

Microblogs y micropolítica

Retratos de datos y data selfies en Twitter
Bitácora por días

Promotor: Offray Luna

Colaboradores:

- Eduardo Riesco.
- Juan Martin
- Ferran Martinez.
- Luciano

Estas son las memorias del proyecto realizado, durante casi dos semanas, en el contexto de Visualizar 2018. Tienen que ver con cómo los datos que producimos cotidianamente en las redes de microblogging, como Twitter, pueden empoderar nuevas prácticas ciudadanas y de aprendizaje en comunidades de base, para vernos como personas y también para ver a nuestros funcionarios e instituciones públicas en dichas redes.

Contents

1 Microblogs y micropolítica: retratos de datos y data selfies en Twitter

- 1.1 Descripción
- 1.2 Puntos Clave

2 CodiMD: Trabajo síncrono

- 2.1 Registro
- 2.2 Crear pads
- 2.3 Descargar pads
 - 2.3.1 Desde el navegador
 - 2.3.2 Con wget.
 - 2.3.3 Con Pharo

3 Markdown

- 3.1 Introducción
- 3.2 Enlaces:
- 3.3 Ejercicio:

4 Exportar datos desde Twitter

5 Publicar datos de Twitter

- 5.1 Antes de empezar
- 5.2 Subir datos a Zenodo
- 5.3 Listo para solicitar tu Data Selfie

6 Preguntas sobre los Data Selfies

7 Procesarlos y visualizarlos

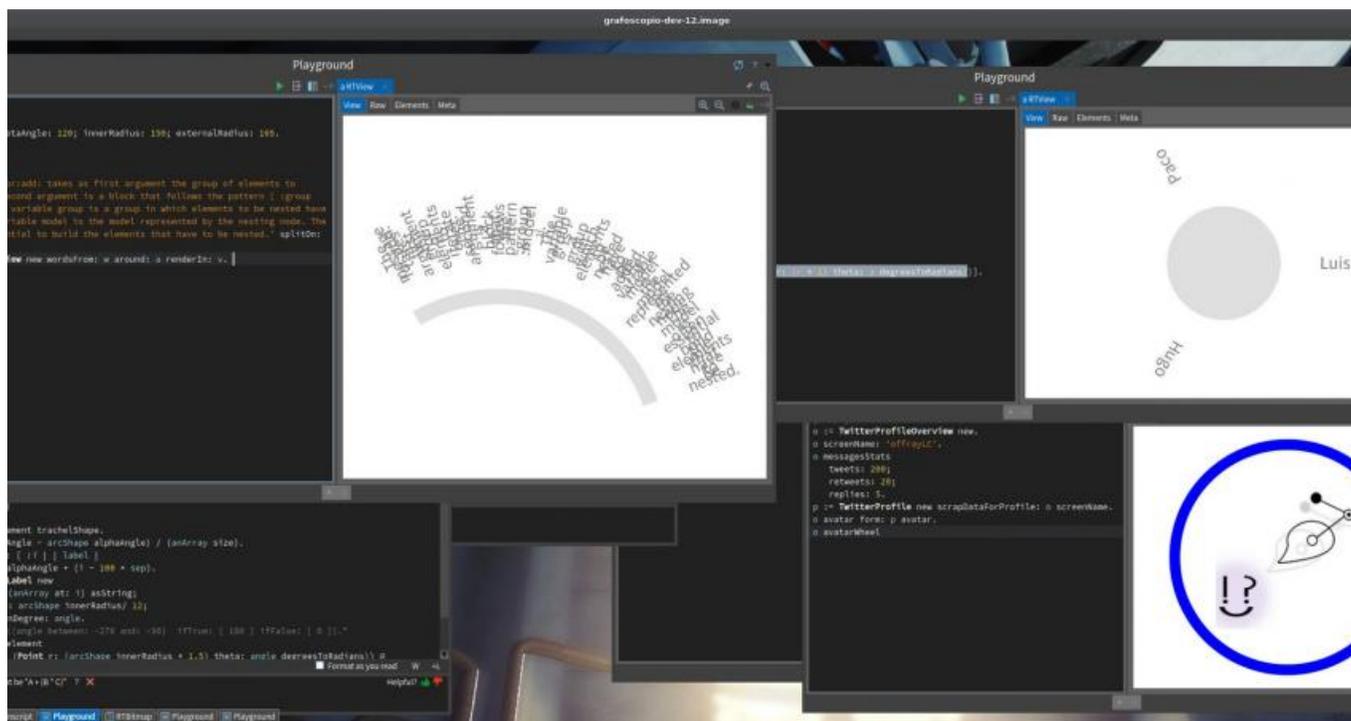
- 7.1 Live Coding
- 7.2 Instalar el software
 - 7.2.1 Pharo
 - 7.2.2 Grafoscopio
- 7.3 Actualizar el software
 - 7.3.1 Monticello
- 7.4 Mob programming para visualización
- 7.5 Libretas interactivas en Grafoscopio

8 Material del Stand para la Exposición

9 Documentación de los prototipos UX

- 9.1 Metodológico
- 9.2 Tecnológico
 - 9.2.1 Diagrama de Flujo
 - 9.2.2 Paralelo de visualizaciones entre las ofrecidas por Twitter y las nuestras
- 9.3 Conceptual

1 Microblogs y micropolítica: retratos de datos y data selfies en Twitter



1.1 Descripción

Una vez liberados de la tiranía del “timeline” de Twitter, ¿cómo nos veríamos a nosotros mismos?, ¿Cómo una visualización creada a medida, puede ayudarnos a percibir nuestras cámaras de eco y aquellas de nuestras instituciones públicas y políticos electos? ¿Pueden los datos pequeños, de la información que producimos en los microblogs agenciar formas de micropolítica en el mundo normalizado, panóptico y enajenante del “Big Data”?, ¿Puede el código convertirse en un lenguaje común, más allá de los programadores y los “tekkies”, para expresar formas de alfabetismo crítico en estas formas de micropolítica? En este proyecto, los participantes estarán en capacidades de exportar, *escraper*, codificar y visualizar, de maneras originales, los datos personales que ellos producen de manera cotidiana en su timeline público de Twitter y, si queda tiempo, hacer algo similar con las instituciones públicas y políticos electos, puenteando la micropolítica y la política, el código, los datos, las visualizaciones y el alfabetismo digital, de modo que se puedan esbozar respuestas a estas preguntas y generar otras, con respuestas tentativas.

Los participantes aprenderán cómo reapropiar su información pública desde redes privadas de microblogging (Twitter) hacia los repositorios de datos como bienes comunes (Zenodo), a consultar, agrupar y visualizar información, expresando en código las diferentes partes del procesos, constituyendo una forma alfabetismo digital crítico (literacy, numeracy y graphicacy,

Data Pop Alliance) y de ciudadanía digital (Schrock), donde el código se convierte en un lenguaje común y se usurpa del dominio exclusivo de los especialistas, de manera que se consoliden acciones críticas desde y sobre el código, tanto en experiencias en primera persona como en colectivo (Mob Programming [1][2]). Los selfies y retratos de datos obtenidos con este procedimiento, serán publicados de vuelta en las redes sociales de los que provienen, revisando el impacto de esta experiencia en la constitución de identidad y discurso público.

El proyecto usará y extenderá “infraestructura digital de bolsillo” Grafoscopio, (sencilla, auto-contenida y que funcionan bien en y fuera de línea) y se basa en dinámicas realizadas y probadas durante una (anti)hackatón-taller recurrente de una semana, llamado el Data Week, que completa más de 350 horas en 12 ediciones desde 2015. Grafoscopio y el Data Week ha producido prototipos funcionales como el de visualizaciones de dominio específico para medicamentos en 16 países, los Panamá Papers como investigación reproducible, la apertura del Manual de Periodismo de Datos y el que se presenta y extiende acá, los Data Selfies de Twitter. Este enfoque que conjuga herramientas amoldables ágiles, con prototipos y alfabetismos digitales críticos desde comunidades de base, así como investigación académica (Doctorado en Diseño y Creación, Universidad de Caldas, Colombia y Proyecto Ciudad de Datos, Universidad Javeriana, Colombia) y ha sido valorado y seleccionado en una decena de eventos internacionales, entre los que están las Smalltalks (Argentina, 2015), Re:publica (Alemania, 2017 y 2018) Internet Freedom Festival (España 2017) y Residencias y Campamentos de Datos (Montenegro 2017 y 2018).

En esta experiencia, el código, la documentación y la visualización de datos, se entremezclan de manera ágil y permanente para co-producir alfabetismos críticos, fortalecer comunidades de práctica y formas de participación, activismo y ciudadanía digital. Los prototipos se mejoran iterativamente de manera ágil, y se enfrentan a problemas reales, dentro de la tradición del Live Coding y el Software como Artesanía (Aaron). Es un proyecto práctico que conversa con perspectivas académicas, históricas, ciudadanas y activistas y que espera ayudar a entretejerlas de maneras orgánicas, pero más permanentes y potentes.

1.2 Puntos Clave

- Recrear y cocrear ciudadanías desde nuevas prácticas mediadas por tecnologías digitales.
 - Ciudadanías normativas (en la Ley) → Ciudadanías enactivas (en la acción).
 - Fuerzas legales + performativas + imaginarias.
 - (anti)hackatón+taller.
 - El código como lenguaje común (usurparlo de los especialistas y “tekkies”).
 - Cadena de montaje vs Ensanchar las fronteras de los saberes. (anti)hackatón+taller.
 - Diversidad de personas y saberes: bibliotecarios, diseñadores, historiadores, profesores, estudiantes, activistas, novatos, programadores, etc.
 - Datos críticos.

- Documentación ágil y permanente (este es un ejemplo).
- De los data silos a los data commons.
- Live Coding:
 - Cambiar en caliente el código para hacer *prototipos progresivamente más funcionales* [3].
 - Lidar con lo contingente y lo no planificado.
 - El error y lo incompleto como oportunidades.
 - La improvisación como cartografía de lo posible.
- Lo polisémico: muchos significados encontrados en un mismo espacio y artefacto.
- Intercambios Norte-Sur (espacios geopolíticos de creación y resistencia, con preocupaciones compartidas y metodologías diferentes.)

2 CodiMD: Trabajo síncrono

En ocasiones queremos escribir a muchas manos. Puede ser que estemos en un taller y queramos tomar notas de manera colectiva. O bien estamos trabajando en un documento y quisiéramos reunirnos con coautores para arrojar ideas, escribir subsecciones o realizar correcciones. Para ello se han creado varias plataformas que con sólo un enlace web permiten iniciar la colaboración con todos los que tengan dicho enlace (suponiendo que tienen los permisos para ello). Debido a que cuando los coautores se encuentran a través de la web, pueden escribir al mismo tiempo y ver los cambios de otros, se dice que dichas plataformas trabajan en “tiempo real”.

Por otro lado, a veces queremos facilitar los procesos de escritura desde múltiples dispositivos y pantallas (incluso las pequeñas de dispositivos móviles) y en este caso un lenguaje de etiquetamiento ligero es una mejor opción que un editor pesado como Google Docs o Office 360, pues permite concentrarse en el contenido y explicitar ágilmente la estructura con unas marcas sencillas. Puedes ver más sobre dichas motivaciones en nuestro texto introductorio sobre [documentación ágil](#).

CodiMD es una de tales plataformas para escritura colectiva en tiempo real, usando lenguajes de etiquetamiento ligeros. Emplea un supraconjunto de [Markdown](#), llamado [CommonMark](#), que permite varias características necesarias en la edición de textos profesionales, como tablas, avison (advertencias, consejos), imágenes, videos, notas a pie de página, entre otros. El documento presente, por ejemplo fue escrito colectivamente empleando CodiMD.

Una de sus principales ventajas es que permite ver en simultáneo el código de lo que se escribe, como su previsualización, empleando para ello un sistema de paneles dobles simultáneos, así las personas pueden enfocarse sólo en la escritura, sólo en la lectura o en ambas.



Agregar imagen con paneles duales de este mismo documento, pero empleando una pantalla más grande que la de un portátil.

2.1 Registro

El registro se puede hacer de dos formas, usando herramientas externas como GitHub, Google o Twitter o a través del registro por correo electrónico.



Revisar definición de las capturas

Luego de elegir una de estas opciones y validar el registro, es posible editar los documentos.

A diferencia de los Pads que se han venido utilizando en los Data Rodas anteriores, en donde los documentos quedan públicos y abiertos a la edición de cualquier persona, en esta oportunidad se ponen en debate la limitación en la edición de contenidos por terceras personas ajenas al proyecto, con el fin de garantizar la privacidad y seguridad de la documentación producida.

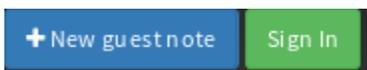


Regístrate en un proveedor de CodiMD como Docutopia o Heroku.

2.2 Crear pads

Esencialmente hay dos formas:

- Desde la portada de la instancia de CodiMD. A través del botón azul en la esquina superior derecha de la pantalla, como se vé a continuación



- A través de un enlace al nuevo documento.

