FOSSA 101: Welcome

Imagine your New Learning Thing is out and launching. Maybe a course, maybe a website, maybe another project. Exciting! High fives all round.

Except, imagine this next: right after the launch, you lose a quarter of your learners. 25%. If you were to launch your New Learning Thing in a room of one hundred people, imagine watching twenty-five of them walk out of that room the moment you walk on stage and unveil your Thing.

Wouldn't you want to know why? Wouldn't you want to do something to bring them back?

Now, there could be dozens of reasons why your New Learning Thing doesn't reach someone. Here at FOSSA, I want to focus on five of them - five problems that I know Learning Things face in the digital world. There must be more, but these are the five I want to tell you about.

I also want to tell you a little bit about the learners who have problems with Learning Things for these five reasons.

And finally - I want to give you suggestions for five experiments, five things to try. After you try them, maybe you'll start building your Learning Things differently, or fixing the Things you built already, or helping others build better Learning Things which try to solve these five problems.

So - ready to try and bring learners back to your New Learning Thing?

First, let's hear what they have to say.

Five FOSSA problems

FOSSA Learning began in response to five problems. I saw each of these problems myself - as a teacher or a director of studies, or later when I was selling or building Learning Things myself. Often, I still see these problems when I try to learn something new.

We'll talk about each of them in turn - and about an experiment or solution for each of them, too. But here they are, as a list of problems or voices. Going back to your New Learning Thing launch, imagine a person saying this as they walk out the door.

1. Cost: "I cannot afford this."
2. Software or system requirements: "My machine can't run this."
3. Size: "The download file is too large for my storage / data plan."
4. Confusing language: "I cannot understand this."
5. Lack of accessibility for people with disabilities: "I cannot access this."

Is it really that bad?

You're probably asking yourself, "Wait - am I really losing one in four learners because of these five problems?" There is no easy way to answer this, because Learning Things come in all shapes and sizes, reaching learners all across the planet.

But let's say that, on average, each of these problems is important to 5% of your learners. So for five of them, your Learning Thing is too expensive - for the next five, it won't run on their device… and so on. Add these together, and the math checks out - 25% of your learners are gone.

Sometimes things will get better, and sometimes worse. Let's say you managed to keep the cost down, and you also fixed the accessibility problems. Fantastic! But then you launch your Learning Thing in a new country, where the internet is much slower than in your previous markets. So the five folks who couldn't download your Learning Thing now become fifteen.

Each of the problems FOSSA Learning cares about can become more or less important from one learner story to the next. But here's what I like about FOSSA: it tries to work on all five of them - just in case your learners find themselves in a place where one of these five problems becomes, well… a problem.

And by working the FOSSA way, you are also preparing yourself to tackle another problem. You are preparing your learners - and your Learning Things - for a changing, uncertain world.

And what if it gets worse?

The network of museums and NGOs in your country loses funding for the next three years. They can no longer afford the subscription fees for your Learning Thing. But there are hundreds of learners still on their projects - and they'll be there for at least a year. Now what?

After launching your New Learning Thing, you get a handful of support tickets. They are about a third-party video player which stopped working. You're still looking into this, when the next day, the handful of tickets becomes a deluge. The player app got deleted from an App Store - and now the videos in your Learning Thing are useless. Now what?

It's winter, and the country's power grid is getting targeted by bombers and missiles. That means power outages. And sometimes, that means no internet. Your Learning Thing needs an over-the-air update before the exams. Your sales reps in the country email you, saying, "We tried all night to get this on our tablets. And we're the lucky ones to have internet at all." Now what?

There were seven new chapters added to your Learning Thing, and all went well, until you saw the data and feedback from your teachers on the ground. All the learners in your international schools did well on the old material, but nearly

failed the new material. You decide to check for yourself, and open up a lesson. Three paragraphs later, you realise that you - a grown Milwaukee man with an engineering degree - can't understand most of what you're reading in the high school physics book you're selling. Now what?

The website for your Learning Thing gets a re-design, and this time - after your in-house developers moved on - you paid someone to design it for you. It works faster than the previous one, and looks sleek. One day after the launch, you're on an urgent call with your old developer team. The new website no longer works with screen readers, which upset several long-standing blind customers ("It always used to work, and now it's as if a hurricane's been through it…!") and the company you hired doesn't know how to fix this. Now what?

Your Learning Thing never works in a bubble. It works with real people, in a changing world, and connects to lots of other Things on people's computers, tablets, in their everyday lives.

Something that used to work last week, now stops working. You know this: we all nearly lose our marbles every time our computer or our favourite website updates and moves things around!

Small problems become big problems. People who had no problem with your Learning Thing change devices, or budgets, and have a problem now.

Wouldn't it be better to have a backup plan in mind? It would. That's what FOSSA tries to be.

Let's now take a closer look at each of these problems in turn. And here's a promise: each of them will come with an experiment that will help you start looking for solutions.

Problem 1: "I can't afford it."

You probably heard of a "cost of living crisis". But did you know about a "cost of learning crisis"? It's a thing.

Learners of all ages - from kindergarten kids to university students and adult lifelong learners - are finding it too expensive to learn.

This happens everywhere. In the US, college textbooks are now over 900% more expensive than in 1978 (the average price index rose by about 300% in the meantime). In England, students are finding it hard to live on the support money they receive.

So you see, this isn't just a problem in a country somewhere far away. Chances are, there are people in your home town who aren't learning because they can't find the money for it.

What does it look like?

Sometimes learners still do their best: they buy second-hand materials, or share someone else's login, or photocopy what they can. Sometimes, teachers and librarians help out - I know of schools where a big chunk of teachers' own money goes to making sure that everyone has the Learning Things they need.

But sometimes it's more serious than that, and it just looks like giving up.

Another year away from college because the home budget just won't stretch that far. Another year with no company training budget, because we're still licking our wounds after being stung by that one "training provider" who overcharged us for years. Another outreach project cancelled, because the volunteers wouldn't get training or materials for the workshops.

And the worst thing about giving up is: this sentiment spreads and lingers like a nasty, depressing cloud. Once someone gives up, it's tough to get them to come round again.

