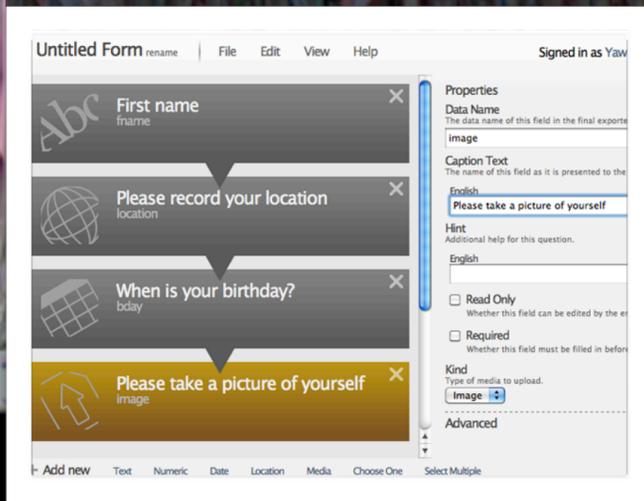
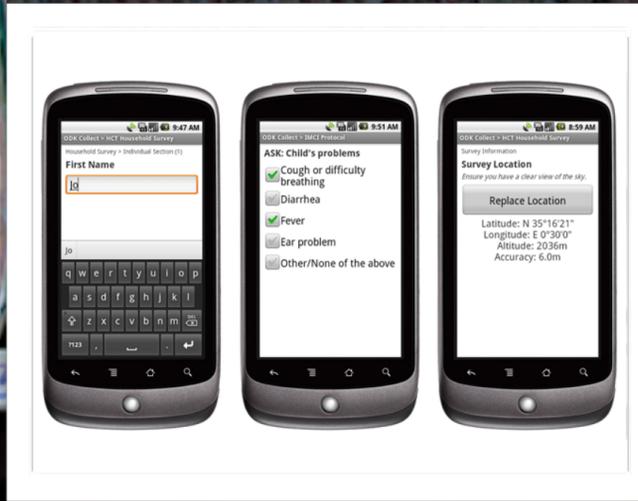


Using paper to collect data is difficult and inefficient.  
ODK uses phones and servers to digitize data collection.

## 1. Build form



## 2. Collect data



## 3. Aggregate results



<http://opendatakit.org>

Open Data Kit (ODK) is a free and open-source set of tools which help organizations create mobile data collection solutions.

ODK provides an out-of-the-box solution for users to:

1. Build a data collection form or survey;
2. Collect the data on a mobile device and send it to a server; and
3. Aggregate the collected data on a server and extract it in useful formats.

# Open Data Kit

Building Information Services for Low Income Regions

Yaw Anokwa, Carl Hartung, Waylon Brunette,  
Adam Lerer, Clint Tseng, Mitch Sundt, Gaetano Borriello

<http://opendatakit.org>

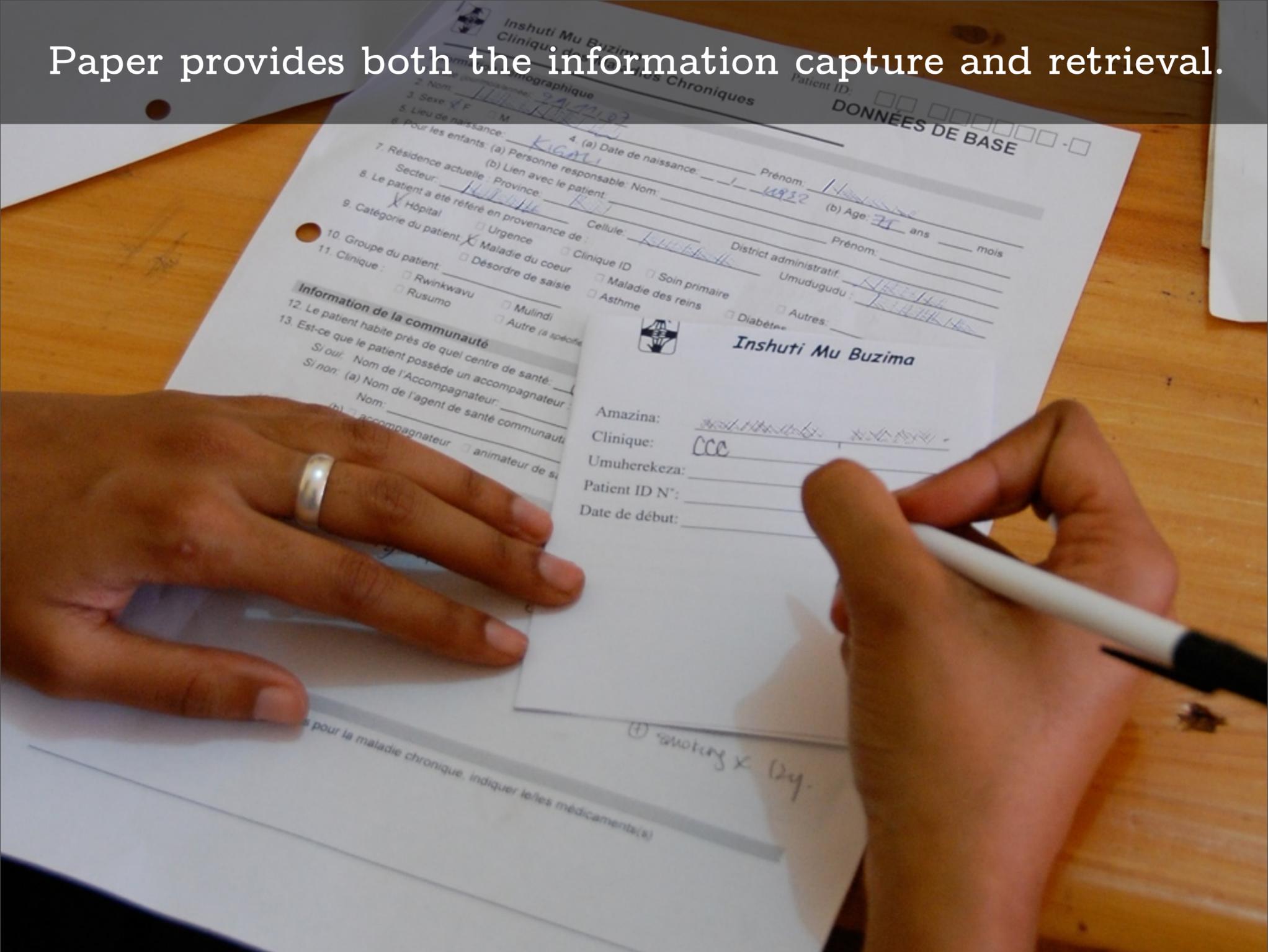


My name is Yaw Anokwa. I'm a Ph.D student in Computer Science from the University of Washington and this talk is about Open Data Kit our free and open source set of tools for building information services in low income regions.

