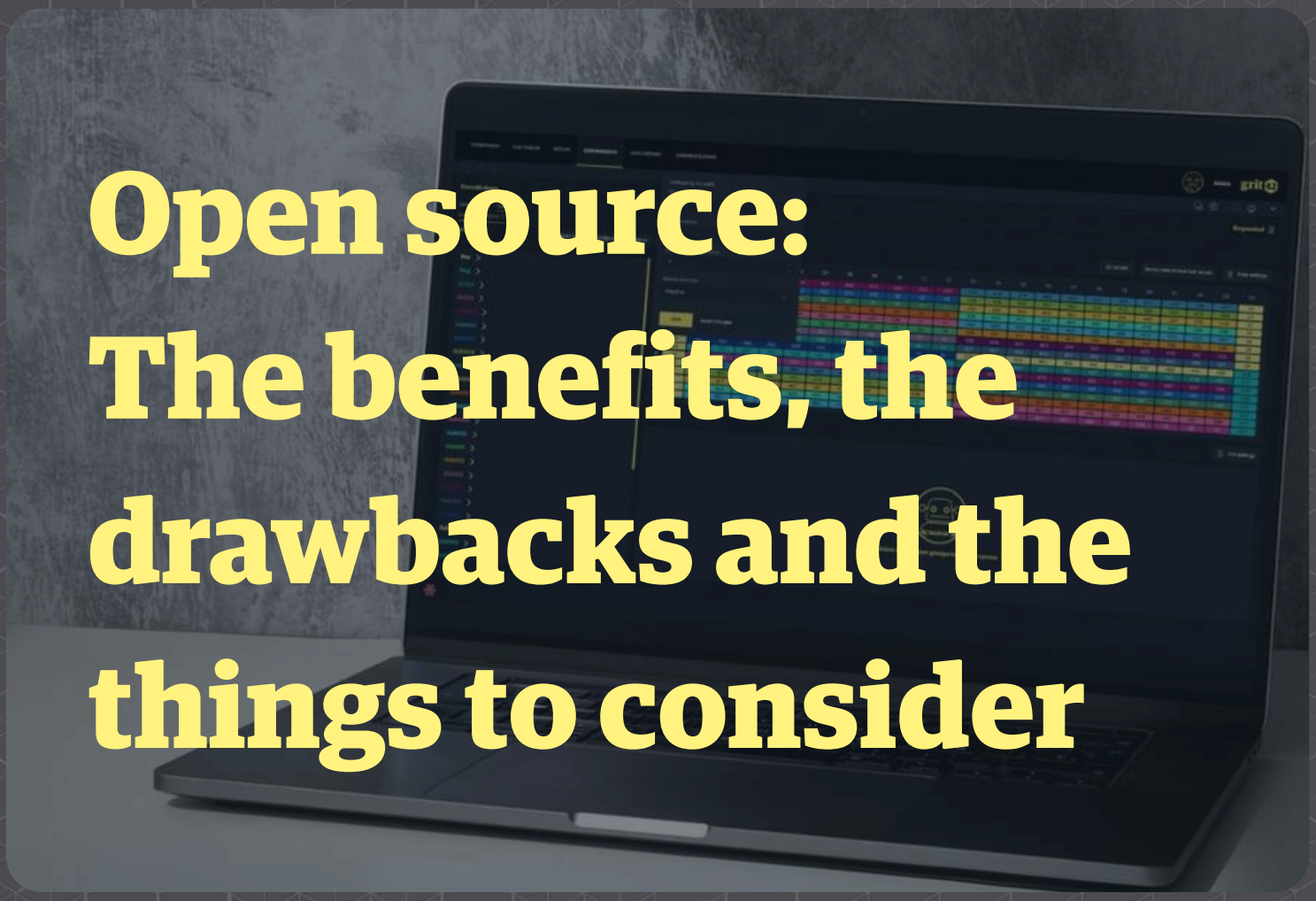


Open source: The benefits, the drawbacks and the things to consider

4 biopharma data management and informatics experts share their views on using open source software in drug discovery

grit42

At grit42, we have decided to change direction and offer our Scientific Data Management Platform, grit, as open source.

We have done so, because we believe it can prove valuable to both a broad range of pharma and biotech companies as well as academic institutions. We have, ourselves, benefited from building on top of a number of open source technologies, so it also seems like a good way to pay back to the open source community.

Of course, we also see it as a stepping stone to development projects and service agreements, which we hope, in the end, will turn into a sound business case for us.

At the same time, we still sell proprietary software, too, so we have one leg in each camp, you could say.

