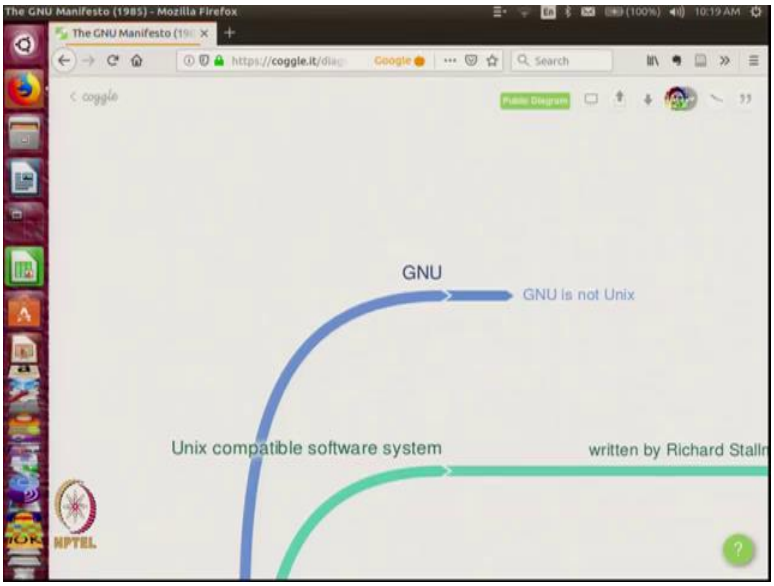
Text, Textuality and Digital Media
Professor Arjun Ghosh
Department of Humanities and Social Sciences
Indian Institute of Technology Delhi
Lecture 25
Richard Stallman: The GNU Manifesto

(Refer Slide Time: 0:27)



We are going to discuss the GNU Manifesto by Richard Stallman. Richard Stallman is a software evangelist as he calls himself. He prepared this GNU manifesto at the very early age of the early point of time of the computer revolution and what he actually, the importance of this particular text is it presents, it helps us get a snapshot of the ownership patterns of software and it is important for us to understand the ownership patterns of software because as we have seen, that writing systems within the electronic media are dependent on the software backend.

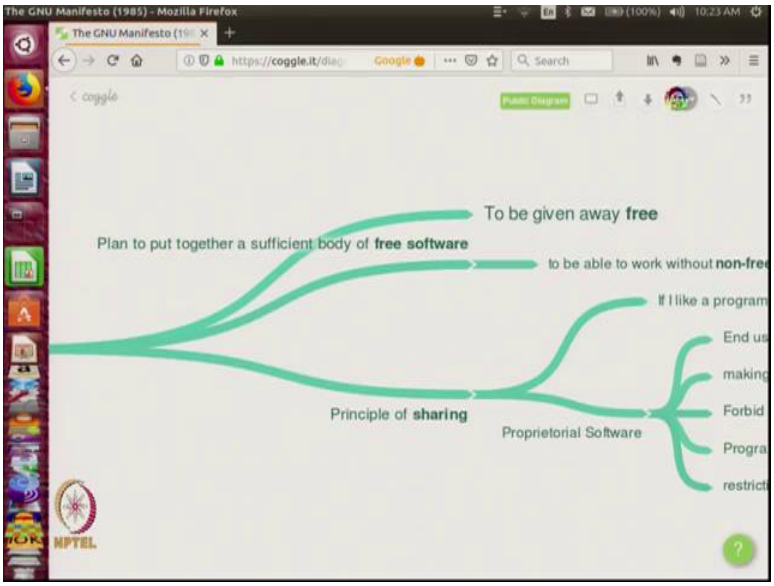
And therefore, the ownership, the control over electronic literature is dependent on the control over the software. Many of the times we go online on these various web platforms which are there and we think it is easy to publish today. We want our friends or the world to know we can just post something on social media and make the setting as public and everybody can read it, some texts can go viral. It has become so easy. You do not have to own a television station, you do not have to own a radio station, you do not have to own any printing press or a distribution mechanism. You can publish your word and it gets transmitted. That is chimera, that is a false notion because somebody owns the software.

And only when you work in confidence with the owner of the software that you can actually publish what you want to publish. If you work contrary to the beliefs or understanding of the owners of that platform you cannot, you do not have the freedom, your account can be deleted, right. So it is important for us to understand the importance of the ownership of software.

So, let us try to understand this. When we look at computers, they do normally every computer has what we call an operating system, an operating system which actually helps the computer run. So for example, this particular computer it has an operating system, it works on Linux operating system. That is the chosen platform that I have to work on. There could be multiple other operating systems, some of the more popular operating systems ones that people use is windows and IOS for users of Apple computer.

And there is a fundamental difference between different kinds of these softwares. Linux is what one would call an open software, whereas Windows or IOS would be what one would call proprietorial softwares, owned by particular companies or corporations. Linux is not owned by anybody. How do these two things operate and what are the implications for that on our understanding of texts let us see. Just to explain GNU this is kind of a puzzle acronym that was chosen by Stallman, GNU is not UNIX. That is the full form of GNU. UNIX is a very important computer back end. IOS is also written over UNIX as an operating system. Of course there are various versions the Kernel versions but.

(Refer Slide Time: 4:29)



So this essay, this particular manifesto, the GNU manifesto was written by Richard Stallman. What happened was that he was working in a particular tech firm and what he figured out as a coder and programmer is that he figured out that he is not able to, you know when he writes a piece of code that piece of code is tied to a particular piece of hardware.