Durante el proceso de documentación colaborativa podemos incluir enlaces en los documentos de manera fluida, para los efectos de esta comunidad se van a relacionar contenidos siguiendo la estructura de enlaces de Markdown, con la siguiente convención:

- se incluirá entre corchetes cuadrados el texto que va a contener el enlace, por ejemplo [texto].
 - Luego se incluirá el complemento de la URL del recurso, determinando la ruta de origen del nuevo documento entre paréntesis "()".
 - El contenido entre paréntesis incluirá la ruta a partir del "./" para referirse al servidor que estamos usando, seguido de dos puntos (:) para diferenciar las “carpetas” o secciones del tema relacionadas con el contenido.
 - Finalmente, el resultado será algo como `texto` o para referenciarlo al ejercicio de esta sesión sería algo como
- `[CodiMD](./inkid:r3p:codimd)`

En donde el texto del enlace “CodiMD” refiere al lugar usando en el servidor de Docutopia, relacionado con los contenidos del *investigation kit* (inkit), subtema 1: *R3P*, subtema 1.2: *CodiMD*.



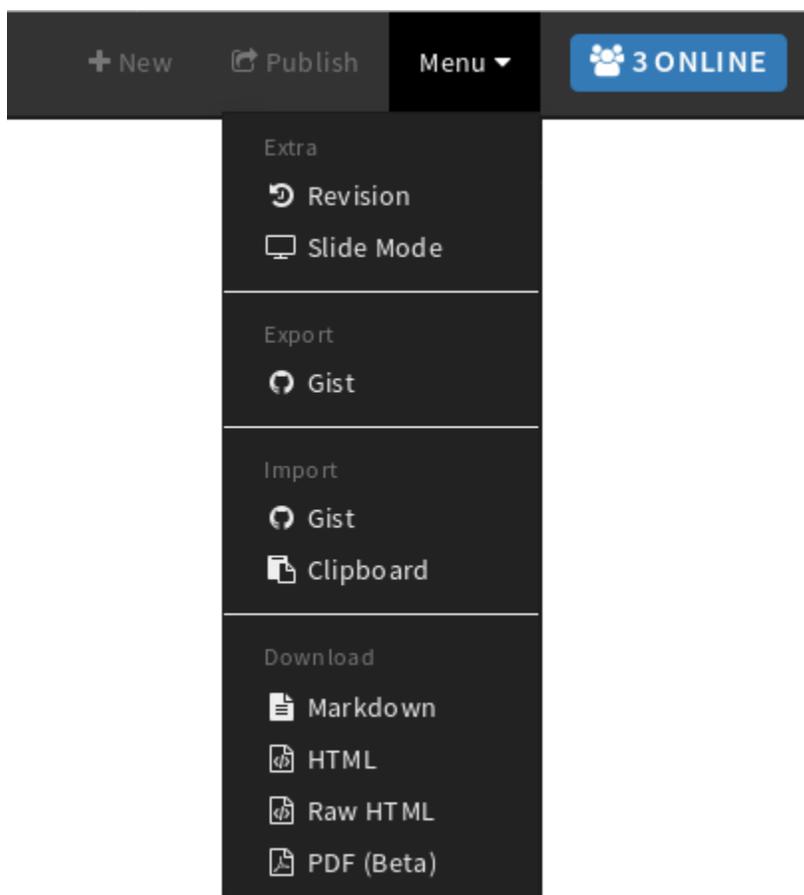
Crea un pad en el proveedor en el que te registraste en la actividad previa.

2.3 Descargar pads

Una vez hemos creado nuestros documentos en línea, puede ser necesario descargarlos para tener una copia de respaldo de los mismos o trabajarlos fuera de línea. Acá veremos tres métodos para ello.

2.3.1 Desde el navegador

A través de la vista de escritura, podemos acceder al menú desde el botón en el margen superior derecho, donde se muestran las diferentes opciones de descarga del documento: Markdown, HTML, HTML Raw y PDF.



Descarga un pad desde el navegador.

2.3.2 Con wget.



Suponemos que para esta parte tienes instalada la terminal o consola, o línea de comandos, que sabes cómo abrirla y los elementos básicos de escribir y ejecutar

comandos en ella. También que tienes instalados los programas que se usan habitualmente en ella, como `wget`.

Usaremos los términos terminal, consola o línea de comandos de manera intercambiable.

Otra de las maneras de descargar un documento es emplear la terminal. La ventaja de este procedimiento es que podemos agilizar muchas descargas con sólo repetir un comando de una línea, lo cual abre a su vez la posibilidad de automatizarlo.

Para descargar este documento, abre la terminal y escribe:

```
wget -O codimd.md https://docutopia.tupale.co/inkit:r3p:codimd/download
```

Con esto hemos creado un archivo llamado `codi.md`, en el disco duro, con los contenidos guardados en la dirección del pad cuyo enlace es `inkit:r3p:codimd` y está hospedado dentro del servidor `https://docutopia.tupale.co`. Si queremos descargar otros pads con otros enlaces, creados en otro servidor, o guardarlos con otros nombres de archivo, deberemos cambiar el comando previo respectivamente.



Descarga un pad utilizando `wget`.

2.3.3 Con Pharo

Pharo es la plataforma sobre la cual está hecho [Grafoscopio](#), que a su vez sirve para integrar y automatizar mucho del trabajo sobre investigación y publicación reproducible y remezclable, así que veremos cómo emplearla en las actividades de este texto de maneras progresivamente más elaboradas.

(pista: elegant code).

3 Markdown

- **Propósito:** Aprender sobre Markdown y ubicarlo entre otros formatos ligeros para documentar y compartir datos. - **Ejercicio práctico:** Crear un Etherpad en Markdown y compartirlo con los demás talleristas. - **Prerequisitos:**

- **Lección:** CodiMD.

3.1 Introducción

Markdown es un lenguaje de etiquetamiento ligero y es, en varias métricas, el más popular. Vamos a deconstruir las dos partes de esta explicación.

Un lenguaje de etiquetamiento ligero, es una forma de representar la estructura y presentación de un texto acompañándolo por un conjunto de marcas sencillas en su contenido, por ejemplo, colocar un guión bajo alrededor de una palabra quiere decir que esta en cursivas (*así*) y rodearla por dos asteriscos, quiere decir que está en negrillas (**de este modo**). Lo anterior permite escribir documentos de manera fluida.

Es popular porque es usado en sitios de alto tráfico donde se crean permanentemente contenidos y se ha migrado desde ahí a poblaciones más específicas. Por ejemplo:

- StackOverflow.
- GitHub.
- Pandoc: Nuestra variante de Markdown.
- Scholarly Markdown.
- Critic Markdown.
- Hypothesis

Hay esfuerzos de hacer publicación e investigación transparente y reproducible usando Markdown. Gracias a ellos, se pueden producir textos extremadamente complejos, sólo usando una variante de Markdown. Por ejemplo en la comunidad de Grafoscopio hemos producido el [Manual de Periodismo de Datos](#) y los [Pasos para una Biblioteca Digital de Bogotá](#). Y además hay canales y comunidades referidas al uso de alguna variante de Markdown para hacer dichos investigación reproducible como el canal de [Nicholas Cifuentes-Goodbody](#).

En contraste, existen varios lenguajes de etiquetamiento “pesado” usados ampliamente, pero engorrosos para ejercicios de memoria y escritura fluida, como:

- HTML.
- SGML.

- XML.
- LaTeX.

Y hay lenguajes llamados de serialización de datos, que sirven para describir la estructura de datos de un modelo y compartirla entre diferentes sistemas de información, que también empiezan a tener tendencia hacia lo ligero, entre ellos:

- YAML.
- JSON (Este lenguaje es el que se usará al exportar los datos de Twitter para el data selfie).
- TOML.
- RAML).
- STON (Inspirado en JSON, pero para Smalltalk).

Tanto los lenguajes ligeros de etiquetamiento y de serialización de datos como los “pesados” pueden ser combinados.

“La notación inadecuada lo hace pensar a uno inadecuadamente”

3.2 Enlaces:

Lenguajes de etiquetamiento ligero:

- En Wikipedia (inglés).
- En Wikipedia (español).
- Markdown Tutorial.
- Learn Markdown in 60 seconds
- Editor de Markdown interactivo en línea.

3.3 Ejercicio:

- Visitar los enlaces de la sección anterior, leerlos y crea un etherpad, usando Markdown,
- Leer el documento introductorio a CodiMD y seguir los ejercicios hasta la descarga del documento creado desde el navegador.
- Crear un pad en Docutopia (que es una instancia de CodiMD), dentro del arenero. !!!
WARNING Para organizar la información, los pads de práctica que creemos van a estar dentro de la arenera, que tendrá esta estructura:

```
subdominio.dominio.algo/categoria:pagina  
docutopia.tupale.co/arenera:<pagina>
```

Reemplazando <pagina> por la url que queramos, por ejemplo prueba1.

!!! note Recuerda que esta instancia soporta CommonMark, un supraconjunto de Markdown con características extendidas.

- Si estás en una Data Week o Data Roda, comparte el enlace de tu pad con los demás participantes, usando el etherpad de la sesión del día.

[^ [Indice](#) | [Fossil](#) >]

4 Exportar datos desde Twitter

Exportar los datos de Twitter consiste en lo siguiente:

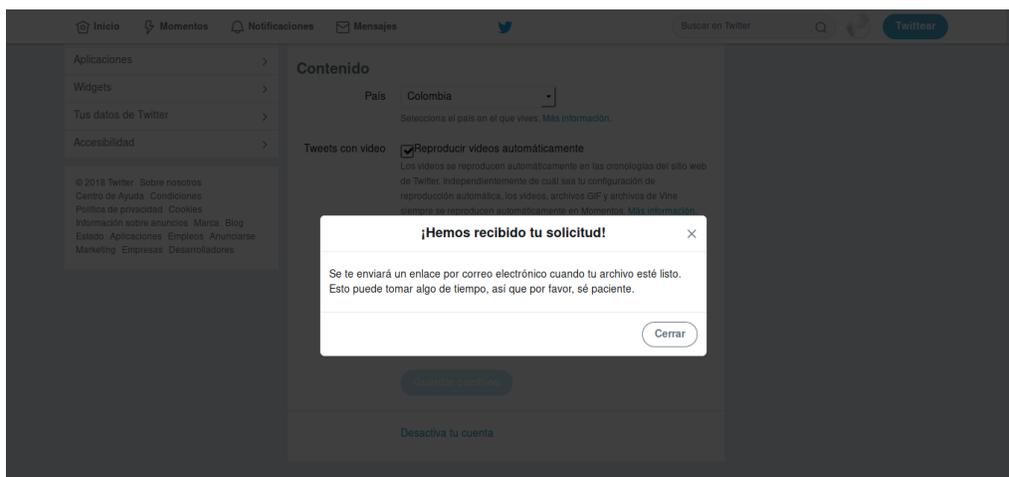
Entramos a nuestro perfil de Twitter y hacemos clic en el avatar. Desde el menú desplegable, elegimos la opción Configuración y privacidad.

Allí vamos a la parte inferior de la página y elegimos la opción Solicita tu archivo



^Arriba | Notificación de exportación en curso.

Aparecerá esta notificación diciendo que el archivo va en camino:



^Arriba | Notificación de exportación en curso.

IMPORTANTE

Esta exportación sólo sacará los datos públicos de Twitter (tweets, favs, replies, retweets), pero ningún mensaje directo o conversación no pública será exportada.

Entramos a nuestro correo, en un tiempo y buscamos un correo de Twitter con un enlace de descarga a nuestros datos.

Si tienes cuenta de Gmail y no ves el correo en tu bandeja de entrada principal, revisa tu bandeja "Social".

En nuestro correo debe llegar un mensaje que dice algo como:

> <nombre de="" usuario=""> tu archivo de Tweets está listo **Descarga ahora**

Si hacemos clic en el botón "Descarga ahora" veremos una pantalla similar a esta:



^Arriba | Pantalla de notificación para descarga de archivo de Tweets.

El archivo que descargamos debe lucir algo como:

xxxYYYzzz_7f4206d8bae26af45ba2905899db4d77eab3e2af.zip

donde xxxYYYzzz corresponde a un identificador único que asigna Twitter a cada uno de sus usuarios y la parte después del guión bajo (_) corresponde a un índice específico asociado al archivo que acabamos de exportar. Si exportamos antes o después nuestro archivo de Twitter, este número final cambiará.

Hay dos caminos:

- Si la exportación fue exitosa, procedemos con la publicación de datos.
- Si no lo fue, tenemos tres caminos: esperar, quejarnos o hacer scraping de datos.

5 Publicar datos de Twitter

Importante: Para realizar esta parte, suponemos que previamente has exportado tu archivo de Twitter.

El resultado clave es que puedas publicar los datos que tu mismo exportaste en el paso anterior y que puedas compartirlos como indicamos al final, ya sea usando Zenodo, o por cualquiera de los métodos alternativos.

Para publicar los datos de Twitter usaremos **Zenodo**, que es un repositorio para distintos objetos de investigación, provisto por los creadores del CERN. Tiene un conjunto de ventajas interesantes, como proveer el DOI, que asignará a cada conjunto de datos un identificador único para rastrearlo después. Sin embargo, si no puedes publicar en Zenodo por alguna razón, puedes usar otros servicios populares, libres como **Internet Archive**, o privados como Dropbox, Google Drive, etc. Incluso puedes enviarnos los datos exportados a nuestra dirección de correo en dataselfie@tupale.co (ten presente los límites que colocan los proveedores de correo respecto al tamaño de los archivos anexos en ese caso).