This is work that is done with Carl Hartung, Waylon Brunette, Adam Lerer, Clint Tseng, and our advisor Gaetano Borriello. All of us are from UW except for Adam, who is from MIT.

Our team mostly works on healthcare in sub-Saharan Africa, so maybe I can share the kinds of scenarios that inspired ODK.

Paper provides both the information capture and retrieval.



When a patient comes into a clinic, the doctor fills out a paper form about the visit.

The paper form goes into a folder, and when the patient comes back a few months later, the doctor reviews the information in that folder and uses that to make decisions.

\*

So paper provides both the information capture and retrieval.

With a paper-based system like this becomes really easy to make clinically relevant mistakes like missing a trend in a record or misread someone's handwriting.

Even if you are really careful, when you get thousand of patients,

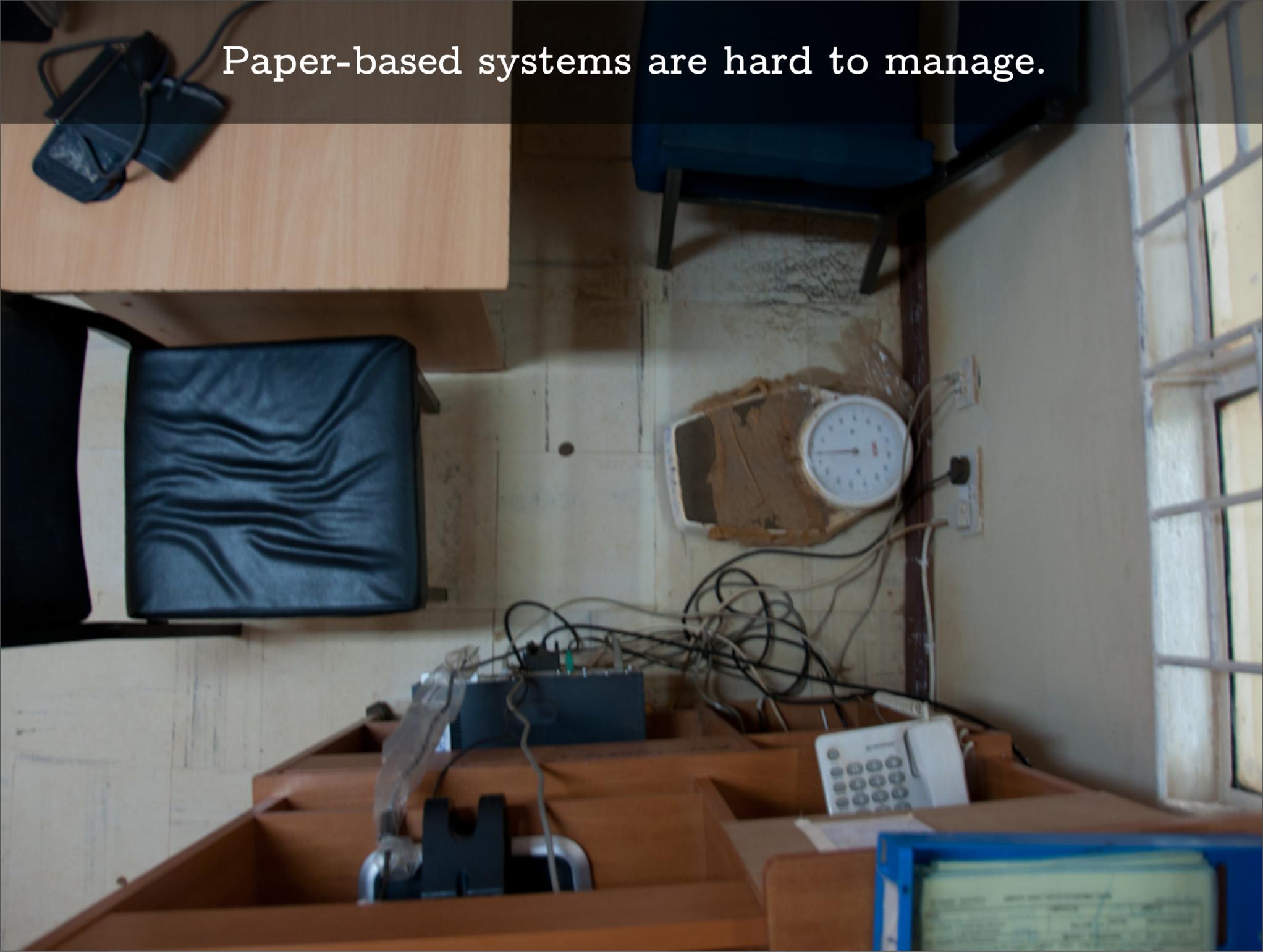
Paper-based systems are hard to search or transport.



The system becomes hard to search or transport.

This is especially important if you are treating HIV/TB patients who generate a lot of paper records. Ideally, you want to move to an electronic system like this.

Paper-based systems are hard to manage.



This process of generating the summaries and getting them to the remote sites is very failure prone.

Take this picture as an example. You see where the problem is? If the patient sits at the left, and has some children with her, those kids are going to yank on those cables of the printing computer. And no one will report it because of the power dynamics.

Without a more transparent monitoring system, it is difficult to correct these failures.

Lag between data collection and actionable information.

