Geographic Information Systems

Dr. Bharath Haridas Aithal
Ranbir and Chitra Gupta School of Infrastructure Design and Management
Indian Institute of Technology– Kharagpur

Lecture – 59
Pros and Cons of Open Source

Hello. Namaste. Welcome back to the course on Geographic Information Systems. In the previous class, we have learnt about different GI software's and libraries etc. We have looked at how graphs performs, how OGR and GDAL as powerful tools or libraries have provided in any of those software and we have looked at QGIS and FAQ stat. So, all of these let us look at what is the pros and cons of an open source software. So, let us understand that in terms of comparing both of the open source and proprietary software.

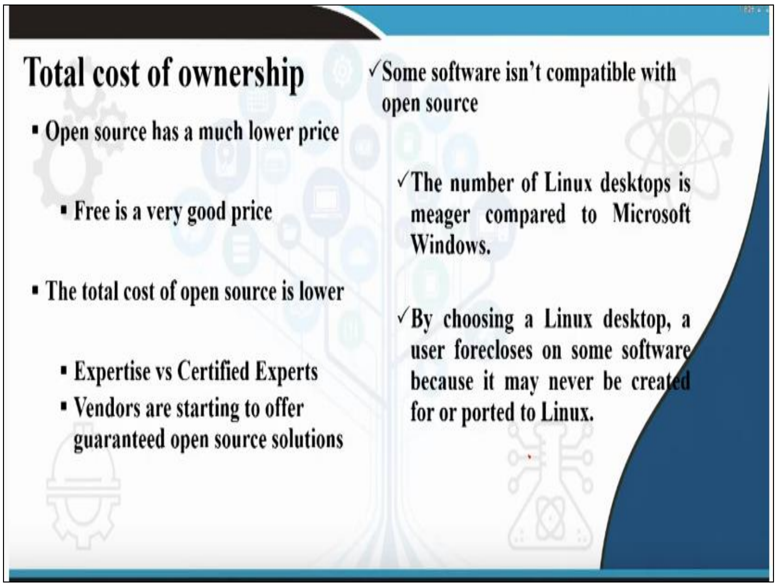
(Refer Slide Time: 01:07)

CONCEPTS COVERED

- Open Source
- Cost of Ownership
- Features and quality
- Principles and rights

So in today's class, I would look at an open source as an open source and how it is useful. Then, the cost of ownership when you have an open source software versus proprietary software, the features, the quality in an open source software and proprietary software, the principles and rights, as a user, as a developer and as a manager. Okay. In an open source software, what are your principles, what are your rights when you compare it with the proprietary software. That is what we would see in today's class.

(Refer Slide Time: 01:47)



Total cost of ownership

- Open source has a much lower price
  - Free is a very good price
- The total cost of open source is lower
  - Expertise vs Certified Experts
  - Vendors are starting to offer guaranteed open source solutions

✓Some software isn't compatible with open source

✓The number of Linux desktops is meager compared to Microsoft Windows.

✓By choosing a Linux desktop, a user forecloses on some software because it may never be created for or ported to Linux.

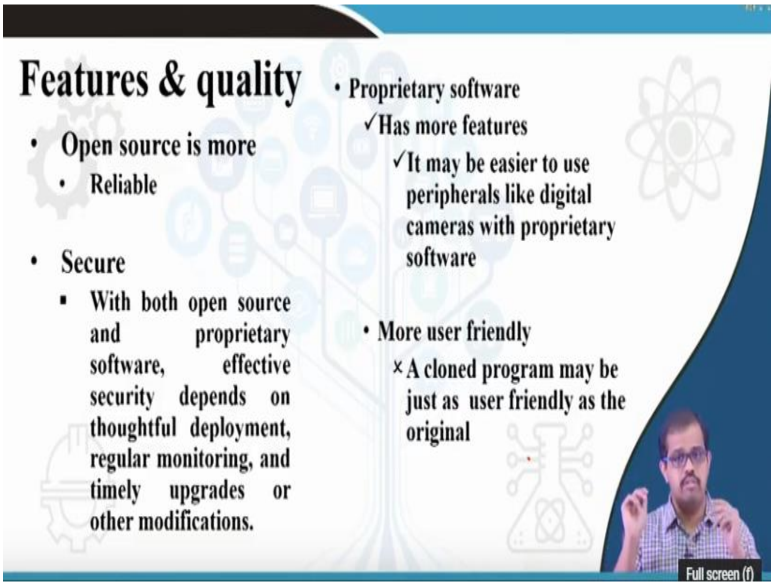
So when you look at the total cost of ownership, open source has much for a lower price. As I said, it may be just an distribution price, otherwise most of the one is software free. Okay, when I say the lower price, it is the free is a very good price normally. So, that is the cost of ownership. The total cost of open source is much lower. One is you have Expertise versus Certified Experts.