5.1 Antes de empezar

Para publicar nuestros datos previamente solicitados a Twitter, debemos registrarnos en Zenodo.

5.2 Subir datos a Zenodo

Desde la cuenta de Zenodo, iniciamos el proceso de publicación:

1. Clic en botón “Upload” ubicado junto a la barra de búsqueda.

The screenshot shows the Zenodo website interface. At the top, there is a blue navigation bar with the Zenodo logo, a search bar, and buttons for 'Upload' and 'Communities'. Below the navigation bar, the main content area is divided into two columns. The left column, titled 'Recent uploads', lists two items: 'Colour 0.3.11' and 'PyFraME: Python tools for Fragment-based Multiscale Embedding'. Each item includes a 'View' button and a brief description. The right column contains three promotional cards: 'Zenodo now supports DOI versioning!', 'Using GitHub?', and 'Zenodo in a nutshell'.

^Arriba | Perfil en Zenodo (clic para agrandar).

2. Clic en “New Upload”. Veremos un formulario como el que muestra la pantalla a continuación:

^Arriba | Formulario de carga de archivos (clic para agrandar).

3. En la parte donde dice “Drag and drop files here”, arrastramos el archivo ZIP que descargamos de Twitter en el paso previo

o usamos “Choose files” para seleccionar el archivo ZIP enviado por Twitter.

Después del cajón de carga se encuentra el formulario de descripción del archivo a publicar. Aquí se incluyen aspectos como: tipo de carga, fecha, descripción y nivel de acceso, entre otros.

Es importante seleccionar “Dataset” en la sección “Upload Type”. Dependiendo de la selección que hagamos acá, se habilitan o no otro tipo de datos. El DOI lo asignará Zenodo después de publicado.

El formulario diligenciado lucirá similar a este (clic para agrandar):

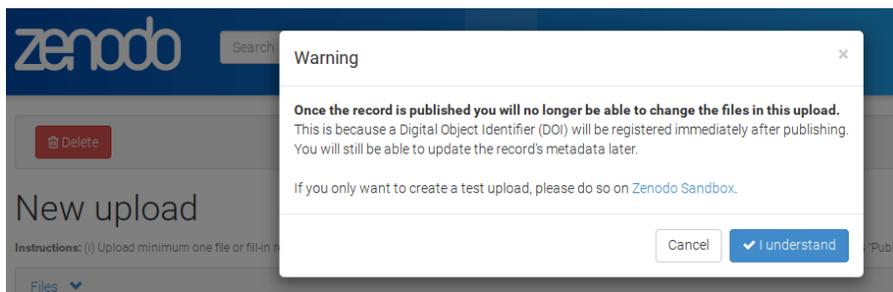
The screenshot shows the Zenodo 'New upload' form. The form is divided into several sections:

- File:** A large area for uploading files, with a 'Choose files' button.
- Community:** A field to enter a community name.
- Upload type:** A dropdown menu with options: Publication, Poster, Preprint, Dataset, Image, Video/Audio, Software, Lesson, and Other. 'Dataset' is selected.
- Basic information:** This section is expanded and contains:
 - Digital Object Identifier:** A text field with a placeholder 'e.g. 10.1234/abcd'.
 - Publication date:** A date picker showing '2016-02-25'.
 - Title:** A text field with the value 'My Twitter data export'.
 - Author:** A text field with the value 'Open Access Research' and a DOI field with '10.5281/1000-0000-1000-0000'.
 - Description:** A rich text editor containing the text: 'This is my Twitter public data as exported by Twitter Inc. It is used with research purposes on Data Activism and political discourse on social networks.'
 - Version:** A text field.
 - Language:** A dropdown menu with 'eng' selected.
 - Keywords:** A list of keywords: 'twitter', 'data science', 'political participation', and 'social networks'.
 - Additional notes:** A text field.
- License:** A section with radio buttons for 'Open Access', 'Embargoed Access', 'Restricted Access', and 'Closed Access'. 'Open Access' is selected.
- Funding:** A section with a text field containing 'European Commission (EU)' and a 'Start typing a grant number, name or abbreviation' prompt.

The bottom of the page features a footer with navigation links (About, Blog, Help, Developers, Contribute) and logos for Zenodo, OpenAIRE, and the European Union.

^Arriba | Formulario de carga de datos diligenciado (clic para agrandar).

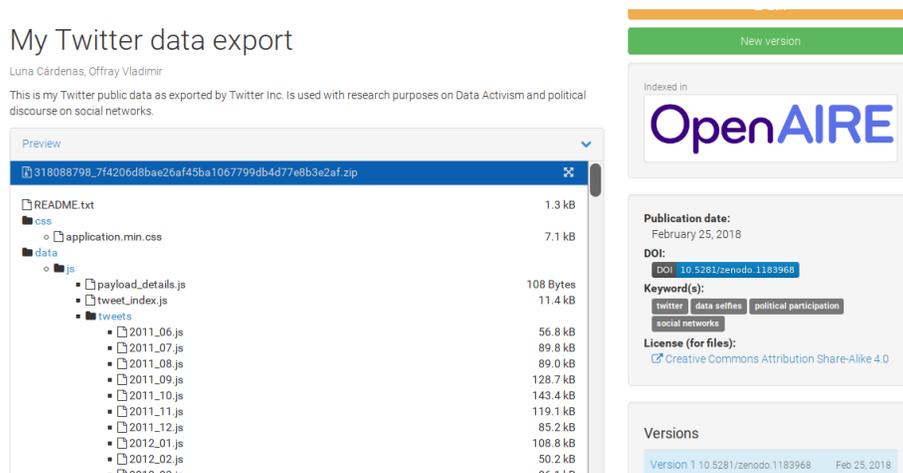
Antes de publicar, nos advertirá que los datos subidos a Zenodo no pueden ser borrados, aunque los metadatos sí pueden ser cambiados, con una ventana similar a la siguiente:



^Arriba | Advertencia sobre lo que se puede cambiar y lo que no en Zenodo (clic para agrandar).

Hacemos clic en el botón “I understand”.

Mostrará una vista previa de los datos que acabamos de publicar:



^Arriba | Vista previa de datos publicados (clic para agrandar).

Ejemplos de archivos de Twitter publicados en Zenodo:

- @Buitrago.Silvia
- @offrayLC
- @Samuel.Nacar
- @ferranmorales
- @azuledu

5.3 Listo para solicitar tu Data Selfie

Una vez Zenodo confirma la publicación de tus datos (o si usaste cualquiera de los métodos alternativos), copia la URL de la publicación del navegador y compártela en Twitter usando el

hashtag #AceptoLaDataSelfie. Nos pondremos en contacto contigo en cuanto generemos tu Data Selfie.

6 Preguntas sobre los Data Selfies

¹(./vis18:dataselfies)

Este documento muestra algunas preguntas y técnicas de visualización que se pueden emplear para resolver inquietudes sobre los data selfies. También otras técnicas para visualizar y fuentes de datos para realizar preguntas.

Consideraremos un catálogo de visualizaciones:

- **Por función:** Qué quiero mostrar.
- **Por técnica:** Cómo puedo revisarlos.

La intención es revisar el catálogo anterior u otros y mirar si podemos hacernos nuevas preguntas o responder las antiguas con las técnicas que el catálogo presenta, de modo que aumentemos el repertorio de data selfies posibles.

Pregunta	Técnica	Por qué
¿Cuántas variables hay un tuit?		Nos permite cartografiar las preguntas posibles
¿Cuándo son más activos los retuits con respecto a la época electoral?	Bar Char	Es una visualización rápida y fácil de entender
¿Cuándo/Dónde están las contradicciones en su discurso?		

Otras preguntas abiertas:

- ¿Dónde están ubicados los participantes en una conversación?
- ¿Hasta dónde acotar la cantidad de tuits o retuits en un visualización?
- ¿Qué tanto aportan los colores en la distinción del tipo de mensaje?
- ¿Están al mismo nivel los conceptos de tuit, retuit, y *reply*?
- ¿Cuándo parte de tu discurso público se vuelve en tu contra?

7 Procesarlos y visualizarlos

7.1 Live Coding

Técnica de programación performativa, en la que los resultados de lo que escribe en código se ven en “tiempo real”. se popularizó en el escenario musical, pero no este reducido a este, exclusivamente. Pharo permite experiencias de Live Coding.

7.2 Instalar el software

7.2.1 Pharo

Pharo is a pure object-oriented programming language and a powerful environment, focused on simplicity and immediate feedback (think IDE and OS rolled into one). [name=Pharo Project]

Pharo es un entorno de programación objetual que soporta live coding y visualización de datos. Es la plataforma sobre la cual está hecho Grafoscopio.

7.2.1.1 Instalación el lanzador

Pharo es un sistema multiplataforma, que se ejecuta sobre Windows, Gnu/Linux y Mac. El lanzador de Pharo (*Pharo Launcher*) es un programa que nos permite administrar distintas versiones de Pharo dentro de cada sistema operativo. Veremos las formas de instalarlo en en cada uno de ellos.

7.2.1.1.1 En Gnu/Linux

- Creamos una carpeta donde esté contenido el programa:

```
mkdir -p ~/Programas/  
cd ~/Programas/
```

- Descargamos el instalador para Gnu/Linux de 64 bits:

```
wget https://files.pharo.org/pharo-launcher/linux64
```

- Descomprimos el archivo antes descargado:

```
unzip linux64
```

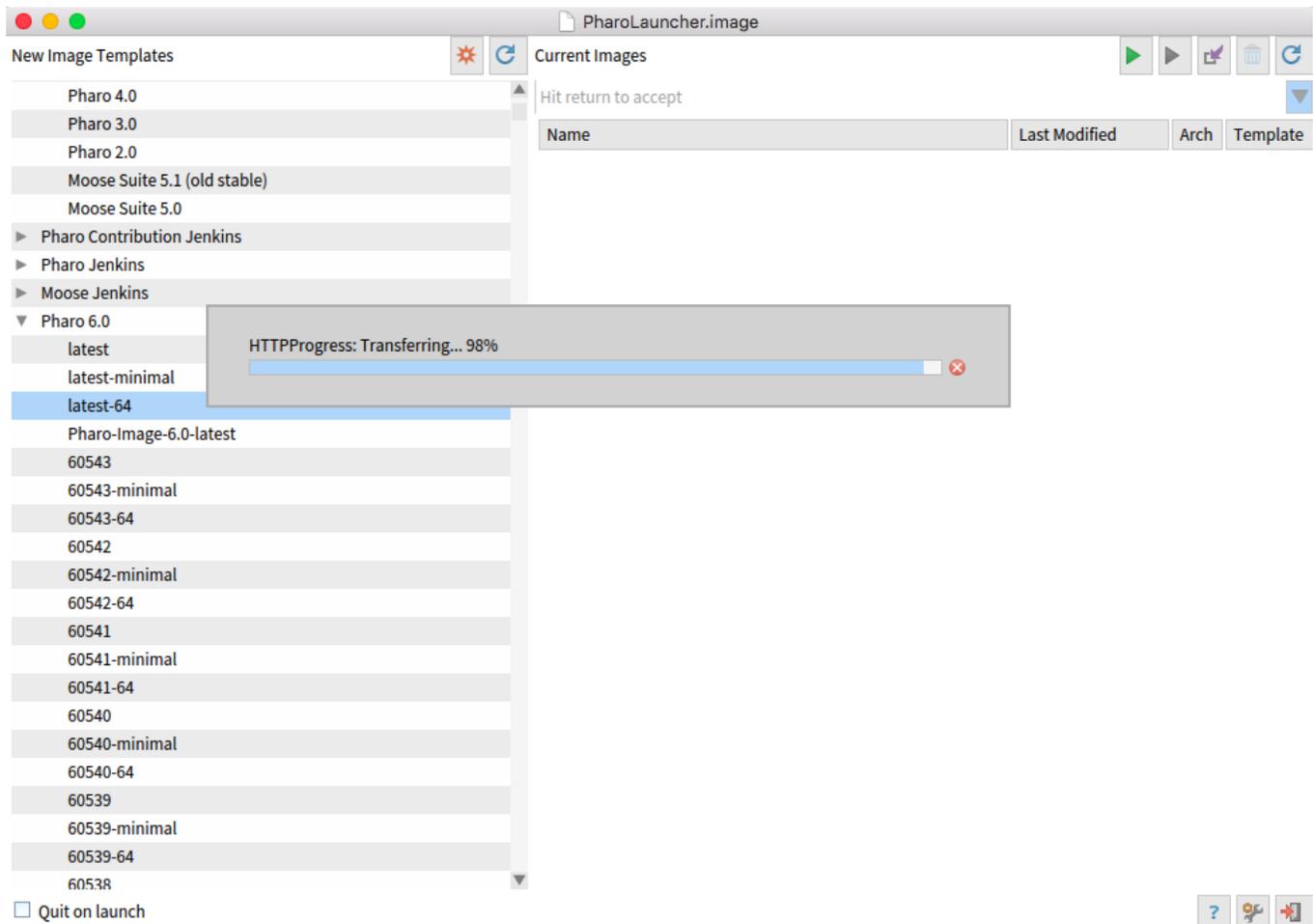
- Ejecutamos el lanzador de Pharo:

```
./pharolauncher/pharo-launcher
```

