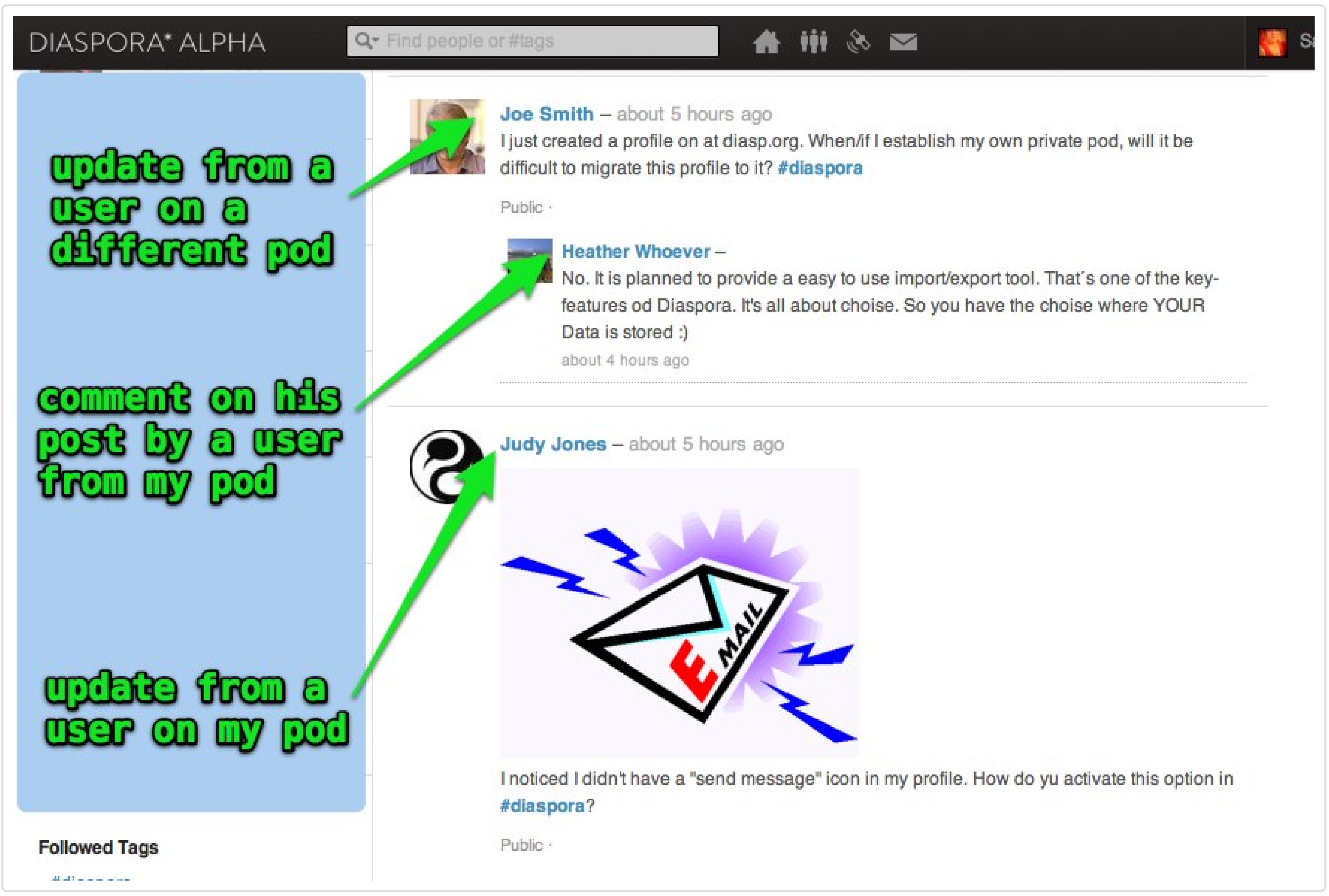
How Diaspora Connects Users

Note: this is the first in a series of technical posts about Diaspora's software architecture and code, and is a slightly modified version of the original on the Diaspora blog. If you have topics you'd like to see covered in future installments, please let me know.

A single installation of the Diaspora software is called a **pod**. The Diaspora distributed network is made up of hundreds of these pods, each with a set of users – sometimes just one on an individual pod, sometimes tens of thousands on a community pod. Each pod is run by a different person or organization. But no matter what pod you sign up on, you can connect with users on any other pod.

When you have friends on different pods, your stream seamlessly mixes updates from remote friends with updates from friends on your pod. In this way Diaspora is a **distributed** social network that resembles, from the user's perspective, a **centralized** social network.