So, that you have look at in terms of whenever you are looking at an open source, then you have vendors are starting to offer guaranteed open source solutions. That is where the cost that I am trying to say here, though I said only distribution cost are cut, but you have a vendor who is actually guarantying solutions through an open source, so then you have a small cost or a certain cost that is associated with any of the software or any of the applications that is being developed. So that is where you may have some cost, but when you look at the proprietary software comparatively, it is very less.

It may be 1/10th of the cost that the proprietary software has. Some software are not compatible with the open source software, for example, number of Linux desktop is very less when compared to that of Microsoft Windows today. Okay, that is there. But when I say some software is not compatible with open source, it means to say that software has not been built for any of the open source software.

So, that is otherwise because of the open source software it is not compatible. By choosing a Linux desktop, a user forecloses on some software because it may be never be created for or ported to Linux. That is what I said. It is never created for Linux. That platform is not created. But, when you look at GI software, most of the software that has been developed, open source software has always been started with the Linux. Then, it has been ported to Windows or Mac etc. Without Linux, there has been no development as far as GIs concerned.

(Refer Slide Time: 04:09)



Features & quality

- Open source is more
  - Reliable
- Secure
  - With both open source and proprietary software, effective security depends on thoughtful deployment, regular monitoring, and timely upgrades or other modifications.

- Proprietary software
  - ✓Has more features
    - ✓It may be easier to use peripherals like digital cameras with proprietary software
- More user friendly
  - ×A cloned program may be just as user friendly as the original

When you look at features, the first thing that any of the user will ask is the feature. What are the features that are there in open source, how the development happens, what kind of development is there, and how well it can compare with the features of that of proprietary software. So, the first thing that I would assure you is that open source is more reliable, always reliable.

It has backed community, it is actually helping you out, it has bug trackers, it has people who are patching it. That development is going on and on and on. So, whenever you report something, it is always a positive outcome of what you have reported. Then, it is more secure, in terms of whether you are using GI service.

So we have emphasized security as a very important measure. So, when you look at open source software, it is highly secure. When both when you are looking at it, effective security through deployment is done in an open source software even before the code is for its effective usage is done whereas in regular monitoring, there are patches, there are upgrades very fast.

Previously when you look at open source software, the gap was there in terms of upgradability in terms of its patches being released, or it is newer version is being released, but today it is not so. It is extremely upgradable and most importantly regularly monitored and modded. That is very important in terms of when you have a GI software, and when you compare the same thing with the proprietary software, it has more features.

