Torbjørn Ludvigsen

Live Software for RepRap Assembly Workshops

MASTER'S THESIS
SUPERVISOR: MARCIN JAKUBOWSKI
EXAMINER: MARTIN ROSVALL
DEPARTMENT OF PHYSICS
UMEÅ UNIVERSITY

This document was compiled on July 8, 2016.
Status: Defended and approved

Abstract

A key step when initiating robot powered production is setting up the control software. This can be a threshold for operators, especially if the software is fragmented and system requirements are extensive. One way to address this is to pre-configure all the control programs and bundle them with a system that fulfills all the requirements.

In this work a live Operating System (OS) is loaded with control software and configured to meet the needs of those who have just assembled their first 3D printer. The problem of downloading, configuring and installing various 3D printer controlling programs is reduced to the problem of distributing and booting the live OS.

The solution of loading it onto bootable USB drives is tested and evaluated in the context of a commercial RepRap Assembly Workshop (RAW), an event where people pay for RepRap 3D printer parts as well as assembly and usage supervision. The RAW is unusually short, so the bootable USB drives' potential to help RAW hosts with particularly tight time schemes is tested.

The results show a limited success. The USB drive is documented not to work for 3 participant groups out of a total of 11 groups. As a solution to fragmented software and diverse system requirements, the live OS is found to work well once booted. Several routes to make the live OS more easily bootable are discussed.

Usage examples beyond drop-in replacing existing RAW software setup procedures are discussed.

Sammanfattning

Alla som startar upp robotiserad produktion måste få igång och börja använda styrprogramvaran. Detta kan innebära en hög tröskel för operatörer, speciellt om programmen är många och har olika gränssnitt och systemkrav. Ett sätt att hjälpa operatörerna komma igång är att konfigurera alla styrprogram på förhand och bunta ihop dem med ett system som uppfyller alla krav.

I detta arbetet laddas ett live-operativsystem med styrprogramvara och konfigureras för att lätt kunna användas av någon som precis har satt ihop sin första 3D-skrivare. Problemen med hemladdning, konfigurering och installation av en rad olika styrprogram reduceras till problemet att distribuera och boota live-operativsystemet.

Lösningen att lasta live-systemet på startbara (eng. bootable) USB-minnen testas och utvärderas som en del av en kommersiell RepRap monteringskurs (eng. RepRap Assembly Workshop), ett evenemang där deltagare betalar för RepRap 3D-skrivardelar, hjälp med att montera ihop dem och med att använda de färdiga 3D-skrivarna. Den undersökta monteringskursen är ovanligt kort, så startbara USB-minnens potential att underlätta för tidsprässade kursvärdar testas.

Resultaten visar en begränsad framgång. Vi visar hur 3 av 11 deltagargrupper inte lyckas använda de startbara USB-minnena. Som en lösning på fragmenterad mjukvara med stora systemkrav fungerar live-operativsystemet bra när det väl har startats. Olika sätt att göra live-systemet lättare att starta diskuteras.

Tillämpningsområden utöver att ersätta existerande programuppsättslösningar för RepRap monteringskurser diskuteras.

Contents

1 Introduction

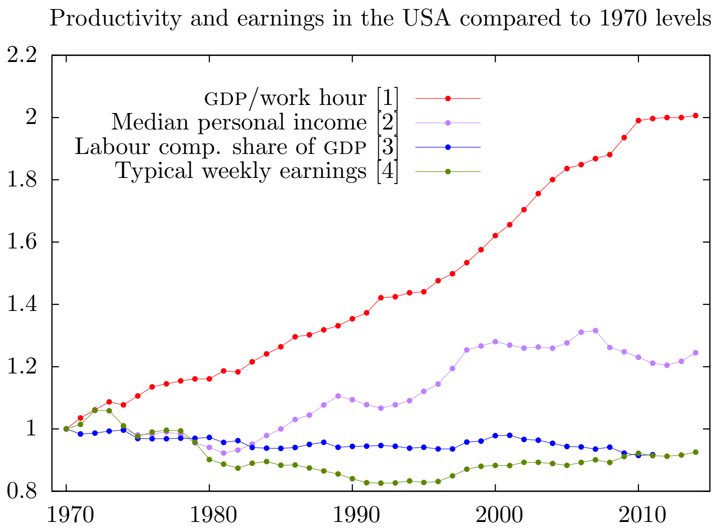

Figure 1.1: This plot shows that labour productivity, defined as Gross Domestic Product (GDP) per work hour, doubled in the USA between the years of 1970 and 2014. It also shows that typical real incomes have only increased ca $8.5 \pm 16\,\%$ in the same period. Labour compensation's share of the total GDP is shrinking. "Typical earning" refers to that of production and non-supervisory workers in private nonagricultural industries. Prices are constant in relevant data sets. Sources: [1, 2, 3, 4].