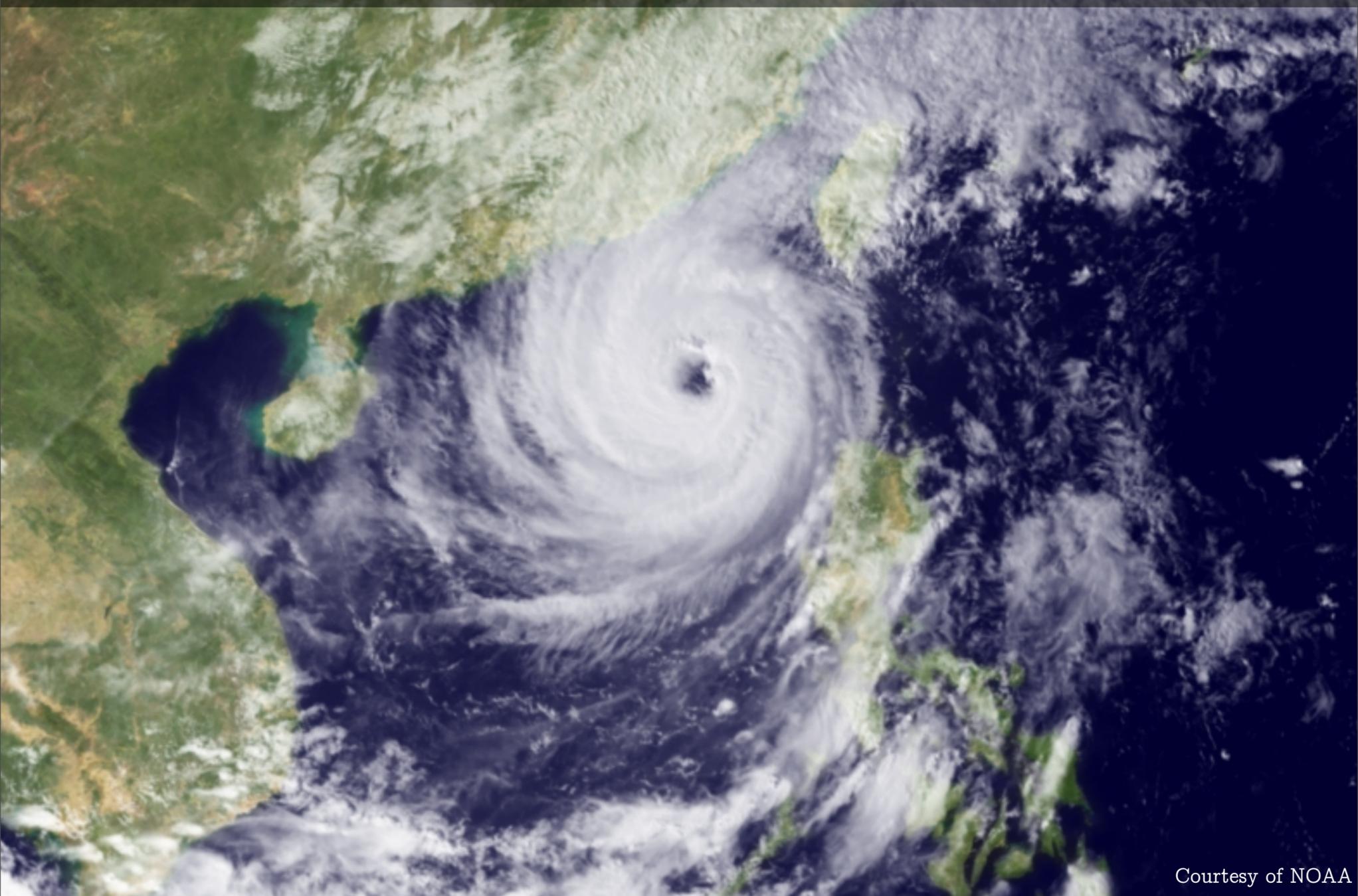


There is a substantial lag between data collection and actionable information.

That is, if you are in remote desert, it's going to take you some time to get all those paper forms back to a city so you can get it double entered.

There are also more complex examples of where paper just doesn't work.

Data collection could provide much richer information.



Courtesy of NOAA

If there is a hurricane, and infrastructure has been destroyed,

\*

You'd like your data collection to provide much richer information.

For example, you might want to task people to go collect images and GPS locations of damaged buildings.

Basically our claim is

**Mobile phones and cloud servers can increase  
the scale and speed of interventions.**

Mobile phones and cloud servers can increase the scale and speed of interventions.

And with the growth of cell phone usage all over the world, there have been lots of projects where people use phones and computers to replace paper.

We looked at a lot of those tools before building ODK, so let me give you three high level reasons some of those tools are lacking.

## Important features are lost when using basic technology.



Staying with the medical domain, here is a picture of a nurse using a PDA to do some decision support.

A lot of projects use PDAs and basic phones. Much of this is cost driven because everyone wants the cheapest device possible...

But

There is a tradeoff between the cost of device and functionality, usability and sustainability of your project

\*

and we argue that important features are lost when using basic technology.

For example, on basic phones, you can't easily take high resolution picture or store a complex medical record system.

Training is also sometimes an issue. We worked with a group who was spending hours just training people on how to capture a location on an external GPS and enter it into a PDA. All that training can be expensive.

We should be able to make that process much easier!

Domain-specific tools are inflexible and keep data siloed.



flickr: RaGardner4

The second issue is many of the existing systems are

\*  
domain-specific and keep the data siloed.

Why can't we build a modular system that works well for healthcare and crisis mapping and surveying?

Why can't we make small pieces so each organization can pick just what they need to connect to their existing systems?

Sustainability is hard to achieve with a small team.



And finally, there is the issue of sustainability and scale.

\*

Growing a project is hard with small academic team.

Instead of a few people working on ODK, we wanted to build a community of developers and implementers and users and enable that community to grow the project as needed.

This way when we finished the research, the project would continue.

So to recap some of the problems are...

**Important features are lost when using the most basic technology.**

**Domain-specific tools are inflexible and keep data siloed.**

**Sustainability and scale are hard to achieve with a small team.**

Important features are lost when using the most basic technology.

