Tech tools for human rights documentation

A snapshot of the landscape

January 2021

PILPG
A Global Pro Bono Law Firm

THE ENGINE ROOM

HURIDOCS
HUMAN RIGHTS INFORMATION AND DOCUMENTATION SYSTEMS

Table of Contents

Acknowledgements

Many thanks to all of the tool authors who offered their time to talk to us about their tools, learnings, challenges and approaches. We hope in turn that this report offers insights that might support your work.

THE ENGINE ROOM

theengineroom.org

Tool research: Cathy Richards, Barbara Paes
Interviews: Laura Guzmán, Cathy Richards, Helen Kilbey
Writing: Helen Kilbey
Editing: Laura Guzmán
Review: Cathy Richards, Laura Guzmán, Julia Keseru, Quito Tsui
Design: Dimitri Stamatis

PILPG

publicinternationallawandpolicygroup.org

Review: Nicole Carle, Bethany Houghton, Manek Minhas

HURIDOCS

huridocs.org

Review: Lucía Gómez Vicente, Kristin Antin

What this report is about and who it's for

In 2019, The Engine Room partnered with PILPG and HURIDOCS to conduct research into how civil society is using technology tools for human rights documentation. The full findings of this research **can be found here**.[1]

This shorter report has been excerpted and adapted to focus solely on specific tools and tool development in the human rights documentation space. We present findings from our explorations of some of the tools currently in use, and from our interviews with eight tool authors (more on our research methodology can be found at the end of the report).

This report has been written specifically for:

- **Civil society organizations** looking for tech tools to support their documentation work
- **Tool developers** working in the human rights documentation space, and
- **Funders and donors** looking for insight into tool development for human rights documentation by civil society.

Tools snapshot

In our research, we found a technological environment in flux. In the last few years, a number of older tools in the human rights documentation space have been retired, while at the same time a batch of newer tools have begun to establish themselves.

Given how broad this landscape is and how quickly it continues to change – in

[1] *Human Rights Documentation Solutions: Phase 1 Report, PILPG,* The Engine Room and HURIDOCS (Nov 2020) https://www.publicinternationallawandpolicygroup.org/hrds-phase-i-report-launch

terms of both available tools and user needs – this research cannot provide a fully comprehensive picture of all tool development in the space. It is offered, however, as a snapshot of some of the tools currently available to support civil society documentation work.

Which tools are included, and which are not

Organizations and individuals currently use a wide range of tech tools in the course of their human rights documentation work, from Google Docs and Microsoft Excel to bespoke database systems created by organizations in-house.

This snapshot focuses, however, only on tools *intentionally designed* to address the needs of civil society documenters working in a human rights or social justice context, and which have been created to accommodate a range of types of rights violations (in other words, the snapshot does not include tools designed around one specific type of evidence; for example, **torture** or **sexual violence**). The snapshot also focuses only on tools that are currently being used in the field – tools still in development were not included.

In line with the criteria used above, the tools we highlight in this snapshot:

- **are almost all free and open source**, meaning that anyone can download the code and, with the right technological knowledge and resources, set up their own instance of the tool.
- **are non-exploitative** in that their underlying business models do not make money through collecting data or locking organizations into ongoing, prohibitive charges.
- **have been developed with input and feedback from documenters** themselves.

Where tools fit into documentation workflows

All of the tools we highlight are designed to support at least one (usually more than one) stage of a typical documentation workflow, which we identified as including the following:

When looking at tools that support this type of workflow, we can roughly put them into two buckets:

- Those that primarily support **collection & verification**, and
- Those that primarily **support management, analysis & visualization**.

Many tools have functionalities from both these buckets, so these should be seen as broad areas of focus, rather than completely distinct categories.

Both types of tools can include **sharing** functionalities, which allow documenters to do things like send information through another application or export data into a report.

Collection & Verification

Collection can include creating and/or collecting photos, videos and audio testimonies, creating and using standardized forms for data entry, and crowdsourcing data, among other possibilities.

Verification can involve, on a technical level, adding a layer of data that can be used to corroborate the veracity of the documentation; for example, metadata and cryptographic signatures.

> Tool Examples: Digital Evidence Vault, eyeWitness to Atrocities, KoBoToolbox & KoBoCollect, ProofMode, Save, Tella, Ushahidi.

Management, Analysis & Visualization

Management tools can support documenters to store and manage their collected data in a way that enables them to do things like find what they need, organize what they've collected and create linkages between pieces of evidence.

Analysis & visualization functionalities help documenters gain insights into the data, including through visualization (for example, charts or graphs).

> Tool examples: KoBoToolbox, Ushahidi, Uwazi.

Tools at a Glance

The table beginning on the next page offers a broad introduction to some of the tools currently being used in the human rights documentation space.

eyeWitness to Atrocities

License: Code is not open source

Launched in 2015, eyeWitness is a mobile camera app focused on collecting verifiable photo and video documentation of international atrocity crimes. Documenters use the app to capture photos and videos with added metadata. This footage is then sent to the eyeWitness team, who – in collaboration with documenters, if contact details have been provided or a partnership agreement entered into – use it to compile reports for international investigators. **eyewitness.global**

Tool author

- eyeWitness to Atrocities.
- The tool is backed by the International Bar Association, which is where the eyeWitness project originated.

Support & services

Through partnership agreements, eyeWitness offers documenters various levels of support, from sharing media back with organizations to manually categorizing and analyzing data or providing support in case-building.