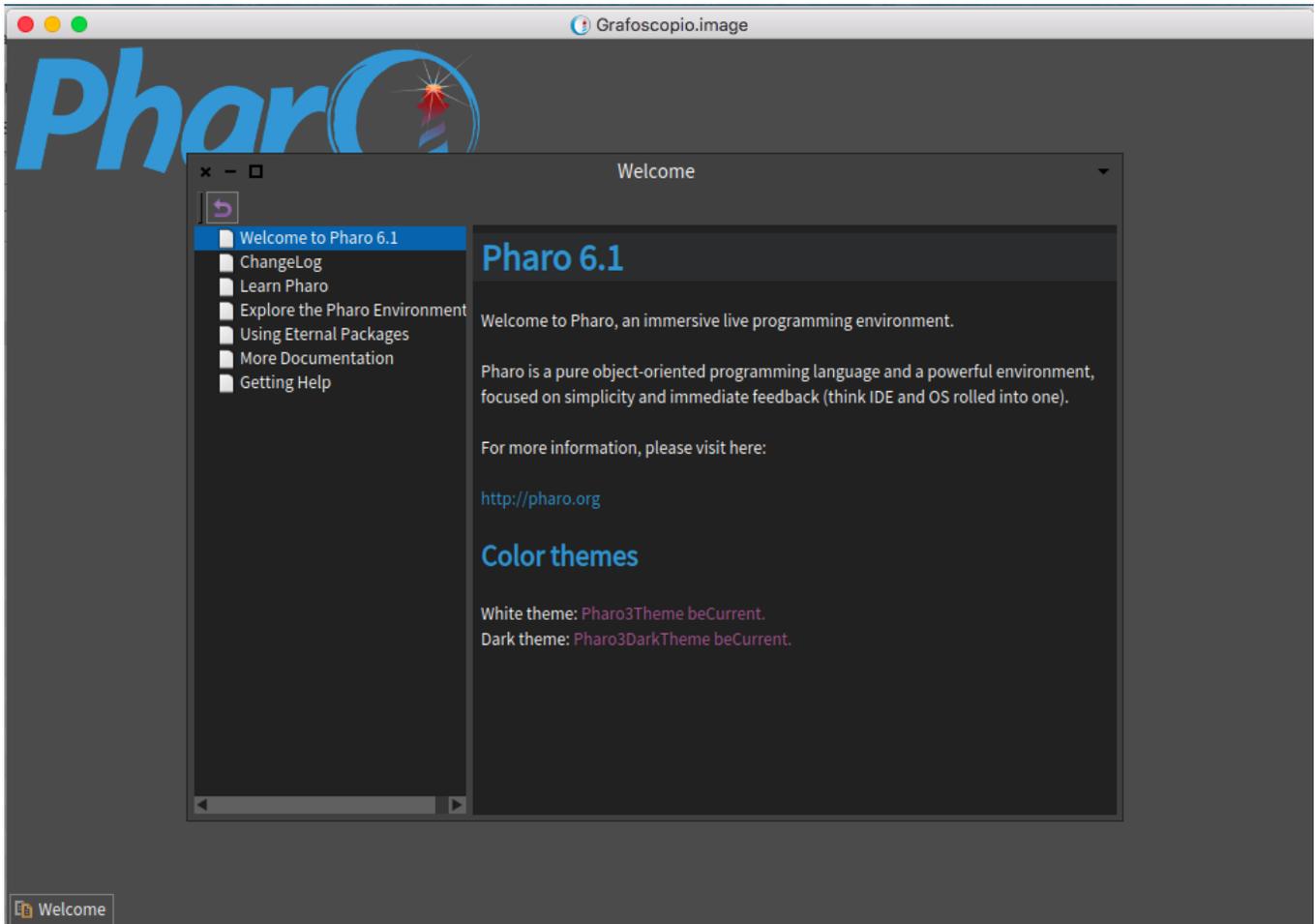
7.2.1.1.2 Mac

Entramos a la página de descargas de Pharo y seleccionamos la descarga para nuestra plataforma y seguimos los siguientes pasos:

- Seleccionamos Pharo 6.0 en el arbol de imágenes de la izquierda.
- Seleccionamos la imagen `latest-64` y hacemos click secundario y luego seleccionamos “load image”. Debera empezar a cargar una imagen como se ve acá:



- En la parte inferior estan los pasos para lanzar la instalación.



7.2.1.1.3 Windows



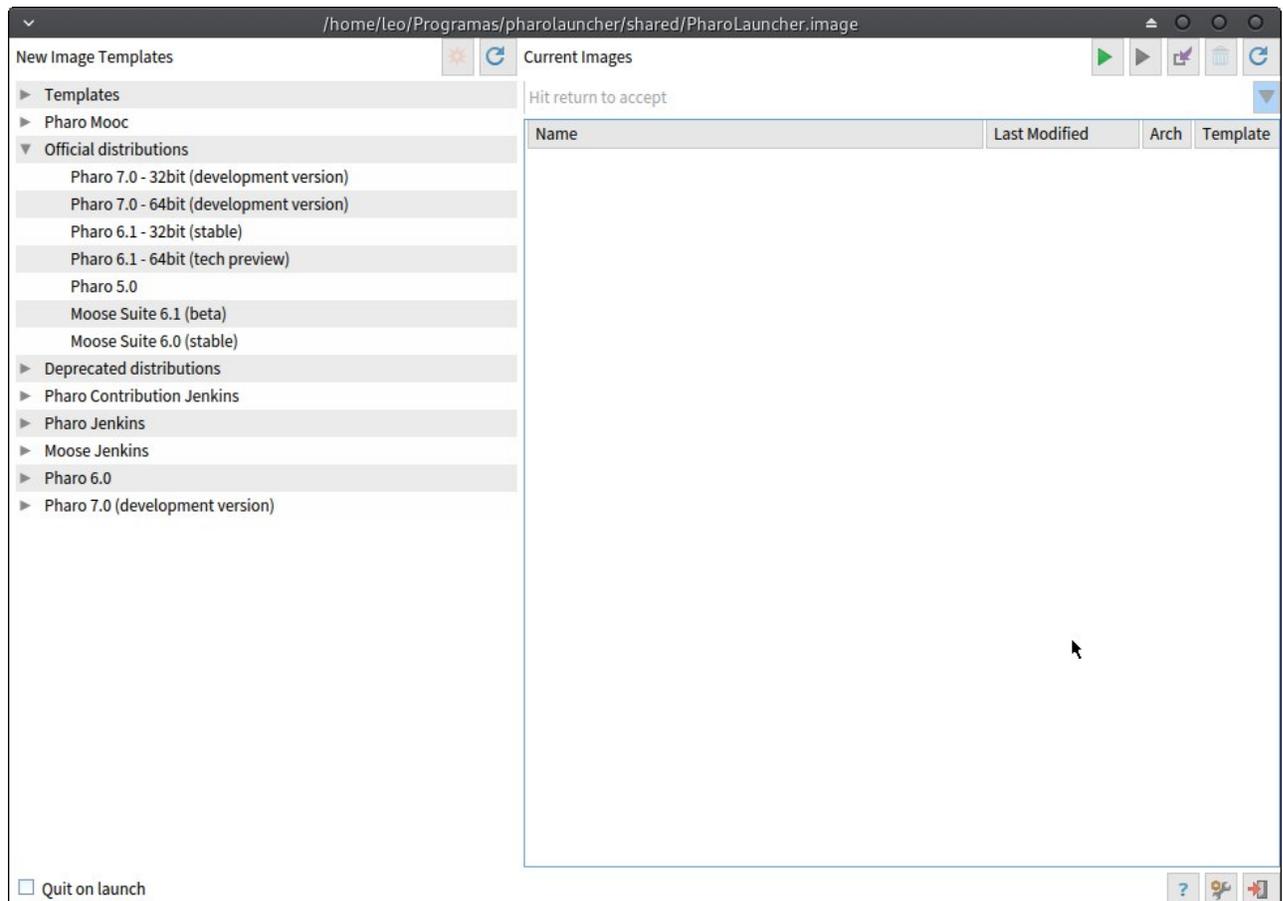
En este momento la instalación en Windows no funciona por problemas con la integración de la librería `libgit2`

7.2.1.2 Usar el *Pharo Launcher*

Pharo emplea algo llamado “la imagen”, que guarda el estado de todo el sistema (la posición de las ventanas, lo programas que hemos creado, los datos almacenados). Es como un computador virtual que podemos transportar entre diferentes sistemas operativos y retomar el trabajo justo donde lo dejamos. También es la manera de “empaquetar” aplicaciones en algún estado y distribuirlas para otros usuarios de Pharo. El *Pharo Launcher*, que instalamos en la sección previa, nos permite administrar diferentes imágenes, descargarlas, lanzarlas, borrarlas, etc.

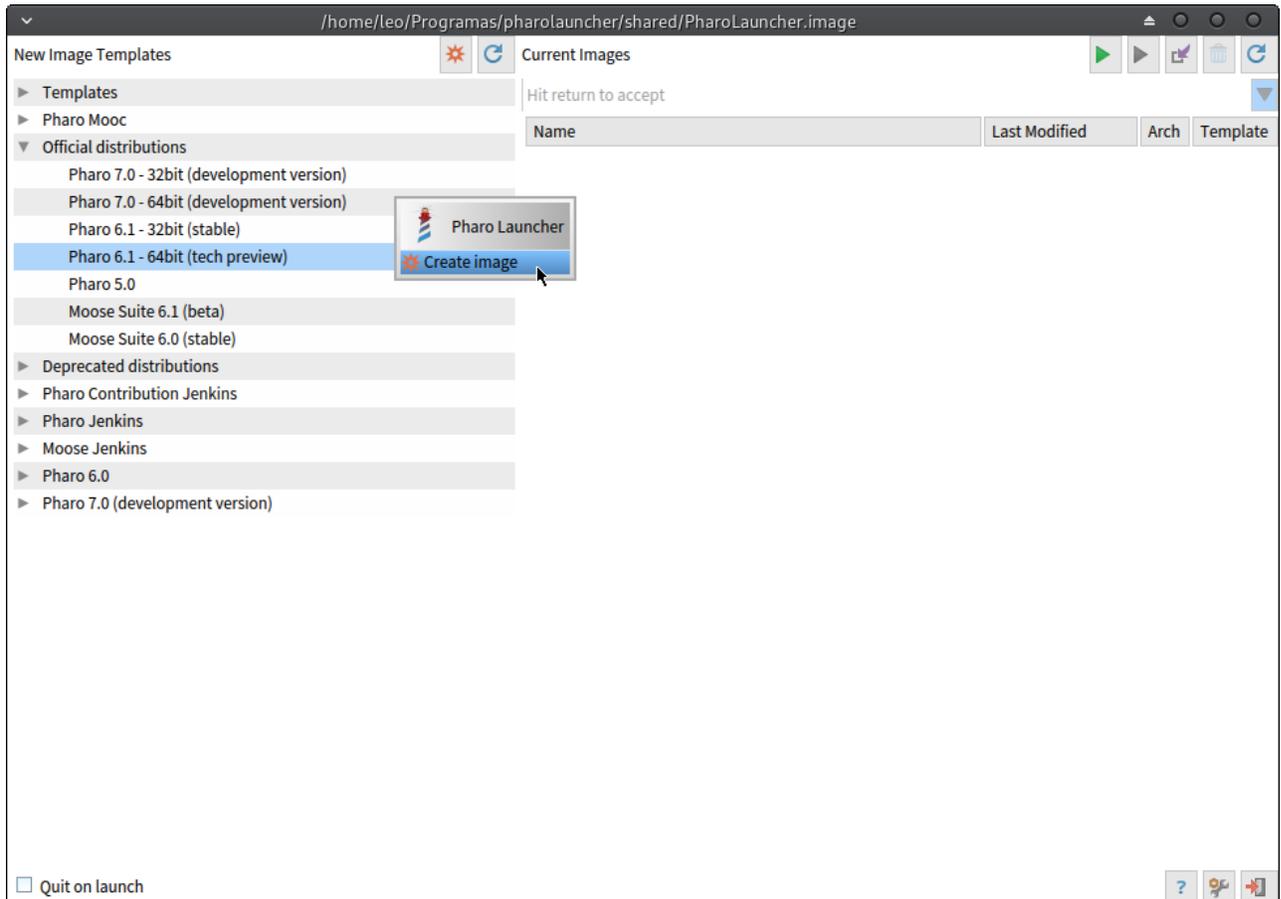
Con Pharo Launcher podemos lanzar imágenes personalizadas de Pharo con Grafoscopio instalado.