DIASPORA* ALPHA
Find people or #tags
update from a user on a different pod
comment on his post by a user from my pod
update from a user on my pod
Followed Tags
EMAIL

Joe Smith – about 5 hours ago
I just created a profile on at diasp.org. When/if I establish my own private pod, will it be difficult to migrate this profile to it? #diaspora
Public ·

Heather Whoever –
No. It is planned to provide a easy to use import/export tool. That´s one of the key-features od Diaspora. It's all about choise. So you have the choise where YOUR Data is stored :)
about 4 hours ago

Judy Jones – about 5 hours ago
I noticed I didn't have a "send message" icon in my profile. How do yu activate this option in #diaspora?
Public ·

But how do these users find each other? In a centralized system, all servers access the same database, so when you search for a friend, there's only one place to look. But in the Diaspora ecosystem, each pod has its own database, inaccessible to the other pods. So how does pod A figure out who's on pod B, or for that matter, pod C that's never been heard from before?

It all starts with searching. Let's say you're setting up your own pod. Once you've downloaded the Diaspora source and gotten it running on a server accessible to the internet, you open it up, log in…and are faced with the vast emptiness of splendid isolation.

Let's get you some friends

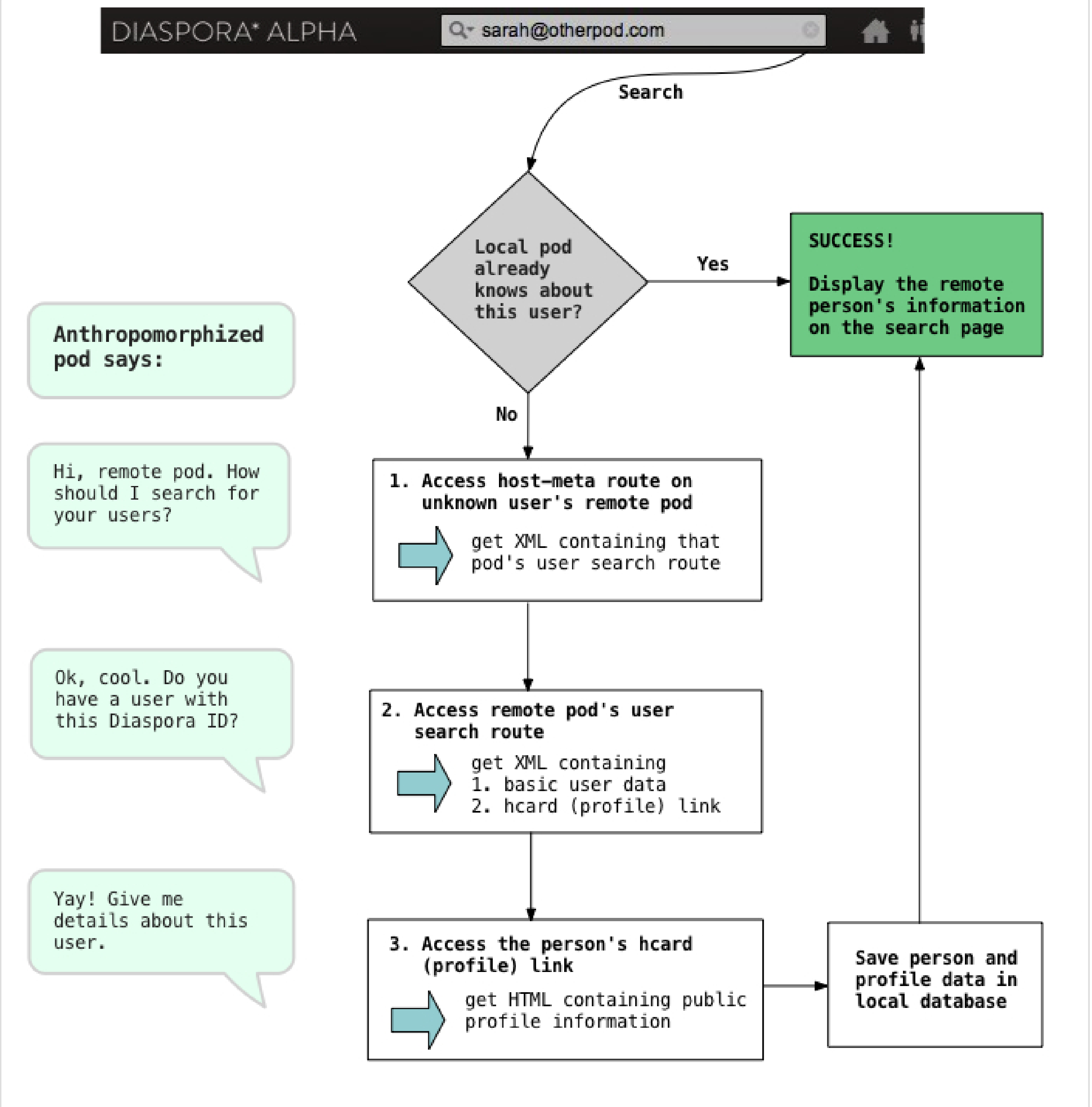
There is no central server that keeps a list of existing pods or existing users. Instead, Diaspora depends on an emerging-standard protocol called webfinger to discover users on remote pods. This all kicks off when you search for someone's Diaspora ID in your pod's search box.