Once you've spent some time thinking "I can't afford college", you'll probably keep thinking that all college is equally expensive. Once you've given up on finding good prices on materials to put on your library computers, you'll keep believing that they all cost this much.

So even if good times come around - the home budget looks healthy, the company finances allow for a training quota - you'll still remain convinced: "Nah, it costs too much. Better spend it elsewhere."

Experiment 1: F stands for "Free"

FOSSA is a collection of five experiments in learning design. Each experiment is a search for a solution. You know the first problem by now - here's how I describe the first experiment:

Free: You will provide a fully functional version of your Learning Thing which comes at no cost to the learner.

It's a big ask, I know. Maybe an impossible ask, and that's also fair to conclude. But I would still like to see you try. Here's why, and how.

First of all - if you think that giving something away for free is absurd in this economy, and in this line of work - I would like you to look at the stats on the prices of college textbooks again. 900% more expensive. In the age of computers, and the internet, and Wikipedia, and other such advances. Isn't this absurd, too?

And secondly - I never said that you are to stop making money on your Learning Thing. The "Free" experiment asks that you provide a free (and fully functional) version of your Learning Thing - not that this is to be the ONLY version of the thing.

You must have seen this a thousand times yourself. A programming book you can read online for free - or buy a PDF download. A course which covers all the

material in its free version - but provides video tutorials as a paid bonus. Online training which you can "audit" without paying a cent - and which charge you later, if you wish to buy the certificate. Or a "pay what you can" download.

I've used free and paid versions of these kinds of solutions. And I keep finding new ones online.

Learning Things don't all need to come with a price tag.

A quick note on "fully functional"

There are many ways to make "Free" work as part of FOSSA experiments. I've described some of them above, and you'll find your own ideas. But first, one ask:

Don't cut the vital stuff. That's not FOSSA.

Each of the FOSSA experiments will use the phrase "fully functional" in its description. Here's the easiest way to describe this:

If there was an exam in what I'm trying to learn from your Learning Thing, and if the exam covered 100% of what is in your Learning Thing, it should be impossible for me to fail after learning 100% of your Learning Thing.

Charge for the bells and whistles. Charge for the bonus content. Charge for something - anything - your company does best. I want you to get rich building your Learning Things!

But the important stuff - that's not getting paywalled. None of it. If the exam changes, then - if you're still in the FOSSA game - your "Free" version changes to reflect that.

How would you go about it? Ways of working will change, but here's what I would always want to do if I were to start building a Free Learning Thing.

Making "Free" work - ten FOSSA ideas

Get all the buy-in and permissions you need, early on in the project. Be crystal clear that you'll be launching a free version of your Learning Thing. Be especially clear with authors, contributors, and rights / legal department, if you have them.

Aim to stay 100% clear on all the "fully functional" things you need. Down to every last lesson, paragraph, map, image, recording, photograph. Make it part of someone's job to always know what the "fully functional" part is, and to protect it.

Connected to the point above: getting this right means you're also 100% clear on the "chargeable" bits. If you know what belongs to the free core, then you can also be pragmatic about deciding: no, that's extra, we're charging for that feature.

Learner stories and learner personas are your tools for getting clarity on points 2 and 3 above. Make them as diverse, vivid, and realistic as possible. Update them often: does your "Alice" or "Bob" still use a "PDA"?

If you're new to FOSSA, pick your first experiment project carefully. Especially if someone's money is at stake. Maybe don't make your flagship money-making Learning Thing a freebie overnight.

Connected to the point above: pick the right timing for the experiment. If you're 80% into building a prestigious Learning Thing, it may be tricky to have a "Free is good" conversation, because everyone wants to launch and cash in. But if you're just about to start something new, and smaller, and more speculative - I'm predicting more favourable conditions here.

If you're building more than one Learning Thing, know your list well. Know how your Things perform. Then it's easier to pick a good experiment candidate for FOSSA. Are any of the Learning Things due for an update? Is there less competition for some of them? And / or maybe these learners are no longer willing to spend as much as they used to?

Hiding your "Free" version in a tiny footnote at the bottom of a page is a bad look.

U-turning on a promise of a "Free" version of your Learning Thing, and no longer offering it, is a terrible look: people will get old copies of your Learning Thing from the internet anyway, while bad-mouthing you to anyone who'll listen.

Creative Commons licenses (look them up) allow you to decide how you want your free Learning Things to be shared.

Problem 2: "My machine can't run this."

We will think about each of the following FOSSA problems and experiments in a similar way. After talking about what a given problem can mean to different people, we'll try to work out a FOSSA experiment that makes things better.

The second problem on our list looks complex at first. If someone's tablet, phone or computer can't work with your Learning Thing, then… how could this possibly be a problem you can solve? And how could there be one experiment to try here - if there are so many machines out there, in learners' hands and homes?

It also often looks like a very technical problem. Let's say your Learning Thing was built to work on all latest computers running System A and System B. If someone comes along now, saying, "I can't run this" - and if they tell you they use System C - then the problem appears to be in the system, right?

In fact, there could be many more reasons why people's machines can't run your Learning Thing. Sometimes, even learners who should have no problems - latest

laptops, all the system updates, and so on - can report that they are unable to use something you built for them.

Here's a short list of examples. I think some of them will look familiar to most people.