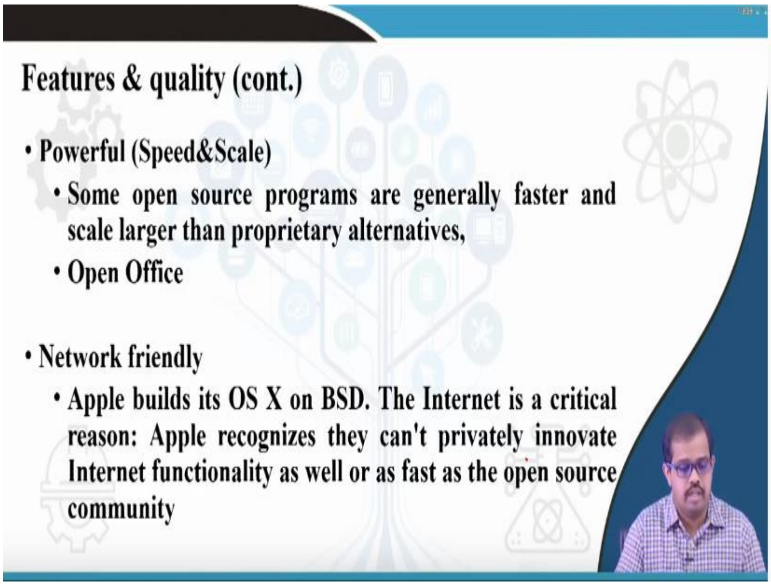
It may be easier to use peripherals like digital cameras with proprietary software. So, that is one thing that has small issue, but probably when I looked at the previous versions of Linux that had installed, it did not give me much of an issue, though I had to just configure it with some drivers, otherwise, there was no issues even with many of the open source tools. So, when there is specific proprietary thing, then it becomes an issue.

Otherwise, if it is not very proprietary and it is drivers are available as an open coded thing, absolutely there is no issues or normally drivers can be evenly installed. You have open source drivers which are available, which are compatible drivers, so that can be installed, which has compatible functions as far as your device is concerned. Only when it is highly proprietary and it is not distributable, only then it creates an issue and most importantly is more user friendly.

As I said, a cloned program may be just as user friendly as the original. Okay. So, when you are looking at proprietary, for example, many of them compare with RGIS and QGIS. It is not about comparing. RGIS is completely proprietary, so it has to give services or develop services very specific to the user community who as bought it. Whereas, a QGIS is more intended to develop it as a powerful tool. Okay, for any user to use it.

So, there are two visions of looking at these. Instead of comparing it, you should look at weighing balances between these two visions. Vision one is proprietary software is developed intended to the users who are actually using it. Whatever they want basically that is being satisfied or more may be visually more attractive or easier to handle, but QGIS as for a larger context where any user can use it, modify it, and make it more useful. So, only thing is that open source may have bit of infer that you have to put in, whereas in a proprietary software you may not have to do that.

(Refer Slide Time: 08:29)



Features & quality (cont.)

- Powerful (Speed&Scale)
  - Some open source programs are generally faster and scale larger than proprietary alternatives,
  - Open Office
- Network friendly
  - Apple builds its OS X on BSD. The Internet is a critical reason: Apple recognizes they can't privately innovate Internet functionality as well or as fast as the open source community

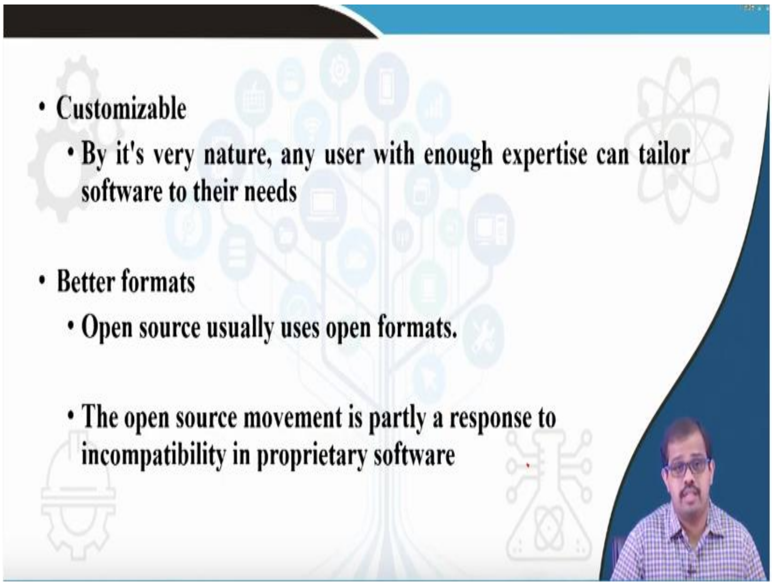
When you look at the next part of it as powerful, the speed and scale. Many open source programs are generally very faster scale larger, than the proprietary alternatives. For example, Open Office. I am sure if many of you start using Open Office, I am sure you are going to quit Microsoft office. Extremely user friendly, everything that you do with Microsoft office can be done with the Open Office.

