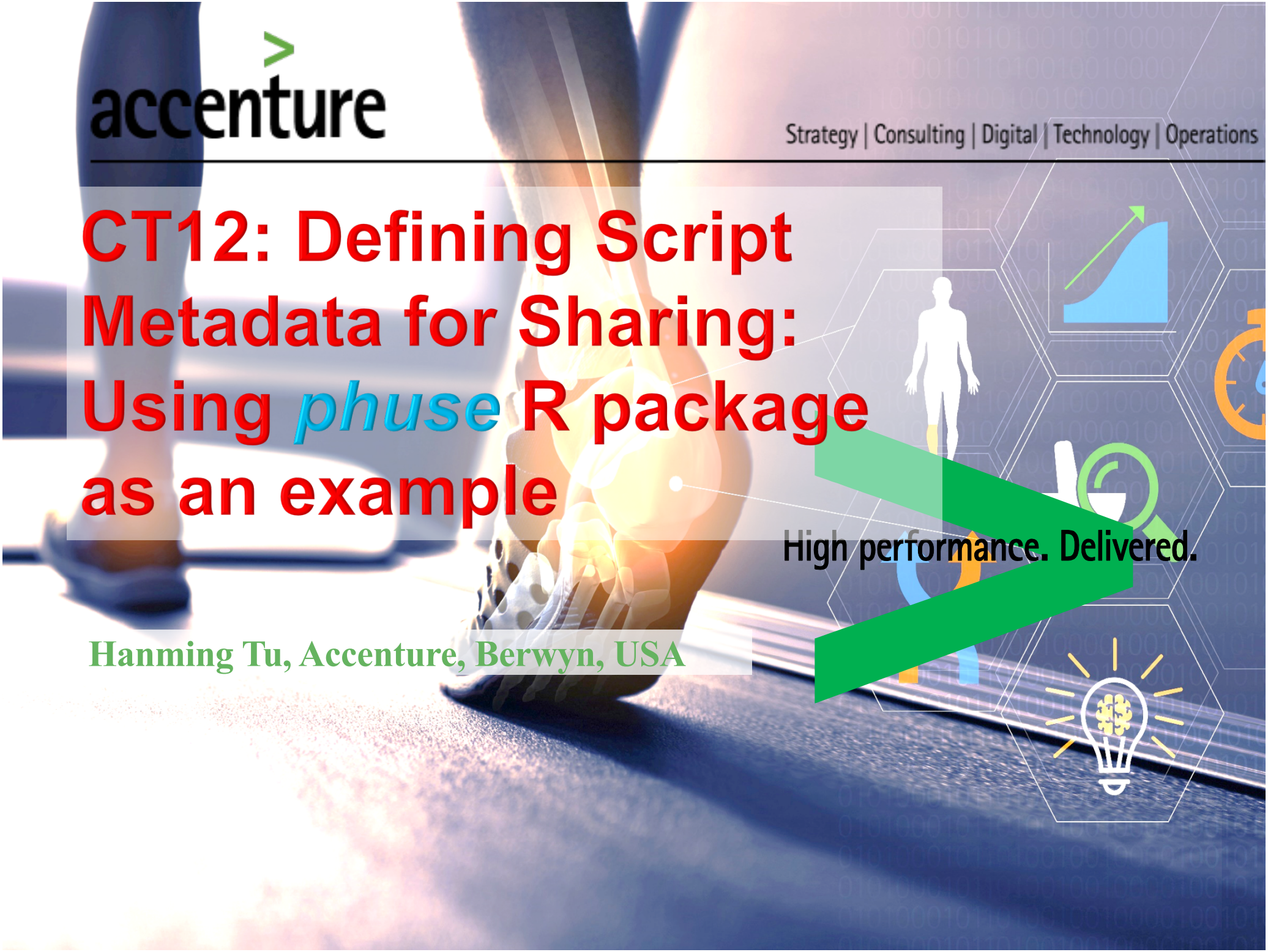
