DIRT_Py
Select the image processing pipeline.

Masking Threshold *
20.00
Please enter the image masking threshold value OR use the Calibrate functionality to get the value.

Scale Marker *
25.40
Enter the scale marker value.

☑ Require Segmentation
Select if images require segmentation.

☑ Has Root Crown
Select if root image has crown root.

Number of Excised Roots
0
Please select the number of excised roots present in the root image.

Computation Settings

Computation Visibility *
● Public - accessible to all site users
○ Private - accessible only to group members
Mark the computation visibility private or public.

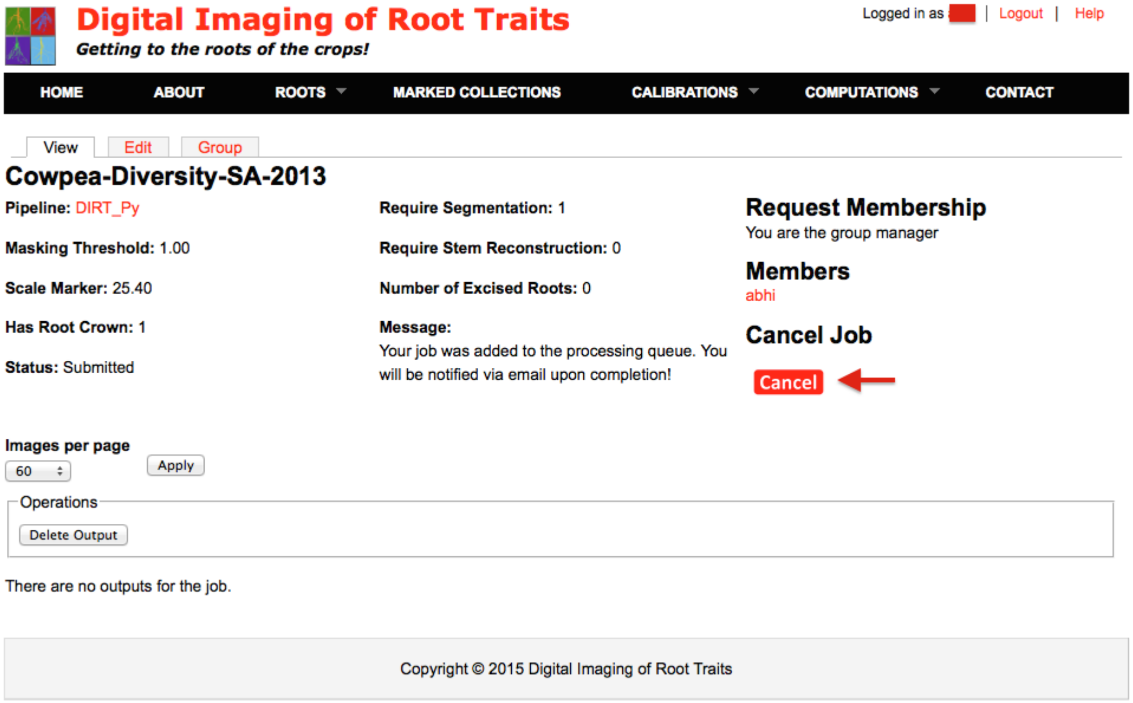
Computation roles and permissions *
Use default roles and permissions
Manage computation roles and permissions.

Figure 41: Screenshot of the DIRT’s interface that enables user to start/create a RSA trait computation process on the underlying high-throughput grid computing environment. The red arrow indicates the fields that need user’s attention/action.

Share privately with collaborators

To share a computation privately with collaborators (i.e. other registered users of DIRT):

1. Select a computation from the 'My' tab by visiting the 'COMPUTATIONS' menu (**Figure 40**).
2. From computations view page (**Figure 42**) click on the 'Group' tab.
3. Click on the 'Add people' link.
4. Add people link will bring up an interface to search for a user and add the user to the computation. To manage membership of the existing users, click on the 'People' link.



Cowpea-Diversity-SA-2013

Pipeline: DIRT_Py

Masking Threshold: 1.00

Scale Marker: 25.40

Has Root Crown: 1

Status: Submitted

Require Segmentation: 1

Require Stem Reconstruction: 0

Number of Excised Roots: 0

Message:
Your job was added to the processing queue. You will be notified via email upon completion!

Request Membership
You are the group manager

Members
abhi

Cancel Job

Images per page
60

There are no outputs for the job.

Copyright © 2015 Digital Imaging of Root Traits

Figure 42: Screenshot of the DIRT's interface that allows the user to edit, manage computation membership and cancel a submitted or running computation. The red arrow indicates the button that need user's action to cancel a computation.

Cancel a computation

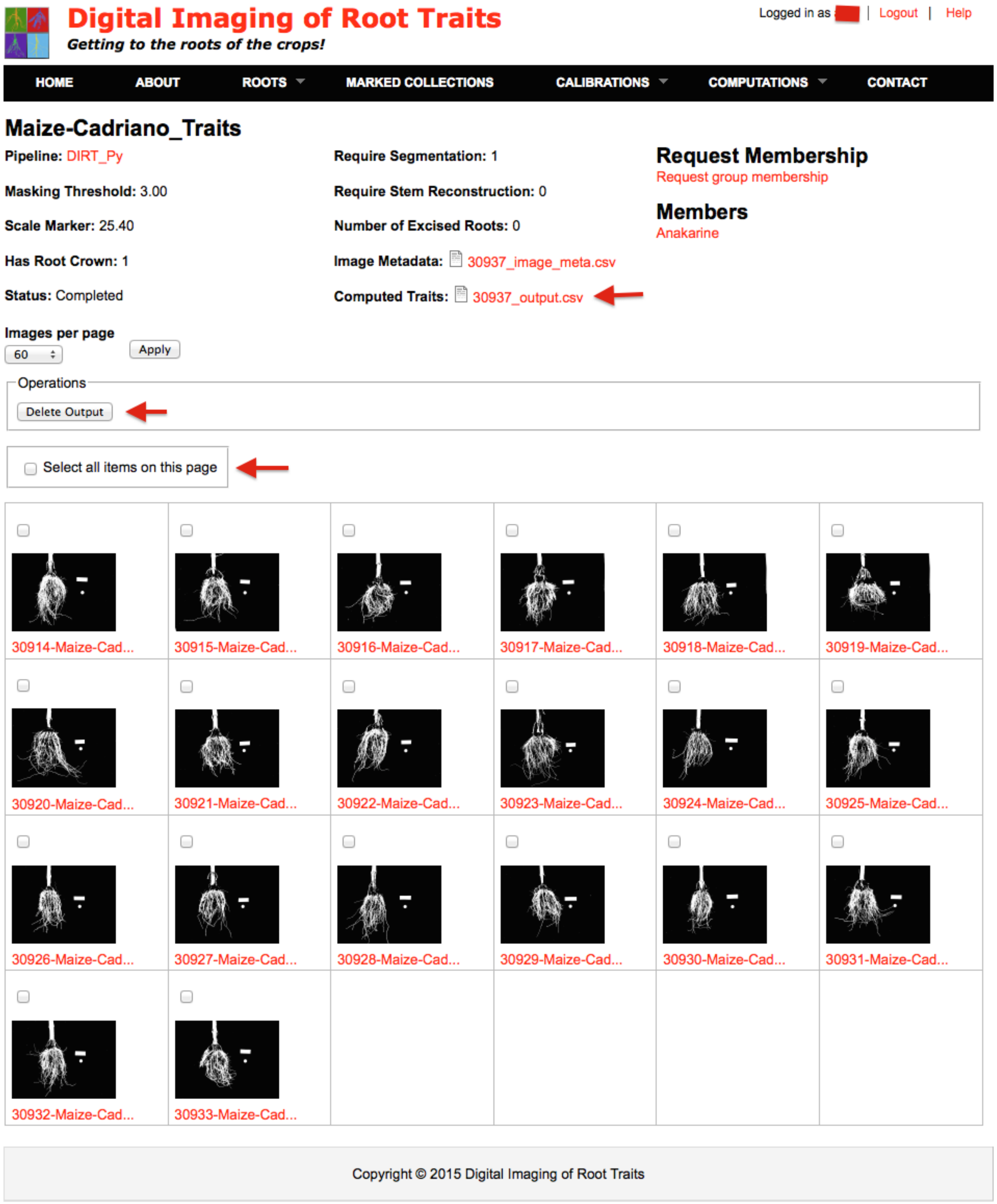
A user can cancel a submitted or running computation anytime during its execution. To cancel a running computation:

1. Go to the computations list page by clicking on the ‘COMPUTATIONS’ menu (**Figure 40**).
2. Select a computation whose status is either ‘Submitted’ or ‘Running’.
3. On the computations view page as shown **Figure 42** click on the ‘Cancel’ button.

Download computed RSA traits

Download computed traits of whole computation

After a computation has completed successfully, the computed traits are available for download in the form of a CSV file. The CSV file and image metadata file are available for download on the computation’s view page as shown in **Figure 43**.



Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT

Maize-Cadriano_Traits

Pipeline: DIRT_Py

Masking Threshold: 3.00

Scale Marker: 25.40

Has Root Crown: 1

Status: Completed

Require Segmentation: 1

Require Stem Reconstruction: 0

Number of Excised Roots: 0

Image Metadata: 30937_image_meta.csv

Computed Traits: 30937_output.csv

Request Membership

Request group membership

Members

Anakarine

Images per page

60

Apply
Operations
Delete Output

□ Select all items on this page

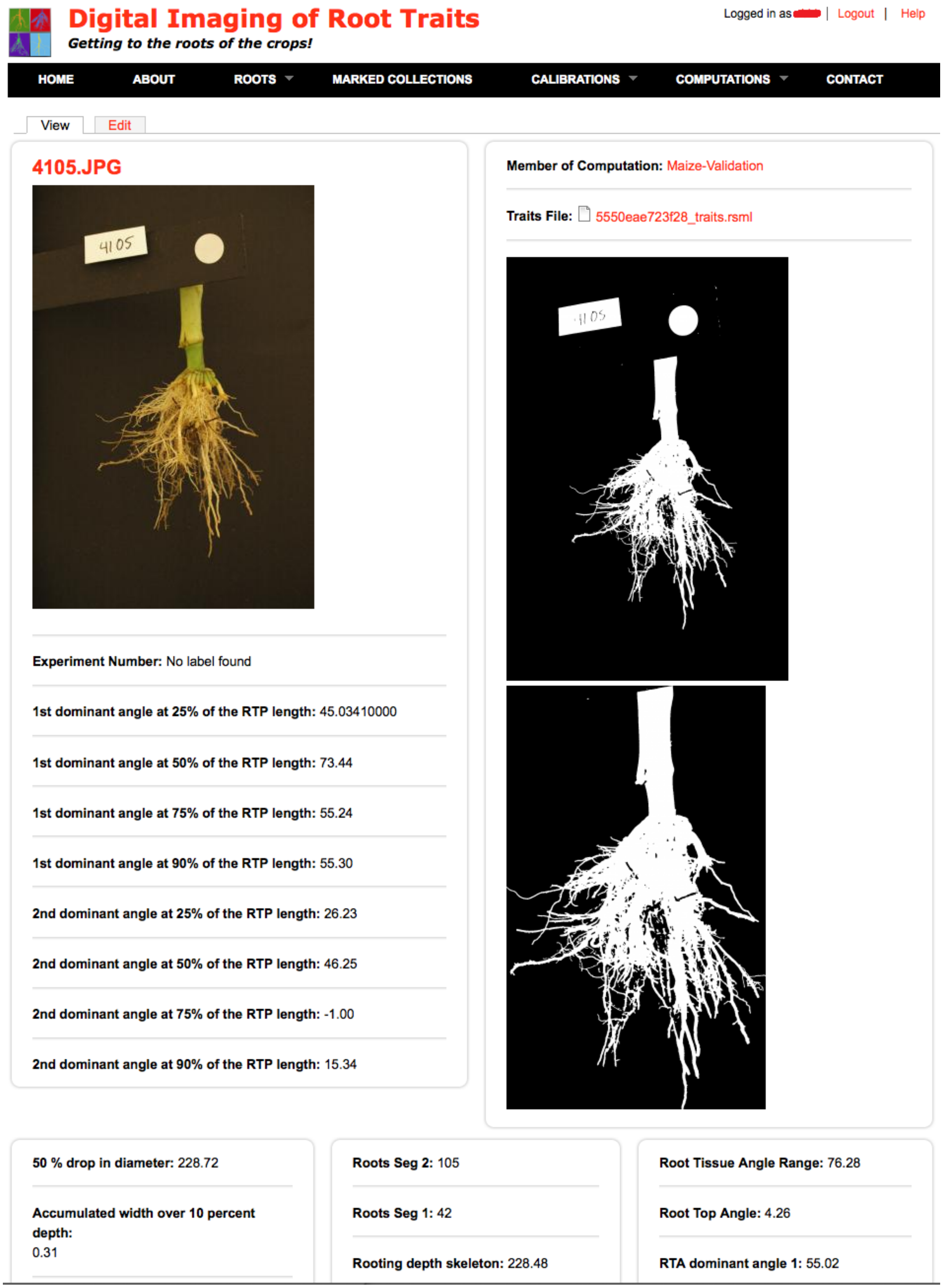
□ 30914-Maize-Cad...	□ 30915-Maize-Cad...	□ 30916-Maize-Cad...	□ 30917-Maize-Cad...	□ 30918-Maize-Cad...	□ 30919-Maize-Cad...
□ 30920-Maize-Cad...	□ 30921-Maize-Cad...	□ 30922-Maize-Cad...	□ 30923-Maize-Cad...	□ 30924-Maize-Cad...	□ 30925-Maize-Cad...
□ 30926-Maize-Cad...	□ 30927-Maize-Cad...	□ 30928-Maize-Cad...	□ 30929-Maize-Cad...	□ 30930-Maize-Cad...	□ 30931-Maize-Cad...
□ 30932-Maize-Cad...	□ 30933-Maize-Cad...				

Figure 43: Screenshot of the DIRT's interface of a completed computation. The user can visually inspect the masked images and download the computed traits and metadata. The red arrow indicates the fields that need user's attention/action.

Download computed traits for one image of a computation

Besides a CSV file containing the computed traits for the whole computation, each image is additionally associated with a RSML [92] file. To access and download the RSML file:

1. Select one by clicking on the link present under the masked image on the computation view interface (**Figure 43**).
2. It will navigate to the individual image's computation view page containing link to the RSML file (**Figure 44**).


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
View
Edit
4105.JPG
4105

Member of Computation: Maize-Validation

Traits File: 5550eae723f28_traits.rsml

Experiment Number: No label found

1st dominant angle at 25% of the RTP length: 45.03410000

1st dominant angle at 50% of the RTP length: 73.44

1st dominant angle at 75% of the RTP length: 55.24

1st dominant angle at 90% of the RTP length: 55.30

2nd dominant angle at 25% of the RTP length: 26.23

2nd dominant angle at 50% of the RTP length: 46.25

2nd dominant angle at 75% of the RTP length: -1.00

2nd dominant angle at 90% of the RTP length: 15.34

50 % drop in diameter: 228.72

Accumulated width over 10 percent depth: 0.31

Roots Seg 2: 105

Roots Seg 1: 42

Rooting depth skeleton: 228.48

Root Tissue Angle Range: 76.28

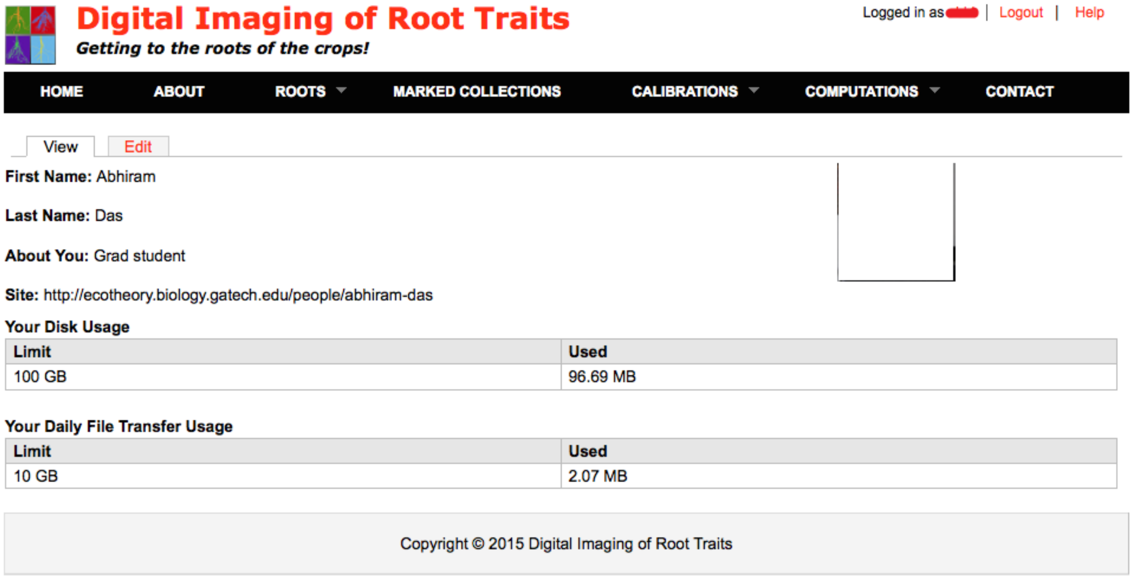
Root Top Angle: 4.26

RTA dominant angle 1: 55.02

Figure 44: Screenshot of the DIRT's interface showing computed traits of a single image. The computed traits are available for download in the form of a RSML file. Note, this figure is not showing all the traits, but all the computed traits can actually be seen on the site by scrolling down the page.

User quota

DIRT allocates a configurable storage and file transfer quota to its users. Each DIRT user on iPlant, has a total storage limit of 100 GB and daily computation (i.e. file transfer/image processed) limit of 10 GB. The user can access this information by visiting the user profile page (as shown **Figure 45**) by clicking on the user name at top right corner. As more iPlant resources become available to DIRT, the limit will be increased.


Digital Imaging of Root Traits
Getting to the roots of the crops!
Logged in as [redacted] | Logout | Help
HOME
ABOUT
ROOTS
MARKED COLLECTIONS
CALIBRATIONS
COMPUTATIONS
CONTACT
View
Edit

First Name: Abhiram

Last Name: Das

About You: Grad student

Site: http://ecotheory.biology.gatech.edu/people/abhiram-das

Your Disk Usage

Limit	Used
100 GB	96.69 MB

Your Daily File Transfer Usage

Limit	Used
10 GB	2.07 MB

Copyright © 2015 Digital Imaging of Root Traits

Figure 45: Screenshot of the DIRT's interface showing user profile details.

APPENDIX C

A DEVELOPER'S GUIDE TO DIRT

This appendix is adopted from the supplementary material of the publication Das A, Schneider H, Burridge J, Martinez Ascanio AK, Wojciechowski T, Topp CN, Lynch JP, Weitz JS, Bucksch A;DIRT: a high-throughput computing and collaboration platform for field-based plant phenomics; 2015,In Review.