Notable features

- Adds metadata to photos and videos at the moment they are captured.
- Photos and videos are stored in a secure gallery separate to the phone's default gallery, which is only accessible via a passcode.
- Fast deletion of the app and its contents.
- Photos and videos are uploaded in encrypted format to secure servers managed by eyeWitness. Documenters can share copies of footage with others via email or social media, or receive copies of their footage from eyeWitness by way of a partnership agreement.

Connectivity requirements

- Collection can be done offline, but upload requires an internet connection.

KoBoToolbox

License: Code is open source

Launched in 2014, KoBoToolbox is a suite of tools designed to facilitate the collection of data in the field. Users can create custom forms, collect data through the dedicated KoBoCollect app or via web form, store their data, conduct light analysis, and export their data in a range of different formats. The tool is based on the Open Data Kit. **kobotoolbox.org**

Tool author

• KoBoToolbox at the Harvard Humanitarian Initiative.

Support & services

• Free, unlimited hosting is offered to humanitarian organizations.
• KoBo hosts a community forum where users can support each other, as well as a help center maintained by volunteer and user contributions.

Notable features

• Facilitates the creation of standardized forms for data collection in the field.
• Collection can be done via the KoBoCollect app, or via web forms. The KoBo backend also works with mobile collection app Tella.
• Users can process collected data via a KoBoToolbox backend.
• Secure transfer of data (using SSL) available with the setup of SSL certificates.
• Allows for bulk export of data in a variety of formats, including Excel, CSV, KML, ZIP (for media) and SPSS – this allows for data to be analyzed and visualized using other commonly-used tools.
• Currently working on speech-to-text functionality.[2]

Connectivity requirements

• Collection can be done offline, but connection to the KoBoToolbox backend requires an internet connection.

[2] *TWB and KoBo Inc Develop Speech Recognition Technology to Capture Voices of Speakers of Marginalized Languages,* TWB Communications (Sept. 1, 2020). https://translatorswithoutborders.org/twb-and-kobo-inc-develop-speech-recognition-technology-to-capture-voices-of-speakers-of-marginalized-languages/

ProofMode (in beta)

License: Code is open source

Launched in 2017, ProofMode is a mobile app that adds verification metadata to photos and videos taken with a phone camera.
guardianproject.info/apps/org.witness.proofmode

Tool author

• Guardian Project

Support & services

• Users can submit issues to the application's GitHub repository.

Notable features

• Adds metadata to photos taken using the default phone camera (i.e. ProofMode is not a dedicated camera app).
• App works in the background and requires little setup.
• Photos and metadata can be shared via the Android share functionality or as a CSV export, or backed up via OpenArchive's Save app.

Connectivity requirements

• No internet required.

Save

License: Code is open source

Created in 2019, Save is a mobile app that facilitates the secure backup of images, videos, audio recordings and other formats (e.g. pdfs and phone notes) to a designated storage location, or allows them to be published via the OpenArchive. **open-archive.org/save**

Tool author

• OpenArchive

Support & services

• Website offers an FAQ and an introductory video that takes a user through setup.
• OpenArchive offers direct support to groups interested in setting up a self-hosted or cloud-based secure archive, as well as training and help with technical or user experience issues.

Notable features

• Users can capture media in-app or import media from a variety of other apps (such as the phone's built-in camera app), including photos, videos, audio recordings, pdfs and notes.
• Secure transfer of data using TLS on all platforms and, optionally, Orbot (Tor for mobile) on Android. Data can be sent to OpenArchive, Dropbox, or a self-hosted webdav-compatible server (e.g. ownCloud or Nextcloud).
• Allows for pseudonymous submission to shared folders.
• Allows for organizations to directly receive and organize submissions from unlimited users.
• Allows users to add some metadata to media manually.
• Android version integrates with ProofMode, which adds verification metadata to photos.

Connectivity requirements

• Internet connection required to upload content to a storage location.

Tella

License: Code is open source

Launched in 2019, Tella is a mobile collection app for Android designed with security features in mind, to protect those collecting data in repressive environments. **tella-app.org**

Tool author

- Horizontal

Support & services

- Tella provides general documentation, as well as direct support to partner organizations. Support can include server installation, training, and technical or user experience issues.

Notable features

- Includes camera function and stores footage in an encrypted location on the phone, separate to the default camera gallery.
- Offers the option to add verification metadata to photos, videos and audio recordings.
- Offers ability to collect data via custom forms.
- Users can import media, such as photos or audio recordings, from other apps on the phone.
- Users can disguise the app (change icon and name).
- Quick app shutdown or deletion.
- Secure transfer of data (using SSL) to a dedicated KoBoToolbox server, for storage and management.

Connectivity requirements

- Media collection can be done offline but upload requires an internet connection.

Ushahidi

License: Code is open source

Created in 2008, Ushahidi allows people to crowdsource data collection and plot reports on a map in real-time. **ushahidi.com**

Tool author

- Ushahidi

Support & services

- Thorough manual available for users and developers. Ushahidi can also provide services such as custom development and support with tool configuration.
- Ushahidi deployments can be hosted on the Ushahidi platform, or be self-hosted.

Notable features

- Can collect data from a variety of sources, such as webform, SMS, email and Twitter.
- Allows users to map submissions via geolocation.
- Has some data review and validation features.
- Secure data transfer using SSL/TLS between the browser and the Ushahidi server.