\*

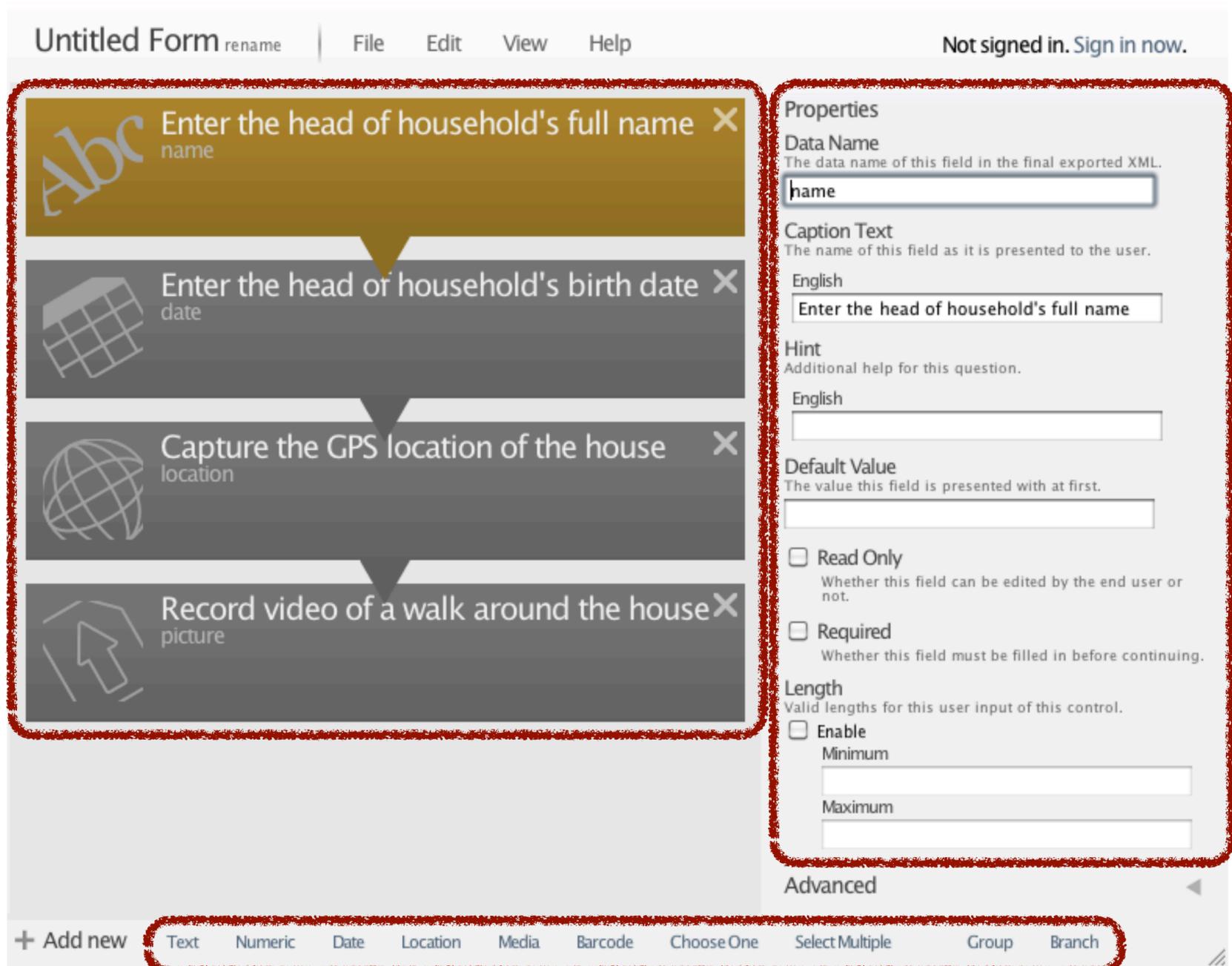
Domain-specific tools are inflexible and keep data siloed.

\*

Sustainability and scale are hard to achieve with a small team.

So with that, let's take a look at three of the popular ODK tools.

## Build: Drag and drop prompts for form creation.



ODK Build is an HTML5 web application where you drag and drop prompts to create forms. It runs in the browser but can also be used offline. It looks like this.

\*

To design a form, you drag and drop each prompt the user will interact with from this button pane

\*

to the canvas. Each prompt has a set of properties which users can edit here.

\*

The prompts are pretty powerful. For example, you can have multiple languages for each prompt, make prompts relevant based on previous answers, or make prompts loop some number of times.

When you are done, the Build gives you a file that describes the form logic and the schema of your data.

XForms describes the form logic and data schema.

```
<instance>
  <data>
    <name/>
    <date/>
    <location/>
    <picture/>
  </data>
</instance>
<itext>
  <translation lang="eng">
    <text id="/data/name:label">
      <value>Enter the head of household's full name</value>
    </text>
    <text id="/data/date:label">
      <value>Enter the head of household's birth date</value>
    </text>
    <text id="/data/location:label">
      <value>Capture the GPS location of the house</value>
    </text>
    <text id="/data/picture:label">
      <value>Record video of a walk around the house</value>
    </text>
  </translation>
</itext>
  <bind nodeset="/data/name" type="string"/>
  <bind nodeset="/data/date" type="date"/>
  <bind nodeset="/data/location" type="geopoint"/>
  <bind nodeset="/data/picture" type="binary"/>
</model>
</h:head>
<h:body>
  <input ref="name">
    <label ref="jr:itext('/data/name:label')"/>

```

That file is an XForm. W3C XForms are a well-defined open standard for forms and we use it to describe the form logic and data schema.

\*

If you can read some of this text, you'll see some of the elements of the form I showed you earlier, like the english translation for the text.

\*

You see also data types -- string, date, geopoint and binary.

\*

Once you have a form, you can give it to a device you want to show it -- like our mobile client, ODK Collect.

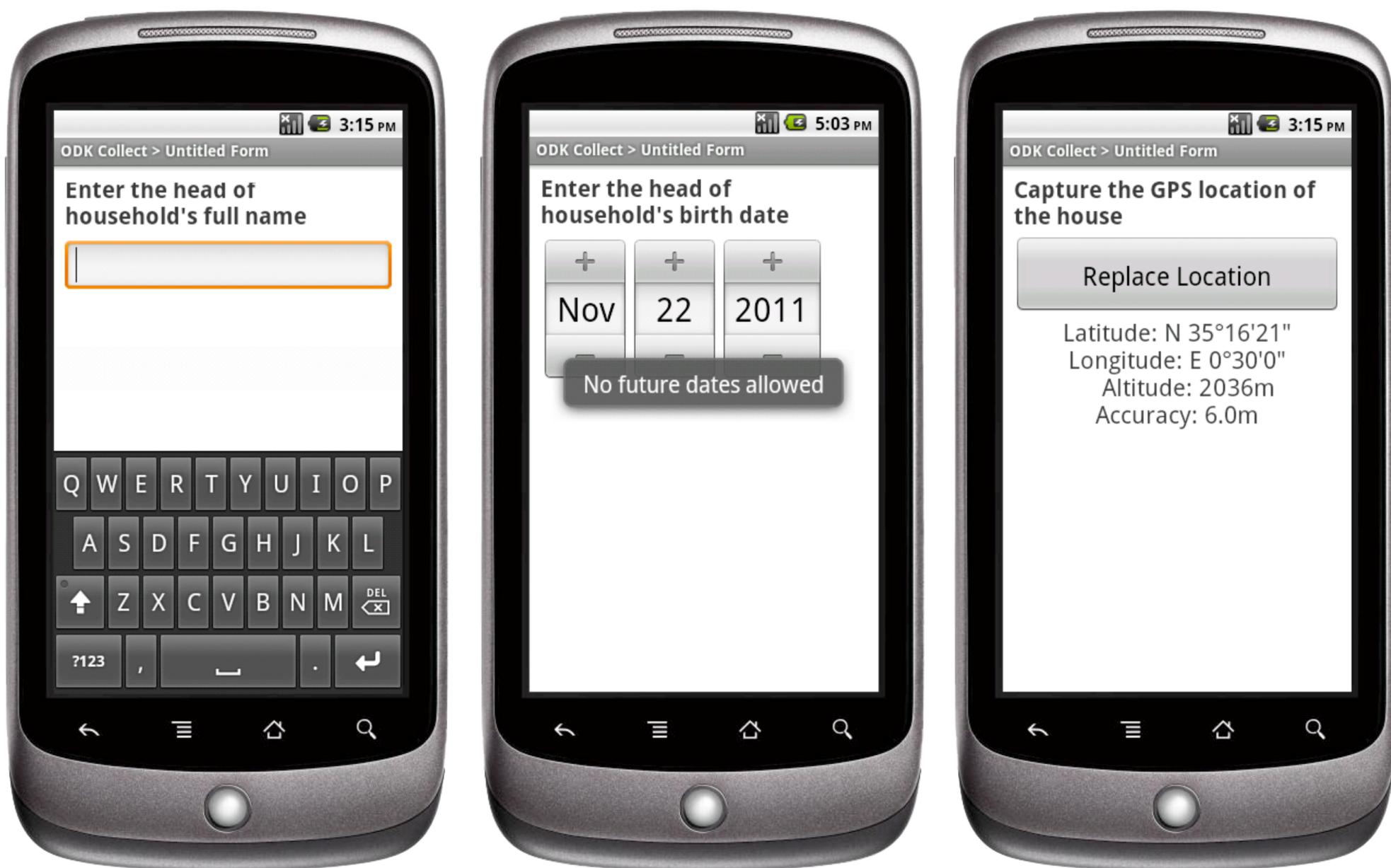
# XLS2XForm: Designs forms using Excel.