So when you work on your computer and you give a print command. That print command what it actually helps is that your computer talks to that printer which is a hardware. So this conversation becomes possible. So you remember that whenever you install a new printer on the computer you may not have done it yourself and someone may have done it for themselves to set it up. They have to install a particular program called a driver, a printer driver on the computer, what the driver actually does is basically make that conversation between one machine and another machine possible.

It could be a printer. It could be a projection device. It could be a mouse or various other kinds of devices that need a driver. So there is a kind of information sharing. Now, what Stallman figured out is that that if there is a certain hardware. Which so if there is a particular printer for which the drivers are not available for the various operating system then that particular hardware can only converge with specific software specific operating systems and that becomes restrictive. For example, in certain games platforms, certain games can be only played on certain platforms certain hardware.

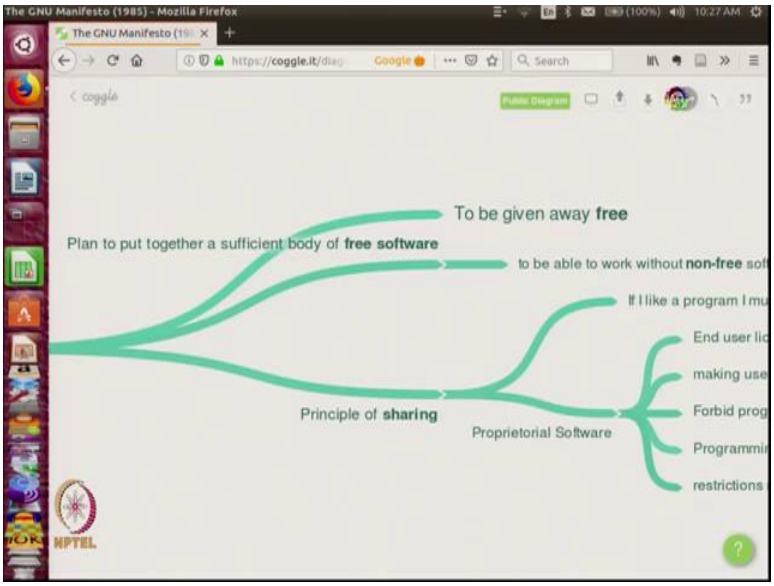
There is no standardization unlike let us say a compact disc you buy a compact disc of the market and you play it on a compact display and you should be able to listen to it if that technology is there. Of course there are various gradations from your compact disc or you have a DVD or you have a Blue ray disk.

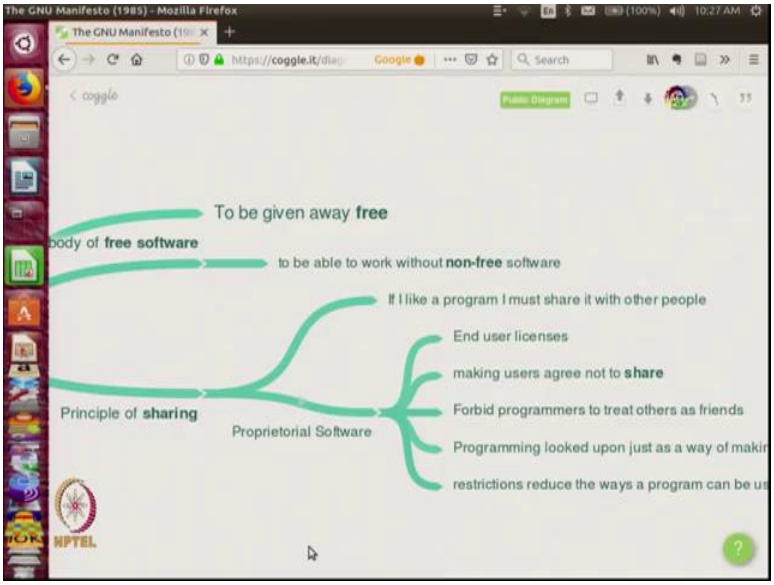
So if you have a Blue ray display player no matter what whichever one or a DVD player no matter what DVD you get from anywhere, you should be able to play that on an audio DVD or a video DVD. You should be able to play it on your platform. So that is where there is a certain kind of standardization, but very often these companies they do not engage in standardization. They develop a certain system.

But in order to play that particular video or audio you need a particular kind of hardware. So that puts a certain lock up because the two kinds of programming they do not talk to each other. And imagine if these drivers were made proprietorial, that you cannot use it without having paid for it. Very often if you install a particular software on your computer, it asks for a key and if you do not have a key you cannot run that software isn't it?

So what it means that you have to pay the company only then that key will open. So that is an important example of proprietorial software. And that means you may have got this question but because you do not have the key and have not paid for the software separately, you cannot use the machine.

(Refer Slide Time: 8:30)





So what Stallman puts a plan together using the GNU manifesto is to create software to be given away free. His plan is to put together a sufficient body of software where one should be able to work without non free software. So if I have enough software so I have a computer. And I need so what are the various tasks that we undertake in any particular computer. We write, we can record stuff, we can edit videos, we can look at pictures, we can watch movies, we can maybe play games or slightly more sophisticated work, we can play around with a little bit of code so I need a development atmosphere, I need to be able to print out things.

So these are the certain basic tasks that we do undertake within a computer. And if there is a particular task. For example, if I want to do photo editing. One very popular photo editing sort of platform which is there is proprietorial. People usually use pirated versions of these software, one uses a pirated version, but it is illegal. I mean one can argue against copyright, but violation of copyright is illegal. What free software would do is that we would not need to work with pirated software. It will it work (without) with only free software.