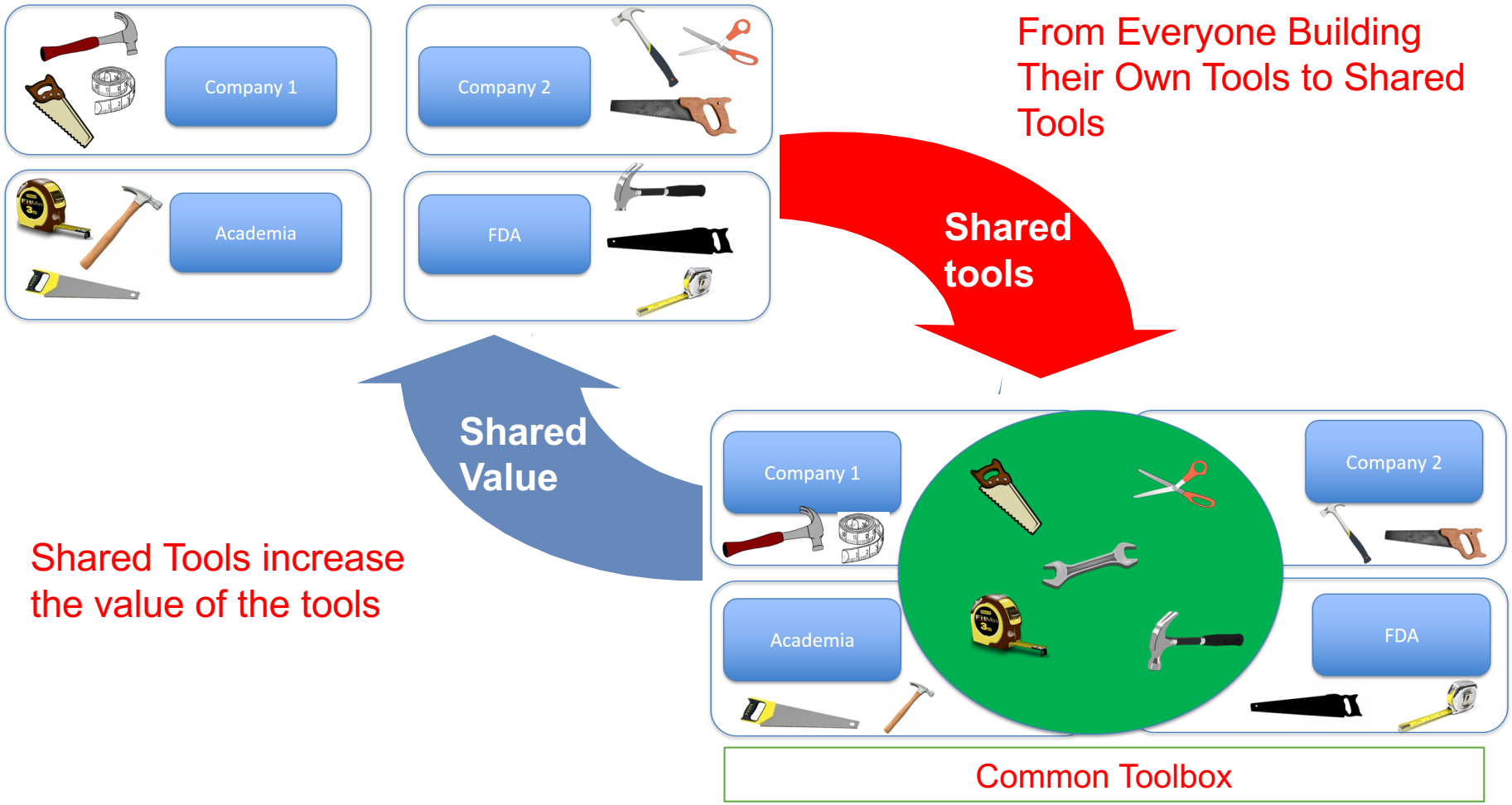