So, instead of us laying out the pros and cons of open source software, we asked 4 people in drug discovery to share their experiences and views on the benefits, drawbacks and important aspects to keep in mind when considering open source software.

In this ebook, you'll meet:

Dr. Frank Bringezu

Associate Director,
Chemical & Preclinical Safety,
Merck Healthcare

Dr. Piotr Klimkowski

Associate Principal Scientist,
Sixfold Bioscience

Teague Sterling

prev. Director of Computational Biology,
BioMarin Pharmaceutical

Morten Lindow

Professor at Dept. of Drug Design and Pharmacology (ILF),
University of Copenhagen and leading the Center for
Pharmaceutical Data Science Education.

What are the most important aspects for you when you consider new software for a project?

> It must be fit for purpose, of course, and it should be adaptable. I mean, we are in an environment that requires a certain level of flexibility as we have different internal needs, and we also have to meet regulatory compliance. So this is a very important aspect.
>
> – Frank Bringezu

> I think one of the most important things is extensibility. Especially in the research world, things need to be tinkered with, adjusted, or added on to. And this can become really, really time-consuming projects when you have to work exclusively through a vendor or you have to use a really limited skillset for it.
>
> Combined with the ability to integrate and an attractive price point, those are the things that can make me excited about a new platform.
>
> – Teague Sterling

> It's often overlooked, but an important point is how easy the software is to use. Whoever is going to use the software needs to be comfortable with it.
>
> There is a lot that goes into that: the interface, the usability, the flexibility, and so on.
>
> – Piotr Klimkowski

What do you see as the main benefit of using open source software?

I think the main benefit comes from connection - connection to a community that keeps the product in line with the state of the art and the thought process of the field.

One of the challenges you will find with a closed solution, is that the team using that can be very productive and develop their own really incredible methods. But those become very disconnected from the way that the community is going and maintaining them becomes harder and harder.

So if you have a community in hand that's constantly evolving this thing, I think you actually end up with a more sustainable product.

- Teague Sterling

In general, in the open source environment, collaboration and innovation are high on the agenda. It's about sharing code across industries, and there is a broad range of expertise and innovation that we can build on.

This collaborative environment can actually help us accelerate drug development and discovery more and lead to even faster drug development in the end.

- Frank Bringezu

What do you see as the main benefit of using open source software?

Open source can become elitist in a good way in the sense that we can work with it ourselves. There's not so much process BS when you don't have to pay and make contracts and service agreements.

You have this tinkering opportunity, so you - if you have the skills in your organisation - can do whatever you want with it.

- Morten Lindow

At Sixfold, we've looked through a lot of software, and some of it was amazing for very specific, well-defined processes. However, where we are building new processes and assays, we need flexibility. It's hard to justify expensive software that's single-use and even more so to spend across multiple expensive platforms.