Though there is Microsoft Office at the end of it has some polishing, I mean, I would call it as a gimmick, so there are lot of things that are there in terms of as it is because a proprietary software, whereas Open Office is direct, straightforward and playing software, which can be utilized the way you need. Most importantly, it is network friendly. So most open source software are network friendly.

When you look at Apple build, it is OS X on BSD. The internet is a critical reason in terms, that is what it said. Apple recognized that they cannot privately innovate internet functionality as well as fast as open source community. That is why it was there. It released it as open source development tools. So, some of it when you see today whatever the tools that apple has released, as an open source is the most fast development than any other tools.

That is completely under the fortes as a proprietary software of apple. That you can see is extremely different. When you compare this you can easily distinguish between the proprietary and an open source and that is how it has developed has a community.

(Refer Slide Time: 10:25)

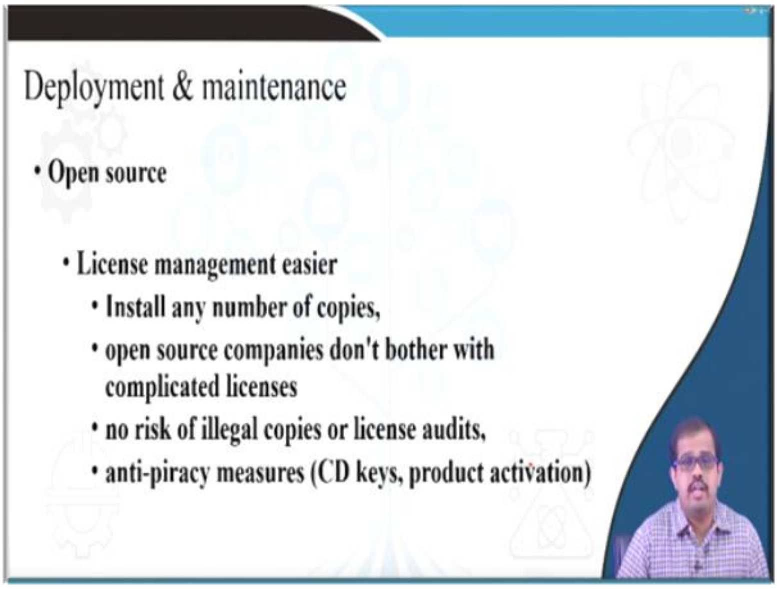


- Customizable
  - By it's very nature, any user with enough expertise can tailor software to their needs
- Better formats
  - Open source usually uses open formats.
  - The open source movement is partly a response to incompatibility in proprietary software

Now customizability. Now, it is very nature any user with enough experience can tailor software to his or her need. So, when you look at open source, but proprietary you will not be able to do it. You cannot do it basically as per the license agreement that you have. Then you have better format open source usually uses one formats available everywhere and internationally recognized format globally used formats are used with open source software, whereas proprietary software will have closed formats.

The open source movement is partly responsible for incompatibility in proprietary software. So, that is what a meaningful exercise, because there has been now the trend is that you should have interoperable extension, interoperable data. So that is there where there has been a change even with proprietary software to provide an interoperable region that is of international data or extension that are of global standards.

