EPISODE 866

[INTRODUCTION]

[00:00:00] JM: Listen Notes is a podcast search engine. Listen Notes allows users to find podcasts by categories, popularity and search queries. Listen Notes is not a podcast client that you download on to your phone, but it allows users to build playlists. These playlists are automatically published to an RSS feed so that users can curate a playlist on desktop and subscribe to that playlist from any podcast client.

Podcasts are growing in popularity. There are more podcasts than ever before, and a podcast search engine needs to refresh its index regularly enough to capture the updates to the podcast universe. A podcast search engine needs to decide what to index. There are many potential fields to choose from within a podcast. You've got the podcast title, the description, the audio transcription, and other metadata. Should all of these factor into the results of a search engine?

Wenbin Fang is the founder of Listen Notes and he's the engineer who has built most of the application. He joins the show to talk about the world of podcasting and his work on Listen Notes, including the business model. Today, Listen Notes makes it money from advertisements and API requests. Yes, there is a growing market for applications that want to make API requests for querying the backend of Listen Notes, a podcast search engine.

FindCollabs is the company I'm building. FindCollabs is a place to find collaborators and build projects. If you're looking for collaborators or you're looking for a project to work on or you're looking for a cofounder for a new business, check out findcollabs.com. There are hundreds of collaborators on there looking for other people to work with, and we're having an online hackathon with $2,500 in prizes. So if you're working on a project or you're looking for other programmers to build a project with, check out FindCollabs.

I've also been interviewing some people for the FindCollabs podcast. If you want to learn more about the community and how the projects come together, you can check out the FindCollabs podcast.

Thanks for listening, and let's get on with the show.

[SPONSOR MESSAGE]

[00:02:32] JM: You probably do not enjoy searching for a job. Engineers don't like sacrificing their time to do phone screens, and we don't like doing whiteboard problems and working on tedious take home projects. Everyone knows the software hiring process is not perfect. But what's the alternative? Triplebyte is the alternative.

Triplebyte is a platform for finding a great software job faster. Triplebyte works with 400+ tech companies, including Dropbox, Adobe, Coursera and Cruise Automation. Triplebyte improves the hiring process by saving you time and fast-tracking you to final interviews. At triplebyte.com/ sedaily, you can start your process by taking a quiz, and after the quiz you get interviewed by Triplebyte if you pass that quiz. If you pass that interview, you make it straight to multiple onsite interviews. If you take a job, you get an additional $1,000 signing bonus from Triplebyte because you use the link triplebyte.com/sedaily.

That $1,000 is nice, but you might be making much more since those multiple onsite interviews would put you in a great position to potentially get multiple offers, and then you could figure out what your salary actually should be. Triplebyte does not look at candidate's backgrounds, like resumes and where they've worked and where they went to school. Triplebyte only cares about whether someone can code. So I'm a huge fan of that aspect of their model. This means that they work with lots of people from nontraditional and unusual backgrounds.

To get started, just go to triplebyte.com/sedaily and take a quiz to get started. There's very little risk and you might find yourself in a great position getting multiple onsite interviews from just one quiz and a Triplebyte interview. Go to triplebyte.com/sedaily to try it out.

Thank you to Triplebyte.

[INTERVIEW]

[00:04:51] JM: Wenbin Fang, welcome to Software Engineering Daily.

[00:04:55] WF: Hello. Happy to be here.

[00:04:58] JM: You started a podcast search engine. Why would you do that?

[00:05:02] WF: It was originally a side project of mine. I started in early 2017, because I wanted to have a way to consume podcast by episodes, not by podcast. Those back then, all of podcast players require you to subscribe to podcast first and then listen to episodes, which was not very intuitive. So, a search engine is the first step. So you can search episodes by topic and then you only listen to those episodes. You don't need to subscribe to a podcast.

But back then I spent about one week finishing the prototype. I didn't touch the code for about 9 months. So consequently I used the side project during the nine months, and then in September 2017, I worked on it fulltime and gradually it became a company.

[00:06:06] JM: Well, the initial problem that you were solving, the fact that you wanted a way to search by episodes as supposed to subscribing to every podcast that you want to listen to an episode of. That is a typical problem of the podcast world. The podcast world has so many of these little rough edges that make it not ideal, but it also makes the industry pretty beautiful in certain ways. Why is it that there are so many small issues with the podcast industry? I mean, we could talk about analytics. We could talk about the number of podcast players that are out there. But why are there all these different little problems in the podcast world?

[00:06:51] WF: Because podcast distributed via RSS and not any – Is an open protocol. Not any company owns the full stake of the podcast world. So, for example, for videos, we have YouTube, which takes care of the player on the client side, the story, the hosting on the server side and it has the analytics and advertising, a bunch of things.

For podcast, well, is a distributed thing. Every podcaster – Theoretically, every podcaster needs to take care of [inaudible 00:07:30] and hosting, things like that. It's pretty fragmented. It's pretty fragmented.

[00:07:36] JM: How did podcast fit into your life?

[00:07:39] WF: Well, I listen to podcast a lot every day. I spend probably more than five hours listening to podcasts. I like consuming audio contents. So when I was a child, when I was in primary school in China, I started to listen in radio a lot. Over these years, I listen to podcast more and more.

The first time I listened to a podcast was 2008 when I got my first iPhone. Back then, there were not many podcasts. So, in my sense, to subscribe to a few podcasts and listen to those podcast again and again. But, gradually, there are more and more podcasts and I'm now able to subscribe to all of them.

[00:08:26] JM: Yeah. We're almost to the point where there is enough podcast content for me to actually continually have audio content that I'm looking forward to. But I don't know about you. I still get to a point – I would say most days, because I'm like you. I probably listen to podcast three to five hours per day. Most days, I reach a point where I'm like, "Man! There's nothing that I have that I want to listen to." I think there will be a point where I will just be drowning in content that I'm looking forward to, but we're not quite there yet.