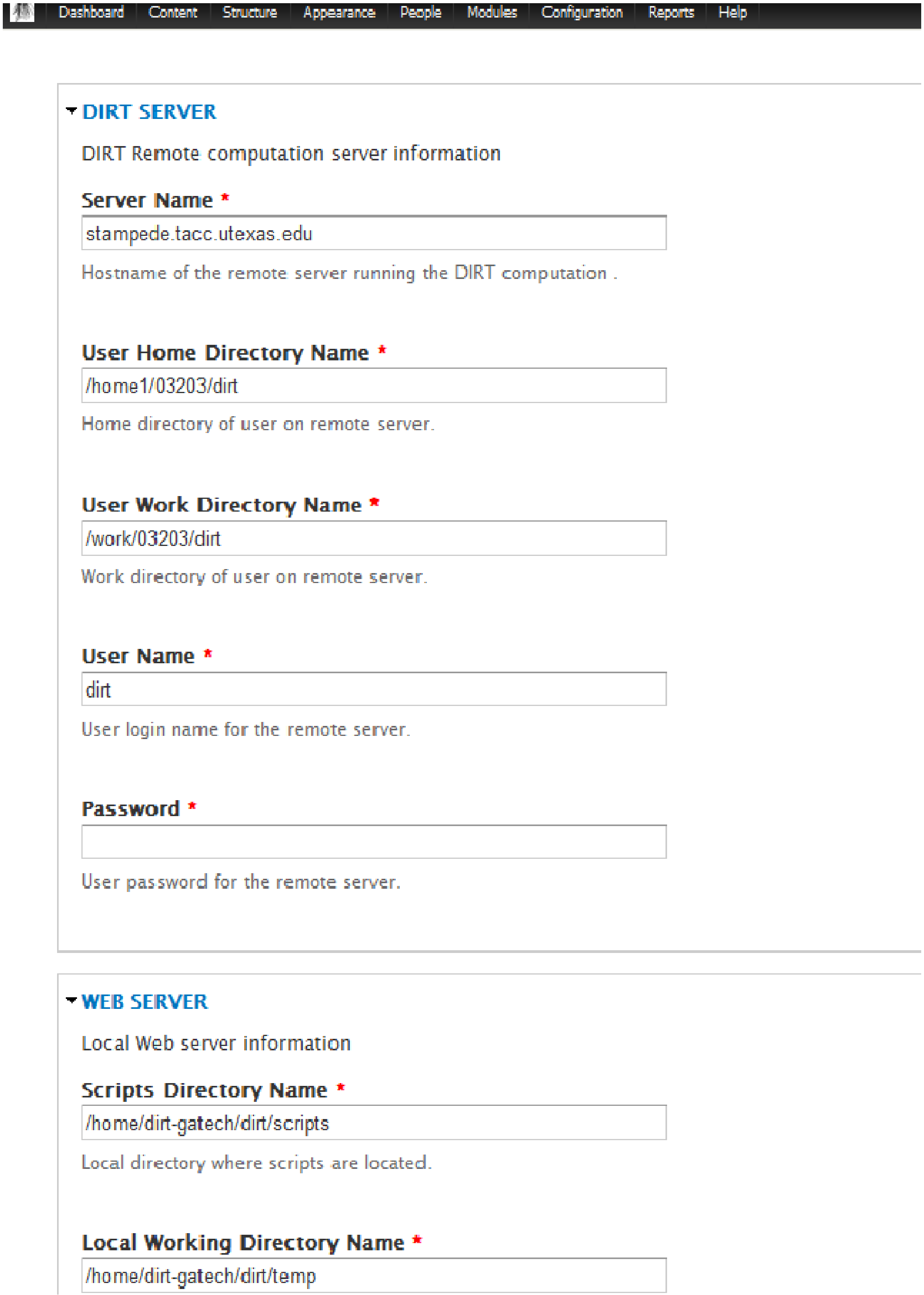
It describes how to install DIRT on a local proprietary computing environment, extend its functionality, add a new RSA trait computation pipeline and discusses current limitations of DIRT.

Installation guide in proprietary environments

DIRT is an open source platform developed on the open source content management system Drupal. Hence, DIRT can be installed on any proprietary environment and be extended to meet specific user requirements. For system requirements and installation instructions, we refer to Drupal's online documentation available at https://www.drupal.org/documentation/install.

1. Web server: To configure the web server follow the Drupal installation document mentioned in the previous paragraph. Once you have a Drupal instance running, download the DIRT source from http://dirt.iplantcollaborative.org/sites/default/dirt_files/dirt-src.tar and extract it to your system. This archive file contains the 'dirt.sql' file and 'public_html' directory. Load the 'dirt.sql' to your local database instance and copy the contents of the public_html to your local web root directory. Make sure that the 'settings.php' file is updated with correct database credentials to connect to your local database.

2. DIRT pipeline: Download DIRT pipeline from https://github.com/abucksch/DIRT and install and configure required software stack on both the web server and the grid computing platform.
3. Install and configure interface scripts: Log in to your local DIRT site as the administrator and go to 'DIRT Server Configuration' interface under 'Configuration' menu as shown in **Figure 46**. Enter your grid server and web server details and save. Similarly access 'DIRT Quota Configuration' interface (**Figure 47**) and enter the daily quota limit and save. Log in to your grid computing environment and download and extract this file http://dirt.iplantcollaborative.org/sites/default/dirt_files/dirt-server-scripts.tar. On extraction this file will create a directory called 'src'. Make sure that 'src' directory is directly under the server home directory as configured using the 'DIRT Server Configuration' page. The 'src' directory contains two sub-directories called 'python' and 'scripts'. Update 'dirt_job.slurm' and 'submit_dirt_jobs.sh' scripts in 'scripts' directory as per your local grid computing environment and scheduler. Log in to your web server system, go to the web server's script directory as configured on the 'DIRT Server Configuration' page and download and extract this http://dirt.iplantcollaborative.org/sites/default/dirt_files/dirt-web-scripts.tar file.



DIRT SERVER

DIRT Remote computation server information

Server Name *

stampede.tacc.utexas.edu

Hostname of the remote server running the DIRT computation .

User Home Directory Name *

/home1/03203/dirt

Home directory of user on remote server.

User Work Directory Name *

/work/03203/dirt

Work directory of user on remote server.

User Name *

dirt

User login name for the remote server.

Password *

User password for the remote server.

WEB SERVER

Local Web server information

Scripts Directory Name *

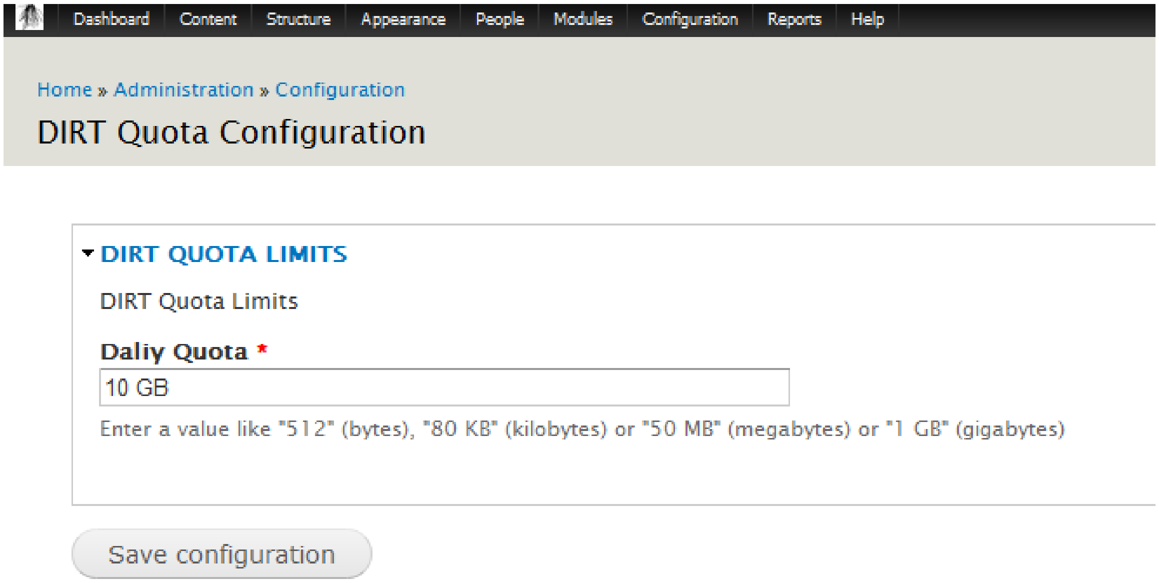
/home/dirt-gatech/dirt/scripts

Local directory where scripts are located.

Local Working Directory Name *

/home/dirt-gatech/dirt/temp

Figure 46: Screenshot of the DIRT's administration interface to configure grid computing platform information.



Home » Administration » Configuration

DIRT Quota Configuration

▾ DIRT QUOTA LIMITS

DIRT Quota Limits

Daliy Quota *

10 GB

Enter a value like "512" (bytes), "80 KB" (kilobytes) or "50 MB" (megabytes) or "1 GB" (gigabytes)

Figure 47: Screenshot of the DIRT's administration interface to configure user's daily computation quota information.

Extending the existing pipeline

Existing DIRT image processing pipeline can be extended to compute new RSA traits. To extend the pipeline take following steps:

1. Download the python source code from https://github.com/abucksch/DIRT and make necessary changes to your need.
2. Log in to Drupal as an administrator and update the existing content called 'DIRT_Py' to reflect the new traits.
3. Log in to Drupal as an administrator and add new fields corresponding to the new traits to Drupal content types 'Computation' and 'DIRT Output'.
4. Update the DIRT Drupal modules 'dirt_run_computation' and 'dirt_job_status' to handle new traits.