American organizations have learned to use new production machines such as computers and computer controlled robots, and reorganized themselves around them to realize a doubling in labour productivity since 1970 [5, 6, 7, 8]. American individuals have not done the same to their personal incomes [9]. Data on these trends are plotted in Figure 1.1.

Previous research on distribution of productivity payoff has focused on terms from macroeconomic models, e.g. alternative real wage measures [10], inflation [9, 11], tax systems [12] and globalization [13]. The same terms are often at the core of proposals that aim to boost median wages.

This paper instead focuses on helping individuals and small groups control computers and robots directly, as a way to bring productivity payoffs to median Americans[1]. This viewpoint offers routes for engineering contributions to incomes of median Americans, as research have shown that they are generally frustrated and confused by, rather than in control of, computers [14, 15, 16]. We can help them to learn and to stay in control by designing simple and small scale robots and computer programs.

[1] In *median Americans* we include all Americans except those with incomes above $46 000 per year (about twice the inflation adjusted median of 1974 [2]) and those who can never afford a $1000 machine or weekend event.

1.1 *Copyability and Structural Virality*

Simple and small scale robots have two important advantages over possibly more efficient large scale robots. First, they are easier to learn and teach. Second, sufficiently simple machines may be copied, remixed and distributed independently by expert users. This lowers prices for beginners and keeps the interest of expert users by offering a way to profit. To make an economical impact, we need these advantages to amplify each other.

Under the slogan "wealth without money" [17] the RepRap Project proved the feasibility of user copying and distribution by demonstrating and publishing the design of a 3D printer [18, 19]. These machines are the driving example of simple and small scale robots in this paper.

In this paper we use the word *copyability* as a shorthand for *practical ease and legal possibility to independently make and distribute copies*. Copyability describes the ease at which users can become independent suppliers.[2] Engineering complexity of course affects copyability but we explicitly include every aspect that can hinder or facilitate copying and distribution here. That is, we consider price, unique part count and documentation to affect copyability, as well as *software freedom*, as defined in [20], and the degree of *self-manufacture*, as defined in [18]. Note that very complex things, such as cats and dogs, can have a high copyability if the copying procedure is available and well known.

[2] A measurement of copyability would require quantitative study of "practical ease" in a wide range of situations, which is outside of the scope of this paper.

A high copyability gives a machine design two important characteristics. First, human operators get a level of control from every part of the copying process that requires human intervention. Second, it enables the machines to spread with a high *structural virality*, meaning a high mean path length in a tree structure that describes transfers [21]. Two such tree structures are contrasted in Figure 1.2.

Structural virality is defined as

$$\nu(T) = \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij}, \tag{1.1}$$

where T is the tree structure, n is the number of nodes and d_{ij} is the length of the shortest path between node i and node j.

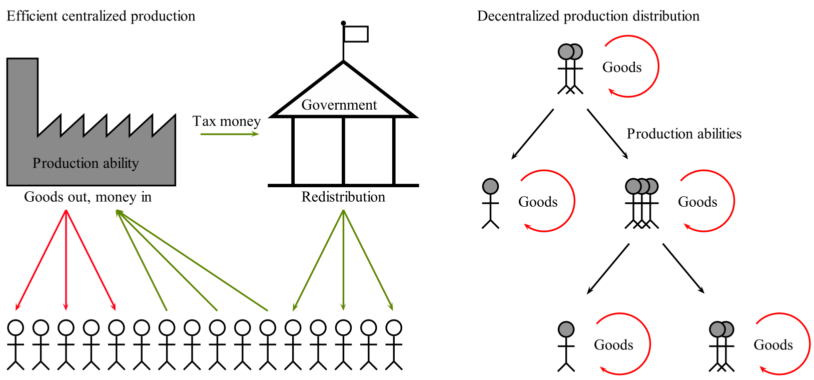

Figure 1.2: Stick figure explanations of two different models of production and distribution. The left model shows distribution of centrally produced goods coupled with a redistribution cycle of money. In the right model, groups of small scale producers are the main actors and production abilities are distributed along the black arrows. The right model would result in exponential growth in the number of producers if every producer would enable new producers at a constant rate.