(Refer Slide Time: 11:33)



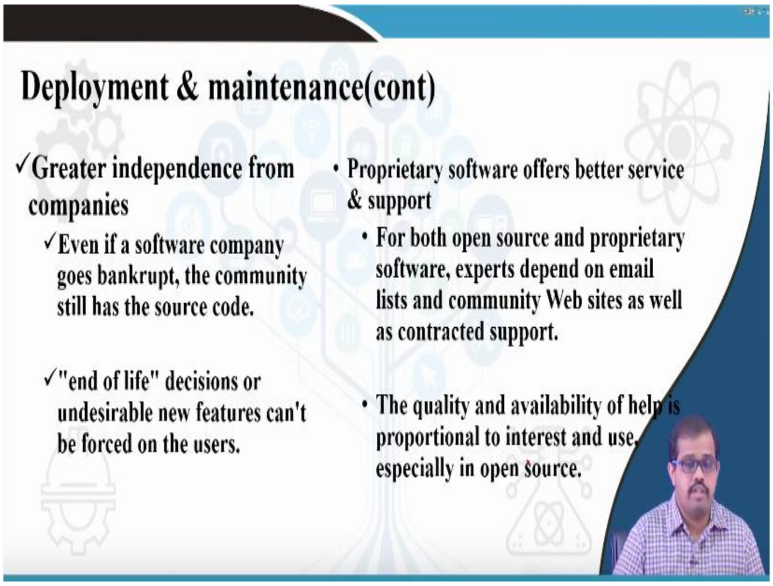
Deployment & maintenance

- Open source
  - License management easier
    - Install any number of copies,
    - open source companies don't bother with complicated licenses
    - no risk of illegal copies or license audits,
    - anti-piracy measures (CD keys, product activation)

When you look at Deployment and maintenance, when you look at open source, license management is extremely easier, you do not need to worry about licenses available. For example, install any number of copies, no one has going and asking you why you have installed 100 copies Linus or 100 systems. Then open source companies do not bother with complicated licenses as normally in the proprietary software, no risk of illegal or license studios, never.

Okay, license audits, you do not need it at all when in an open source software and the most importantly the curbing of piracy. So, much easier and straightforward. You do not need CD keys, you do not need an activation you like to do you normally do, so normally when you have installed any of the proprietary software, the first thing it ask for is that you have a CD key or license that you have to activate it online, which there is an open source software never hacks it at all. So, that is exactly why you need an open source in terms of all of these issues.

(Refer Slide Time: 12:40)



Deployment & maintenance(cont)

- ✓Greater independence from companies
  - ✓Even if a software company goes bankrupt, the community still has the source code.
  - ✓"end of life" decisions or undesirable new features can't be forced on the users.
- Proprietary software offers better service & support
  - For both open source and proprietary software, experts depend on email lists and community Web sites as well as contracted support.
  - The quality and availability of help is proportional to interest and use, especially in open source.

When you look at deployment and maintenance, open source has a very greater independence from companies. So, that is one of the reasons that open source community has been able to develop. Today, as I said many of them as I said compare with QGIS. If you look at QGIS functionalities, nowhere it lacks with any of the functionalities with RGIS. Only it lacks is that it does not give you a feel of where all these functionalities are located.

So you have to be a powerful user to understand which means you should know everything where to understand what kind of software it is, whereas RGIS, it is just a dropdown. So, that is what is the difference, otherwise, QGIS has extreme functionalities in terms of whatever the usage today’s user is concerned. Then even if the software company goes blank very important thing.

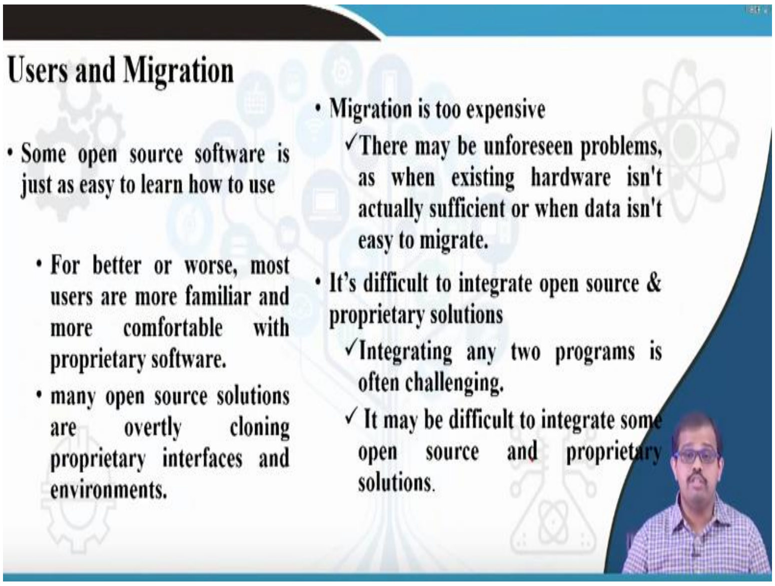
See, for example, if there is a software which you are completely dependent on. You have bought that software. It is a proprietary software. Okay, now tomorrow if the company goes bankrupt and there is no updates there is no patches, and it shuts down. So the thing that you have completely dependent on for your research, the entire thing goes in vain, whereas here when you have an open source software, even if the software company goes bankrupt, the entire committee still has a source code.