accenture
Strategy | Consulting | Digital | Technology | Operations
CT12: Defining Script Metadata for Sharing: Using phuse R package as an example
High performance. Delivered.
Hanming Tu, Accenture, Berwyn, USA

Agenda

- Background Info and Issue Statement
- Metadata and Script Metadata
- YML and Script Metadata Format
- R, R Package and RStudio Project
- R *shiny* and *phuse* Package
- Conclusion

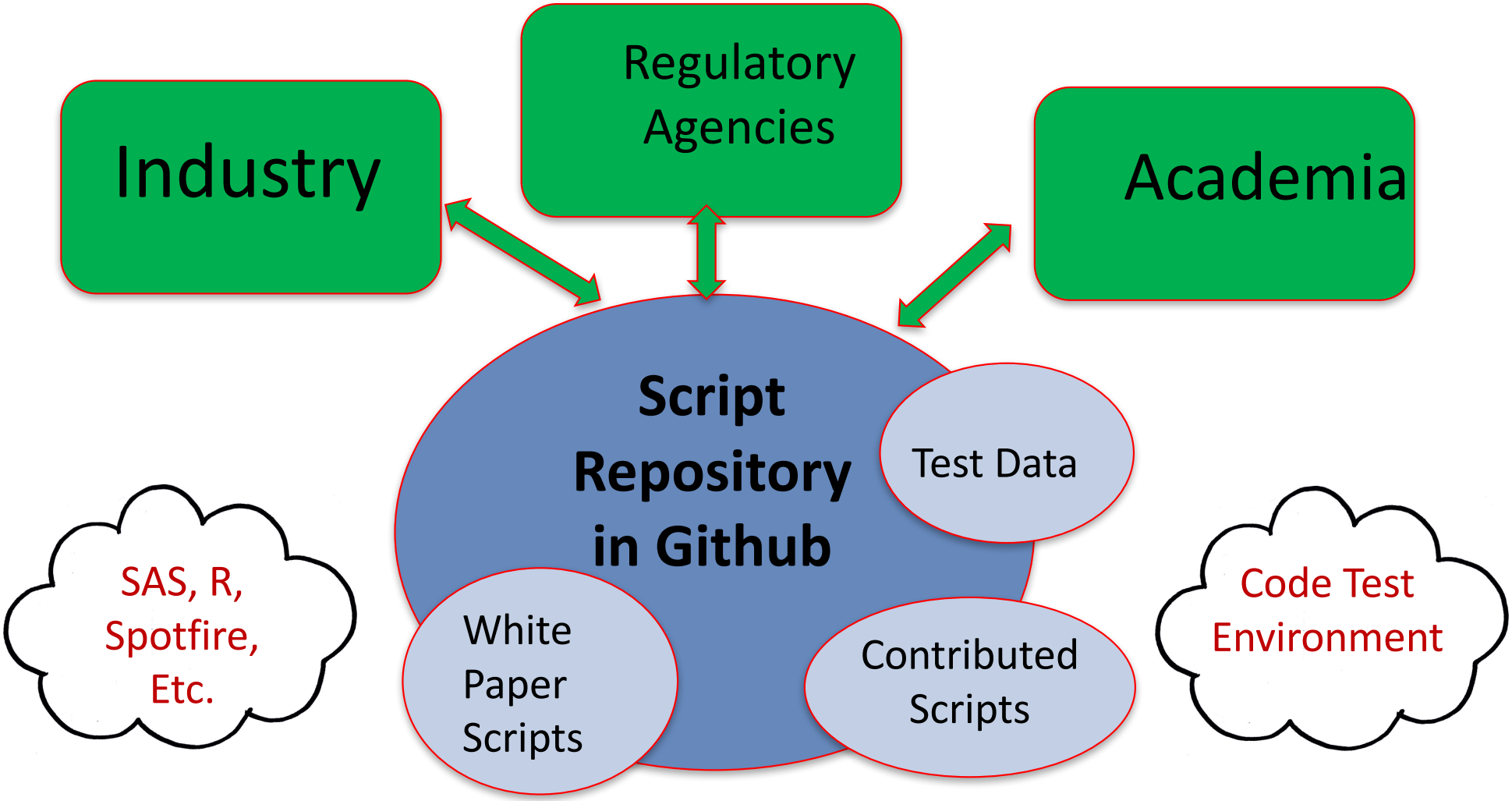
Background: Working Group Vision

Standard Analyses and Code Sharing Working Group



Background: Script Repository

Use github to host the shared reusable code library



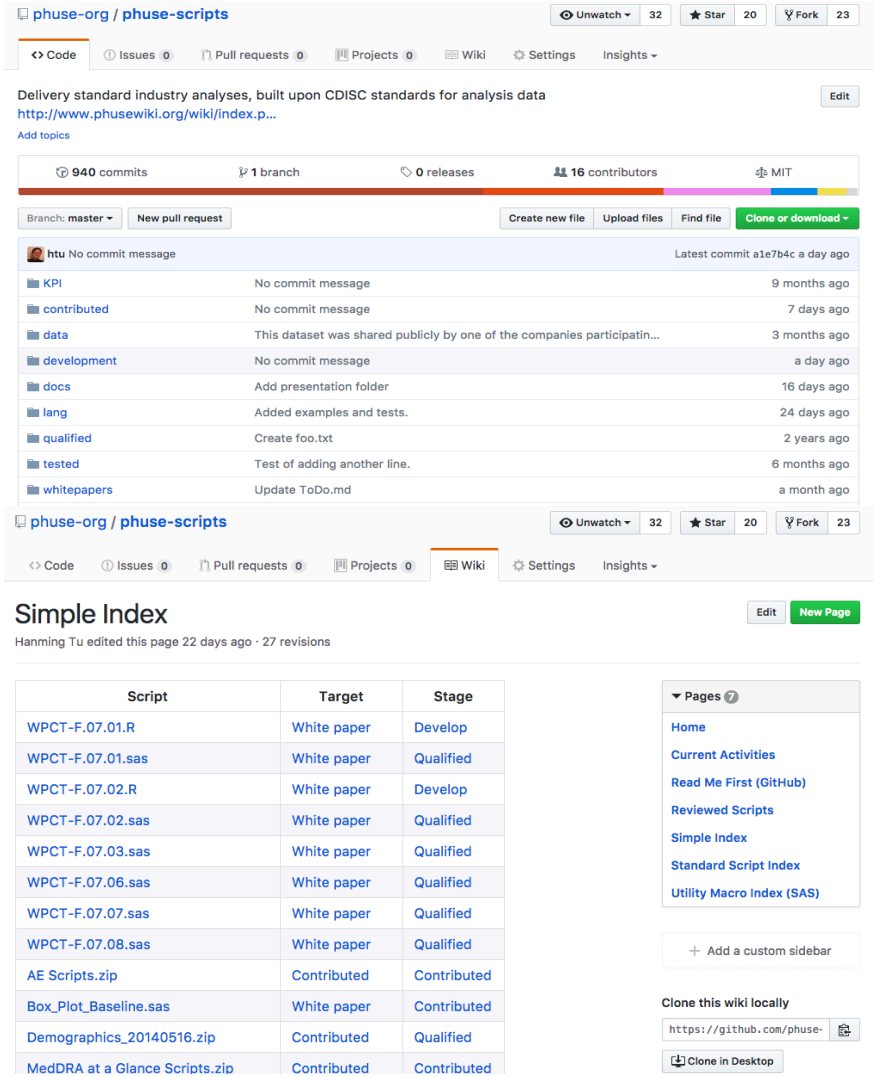
Accomplishments: Scripts

- Scripts developed by volunteers
  - 6 Scriptathons (plus additional work by project members) resulting in several scripts at various stages
  - Scripts developed based on white paper
- Scripts contributed by other groups
  - FDA: https://github.com/phuse-org/phuse-scripts/wiki/Reviewed-Scripts
  - Non-clinical: https://github.com/phuse-org/phuse-scripts/tree/master/contributed/Nonclinical
  - Data Handle: https://github.com/phuse-org/phuse-scripts/tree/master/lang/SAS/datahandle
  - Spotfire Templates: https://github.com/phuse-org/phuse-scripts/tree/master/contributed/Spotfire

Issue Statement

How to manage the development and usage of the scripts

- Find the scripts
  - Index pages are not updated promptly.
- Navigate in the repository
  - It is complicated and deep
- Use the scripts
  - Need to be downloaded
  - Need to be updated



Script Metadata

Script and its environment

- **Keywords**: a list of words used to categorize the scrip such as analysis, boxplot, etc.
- **Script**: this metadata group defines the name, version, short and long description of the script.
- **Language**: this metadata group provides the information about the script language such as SAS 9.4.0, R 3.4.0, etc.
- **Environment**: provides the computing environment of the script language and the special language configuration.