- Una vez hallamos abierto el *Pharo Launcher* al final de la sección anterior, deberemos ver una pantalla como esta:

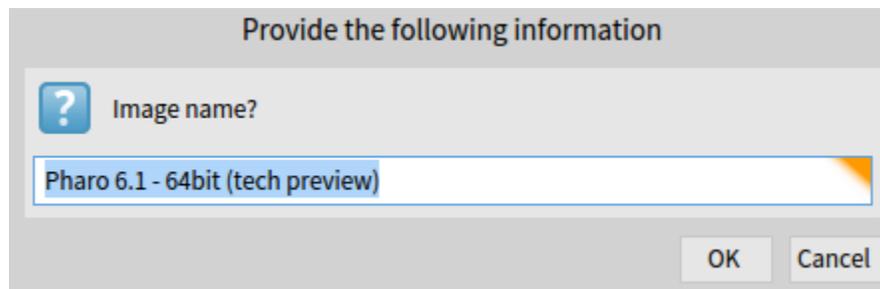


Pharo launcher

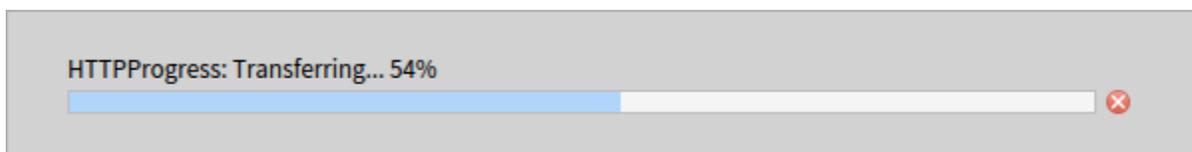
- Hacemos click con el botón secundario en el panel de la derecha sobre la imagen que dice Pharo 6.1-64bit (tech preview) y seleccionamos la opción que dice "Create image".



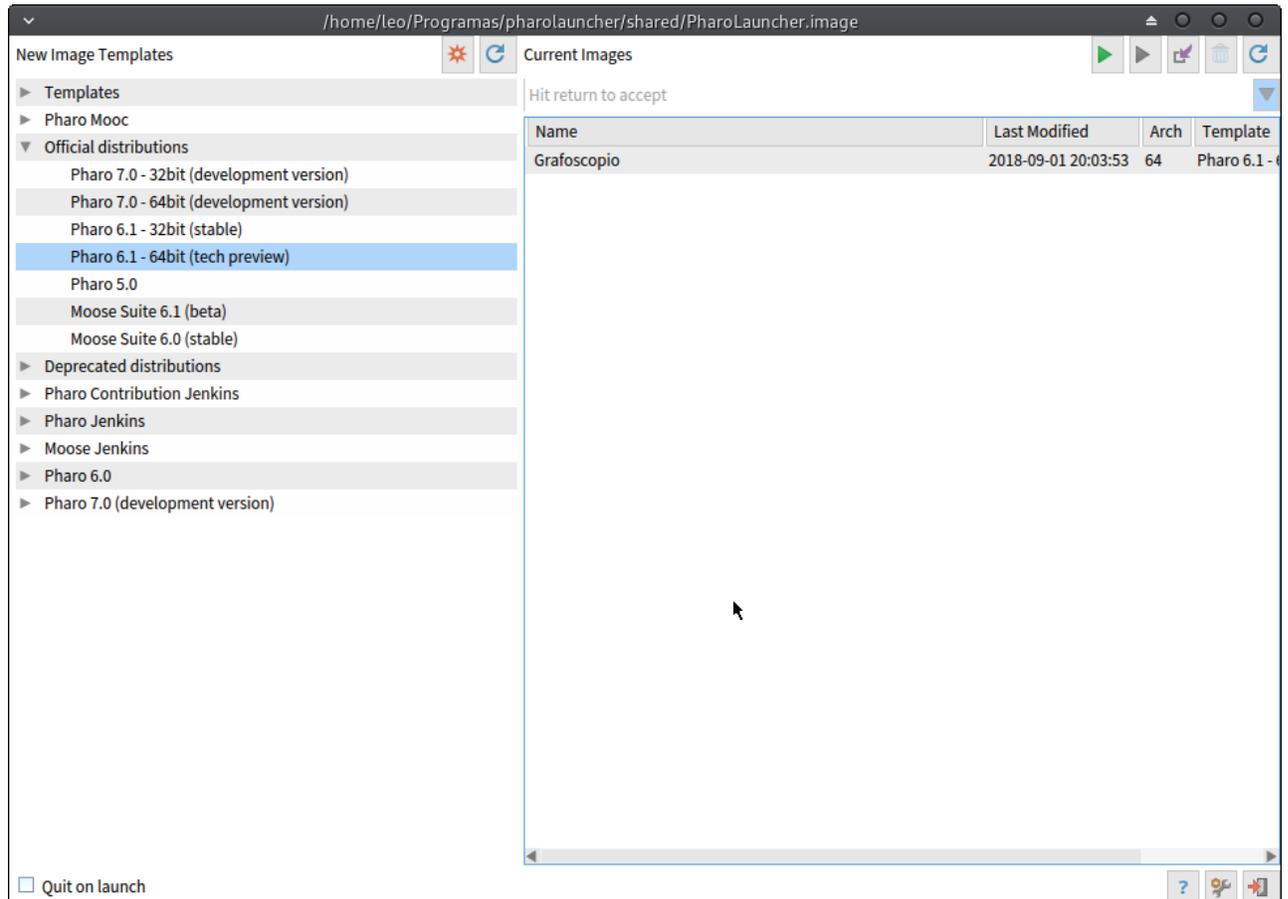
- Nos pedirá el nombre que queremos darle a nuestra imagen. Podemos escribir cualquier cosa, pero es bueno usar algún nombre mnemotécnico. Escribamos Grafoscopio, pues esa imagen la personalizaremos después con ese software.



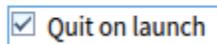
- Empezará una descarga de la imagen genérica (que luego personalizaremos).



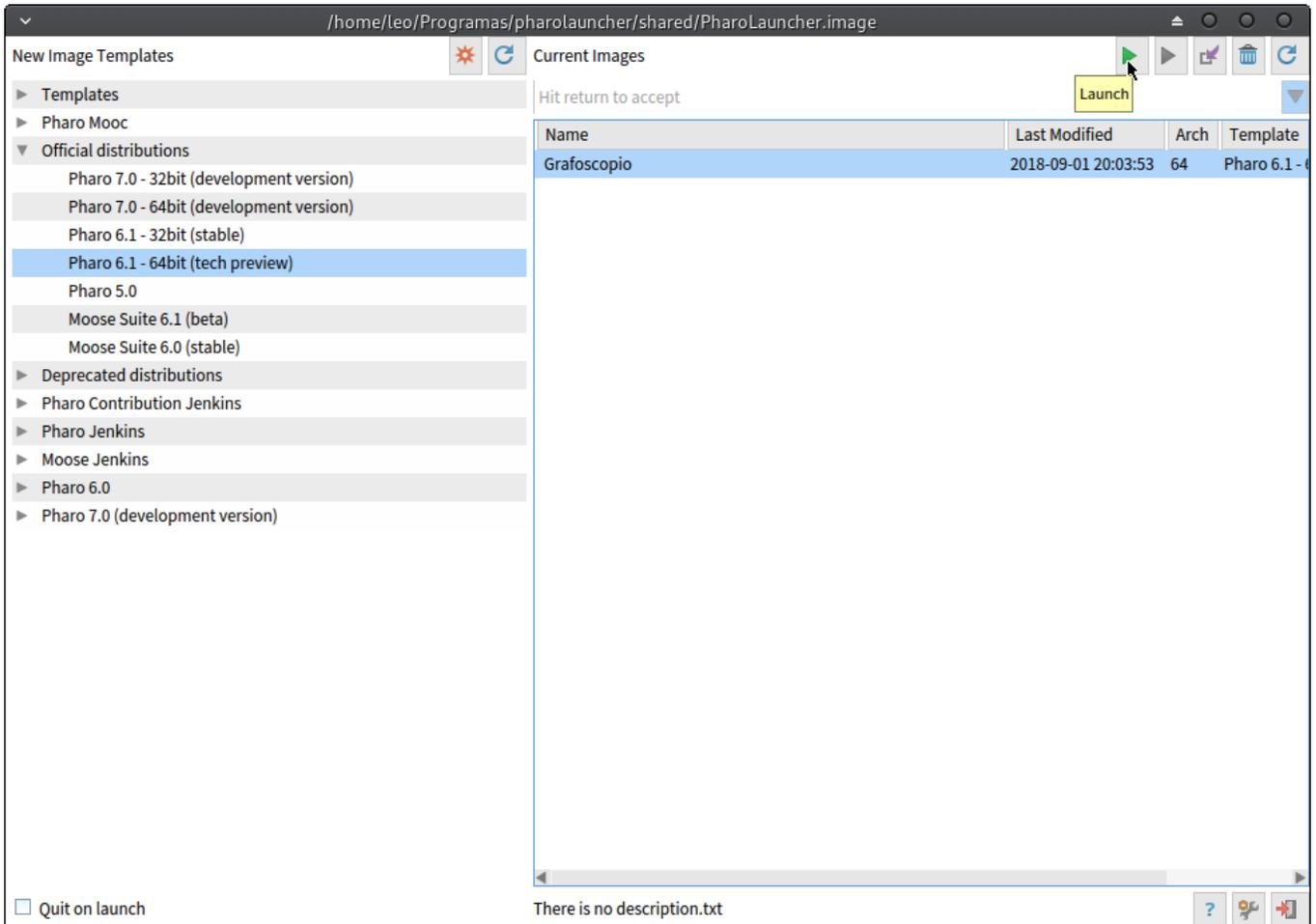
- Cuando haya finalizado, veremos la imagen que hemos descargado y renombrado en el panel de la derecha:



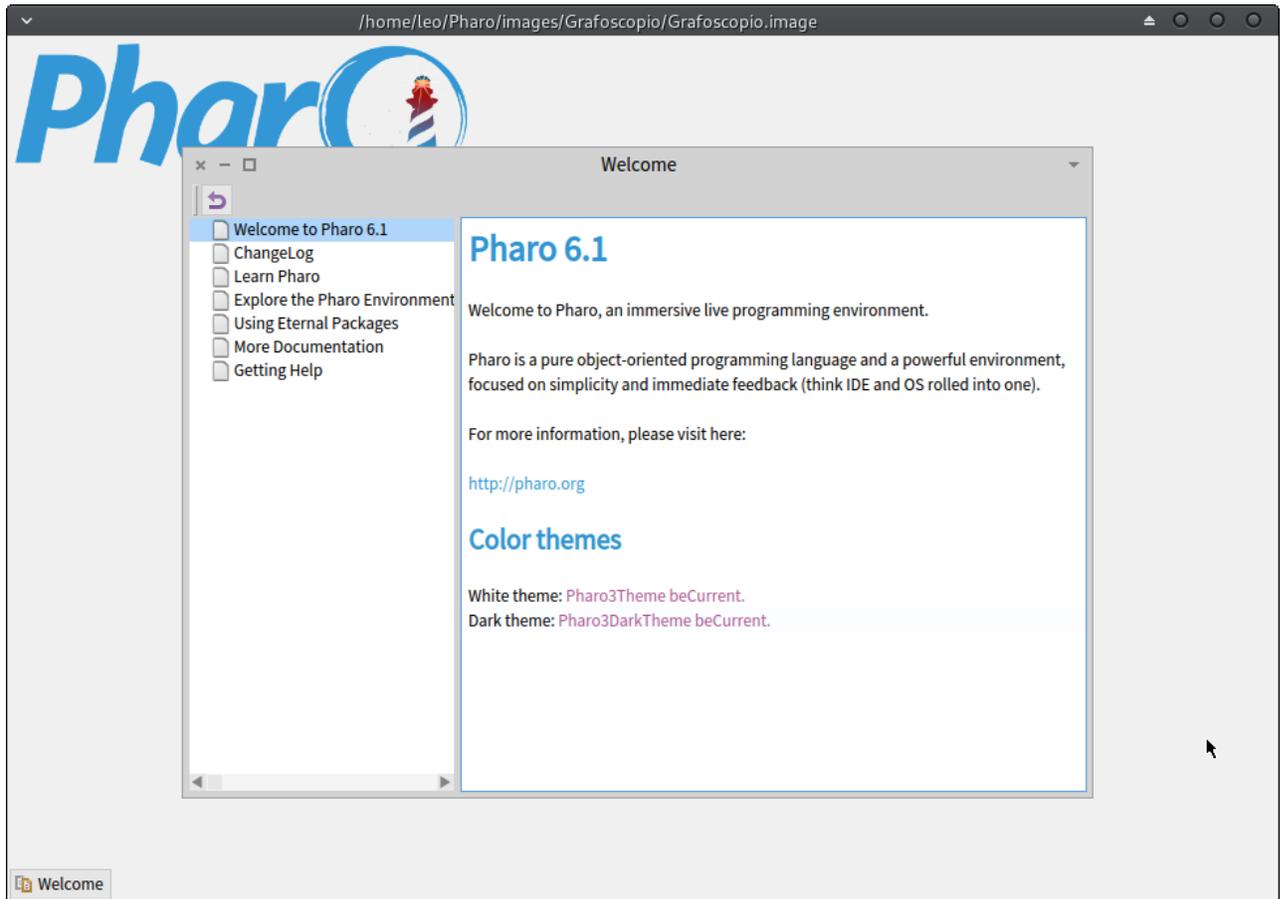
- Marcamos la casilla “Quit on launch”, que está en la esquina inferior izquierda, antes de lanzar la imagen de Grafoscopio.