So, most importantly end of life decisions are undesirable new features cannot be forced on to the user. Okay, that is what is extremely different from an open source and proprietary software. Whereas, when I look at proprietary for both open source and proprietary software as experts depend on e-mail list and community website as well as contracted support. The quality and available of help is proportional to the interest and uses especially in the open source.

That is the lacunae. So for example if there are less number of people who are working on a specific aspect or on a specific software, then your community support may be much slower or much not available in terms of open source software, whereas in a proprietary software, it has to be available all the time. So that is how the proprietary software is far better, but with today’s scenario the context of setting this has slightly changed any aspect you have someone who have replied to your questions even in the open source community.

So, it is not lacking behind, may be in another few years, you will not find this point mentioned anywhere in the slides.

(Refer Slide Time: 15:30)



Users and Migration

- Some open source software is just as easy to learn how to use
  - For better or worse, most users are more familiar and more comfortable with proprietary software.
  - many open source solutions are overtly cloning proprietary interfaces and environments.
- Migration is too expensive
  - ✓There may be unforeseen problems, as when existing hardware isn't actually sufficient or when data isn't easy to migrate.
- It’s difficult to integrate open source & proprietary solutions
  - ✓Integrating any two programs is often challenging.
  - ✓ It may be difficult to integrate some open source and proprietary solutions.

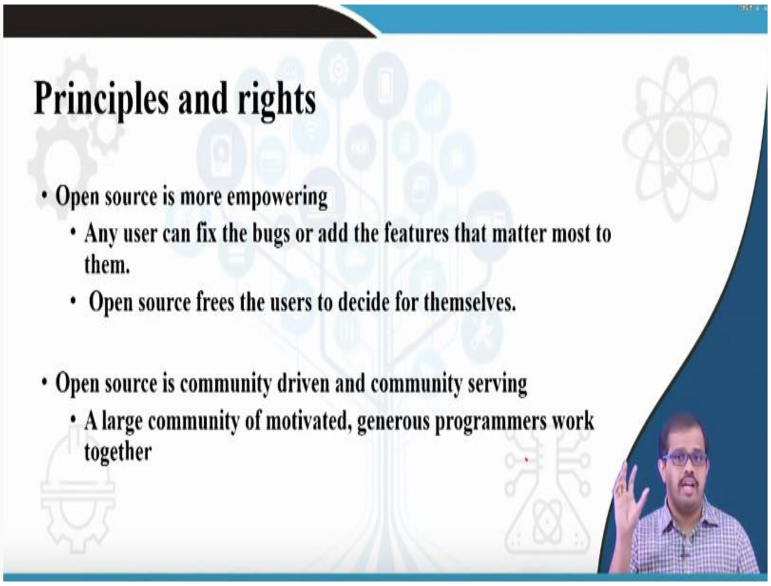
And users and migrations. So, this is where many users are concerned. Let us say I have Linux built, which is 12.02. Then it came as 14.10, 16.10, 18.10 and now 20.10 and it came up with long term support. So, now your user will be thinking about okay, if I have this continuous support, then it would be much easier for him or her to develop along with bug fixes that has been done, but open source is just easy in terms of doing all of these.