Connectivity requirements

- Mobile version of the tool (web app) supports offline data collection.

Uwazi / Uwazi Reveal

License: Code is open source

Created in 2017, Uwazi is a responsive web app for storing, organizing, analyzing and publishing collections of documents. Uwazi Reveal, launched in 2018, includes all the same features as Uwazi but allows users to keep their collection private (allowing access only to those with an account). **uwazi.io**

Tool author

- HURIDOCS

Support & services

- Thorough user guide available. HURIDOCS also provides direct support to partner organizations, ranging from advice on documentation methodologies, tools and strategies, to customizing software to meet an organization's needs.
- Uwazi deployments can be hosted by HURIDOCS, or self-hosted.

Notable features

- Tagging templates.
- Data visualization capabilities.
- Entries can be tagged with geolocation via the use of coordinates.
- Can be used to help uncover the frequency of references and patterns within content pieces as well as finding relationships between pieces of information in the collection.
- Allows import of various file types, such as PDF, .doc, .txt, .odt, .jpg and CSV.
- Data export in CSV format.
- API available to create custom connections to other platforms.
- Digital Evidence Vault plugin allows data to be saved directly from online sources.

Connectivity requirements

- Internet connection required.

The tools below came up in our research but for various reasons fell outside the scope of the types of tools we were focusing on. They are still, however, worth noting in the context of human rights documentation.

- **Check:** An open source collaborative reporting and verification platform, designed by Meedan to support journalists and academics in collecting, organizing and fact-checking content sent to them via Whatsapp and other sources, and in communicating their findings at scale through means such as automated responses. **meedan.com/check**

- **CiviCase:** An extension for CiviCRM, an open source, civil-society oriented customer relationship management tool. Used together, CiviCase and CiviCRM allow a user to maintain detailed information about relationships within a case-management workflow. While neither CiviCRM nor CiviCase were developed specifically within a human rights documentation context, the tool is worth mentioning for its case management functionalities. **docs.civicrm.org/user/en/latest/case-management/what-is-civicase**

- **Digital Evidence Vault:** A browser plugin designed, in collaboration with Uwazi, to allow users to import digital content with metadata directly from the browser into Uwazi.

- **ownCloud and Nextcloud:** Tools for file storage, sharing and collaboration. Nextcloud (2016) is based on ownCloud (2010). Though German companies ownCloud GmbH and Nextcloud GmbH offer paid-for subscription services aimed at a broad range of clients, both tools are open source and can be self-hosted on a private server for those organizations with the relevant technological skills and capacity. **owncloud.com / nextcloud.com**

- **The Whistle:** A reporting platform developed at the University of Cambridge that allows organizations to receive reports from a variety of sources, and then work with the data through a dashboard. The tool has been used in some pilot projects and is currently in further development, with potential to be used more widely in future. **thewhistle.org**

Two HURIDOCS tools, **Casebox** and **OpenEvsys** – worth noting here as they have been fairly widely used – have recently been sunsetted. Some of these tools' functionalities will be incorporated into an expanded version of Uwazi.[3]

- **Casebox:** Case management tool, designed for use in a legal context. Originally created by HURIDOCS and Ketse.com in 2011, HURIDOCS managed its own version of this tool specifically for partner human rights organizations from 2017.

- **OpenEvsys:** Free and open source database tool launched in 2009 by HURIDOCS, built on an events-based, "who did what to whom?" methodology for recording violations.

We also came across **information and support resources for human rights documenters working for truth, justice and accountability**. Of particular note are the following:

- **BIS (Basic Investigative Standards for International Crimes) app**: A mobile app for iOS and Android, developed by international legal partnership Global Rights Compliance, that provides detailed guidance around how to collect information in ways that "preserve its potential to be useful evidence in future national or international trials or accountability mechanisms." **globalrightscompliance.com**

- **WITNESS:** A non-profit organization aimed at helping people use video and technology to protect and defend human rights, WITNESS offers a number of relevant resources for documenters on its website. Its blog also offers up-to-date guidance on topics like Documenting During Internet Shutdowns[4] and Making Your Metadata Matter.[5] **witness.org**

[3] *Announcing the Sunset of Human Rights Software Casebox and OpenEvsys,* HURIDOCS (Oct. 22, 2020), available at https://huridocs.org/2020/10/sunset-of-casebox-openevsys-to-expand-uwazi/

[4] Yvonne Ng, *Documenting During Shutdowns,* Witness (Jan. 31, 2020), available at https://blog.witness.org/2020/02/documenting-during-internet-shutdowns

[5] Wendy Betts and Raquel Vazquez Llorente, *Making Your Metadata Matter,* available at https://blog.witness.org/2020/03/making-your-metadata-matter

Key considerations for human rights documentation tools

Security

Security is an important consideration for anyone working in the field of human rights documentation, given the sensitive and often risky nature of the work – but there is no single set of risks that all human rights documentation tools are designed to respond to.

As such, documenters need to weigh the specific risks they face in their context against the capacities and limitations of the tools they are using. This section looks at some of the ways in which documentation tools address security. It is important to note here that higher security can often come at the expense of convenience, and the acceptable trade-offs will be different for each documenter.

Encryption

Encryption is a key strategy for mitigating risks around evidence being accessed, tampered with, stolen or deleted by unauthorized parties. This can be employed in different ways:

- **Encrypting data within a tool.** Collection apps Tella and eyeWitness, for example, enable images and videos to be taken through the app, where they are automatically encrypted. This mitigates the risk of the data being accessed or tampered with if the device is stolen and broken into.