[00:08:59] WF: Yeah. So, it's actually very interesting. So, before this announced, that would be around 2016. So I subscribed to quite a few podcasts. On one hand, I couldn't listen to every single episode of all those subscribed podcast. On the other hand, I am running out of content, because not every episode of those subscribed podcasts are interesting.

[00:09:27] JM: So, the domain of podcast search, there are some domains where we want a vertical search engine, and there are some domains where we just want a horizontal search engine. We basically want Google. What are the domains where a vertical search engine makes sense? Why is podcasting one of those vertical domains that makes sense to have its own search engine?

[00:09:52] WF: So, search engine – Let's take Google for example. How many search results on Google's first result page by default is 10, right? Among these 10 search results – These are webpages. Gradually, Google added more media types on the search result page, like when you search your thing, you can see images. You can see videos. You can see webpages, local

businesses, Google Maps. But on the screen, the screen size is limited. If you want to put one more media types in the search result page, you have to squeeze out all the media types.

So if you were Google, what kind of media types do you want to sacrifice to make way for podcast? In this case, I think podcast is creating some momentum. But I don't think podcast is mainstream yet. I don't think podcast is important enough for Google to sacrifice, let's say, image, video, local business or other types of media contents from the first result page. That's my take.

[00:11:10] JM: But shouldn't Google be able to figure out if a user is a frequent podcast listener and that they're interested in podcasts and shouldn't they be able to detect from the user's query. If I search like Joe Rogan software, then it should be sending me like podcast episodes about –

[00:11:30] WF: So, Google is actually doing a good job for this type of query. When you search your podcast name, you would see some related podcast in a search result page. So, in my case, is when we search a person's name, let's say if it is a person, it's just VC, a venture capitalist. So, Google presents podcast search results on a first page or Google present some podcast interviews. I think this is quite tricky. But if you search a venture capitalist names, all you get is podcast.

[00:12:12] JM: So, describe how people use Listen Notes, which is your podcast search engine.

[00:12:18] WF: So, Listen Notes is a search engine and database for podcasts. Actually, we provide tool interfaces. One is website for most people in the world no matter if you are technological or not. The second interface is API for developers. API stands for application programming interface. It's an interface.

So, let's look at websites first. So, on a website, you can use the search engine, but it's a playlist feature called Listen Data, which is as popular as the search engine itself. So, people can listen or they want to find some episodes about a person, about a topic. So they search and they find some episodes in the search result. Then what's next?

Typically, they will save these episodes into the listen data playlist, and many people actually bring the playlist to their favorite podcast apps. Because the playlist provides an RSS feed. So you can subscribe to these ISS feeds.

Myself, I don't subscribe to any podcast nowadays. I only subscribe to one RSS feed, which is my listen data playlist. So, by this point, I think I already listened to around 2,000 episodes in the listen data playlist.

[00:13:50] JM: Wow! Okay. So, this is blowing my mind. I just want to pause here, because I hadn't actually understood how you built this feature or how this feature interacted with podcast clients. So what you're describing is I can go on to listennotes.com. I can have a web browser based experience where I'm looking through all these episodes. You got a fully-fledged web search engine where I can find cool episodes by categories or by actually searching for – For example, I want to hear all the podcast interviews with – Oh! Who's a non-controversial example? Stewart Butterfield. I want to find all the interviews with Stewart Butterfield. I can add all of those to my listen later playlist, which creates an RSS feed, and then I can send that RSS feed to my podcast client, whatever the podcast client is, and that gives me one big RSS feed that can be my only RSS feed I needed to subscribe to.

[00:14:48] WF: Right, and you can share this RSS feed with any one. So, whenever you add a new content, all those people who subscribed to this feed will get the latest contents as well. Actually, some content curating services use this and knows to do the podcast curation. I don't know whether you've heard the service called The Browser.

[00:15:11] JM: What is it called, The Browser?

[00:15:14] WF: The Browser, that curates articles. You can Google it. It's on the first search result, The Browser. They are quite big now. Then recently, they entered the podcast space. Previously, they curated articles. Right now, they are curating podcasts as well. Basically, they use this end data. Use this and now to create a playlist and then they distribute this playlist to their audience.

Then, on this end data, you can also annotate episodes. So, like you find an episode, you can Listen Notes about this episode. These notes will appear in the RSS feed as well. So, people use whatever podcast app to subscribe to this feed and they can see the notes.

[00:16:07] JM: So, one of the challenges of the podcast space is that it is extremely decentralized. You have this RSS infrastructure, and the tension is that not only is it decentralized, but there are reasons for data gravity in order to build good podcast infrastructure. Specifically, to get a good podcast search experience or to get an ideal podcast search experience, you would want to take the transcription of every podcast episode. Typical podcast episodes are 40 to 80 minutes. You want to put it through a transcription algorithm, which is expensive. You get a transcription and then you want to index that transcription. So you not only have the expensive transcription process. You have the expensive metadata creation in order to build a search index across that and you want to do that for every podcast. Podcasting is this long tail world. So, that is one of the fundamental tensions that has made the podcast infrastructure kind of so hard for so long, because there's no centralized – Go ahead.

[00:17:19] WF: I'll argue that transcription is unnecessary. So, we can take YouTube, for example. Can you search transcripts on YouTube?

[00:17:27] JM: I think so. I think, actually – I don't know if it's explicit. I thought that YouTube actually does transcribe most videos. Maybe not most videos. Is it just the fat head perhaps?

[00:17:41] WF: You can. You can see captions, auto-generated captions for videos, but can you search? If you cannot search, why? Why not? Why not Google provide this kind of feature?