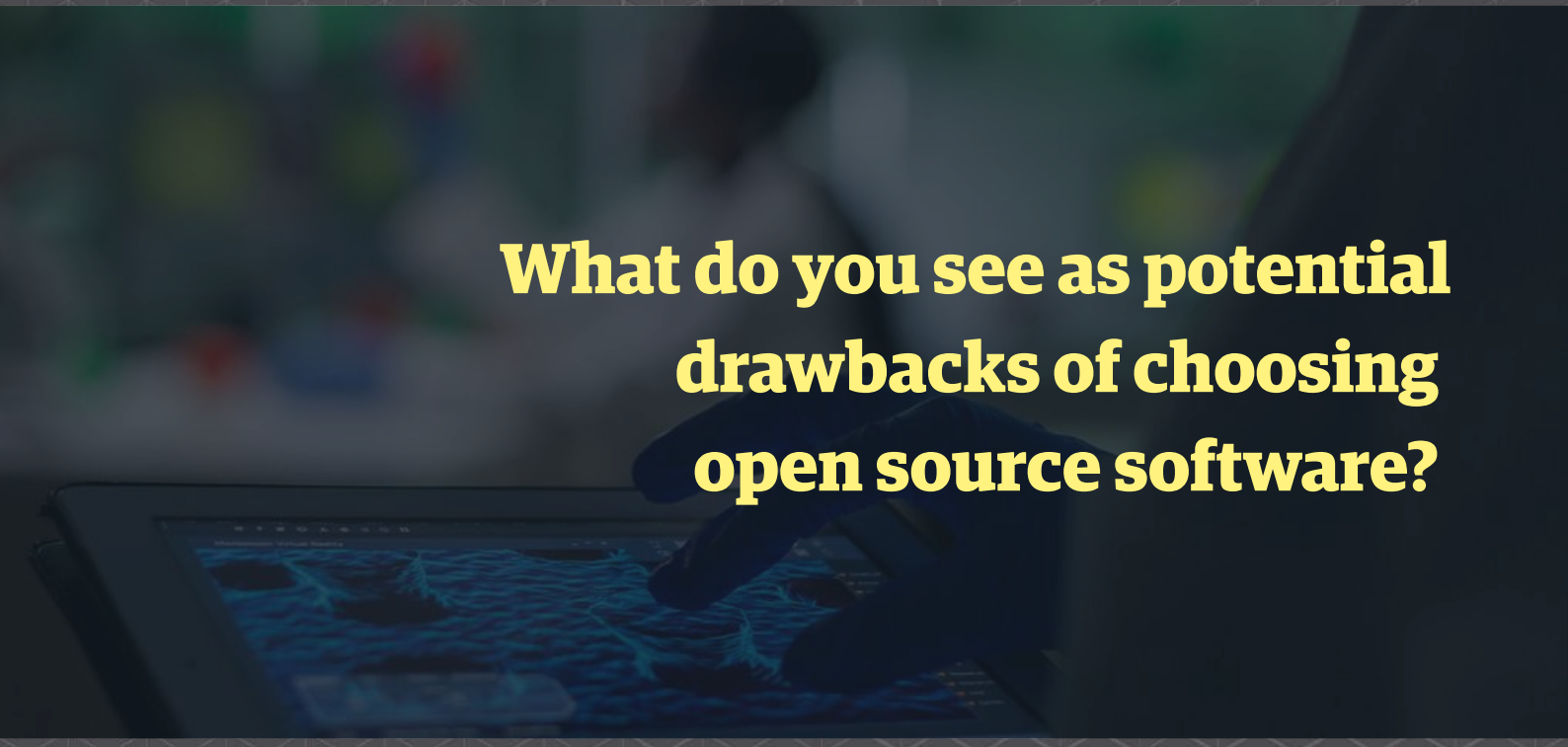
The advantage of open source is that you have a layout and a backbone that you can repeat and build on.

- Piotr Klimkowski

How important is the fact that open source software is free?

> For big pharma, the financial aspect is probably not the most important part, but it can be for smaller companies, and especially for academia. It's great that you can just download and try it out; it lowers the barrier to getting started.
>
> And even though the funding might be there in big pharma, sometimes the whole RFP circus that you have to go through sometimes makes you just give up. It can be very cumbersome to get through the procurement process, and you suddenly get a lot of internal stakeholders who don't understand the business need but have their enterprise view on things, and that brings everything to a stall.
>
> - Morten Lindow

> Cost efficiency is a very important factor because often, we are a bit under pressure to justify expenses.
>
> And if we can reduce costs associated with licensing and development, this is for sure very important from a business perspective. It means we can allocate resources to other critical areas.
>
> And I mean, we should discover new drugs, not develop software.
>
> - Frank Bringezu



What do you see as potential drawbacks of choosing open source software?

The benefit of buying a customized solution is that you have customized support. You can ask the vendors when you need something or if something breaks. If you ask a community, you might get an answer, but you can't be sure.