Structural virality of the two tree structures in Figure 1.2 can be determined by identifying all the node pairs and the lengths of the shortest paths between them. To simplify calculations, we can reduce the left tree structure to five nodes, connected like this: . Structural virality of goods spread is then calculated like

$$\frac{4 \cdot 1 + (3 + 2 + 1) \cdot 2}{4 + 3 + 2 + 1} = 1.6, \tag{1.2}$$

since 4 shortest paths between pairs (those including the factory) have the length 1. All other shortest paths between pairs go via the factory and have the length 2.

The right pattern in Figure 1.2, which describes ability spread, also involves five nodes but has the slightly higher structural virality of 1.8. A more detailed description and discussion of structural virality is found in ref. [21].

A structural virality number must always be accompanied by a description of the nodes in the tree-structure, what is transfered along the edges and how "old" the tree is, to facilitate meaningful comparison in the future. Structural viralities below 2 will however be considered small regardless of node type, edge meaning and time scale. This is because only star networks or other very compact networks can have so low structural viralities.

Previous literature contains few recorded structural viralities that are relevant to our work. We therefore limit ourselves to state that our goal is to optimize production and distribution processes that have a high structural virality. Some insight into what "high" might mean to us is presented in Appendix B. A useful insight when comparing structural viralities is that (perfect and full) k-ary trees have structural viralities roughly equal to their depth.

1.2 RepRap

RepRap 3D printers were invented during 2005–2008 [18, 19]. Structural virality of development was high enough that the originators were no longer in control of development by October 2010 [19].

The copyability of RepRap 3D printers comes from their free licencing, low price, widely available parts and design files, helpful Internet community and the ability to manufacture a large fraction of their own parts [18]. They have been shown to be a very attractive economical investment for Americans who manage to print household items with them [22]. However, usage and market studies points at major barriers to wider adoption of desktop 3D printers in general [23, 24]. Usage and maintenance complexity have kept adoption mainly within groups of hobbyists with special skills and interests. The number of RepRaps worldwide is probably well below 1 million as of May 2016 (see Appendix D).

The RepRap Project was aware of and actively promoting copyability and the possibility of a high structural virality, particularly a high structural virality of machine spread. The project's originator, Adrian Bowyer, commented in 2013 [25]:

> "I don't really think that maintaining the position [in the do-it-youself 3D printer- and maker community] is a problem. After all, if every non-replicating 3D printer makes just one RepRap at some point in its life, you can see what that does to population dynamics."

1.3 Structural Virality of RepRap Spread

Bowyer's hypothetical scenario is exciting, but we should not tempt ourselves to believe that everything copyable gets forwarded. It has been shown that things as copyable as simple Twitter messages generally spread with relatively low structural virality [21].

Looking at 3D-Hubs' data presented in Appendix D it is clear that RepRaps don't outnumber non-replicating 3D printers yet. On the other hand, the Clone Wars data, described in Appendix B, shows that 276 person-to-person RepRap 3D printer part transfers have reached a structural virality of 5.0 within only 3 years.[3]

[3] This is higher than the median structural virality of heavily re-tweeted images and videos (ca 3) but lower than the median of heavily re-tweeted petitions (ca 7.5) [21]. Take this comparison with a grain of salt since machine parts are very different from tweets.

This is fast in terms of entrepreneurship and manufacturing incubation. However, at the time scales of Free, Libre and Open source (FLOS) communities like RepRap, 3 years is long enough to represent a barrier. Median Americans must stay motivated for months and years in order to achieve high structural virality, and harvest the economical benefit of RepRap's copyability.

Research on what differentiates the motivation of long term members in the RepRap community is sparse. However, all RepRap software is FLOS software, so we expect an overlap between motivational factors of FLOS software communities and the RepRap community.