> **Aside:** A Diaspora ID is made up of a username, followed by an @ sign, followed by the pod url. It looks a lot like an email address. But just like with Jabber IDs which also look like email addresses, you can't send email to it. It's just a unique identifier within the Diaspora ecosystem.

So you've talked to a friend who's on a (fictional) pod called otherpod.com, and you've gotten her Diaspora ID – sarah@otherpod.com. You want to connect, so you put sarah@otherpod.com into the search box on your pod and hit go. A few seconds later, you see her information on the search page with a nice "Add to Aspect" button alongside.

Magic!

I'd love to claim that, but sadly ponycorns are in short supply around here. Here's how it goes down behind the scenes. A detailed explanation follows the diagram.



DIASPORA* ALPHA
sarah@otherpod.com
Search
Local pod already knows about this user?
Yes
SUCCESS! Display the remote person's information on the search page
No
Anthropomorphized pod says:
Hi, remote pod. How should I search for your users?
1. Access host-meta route on unknown user's remote pod
get XML containing that pod's user search route
Ok, cool. Do you have a user with this Diaspora ID?
2. Access remote pod's user search route
get XML containing 1. basic user data 2. hcard (profile) link
Yay! Give me details about this user.
3. Access the person's hcard (profile) link
get HTML containing public profile information
Save person and profile data in local database

How a Diaspora pod finds a user on a different pod

When it gets a search request for a Diaspora ID, the first thing your pod does is look in its local database to see if it already knows about this person. This is the grey diamond in the diagram. If it can skip all this drama and just show you the information, it does so. But because you're on a brand new pod, the only user it knows about is you. So it prepares to retrieve the information you requested from the remote pod.

From here the process is:

1. Figure out where to search
2. Search
3. Retrieve detailed information

4. Cache data locally
5. Profit!

I suppose the last one is optional.

1. Find out where to search

Your pod extracts the pod URL from the Diaspora ID (sarah@otherpod.com becomes otherpod.com) and appends a standard location called “the host-meta route” to get this URL:

```
http://otherpod.com/.well-known/host-meta
```

This route is part of the webfinger standard. It’s the basic way you ask a server whether or not it supports webfinger.

Your pod then accesses this location and gets back a piece of XML in a format called XRD. Basically, accessing the host-meta route is the same as asking the pod, “How should I send you inquiries about users?” The XRD document that it returns tells your pod how to construct the query for the particular user you’re interested in. Here’s what it looks like:

The “template” on line 4 is the key. It tells your pod to query for the user by substituting their Diaspora ID for {uri}. So to search for your friend, your pod needs to construct need a URL like this:

```
https://otherpod.com/webfinger?q=sarah@otherpod.com
```

> **Aside:** All Diaspora pods accept user queries at the same location, so this step might seem redundant. But Diaspora pods also inter-operate with other, non-Diaspora social systems, and those may have different locations for querying for a user. In other words, when we get a Diaspora ID, we don’t actually know whether the pod is a Diaspora pod or something else. So we ask for the search route each time.

2. Search

Having figured out the right way to ask, your pod now queries for the user it wants. It accesses the query URL:

```
https://otherpod.com/webfinger?q=sarah@otherpod.com
```

This returns us another piece of XML – another XRD document – that gives us some basic information about the user, but mostly just tells us where to go to find more detailed info. Here’s what it looks like:

> **Aside:** The webfinger XRD document is supposed to be just links to information elsewhere. However, as you can see, Diaspora embeds some actual information, such as the person’s public key, in the document. We implemented this before we fully understood how XRD was supposed to work. We should at some point move that information to the hcard (see next section).

3. Retrieve profile

To fill out profile details, the pod extracts the “hcard location” from the webfinger XRD document. An hcard is a standard, structured way to represent profile data in HTML. The hcard location is on line 5 of the document above, with URL:

```
https://otherpod.com/hcard/users/4cec1e372c174347b90000ad
```

Your pod accesses the hcard location, and gets back a piece of HTML with additional profile details for the remote user, such as name. Here’s an excerpt of that hcard – it’s quite long.

It goes on, but I think you get the idea.

4. Cache data locally

Finally, having searched for the user and then retrieved her hcard, your pod extracts the profile details and saves them in its local database. sarah@otherpod.com now shows up in searches you do on your pod.

5. Profit!

Once you start following your friend, you'll get her updates as though she were a user local to your pod. If she also follows you, she'll get your updates in her stream too. From there…who knows what could happen.

This walkthrough covered just searching and basic user discovery, which is a tiny part of how Diaspora pods interoperate. Once you get into federation of posts and other content between pods, it's a whole different ballgame. Stay tuned for that in an upcoming installment.

September 17th, 2011 | Tags: diaspora | Category: Uncategorized

1 comment to How Diaspora Connects Users

Clemens
September 26, 2011 at 3:22 pm

If I got it right I'm able to move my profile to another pod. This would change my diaspora id name@pod1.com -> name@pod2.org or even worse NEWname@pod2.org. Does my buddies get notified that I changed the pod and that I have a different id now?

What happen if two friends change their pod at the same time? are they still able to find each other? thanks!