Adding a new pipeline

The DIRT platform can be extended to compute root traits from 2D images using a new pipeline. Following steps must be taken to add a new pipeline to the platform:

1. Develop the new pipeline in the language of your choice.

2. Install and configure required software stack on the grid computing platform. Place the scripts/executables under the ‘src’ directory on the server.
3. Develop required scripts to run the pipeline on the grid computing platform and place them under ‘src’ directory.
4. Log in to Drupal as an administrator and add a new content of type “Image Processing Pipeline”.
5. If the new pipeline compute new traits, log in to Drupal as an administrator and add new fields corresponding to the new traits to Drupal content types ‘Computation’ and ‘DIRT Output’.
6. Log in to Drupal as an administrator and add new fields corresponding to parameters of the new pipeline to the ‘Computation’ content type. Add validations to ‘Computation’ content type to show parameter fields corresponding to the pipeline.
7. Update the DIRT Drupal module ‘dirt_run_computation’ to check for the pipeline and call appropriate server scripts.
8. Update the DIRT Drupal modules ‘dirt_run_computation’ and ‘dirt_job_status’ to handle new traits if any.

Limitations

Even though DIRT is a high-throughput online scalable platform, it has technological constraints. The iPlant installation and the easy web access to high-throughput computing system restrict data upload and computation time. Furthermore the current algorithm restricts RSA trait computation to images taken according to DIRT imaging protocol [1]. In the following subsections we describe the limitations of the publicly available iPlant installation of DIRT. DIRT on iPlant installation can be accessed via any web browser; however it has been tested on Google Chrome (Version 43), Firefox (Version 38), Internet Explorer (Version 11) and Safari (Version 8). The web browser's privacy has to be set to accept third-party cookies and allow the site to run JavaScript.

File type and size

The iPlant installation of DIRT exclusively accepts image files of types png, gif, jpg and jpeg. While no restrictions are imposed on the image resolution, single image file is not allowed to exceed 1 GB.

Number of image files per upload

The batch upload to a root image collection is limited to 200 images at a time. As explained earlier in Appendix B, this restriction gives end-users easy and quick web experience, while taking into account the limitations of http POST protocol, browser support and internet speed.

Number of images per computation

The RSA trait computation pipeline runs on STAMPEDE [96] environment at TACC. Stampede limits wall-clock time of productions queue's to 48 hrs and the processor number to 4000. The priority of allocated resource for a computation is decided by the underlying Simple Linux Utility for Resource Management (SLURM) batch environment. Therefore, jobs requesting more resources may have to wait longer for resource allocation. DIRT estimates appropriate wall-clock time and processor needs for each submitted computation based on the image size and number. Currently, DIRT does not have a limit on the number of images that can be submitted per computation, but we recommend our users not to submit a computation of marked collection with more than 4000 images.

Calibration of one image at a time

The calibration functionality of DIRT is a synchronous process that computes mask images for six different threshold values per image simultaneously and uses web server memory. Depending upon the image size this process takes from seconds to couple of minutes to finish. Therefore, as of this release DIRT only allows the calibration of one image at a time. Users wanting to calibrate multiple images of a marked collection have to bear with this limitation. However, it is possible to calibrate multiple images (one at a time) and visually explore the threshold values.

APPENDIX D

A METHOD TO ESTIMATE ROOT ENTRY ANGLE FROM 3D ROOT IMAGES

In this appendix we present a method to estimate the angle of root segments with respect to the gravity, in a root system from reconstructed 3D images of rice roots grown in gellan gum medium that were used in an earlier publication [82].

Method

Most computer vision or image processing solutions applied to measuring plant structural properties from images broadly depend on segmentation and analysis of structures. Segmentation is the process by which boundaries of the objects of interest are identified from the image background. Many techniques and algorithms can be employed to differentiate foreground pixels from the background including: (i) thresholding of pixels based on frequency histogram, (ii) differentiating pixels based on Bayesian probability, (iii) differentiating pixels based on machine learning techniques like support vector machine or neural networks [108-111]. To measure and analyze structural properties of biological importance to plant and/or roots, the next important step is to identify the midline of the enclosed object. The process of identifying the midline of the elongated or connected object is called thinning or skeletonization. Similar to segmentation, many techniques and algorithms can be employed to obtain the midline of an elongated object. Popular techniques employs to obtain midline are: (i) medial axis transform, (ii) Voronoi diagram of the boundary points, (iii) layer by layer erosion called thinning, and (iv) Reeb graph.

In our case [112] the image data has already been segmented and we have voxel coordinates of the object of interest, i.e., a single connected component of a rice root. Our objective is to measure the angle each root segment makes with respect to the gravity. To get the midline of the root system, we use a simple thinning algorithm, where we loop through each slice of the 3D image from top to bottom and compute the midpoint of each label or connected component in the slice. We have validated this method against a ground truth model (GTM) 3D root structure designed *in silico*, printed in 3D from resin (**Figure 48-A**), imaged and reconstructed (**Figure 48-B**).

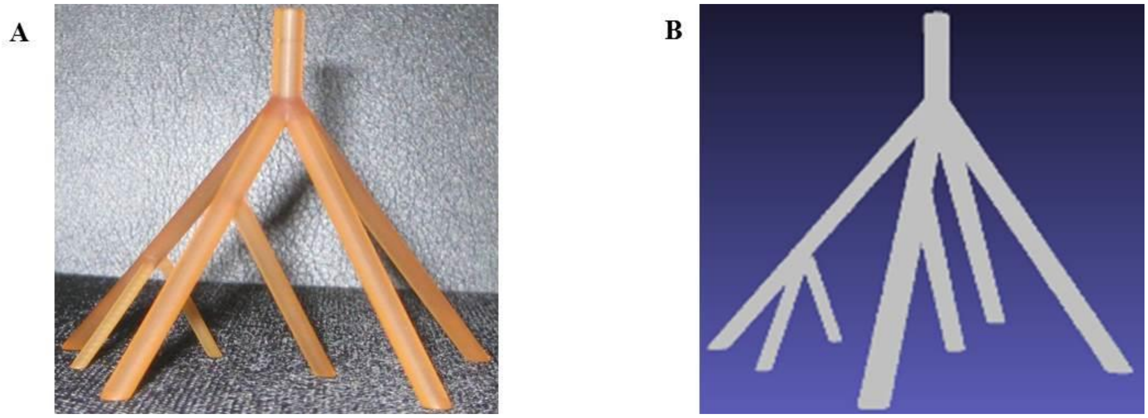


Figure 48: (A) Image of the printed 3D ground truth model designed in silico, (B) Ground truth model reconstructed from the image of the printed model in (A)

Principles

The above mentioned method is implemented in Matlab using standard image processing and statistics toolbox. The basic principle of the method is to shave the x and y coordinates of a voxel from both directions to get one voxel width representation of the original image. This is achieved by taking the floor value of the mean of x and y coordinates of the voxels of a connected component in each slice of the image. Once we have the thinned image, angle of each connected component to the normal vector of x-y plane is computed by predefined Matlab functions. The method can be summarized in following 3 major steps:

1. Thin voxel data

Initialize a 3D zero matrix of the size of the original image to hold the thin image.

Iterate over z coordinates of the 3D matrix initialized in the previous step.

```
Get the slice of matrix for each z-coordinate (Figure 49-A).
Find the number of connected components in each slice.
Iterate over each connected component in the slice (Figure 49-B)
    Compute floor value of mean of x and y coordinate of the connected component.
    Set the corresponding coordinates of the 3D zero matrix to 1 (Figure 49-C).
```

2. Compute angle of each component (root segment)

```
    Use the thinned 3D matrix of the previous step as input to this step.
    Get number of connected components in the thinned 3D matrix.
    Iterate over the number of connected components.
        Get the matrix representing list of voxels in the component.
        Compute the angle the voxels make w.r.t to the normal vector to the x-y plane.
```

3. Angle measurement (Standard Matlab function is used to compute the angle)

```
    3D matrix representing the voxels of a component is the input to this method.
    Compute the mean of columns to do the fit without a constant term.
    Subtract the column mean from the main matrix.
    Use the svd() function on the subtracted mean to get the regression as a planner fit.
    Get the fit line vector and compute it angle to the normal vector to the x-y plane.
```

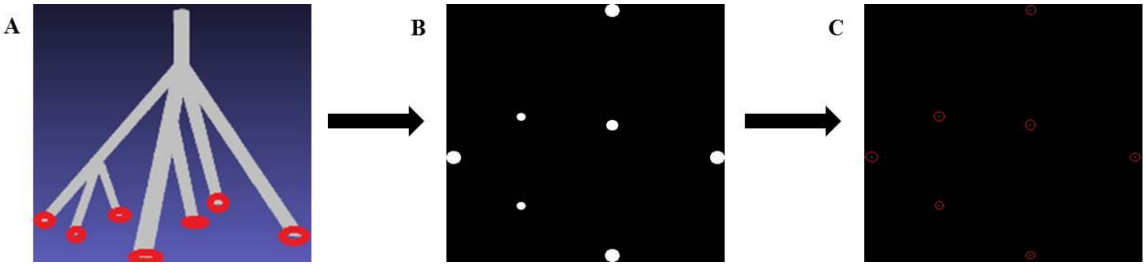


Figure 49: Method to get the thinned image from the original 3D image. (A) In the original 3D image the red marks represent the bottom most slice of the image. (B) Top down view of the components or labels of the bottom most slice represented in Figure A. (C) Top down view of the thinned components or labels in the bottom most slice of the image represented in Figure B.