- **Making sure data travels through a secure connection** (for example, via TLS/SSL). This could be, for example, data traveling from a collection app to a designated secure server (where it is managed or archived), or data travelling between the browser and a website or web app. A secure connection mitigates the risk of data being accessed or tampered with in transit and is generally standard practice for any tool that takes data protection seriously.

- **End-to-end encryption.** End-to-end encryption goes a step further, security-wise, and encrypts data from one end device or system to another. Though this type of encryption is not used by any of the tools considered here – it can involve substantial usability trade-offs – it is used currently in a few secure messaging apps, such as **Signal**. End-to-end encryption has also been used by some well-known documentation tools in the past – for example, Martus, which was sunsetted in 2018 after 15 years.[6]

This resource from the Electronic Frontier Foundation (EFF) shows the key differences between transport-layer encryption (e.g. TLS/SSL) and end-to-end encryption.

Storing data securely

Once data leaves a user's collection device (e.g. a smartphone, tablet, or computer), it is generally stored in a server. With many of the tools considered, organizations have the option to set up their own server, but this requires a certain level of technical knowledge and skill.

Some tool authors in the human rights documentation space offer hosted instances of their tool – that is, an organization's data is hosted on servers maintained by the tool author themselves, or by a trusted hosting partner. Tool authors will have varying levels of control and access here.

All the tools featured above take data security into consideration, and key security measures can include:

[6] *Martus Sunset,* Benetech (May 15, 2018), available at https://benetech.org/martus-sunsets-human-rights-data-collection

- making regular, secure backups.
- making sure a systems administrator maintains and updates the servers and dependency software.
- offering hosted versions of the tool in different countries.
  building in deliberate redundancies in terms of where data is hosted.

Passwords and 2FA (Two-Factor Authentication)

Passwords are an important part of any secure system. Though mobile apps in general tend not to require a user to enter a separate password after they have unlocked the phone, some of the apps featured above require this as an extra layer of security.

For web-based databases, an extra layer of protection can be added through 2FA (Two-Factor Authentication), which helps keep data secure in case a documenter's password is compromised. Data management tool Uwazi, for example, recently added this functionality.[7]

(For more explanation on what 2FA is and how it works, Electronic Frontier Foundation **have a useful guide**[8]).

Protecting data collectors in the field

For documenters working on the ground, merely being seen to be documenting or looking for evidence of wrongdoing can put them in danger. Many face scenarios where they might be stopped by an authority or perpetrator (these might be the same) and have their phone searched.

This could lead to evidence being accessed and also potentially deleted, and could put documenters' physical safety, as well as the safety of others who appear in the documentation, at risk.

[7] *More Security, Collaboration and Efficiency: Uwazi Version 1.6*, HURIDOCS (Apr. 20, 2020), available at https://huridocs.org/2020/04/more-security-collaboration-and-efficiency-announcing-uwazi-version-1-6

[8] *How To: Enable Two-Factor Authentication*, Electronic Frontier Foundation (Oct 29, 2019), available at https://ssd.eff.org/en/module/how-enable-two-factor-authentication

Some of the secure collection apps above offer functionalities that specifically address this scenario. These include:

- The option to replace the app's icon and name on their phone with something more innocuous, such as a calculator icon. (e.g. Tella, eyeWitness)
- One-button instant app shutdown, and automatic removal of the app from the phone's "recently used apps" list. (e.g. Tella)
- Easy and quick deletion of the app, and media captured, from within the app itself. This can be useful if, for example, a documenter sees they are about to be stopped. (e.g. Tella, eyeWitness)

Protecting documenters' identities

Some of the apps above (e.g. Save) allow users to upload data to a shared folder or server pseudonymously, to protect their identity in case the data is compromised at the management or storage location.

A note on security audits

Security audits – which involve external experts checking the security of a tool's code on a regular basis – can add an extra layer of security. If the code of a tool is open source, organizations wishing to adopt the tool can also get an independent security audit done themselves. However, security audits, whether done by tool developers themselves or organizations working with an open source tool, can be complicated and/or expensive. Particularly for organizations with lower technical proficiency, additional support may be necessary to navigate this process.

Verification

"Our extensive research has found that metadata and a protected chain of custody are the keys to ensuring verifiable footage." – eyeWitness to Atrocities [9]

Verifiability of evidence is an ongoing challenge, particularly as photos and videos become increasingly easy to manipulate. Human rights documentation tools have addressed this challenge through a variety of verification strategies – in particular, through automatically adding metadata and through supporting chain of custody.

Adding metadata

Automatically adding significant metadata to collected data at time of capture is one strategy for enhancing its verifiability, especially for photos and videos. This metadata can include:

- information about the file - including a cryptographic hash, which can be used to determine if a file has been altered.
- the device the photo or video was captured on (manufacturer, hardware, device ID, screen size, and so on).
- the environment in which the photo or video was captured (GPS location, information about nearby cell towers, wifi networks, and bluetooth signals, date, time, and so on).

Mobile camera apps Tella and eyeWitness offer this functionality; ProofMode offers it as well, but in a different way, as ProofMode is not a camera app as such: instead, it runs in the background of a user's phone and adds metadata to photos taken using the phone's default camera app.

Extra metadata added by a user can also be useful. The Save app, for example, allows a user to manually add location information and other notes to footage backed up via the app.