- Hacemos click sobre la imagen recién creada, en el panel derecho y luego hacemos click en el botón con el ícono de Play que dice Launch.



- Veremos la ventana de inicio de Pharo:



- Acá podemos empezar a usarlo directamente o personalizarlo para instalar Grafoscopio.

7.2.2 Grafoscopio

See English version

“Grafoscopio es una herramienta amoldable para documentación interactiva y visualización de datos, que está siendo usada para ciencia abierta, ciudadanas y de garaje, investigación reproducibles, (h)ac(k)tivismo, innovación abierta y comunitaria, visualizaciones de dominio específico, y periodismo de datos, entre otros usos actuales y potenciales. Grafoscopio está cubierto por una licencia libre y de código abierto (MIT) y se socializa, realimenta y modifica en un taller-hackatón recurrente de una semana llamado el Data Week, que está orientado principalmente desde preguntas ciudadanas mediadas por datos y visualización.

Grafoscopio es y usa “infraestructuras de bolsillo”, sencillas y autocontenidas, que pueden ejecutarse On/Off-line, desde una memoria USB, una raspberry-Pi, un servidor modesto y cualquier otra infraestructura intermedia o más

potente.

”

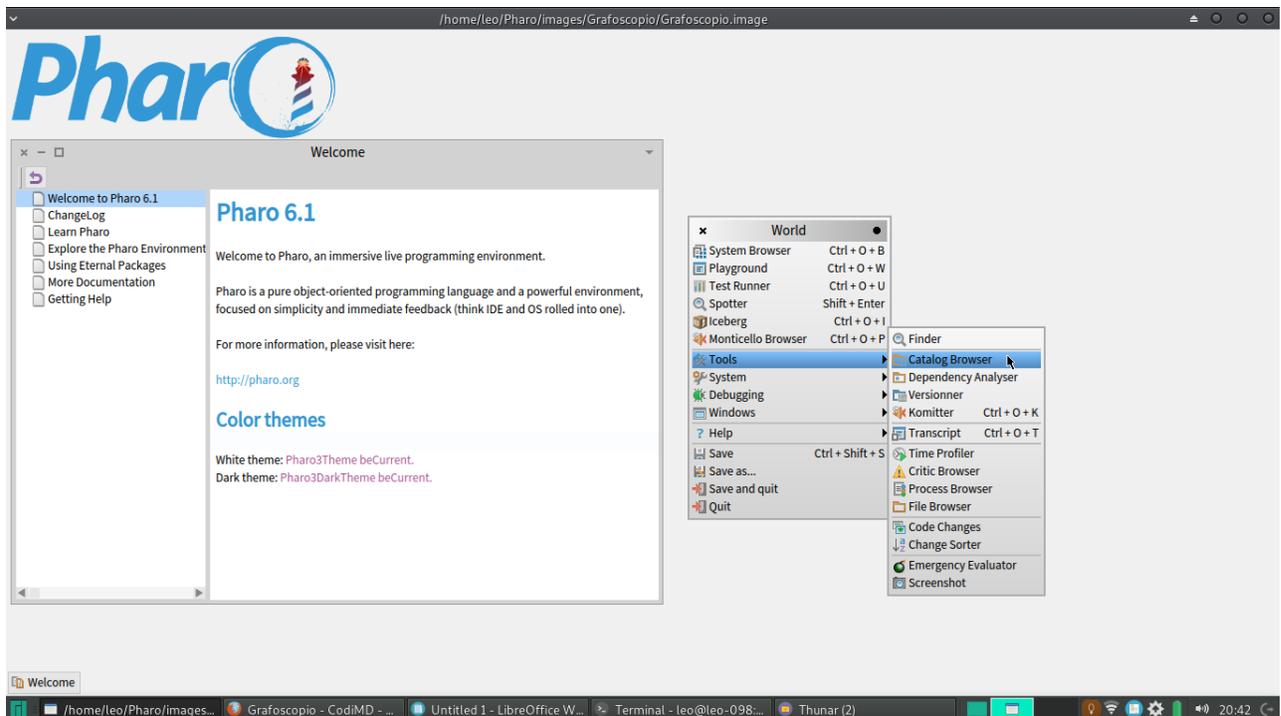
7.2.2.1 Instalación

Para la instalación supondremos que ya tienes instalado y abierto Pharo 6.1. De no ser así, sigue las instrucciones de instalación de Pharo.

- En sistemas Gnu/Linux, debemos tener previamente instalada la biblioteca `libcurl-gnutls` (en sistemas Mac y Windows no sabemos si es necesaria). Instálala usando el gestor de paquetes de tu distribución. Por ejemplo en Arch, Manjaro y sus derivados se instala con:

```
yaourt -S libcurl-gnutls
```

- Desde la imagen de Pharo seleccionamos el catálogo, haciendo click con el botón principal en cualquier espacio libre (no ocupado por una ventana dentro de Pharo) y luego en `Tools > Catalog Browser`:



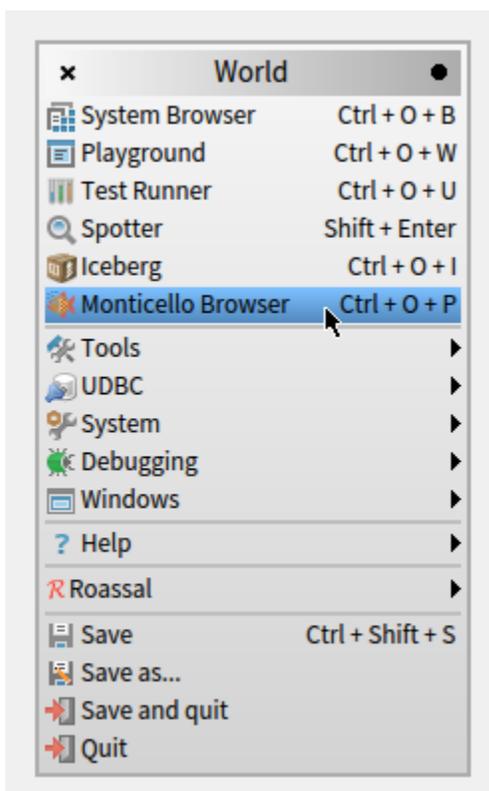
- De acá en adelante se pueden seguir los pasos del Manual de Usuario de Grafoscopio para terminar la instalación.

7.3 Actualizar el software

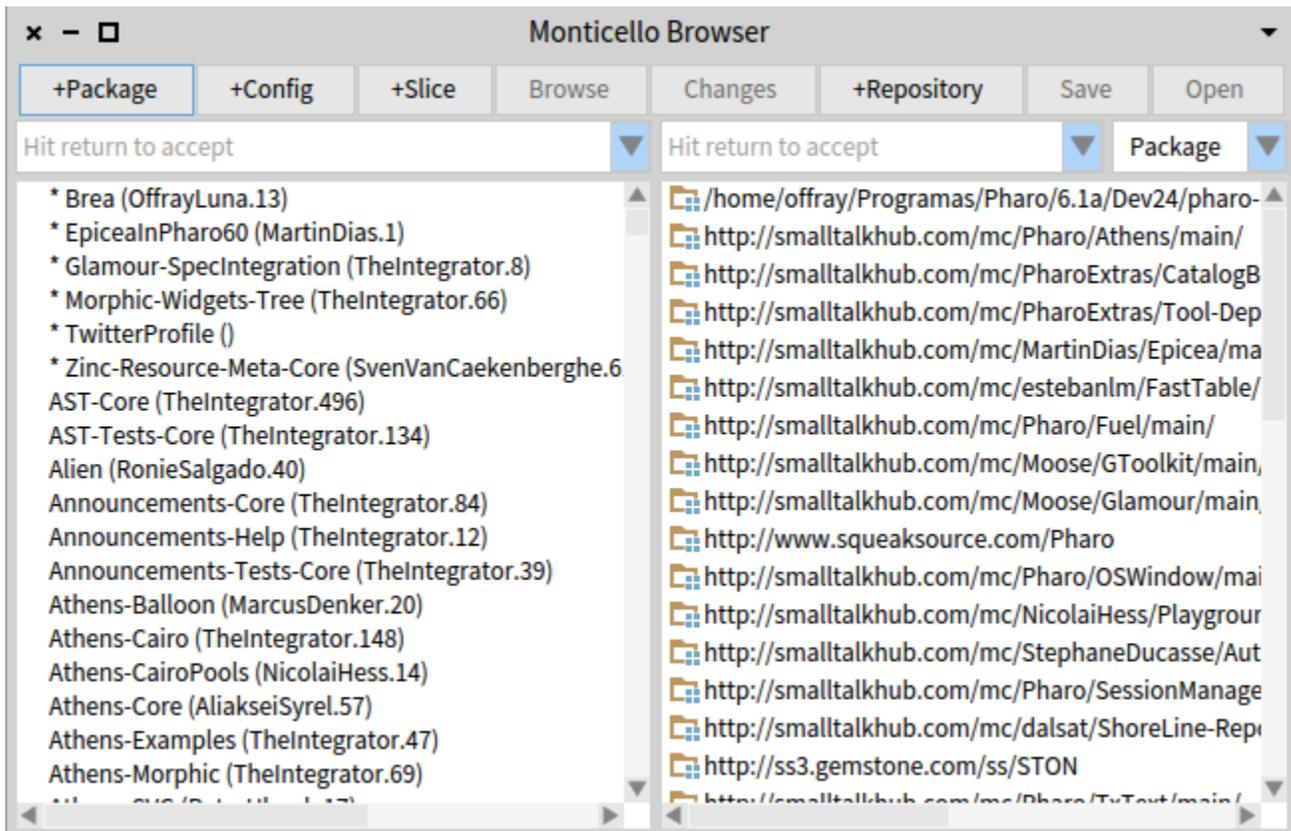
7.3.1 Monticello

Monticello es un sistema de gestión de paquetes y actualización de software (para los usuarios de Gnu/Linux, es como un Git/Fossil combinado con un apt-get/Yaout).

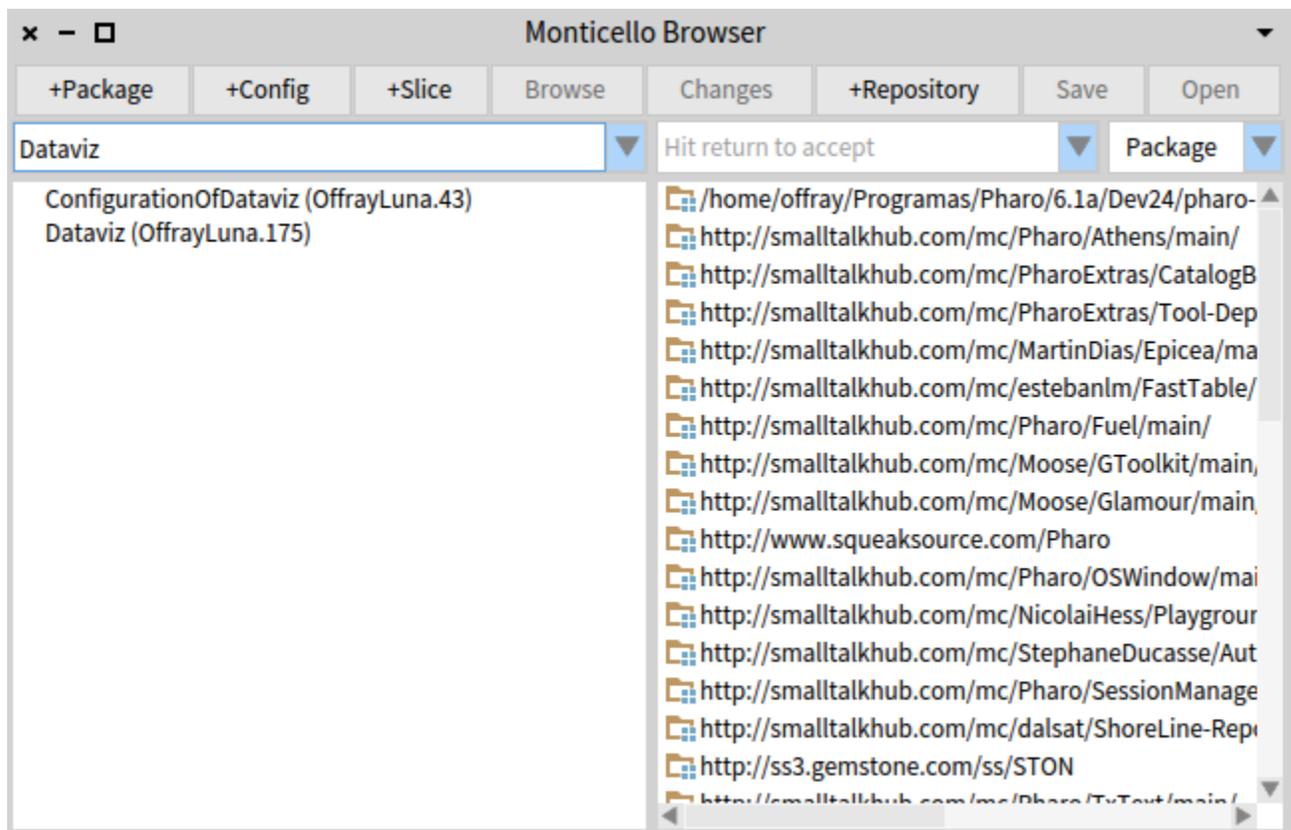
Para lanzarlo, desplegamos el menú del mundo, haciendo click en cualquier espacio libre del entorno, es decir no ocupado por ninguna ventana ni barra y seleccionamos Monticello, tal como aparece en esta pantalla:



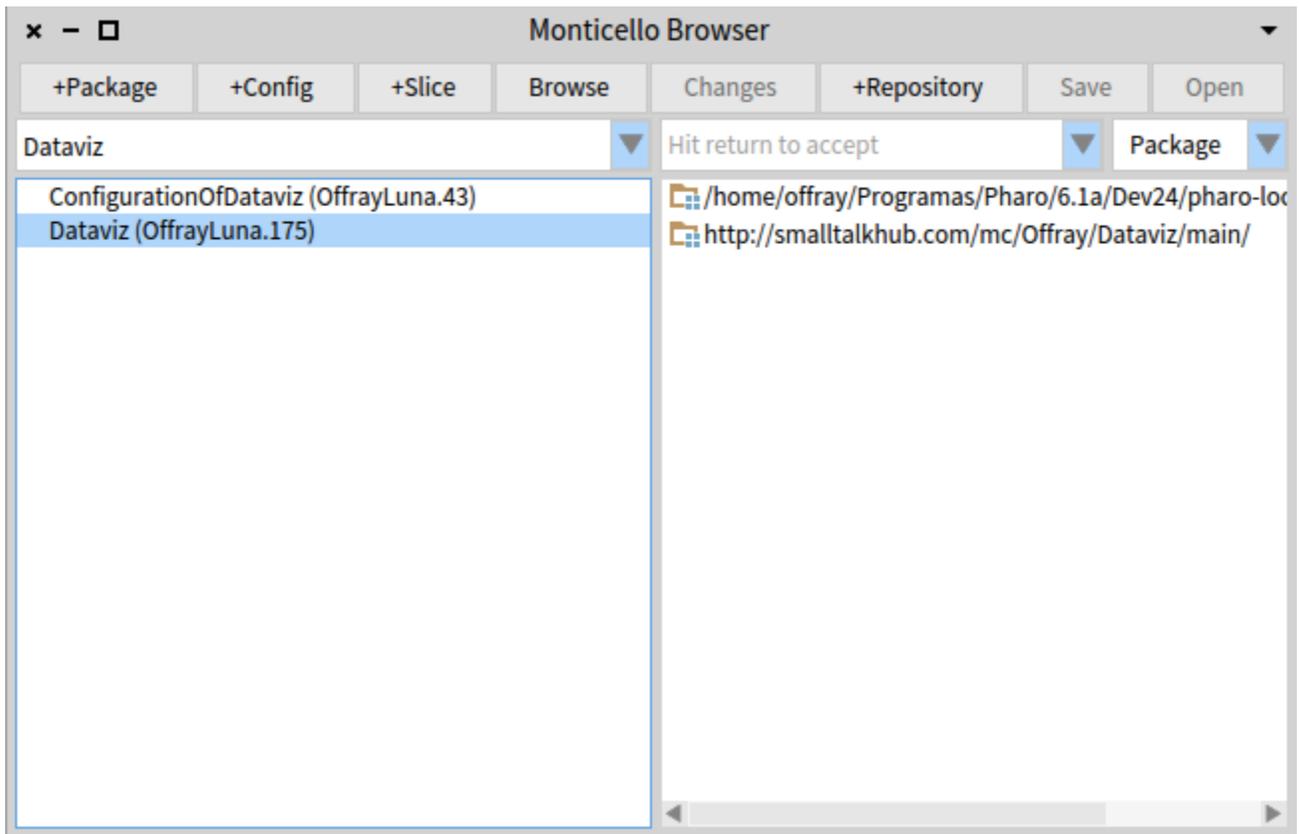
Aparecerá la venta principal de Monticello:



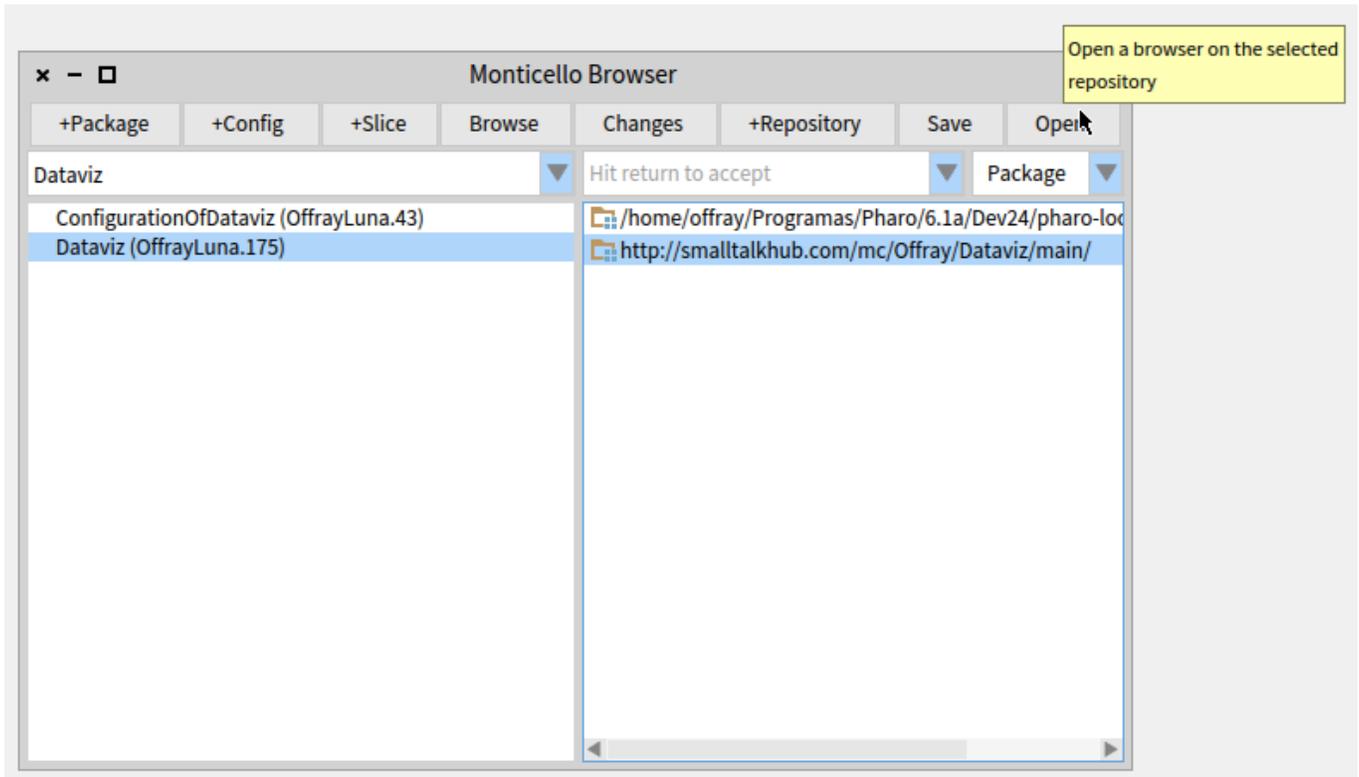
Escribimos en la caja de búsqueda del panel de la izquierda el nombre del paquete que queremos actualizar, por ejemplo Dataviz y Monticello se actualizará progresivamente:



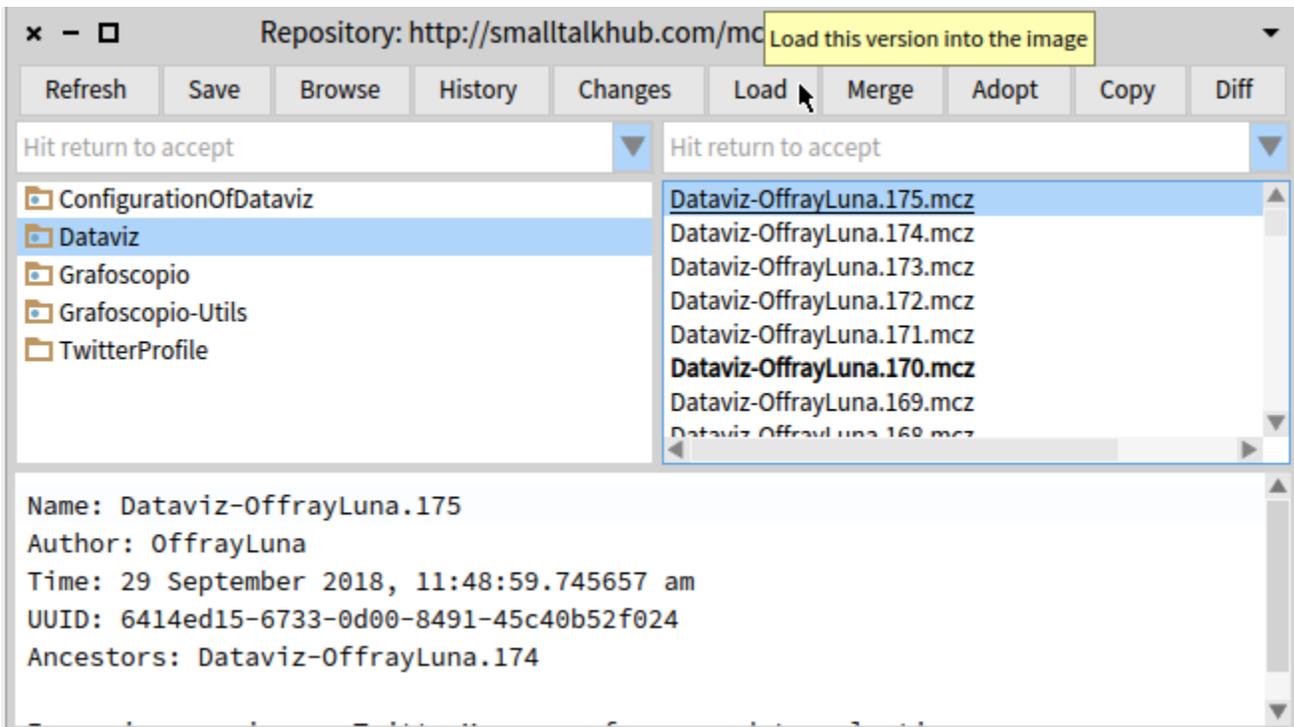
Hacemos click en el paquete cuyo nombre coincida con aquello que queremos actualizar, el listado de repositorios en el panel de la derecha que proveen dicho paquete se acotará, como aparece en esta ventana:



Seleccionamos el repositorio desde el que nos queremos actualizar, usualmente el repositorio remoto (que empieza por http):



Aparecerá una ventana de dos paneles. El de la derecha contiene el listado de todas las versiones de un paquete particular. Nos paramos en la última versión y le damos Load, como se ilustra a continuación:



7.4 Mob programming para visualización

Es una técnica en la que varias personas comparte un teclado y dan ideas sobre el código que se escribe, por una de ellas (el conductor), que es un puesto que se puede rotar.

- [Mob programming en la Wikipedia.](#)
- [Libro Mob programming en LeanPub.](#)

7.5 Libretas interactivas en Grafoscopio

El siguiente es un listado de algunas libretas interactivas hechas en Grafoscopio:

- [Techniques for data activism.](#)
- [Apprentice Notebook.](#)
- [Data Selfies.](#)
- [Manual de Periodismo de datos.](#)

8 Material del Stand para la Exposición

La intención de la expo es mostrar lo que hemos hecho para varios públicos que puedan asistir a la misma. Dicho material debe ser entregado a las personas del Medialab para su impresión y montaje en la semana que inicia el 8 de octubre.

En una conversación preliminar dijimos que nuestros aportes y trabajo habían sido principalmente en 3 frentes:

- **Metodológico:** Acá nos referimos al detrás de cámaras, a cómo hemos creado lo que hemos creado.
- **Tecnológico:** Las infraestructuras que hemos apropiado, extendido y cambiado.
- **Conceptual:** Las preguntas, datasets, herramientas y visualizaciones que planteamos.

Preparar la expo es preparar los materiales que soportan mostrar el trabajo realizado en esos tres frentes (podrían surgir más).

9 Documentación de los prototipos UX

- Diseño Data Selfies: Documentaremos todos los elementos que utilizaremos para realizar los datasefies. Adjuntamos documentación en formato PDF (donde puedo alojar el PDF ???)