Method Validation

This section presents the preliminary results obtained by applying the above method to the reconstructed rice root system data i.e. published data [82].

Validation of method on ground truth model (GTM)

The ground truth model used in the published data [82] is used to validate the angle measurements of the roots. The angle of each root is measured manually from the resin model matches the measurements from this method. This method works well with the GTM as observed in **Figure 50**. The angle measurements are not affected even though root segments are disconnected at the branching points.

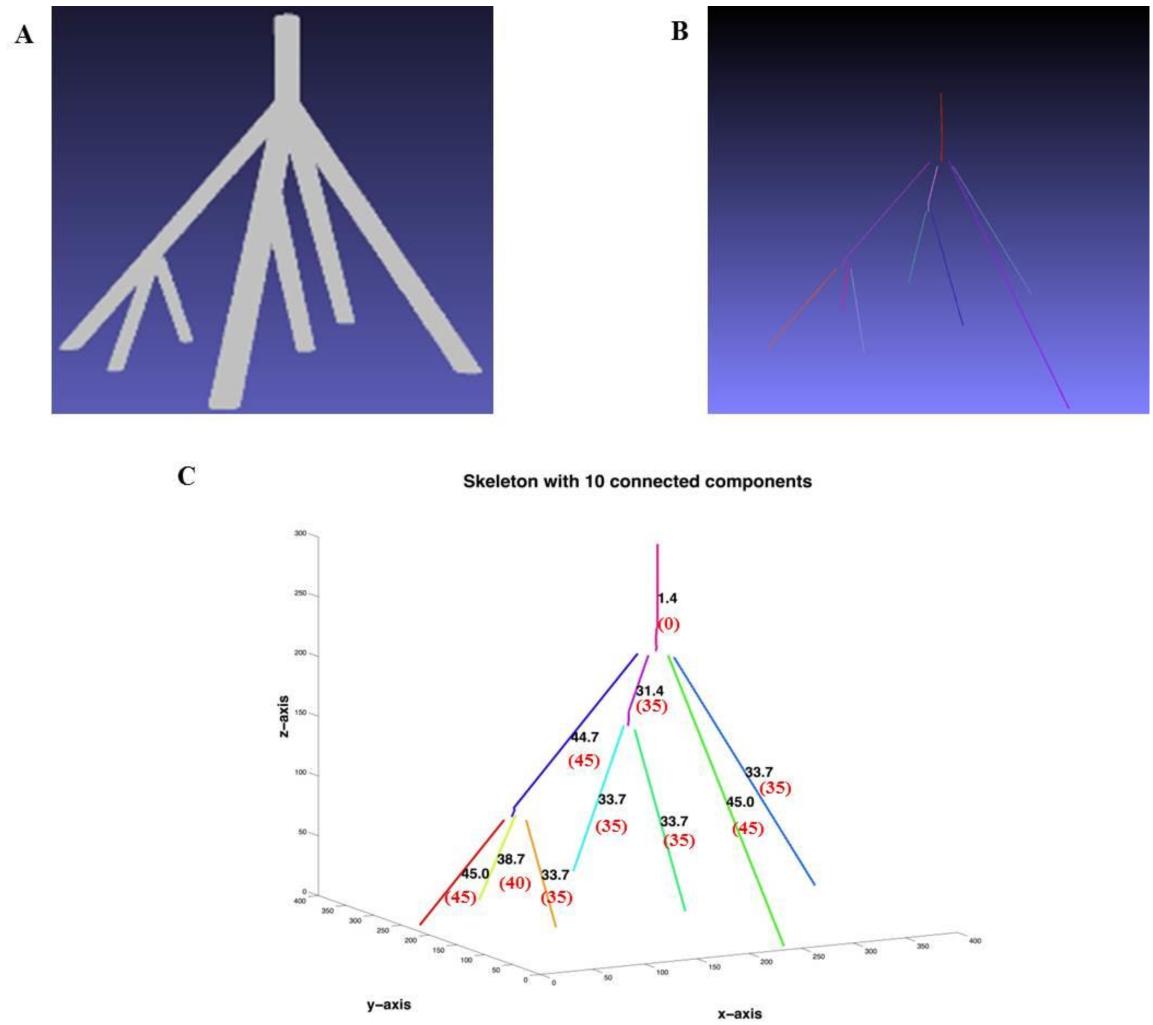


Figure 50: (A) Image of the reconstructed ground truth model. (B) Thinned image of the reconstructed ground truth model. (C) Angle measurement of each root segment of the thinned image. The values shown within brackets are the manual measurement of angles in degrees from the printed 3D ground truth model.

Table 9: Table showing the comparison of angles (degrees) between the automatic and manual measurements. Standard deviations were calculated by bootstrap resampling of the voxels for each fit. The – in the last column denotes the absence of variability in 1000 bootstrap resamples.

Segment #	Manual Measurement	Automatic Measurement	Standard Deviation
1	45	45	-
2	40	38.7	0.046
3	35	33.7	0.049
4	35	33.7	0.023
5	35	33.7	0.023
6	45	45	-
7	35	33.7	0.013
8	45	44.7	0.086
9	35	31.4	0.476

Table 9: continued

10	0	1.4	0.196

System Data

The method was applied on the two different rice varieties: (i) Azucena parental line (**Figure 51-A and Figure 52-A**) and (ii) recombinant inbred line (RIL) of Azucena and Bala (**Figure 51-B and Figure 52-B**) and distribution of root angle was observed. Distribution of angles differs (**Figure 53**) with respect to the root genotype. The angle measurements do comply with the 3D structure of the roots i.e. in Azucena most of the roots are close to the central axis whereas in the inbred line most are away from the center. In some aspect these angle measurements do comply with previous investigation [42] i.e. the high measure of root initiation angle in Azucena in the previous study.

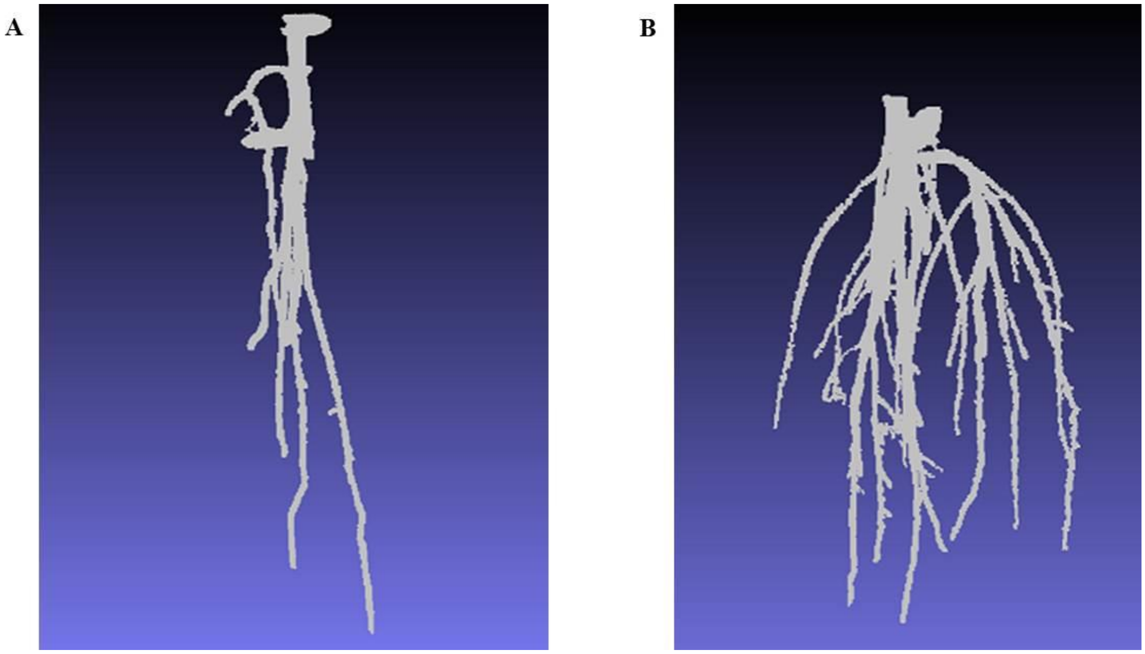


Figure 51: (A) Reconstructed 3D image of Azucena rice genotype. (B) Reconstructed 3D image of recombinant inbred line of Azucena and Bala.