[9] *Choosing a Secure Camera App to Document and Monitor Human Rights Abuses and Atrocities*, eyeWitness to Atrocities, available at https://www.eyewitness.global/Choosing-a-secure-camera-app

Chain of Custody

For evidence to be admissible in a legal context, chain of custody is key. Chain of custody is a legal concept that refers to *a sequential record of the individuals in custody or possession of the information sought to be admitted as evidence*.[10] This record takes the entire data cycle into account, from capture to eventual presentation in court.

Only one tool included in this snapshot, the eyeWitness app, has been designed as part of a system explicitly aimed at preserving chain of custody: photos and videos captured securely through the eyeWitness app are sent through secure connection to a server that eyeWitness maintains. The eyeWitness team organize and analyze the collected data themselves, regularly compiling reports for external investigators and legal mechanisms.

For documentation organizations, using a service like the one provided by eyeWitness comes with the advantage that their photos and videos are more likely to be useful as evidence in a legal environment. The limitation is that the organization does not keep full control over the media they capture, though eyeWitness will, as part of a partnership agreement, share copies of the images back with the partner organization if desired and seek consent from the partner organization before information is shared. The app also enables documenters to share with others the media they have captured, without the added metadata.

Some of the tool authors we spoke to noted, however, that chain of custody is as much – if not more – about policies and practices around how data is managed as it is about technology. One tool author pointed out that though technology can support chain of custody through security protocols and features such as audit logs (which can keep a running list of when something was accessed or modified and by whom), these are not enough on their own: “If someone receives, say, a pen drive – is there a procedure to follow? And if pressed, can the organization reliably demonstrate that the procedure was followed?”

For guidance on how to record chain of custody, Global Rights Compliance’s **BIS app** is a good place to start.[11]

[10] See *International Criminal Tribunal for the Former Yugoslavia, ICTY Manual on Developed Practices*, 28 (2009), available at https://www.icty.org/x/file/About/Reports%20and%20Publications/ICTY_Manual_on_Developed_Practices.pdf

[11] *Basic Investigative Standards for International Crimes Investigations, Global Rights Compliance, available at* https://www.globalrightscompliance.com/en/projects/basic-investigative-standards-for-international-crimes-investigations

Tool development

Challenges and strategies

This section is informed by interviews with tool authors, alongside published sources such as tool websites and blog posts. It is designed to offer insight into common challenges faced by tool developers working in the human rights documentation space, as well as to document strategies and learnings.

Challenge #1: Sustaining a tool over the long term

This section is informed by interviews with tool authors, alongside published sources such as tool websites and blog posts. It is designed to offer insight into common challenges faced by tool developers working in the human rights documentation space, as well as to document strategies and learnings.

What's on a tool author's sustainability to-do list

To talk about the challenges involved in successfully sustaining a tool over the long term – in other words, what's needed to make sure the tool continues to function well and to meet user needs – it's worth breaking down what this actually involves. In interviews, tool authors mentioned the following:

- Responding to bugs or other issues in a timely way.
- Responding to changes in tool dependencies and programming languages – in particular, security components and dependencies, which can require significant ongoing resources.
- Conducting general backend support, including server maintenance and app security.
- Providing user support, including responding to queries and developing requested features.
- Adapting to changing user norms and expectations around how tools should look, feel, and function.
- Conducting user testing, training, workshops and outreach.
- Doing regular audits or penetration testing to find security vulnerabilities.

Difficulties in getting funding to sustain tools

As summarized by one tool author we interviewed: “It takes a lot of commitment to support a tool and have a critical mass of users. And then, of course, there’s funding.”

A number of the tool authors we spoke to mentioned that they had experienced difficulty in getting repeat, sustained grant funding for tools (i.e. beyond the initial building of the tool), particularly when it comes to things like workshops and outreach. As one tool author said: “It feels like this is something that needs to be bolstered, but it’s hard to get money for.”

Conducting regular security audits also came up as a challenge. While open source licensing in theory allows others to check the security of a tool’s code, in practice, tool authors tend to need to put resources into regular security audits, whether their code is open source or not.

One tool author noted that they did penetration testing a few times per year, alternating testing companies where possible and re-testing after changes are made: “This is a priority and in the budget.”

Another, however, said that due to resource limitations most audits of their tool had been done not by themselves but by organizations wishing to adopt the tool, using standardized software that produced superficial and sometimes “haphazard” results. The tool author noted that they would love a good guide to security testing for small organizations with little funding, such as their own.

Alternative funding models

Fees for services

Though some tools considered in this research are entirely grant-funded, some rely on a mix of grant funding and fees for services, and some do not rely on grant funding at all. Services offered for a fee range from setup and hosting to customizing tools to fit specific needs and workflows, trainings and ongoing support, and even data analysis and legal support.

Some tool authors who work primarily on a fees-for-services model noted that being more demand-driven meant that they might not have as much capacity to work on features that have not specifically been asked for; however, they have also been able to feed additional features from customized versions back into the core tool. Funding from higher-resourced organizations has also made it possible for them to offer services at a nominal charge for lower-resourced organizations.

Funded hosting

Some apps are able to offer hosted versions of their tools via specially-funded servers: KoBoToolbox, for example, offers free-of-charge, unlimited

hosting to humanitarian organizations on a designated server provided by the United Nations Office for the Coordination of Humanitarian Affairs.