Script Metadata

It is really about the script!

- **Inputs**: defines the input datasets and parameters required for the successful executing the scripts.
- **Outputs**: provides the expected output datasets and variables.
- **Repo**: provide the hosting repository information.
- **Authors**: documents the developers who create or contribute to the development and qualification of the script.
- **Qualification**: documents the qualification state and process.
- **Stages**: provide the historical states of the scripts.
- **Ratings**: records the users who review and give the rating about the script.

YML- Metadata Format

YML is chosen format!

- YML is a short name for YAML
- Yet Another Markup Language
- YAML Ain't Markup Language
- YML is a data serialization language that can be read by both human and machine

```
Keywords: Test, Metadata, Dual Box
Script:
  name    : metadata_example_rep.yml
  title   : Metadata example on local drive
  desc    : > This script demonstrates how to use YML to store the
          metadata about your program and define your input parameters
          and their values.
  version: 0.1.1
Language:
  name    : YML
  version: x.x.x
Environment:
  system: Linux or Window 2010
  os_version: OEL 5.8, Window 2010
  desc: Description of the computing environment.
Inputs:
  datasets: dm.xpt,ae.xpt,testfile.xlsx
  p1: String - dataset name
  p2: Number - depart id
Outputs:
  datasets: out1, out2, out3
  v1: Date - script execution date and time
  v2: User - user who executes the script
Repo:
  base_dir: https://github.com/phuse-org/phuse-scripts/raw/master
  …
  lib_files: Func_comm.R
Authors:
  - name    : Jon Doo
    email   : jon.doo@phuse.com
    company: PhUSE
Qualification:
  last_date: DD-MON-YYYY
  last_by: FirstName LastName
  stage: T
  doc_url: a link to latest documentation
  note: C - Contributed; D - Development; T - Testing; Q - Qualified
Stages:
  - date: 01-JAN-2016
    name: Jon1 Doo
    stage: C
    docs: a link to qualification documents
Ratings:
  - user: htu
    date: 25-AUG-2017
    asso: Accenture
    stars: 5
# end of file
```

R, R Package, RStudio Project

R is chosen language for developing phuse project

- **R** is an open source programming language and software environment for statistical computing and graphics
- **R Package** is the fundamental unit of shareable code bundled with data, tests, examples, and documentation.
- **RStudio** is a free and open-source integrated development environment (IDE) for R.
- **Rstudio project** helps you organizing your development and build of a R code or a Package.