- Alun bought your Learning Thing to prepare for an exam, which is in 48 hours. He didn't know that it comes with an executable file he must install on his laptop. He would need an admin password from IT for that, and as it's Friday afternoon, the helpdesk has clocked off. He's panicking and furious and wants his money back.
- Beatriz has used Linux for 20 years. She's helped build software for it. And for the past five years, she's made all the computers in her business run Linux. As she's looking through your catalogue, she notices that half of your Learning Things are written for Windows users. "Do you have a Linux-friendly version?", she asks. "Or at least something that doesn't care which machines we use?" You don't. You lose the sale.
- Chika leads training sessions for a group of teachers in his region. Chika is lucky enough to have a new, powerful laptop. But three years ago, Chika was still one of the many local teachers who rely on older machines. Every year, as they pick the Learning Things to use in their sessions, Chika switches from the new machine to one of their colleagues' old, battered laptops. "If it doesn't work on that", Chika says, "it won't work in our classrooms at all."
- Daiya is from a government agency. She can't tell you what she does for a living - it's classified. She can't tell you much about her location, or her learners, or… not much at all. This is what Daiya knows: your Learning Thing won't work for her if it tries to check her location, or run JavaScript, or make any API calls, or as much as scan a port. "Fully offline, fully anonymised," Daiya says. To prove her point, she shows you exactly which of your Learning Things' files fail her tests right now. "The content is perfect, but the delivery won't work." Oh, and her "legal liaison" joins all the calls. Gulp.
- Emil leads your project team. The team spent the past 6 months re-working all your Learning Things, so they work independently of Software A. Emil's team had to do this, after the company that owns Software A decided it was going to use all clients' data to train "AI" models. Your customers were not OK with that, so you had to pivot. Now all your Learning Things no longer rely on Software A. One day, Emil runs into your office with a news article on his laptop. "Guess what Software B just did," he groans.

We don't always talk about these kinds of problems when building Learning Things. I think that's because we've tried really hard to make schools and learning work just like businesses and jobs. And that means we've tried to make

all our machines as similar as possible - and our apps, websites and Learning Things, too. Think about it: when was the last time you saw a website that was different from everything else out there?

I have a problem with that kind of thinking. It's OK for you if you want to run a business that way, I guess. But learning doesn't work like that.

Many Learning Things I'm working on look like they were built for a perfect world - where we all have one of 2 kinds of learning machines, and every year, we buy a more powerful model, without thinking.

Look around you. Even if you lived in a world like that once - do you still think that's the case?

And if not - what could you do for the learners who need something else?

Experiment 2: O stands for "Open"

There is no magical way to solve all the problems we just talked about. But there is a way to start searching for a good solution that works for you and your Learning Things. This is how I describe the second FOSSA experiment:

Open: You will provide a fully functional version of your Learning Thing which uses only open file formats.

An open file format does what the name suggests. Our computers and other machines work with all kinds of files. Files come in all kinds of formats. My machine can run files which yours can't, and vice versa. That's because the company that sells my machine owns the file format, and wants to keep making money from it.

No company owns open file formats. They are managed and shared by standards organisations. And the file formats are free for anyone to use.

If you think that's probably no good at all, think again. A huge chunk of everything you do online, and every website you visit - runs on open file formats.

Look up "list of open file formats" on Wikipedia, and if you've ever built a Learning Thing, I bet you'll recognise a few of these. There are plenty of useful formats for you to rely on.

This means you can build a fully functional version of your Learning Thing with them. A website. An ebook. an animation, or a video. Or an audio file.

Your Learning Thing could be used in almost any internet browser. Or with almost any office software. On new and old computers. On more than just one app.

This won't work for every single learner - but it works for a lot more of them.

A quick note on combining experiments

By now, you may be thinking, "Hold on, can I make the Free FOSSA Learning Thing the same as the Open FOSSA Learning Thing?"

That's the beauty of the FOSSA way - yes, you can, and I think in many cases, you should. But you don't have to.

The FOSSA Learning isn't a pact you sign in blood! If you want to try these experiments, you're free to follow your own ideas and work with what's around you.

There are five FOSSA experiments, and I tried to build them so that they make each other stronger, while also working well on their own.

So you can already see how the thinking you'll do on the Free Learning Thing can come in handy when building an Open Learning Thing. Your free download can use only open file formats - so that more users' machines can work with it. Meanwhile, you're more confident in charging money for the non-Free, non-Open Things, with all their bells and whistles. If something breaks - like in the stories below - you'll always have a backup version to direct your learners to.

But you don't HAVE TO go full FOSSA all at once. Join forces if it works for you, or take it one experiment at a time if that's what you prefer.

Making "Open" work - ten FOSSA ideas

1. A good place to start here is a quick check: which of these formats is your Learning Thing already using? That's free progress - just like figuring out that a word in your language is the same as in a language you're learning!

2. As before - remember the "fully functional" part, and start from there. There are very, very few Learning Things which absolutely need a non-Open file to deliver their message. Map your "fully functional" thing using the open files.

3. The things you're used to, which you're now having to cut from the "Open" FOSSA version - don't give up on them. These are the features you can still market and charge for. Just make sure they are not essential to the "fully-functional" part.

4. You will find that you're giving up on some of the "fancy stuff" which used to come with the non-Open formats. Well, guess what. You still have the text, images, maybe animations, audio, or video. This is your chance to work on making these as powerful, convincing, helpful and beautiful as possible. And after you're done and decide to add the non-Open fireworks back in for another version - then you've really got a strong, Premium Learning Thing.

5. Try working on an older machine as you're preparing this one. We're always used to the latest, sleekest versions - but does your Open idea still

work well on something that isn't the latest model?

6. CSS is an open file format. It's so, so tempting to make the most of this - after all, it's a proven way to jazz up a website… All I want to say is - test what you did, on every device you can lay your hands on. And here's a sneak preview of the fifth FOSSA experiment: "A" stands for "Accessible". Don't get drunk on the CSS power.

7. A variant of point 1 above: check how many of the open file formats your production software can already export to - and check that the exports *actually work*! You're free to use non-Open software, the experiment only focuses on what the learner uses.

8. It is almost impossible to be "trendy" when working on this FOSSA experiment. The latest software, the newest approach - that's always stuff you have to pay for, and something that separates the "haves and have-nots". But it's still possible to be relevant and tasteful, and maybe even beautiful. Train, cherish and respect your writers and illustrators.

9. Preparing elements for your Learning Thing can be done in batches, or automated. Let's say you export a chapter of the Thing - you can tell your software to export the same chapter, in all open file formats, to folder A, and then the non-Open version to folder B.

10. Version control matters here, and switching between open and non-open formats is a nightmare. If you commit to the Open experiment, then my advice is: see if it's possible to have everyone work only with open file formats until it's time to lock them down, and convert the final versions to non-Open formats if you wish. That way, the "Open" FOSSA experiment happens on the way to your business-as-usual.