Motivating factors within FLOS software communities have been found to be diverse [26, 27, 28]. A good review is given in ref. [29], who focuses on understanding sustained participation in FLOS software projects. It finds that social feelings and experiences within the community, especially active contribution, learning and raising expert status, predicts long-term participation far better than factors of initial motivation.

Another study found that the level of collaboration among RepRap community members was higher for hardware than for software [30]. We therefore adopt the view that many RepRap community members want to focus on and contribute with hardware modifications, and not software modifications. The relatively large number of FLOS software projects compared to FLOS hardware projects makes this assumption plausible, since it gives software developers many more projects to choose from.

We assume that social feelings and experiences is as important to RepRap developers as they are to FLOS software developers, but that software skill requirements risk demotivating them from long-term participation.

1.4 RepRap Assembly Workshops

RAWs are an example of social RepRap community events that focus on hardware. Participants meet up in person to get guided through series of assembly steps, and get the RepRaps they assemble with them home. To show how RAWs have contributed to RepRap spread we shortly describe Josef Prusa's work.

In 2010–2013 he hosted a series of RAWs across Europe, funded by pre-selling 3D printed RepRap parts to participants [31]. He instructed 2-day workshops, which was unusually short at the time, using the new and simple Mendel Prusa design [31]. The investigation in Appendix D shows that Prusa RepRaps are the most numerous desktop 3D printer design in 2016, counting well over 85 000 copies. Prusa designs are by far the most popular ones among RAW hosts, as shown by another short investigation, presented in Appendix C.

RepRap Assembly Workshop Software Procedures

Table C.1 in the mentioned investigation also shows which software RAWs have used, and the most frequently observed toolchain consist of the following programs:

- Marlin [32]
- Arduino Integrated Development Environment (IDE) [33]
- Slic3r [34]
- Pronterface[4] [35]
- OpenSCAD [36]

[4] Pronterface is the Graphical User Interface (GUI) of a software suite called Printrun.

Marlin is a RepRap firmware, running on a microcontroller, handling sensors and motors. Arduino IDE runs on a PC or a laptop and is used to install Marlin onto the RepRap's microcontroller. Slic3r translates 3D models into commands that Marlin understands. Pronterface sends commands (possibly generated by Slic3r) from a PC or a laptop to Marlin. OpenSCAD is a program for making 3D models.

Some investigated workshop hosts provided web archives to ease downloading (for example ref. [37] and ref. [38]) for participants. Others offered pre-configured computers for loan during workshop. The Michigan Tech Open Sustainability Technology (MOST) lab used Franklin Firmware and Server instead of Marlin, Arduino IDE and Pronterface, even though Franklin Server does not work on Windows [39]. Some hosts provided configuration files for Marlin and Slic3r but no further support, and others didn't offer software support at all [40].

We assume that required software knowledge limits the copyability of RAWs. This is confirmed by a study on RAWs in American high schools. It lists software issues, both troubleshooting and installing, as great barriers to fully realizing RepRap's potential in the classroom [24]. It describes the RepRap software tool chain as immature, long and complex, and 12 % of asked teachers rates "3D printer inoperable due to software issue" as an obstacle to integrating 3D printers into academic lessons.

We therefore look for ways to deliver pre-configured software toolchains in simple to use packages.

1.5 Live Operating Systems

Live OSes are a packaging alternative. They are made to be loaded into portable data storage media such as CDs, DVDs and flash drives. Most laptops can be configured to boot directly from portable media.

Once booted, users can easily copy live OSes onto more pieces of portable storage.[5] A configured live OS may therefore trade away various software download-, configuration and installation procedures, at the cost of requring physical storage media, a copying procedure, and a boot configuration procedure per laptop.

[5] . . . if their licences permit this. GNU/Linux based ones carry FLOS licences that explicitly permit such copying.

For an overview of how custom live systems can serve specialized communities' needs, see ref. [41]. Previous examples of live OSes configured to portably run a narrow category of applications to serve makers, bioinformatics researchers, scientific computing researchers and mathematicians communities include Meikian [42], MASSyPup [43], Knoppix/Math [44], ClusterKnoppix [45], Bio-Linux [41] and TAILS [46].