get start time	start		
get end time	end		
get today	today		
get device id	deviceid		
get subscriber id	subscriberid		
get sim id	simid		
get phone number	phonenumber		
add text prompt	string	string widget	can be very short or very long
add integer prompt	int	integer widget	try entering a number > 10
add decimal prompt	decimal	decimal widget	only numbers > 10.51 and < 18.39
add date prompt	date	date widget	only future dates allowed
begin group	multi_question_page	multiple questions on a page	
add text prompt	example_text_questio	example text question	
add integer prompt	example_integer_que	example integer question	
add decimal prompt	example_decimal_que	example question	
end group			
add select multiple prompt using alphaOptions	select	select multiple widget	don't pick c and d together
add select one prompt using numOptions	select1	select one widget	scroll down to see default selection
add select one prompt using alphaOptions	quick_advance	quick advance select widget	click a choice to select it and advance to the next quest
add select one prompt using numOptions	spinner	spinner widget	click the button to provide a response
add select multiple prompt using imageOptions	grid	grid widget	click an image to select it (you must have the images on
add select one prompt using imageOptions	grid_two_columns	grid with a maximum of two columns	regardless of screen size this widget will only show two
begin group	choice_table	choice table	
add select one prompt using yesno	choice_row_1	question 1	
add select one prompt using yesno	choice_row_2	question 2	
add select one prompt using yesno	choice_row_3	question 3	
end group			
begin group	choice_table_label	choice table with label	
add select one prompt using yesno	label	table of answers	
add select one prompt using yesno	choice_row_1	question 1	
add select one prompt using yesno	choice_row_2	question 2	
add select one prompt using yesno	choice_row_3	question 3	
end group			
add select one prompt using states	autocomplete	autocomplete widget	start typing a state name. any state name that contains
add select one prompt using states	autocomplete_chars	autocomplete widget with chars option	start typing a state name. any state name that contains
add select one prompt using yesno	branch	branch widget	yes to skip languages, validator, sections
begin group	section_a		
begin group	section_b	section b	
begin group	section_c	section c	
add text prompt	language		
end group			
end group			
end group			
add text prompt	regex	validate widget	email validator
add location prompt	geopoint	geopoint widget	this will get gps location
add barcode prompt	barcode	barcode widget	scans multi-format 1d/2d barcodes

XForms describes the form logic and data schema.

```
<group appearance="field-list" ref="/sample/multi_question_page">
  <label>multiple questions on a page</label>
  <input ref="/sample/multi_question_page/example_text_question">
    <label>example text question</label>
  </input>
  <input ref="/sample/multi_question_page/example_integer_question">
    <label>example integer question</label>
  </input>
  <input ref="/sample/multi_question_page/example_decimal_questoin">
    <label>example question</label>
  </input>
</group>
<select ref="/sample/select">
  <label>select multiple widget</label>
  <hint>don't pick c and d together</hint>
  <item>
    <label>option a</label>
    <value>a</value>
  </item>
  <item>
    <label>option b</label>
    <value>b</value>
  </item>
  <item>
    <label>option c</label>
    <value>c</value>
  </item>
  <item>
    <label>option d</label>
    <value>d</value>
  </item>
</select>
<select1 ref="/sample/select1">
  <label>select one widget</label>
```

# Collect: Display prompts for data collection and delivery.



ODK Collect is an client that takes the XForm and can display prompts to collect and deliver different types of data.

Collect runs on the Android operating system so you can use it on phones, tablets and netbooks. It runs great offline and looks like this.

\*

You can collect text, numbers

\*

dates

\*

you can put constraints on all prompts. So for example, birth dates can't be in the future.

\*

You can also capture GPS location using one click.

## Collect: Display prompts for data collection and delivery.



The prompts can be multimedia rich. In this example, you can both see the bird and play the sound of each bird before continuing.

\*

The prompts can be translated into any language (say Swahili). You can switch them on the fly (to say English).

The prompts can have logic and are powerful enough to do clinical decision support. This is what we mean by delivering information.

This form is from the WHO's diagnosis protocol for sick children under five. As you tell the phone the symptoms, it can suggest treatments.

You can combine this logic with the multimedia. So if say the child is coughing or has a fever

\*

ODK Collect can record the sound of a cough or even play a video about how to treat fever.

# Collect: Display prompts for data collection and delivery.

ODK Collect > Hypertension Screening

Clinical Elements

**Systolic BP (mmHg)**  
*Measure the right arm while seated*

1	2 ABC	3 DEF	-
4 GHI	5 JKL	6 MNO	.
7 PQRS	8 TUV	9 WXYZ	✕
* # (	0 +	␣	↩

ODK Collect > Hypertension Screening

Clinical Elements

**Diastolic BP (mmHg)**  
*Measure the right arm while seated*

1	2 ABC	3 DEF	-
4 GHI	5 JKL	6 MNO	.
7 PQRS	8 TUV	9 WXYZ	✕
* # (	0 +	␣	↩

ODK Collect > Hypertension Screening

Referral Information

**Blood pressure extremely high. Patient needs immediate medical attention.**

ODK Collect > Hypertension Screening

Referral Information

**Appointment Date**

+	+	+
Nov	04	2011
-	-	-

# Collect: Display prompts for data collection and delivery.

ODK Collect > BRAC Videos

Choose the videos you want to watch to build a playlist.

Introduction to BRAC



TB/HIV + Summary



TB



ODK Collect > BRAC Videos

Press the Play button to watch "Introduction to BRAC (~1 min)". When you are done, please rate it below.