Problem 3: "I can't download this."

This problem is sometimes related to the previous one. The reasons why our Learning Things break here are also sometimes related to the previous reasons. But it goes a bit beyond software requirements, and the FOSSA experiment / solution will be different here. That's why it deserves its own conversation.

In the heading above, I gave you one of the possible ways to phrase this problem. Sometimes, your learners simply can't find the space on their machines for your Learning Thing.

There are other ways in which your learners can talk about this kind of problem. Let's just look at three of them:

"This takes ages to load." For software, long waiting times are frustrating, but learners will generally sit through the loading screens after they've managed to run your Learning Thing. If a learner is visiting a website, though, they're

much less patient. There are probably other websites they can try, and they'll be hoping to find speedier ones if yours is sluggish.

"You're slowing down my machine with this, what gives?" It's related to the previous voice. In this case, though, imagine that someone is running your Learning Thing alongside other things they're working on. Let's say you're trying to teach them how to use a video editing program. If your Learning Thing eats up too much memory or processor time, then they won't be able to follow along, no matter how good your lessons are. Their machines will just freeze or stop being usable.

"I'm out of data already." I recently read an article written by a scientist on a research station on Antarctica. They described how frustrating it was to have to download people's (oversized) software updates just so they could send a chat message! It doesn't only happen in such extreme situations, though.

Your learners will think twice before downloading your Learning Things if they're on a data plan. And after they get unpleasantly surprised ("what do you mean I'm out of data… how big was this one history worksheet?"), they'll stop downloading altogether.

So how did our Learning Things get disturbingly large? That's probably a story for another time. In short, I think many of us often think this when building our Learning Things: "just another fancy animation won't hurt… just another JavaScript solution on top of what I've got."

These things look innocent, when we look at them one by one. But then you publish your work, and you begin to see: there are hundreds of bells and whistles on your Learning Thing.

They slow down your learners. They make their machines sluggish.

They slow down your servers, too. This costs you - in cloud or server costs, and in search engine rankings (slow pages rank lower!).

So how can you fix that with a FOSSA experiment?

Experiment 3: "S" stands for "Small"

This should be one of the easiest experiments to describe. The opposite of "large" is "small", right? And yet - how small is small enough?

Here is my take on this, and here's how I describe this experiment:

Small: You will provide a fully functional version of your Learning Thing which is no larger than 5 MB (megabytes) in its final compressed size.

If you think 5 megabytes is too small, please note that I was originally planning to make this 1.44 megabytes - the capacity of an old floppy disk. I'm all about symbolism here!

Still, the size limit can feel excessive. How can you achieve it? And should you even try making your Learning Thing that small?

A quick note on doing what you can with FOSSA

A good computer program is going to try to do what it can - if it can't do 100% of a job, it might still be able to give you something you can work with.

A good practical learning project is going to let you, the learner, do what you can - instead of telling you "sorry, you failed" at the finish line, it will be built so that every part of it feels like a success and a celebration.

Please think about a FOSSA experiment in a similar way. It's not a test you will pass or fail. It's an invitation to work and explore, and do what you can.

The "Small" Experiment is possibly the best illustration here. No, I'm not expecting you to make your seven-gigabyte Learning Thing fit into five megabytes. But I am encouraging you to see what you can do, try breaking the Thing down, try working with other experiments to find efficiencies. And, most importantly, to really understand the reasons for having a seven-gigabyte Learning Thing in the first place.

Do what you can. The Learning Thing you'll end up with will still be an improvement.

Making "Small" work - ten FOSSA ideas

1. Build Small first. If you build your Learning Thing around twenty long videos, then there will be no chance to make a FOSSA version that's Small enough. But if you start with the Small version in mind, then you have a better chance of ending up with a Small and fully functional version.
2. Charge for the large. This ties in with point 1 above: make sure that the large files are not part of the Free version, either. I've learned Python from many courses which give you the text and the exercises - and then you can pay more to see the video files with extra hints or Q&As.
3. Compress, compress, compress! There is a difference between an audio file with perfect quality, and a file which is good enough but compressed. Many graphics files will work the same way. Tell your producers to use this Experiment to see how far they can push things with compression.
4. Resize. Not every image file needs to fill the whole screen. Crop or resize until you are happy with the file size, and still have something you can use.
5. Go from videos to animated GIFs. This is often a good option for using Open file formats, too - but it's also likely to save you some file size.
6. Rethink your web framework. For the Open FOSSA Experiment, you would probably need to give up on the usual Javascript frameworks. Is there a lightweight way for you to build the web pages of your Learning Thing? Many of them will be bare-bones, but will do the trick.

7. Make your old Learning Things smaller when you upgrade them. Often, the first idea when refreshing an old Learning Thing is to add more stuff: video! New podcasts! Animations! Sometimes, all this effort is not useful and doesn't get you more learners. How about making a Small Learning Thing instead? At least it won't take up space on the server!
8. Break things down. Maybe you don't need to give up on multimedia. Maybe you can still have your interview audio files. But do they have to come in one big package? What if you broke them down into several smaller Learning Things? They can still work as a series - and will still make sense to learners. But they can download the packages one at a time.
9. Put learners in control. There is a setting in my email software that gives me a choice, each time an email comes with pictures or videos: do I want to download them? Maybe it's possible to do the same for your Learning Thing. So, for example, a learner who knows how a free throw works will skip that video - but will download all videos about rebounds, because they want to learn more about that instead.
10. Zip things up. When it's all done, the Learning Thing doesn't need to just sit in a folder - you can use compression again to zip the folder up. Check whether your "zipping" software lets you use different compression rates - squeeze that file size!

Problem 4: "I cannot understand this."

This part of the story is close to my heart. This is because it's about language, and I have spent my whole adult life working with languages. And it's also because it's about learners who don't feel that one perfect language exists.

It's not all optimism and good news, though. I would now like us to talk about the problem with Learning Things that don't get their language right.