Five technical factors make live OSes and their portability increasingly functional in 2016.

1. Most laptops now support the same 64-bit processor architecture.

2. Lower price and less technical constraints have made more Random-Access Memory (RAM) available to laptop OSes, with 4 GiB or more being fairly standard. This allows small but complete OSes to fit comfortably in RAM.

3. Uniprocessor performance growth has slowed down [47, p. 3] which have led to a slower growth in processor requirements of common software. This means both laptops and OSes stay relevant and compatible for longer periods of time.[6]

4. Flash storage lifetimes have increased greatly [50, 51] to a level that is usable for live OSes in frequent use.

5. USB, the bus that portable flash storage is commonly connected to, has gotten faster standards over the past few years. This shortens load times from flash drives to RAM.

[6] A laptop capable of running Windows 7, released in 2009, should be able to run Windows 10, released in 2015, according to Microsoft [48, 49].

Research Question

We have described how computers have increased labour productivity more than real median wages, how simple and small scale robots may bring computerized productivity payoffs to median Americans, how the copyable RepRaps 3D printers are already numerous and economically beneficial, how RAWs contributes to their spread, that RAWs can be made more copyable by shortening and minimizing required software knowledge, and that a configured live OS might achieve this. We have thus motivated the following research question:

> Can RAWs aimed at the general public be shortened without decreasing their copyabiliy by swapping the steps of downloading, installing and configuring software with booting a live OS with pre-packaged software?

The following chapters describe how we designed an experiment along with some technical motivation. Host and participant experiences, rather than technical specifications, are collected and used to suggest and discuss a convincing answer.

2 Method

2.1 Overview

The research question was tested through trial-and-error in a full experiment. A live OS was configured in Sweden and sent to Open Source Ecology (OSE), a small organization in Missouri, USA, who tested it in a one-day (12 h) RAW using a Prusa i3 design. RAW participants were asked to fill out a web survey after the workshop.

We did some separate testing of booting the live OS from a USB drive on a range of different laptop models in Sweden, noting down if they would cause us trouble in a workshop situation.

2.2 Subjects

The main subjects of the study were 2 workshop instructors from OSE, the live OS and 24 workshop participants organized in 11 groups.

Minor subjects of the study were 11 laptops of various models and 11 Folgertech 2020 Prusa i3 kits.

Open Source Ecology

OSE's role was to test the live OS's fitness for workshop usage. They had previous experience with hosting assembly workshops for tractors and other large machines, and also some experience with using desktop 3D printers from before. They had little experience with software development and GNU/Linux administration. OSE's Internet connection was slow and unreliable during development.

OSE's motivation was twofold. As an organization they depended on workshop revenue to support further activity. They were also motivated by a will to bootstrap viral machine spread. The organization's mission statement revolves around creating an open source economy through distributing production [52].

The workshop was the first in a planned series of RAWs intended to make participants capable of hosting their own RAWs. OSE call this type of enterprise a *distributive enterprise* [53, 54] and the workshop was part of a larger project called *Distributive 3D Printing Enterprise*, often shortened to D3D. More on D3D and distributive enterprises are found in ref. [55] and ref. [56].

D3D-Porteus Live Operating System

Keeping size down was a major priority throughout choosing and customizing of live OS because of OSE's slow internet connection and because we wanted to load the entire system into RAM.

Porteus was chosen among many good GNU/Linux live distributions because it was minimal, could be entirely copied RAM, was actively maintained and easy to remaster. It also included an install script that loaded Porteus onto a USB drive without overwriting previous contents.

Other live distributions share these qualities but the Porteus web page also offered a GUI to easily start a custom system build [57]. This gave Porteus a head-start at meeting our customization needs. The simple module system was also considered helpful for customization. Its basic usage concepts are briefly described in Appendix E.

The customized Porteus system was dubbed D3D-Porteus referring to its place in the D3D project.

The web interface gave us a 250 MiB ISO image of Porteus v3.1 as a starting point. The parameters chosen in the Porteus system builder are listed in Table 2.1 and some of them are briefly commented in Appendix E.