[00:17:55] JM: Well, it's not explicit. When you enter a search into Google search, there's a whole lot going on beneath the covers, and that's the same thing that goes on at YouTube. It doesn't tell you, "Your query is now being run against the transcript." It's just happening. I believe that the transcript of the YouTube videos are there.

[00:18:13] WF: Yeah, so –

[00:18:13] JM: You don't think so?

[00:18:14] WF: I don't think so. Because I think for YouTube videos or for podcast, a lot of content, like what we are doing right now, conversational contents, and the key was in this type of content are buried on a bunch of [inaudible 00:18:31] which would dilute the signals for search. I'll argue that if anything important in a podcast, they should already be in the metadata, either in title, in description, in show notes.

[00:18:48] JM: No. That's not true. No way. Not a chance. Not a chance. No, I mean, I do so many shows where – Okay. I think I'm an exception, because I write a pretty long introduction for most of my interviews. So, like, the introduction goes into this preamble, which is in the WordPress post, which ends up as part of the metadata. So that's pretty useful. But think about Joe Rogan's episodes. They're three hours long and the only metadata is episode 1,345, Jack Dorsey. A conversation with Twitter's CEO, Jack Dorsey. That is not nearly enough metadata.

[00:19:27] WF: Yeah, there are always some cases like that, you get a name in the title. People search Jack Dorsey, they can find the episode. Yeah, there are more key words you can put in show notes and some podcasts are doing good job in providing informative key notes in the show notes.

For Listen Notes, we indexed all metadata, but we only index a small portion of the episodes in the transcripts.

[00:20:03] JM: Really? So you just transcribe a subset?

[00:20:05] WF: We subscribe a few episodes. As you said, transcription is expensive and there are so many episodes in the world, right? We cannot –

[00:20:17] JM: How expensive is it? Yeah, because I'm trying to do this. I've done a little bit of research. It's so expensive.

[00:20:23] WF: We cannot transcribe all of these episodes all at once.

[00:20:26] JM: How much does it cost per episode or per minute or per hour?

[00:20:29] WF: So, we use Google's speech to text API to do transcription. If I remember correctly, every minute, it costs 0.0 something dollars. Yeah, I don't even remember how much. Let me do a quick search.

[00:20:49] JM: It doesn't matter. It doesn't matter. I can give you some numerical context. There was a winner break a couple of years ago, and I don't really celebrate holidays. So during the break I was like, "Cool. This is a great set of two weeks and I can hack on something," and I was like, "I'm going to transcribe some of my episodes and see if I can do some clustering across those episodes," because I think that'd be interesting if I could naturally cluster episodes around React.js or Bitcoin or Kubernetes or something. So I get on to Google Cloud, and you got your free $300 in credits and I'm like, "Oh! Surely, $300 will be enough to get all of my episodes transcribed and clustered." I got through maybe 25 episodes. So it was like $300 to index or to transcribe about 25 episodes. I think it was 25 or 30 episodes.

[00:21:39] WF: Yeah, I can remember now. It's 0.024 per minute.

[00:21:46] JM: 0.024 per minute – Sorry, 60 minutes is 10 bucks?

[00:21:50] WF: No, 1.44. $1.44

[00:21:54] JM: Oh! It's only a dollar per hour. I guess the price has gone down. That's not too bad. A dollar per hour is not too bad.

[00:22:02] WF: Not too bad.

[00:22:03] JM: It will definitely get cheaper. Definitely get cheap enough eventually.

[00:22:08] WF: Yeah.

[00:22:09] JM: But you said you transcribe a subset of the episodes.

[00:22:13] WF: Only a few.

[00:22:15] JM: The most popular ones or what?

[00:22:16] WF: No. So, actually, people can order transcripts through Listen Notes website. If they want a transcript, they can pay Listen Notes and then we use the money to do the speech to text transcript from Google. Basically, the price we charge barely cover the cost for the API pricing. Yeah.

[00:22:40] JM: Right. Is the automated transcription, is that good enough, or do you have to kick it to like a mechanical turk and get it proofread?

[00:22:48] WF: Well, it's good enough for the search engine to index. So you service some keywords. I think it's also good enough for human beings to read.

[00:23:05] JM: To sort of read.

[00:23:07] WF: Certainly, it's far from perfect. But many people, they have disability on hearing. So they cannot listen to podcast contents so they need to read. This is primarily use case for these transcripts. Yeah.

[SPONSOR MESSAGE]

[00:23:19] JM: DigitalOcean is a simple, developer friendly cloud platform. DigitalOcean is optimized to make managing and scaling applications easy with an intuitive API, multiple storage options, integrated firewalls, load balancers and more. With predictable pricing and flexible configurations and world-class customer support, you'll get access to all the infrastructure services you need to grow. DigitalOcean is simple. If you don't need the complexity of the complex cloud providers, try out DigitalOcean with their simple interface and their great customer support, plus they've got 2,000+ tutorials to help you stay up-to-date with the latest open source software and languages and frameworks. You can get started on DigitalOcean for free at do.co/sedaily.

One thing that makes DigitalOcean special is they're really interested in long-term developer productivity, and I remember one particular example of this when I found a tutorial in DigitalOcean about how to get started on a different cloud provider. I thought that really stood for a sense of confidence, and an attention to just getting developers off the ground faster, and they've continued to do that with DigitalOcean today. All their services are easy to use and have simple interfaces.

Try it out at do.co/sedaily. That's the D-O.C-O/sedaily. You will get started for free with some free credits. Thanks to DigitalOcean for being a sponsor of Software Engineering Daily.

[INTERVIEW CONTINUED]

[00:25:36] JM: Taking a step back, the format that has been the most popular for podcasts has been the two-person interview format, the kind of conversation we're having right now. What I wonder is, is that format just popular because it's convenient, because it's kind of easiest to make, or is this something that we just crave, this conversational format? Whether it's in audio or written format? Do we just want more conversations?