R shiny and phuse package

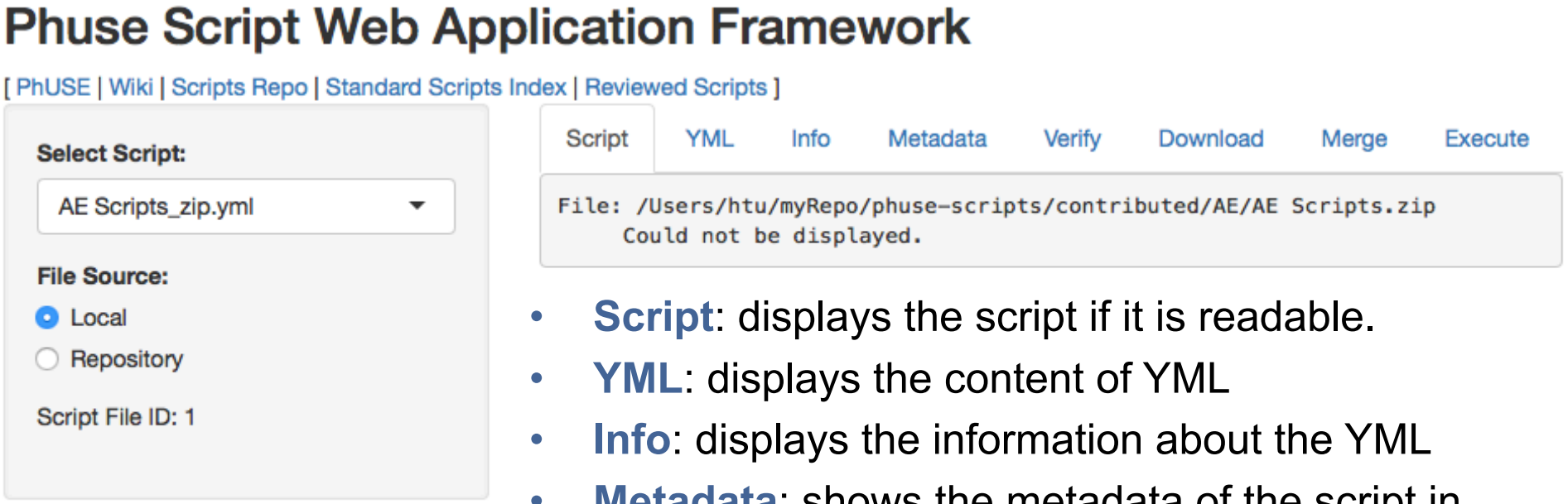
Use shiny as interface to develop phuse web application framework

- Shiny is an R package that makes it easy to build interactive web apps straight from R.
- Use R shiny develop the *phuse* package to help finding, downloading and executing scripts.

How to get R *phuse* Package	How to run R *phuse* Package

`install.packages("devtools")`
`library(devtools)`
`install_github("TuCai/phuse")`
 	
Or directly install once the package accepted by CRAN
 	
`Install.packages("phuse")`
	`library(phuse)`
	`Start_phuse()`
	
	• Clone the phuse-scripts repository to your local computer if you are the first time to start the interface or the local repository is old
	• Grep all the YML files from the local repository
	• Build a data frame to hold the information for all YML files
	• Write the data frame to a local file
	• Populate the "Select Script" dropdown list

The phuse R Package



- **Script**: displays the script if it is readable.
- **YML**: displays the content of YML
- **Info**: displays the information about the YML
- **Metadata**: shows the metadata of the script in table format

```
# phuse functions:
build_script_df
read_yml
extract_fns
download_fns
cvt_list2df
merge_lists
run_examples
```

- **Verify**: verifies the existence of the files defined in YML
- **Download**: downloads the script to local computer
- **Merge**: merges online and local metadata files
- **Execute**: executes the script if it is executable.

Tasks performed by phuse interface

Improve user experience

- Clone the repository dynamically
- Build script index/dropdown list dynamically
- Display script and metadata (YML)
- Verify data and lib files associated with the script
- Download the script and associated files
- Merge the online and local metadata if local one exists
- Execute R scripts

Tasks to be developed

Improve collaboration among developers, reviewer and users

- Use a predefined template to build script metadata file
- Search for scripts and update the metadata files
- Add script metadata to the newly contributed and developed scripts.
- Facilitate script review and qualification process
- Expand the functionality to other type of scripts such as SAS, Java, PL/SQL, etc.

Conclusion

- Script metadata provides the information about the script's purpose, version, execution environment, library and data files used, inputs, outputs, review history, ratings, etc.
- The metadata make it easy to share, access and execute scripts in the repository.
- The phuse R package provides a web application framework for further building a platform for sharing and accessing the scripts in the repository

Conclusion

- R, RStudio and R shiny are the important tools for the statistical computing environment.
- Building R packages with metadata is the first step to make script repository into CRAN alike (Comprehensive R Archive Network) for the shared scripts.
- PhUSE started renting some servers from Amazon to explore new technologies and analytical tools for collaboration and sharing.

Phuse Web Application Demo

Q & A

Questions and Answers

Contact Information

Hanming Tu	
P: 610-407-1817; C: 484-881-2384 E: hanming.h.tu@accenture.com	
Address: 1160 West Swedesford Road, Berwyn, PA 19312, USA Web: www.accenture.com Fax: 610-535-6615	