Name	Value
Architecture	64-bit
Type	EFI
Boot Mode	GUI
Desktop	Xfce
Web Browser	Firefox
Word Processor	None
VoIP Client	None
Development Tools	Yes
Video Card Driver	Open Source Drivers
Printing Support	None

Table 2.1: Parameters chosen at ref. [57] when building D3D-Porteus.

A special boot mode called "D3D Workshop Mode" was configured. It enabled copying the entire live OS to RAM and executing Pronterface automatically upon boot. It specified no automatic storage of system changes. That is, any changes to files or folders while in D3D Workshop Mode were discarded upon reboot. See ref. [58] for all boot flags used and ref. [59] for explanations. Figure 2.1 shows the screen that D3D-Porteus booted into, with the Workshop Mode boot option pre-selected.

Installation instructions were compiled and published at ref. [60] to help hosts create live USB drives with D3D-Porteus.

The D3D-Porteus files are hosted at ref. [61]. The ISO image used in OSE's RAW had a size of 480 MiB, and can be downloaded from ref. [62].

Programs

We packaged and included the programs listed in Section 1.4 into D3D-Porteus. Arduino IDE, OpenSCAD and parts of Printrun (Pronterface)

Figure 2.1: The screen that the tested D3D-Porteus boots into. Pressing Enter loads D3D-Porteus into RAM and starts Pronterface in full-screen mode. This screenshot was taken by booting the D3D-Porteus ISO on a virtual machine created by the program VirtualBox.

were compiled from source. Technical aspects of the compilation process is outside of the scope of this paper but the packaging process is briefly described in Appendix E.

All the D3D-Porteus specific configurations of these programs were put in a separate module called `D3D_Workshop_Configuration_64-bit_4.xzm`. These configurations were aimed to save in on the number of clicks required to upload firmware and start a test print. With the configuration in place each of these tasks took 5-7 clicks each.

No code outside of configuration files was changed.

Hardware

The workshop had 11 unassembled Folgertech 2020 Prusa i3 kits and 11 USB drives loaded with 64-bit D3D-Porteus.

Participants

The workshop had 24 participants. They were of mixed age and skill level. Marketing prior to the workshop was done through Facebook, OSE's home page and local newspapers. It targeted people with an interest in hosting workshops but no particular skill level or age.

Participants paid \$304 on average, and mean payment per workshop machine was \$608. Details on workshop economics are provided in Appendix G.

Most participants brought their own laptops, a few borrowed laptops from OSE.

2.3 Measures

A qualitative thematic analysis of web survey and interview responses was conducted. The themes that the analysed focused on was time shortage, long term participation prediction, copyability, and experiences with D3D-Porteus.

The number of successfully booted live USBs drives were counted during workshop.

Lastly, our own boot tests were summarized and compared with the workshop boot count.

The Web Survey

The web survey that users were asked to fill out after the workshop can be found at ref. [63]. A copy is included in Appendix F.

Most questions were open ended and allowed long answers. It addressed the RAW as a whole, and the main focus was measuring social aspects and satisfaction. Different aspects of copyability were also highlighted.

- Questions 4–7, 12–14 and 26 focused on overall satisfaction.
- Questions 11 and 15–17 focused on social aspects to try to predict if the RAW arrangement could initiate long-term participation.
- Questions 18–20 tried to probe copyability of the 3D printer and tool chain by asking about general level of self-confidence and insecurity associated with the assembly and toolchain.
- Question 20 was the only one that mentioned software explicitly. It asked participants if the mechanics-, electronics or software- parts of their RepRap toolchains were most likely to break in ways that they couldn't debug or repair.
- Questions 8–10 and 21–25 tried to probe copyability of the workshop as a whole by asking questions about tools, support and economic feasibility.
- Self-rated participant enthusiasm/enjoyment was also collected through the web survey.

To better understand details of the usage problems that participants had with D3D-Porteus, instructors were asked technical questions via a series of emails. These emails focused only on software and were unstructured.

Boot Testing

Any laptop that booted into a usable desktop with a functioning screen image, touchpad and keyboard on first try with the 64-bit version of D3D-Porteus were considered unproblematic. Laptops with 32-bit processor architectures were tested with a 32-bit version of D3D-Porteus but were considered problematic even if the 32-bit version worked.