[00:26:08] WF: I think it's both. So, on the supply side, certainly, it's easier to produce. You and me, we are talking right now. For me, I don't need to prepare anything. You ask question, I answer, right? So on the supply side. On the creator side, it's easier to create. On the demand side, I think listeners want to have this type of content, conversational content. One podcast to company. So, it is more genuinely for this type of conversation. It's not manufacturer to some degree, yeah.

[00:26:42] JM: Right. I thought it was so interesting. Did you listen when Jack Dorsey went on the podcast tour recently? He did five or six podcasts. Did you listen to some of those?

[00:26:51] WF: I listened to the Joe Rogan one, and I listened to maybe others. It goes through my playlist. I listen to so many podcasts and –

[00:27:00] JM: You got to use Listen Notes. Those ones are good. They were good. But what was interesting was there was a lot of reporting on his podcast episodes after he had done

them. I thought it was interesting, because the podcast episodes were basically the event that happened and the news sources became the derivatives of the podcast content. So, I think in some ways, the podcast world is like a front-running of the news cycle. It's like the news is developing on the podcasts now.

[00:27:36] WF: Yeah, I agree. It's very easy to create a podcast now. You just talk. You don't need much preparation and you can capture a lot of attention from Listen Notes. You can spend one more or two hour recording a podcast. Maybe you'll spend another few hours editing. You capture a full hour's attention from listeners. Compare the ways of blogging. You spend maybe one hour writing and you capture two minutes of reader's attention. So the ROI doesn't work.

[00:28:14] JM: Yeah, it does. Let's talk a little bit about one more just quick question on this cultural issue, cultural aspect of podcasting. Do you think that there is an increased desire for these long form conversations because we are exhausted from the written word or exhausted from the social media formats, the tweets, the Facebook posts? We just want something different. Is there a sense of exhaustion that is causing people to really want podcasts more?

[00:28:47] WF: To be frank, I don't know how to answer this. I can only talk about my own experience.

[00:28:51] JM: Yeah, tell me about your experience. We have to admit, this is a strange world you and I have gotten ourselves into when we listen to four or five hours of podcast per day. You live in San Francisco, right?

[00:29:05] WF: Yeah, I'm in San Francisco, yeah.

[00:29:07] JM: Yeah. So, we both live in San Francisco, the most vibrant, interesting – Well, okay. I don't want to say most vibrant, interesting. There's a lot of people here we could have great conversations with, and yet we are listening to podcasts for 4 or 5 hours a day. Why are we doing this?

[00:29:21] WF: Where do you listen to podcast? When do you listen to podcast? I'm curious.

[00:29:25] JM: When I'm cooking. When I'm going for a run. When I'm going for a walk. Yeah, that's the main thing. But, I mean, it causes me to cook more. It causes me to go on more runs and walks and go to the gym more.

[00:29:37] WF: Yeah. Similarly, I also listen to podcast when I am writing code. Nowadays, more than software engineering is mostly about Googling [inaudible 00:29:49], GitHub and moving code around. So, there's also a time you don't need to think much is basically my clinical activities. But sometimes you don't need to think. So you can double tap Airpods to stop for 5 minutes, 10 minutes after you finish some critical code and then, okay, resume listening.

I also listen to podcast in situations like yours with my eyes, with my hands occupied. But my ears are available to consume information. Often times, I just want to get some information. It doesn't need to be a knowledge or anything. I just want to get entertained. I listen to a lot of gemstones podcasts, and if I miss 10 minutes' content, probably that's fine. So, that's my case.

[00:30:44] JM: Okay. Well, let's get to engineering. You index all the podcasts. That sounds complicated. How often are you crawling the world of podcasts and how do you do that crawling?

[00:30:58] WF: So, the answer is 24/7. So it's like any search engine. You need to have crawlers running every second, every minute. Essentially, Listen Notes is a directory of podcast. On top of a directory, they also search engine. We can generalize these a bit. Many online services are directory of things. Or they start here as directory of things.

Listen Notes is a directory of podcast. YELP, we consider as directory of businesses. IMDd, directory of movies. Giffy, directory of GIFs. All these directory of things, how do we build the directory? Yeah, there are several ways. First, you can run crawler to automatically crawl data from the open Internet. Second, you can have human beings to manually type in the contents either from users or from yourself as a developer. Third, you can buy data from third-party services.

So, for Listen Notes, obviously, we'd run crawlers. We can get into a little bit detail on these parts in a minute. We also rely on human beings who type in content to submit new podcasts.

So, if I cannot find any podcast on Listen Notes, I myself would submit the RSS feed to Listen Notes. But as Listen Notes gets more and more popular, there are dozens of new podcasts submitted to Listen Notes by podcasters themselves. So they come to Listen Notes. They submit an RSS feed. Most of these podcasts are new podcasts.

They may submit to iTunes or other directories at the same time, but other directories may have very long approval process. Let me wait a few days to get approved. But for Listen Notes, we approve almost instantly.

The third, buying data from a third-party. We don't do that. Other directory of things type services, they may data from a third-party. For example, if you want to start a competitor of YELP. Somehow, you need build a local business directory. How do you do that? I believe if you Google local business database, you will find some companies selling this kind of database. So you can boost drive by just paying people to buy the database.

[00:33:44] JM: What's your crawling infrastructure look like? Do you have long-running servers that are sitting there or are you spinning up Lambda functions? What are you doing?

[00:33:53] WF: No Lambda. I'm using very boring technologies.

[00:33:58] JM: Boring is fine. Tell me about the boring.

[00:34:01] WF: Yeah. So, at this moment, I have eight worker servers running, eight servers. I run crawlers on these 8 servers. So, first, we need to get the podcast, get RSS feed. You can imagine I maintain a list of destination websites that we constantly crawl to see if there are RSS feeds. We also have crawlers to check if podcasts have new episodes. So, that's pretty much it.