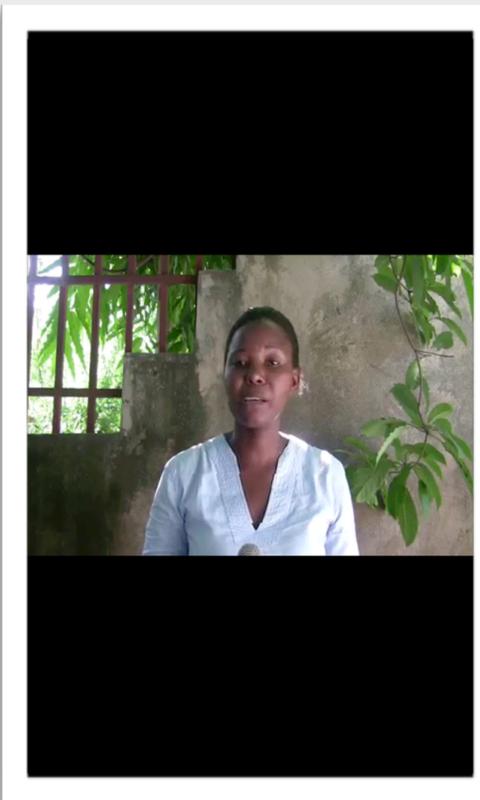
1 - Hated It

2 - Didn't Like It

3 - Liked It

4 - Really Liked It

5 - Loved It



ODK Collect > BRAC Videos

Press the Play button to watch "Introduction to BRAC (~1 min)". When you are done, please rate it below.



1 - Hated It

2 - Didn't Like It

3 - Liked It

4 - Really Liked It

5 - Loved It

# Aggregate: Host data and provide extraction interfaces.

Location-Latitude	Location-Longitude	Location-Altitude	Location-Accuracy	Description
374240020636755	151.1692304632708	28.0	5.0	End other side
37486958941743	151.1690486118154	29.0	5.0	Willow
37504302300135	151.16903599729017	29.0	5.0	Cigarette dump
375864156698455	151.16889189846407	28.0	5.0	White flowers
37685209640148	151.1684238513032	28.0	10.0	Trees
37923633851942	151.16784138112294	40.0	5.0	Start white creek
9163298333333	-105.22232958333333	1633.1	12.0	Eddy at dog park
26413130760193	-80.32614648342133	1.0	4.0	Baseball field in trop
37890696525574	151.16791605949402	28.0	6.0	Whites creek algae
4937171666667	-121.93441071666666	27.4	3.0	Bob at Santa Clara U
378719210624695	151.1679643392563	21.0	3.0	White creek running
37857973575592	151.16811990737915	25.0	3.0	Rubbish
37859582901001	151.16813600063324	28.0	3.0	Blue tongue lizard
4937171666667	-121.93441071666666	27.4	3.0	Bob at Santa Clara U
9163298333333	-105.22232958333333	1633.1	12.0	Eddy at dog park
26413130760193	-80.32614648342133	1.0	4.0	Baseball field in trop

Aggregate hosts the submitted data and provides extraction interfaces such as

\*

spreadsheets, maps, and queries. We don't run one big server, you download an installer, and it configures one just for your organization.

Aggregate: Codebase runs locally and in the cloud.



flickr: theplanetdotcom

Aggregate is designed to run on local machines as well as run in the cloud.

In fact, the same code base that runs on a local server backed with MySQL will run on Google App Engine or Amazon Web Services.

The benefit of the cloud over the local server is with a double click and you get a fast server that is basically free.

You don't have to worry about viruses, maintenance, scaling, or the other problems of running a server in a low-income region.

# Aggregate: Designs database backend using XForm.

The screenshot shows the MySQL command-line interface. The title bar indicates the connection is to 'openmrs/mysql/event'. The left sidebar shows a tree view of databases, with 'event' selected. The main window displays the 'Table Info' tab for the 'event' table. Below the table structure, the 'INDEXES' section shows two primary keys. The 'TABLE INFORMATION' section at the bottom left provides summary statistics for the table.

Field	Type	Length	Unsigned	Zerofill	Binary	Allow Null	Key	Default	Extra
name	char	64	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	PRI		None
body	longblob		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		NULL	None
definer	char	77	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			None
execute_at	datetime		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		NULL	None
interval_value	int	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		NULL	None
interval_field	enum	'YEAR','QUAR...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		NULL	None
created	timestamp		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		CURRENT_T...	on update ...
modified	timestamp		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		0000-00-0...	None
last_executed	datetime		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		NULL	None
starts	datetime		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		NULL	None
ends	datetime		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		NULL	None
status	enum	'ENABLED','DI...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		ENABLED	None
on_completion	enum	'DROP','PRES...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		DROP	None
sql_mode	set	'REAL_AS_FL...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			None
comment	char	64	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			None
originator	int	10	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		NULL	None
time_zone	char	64	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		SYSTEM	None

Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Comment
0	PRIMARY	1	db	A	NULL	NULL	NULL	
0	PRIMARY	2	name	A	0	NULL	NULL	

TABLE INFORMATION

- created: 8/18/10
- updated: 8/18/10
- rows: 0
- size: 0 B
- encoding: utf8

Another feature that makes Aggregate unique is that it designs the database backend for you using the XForm.

\*

You don't have to know how to use complex relational tools databases like these.

You give Aggregate the same form you give to Collect, and it builds a custom database for you.

Aggregate: Stores or forwards data to external systems.



DeviceId	351676030226627
SurveyorName	Shadrack
TreeLocation-Latitude	-4.9192410707473755
TreeLocation-Longitude	29.60762321949005
TreeDBH	57.0
TreeName	Myombo
TreePicture	<a href="#">View</a>

Finally, Aggregate can either store or forward your data to other systems.