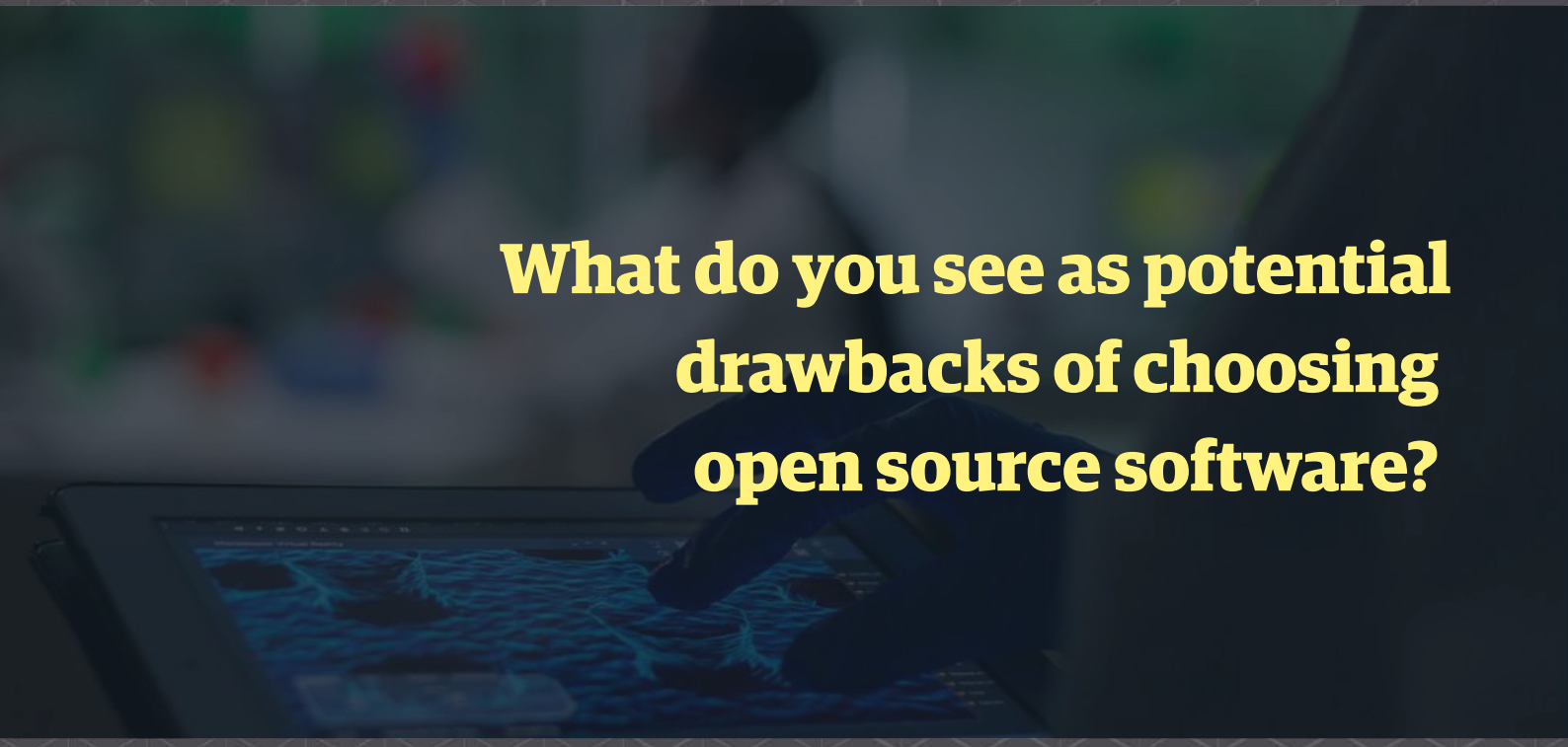
This is critical and there's always a need for a service support of some sort so you have somebody to talk to and get an answer in a given time frame. So it can be more demanding on the expertise that you have in-house.

- Frank Bringezu

If we're developing something in-house, the timeline can be a challenge as we're optimising for development speed. We have built software in-house for automation platforms, and we are happy about the result.

But you need to be comfortable with the set-up time – and don't always have that. Sometimes, you just need the solution quickly. So, this is not necessarily a drawback, but it depends on your priorities.

- Piotr Klimkowski



What do you see as potential drawbacks of choosing open source software?

It’s the risk of enterprise-type support not being there. That's not always the case, but it's a risk.

Also, you're working within a community which in general is a good thing but can also be a detriment if you need the product to go in a very specific direction.