[00:34:35] JM: Okay. So, do you just query – Are you mainly looking at the Apple index. What's the best thing to query to actually understand if new podcasts have been added?

[00:34:46] WF: So, if you Google podcast directory, you can see a lot of directories. ITunes directory is the biggest one. There are other directories on the Internet. For example,

Soundcloud, Stitcher. A lot. A lot of them. I think these are the common criteria for all the podcast apps if they build their own database, they build their own directory.

[00:35:14] JM: Does Spotify publicize what's in their directory, or is it a walled garden?

[00:35:19] WF: I don't cross-modify. So I cannot answer this question. Spotify's directory – I think Spotify's directory is quite small compared with others.

[00:35:29] JM: So they don't bring in everything? It's just like a subset?

[00:35:33] WF: So, their podcast business is quite new. It's growing fast, but it's quite new. I need to do some research on my end as well.

[00:35:42] JM: Yeah, that's an interesting one. Tell me about what you do to build your search engine. You're using Elasticsearch, right?

[00:35:50] WF: Yes, of course. It's 2019 now.

[00:35:54] JM: It's 2019. Okay. So, what's the process when you're ingesting a new set of episodes and you want to add those episodes to your Elasticsearch index? What kind of code runs?

[00:36:05] WF: Well, it's just plain, old Python code. Basically, when I check all podcasts RSS feeds, constantly checking. Whenever we find new episodes, we add these episodes into PostgreS, which is the single source of tools, PostgreS database. At the same time, we add a metadata into Elasticsearch. You use Elasticsearch index API. So, not very fancy.

[00:36:34] JM: Nothing fancy. Okay. If cost management an issue for you? How much data do you have and is it problematic?

[00:36:42] WF: Let me see. For Elasticsearch data, a few hundred gigabytes, I think at this moment. A few hundred. It's not a lot. A few hundred gigabytes is not a big data.

[00:36:54] JM: Are there any other elements of the infrastructure that are particularly expensive or do you feel like it's pretty cost efficient operation?

[00:37:02] WF: So, in total, I'm spending a little bit over $1,000 per month for the entire infrastructure. So I'm running everything on AWS, and there are 20 EC2 instances, 8 workers, worker servers, 3 Elasticsearch servers, 3 database servers running PostgreS, and 3 API servers, 2 web servers. I think that's pretty much it. Oh! One load balancer, 20 servers.

[00:37:43] JM: Did you say two web servers and three API servers?

[00:37:47] WF: Yes, three API servers. So, the API business is not very obvious to most people. So many people come to Listen Notes, oh it's a website service, where before we provided tool interfaces. One for most people in the world, website and the second one is for developers. So, right now, we have been tending two versions of APIs, version one and version two. Version one is the legacy stuff. So, one server is enough. Version two is the main focus —

[00:38:21] JM: Well, I'm a believer in your strategy, because I don't think that the podcast world wants to be centralized. If it doesn't become more centralized – Well, okay. People are going to centralize it. People are also going to decentralize it at a faster pace. It's like what happened with Linux? Well, it was centralized is the sense that like Red Hat made an enterprise version of it. But it was also just like massively decentralized at a much faster rate. I think podcasts are going to work in the same way where people are going to build centralized indexes of them and data gravity stuff around like subsets of the podcast world. But at the same time, people like the podcast world as it is and they're going to double down on the way that it's already been built, which is kind of where you're going. The idea of a podcast API is really smart, because people want to build their own clients, and people want to have their own clients. People want strange clients.

[00:39:23] WF: Yeah. People have different tastes. So this API business is an extent.

[00:39:30] JM: It's a what?

[00:39:30] WF: It's an extent. I didn't plan for it, but some developer requested it from me. They asked – One developer asked me if I can provide the API for them to access the search engine. So, it was late 2017. So, I quickly put together a prototype providing three endpoints. One for search and one for getting metadata for particular podcast, and the third one, getting metadata for a particular episode. Then I hook up with a third-party API gateway, which handles the building and documentation, things like that. So, it's a pretty quick process. So I joined API end of 2017 and I didn't touch it for many months. So, the API business typically grow very slowly, because when people discover your API, they need to spend a few months to build an app first and they need to spend another few months to market the app to grow the user base to iterate. So, if you start an API business, you need to wait for a longtime to see any kind of traction. So, for us it's the same so be joined API in late 2017 and we started to see some traction in the end of 2018. So almost one year. After some apps launched and gained some usage.

[00:41:03] JM: Let's unpack the API detail. So, first of all, you mentioned this third-party API gateway. Explain the role of an API gateway and who is your API gateway provider.

[00:41:14] WF: Right. I actually use the word API marketplace.

[00:41:20] JM: You use Rapid API?

[00:41:22] WF: Yes, Rapid API.

[00:41:24] JM: Okay. I love Rapid.

[00:41:26] WF: I need to explain a bit. So, initially, I used Mashape.

[00:41:32] JM: Mashape. Yeah, Kong. They sold their business – Sold the API marketplace.

[00:41:34] WF: Yeah, exactly. It's a big complex. Initially, I used that service and then kind of merge into Rapid API, and then Rapid API rebranded a bit. It's essentially a marketplace. It's like Uber, but for API, and it's two-side market. You have API providers, like me, providing podcast API. You can imagine, there's an API for everything, weather API, maps API, music API,

whatever. On the other side, developers, they want to use API to accelerate the development of their apps.

The API marketplace, essentially API gateway. So, any API request need to go through the API gateway so they can manage how many request a particular developer is using and then can do the accounting how much money an API developer need to pay for basically the API provider, things like that. They handle the building. They take I think 20% cut for the fee.

[00:42:47] JM: Holy smokes. I didn't know it was 20%. That's a lot.

[00:42:51] WF: It makes sense. App store takes 30%.