Now, it is important to understand this word free. When he says, when Stallman says free, he does not mean free as in free lunch. Not that you do not have to pay for it. But also the fact that free as in freedom of speech. I am free, so the entire idea is that if I like a program I should be able to share with other people.

So look at this particular situation. If I like a book I share it with a friend. So this goes back to the very nature of intellectual property. So if I like a banana, I cannot share it with the friend. Because I like it because I have eaten it and once I have eaten it my friend has to have another banana. The only thing I can share with my friend is the idea that this banana is good from this particular plant or this particular shop is good. So you go ahead, pay for it and get it.

Whereas, if I like a book. I do not have to get another book to be able to read it. Once I have read a book I have read it. I can refer to it again but I really do not need the book again. I can pass it around. That is the entire idea of the library, remember. But that is illegal. I cannot pass it around. Certainly a physical book you pass along because nobody is keeping a track of it because the physical book there is just 1 book, either I have it or my friend has it. But imagine an e-book.

Once I have purchased an e-book, a PDF or I have got a got a PDF I can pass it around. This happens with MP3 files with music, you have it on your phone you like a particular music you shared it with a friend, you just pass it around. Put it on a pen drive, transfer it through Bluetooth. That is illegal. So, now what we are now seeing are new kinds of formations of making sure this kind of file sharing becomes impossible through digital rights management.

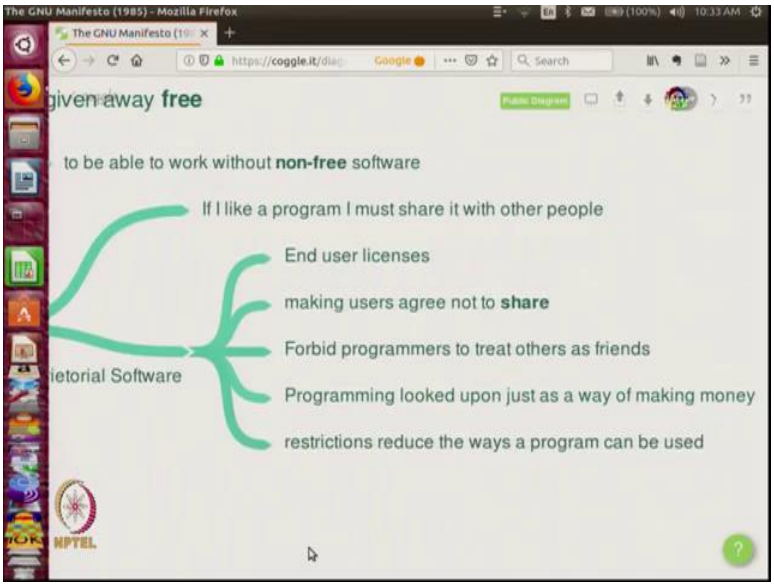
So e-books many books have platforms. So once I have a particular e-book reader. And I pay for a particular book and I download it on my machine or my device. There are certainly workarounds. But legally you cannot share it with another person on that person's reading

device. What also is happening more and more now is that once I get hold of a particular book or a particular text on my device, I might lose it at some point of time.

Because the ownership of the device continues to remain with the manufacturer. The manufacturer can change the operating system that is there within it. I mean when you have a set top box at home and if that set a box has a recording device on it and you record certain videos. You have recorded it. But if you stop your subscription you cannot access those videos anymore. Or the owner of the satellite company can actually reprogram or upgrade that software within it even remotely.

So these are ways in which newer kind of ownership systems which are coming into being that the fact that I cannot have complete control over specific texts that I have, I cannot share with you people. This is what happens with non-free software platforms. But in software platforms which are free, I should be able to share with other people. So the one way in which non free software or proprietorial software or ownership software operate is they have these end user licenses.

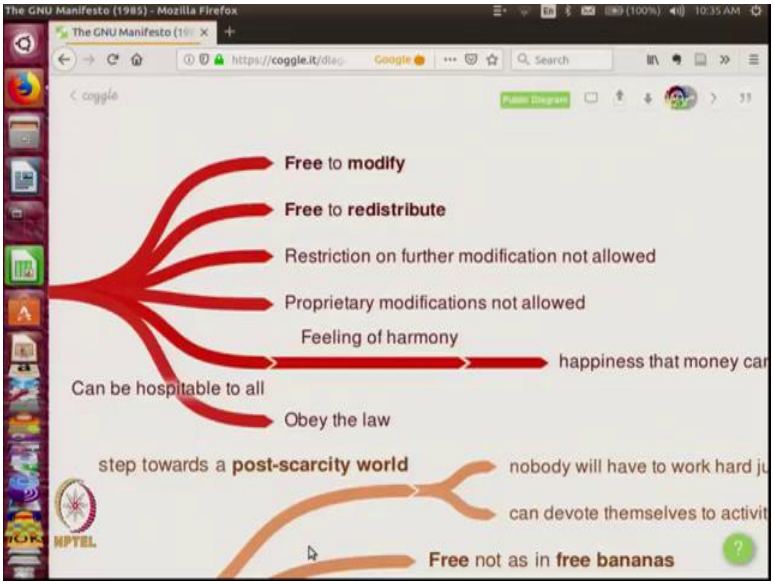
(Refer Slide Time: 14:43)



Make users agree not to share. And to not to share ideas Stallman says goes against the very idea of knowledge. Because the world of knowledge has actually grown through sharing. If we do not share with each other, knowledge does not grow. We do not learn from each other. It forbids programmers to treat each other as friends. And programming is just looked upon as a way of

making money. But programming is a way also of getting by, making use of, making better and better use of the human knowledge and technology. There are restrictions to reduce the various ways in which a program can be used. You can use a program for a particular purpose. It cannot be used for other kinds of purposes. So this goes against the very principle of sharing Stallman says.