2.4 Procedures

The execution of the RAW and the subsequent web survey and interviews was the main procedures. The transmission of D3D-Porteus to

OSE, copying of D3D-Porteus onto multiple USB drives by OSE and the separate testing of D3D-Porteus were minor procedures.

Transmission of D3D-Porteus to OSE

Transmission of D3D-Porteus packaged in one ISO file via HTTP (simple web link) was the preferred method of transmission. Sharing the ISO file via Dropbox was used as backup transmission solution.

Workshop Execution

The workshop was conducted on March 19, 2016 at the Kauffman Foundation Conference Center, Kansas City.

It started with the RepRaps unassembled, almost all screws unscrewed, almost all wires disconnected, some wires not soldered and with no firmware uploaded on the microcontroller. The extruder and the microcontroller board came pre-assembled from the kit supplier.

Participants were instructed to first assemble the mechanics, then wire and solder the electronics, and finally setup the software. Mechanical assembly was subdivided into pedagogical modules and a large fraction of the mechanical assembly had video instructions. The electronics assembly was instructed through a document with text and images. D3D-Porteus was explained orally to all participants at the same time and there were no videos or documents with software instructions.

Booting and using D3D-Porteus was a separate step at the end of the workshop. Instructors held a common walk-through on how to boot and use D3D-Porteus at 18:00, that is 10 hours into the workshop, 2 hours before the planned end. The oral instructions covered how to compile and upload Marlin through Arduino IDE, start Pronterface, connect to the printer, slice a simple 3D model, and start printing it.

The Web Survey

OSE sent an email to all addresses on the participant list, asking participants to fill out the survey. The request to fill out the survey was not repeated.

Boot Testing

Laptops were tested using the following procedure:

1. Boot the laptop and look if the boot-screen informs about which button to press to enter boot configuration.

2. If it didn't, reboot while pressing Esc, F1, F2, F10, F11 and F12 repeatedly.

3. If the laptop still didn't enter boot configuration, do web search of laptop model name + boot USB.

4. Inside boot configuration look for options called "boot override" or similar.

5. If there exist no boot override enable “legacy mode” and/or “legacy first”, disable “secure boot” and put USB first in “boot priority order” or similar.

Laptops that required more research to boot or booted into an unusable state, with severe errors in screen, touchpad or keyboard handling, were considered problematic. How to enter boot configuration was noted down.

Results 3

3.1 Pre-Workshop Copyability

HTTP transfers of D3D-Porteus ISO from Sweden to Missouri were unsuccessful as long download times resulted in timeouts. Dropbox was successfully applied as backup transfer solution.

After several tries, OSE successfully loaded D3D-Porteus onto an initial USB drive using instructions at ref. [60]. Several tries were needed because OSE's Ubuntu installations did not give users the permissions needed to write on external USB drives with VFAT allocation tables. See ref. [64] for OSE's notes on how they experienced and overcame the permissions problem.

OSE managed to copy D3D-Porteus onto 11 more USB drives from within D3D-Porteus. This was done without issues through graphical interfaces.

3.2 The Workshop

The workshop went over time by two hours and had to relocate at 20:00, when the workshop was planned to end and the conference center closed. Participant enjoyment dropped towards the end of the workshop day, as shown in Figure 3.1.

Six out of eleven participant laptops booted the live USBs successfully. Participant groups generally used D3D-Porteus successfully, except one who had missed the information that firmware upload was required.

Two usability problems increased the number of required clicks dramatically. The first problem emerged when a Marlin configuration file needed to be changed on all live USBs. This problem was amplified when Arduino IDE and Pronterface disturbed each others' USB communication. Instructors solved this by rebooting D3D-Porteus, which reverted the change in the Marlin configuration file.

Details of economic outcome is presented in Appendix G.

3.3 Thematic Analysis of Survey Responses

The survey was sent to 16 participant email addresses and received 6 answers, which gives a response rate of 37.5 %. Despite these are low response rates the survey gave some useful pointers for future development that we summarize here.