[00:42:58] JM: I don't know if that's the healthiest benchmark.

[00:43:00] WF: Right now we have our own API gateway. So, we are not relying on Rapid API anymore.

[00:43:05] JM: Okay. All right.

[00:43:06] WF: [inaudible 00:43:06]. Yeah.

[00:43:09] JM: You moved to Kong?

[00:43:10] WF: No. I build my own from scratch.

[00:43:13] JM: Build your own. Okay. All right. Well, that's probably going to be your next product, the accidental API gateway product. So, how do people use the API? Are they building clients around it or are they doing weird things?

[00:43:27] WF: All kinds of things. So, if you go to listennotes.com/api, and I have a list for apps that are using Listen Notes API, and those apps are already launched. More apps are still under development. People need to spend many months to build an app.

So, for those apps, let's see, most of these apps are some kind of player apps. So there's an app called Part Talk, is to discuss podcasts in groups. There's an app called Podcast Lounge, is Windows 10 app, a desktop app. There's an app called Podcoin. You listen to podcast, you earn coins. It has nothing to do with crypto. It's virtual currency, and you can use the coin – You can use the virtual currency to buy things, I think, exchange it for Amazon gift card to donate to charity. This app is pretty popular actually.

So, Jeff, let me ask you a question. So, which category in app store does most podcast apps belong to? Which category?

[00:44:48] JM: Oh! I don't know the categories well, but media maybe, or audio.

[00:44:53] WF: News.

[00:44:54] JM: News. Oh! Of course.

[00:44:56] WF: Okay. So, I don't have really the answers, but afterwards, if you're interested, go to the news category and then see the top charts. You can see a bunch of podcast apps there. So, in top charts of the news category in app store, you can see a bunch of podcast apps. Yeah, you'd be surprised. There are so many podcast apps nowadays, and there will be new podcasts, more and more new podcast apps coming up.

[SPONSOR MESSAGE]

[00:45:36] JM: Commercial open source software businesses build their business model around an open source software project. Software businesses built around open source software operate differently than those built around proprietary software.

The Open Core Summit is a conference before commercial open source software. If you are building a business around open source software, check out the Open Core Summit, September 19th and 20th at the Palace of Fine Arts in San Francisco. Go to opencoresummit.com to register.

At Open Core Summit, we'll discuss the engineering, business strategy and investment landscape of commercial open source software businesses. Speakers will include people from HashiCorp, GitLab, Confluent, MongoDB and Docker. I will be emceeing the event, and I'm hoping to do some on-stage podcast-style dialogues.

I am excited about the Open Core Summit, because open source software is the future. Most businesses don't gain that much by having their software be proprietary. As it becomes easier to build secure software, there will be even fewer reasons not to open source your code.

I love commercial open source businesses because there are so many interesting technical problems. You got governance issues. You got a strange business model. I am looking forward to exploring these curiosities at the Open Core Summit, and I hope to see you there. If you want to attend, check out opencoresummit.com. The conference is September 19th and 20th in San Francisco.

Open source is changing the world of software and it's changing the world that we live in. Check out the Open Core Summit by going to opencoresummit.com.

[INTERVIEW CONTINUED]

[00:47:37] JM: Okay. Hold on. Let's focus on this Podcoin thing. Why is there a podcast that rewards you with Amazon gift cards for listening to podcasts?

[00:47:52] WF: I don't know. I don't know. So, I think A, marketing is very powerful nowadays, right? You can spend money on advertising on Facebook, on Google, or you can hand out money directly to users. I don't know what it is the answer.

[00:48:10] JM: That sounds like – That's got fraud written all over it. It's like got to be some kind of scamminess.

[00:48:17] WF: No. I think it's quite legit. You can download an app and then try it out. Yeah.

[00:48:23] JM: I don't understand. What's the economics? Okay. If I'm the Podcoin owner – I guess there's in-app advertising maybe? So like the premise is. So I download Podcoin –

[00:48:34] WF: I assume they are still very early in this app development. They can build the audience first and then monetize in-app, right? Maybe they can do partnerships. So, oh you want to promo your product here? Then people can use coins to exchange it for products. Something like that.

Let's get back to a topic, who are using Listen Notes API. These are apps.

[00:49:01] JM: Sure. Right. Well, what about research? Are there like researchers who are querying for like podcast statistics or something?

[00:49:10] WF: Noto for statistics. We don't provide statistics. But a lot of PR agencies or marketing agencies, they are actually using Listen Notes, either the API or the website to do research on their own. So, for example, a PR agency may want to check whether their clients appear on certain podcasts, something like that. Some of them are using Listen Notes API to build internal tools to streamline their process. For these kinds of tools, I don't think I can [inaudible 00:49:48] on my public page. So it's internal tools.

[00:49:53] JM: So, the transcription business, when somebody makes an API request –

[00:49:56] WF: We don't provide transcription through API.

[00:50:00] JM: Why not?

[00:50:02] WF: The transcripts – To be frank, transcripts are not very accurate, and only a small portion, a tiny portion of all episodes in the world have transcripts on Listen Notes.

[00:50:13] JM: But you said earlier that like you can request a transcript, and then you will get the transcript made and then there's a callback eventually and then –

[00:50:24] WF: We only provide this on the website for now. Maybe there's an extension to API in the future. I'm not sure now.

[00:50:32] JM: That'd be cool. I think it'd be cool if – Because the thing is, what you could do is you could charge the first person who request the transcript, the full cost for the transcription plus they actually get the transcription. But if future people request the same transcript, you've already got it made. You can keep charging and you can make a profit. But that's probably not a very profitable business.

[00:51:01] WF: Right. Transcripts are not my main focus. It's nice to have this thing.

[00:51:06] JM: What do you think is going to be the viable business model?

[00:51:09] WF: It's very interesting. So, last month was actually the first month that we broke even.