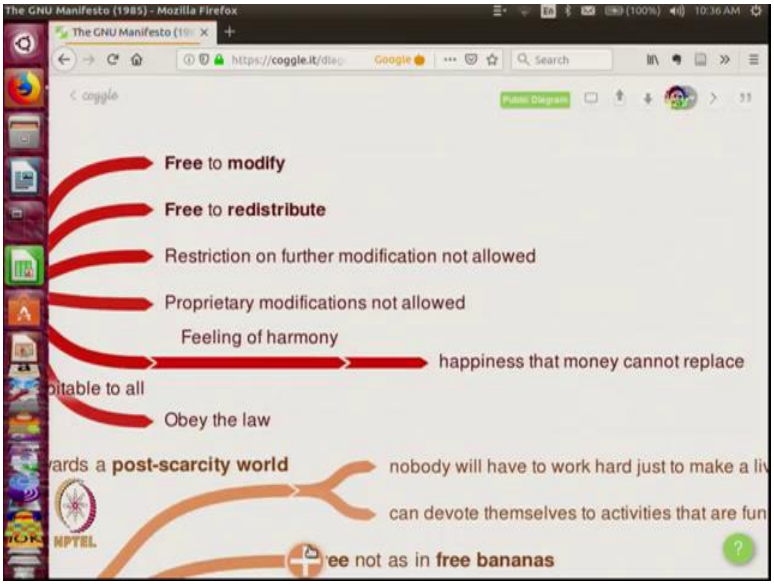
(Refer Slide Time: 15:39)



So instead of that he suggests certain principles of GNU free software. He says this would be a software which would be free to modify and free to redistribute, that okay I have a particular software and I share it with you or it is there for you to use and you find a way to improve upon it. Then you can modify it and then redistribute. You should be free to redistribute, share it with anybody and you do not have to worry about doing something illegal. So there is a restriction on, there is no restriction on further modification.

You cannot modify a particular within the GNU license has 1 quality. Within the GNU license one cannot take this free software and modify it and then put it under copyright restrictions. If you have read it on the basis of a certain free software, then you cannot make it unfree henceforth. So it creates a certain kind of ecosphere within which a software circulates.

(Refer Slide Time: 16:57)

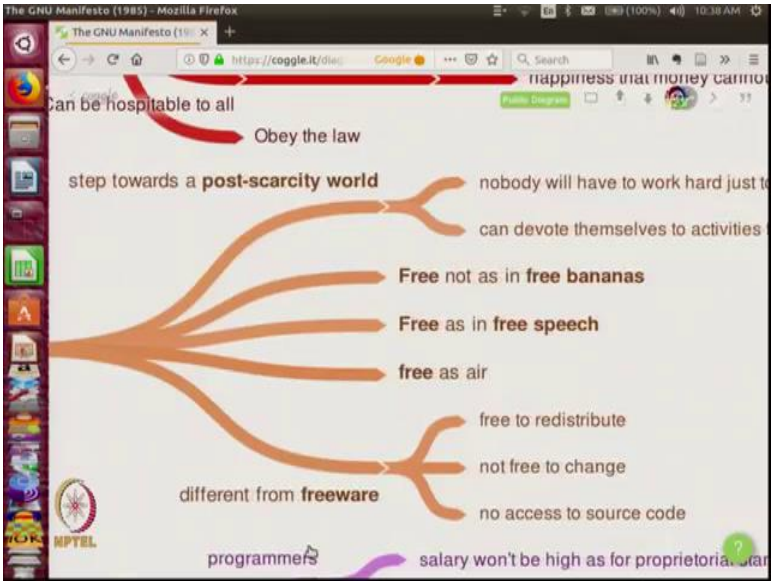


So this can be hospitable to all, he says that this this brings about a certain kind of feeling of harmony a happiness that money cannot replace. This act of sharing brings in a certain joy, a certain happiness which is not possible by simply through money. So what this encounters is this kind of a concept of happiness that I have a certain thing which someone nobody has looked at. Look at my watch. Look at my shoes. It is mine, you do not have it. That might bring a certain kind of happiness along with it, whereas a certain kind of ownership built on jealousy.

But the other kind of happiness is. I have a certain great thing. Would you like to be my friend? Would you like to have a share of it, that happiness? That certain kind of joy. So it goes beyond this kind of profit oriented market. Market controlled kind of sense of what happiness really is, a kind of a consumerist world to a post consumerist kind of world, he says that. And everybody, and this act of sharing does not break the law. This kind of sharing is hospitable. So it is not a sharing of the kind of piracy.

This does not require people to become thieves. It is another argument to say that we need to share we need to share for learning and therefore I shall pirate the software, but there is no need of that notion of illegality also to be there because software when it is developed can be if it is put under a different form of ownership can make everybody free for sharing. Sharing becomes possible.