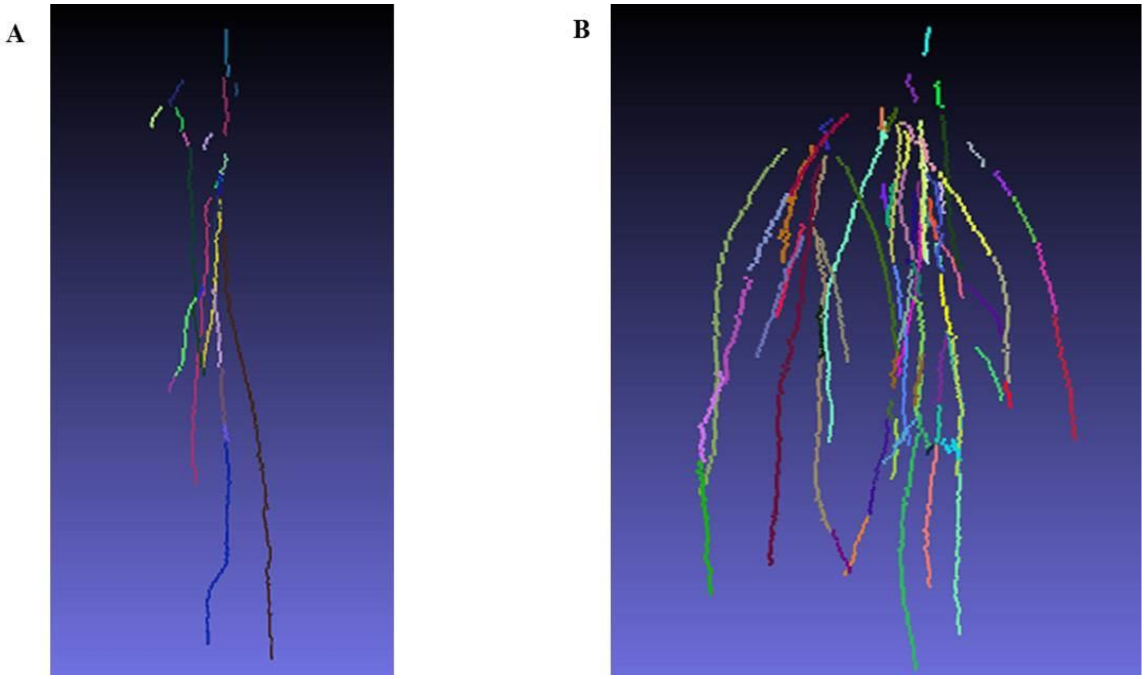


Figure 52: (A) Thinned image of Azucena (B) Thinned image of the recombinant inbred line of Azucena and Bala

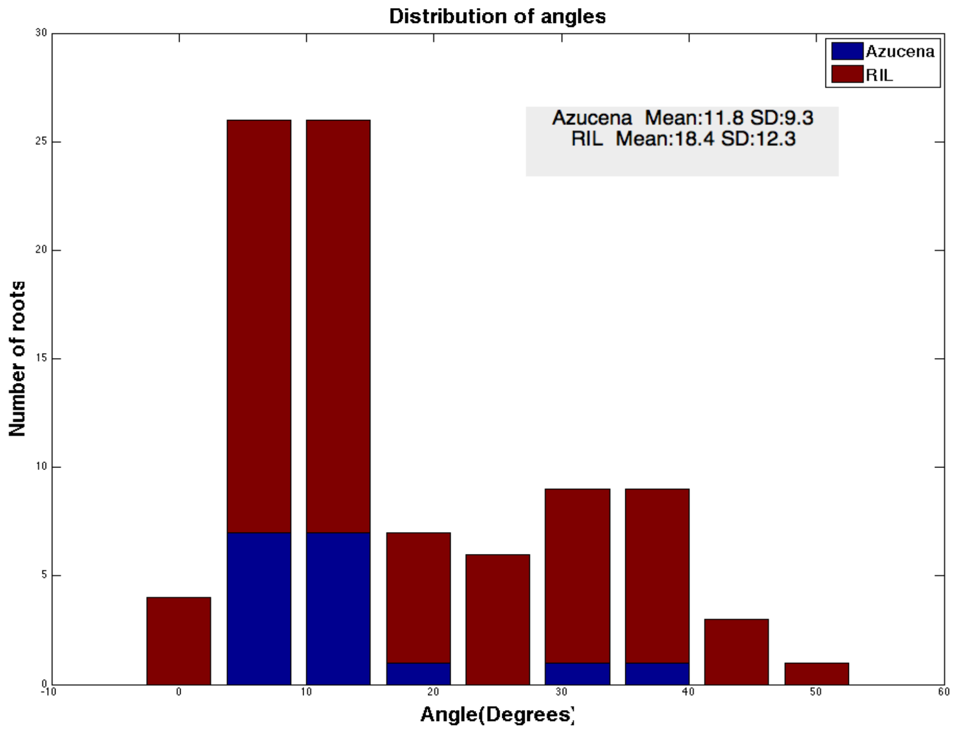


Figure 53: Distribution of root angles in Azucena and the RIL.

REFERENCES

[1] A. Bucksch, J. Burridge, L. M. York, A. Das, E. Nord, J. S. Weitz, and J. P. Lynch, “Image-Based High-Throughput Field Phenotyping of Crop Roots,” *Plant Physiology,* vol. 166, no. 2, pp. 470-486, 2014.

[2] A. Byron, A. Berry, N. Haug, J. Eaton, J. Walker, and J. Robbins, *Using Drupal*: O'Reilly, 2008.

[3] “Division, U. N. P. World population prospects, the 2010 revision., (Population Estimates andProjections Section., 2010),” 2010.

[4] W. Bank, *Agriculture for Development*, The World Bank, Washington, D.C., 2007.

[5] J. P. Lynch, “Roots of the second green revolution,” *Australian Journal Of Botany,* vol. 55, no. 5, pp. 493-512, 2007.

[6] K. G. Cassman, A. Dobermann, and D. T. Walters, “Agroecosystems, nitrogen-use efficiency, and nitrogen management,” *Ambio,* vol. 31, no. 2, pp. 132-140, 2002.

[7] S. B. St Clair, and J. P. Lynch, “The opening of Pandora's Box: climate change impacts on soil fertility and crop nutrition in developing countries,” *Plant and Soil,* vol. 335, no. 1-2, pp. 101-115, 2010.

[8] T. J. Brodribb, T. S. Feild, and G. J. Jordan, “Leaf maximum photosynthetic rate and venation are linked by hydraulics(1[W][OA]),” *Plant Physiology,* vol. 144, no. 4, pp. 1890-1898, 2007.

[9] C. K. Boyce, T. J. Brodribb, T. S. Feild, and M. A. Zwieniecki, “Angiosperm leaf vein evolution was physiologically and environmentally transformative,” *Proceedings of the Royal Society B-Biological Sciences,* vol. 276, no. 1663, pp. 1771-1776, 2009.

[10] C. A. Price, S. Wing, and J. S. Weitz, “Scaling and structure of dicotyledonous leaf venation networks,” *Ecology Letters,* vol. 15, no. 2, pp. 87-95, 2012.

[11] J. C. Bates, “A method for clearing leaves,” *American Naturalist,* vol. 65, pp. 288-288, 1931.

[12] Y. Waisel, A. Eshel, T. Beeckman, and U. Kafkafi, *Plant roots: the hidden half*: CRC Press, 2002.

[13] D. Robinson, “The Responses of Plants to Nonuniform Supplies of Nutrients,” *New Phytologist,* vol. 127, no. 4, pp. 635-674, 1994.

[14] J. Lynch, “Root Architecture and Plant Productivity,” *Plant Physiology,* vol. 109, no. 1, pp. 7-13, 1995.

[15] A. M. Bonser, J. Lynch, and S. Snapp, “Effect of phosphorus deficiency on growth angle of basal roots in Phaseolus vulgaris,” *New Phytologist,* vol. 132, no. 2, pp. 281-288, 1996.

[16] Z. Y. Ge, G. Rubio, and J. P. Lynch, “The importance of root gravitropism for inter-root competition and phosphorus acquisition efficiency: results from a geometric simulation model,” *Plant and Soil,* vol. 218, no. 1-2, pp. 159-171, 2000.

[17] T. L. D. Brooks, N. D. Miller, and E. P. Spalding, “Plasticity of Arabidopsis Root Gravitropism throughout a Multidimensional Condition Space Quantified by Automated Image Analysis,” *Plant Physiology,* vol. 152, no. 1, pp. 206-216, 2010.

[18] A. S. Iyer-Pascuzzi, O. Symonova, Y. Mileyko, Y. L. Hao, H. Belcher, J. Harer, J. S. Weitz, and P. N. Benfey, “Imaging and Analysis Platform for Automatic Phenotyping and Trait Ranking of Plant Root Systems,” *Plant Physiology,* vol. 152, no. 3, pp. 1148-1157, 2010.

[19] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker, “Cytoscape: A software environment for integrated models of biomolecular interaction networks,” *Genome Research,* vol. 13, no. 11, pp. 2498-2504, 2003.

[20] M. T. Gastner, and M. E. J. Newman, “The spatial structure of networks,” *European Physical Journal B,* vol. 49, no. 2, pp. 247-252, 2006.

[21] M. E. J. Newman, *Networks: An Introduction*: Oxford University Press, 2010.

[22] P. Armengaud, K. Zambaux, A. Hills, R. Sulpice, R. J. Pattison, M. R. Blatt, and A. Amtmann, “EZ-Rhizo: integrated software for the fast and accurate measurement of root system architecture,” *Plant Journal,* vol. 57, no. 5, pp. 945-956, 2009.

[23] J. Le Bot, V. Serra, J. Fabre, X. Draye, S. Adamowicz, and L. Pages, “DART: a software to analyse root system architecture and development from captured images,” *Plant and Soil,* vol. 326, no. 1-2, pp. 261-273, 2010.

[24] G. Lobet, L. Pages, and X. Draye, “A Novel Image-Analysis Toolbox Enabling Quantitative Analysis of Root System Architecture,” *Plant Physiology,* vol. 157, no. 1, pp. 29-39, 2011.

[25] T. Galkovskyi, Y. Mileyko, A. Bucksch, B. Moore, O. Symonova, C. Price, C. Topp , A. Iyer-Pascuzzi, P. Zurek, S. Fang, J. Harer, P. Benfey, and J. Weitz, “GiA Roots: Software for the High Throughput Analysis of Plant Root System Architecture,” *BMC Plant Biology,* vol. 12, no. 116, 2012.

[26] J. M. Zhu, P. A. Ingram, P. N. Benfey, and T. Elich, “From lab to field, new approaches to phenotyping root system architecture,” *Current Opinion in Plant Biology,* vol. 14, no. 3, pp. 310-317, 2011.