[00:51:17] JM: You broke even. Congratulations!

[00:51:18] WF: Yeah.

[00:51:19] JM: Does that include your salary?

[00:51:21] WF: Including my living expense in San Francisco plus from business calls.

[00:51:27] JM: Congratulations!

[00:51:29] WF: Thank you.

[00:51:30] JM: That's fantastic.

[00:51:30] WF: Just broke even. No profit yet. So, still need to work my way up.

[00:51:35] JM: Right. That's mostly off of API payments?

[00:51:38] WF: Some from API and some from advertising. If you go to my website, you'll see the [inaudible 00:51:43]. The primary sources of revenue are advertising website and the API, from two interfaces.

[00:51:53] JM: Right. Yeah. So the website itself is pretty popular.

[00:51:57] WF: Okay. Creating something like 1.4 million page views per month right now.

[00:52:03] JM: That's solid.

[00:52:05] WF: And it's growing. Yeah, not bad. API is growing as well.

[00:52:10] JM: How frequent is this use case where somebody goes to your website and they build a feed for themselves and then they subscribe to the –

[00:52:19] WF: Actually, the playlist feature is very popular.

[00:52:23] JM: I got to do that. I'm going to do that.

[00:52:26] WF: As I said, some people are actually building a business on top of the playlist feature for free.

[00:52:33] JM: They're building like a client around that, because that's missing a client, right?

[00:52:37] WF: No. So they are curating content and they are charging their audience for accessing that. So, because my playlist can be private, it can be unlisted, it can be public. So you can make a playlist private or unlisted. So you can only share this playlist with people who pay you. Something like that.

[00:52:57] JM: Well, see. That's actually a pretty good business, because I'm sure you've had this experience where you just get like – Maybe it's Saturday and you've got nothing to do the whole weekend. Like your whole calendar is clear and you're like. I want to spend the weekend.

I want to listen to even more podcast than in normally do. I'm going to total binge. I'm going to do 10 hours this weekend. But I really want to find some good stuff to listen to. So you spend like 15 to 30 minutes just looking for good interviews. You can always find weird old stuff or like interviews with some of your favorite people.

I remember, I was interested in – There was some investor that I found interesting and like I found like really old interviews with him. It was like super –

[00:53:42] WF: Exactly.

[00:53:42] JM: I was like, "Wow! This is really interesting," because you get to go back and kind of hear, "Oh! So this person –" They sounded as interesting back then as they do today. But they were not as prominent.

[00:53:53] WF: So this is my favorite use case. So, whenever I see some famous people, I would want to find their old podcast interviews.

[00:54:03] JM: Yeah.

[00:54:04] WF: Previously, maybe last year, when GitHub got acquired and I found some old interviews from GitHub founders. Really old, 2008 or 2010, something like that.

[00:54:17] JM: How were there?

[00:54:18] WF: It's very interesting. They were very smart, obviously. Sounds very smarty in old interviews. You can listen yourself.

[00:54:26] JM: I got to check those out. Do you have public playlists of the stuff you're curating?

[00:54:31] WF: I have my public playlist. I can send it to you.

[00:54:34] JM: Oh my god! Please do.

[00:54:36] WF: Right now I'm checking, it's 1977 – I guess so.

[00:54:42] JM: Wonderful! That should take care of the plane ride I have tomorrow.

[00:54:47] WF: Yup.

[00:54:48] JM: I know you've written about why you don't have a client yet. I mean, just talking to you, like you've got a really good taste for how people – I mean, your description of how to use – This idea of like, "Oh! You build your own RSS feed and just makes an endpoint for you," and then you can put it into your podcast client. I had never thought about that. I've thought so much about the podcasting landscape and I had not thought about that idea. But that seems really smart.

[00:55:18] WF: I do admit, this not my original idea. So, my original idea was, "Okay. I need to have a playlist, a feature with episodes. There was not an RSS part. But one user of Listen Notes suggested this feature. So, I implemented a feature. So, I didn't come up with the RSS idea.

[00:55:41] JM: Well, okay. That's fine. But have you thought anymore? Since you wrote that article, have you thought more about making a client? What are the priorities right now? What do you think are the biggest impacts you can make on your business, or what are the biggest improvements you can make?

[00:55:55] WF: Yeah. So, I'm not going to make a podcast app or anything, because making an app is very resource intensive. You spend time building the app and it's just the beginning. Launching the app is just the beginning. You need to continuously iterate the app and you also need to market the app, and it's tough.

Given that, right now, there's only one fulltime employee in the company. Me. It's very hard to do that. I'm not going to do that. It also have a conflict of interest with my user's API.

[00:56:32] JM: That's a good point.

[00:56:34] WF: Better, I don't need to. Yeah. Actually, the API is my main focus right now. [inaudible 00:56:41] to accelerate the innovation of the podcast application space. We provide API, and there are many smart people out there. They can creatively use their API to build something wonderful.

[00:56:59] JM: I totally agree with you. I think that's a – Man! What an interesting – Who have thought? Podcast API. You're from China, right?

[00:57:08] WF: Yeah, I'm from China. I grew up in China and I come to U.S. in 2010 for computer science Ph.D.

[00:57:17] JM: How do the engineering philosophies of Chinese engineers differ from those in the U.S.?

[00:57:26] WF: I don't know how to answer this question. So, because I spend most of my professional career in U.S. So, I'm actually not very familiar with the engineering side in China. I have friends working in China and we talk a little bit. But I don't think in the position to make comments here.

[00:57:49] JM: Are we so different? I'm just trying to understand. As we see more and more news and content that is drawing divisive lines between the American culture and the Chinese culture –

[00:58:03] WF: Are you talking about 996, something like that?