Membership-based funding

This model was floated by one tool author as a future plan, where membership funds would be used to finance maintenance fees, and anything left over would be used for new features, as decided on by the members themselves.

Open source development as a sustainability strategy

Most of the tools looked at in the human rights documentation space publish their code under open source licenses. One tool author mentioned both using and building open source technology as an important part of their sustainability strategy, in case they are unable to maintain the tool themselves in the future: “It’s important to build your tool using trusted, well-maintained, widely-used software so that other developers can make changes and improvements if funding ever runs out.

The risk of having only one or two developers who know the code is that it is very difficult to create a community of developers around the tool.”

Keeping up with changing user expectations

As the general technological environment has changed rapidly in recent years, so have people's general expectations in terms of how they want tools to look, feel, and function. Tool authors we spoke to mentioned that people have now come to expect "intuitive web-based interfaces and easy cross-device access," as well as intuitive functionalities, increased visualization and analysis capabilities and customer support.

Tools that have been around longer seem to face particular challenges in bridging the gap between the environment in which they were built and the environment in which they are currently operating – both technologically (for example, in terms of development languages used) and in terms of user expectations and norms.

Some tool authors mentioned underestimating just how much would be needed (in terms of resources and funding) to properly maintain their tools over the years.

Deciding when to call it quits

Some of the tool authors we consulted had made the difficult decision to sunset a tool (i.e. to no longer fix bugs or develop, update and provide support for the tool). For them, when the tool started to become too resource-heavy and less relevant to current user expectations, it made more sense to retire it than to continue.

Some tool authors found that it made more sense to start again with a new tool, or to put more resources into an existing tool in their portfolio that was more in line with the times.

Challenge #2: Getting the balance right

Tool design involves many decisions about how the tool should work, and this inevitably involves making trade-offs. In our research, a few key trade-offs came up as significant.

Security vs. usability

Security in the context of human rights documentation is complex and can be done in different ways, respond to different scenarios and address different types and levels of risk. In consulting with tool developers about how they approach security, many brought up the inevitable tension between providing – particularly with limited resources – both high security and high usability.

Many of the tool authors interviewed said that the particular model they landed up with was arrived at in stages, through learning from past challenges, experimenting with features, and listening carefully to user feedback.

Addressing security in data management, analysis and visualization tools

The security/usability trade-off was noted as a particularly sticky problem in tools that have data management, analysis and visualization as their primary functionality. As a number of tool authors pointed out, some high-security features, such as end-to-end encryption, can make accessing, managing and analyzing data very difficult, and can lead to irrecoverable data loss if encryption keys are lost.

Some noted that tools built with features that respond to highly complex threat models, but that impact negatively on usability, might not just struggle with adoption, but could also result in security being compromised in unintended ways. Said one tool author: “What we’ve seen is that when [security features] make working with the data too hard, people work around it. So you have this beautiful [i.e. extremely secure] system in theory, but people subvert it.”

Prioritizing usability over responding to highly complex threat models

As a result of this tension, and of learnings gained in the field in recent years, tools in the human rights space that have organization, analysis or visualization as their primary functionality lean toward workability and functionality over trying to respond to highly complex threat models.

Case study: Different ways to use Kobo ToolBox

KoBoToolbox offers an explicit example of what a security/usability tradeoff can look like in practice. KoBo allows users to collect data via forms and then work with this data (unencrypted) in KoBoToolbox. Users can, however – with some additional work and technical know-how – set up the tool so that completed forms are sent to the KoBo backend as encrypted files. KoBo warns, however, that "In this case, KoBoToolbox serves simply as a storage locker for your encrypted files […] [A]nything that requires access to the data, like the map view or data export, won't work within KoBoToolbox."

Case study: Uwazi's approach

Uwazi is a tool designed primarily for storing, organizing, analyzing and publishing collections of documents. Since Uwazi is primarily designed to help documenters work with the data they have collected, security relies not on end-to-end encryption but rather on passwords, two-factor authentication, an activity log, access permissions, publicly available security audit reports, and SSL protocols that protect the data in transit.

Addressing usability in security-focused tools

For some tools, providing documenters with certain security features is part of the tool's core mandate. (For example, secure camera apps Tella and Eyewitness.)

Many security features, however, rely on a willingness or capacity to use them, and here decisions must be made. One tool author consulted, for example, noted that arriving at the right balance of features took time and iteration: "We overengineered the first version, it was too 'James Bondey' and secretive."

Dedicated camera and video apps have a special challenge in that they are in 'competition' with the built-in camera apps that people are used to using on a regular basis. As one tool author noted, "Ordinary citizens don't have an incentive to have [a dedicated camera app] on their phone; but even then, the instinct wouldn't be there to pull it out."

Depth over breadth

A common approach by developers of apps that face significant "security vs usability" challenges has been to put effort into working directly with specific groups and organizations who are interested in using the tool or who fit the ideal use-case of the tool, rather than aiming for widespread individual uptake.

Tool authors noted that this approach has led to more successful uptake and use of these tools, though the number of users may be smaller.

Case study: CameraV → Proofmode

For camera app CameraV, usability challenges were cited as a primary factor behind the Guardian Project's decision to retire it in favor of "lighter reboot" ProofMode.