The end result will often be a learner who says something like the quote in the heading above. "I can't understand this," or, "This doesn't make sense to me," or maybe "I've just spent five minutes trying to read this one paragraph."

As before, let's look at some reasons behind this, and some common language-related pitfalls for anyone who works on Learning Things. Then we'll move on to a FOSSA Experiment that might help.

Language is a very personal thing. Every person's language is different. The way they learned their languages, the way they use them - there is no way to find two people who have exactly the same language stories.

And still, language is what connects us, and what makes it possible to understand each other. This kind of contrast is a fascinating puzzle - how can something so intimate be the thing that I share with the rest of the world?

We'll come back to this question in a little while, and look for practical ways to

use our ideas here. Before that, though, let's talk about the problems. What kinds of things can lead to a communication breakdown?

When people talk about learners and language, a place where they often spend a lot of time is a discussion of learners with learning difficulties or learning disabilities. In all schools and colleges, these learners and their needs often mean that language has to change. When these learners speak with their teachers or their class, a change in language happens. When they use a Learning Thing, it also makes a difference when the language is helping them learn.

But learning difficulties and disabilities are not the only reason why a learner can have a problem with a Learning Thing's language.

The second group can be described as learners who are still learning the language that your Learning Thing uses. If you use English, then here's an interesting fact: there are more English language users out there who weren't born in an English-speaking country than those for whom it's their first, or only, language. The bilingual and multilingual English users outnumber the monolinguals!

Any other places worth looking? Well, how about… all of us, in a few decades' time? It's no secret that as we get older, our memory and learning skills change. Just as our bodies do. It's normal for a twenty-year old lady to have a different gym routine than her sixty-year old grandmother. They'll use different exercises, different weights, different set-ups. Why should we not think about the change in the way they learn, and the way they work with language when they do so?

I hope you begin to see this now: there are many places our learners start from, before they arrive at this FOSSA problem. When they say, "I cannot understand this" - it could be because of their learning disability, or because of the fact they're multilingual, or they're an older learner - or any combination of these things and other reasons.

Besides - we've not even talked about all our learners who are just having a bad day! Going through three pages of lecture notes when you're in your ideal learning zone is very different from going through the same three pages in a cold, crowded library after a bad night's sleep. It's the same learner, and the same Learning Thing - but we've all been there, and the effects are not the same from day to day.

Language matters, then. And for some of our learners, language is a problem. Let's talk about experimenting with it, and using the power of languages to start fixing things for these learners.

Experiment 4: "S" also stands for "Simple"

Our fourth FOSSA problem shows us all kinds of learners getting to the same place, connected with language they can't understand. Our fourth FOSSA Experiment uses language to get them to a place where understanding is possible. We're lucky to have language as our tool here - it's a good one.

This FOSSA Experiment can be described and tested this way:

Simple: You will publish a fully functional version of your Learning Thing which a CEFR B1 language user will be mostly (80%) able to understand.

Right. I owe you an explanation here. It's about the "CEFR B1" thing, isn't it?

CEFR means "Common European Framework of Reference". It's a set of tools and measures to describe languages. It's okay to use it outside Europe, by the way - many countries and companies around the world use it.

CEFR has levels. Each level describes what a language user at this level can do. What can they say? What are they comfortable reading, or listening to? How good is their grammar? What kinds of words do they know? When they use a language - what can they easily get done?

B1 is a good place for an "intermediate" language user to be. When I was a language teacher, most of my learners and classes would be this level. And when we learn languages, we often spend more time around this level than elsewhere. So it makes sense to focus our FOSSA Experiment on these learners.

Now, you may be thinking something like, "But my Learning Thing is about This Complicated Topic! There's no way I can make it simple enough!" There are tree things to say here.

First of all, please remember: doing what you can is the FOSSA way. If you try this Experiment, and end up with a Learning Thing whose language is a bit more simple and a lot more helpful - then that's already good work.

Secondly: please note the 80% number above, in the way the Experiment is described. All learners expect to learn new things all the time, and getting out of our comfort zone is okay here. You can use the 20% of your Learning Thing's language to fit the difficult words, the jargon, the technical language. If your 80% is written well, it will help your learners get the point of the 20% better.

And finally: In my 20 years as an editor, producer and translator, I learned that complicated topics don't always need to use difficult language. Many people believe they need to use big words to make their topic look important. And many learners believe that the more fancy words they learn, the smarter they'll appear. Language is important, and it's wonderful - also because it can be fancy or simple, and be just as powerful and helpful.

A quick note on coming up with your own tests

This Experiment is a good place to answer a question you may be asking. It's about the way in which we measure and test the FOSSA Experiments.

Simply speaking - is it OK to come up with a test of your own, when doing FOSSA work? If the measuring tool or test doesn't work for you, is it OK to change it?

My gut feeling on this changes from one Experiment to another, but the answer will always be: "Yes, if the effect for the learner is just as positive, and fixes the same problem."

CEFR levels are a tool I talk about here, because I know and trust them. The CEFR is a strong and powerful way of measuring what people can do with words and languages. And it works for almost every language out there, not just English.

But maybe there is a better way to do it for your Learning Thing, and your language. Maybe you don't know CEFR, and don't want to spend more time having to learn about it. If, in the end, the result is the same as it would be with CEFR - go ahead and change the test.

This applies to other Experiments, too. I'm working with file sizes for the "Small" Experiment. Maybe you want to work with page load times? If the result is an improvement for your learners, then go for it.

In the next section, we talk about accessibility. For this Experiment, the way we measure things online is actually the same all across the world, and anyone who works with web accessibility knows WCAG guidelines. So for the "Accessible" FOSSA Experiment, I would suggest you keep the test, even though you might not like it. WCAG is important and understood.

But things may change in the future. Feel free to update these tests and ideas. Just as long as they lead to the same positive effect for the same learner problem.