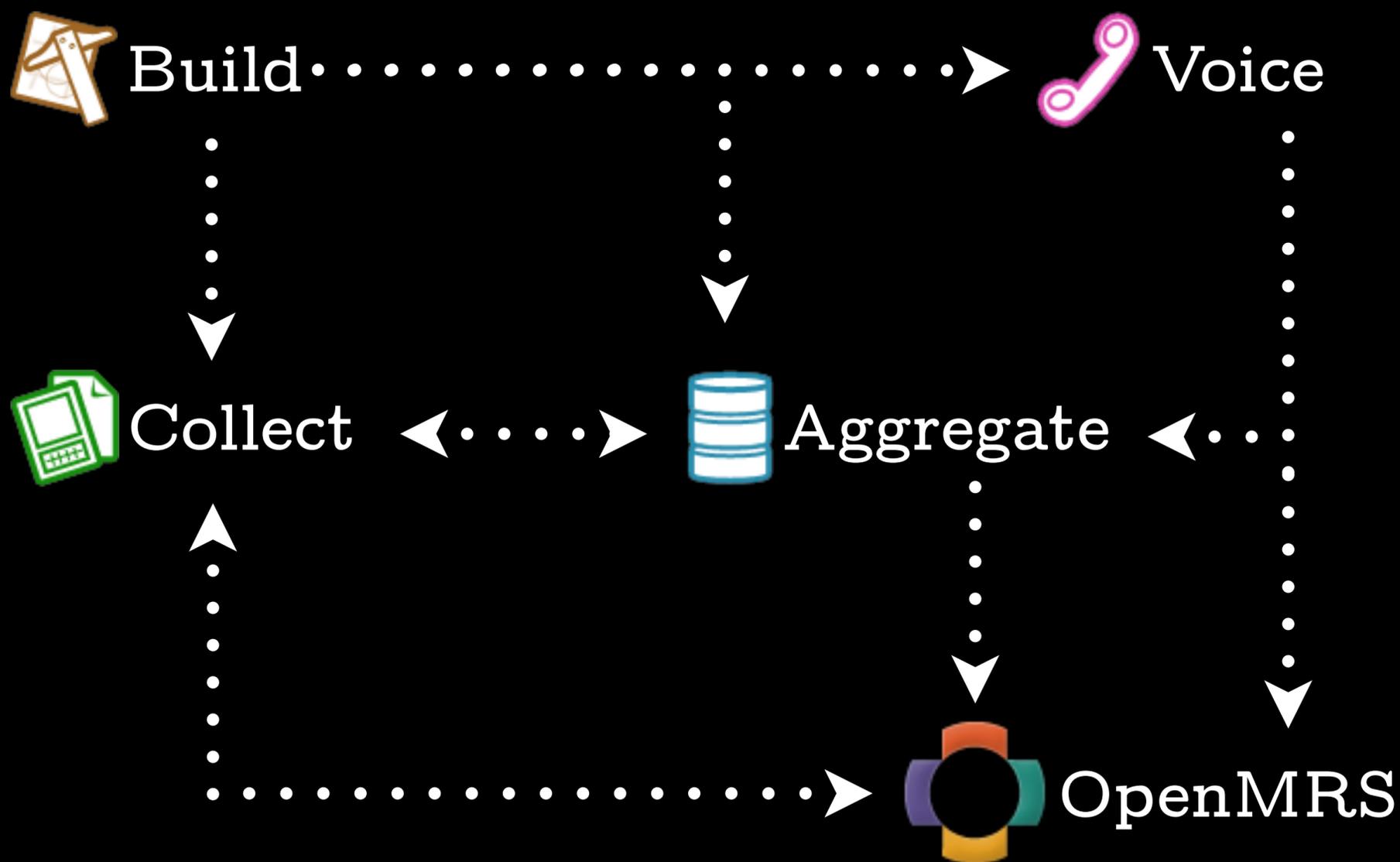
In this example, forestry workers with the Jane Goodall Institute in Tanzania, submitted data from Collect to Aggregate and then exported to Google Earth.

Managers could then click on each yellow point and get the data that was submitted.

The process of multilingual form design, training people who hadn't seen smartphones, data collection and reporting took half a day.

So I've showed you three of the ODK tools, let me show how they fit together.

# ODK tools are designed to fit together.



You use Build to design a form

\*

and you give that form to Collect to display it.

\*

You give it to Aggregate, and it will build the storage for the responses.

\*

To complete the loop, Collect can send data to Aggregate, and it can also download new forms from the Aggregate.

But ODK is more than those three tools.

\*

For example, you can give the form to ODK Voice.

Voice is a tool we have that builds an IVR tree out of that form and connects it to the phone network. It does all the things Collect does with logic and multiple languages, but it does it using the voice channel on basic phones.

For example, you can have Voice call all the people in a district and ask them for their gender and language.

Based on their responses with their keypad, you can play the women pregnancy advice and the men farming advice. And it will do that in their native language.

\*

Naturally, all their responses get fed back to Aggregate.

\*

We support connecting to other systems, say like OpenMRS.

\*

You can treat OpenMRS just like Aggregate, and use it to share medical forms and data with Collect.

Users can design an end to end system in under an hour.



You might be asking how long it takes to build the basic end to end system.

\*

If you are an implementer with basic computer skills you can be up and running with the form, the mobile client and the server in about hour.

More importantly, we are compatible with a bunch of other tools that may work better for organization.

If you have really large forms, you might want to use ChildCount's XLS2XForms and that will work just fine.

\*

If you think Android phones are too fancy, you can use a J2ME client like JavaRosa. You loose the Android specific features, but everything will work.

\*

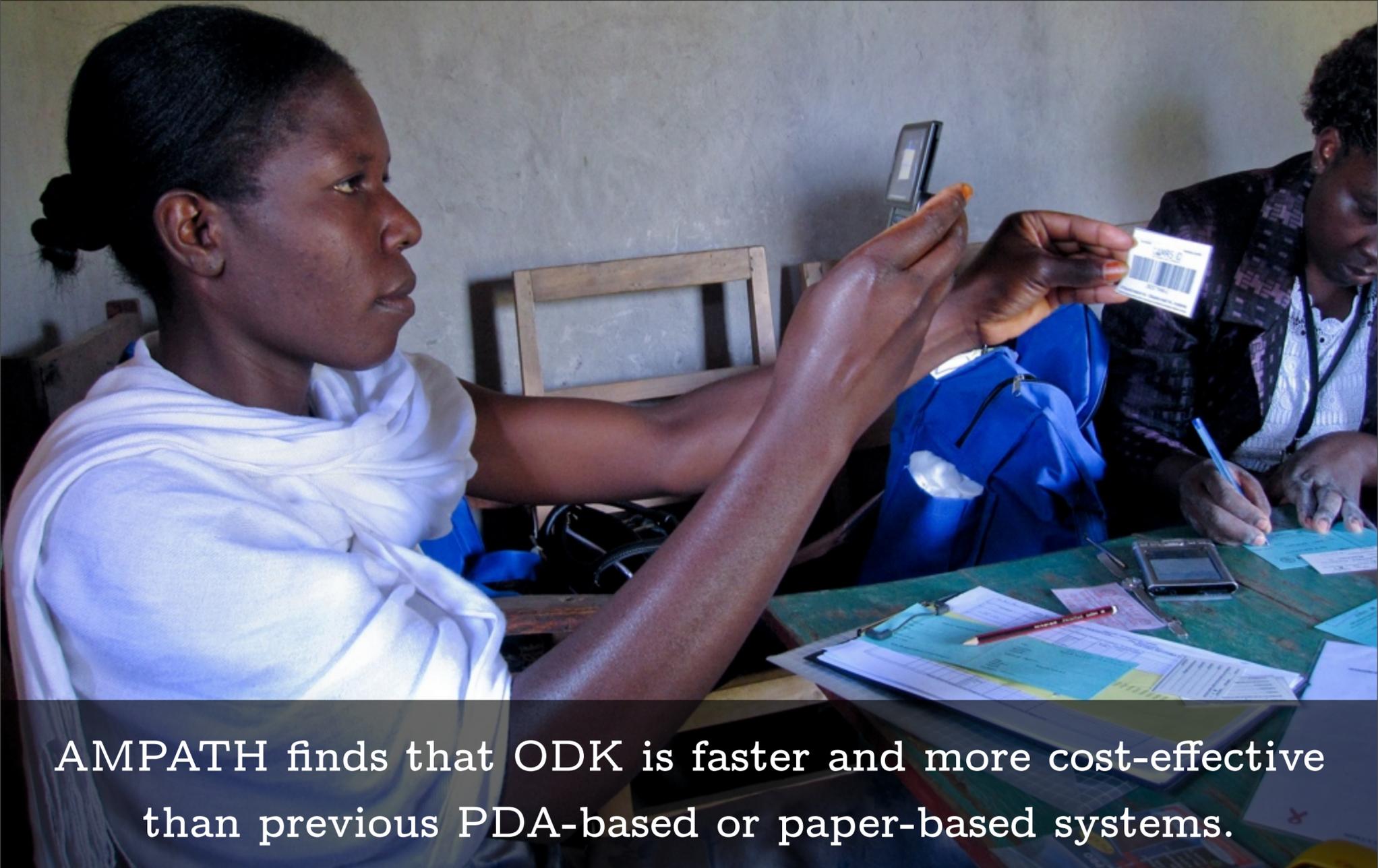
If you don't like our Java server, Dimagi makes an awesome Python one called DataHQ -- you can use that as well.