As they wrote in 2017, "While we are very proud of the work we did with [CameraV], the end result was a complex application and novel data format that required a great deal of investment by any user or community that wished to adopt it. With ProofMode, we both wanted to simplify the adoption of the tool, and make it nearly invisible to the end-user, while making the adoption of the tool by organizations painless through simple formats like CSV and known formats like PGP signatures."[12]

Flexibility vs. Structured Workflows

In general, tool authors noted that flexibility within apps was appreciated by users, and tended to also reduce the support burden on the tool author, as organizations could adapt the tool on their side rather than having to ask the tool author for every change.

Flexibility, with "sensible defaults"

As one tool user noted, however, too much flexibility and the user has no pathway or guidance through the app. "An ideal future – though not necessarily new – is where you have flexibility, but are presented with sensible defaults. Organizations working within a workflow like it because it moves you through it, and removes some of the complexity."

Defaults could include things like:

- pre-loadable templates and forms (for example, a form that covers the minimum information needed for a certain type of submission to a particular justice mechanism).
- default categories (for example, countries or types of violation) and data structures (for example, relating different types of data to each other). default interoperability with another app for a different part of the data

[12] *Combating "Fake News" with a Smartphone "Proof Mode*, Guardian Project (Feb. 24, 2017), available at https://guardianproject.info/2017/02/24/combating-fake-news-with-a-smartphone-proof-mode

collection and management workflow.
- options for data visualizations.

Within a tool that maintains some flexibility, these defaults could then ideally be changed, deleted or added to by the organization setting up the tool, to fit their own workflows.

Challenge #3: Knowing when to stop

In general, our research showed a growing shift in the human rights documentation tools community away from big "kitchen-sink" style apps[13] and towards, instead, an ecosystem of smaller apps aiming to respond to a more limited set of needs (but able to be used together). As one tool author shared: "Something we're considering is that people can use other apps."

Interoperability: Smaller Tools, Bigger Ecosystem

Martus (mentioned earlier) came up in interviews as an example of a "does everything" type of tool, including providing end-to-end encryption. A number of pros and cons to this approach came up in the research: While these kinds of big, feature-heavy tools can ensure, for example, a high level of security and/or a range of different functionalities, they can also become difficult to use for individual organizations who might not need all those features. Importantly, they can also become a huge burden on a tool developers' capacity to maintain them.

As one tool author said: "Combining tools allows a more tactical, agile approach, which more accurately meets the needs of changing contexts. There's less often a single point of failure that cripples the whole workflow."

[13] From the English phrase 'everything but the kitchen sink', i.e. in this case an app that attempts to do a large number of things. https://dictionary.cambridge.org/dictionary/english/everything-but-the-kitchen-sink

Case study: Deciding on features for Tella

Weighing up the pros and cons of each approach in the development of collection app Tella, tool authors Horizontal wrote in 2019: "While one obvious need was that of a comprehensive, secure data collection system that would accommodate the collection of data for criminal prosecution, Horizontal's current capacity is too limited to develop such a solution. We've instead decided to focus on a different aspect of the documentation process, one that was within reach for our small team: a mobile client for those individuals doing the documentation work on the ground, often in very difficult environments."[14]

Case study: Incorporating code vs forming an ecosystem

One tool author also talked about the costs and benefits they had encountered in trying to incorporate another tool's (open source) code into an expanded version of their own tool. Though they were able to customize the code, "Bugs took longer to figure out, and we were also unable to take advantage of improvements made to the original tool." In the end, the cost/benefit ratio worked out in favor of making the tools interoperable instead of merging the code into a customized tool.

[14] *Our Vision for Tella*, Horizontal (Sep. 5, 2019), available at https://wearehorizontal.org/2019/09/05/our-vision-for-tella

Inter-App Collaboration

ProofMode and Save

Save, launched by OpenArchive in partnership with The Guardian Project in late 2019, facilitates the secure backup of images and other media to an external location (such as ownCloud or Nextcloud, Dropbox, or the OpenArchive itself).

Save is also designed to work (on Android phones) with the tool ProofMode, which adds metadata to images and videos taken using a user's camera. This means that when used together, Save will make sure the added verification data is backed up to the server with the image or video.

Tella and KoBoToolbox

Tella is a mobile app that allows users to take images and videos and to build forms for standardized data collection in the field. But it doesn't have its own backend system (i.e., a location for data to be sent to and managed or stored). Instead, the app integrates with KoBoToolbox – users can either set up an instance themselves, or use one of KoBo's hosted options. This means that documentation collected by Tella can be sent directly, via a secure connection, to a KoBo database.

Uwazi and Digital Evidence Vault

Digital Evidence Vault allows users to preserve digital content directly from the browser; Uwazi allows users to store and organize data. An integration between the two, announced in late 2019 after piloting the integration with documentation organization GLAN Law[15], means that online content preserved using Digital Evidence Vault can be sent automatically to Uwazi, where a documenter can work with it further (for example, through adding tags and other information about it) and store it in their database.

[15] Laurel L. Finch, *How Global Legal Action Network is Documenting Digital Evidence of Airstrikes against Civilians in Yemen*, HURIDOCS (Nov. 12, 2019), available at https://huridocs.org/2019/11/glan-documents-airstrikes-in-yemen-with-uwazi-digital-evidence-vault-integration

Interdisciplinary Collaboration

Some tool authors also mentioned interest in collaborating not just with other tool authors but also with organizations that have expertise in related areas, such as building legal cases.