Making "Simple" work - ten FOSSA ideas

1. This Experiment will take time. There will be more drafts than you think. This is normal, but please plan for this. If you're on a tight deadline, plan another time for this Experiment.
2. Let experts do the work. You hire a video editor for your videos. You let your artists do your art. But because humans use language all the time, we sometimes fall into this trap: "I'll just proofread and edit it myself." Good editors are out there. Work with them on this one.
3. Use machines and humans to test things. A good computer checker will help you get the words and the grammar right. A good editor, or an interview with a language user, will show you what a learner can try to do with your language. Use both for the right results.
4. Make sure the authors are on your side. Talk to them, and explain what you are doing and why. Make sure they know what this means for the shape their Learning Thing will be in. Not every author will be happy with this.
5. This is a team effort. Someone will write the text for your Learning Thing. Someone else will edit the drafts. Others will proofread this. Maybe you'll need an expert to review it. And it's good if your marketing people are talking the same language, too. Keep building a team that will understand

what this Experiment is about.

6. It's not just about words and grammar. There are many parts to this Experiment. How long are your sentences? How big are your paragraphs? Are your headings useful? Are you helping the learner make sense of the text?
7. Prepare a style guide and a tone of voice. If you haven't got these documents yet, then this Experiment is a good chance to start working on them. Get your editors and marketing / communications teams to work on these. And please, pretty please: make these two documents simple and useful, too. Busy people will be using them on a daily basis.
8. This Experiment is about consistency. You can use the 20% of your Learning Thing to be creative, unpredictable, fresh… That's good, and all Learning Things need this to be interesting. But eight times out of ten, your learners will be happy to see similar things done in a similar way.
9. Do your research, but feel free to mix things up. English already has some ways of making itself simple: there is Plain English, Simple English, and other similar ideas. Many of these solutions are useful - but it's not always great if you only use one of them. Language changes, and your Learning Things are unique. Feel free to mix things up.
10. "AI" is a seriously bad idea here. Large language models cannot give you consistency. They will give you language that is different from day to day. And they will change in the future. This one is about control and accuracy. "AI" is bad at both. Sorry.

Problem 5: "I cannot access this."

We have one more problem to discuss. One more reason why people sometimes give up on your Learning Thing. And this one - well, there are many ways to get this wrong, but also many ways to get this right. This time, we're talking about digital accessibility.

In a way, everything we spoke about before is part of this big idea called "being accessible" - because every time you try to fix a problem with your Thing, more people get to access and use it. But this time, when I say "digital accessibility", I mean making your Learning Thing work for people with disabilities.

Let me tell you something that not everyone knows about me. I cannot watch anything that has flashing lights. Often, when I'm at a concert or watching a music video, I need to close my eyes. And sometimes, if a sci-fi movie or a gig uses a lot of these flashing lights, I need to leave. Otherwise - 99.9% of the time - I am just a normal guy. But if you decided to use a stroboscope effect in each one of your videos - or on each of your websites - then I'd have to look for somewhere else to learn.

This is just one example of "digital accessibility", and many of them you will be more familiar with. Today, the internet is a wonderful thing for many people

who cannot see, or hear, or use their keyboard or mouse the way I do. From your operating system to your websites, and video games - things are so much better these days than they used to be.

At the same time, though, everyone who builds Things online is still learning. We discover new ways in which people use stuff on the internet, and we learn more about how some parts of our Things are broken for them. This is a process, and it never stops. You don't get to say, "that's it - everything we made is now 100% accessible for everyone!", because you will discover a new problem pretty soon.

That is partly because internet users with disabilities found their confidence. They know there are parts of the internet that are still broken, but they have also explored the non-broken parts. More and more people get online each day - that also means people with disabilities.

And partly, that is because good Learning Things will try to be accessible for as many learners as they can, in as many moments as possible. Here's what I mean by that. Some learners live with their disabilities for a long time (for example, if you haven't been able to use one arm since kindergarten). Some others are living with a disability now, but not for long (for example, if you broke your arm and can't use it). And some of them are maybe only finding that this disability depends on a situation (for example, if you are a new parent and are holding a baby in one arm for two hours while you learn on your laptop). Trying to make your Learning Thing accessible for each type of user means you're never really "done".

So - when your learners pass on your Learning Thing because they cannot access it, they may mean a lot of things. Doesn't this make that problem the hardest to fix?

Don't be worried - this one is something the internet has been working on for a while. Onwards - to our last experiment!

Experiment 5: "A" stands for "Accessible"

Internet looks pretty much the same around the world. And if you're living with a disability, often you share similar internet problems and difficulties with others in different countries. This means that everyone who tried to fix internet problems for people with disabilities was quickly able to come up with a set of good ideas. This is what the last FOSSA experiment is about:

Accessible: You will publish a fully functional version of your Learning Thing which meets the AA level of WCAG web accessibility requirements.

Let's unpack some of this. "WCAG" stands for "Web Content Accessibility Guidelines". It's a set of rules ("requirements") for people who make things on the web. These come in levels, and the levels are marked by "A" letters. There are three levels, and "AA" is the middle level ("AAA" is the top level).

There are many ways in which you could measure how accessible your Learning Thing is. But the WCAG is what almost everyone on the internet follows already. In several countries, official websites need to meet these requirements already. So I was in no doubt that this is the most useful test to recommend for our last experiment.

The next reason for WCAG is that it comes with its own set of good techniques. In other words, the people who developed the rules can also show you some good tools you can use to build a Learning Thing which follows the rules. It's always more useful to work with something that offers solutions alongside questions, right?

Finally - I like that WCAG offers some flexibility thanks to its three levels. I picked the "AA" level for FOSSA, because I know that there are several "AA" rules which learners will definitely like to see followed. But if you look at the "AA" level and feel daunted, then you don't have to give up. Instead, you can start by aiming to meet the "A" level rules. It can already be a big improvement on how accessible your Learning Thing is. The next levels build on the previous ones, so you won't be doing the work all over again, if you decide to work on higher levels next.

A quick note on integrating FOSSA experiments into your main work

When we talked about the problems and experiments for FOSSA, I always made sure to remind you of one idea. Your "fully functional version" didn't need to be the only version of your Learning Thing. So you could have a "Free" version and a paid one - remember? Or a "Small" version and a full-sized download. As long as they were fully functional, that was OK by me.