(Refer Slide Time: 19:03)

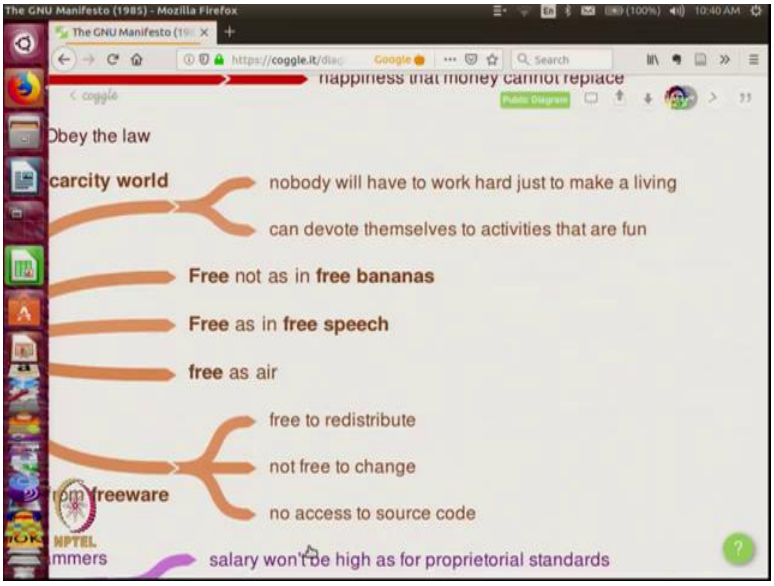


So further into the philosophy of free software it is a step towards a post scarcity world as to go back to that example as he says, this as free not as in free bananas. That is if there are just 10 bananas and there are 10 people they can eat one each, but if they are 11, the 11th person will not get a banana. But that is not true with music.

If I have a particular file of music any number of people, 1000 people or any number of people can listen to that music. If there is knowledge and let us say the alphabet. So anybody can read the alphabet, anybody can learn how to read and write. There is no restriction. There is no scarcity in that. It is not scarce in terms of being a producer, it is not a physical object. Ideas are not physical. So the ideas can spread as much as it does.

In the laws of copyright, laws of intellectual property and the ways in which (())(20:11) restricts the spread of ideas, creates an artificial scarcity of ideas- that certain ideas are not for everybody, you cannot access an idea till you have paid for it. That scarcity would have been there in the world of print because print runs a certain volume, you can print 200 copies of a book, you can print 2000 copies of a book or 2 million copies of a book, but that 2 million plus 1 would not be there. Then the book will become scarce at that point of time.

(Refer Slide Time: 20:57)



So, whereas in electronic format, there is no scarcity, there could be any number of copies of a particular text and it will never become scarce. So it points out towards a certain post scarcity world which is futuristic, says that if we have enough to eat, if we have enough shelter, if we all have health care, then human beings can all be involved in activities which are liberating. That movement towards that. And having free software is something that will allow that movement, will make that movement possible.

So he says free software is free as in free speech. It is a freedom. And it is also free as in air. It is a public good. It is available to everybody. Not so much as air being free in a polluted universe. Only few people can actually get pure air, if you have a room purifier and so that is like the privatization of air or the privatization of water where you all have to drink some RO generated water or bottled water because the water or the free water the one that you do not have to pay for will make you unhealthy. So that is there.

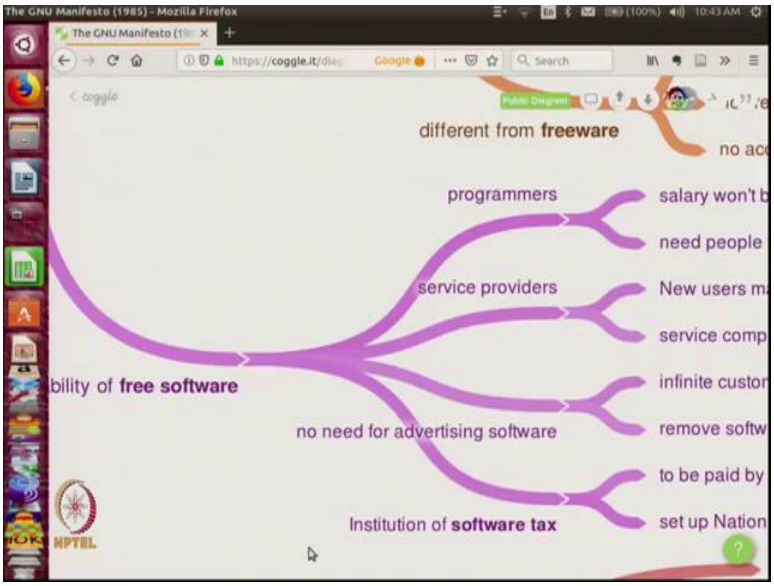
But conceptually unless the air has been polluted, air is supposed to be free. Everybody breathes it, you do not pay for it. So that is the idea. And this is different, he says free software is different from freeware. So there is a different word. There is a free and open source software. Free and Open Source Software is different from freeware because what freeware does it is just free to, freeware does not allow, it is free just you do not have to pay for it, but it may not be free to redistribute.

It may not be free to allow you to modify and it does not give you access to source code. Whereas, free and open source software under the GNU license would allow people access to the source code, the code that is written because the code, that backend within which it is written is important.

It is necessary for one to make modifications and make improvements. So a large part of for example the software that I am using for this particular computer. That is a Linux Ubuntu operating system, it is created purely by volunteers from across the world. Everybody makes little modification and it becomes. If there is a particular problem, it is solved through volunteer effort.

And that volunteer effort becomes possible only when the source code is available that the programming backend to this particular computer which is there which helps the software the hardware of the computer talk to the binary code that is underlying the computer. Access to that source code is important in order to make modifications possible.