Expanding import and export capabilities

Many tools are designed to enable users to work with their data in other tools more generally – including commercial proprietary tools that they might already be using. KoBoToolbox, for example, has fairly limited analysis and visualization functionalities, but it allows data to be bulk-exported in a variety of formats, including Excel, CSV, KML, ZIP (for media) and SPSS. This allows users to import their data into spreadsheets and other analysis and visualization tools. It should be noted, however, that importing and exporting data adds an extra manual step to a workflow, which also has its own security implications.

Some tools, particularly those that facilitate data submission from the general public (as opposed to, say, a pre-established network of documenters using a shared, closed system) connect with a variety of input sources: Ushahidi, for example, can process submissions sent via SMS, email and Twitter.

Challenge #4: Providing ongoing user support

Almost all the tool authors consulted offered some level of user support. In most cases, they also had more established and intensive relationships with particular organizations who were using the tool, sometimes working closely with these organizations to adapt the tool to their specific needs and workflows.

Tool authors we spoke to also offered, in many cases:

- Tool set-up and hosting.
- Technical troubleshooting.
- Direct support and/or training. These were also cited by a number of tool authors as an important avenue for feedback, flagging problems, and improving the tool to better meet needs. Feedback might also come in on a more ad-hoc basis: one author said that users "might just text us from the field".

Other, more general avenues of support include:

- Online support pages and FAQs.
- Community support forums. One tool author mentioned that their community support forum, while providing a space for users to talk to each other and help each other, also allows ideas for new features to be discussed publicly long before they get to the design and implementation stage.

Resources are needed on both sides

In talking about supporting their users, tool authors noted that low capacity – whether tech capacity, time, funding resources, or low connectivity – is a persistent challenge when working with organizations that operate in low-resource contexts. These challenges can also impact tool authors' co-development efforts with documenter communities. As one tool author said, "They just don't have the time."

Tool authors also said that organizations tended to underestimate what is needed to set up and work with a tool, particularly when it comes to bigger projects like setting up a database to manage collected information. As one tool author explained, work needed can include developing and/or capturing the organization's methodology, determining their data structure and moving their information into this structure. Then, when the system is in place, time is needed to "vet, verify, clean, capture and manage" their information. "People expect databases to do the work themselves, but it requires documentation and intentional focus and work. There's a gap there; there are not a lot of resources that address that."

Conclusion

For this report, The Engine Room conducted one-on-one interviews with tool developers and carried out independent research into technology tools currently being developed and used by civil society for human rights documentation.

Within a fast-changing technological environment, and given the particular concerns that come with building tools in a human rights documentation context, tool developers identified a number of challenges in making sure that their tools continue to both work well over time and continue to meet the changing needs of those using them.

We found some of these challenges and changes reflected in the selection of tools we looked at: some established tools have been retired, and there are a number of newer tools being used in the space. Some of these are iterations of previous tools, taking learnings from these tools into account.

In meeting these challenges, tool developers are employing a diverse set of approaches. Our research also found efforts to collaborate and to enable tools to work together as smaller elements of a bigger ecosystem of available tools.

Moving forward, there are ample opportunities for funders, developers and tool users to work together to expand and enhance this ecosystem, and to collaborate on varied tools to match the varied contexts in which they're deployed.

Research Methodology

To arrive at a shortlist of tools to include in our snapshot, The Engine Room combined existing knowledge with desk research, including sources such as recent, relevant blog posts, articles and online discussions, tool websites and documentation, and GitHub repositories and issues. We also reviewed the tools themselves, and/or demos of the tools.

For the snapshot, we focused on tools that:

- are **currently being maintained**.
- are **currently being used in the field** (i.e. we did not include tools that were still in development).
- have been intentionally designed to **address the needs and contexts of civil society documenters**.
- **support at least one key aspect of a documentation workflow**, which for the purposes of this report we have identified as including data collection, verification, management, analysis, visualization and sharing.
  are designed to accommodate a **range of types of rights violations**.

In line with these criteria, our shortlist consisted of tools which were also:

- almost **all free and open source**, meaning that anyone can download the code and, with the right technological knowledge and resources, set up their own instance of the tool.
- **non-exploitative** in that their underlying business models do not make money through collecting data or locking organizations into ongoing, prohibitive charges.
- developed with **input and feedback from documenters themselves**.

Since the tool research was designed to focus on tools that are fairly broadly applicable within human rights documentation work, for the purposes of this report we did not include:

- **Bespoke databases created by organizations in-house** to fit their specific workflows and needs.
- **Tools designed for the documentation of very specific types of evidence and/or for use by specific groups or networks** – for example, the Anti-Torture Database,[16] designed to facilitate the documentation work of the International Rehabilitation Council for Torture Victims, or MediCapt,[17] designed by Physicians for Human Rights for the documentation of medical evidence related to sexual violence.

[16] International Rehabilitation Council for Torture Victims, *Global Torture Data*, available at https://irct.org/campaigns/global-torture-data

[17] Physicians for Human Rights, *PHR's Mobile App MediCapt puts Cutting Edge Technology in the Service of Preventing Sexual Violence*, available at https://phr.org/issues/sexual-violence/medicapt-innovation-2

Alongside this research, remote interviews were conducted with eight tool authors and background notes received from one more.

Interviews were aimed at filling in gaps in understanding around some of the tools we were looking at, as well as talking about learnings, challenges, current approaches and visions for the future. Tool authors were asked about development decisions made, challenges faced, and approaches adopted, as well as how they have approached issues such as security and verifiability.