But at some point, I think it would be great to think about building the FOSSA experiments into your "main" Learning Things. To build just one thing, that happens to also solve as many FOSSA problems as you can.

I'm saying this now, because this last experiment really is the one I'd like you to consider for your "main" Learning Thing. If we were brainstorming this in a room or in a call, I would probably understand your concerns if you told me, "Vic - we can't make this one Free just yet." Or - "It's tempting to make the main product this Small - but we still need the video". But I would always insist on making your "main" Learning Thing accessible from the start.

There are hundreds of reasons. Lack of accessibility is a problem for your legal team in some markets. It's always a bad look, and can make you lose your good name. And (this one is most important for me) trying to get accessibility right puts you on a great path. It makes everyone who builds your Learning Things more open to what your learners need.

So if you decide to build just one FOSSA experiment into your "main" Learning Thing from the start, make it this one.

Making "Accessible" work - ten FOSSA ideas

1. Get everyone on board. Accessibility is everyone's job. Each person in your company is very likely to be old and frail one day, and maybe will care for others who live with disabilities. Their interest in making things accessible is for themselves, and their loved ones.

2. Be clear on your legal requirements. Maybe you already need to be passing some accessibility checks because of where in the world your Learning Things work. Make sure you know what these are, to avoid extra effort (or last-minute panic).

3. Build accessible things from the start. Which do you think is easier: creating 100 accessible graphics each year, for 10 years in a row - or fixing 1000 inaccessible graphics at the end of Year 10, when everyone realizes you've got a problem?

4. Read the rules and techniques, together as a team. This means that people can suggest their ideas for fixing your Learning Things, or for checking which parts already work in an accessible way.

5. Have accessibility champions. If anyone is trained in helping others build accessibility into your Learning Things, make sure they have the power to help out.

6. Start with the little things. A quick read of the WCAG rules makes it clear: some things take five minutes to fix, and solve a huge chunk of people's problems. These are a nice team win to get everyone excited about accessibility.

7. Bust some myths. This is not just a "box-tick" exercise. It's not a soulless paperwork job. In every team, there can be someone who believes accessibility is a waste of time. Learn to work with them to change their minds.

8. Could your teammates or learners help? Perhaps you have learners who left good (or bad) feedback on your Learning Thing. Or maybe someone on your team would have problems using what you built, because of accessibility. Can you ask them to share their stories?

9. Aim for "always in progress", instead of "perfect". Everyone is always building new things on the internet. This means you will build new Learning Things, and they will not be 100% accessible. That's OK. This is a journey, and you don't have to be perfect - but you should definitely keep doing the work.

10. Cut out inaccessible gadgets. If something looks flashy, but can't be fixed to be accessible - think about how badly you need it. Using this in the future could mean that you lose business - because your Learning Thing can't sell in some countries, or because people will prefer someone whose

Thing they can access. Maybe it's better to fire the inaccessible gadget today, and start building something that works?

Checking my math: numbers, stats, bad news, good news

I keep referring to losing "a quarter of your learners" because of the five FOSSA problems. And the solutions or Experiments I'm suggesting are the work you'll do to bring the learners back - to make your Learning Thing work again, for the 25% of them. So it's fair that we talk about the numbers for a little while.

The question you are maybe asking yourself is, "Is it honestly that bad?"

Am I really going to lose one in four learners because of these five problems? And if this guy says so - where are his numbers?

Here's the math. I am writing this in 2024. Things may change in the future, but here is where we are.

First of all - just to keep things simple, I am saying that each of the five FOSSA problems makes 5% of your learners give up. So five times five gives me the 25% we're talking about.

In real life, things will never be this obvious. But I want this to be a story about your Learning Things and the people who use them - not just about the math behind the problems. So I keep things simple here.

Now, then - where are the numbers for each of the problems?

Free: how many learners cannot afford your Learning Thing?

In the UK, in 2022/2023, 18% of the people were in absolute poverty. (https://commonslibrary.parliament.uk/research-briefings/sn07096/)

In the US, in 2022, the official poverty rate was 11.5%. (https://www.census.gov/library/publications/2023/dem
280.html)

Across the world, in 2022, 8.98% of people lived on less than $2.15 per day. (https://ourworldindata.org/poverty#key-insights).

Conclusion: I told you that you're likely to lose 5% of your learners because they can't afford your Learning Thing. The real number is likely to be much higher. You will be able to help your situation with good marketing and sales - but even if your Learning Thing goes to learners in countries such as the UK or the US, you will still have this problem.

Open: how many learners cannot make your Learning Thing run on their software?

In the US, in 2022, between 20 and 30% of all people said that they don't own a computer, or a smartphone, or home broadband. About half of them didn't have a tablet. (https://www.pewresearch.org/short-reads/2021/08/19/some-digital-divides-persist-between-rural-urban-and-suburban-america/)

In the UK, in 2022, 10% of device users didn't go online at all. About 11% of them didn't know how to turn on their device. (https://digit-research.org/data_commentaries/measuring-digital-exclusion/)

Across all developed countries, in 2019, 13% of people didn't have internet access. (https://www.internetsociety.org/blog/2022/03/what-is-the-digital-divide/)

Across the world, in 2024, 3.91% of windows users use a version that's no longer supported (https://gs.statcounter.com/os-version-market-share/windows/desktop/worldwide). 13.44% of all Android users use an old version that hasn't been updated in at least 1.5 years (https://gs.statcounter.com/os-version-market-share/android/mobile-tablet/worldwide)

Conclusion: it is very difficult to find definite numbers for this one. People use different machines to do different things. But even in the developed countries, there will be more than 5% of learners who will have problems with your Learning Thing. Often their machines aren't even old - they're just not the ones you thought of.

Small: how many learners cannot download your Learning Thing?

In the UK, in 2021, at least 18% of all households struggled to pay for the data plan for at least 1 device, and 5% of all households found it hard to afford broadband internet. (https://www.ofcom.org.uk/siteassets/resources/documents/research-and-data/multi-sector/affordability-of-communications-services/2021/affordability-of-communications-services-summary.pdf)

Across the world, in 2021, 1GB of mobile data became less affordable than in the previous year (https://adi.a4ai.org/extra/baskets/A4AI/2021/mobile_broadband_pricing_gni.php).