[27] S. Trachsel, S. M. Kaeppler, K. M. Brown, and J. P. Lynch, “Shovelomics: high throughput phenotyping of maize (Zea mays L.) root architecture in the field,” *Plant and Soil,* vol. 341, no. 1-2, pp. 75-87, 2011.

[28] D. Tilman, C. Balzer, J. Hill, and B. L. Befort, “Global food demand and the sustainable intensification of agriculture,” *Proceedings of the National Academy of Sciences,* vol. 108, no. 50, pp. 20260-20264, 2011.

[29] S. M. de Sousa, R. T. Clark, F. F. Mendes, A. C. de Oliveira, M. J. V. de Vasconcelos, S. N. Parentoni, L. V. Kochian, C. T. Guimaraes, and J. V. Magalhaes, “A role for root morphology and related candidate genes in P acquisition efficiency in maize,” *Functional Plant Biology,* vol. 39, no. 10-11, pp. 925-935, 2012.

[30] M. D. McMullen, S. Kresovich, H. S. Villeda, P. Bradbury, H. H. Li, Q. Sun, S. Flint-Garcia, J. Thornsberry, C. Acharya, C. Bottoms, P. Brown, C. Browne, M. Eller, K. Guill, C. Harjes, D. Kroon, N. Lepak, S. E. Mitchell, B. Peterson, G. Pressoir, S. Romero, M. O. Rosas, S. Salvo, H. Yates, M. Hanson, E. Jones, S. Smith, J. C. Glaubitz, M. Goodman, D. Ware, J. B. Holland, and E. S. Buckler, “Genetic Properties of the Maize Nested Association Mapping Population,” *Science,* vol. 325, no. 5941, pp. 737-740, 2009.

[31] J. L. Araus, and J. E. Cairns, “Field high-throughput phenotyping: the new crop breeding frontier,” *Trends in Plant Science,* vol. 19, no. 1, pp. 52-61, 2014.

[32] A. P. Wasson, R. A. Richards, R. Chatrath, S. C. Misra, S. V. S. Prasad, G. J. Rebetzke, J. A. Kirkegaard, J. Christopher, and M. Watt, “Traits and selection strategies to improve root systems and water uptake in water-limited wheat crops,” *Journal of Experimental Botany,* vol. 63, no. 9, pp. 3485-3498, 2012.

[33] R. T. Furbank, and M. Tester, “Phenomics - technologies to relieve the phenotyping bottleneck,” *Trends in Plant Science,* vol. 16, no. 12, pp. 635-644, 2011.

[34] G. Rubio, T. Walk, Z. Y. Ge, X. L. Yan, H. Liao, and J. P. Lynch, “Root gravitropism and below-ground competition among neighbouring plants: A modelling approach,” *Annals of Botany,* vol. 88, no. 5, pp. 929-940, 2001.

[35] K. Boonsirichai, C. Guan, R. Chen, and P. H. Masson, "Root gravitropism: An experimental tool to investigate basic cellular and molecular processes underlying mechanosensing and signal transmission in plants," *Annual Review of Plant Biology,* vol. 53, pp. 421-447, 2002.

[36] E. B. Blancaflor, J. M. Fasano, and S. Gilroy, "Mapping the functional roles of cap cells in the response of Arabidopsis primary roots to gravity," *Plant Physiology,* vol. 116, no. 1, pp. 213-222, 1998.

[37] J. Z. Kiss, "Mechanisms of the early phases of plant gravitropism," *Critical Reviews in Plant Sciences,* vol. 19, no. 6, pp. 551-573, 2000.

[38] D. R. Lewis, N. D. Miller, B. L. Splitt, G. S. Wu, and E. P. Spalding, "Separating the roles of acropetal and basipetal auxin transport on gravitropism with mutations in two Arabidopsis Multidrug Resistance-Like ABC transporter genes," *Plant Cell,* vol. 19, no. 6, pp. 1838-1850, 2007.

[39] N. D. Miller, B. M. Parks, and E. P. Spalding, "Computer-vision analysis of seedling responses to light and gravity," *Plant Journal,* vol. 52, no. 2, pp. 374-381, 2007.

[40] A. Rahman, M. Takahashi, K. Shibasaki, S. A. Wu, T. Inaba, S. Tsurumi, and T. I. Baskin, "Gravitropism of Arabidopsis thaliana Roots Requires the Polarization of PIN2 toward the Root Tip in Meristematic Cortical Cells," *Plant Cell,* vol. 22, no. 6, pp. 1762-1776, 2010.

[41] J. T. Arredondo, and D. A. Johnson, "Allometry of root branching and its relationship to root morphological and functional traits in three range grasses," *Journal of Experimental Botany,* vol. 62, no. 15, pp. 5581-5594, 2011.

[42] R. T. Clark, R. B. MacCurdy, J. K. Jung, J. E. Shaff, S. R. McCouch, D. J. Aneshansley, and L. V. Kochian, "Three-Dimensional Root Phenotyping with a Novel Imaging and Software Platform," *Plant Physiology,* vol. 156, no. 2, pp. 455-465, 2011.

[43] J. H. Shi, X. Hao, Z. C. Wu, and P. Wu, "A new genetic factor for root gravitropism in rice (Oryza sativa L.)," *Journal of Zhejiang University-Science B,* vol. 10, no. 10, pp. 777-783, 2009.

[44] D. C. D. B. Ellis, L.J.Hickey, K.R. Johnson, J.D. Mitchel, P. Wilf, S. L. Wing, *Manual of Leaf Architecture*: Cornell University Press, 2009.

[45] L. Sack, and N. M. Holbrook, "Leaf hydraulics," *Annual Review of Plant Biology,* vol. 57, pp. 361-381, 2006.

[46] U. Niinemets, A. Portsmuth, and M. Tobias, "Leaf size modifies support biomass distribution among stems, petioles and mid-ribs in temperate plants," *New Phytologist,* vol. 171, no. 1, pp. 91-104, 2006.

[47] U. Niinemets, A. Portsmuth, and M. Tobias, “Leaf shape and venation pattern alter the support investments within leaf lamina in temperate species: a neglected source of leaf physiological differentiation?,” *Functional Ecology,* vol. 21, no. 1, pp. 28-40, 2007.

[48] T. J. Brodribb, and T. S. Feild, “Stem hydraulic supply is linked to leaf photosynthetic capacity: evidence from New Caledonian and Tasmanian rainforests,” *Plant Cell and Environment,* vol. 23, no. 12, pp. 1381-1388, 2000.

[49] T. J. Brodribb, and T. S. Feild, “Leaf hydraulic evolution led a surge in leaf photosynthetic capacity during early angiosperm diversification,” *Ecology Letters,* vol. 13, no. 2, pp. 175-183, 2010.

[50] T. S. Feild, T. J. Brodribb, A. Iglesias, D. S. Chatelet, A. Baresch, G. R. Upchurch, B. Gomez, B. A. R. Mohr, C. Coiffard, J. Kvacek, and C. Jaramillo, “Fossil evidence for Cretaceous escalation in angiosperm leaf vein evolution,” *Proceedings of the National Academy of Sciences of the United States of America,* vol. 108, no. 20, pp. 8363-8366, 2011.

[51] J. A. Raven, “The Evolution of Vascular Plants in Relation to Quantitative Functioning of Dead Water-Conducting Cells and Stomata,” *Biological Reviews of the Cambridge Philosophical Society,* vol. 68, no. 3, pp. 337-363, 1993.

[52] L. Sack, M. T. Tyree, and N. M. Holbrook, “Leaf hydraulic architecture correlates with regeneration irradiance in tropical rainforest trees,” *New Phytologist,* vol. 167, no. 2, pp. 403-413, 2005.

[53] U. Kull, and A. Herbig, “Leaf Venation of Angiosperms - Form and Evolution,” *Naturwissenschaften,* vol. 82, no. 10, pp. 441-451, 1995.

[54] L. Sack, C. Scoffoni, A. D. McKown, K. Frole, M. Rawls, J. C. Havran, H. Tran, and T. Tran, “Developmentally based scaling of leaf venation architecture explains global ecological patterns,” *Nature Communications,* vol. 3, 2012.

[55] L. Sack, E. M. Dietrich, C. M. Streeter, D. Sanchez-Gomez, and N. M. Holbrook, “Leaf palmate venation and vascular redundancy confer tolerance of hydraulic disruption,” *Proceedings of the National Academy of Sciences of the United States of America,* vol. 105, no. 5, pp. 1567-1572, 2008.

[56] R. O. Gardner, “Overview of Botanical Clearing Technique,” *Stain Technology,* vol. 50, no. 2, pp. 99-105, 1975.

[57] “Smitsonian Institution. http://www.si.edu/. [Accessed 30 May 2013].”

[58] “Yale Peabody Museum. http://peabody.yale.edu/. [Accessed 30 May 2013].”

[59] “New York Botanical Garden. http://www.nybg.org/. [Accessed 30 May 2013].”

[60] "UC Berkeley Cleared Leaf Collections. http://www.ucmp.berkeley.edu/collections/plants/clearedleaf.html. [Accessed 30 May 2013]."

[61] "CalPhotos: A database of photos of plants, animals, habitats and other natural history subjects. http://calphotos.berkeley.edu/. [Accessed 04 Mar 2013]."

[62] "Morphbank :: Biological Imaging. http://www.morphbank.net. [Accessed 03 Mar 2013]."

[63] "The PLANTS Database. http://plants.usda.gov. [Accessed 03 Mar 2013]."

[64] "Encyclopedia of Life. http://www.eol.org. [Accessed 02 Mar 2013]."

[65] K. Kvilekval, D. Fedorov, B. Obara, A. Singh, and B. S. Manjunath, "Bisque: a platform for bioimage analysis and management," *Bioinformatics,* vol. 26, no. 4, pp. 544-552, 2010.