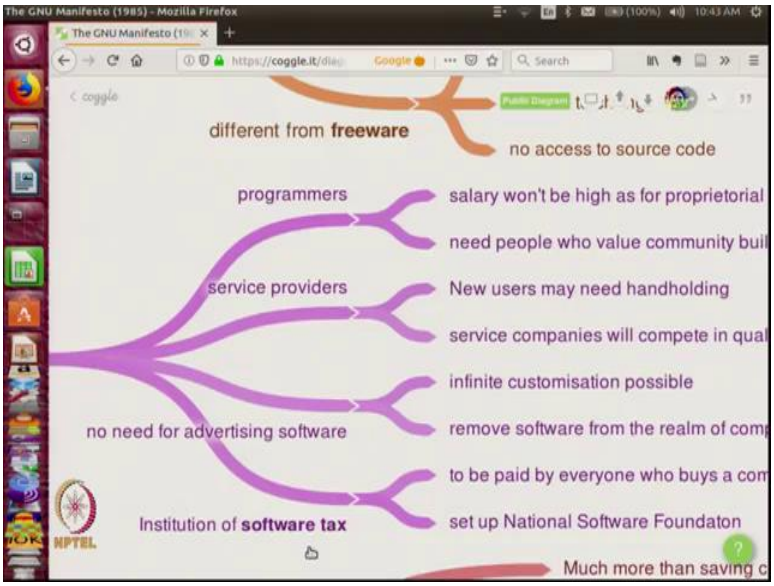
(Refer Slide Time: 24:13)



Now, how does and he makes a certain conjecture as to how free software can be sustainable because what Ubuntu make free software sustainable you need certain actors, you need an organization so there will be programmers. There will be service providers. And how would they be paid? Because within a corporate system. A software, corporate software company they pay

the programmers and they also have service providers people who would come and install things on your computer at a charge.

(Refer Slide Time: 24:53)



So, if everybody, if all software is free, how would these programmers actually survive? So he says that it is possible that that the salary will not be very high vis-a-vis the proprietor he is making some conjectures and remember he is making these conjectures in 1985. So some of these ideas may be dated, but some of these ideas are also because they are fundamental. They are talking about a fundamental way of ownership of software.

So therefore, they are important for us to understand. So what these programmers will depend on would be a certain kind of valuation, they would get a certain kind of ideological support, motivation towards this kind of programming. And this is not an out of the world idea. Remember Voltaire pirated his own books. He did not care if he left out on some money because he was ideologically motivated. He wanted to spread the ideas of the enlightenment.

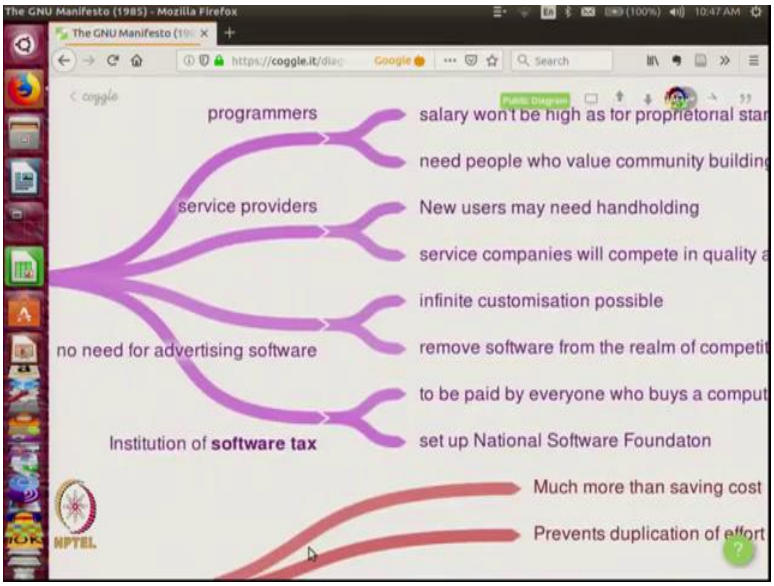
He wanted to challenge the Roman Catholic Church. Or Luther for example, very important example, would Luther have cared for being paid for his pamphlets. No, he what he really cared for was a spreading of certain kind of ideas. So these programmers would be motivated by that, so there are two kinds of motivations at work here that he is talking about. One is monetary motivation that I have earned more and more salary and thereby I will create this kind of I will write better and better code. Or I will create a better community resource I will help the

community solve its own problems. If I create free software platforms, computers will become cheap.

Remember every machine that you buy, you pay a lot for the software which is there. Like for example this particular machine it came loaded with a particular proprietorial software. I paid for the software. I do not use it. I do not use it at all. I use free software on this particular machine so I paid that few thousand rupees unnecessarily. That can of course, that money can of course go into the servicing of the free software industry. If such a software industry could come into being.