The size of a typical page downloaded to your desktop device increased from 467 kilobytes in 2010 to 2643 kilobytes in 2024. (https://httparchive.org/reports/state-of-the-web)

Conclusion: across the planet, we keep making bigger pages and keep making our broadband more expensive. 5% is a number that fits countries like the UK right now. It's worse in other countries, and it will be getting worse everywhere in years to come.

Simple: how many learners cannot understand your Learning Thing?

In the UK, in 2024, about 2.1% of adults have a learning disability (https://www.mencap.org.uk/learning-disability-explained/research-and-statistics/how-common-learning-disability).

In the US, in 2002, about 40% of people aged 65 or over had age-related memory impairment (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1123445).

Across the world, almost 95% of all English speakers weren't born in an English-speaking country (https://www.nature.com/articles/d41586-023-02320-2).

Conclusion: most of the learners on Earth speak more than one language, and many of them will have problems with difficult words - English or not. Making your Learning Thing simple helps your older learners, and your learners with learning disabilities, and your multilingual learners, too. The number is far greater than 5%.

Accessible: how many learners with disabilities cannot access your Learning Thing?

Across the world, about 16% of people live with "significant disability" (https://www.who.int/news-room/fact-sheets/detail/disability-and-health)

This number probably does not include people with temporary disability (such as a 4 week recovery after eye surgery) or situational disability (for example, having to sit in a noisy room with a child climbing over you).

Across the world, over 95% of the most popular websites have an accessibility problem. (https://webaim.org/projects/million/#wcag)

Conclusion: in real life, the number of learners who can't access your Learning Thing is higher than 5%. And it's really hard to find a Learning Thing which got built with no accessibility problems.

Bad news, good news: making sense of the numbers

There are so many ways to look at all these numbers. So many learner stories I could tell.

I want to spend a moment on just two of these ways: I want to call them the Bad News Spiral and the Good News Spiral.

The first story is the Bad News Spiral. Maybe you had an idea about this already, when you were reading the numbers. In the Bad News Spiral, there is never just one problem for one learner.

Instead, one problem connects to another problem. For example - your computer breaks. So you have to budget for a new one. So you can't afford the data top-

up for three months. So you fall behind on your video assignments. So you are stressed and find it hard to do the reading…

Or: you are recovering from chemotherapy, but you still log on to your online lectures. Your arms hurt when you try to type up the notes. The pain and the brain fog makes it hard to understand what the lecturers are saying. And the hospital wi-fi keeps cutting out…

You can probably come up with more of these Bad News Spirals, just by looking at the numbers I shared with you.

The Good News Spiral is the opposite of this. We already saw how FOSSA Experiments work on one problem at a time. We also saw that a FOSSA Experiments can connect with each other.

This is when you begin to build your FOSSA Experiments to help learners with more than just one problem.

For example - you decide to make a Free package of your Learning Thing. You discover that the best way to do it is to save articles as plain text. You go on to save images and audio files in two more Open formats. Then you look at how Small the package is - what a nice surprise!

Or: you try to rework your Learning Thing to make it more Accessible. You work with an editor who writes alternative text for your images. You read her work, and really like her style of writing. She has two more weeks of availability - so she uses this time to work through the text on the web pages, too. As a result, the text is more Simple. She asks you about the file formats to send back all her work - you use this as a chance to ask for Open file formats.

It's likely that problems with your Learning Thing will be important to more than 25% of your learners. But it's also likely that when you start working on them - when you try a FOSSA Experiment or two - you will start helping more learners, all at once.

Congratulations, next steps, and thank-yous

This is the end of "FOSSA 101". Well done on making it this far!

I hope this was useful. There are many ways to build Learning Things, and every time I worked on a new one, I was excited to see what problems and experiments I will find. I hope that FOSSA helped you find some good problems to fix - and that it gave you an idea on which experiments to use to fix them.

So now what?

I cannot answer this for you, of course. Your learners, your teams, your users - these are the people you should ask. The next step should take you to them - and to their ideas on how to make your Learning Things even better.

But can leave you with some useful questions. Even better - I can give you a checklist filled with them. No, wait - two checklists! Aren't you lucky :)

The first checklist is a useful starting point for trying to build a FOSSA Learning Thing from scratch. Good for starting new projects. The second one is about projects and Things you built earlier and want to fix now by making them more FOSSA.

Feel free to mix and match these, use the questions in any order, rewrite, add, remove a question or two… They're yours to work with, so make them work.

Checklist 1: Building a new FOSSA Learning Thing

[ ] Did you talk about FOSSA with your team?

[ ] Did you make a decision together about which FOSSA experiments to work on?

[ ] Did you check your budgets and timelines - can you afford to work on your FOSSA experiments?

[ ] Did you agree on the objectives and tests for all the FOSSA experiments you'll work on?

[ ] Did you make changes to your project plans to make the FOSSA versions easier to prepare?

[ ] Did you talk about FOSSA with your authors, suppliers, and partners?

[ ] Did you agree with any sales, marketing or communications people on the best way to talk about your FOSSA versions?

[ ] Did you find all the tools, techniques, documents you need to work on the FOSSA experiments?

Checklist 2: Editing a published Learning Thing to make it FOSSA

[ ] Did you talk about FOSSA with your team?

[ ] Did you agree on which Learning Thing to edit - and why it's good for a FOSSA approach?

[ ] Did you make a decisions about which FOSSA experiments to work on?

[ ] Did you check your budgets and timelines - can you afford to work on your FOSSA experiments?

[ ] Did you agree on the objectives and tests for all the FOSSA experiments you'll work on?

[ ] Did you make sure that your FOSSA experiments won't break anything important in the existing Learning Thing?

[ ] Did you decide on how to connect the FOSSA fixes with the existing Thing? Will they be separate or together?

[ ] Did you make copies of the existing Learning Thing for backup?

[ ] Did you agree on the best way to divide work between the FOSSA fixes and any other fixes?