It is just as easy to learn how to use it. If you start using it, then it just a cake in terms of having it or using it. For better or worst most users or more familiar and more comfortable with proprietary software, it is because of the way it has represented and many open source solutions are overtly cloning proprietary interfaces and environments. So, you can see most of the proprietary environment in open source software.

When you look that, you will not find much of a difference in open source software, but when you look at migration, normally the proprietary software migration is extremely expensive. There may be unforeseen problems when existing hardware is a naturally sufficient or when data is easily to migrate. So, in that cases, you will have to spend a lot of effort or spend your precious time in order to look at the migration issues.

Whereas in an open source, it will never have such issues; and it is difficult to instigate open source and proprietary solutions. Though today yet it is a problem, I am sure that in another 2 to 3 years, this is completely sorted out. So, we will not have such issues. But most of the things that today is available is under a particular format or a particular data observation. So, we do not need to have such issue that is available as far as the users or the migration is concerned.

(Refer Slide Time: 17:48)

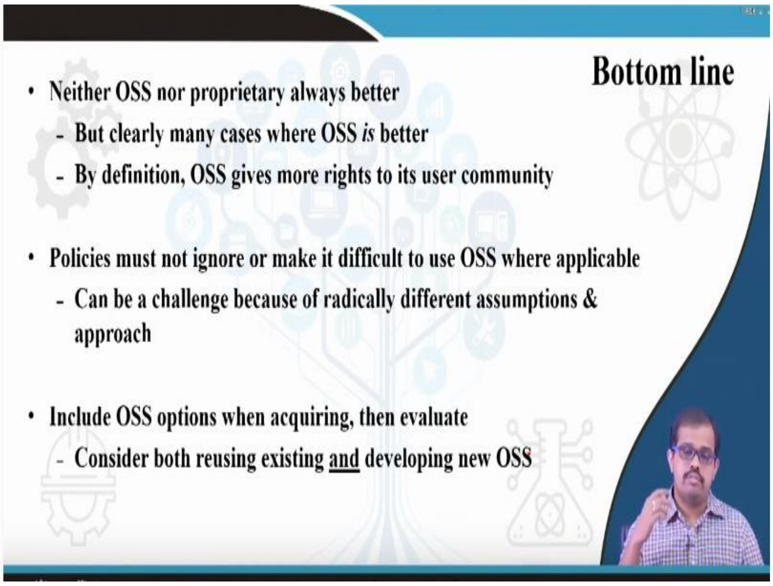


Principles and rights

- Open source is more empowering
  - Any user can fix the bugs or add the features that matter most to them.
  - Open source frees the users to decide for themselves.
- Open source is community driven and community serving
  - A large community of motivated, generous programmers work together

And when we look at the principles and rights, it is very important in terms of your usage of your software. Now, open source is actually more empowering because any user can fix the bugs or add the features that matter the most to them and open source frees the user to decide for themselves. Open source is a community driven and a community serving a large community of motivated, generous programmers work together in order to develop these things. So, that is how open source becomes more useful and more engaging in terms of having the principles and right checked.

(Refer Slide Time: 18:29)



Bottom line

- Neither OSS nor proprietary always better
  - But clearly many cases where OSS *is* better
  - By definition, OSS gives more rights to its user community
- Policies must not ignore or make it difficult to use OSS where applicable
  - Can be a challenge because of radically different assumptions & approach
- Include OSS options when acquiring, then evaluate
  - Consider both reusing existing and developing new OSS

So, on a bottom line, if you say that if you can give me in a nutshell, what is actually this open source movement means. So it is neither when you are comparing both of them, you have to always compare apples to apples, and not apples to oranges, so when you are comparing them neither NOSS or the open source software, nor a proprietary software is always better. Any can be better; it is because of the user.

Only thing is but clearly many cases where you are looking at research aspects, you are looking at academic purpose, or whether you are looking at your application purpose, open source software is always better. By definition, open source software gives you more rights. So, when I say rights, rights to both user community, both rights to both developer community, rights as a complete package is provided to you by the open source software whereas proprietary that is curtailed.