[00:58:07] JM: Well, not really the 966 stuff. I'm actually a huge fan of the 966 stuff. I think that what people complain about the 966 stuff on Twitter, it's hilarious, because I'm just like, "Why don't you go back to work instead of tweeting? If you tweeted less, you probably have more time to do 996." Dude, I'm 977. I'm all about the work. Well, I mean, I guess a lot of my work is listening to podcast. I guess work is perhaps easier for me. Some of those 4 to 5 hours of podcast listening are considered like "work" for myself.

[00:58:37] WF: Me too.

[00:58:38] JM: I'm more referring to the – I guess the perception. There's just perception of China as this big monolith that operates with cultural norms that are distinctly different than that of the U.S. I went to KubeCon for China, and this was a – I went to China for KubeCon. This was a very small interaction with China. But my sense just from talking to engineers, is people in China are just – They're just like the Americans. Maybe there's a difference in the leadership, but – I don't know. I'm just trying to ease my fears of divisiveness in the future. I'm just trying to understand how globalized our current world is.

[00:59:17] WF: I think engineers are pretty similar. We all use search engine. We all search answers on StableFlow. We all use GitHub.

[00:59:26] JM: Although podcast usage is different, right? They pay for podcasts.

[00:59:29] WF: No. I disagree.

[00:59:30] JM: They do? Oh, they don't. Okay.

[00:59:32] WF: I can send you a couple podcasts episodes about Chinese podcasting.

[00:59:38] JM: Oh, please do.

[00:59:39] WF: "Podcasting". Those are very good contents. Demystifies the China podcasting stuff. So, probably you read the article from – I don't know which news media write the article saying China's podcasting market is like 7 billion or something like that, which is not true.

We should first define what's a podcast. Well, audio distributed via RSS. It's open. It's free. If you build a wall garden and then you put audio inside a walled garden, you put up the subscription model. I don't think that's podcast.

In China, yes, there are a bunch of paid audio contents, but mostly they like Coursera for audios. They are all educational contents mostly.

[01:00:35] JM: Very different from podcast. Very different from the open ecosystem.

[01:00:39] WF: Yeah.

[01:00:39] JM: Okay. Last question, you have worked at a couple other companies before Listen Notes. You worked at Nextdoor. Nextdoor looks to be very successful. You've also worked at an early startup that failed. So you've seen what can go right a company. You've seen what can go wrong. Tell me some of the most important lessons that you've learned in the startup world.

[01:01:06] WF: Don't give up too early. So, a startup can only fail when the founders give up. Yes, startups can run out of money, but you can keep going. You can be scrappy. Eventually, if you stick around long enough, probably you can make it work. Don't give up too early.

[01:01:27] JM: That's a great piece of advice and easier said than done.

[01:01:31] WF: Yes. When you start you should give it two years, two full years to see how things will work. I don't know where I had read this. Maybe it's from some tweet. So, someone is saying that for SaaS business, you need to give two years before seeing any meaning for revenue, something like that.

[01:01:57] JM: Jason Lumpkin always says it. He's always like, probably – He's like, "After two years, it starts to get okay. After three years, it starts to get positive. After four years, you really start to gain momentum."

[01:02:11] WF: Yeah. For Listen Notes, I've been working on this close to two years by this point. So, we just broke even last month.

[01:02:19] JM: Hey, man! That's a milestone. That's a huge milestone.

[01:02:22] WF: That's the beginning. So, yeah.

[01:02:24] JM: All right. Well, just the beginning.

[01:02:25] **WF**: Typically, when we hear about some wonderful digital product or some famous stop, it's already many, many ears after they were founded. So, companies don't grow very fast, typically. Sometimes we hear "overnight success" isn't always true. We don't know the backstories.

[01:02:48] **JM**: Totally agree. Well, Wenbin, I'm sure we'll do more in the future, because you're not going to give up and I'm sure the product is going to evolve. It's great to talk to you. We got to meet up in person at some point when we're not too busy listening to a podcast for 5 hours per day. We're so busy listening to podcast for five hours a day. We can't even meet in person to do an interview in San Francisco. I bet we're like less than a mile from each other right now.

Well, Wenbin, thanks for coming on the show. It's been really fun talking to you.

[01:03:14] **WF**: You too. Thank you.

[END OF INTERVIEW]

[01:03:19] **JM**: FindCollabs is a place to find collaborators and build projects. The internet is a great place for communicating with each other. Why aren't we building projects together? Why is it so hard to find other people that will work with you on a new project reliably? Why is it so hard to find cofounders to start a small business with?

We started FindCollabs to solve these problems. I love starting new projects, software platforms, musical songs, visual art, podcasts, and when I create these projects, I want to share them with people. I want people to be able to follow my progress, because as they see that I'm dedicated to consistently delivering quality work, maybe they'll be inspired to join me, and it doesn't have to be on a financial basis. It could be just to collaborate and to make artistic progress.

You can see my profile on findcollabs.com by searching for Jeff Mayerson. You can see the different projects I've made; games, podcasts, open source software. Some of these projects are incomplete. Some of them are well-developed, and this is how innovation and invention

happens. Projects start out in an incomplete state, and if you can prove to the world that you work on projects with dedication and consistency, you will get other people to join you. Whether you want to hire them or if you just want to collaborate casually.

You can see my profile and many other people's profiles, projects that they're posting on FindCollabs, and I would love to see your projects there too. You can check it out by going to findcollabs.com, F-I-N-D C-O-L-L-A-B-S.com. You can post a project that you're working on no matter how incomplete it is.

The best projects in the world start with an inspiring vision, whether or not they have code attached. If you like to compete, the FindCollabs Open is for you. The FindCollabs Open is a $2,500 hackathon with prizes for the best React JS project, the best machine learning project, the best game. There are lots of prizes, and you can check it out by going to findcollabs.com/ open and post your project.

Thanks for being a listener to Software Engineering Daily and I hope you post your cool ideas on findcollabs.com.

[END]