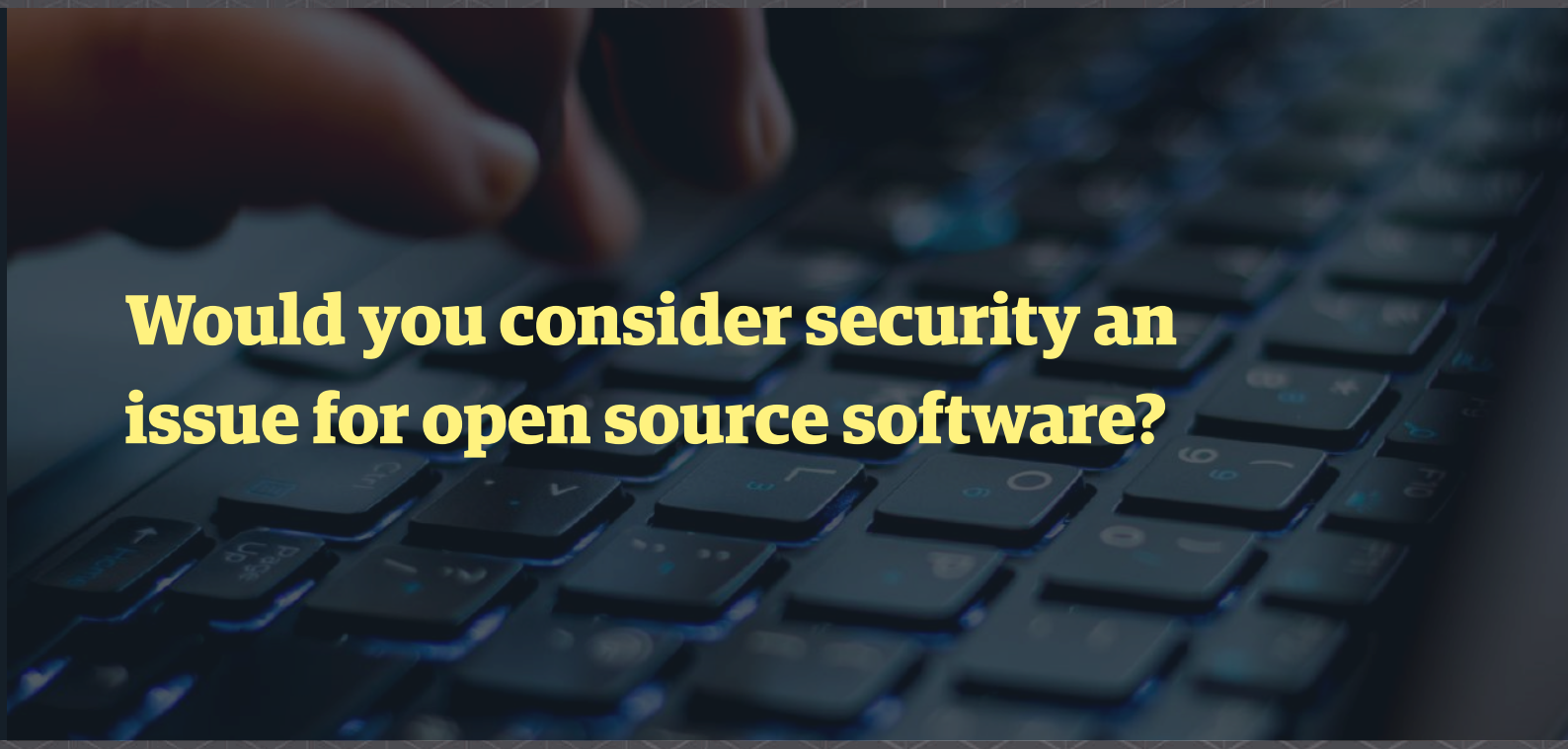
If you want something specific developed, this can be solved through a contract, but if what you need is sensitive from an IP perspective, there can be an internal organizational barrier to sharing that you need to overcome

- Teague Sterling

Can we get support for it? Is it future-proof enough if we store a lot of valuable data in it? But that's not really a question of whether it's open source or not, but more a question of the supplier you choose.

You have to make sure that you can get someone other than those who developed the product to provide service for it.

- Morten Lindow



Would you consider security an issue for open source software?

> It's an argument I have heard a lot in the corporate world. I don't agree with it, but it was something that always needed to be addressed. I personally prefer open source for security because I can vet it myself. If I have a hunch that something is not working the way that it should I can actually go and look.
>
> I think it's a fallacy that closed source software is simply more secure because it's closed source. It doesn't really mean anything except that fewer people are actually working on trying to fix it.
>
> - Teague Sterling

> I would definitely consider the security of the code. If it's written by an in-house non-expert or a community member, you can't always rely on its security. I'm not thinking about data breaches but more about the execution – for example, if something is written by a scientist who is not an experienced coder, there is a greater opportunity for an honest mistake. That would be on my mind, so we look at cross-validation of the software or sufficient testing.
>
> - Piotr Klimkowski

> In principle, it's safer, I believe. There's nothing hidden. But of course, it still has to comply with the standards you have, and it can. It just needs to be tested and documented etc.
>
> - Morten Lindow

How important is the community around open source software to you?

I would look at that quite a bit to make sure it's not a dead product. I would typically look at how it looks on github and how many people are contributing to it, how many forks have been created, etc. The more activity the better.

Of course, bearing in mind that it may be a niche product, which naturally won't be widely adopted.

- Morten Lindow

We definitely look at what projects were run with this specific software. So, if we cannot find a great deal of structured documentation and engagement, that's not the best sign and, in my experience, makes development for your particular application much slower.

- Piotr Klimkowski

If there is no active community, it’s definitely harder to work with. But still, if there's existing work that you can join and collaborate with, I think you have a better chance of being able to make your work sustainable than if you were to just go and create something on your own.

- Teague Sterling

www.grit42.com