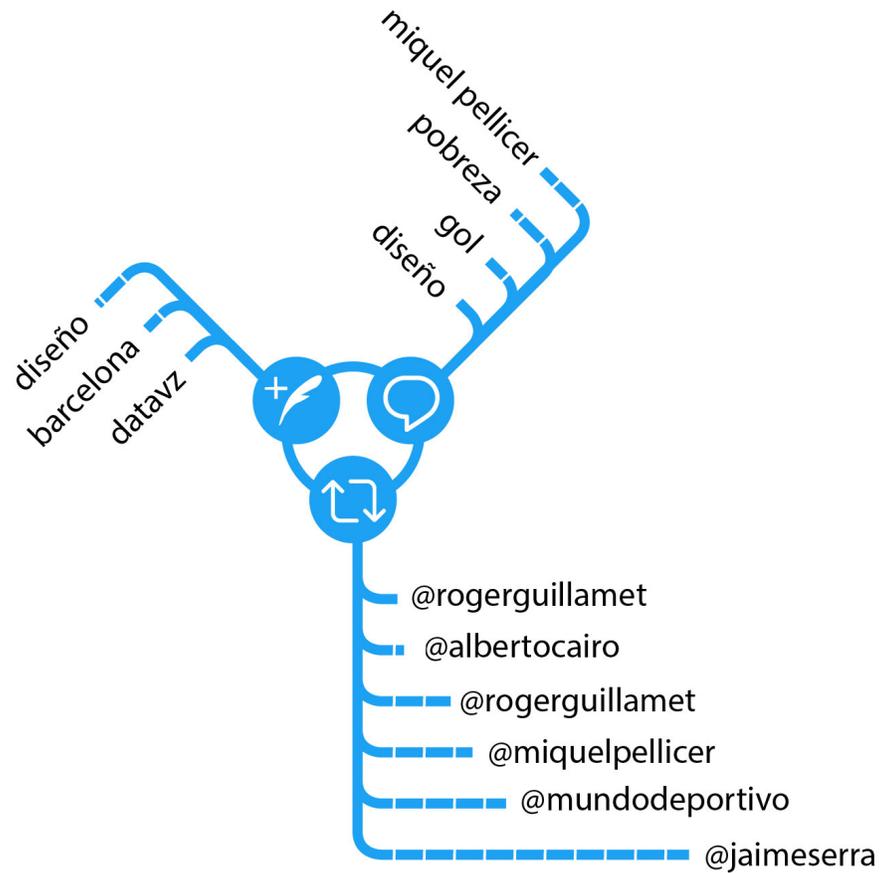


DATA SELFIES EN TWITTER

++++k ksdkmsmdkadmkadklamkdmakldmkalskdamsklmadmkladski

Ferran Morales

@ferranmorales



 Tweet (1.200)

 Coments (200)

 Retweet (900)

- **Imágen fondo titulo en PNG.**

El texto que aparece debajo del título podemos realizar una pequeña explicación del trabajo.

MI DATA SELFIES DE TWITTER

- **Imágen fondo inferior en PNG.**

(creo que será mejor pasar los pictos por separado y realizar el montaje por separado):



- **Picto tweet. PNG:**



SVG:



- **Picto comentarios. PNG:**



SVG:



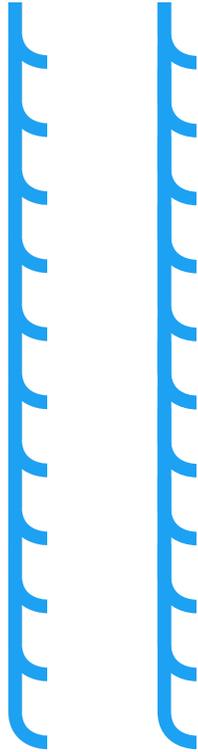
- **Picto retweets. PNG:**



SVG:



- **Línea completa en PNG y SVG:**



PNG: SVG:

- **Linea recortada en PNG y SVG.** PNG:  SVG: 

- **Picto:** PNG:  SVG: 

• **Detalle ubicación elementos.** Guía para poder saber dónde colocar los elementos que tenemos en la visualización.

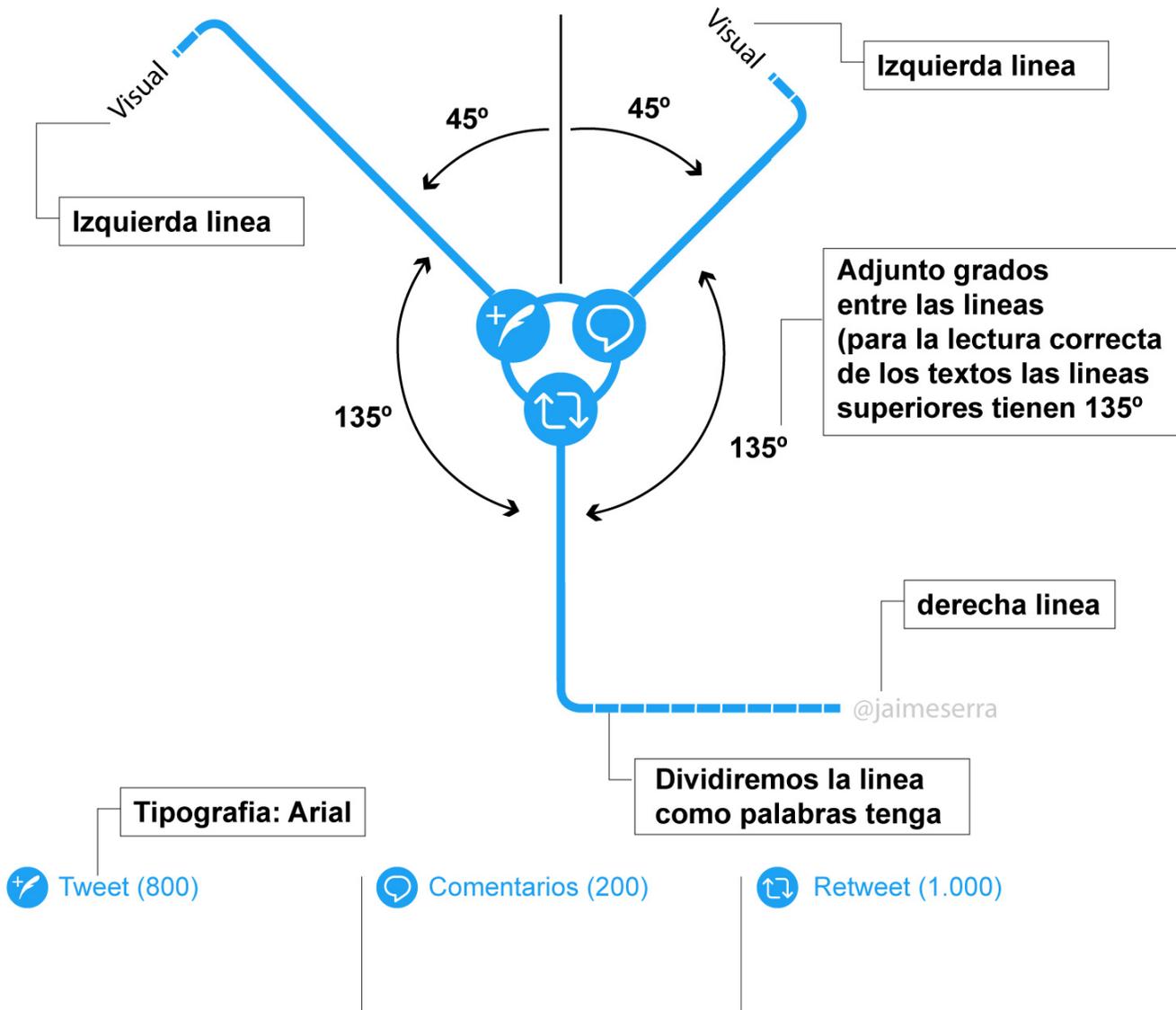


MI DATA SELFIES DE TWITTER

++++k ksdkmaksmdkadmkadklamkdmakldmkalskdamsklmadmkladskl

Ferran Morales
@ferranmorales

variable nombre



9.1 Metodológico

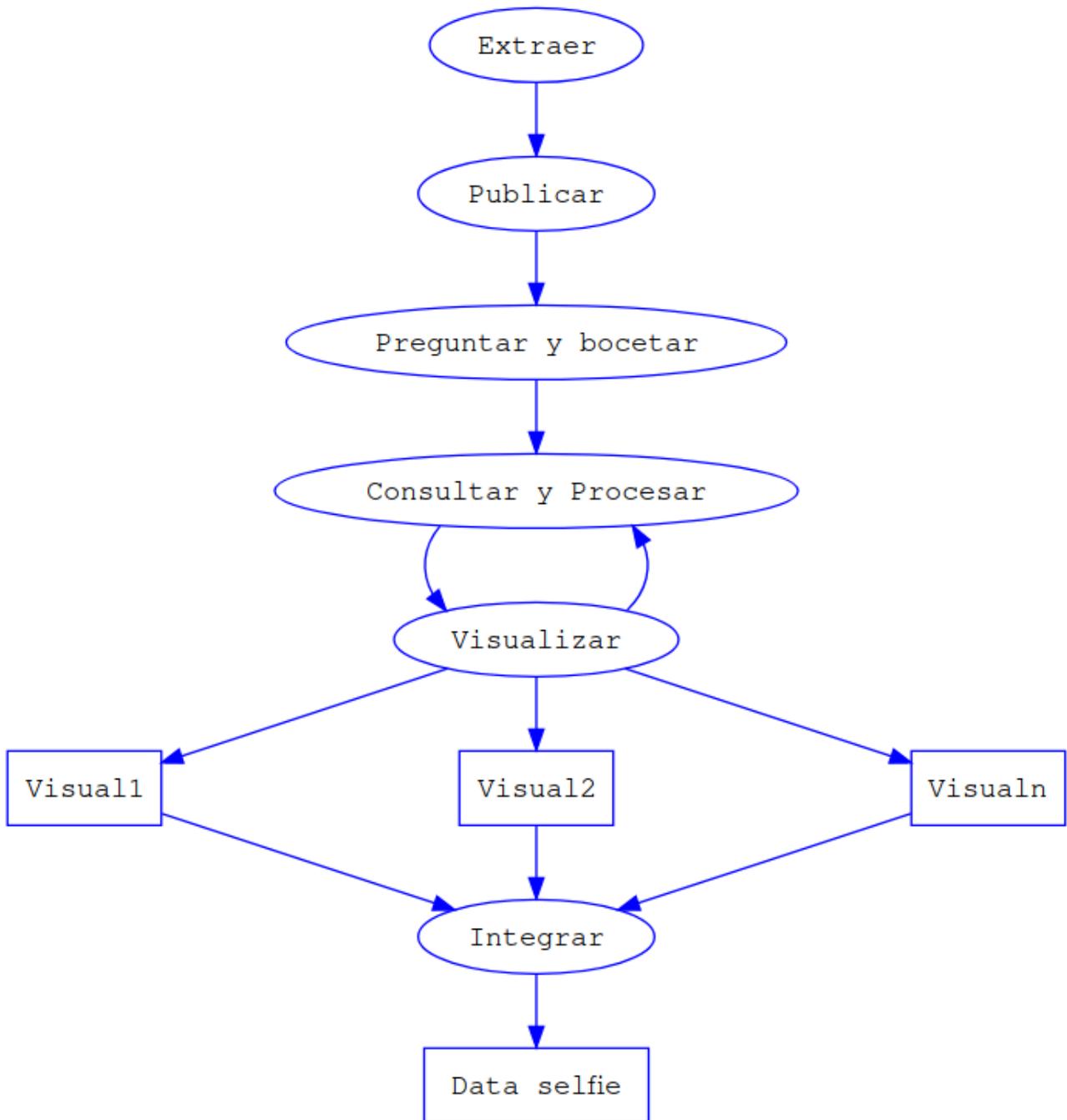
- Mob Programming: Foto de nosotros haciendo mob-programming y explicación corta extraída de acá.
- Live coding:

- Trozo corto de video o gif animado que muestra live coding o una sucesión de imágenes estáticas que lo muestran.
- Documentación ágil: Libritos en PDF de
 - Bitacora del trabajo por días.
 - Cuadernillo por temáticas.

9.2 Tecnológico

9.2.1 Diagrama de Flujo

- Diagrama de flujo de trabajo con datos:
 - **Extraer:** Tomar nuestros datos de Twitter y exportarlos (ver documentación).
 - **Publicar:** Hacer nuestros datos públicos disponibles en repositorios de datos comunes (ver documentación).
 - **Preguntar y bocetar:** Concebir un listado de preguntas para hacerse con sus datos (o su ausencia) y unos bocetos a mano de cómo expresar la respuesta.
 - **Consultar y procesar:** Seleccionar un subconjunto de los datos y realizar operaciones sobre ellos .
 - **Visualizar:** Crear distintas visualizaciones que nos permitan entender el comportamiento de los datos o expresar digitalmente los bocetos en papel.
 - **Integrar:** Componer todas las visualizaciones en un único *Data selfie*.



Flujo de trabajo para trabajar con datos: de las preguntas a las visualizaciones.

9.2.2 Paralelo de visualizaciones entre las ofrecidas por Twitter y las nuestras

Comparamos las visualizaciones ofrecidas por Twitter y las creadas por nosotros mismos.

9.3 Conceptual

Extender la tabla de preguntas y visualizaciones posibles realizadas en los días previos.

formatted by [Markdeep 1.03](#)