[66] R. A. Jorgensen, L. Stein, S. Rain, G. Andrews, and V. Chandler, "The iPlant collaborative: A cyberinfrastructure-centered community for a new plant biology," *In Vitro Cellular & Developmental Biology-Animal,* vol. 44, pp. S26-S26, Spr, 2008.

[67] "Drupal. https://www.drupal.org/ [Accessed 16 June 2015]."

[68] "The Apache HTTP Server. http://httpd.apache.org/. [Accessed 03 Mar 2013]."

[69] "MySQL Database Server. http://www.mysql.com/. [Accessed 04 Mar 2013]."

[70] "PHP. http://php.net/. [Accessed 03 Mar 2013]."

[71] C. A. Price, O. Symonova, Y. Mileyko, T. Hilley, and J. S. Weitz, "Leaf Extraction and Analysis Framework Graphical User Interface: Segmenting and Analyzing the Structure of Leaf Veins and Areoles," *Plant Physiology,* vol. 155, no. 1, pp. 236-245, 2011.

[72] M. Bylesjo, V. Segura, R. Y. Soolanayakanahally, A. M. Rae, J. Trygg, P. Gustafsson, S. Jansson, and N. R. Street, "LAMINA: a tool for rapid quantification of leaf size and shape parameters," *Bmc Plant Biology,* vol. 8, 2008.

[73] S. Dhondt, D. Van Haerenborgh, C. Van Cauwenbergh, R. M. H. Merks, W. Philips, G. T. S. Beemster, and D. Inze, "Quantitative analysis of venation patterns of Arabidopsis leaves by supervised image analysis," *Plant Journal,* vol. 69, no. 3, pp. 553-563, 2012.

[74] B. Blonder, C. Violle, and B. J. Enquist, “Assessing the causes and scales of the leaf economics spectrum using venation networks in Populus tremuloides,” *Journal of Ecology,* vol. 101, no. 4, pp. 981-989, 2013.

[75] B. Blonder, C. Violle, L. P. Bentley, and B. J. Enquist, “Venation networks and the origin of the leaf economics spectrum,” *Ecology Letters,* vol. 14, no. 2, pp. 91-100, 2011.

[76] “iRODS. http://irods.org. [Accessed 16 June 2015].”

[77] OECD, *OECD Environmental Outlook to 2030,* Paris, France: OECD Publishing, 2008.

[78] J. P. Lynch, “Turner review no. 14. Roots of the second green revolution,” *Australian Journal of Botany,* vol. 55, no. 5, pp. 493-512, 2007.

[79] F. Fiorani, and U. Schurr, “Future scenarios for plant phenotyping,” *Annual review of plant biology,* vol. 64, pp. 267-291, 2013.

[80] E. Finkel, “IMAGING With 'Phenomics,' Plant Scientists Hope to Shift Breeding Into Overdrive,” *Science,* vol. 325, no. 5939, pp. 380-381, 2009.

[81] N. Fahlgren, M. A. Gehan, and I. Baxter, “Lights, camera, action: high-throughput plant phenotyping is ready for a close-up,” *Current Opinion in Plant Biology,* vol. 24, pp. 93-99, 2015.

[82] C. N. Topp, A. S. Iyer-Pascuzzi, J. T. Anderson, C. R. Lee, P. R. Zurek, O. Symonova, Y. Zheng, A. Bucksch, Y. Mileyko, T. Galkovskyi, B. T. Moore, J. Harer, H. Edelsbrunner, T. Mitchell-Olds, J. S. Weitz, and P. N. Benfey, “3D phenotyping and quantitative trait locus mapping identify core regions of the rice genome controlling root architecture,” *Proceedings of the National Academy of Sciences of the United States of America,* vol. 110, no. 18, pp. E1695-E1704, 2013.

[83] H. Downie, N. Holden, W. Otten, A. J. Spiers, T. A. Valentine, and L. X. Dupuy, “Transparent soil for imaging the rhizosphere,” *PLoS One,* vol. 7, no. 9, pp. e44276, 2012.

[84] G. Lobet, X. Draye, and C. Perilleux, “An online database for plant image analysis software tools,” *Plant Methods,* vol. 9, 2013.

[85] “Drupal. http://drupal.org/. [Accessed 02 Mar 2013],” vol. 2013, pp. Drupal is an open source content management platform powering millions of websites and applications. It’s built, used, and supported by an active and diverse community of people around the world.

[86] “Agave API. http://agaveapi.co. [Accessed 16 June 2015].”

[87] R. Dooley, M. Vaughn, S. Terry, D. Stanzione, E. Skidmore, and N. Merchant, "Your Data, Your Way (The iPlant Foundation API Data Services)," *http://www.cse.buffalo.edu/faculty/tkosar/datacloud2012/papers/datacloud2012_paper_3.pdf [Accessed 16 June 2015]*.

[88] R. Dooley, M. Vaughn, D. Stanzione, S. Terry, and E. Skidmore, "Software-as-a-service: the iPlant foundation API," *In 5th IEEE Workshop on Many-Task Computing on Grids and Supercomputers (MTAGS)*, 2012.

[89] S. A. Goff, M. Vaughn, S. McKay, E. Lyons, A. E. Stapleton, D. Gessler, N. Matasci, L. Wang, M. Hanlon, and A. Lenards, "The iPlant collaborative: cyberinfrastructure for plant biology," *Frontiers in plant science*, vol. 2, 2011.

[90] T. Colombi, N. Kirchgessner, C. A. Le Marie, L. M. York, J. P. Lynch, and A. Hund, "Next generation shovelomics: set up a tent and REST," *Plant and Soil*, vol. 388, no. 1-2, pp. 1-20, 2015.

[91] M. A. Fischler, and R. C. Bolles, "Random Sample Consensus - a Paradigm for Model-Fitting with Applications to Image-Analysis and Automated Cartography," *Communications of the Acm*, vol. 24, no. 6, pp. 381-395, 1981.

[92] G. Lobet, M. P. Pound, J. Diener, C. Pradal, X. Draye, C. Godin, M. Javaux, D. Leitner, F. Meunier, and P. Nacry, "Root System Markup Language: toward a unified root architecture description language," *Plant Physiology*, pp. pp. 114.253625, 2015.

[93] M. P. Pound, A. P. French, J. Atkinson, D. M. Wells, M. J. Bennett, and T. P. Pridmore, "RootNav: Navigating images of complex root architectures," *Plant Physiology*, vol. 162, no. 4, pp. 1802-1814, 2013.

[94] Y. L. Chen, V. M. Dunbabin, J. A. Postma, A. J. Diggle, J. A. Palta, J. P. Lynch, K. H. M. Siddique, and Z. Rengel, "Phenotypic variability and modelling of root structure of wild Lupinus angustifolius genotypes," *Plant and Soil*, vol. 348, no. 1-2, pp. 345-364, 2011.

[95] "Python. https://www.python.org/ [Accessed 16 July 2015]."

[96] "STAMPEDE at TACC. https://www.tacc.utexas.edu/systems/stampede [Accessed 16 June 2015]."

[97] "NumPy. http://sourceforge.net/projects/numpy/ [Accessed 16 July 2015]."

[98] "SciPy. http://www.scipy.org/ [Accessed 16 July 2015]."

[99] "Mahotas. http://luispedro.org/software/mahotas/ [Accessed 16 July 2015]."

[100] "PIL. http://www.pythonware.com/products/pil/ [Accessed 16 July 2015]."

[101] “Leptonica. http://www.leptonica.org/. [Accessed 16 July 2015].”

[102] “Tesseract-ocr. https://github.com/tesseract-ocr [Accessed 16 July 2015].”

[103] “ZBar. http://zbar.sourceforge.net/ [Accessed 16 July 2015].”

[104] “Graph-tool. http://graph-tool.skewed.de/ [Accessed 16 July 2015].”

[105] “Drush. http://www.drush.org/en/master/ [Accessed 16 July 2015].”

[106] C. R. Darwin, *The power of movement in plants,* London: John Murray, 1880.

[107] R. D. Firn, and J. Digby, “Solving the puzzle of gravitropism - Has a lost piece been found?,” *Planta,* vol. 203, pp. S159-S163, 1997.

[108] L. O. Hall, A. M. Bensaid, L. P. Clarke, R. P. Velthuizen, M. S. Silbiger, and J. C. Bezdek, “A Comparison of Neural Network and Fuzzy Clustering-Techniques in Segmenting Magnetic-Resonance Images of the Brain,” *Ieee Transactions on Neural Networks,* vol. 3, no. 5, pp. 672-682, 1992.

[109] D. Q. Zhang, and S. C. Chen, “A novel kernelized fuzzy C-means algorithm with application in medical image segmentation,” *Artificial Intelligence in Medicine,* vol. 32, no. 1, pp. 37-50, 2004.

[110] A. Buades, B. Coll, and J. M. Morel, “A review of image denoising algorithms, with a new one,” *Multiscale Modeling & Simulation,* vol. 4, no. 2, pp. 490-530, 2005.

[111] P. Etyngier, F. Segonne, and R. Keriven, “Active-contour-based image segmentation using machine learning techniques,” *Medical Image Computing and Computer-Assisted Intervention - Miccai 2007, Pt 1, Proceedings,* vol. 4791, pp. 891-899, 2007.

[112] Y. Zheng, S. V. Gu, H. Edelsbrunner, C. Tomasi, and P. Benfey, “Detailed Reconstruction of 3D Plant Root Shape,” *2011 Ieee International Conference on Computer Vision (Iccv),* pp. 2026-2033, 2011.