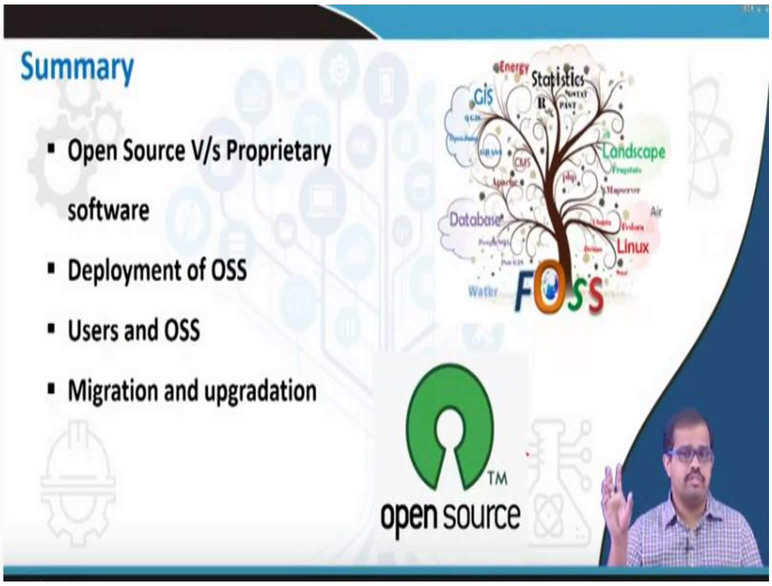
So that are some reasons to say that open source software is slightly better in terms of whatever you are trying to use. But on a whole context, if you look at the entire context or when you are weighing, neither the open source software nor the proprietary software is always better. When you are looking at policies, policies must not ignore or make it difficult to use an open source software wherever applicable that is there today.

But there is a lack of more policies driven initiators, so that open source software has really used in effective decision making and it can be challenged because of radically different assumptions and approach. Till the movement that started especially in India in late 2000, open source software had a completely different image. No one use an open source software, they used to think that an open source software is mainly.

And everything was based on proprietary software, even today you can see a lot of proprietary software being programmed, but open source software, the approach and the challenge that is there is being overcome slightly and much faster. Include OSS options, then acquiring and evaluate. That is considered both reusing, existing, and developing new OSS. So, that is one thing. I would say that this is what is crux of an OSS.

So using an existing OSS becomes a powerful user than developing a new OSS and providing it to the community will actually help the entire community, who are using the open source software. So, if every one of you can develop in such a context, so it would really develop us in the future can develop as a great aspect for more researchers to go forward.

(Refer Slide Time: 21:44)



Summary

- Open Source V/s Proprietary software
- Deployment of OSS
- Users and OSS
- Migration and upgradation

Because whenever most of the researchers had done, normally whatever the code that has been developed normally is in shelf, or it has just printed in your thesis and finally it is dusted. So, instead if you can put into a software as a tool, or put it as a library which can be shared with many or someone who would be researching on it, can use this thing and take it forward it as much easier that gives you more credit, in terms of happiness in terms of data or library being forwarded.

So, when you look at this thing, we have looked at open source versus proprietary software basically we have compared both of them. We started with deployment of open source software, how the deployment is done, what are the intricacies in deployment, the license issues, then we looked at users and open source software, that is where the major difference is there.

The security is also is an aspect that you have to consider in terms of open source software, and open source software is quite secure than a proprietary software and if you look at migration and upgradation, open source software today has extensive community, which looks at migration and upgradation further. So there is absolutely no issue or lacuna in terms of migration or updation.

So, this is what about the open source software. So, in the next session, I would look at the open source geo spatial foundation, how this data we are trying to put in on a global scale or make it a standard data, so that there is no issue about the data and whatever the data generated can be easily shared, used by others and most importantly it does not go a waste what are the data that is generated. So that is; what is the next crux of the next two lectures that I would be handling in this particular session. Thank you very much.