Now these are nice ideas, but like Ken said in the panel, does any of this actually work in the field?

We surveyed a few groups who were familiar with existing systems and had used ODK.

The details of the survey are in the paper, but I want to highlight what four of those groups said.

AMPATH's health workers have used ODK to counsel and test over half a million people for HIV.



AMPATH finds that ODK is faster and more cost-effective than previous PDA-based or paper-based systems.

AMPATH uses 300 community health workers equipped with ODK-powered smartphones to do home-based counseling and testing.

This image shows one of those workers using the barcode scanner on the phone to identify a patient.

The project started in early 2010 and as of mid-2011 has visited 568k individuals and tested 355k.

Users found the system easy to use and thought it allowed them to collect higher quality data.

Supervisors saw higher cost savings over our previous PDA-based system as well as over pen and paper systems.

Electronic collection also facilitated earlier reporting, which means more people get on treatment much faster.

<http://ampathkenya.com>

Carbon For Water collected over 1,000,000 forms with 4,000 ODK-powered phones in 6 weeks.



Carbon For Water finds that ODK enables real-time collection of image, GPS and survey data at scale.

Carbon For Water used 4000 ODK-enabled phones to collect over a million forms in six weeks.

The data is used to support Vestergaard Frandsen's campaign to distribute LifeStraw Family water treatment units to 4 million people in Western Kenya.

<http://www.vestergaard-frandsen.com/carbon-for-water/index.html>

Deployments: 10-50k active users of ODK tools globally.



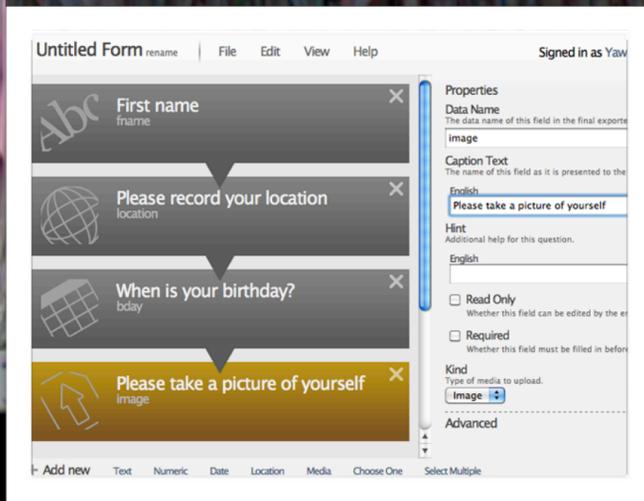
We scale horizontally. We build a platform and enable people to use it.

There are 5,000–10,000 users of stock ODK tools and 10,000–50,000 users of ODK derivatives. ODK is used at scale on in 30 countries (six continents with Antarctica coming soon) and used in a variety of domains.

- \* Jane Goodall Institute, Amazon Conservation Team and Google is using ODK for deforestation monitoring.
- \* New York City's Department of Health uses ODK to gather emergency preparedness and response-related data in the field.
- \* The Africa Soil Information Service (AfSIS) uses ODK to map soil conditions.
- \* Johns Hopkins uses ODK to improve provider communication and education through their eMOCHA project.
- \* Makerere University's uses ODK for automated diagnosis and mapping cassava mosaic disease.
- \* D-Tree is using ODK in a UNICEF-funded project to better to identify and treat severely malnourished children.
- \* ODK has been used for as an Early Grade Reading Assessment instrument to measure student literacy.
- \* Berkeley Human Rights Center uses ODK to documenting human rights violations.
- \* The Millennium Villages Project has collected over 3–5 million data points on 30,000

Using paper to collect data is difficult and inefficient.  
ODK uses phones and servers to digitize data collection.

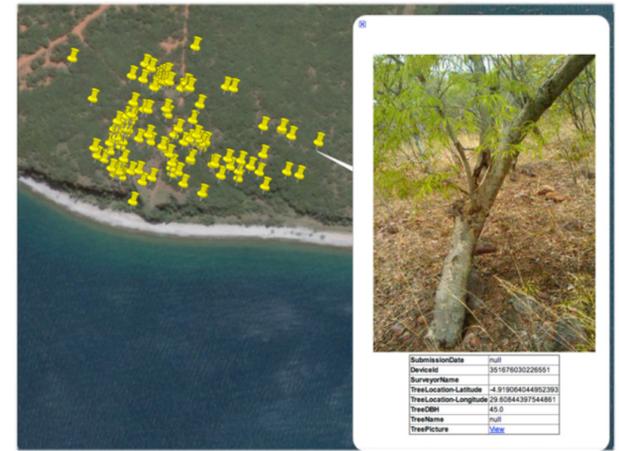
## 1. Build form



## 2. Collect data



## 3. Aggregate results



<http://opendatakit.org>

Open Data Kit (ODK) is a free and open-source set of tools which help organizations create mobile data collection solutions.

ODK provides an out-of-the-box solution for users to:

1. Build a data collection form or survey;
2. Collect the data on a mobile device and send it to a server; and
3. Aggregate the collected data on a server and extract it in useful formats.