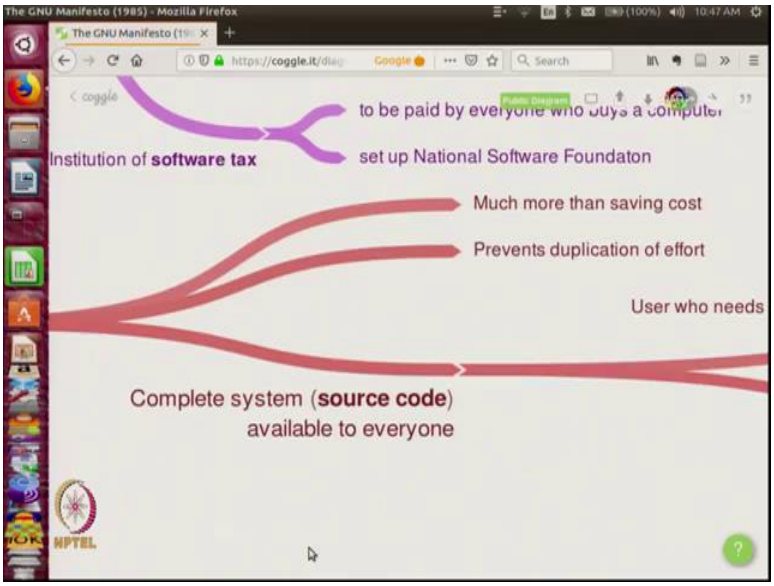
And the service providers new users may need some handholding. So these service companies will of course pay for the service. The service will not be provided by the programmers who are creating the software but people on the ground. And that the service person can charge whatever for whatever work that they do.

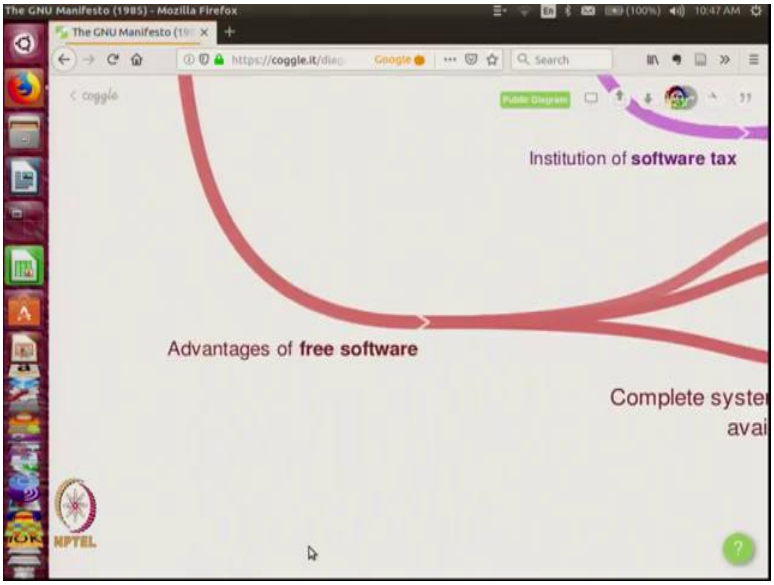
(Refer Slide Time: 28:10)

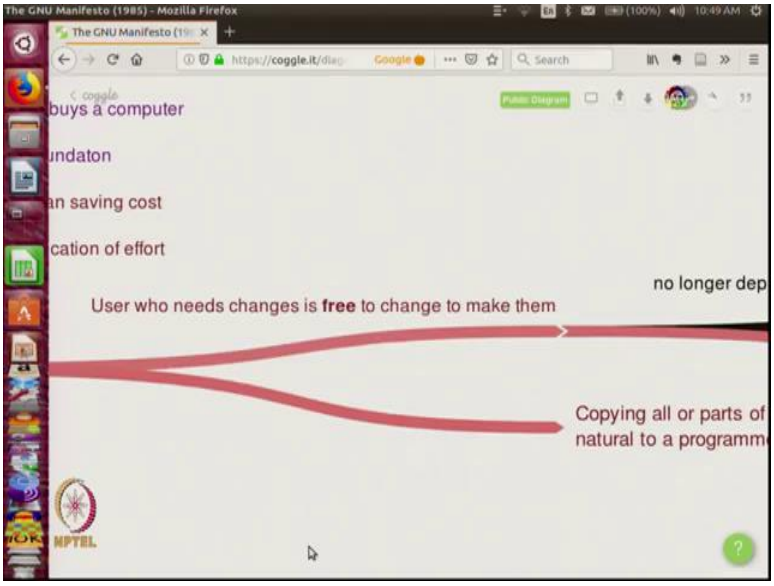


And so there would be no need for advertising the software because the software would be infinitely customizable because modification is possible, if I have access to the source code, I can change it to operate or work in exactly the way that I would want to work. And he suggests that they should be publicly funded, the creation of free software should be publicly funded through tax money. Within the context of the United States, he suggested the creation of a national software foundation.

(Refer Slide Time: 28:46)



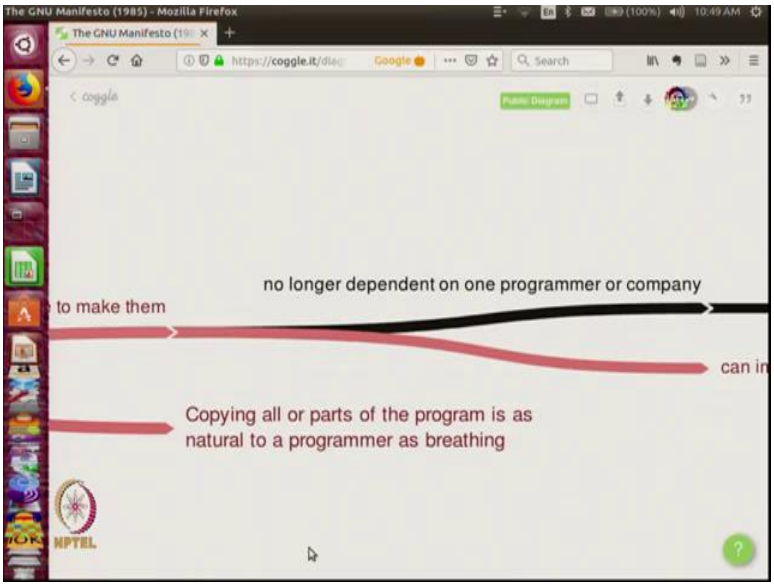


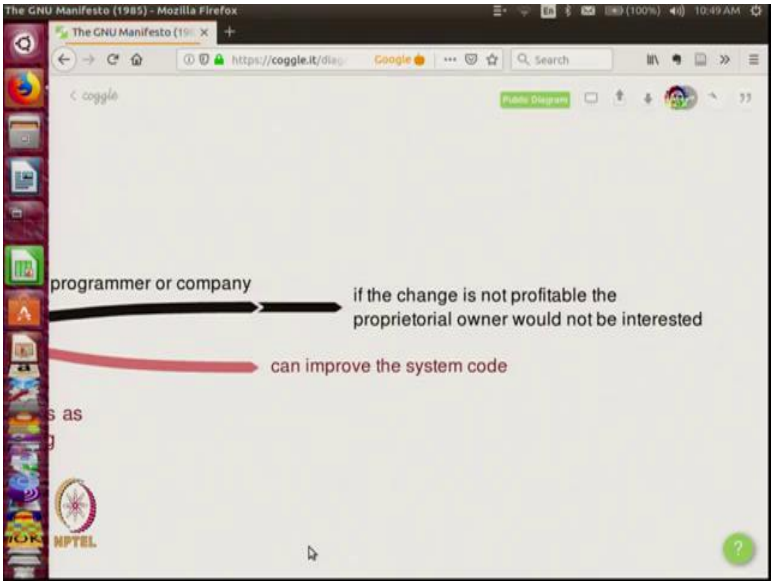


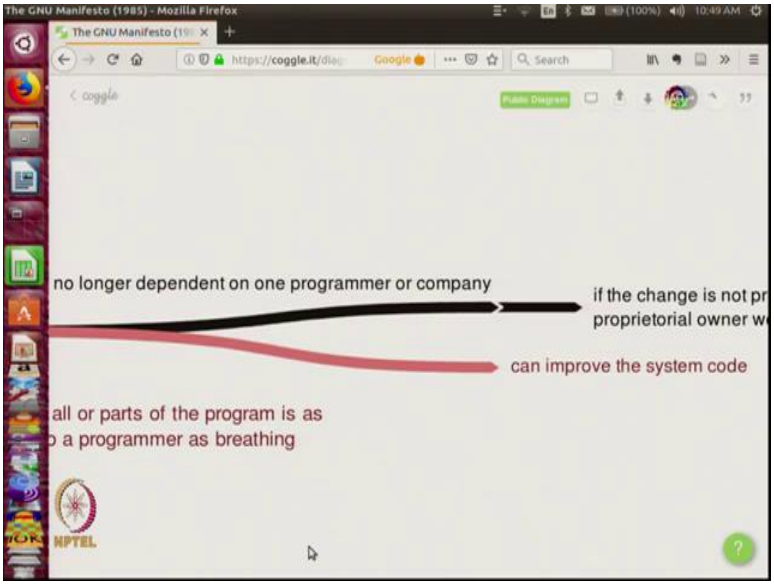
So what are some of the advantages that he says are of free software. So he says that it is much more than saving costs. That one does not use open software in order to merely to save cost. But what it does, it provides a much more robust software backend. You may be surprised to hear many of you that most of the servers across the world, they use Linux based server software because they are much more robust. Less prone to attacks because the source code is open so if there is a problem identified within a software, anybody who has the necessary know how can change it.

Whereas in the case of a closed source only people who are authorized can actually make corrections and offer patches or upgrades to particular systems. So it helps prevent duplication of effort. It is a much more efficient way of functioning and the source code is available to everyone so anybody who needs to make a change is free to change them.

(Refer Slide Time: 30:16)







And so they do not have to depend on one programmer or one company. So if it is dependent on one company for a particular change or a particular patch is not profitable, then the proprietorial owner would not be interested. Whereas in the case of an open software system, it can actually improve incrementally and make the software much better in operation. That is the fundamental idea that he puts forward.

Certainly when understanding this particular essay we must say that it does not mean that proprietorial software is always inferior. There are some very good examples of programming proprietorial environment producing very good software. But what he says is that proprietorial software the two basic reasons why he argues against them.

One is that proprietorial software is not necessary in order to produce good programming. It can be produced within a free software environment. And two is that free software environment actually favors the principle of sharing which is a natural human impulse to share ideas, to share happiness and therefore he argues for free software and the importance for us to understand here is the ownership patterns.

The crucial point that we need to remember is that it is possible within the digital universe to imagine a kind of a set up with minimal capital input, we do remember that when we were discussing the coming of print, we did discuss also the fact that print is a capital intensive industry or any form of mechanical production is a capital intensive production. It requires the

investment of a large degree of capital in order for it to happen. So that leads to that linkage between proprietorial ownership of ideas.

Whereas preprint, there was no private ownership of ideas, these ideas would be owned by certain communities not by individuals and they would be shared on the basis of community sharing.

We will we will explore these things a little bit more in other lectures. But the point to be made is that Stallman's interventions in the GNU manifesto are important for us to understand the economic relationships that the digital media is beginning to question. Is beginning to question the very basis of intellectual property and copyright. Thank you.