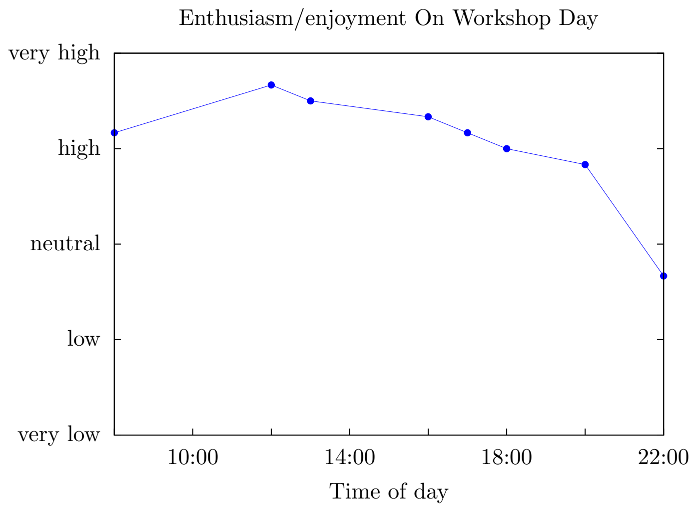

Figure 3.1: Participants were asked to recall their feeling of "enthusiasm/enjoyment" at various times of the workshop day and rate it along a five-step scale. The workshop was planned to start at 08:00 and end at 20:00 but went overtime by two hours. The plot shows the mean of their answers, assuming a linear scale between the five response alternatives. Lines between data points is not meant to imply perfectly linear development, only to highlight the trend.

Time Shortage

Participants were frustrated by time shortage, which is visible in Figure 3.1. The following comments were made on time shortage of the workshop.

> "I'm not computer or tech savvy so felt rushed.
> ...
> For me it would have been better to do the workshop over 2 days."

> "The conclusion wasn't smooth - it went overtime and had to change locations"

> **Q: What was your least favorite part of the workshop, and why?**
> "Time! [...]"
> "Relocating when we ran out of time caused an upset."

Long-Term Motivation

The time shortage limited the social interactions between participants

> "...I didn't feel like I could help others most of the time because I didn't want to fall behind the group."

> **Q: Were participants able to help each other out? Why/why not, and in what ways?**
> "Yes. Although, at times it seemed people felt rushed and got significantly ahead from others who were slower, instead of helping."

Four participants mentioned meeting other participants among their favourite parts of the workshop. One mentioned meeting instructors as a favourite part of the workshop.

Those who did not focus on social interactions when describing their favourite parts of the workshop mentioned challenge, pride and satisfaction with building the machines and using them for the first time.

> "I have a 3D printer...! This should enable me to move forward with some personal projects and skill building..."
>
> **Q: What was your most favorite part of the workshop, and why?**
> "seeing it move for the first time, "I built this""

One out of six participants mentioned prior workshop participants as someone to ask for help with eventual hosting preparations. One mentioned an instructor.

Copyability

Three participants responded that they intend to host a workshop themselves, two responded "Maybe" and one responded "No". No participants mentioned software among what they would consider challenging or needed support with if they were to host a RAW themselves. The challenging subjects and needed support that did get mentioned were very diverse.

> "Next phase design [...]"
> "resource channels, parts sourcing etc..."
> "Assistance. Motivation."
> "[...] marketing, networking, and financials [...]"

D3D-Porteus Functionality

Four out of the six respondents regarded software as the single link in their 3D printing toolchain that was most likely to break in ways that they were unable to debug or repair. Two participants mentioned software among the most challenging parts of the assembly. Electronics wiring were considered more challenging than software on average, as it was mentioned three times.

There were frustration associated with getting D3D-Porteus up and running.

> **Q: How would you rate your instructors? Did you feel you got sufficient support? What was missing?**
> "Definitely spread thin on instructor ratio regarding software, computer setup"

Two participants were unable to boot D3D-Porteus on their Macbook laptops.

> "...my older Macbook Pro didn't boot from the USB stick"
>
> "...getting the D3D live Linux ISO to boot on my borrowed Macbook did not work, it had something to do with the OS X version and EFI bootloader, so I had to borrow someone else's laptop which slowed both of us down..."

Another participant missed the firmware upload step completely and thus failed to connect with Pronterface.

3.4 Instructor Comments on D3D-Porteus

Host and head of OSE, Marcin made the following conclusive comment [65]:

> "Software remains to be addressed. Half the people had issues with the live USB, perhaps the 32 bit version could have helped - but not for certain, as nobody had an older computer."

Instructor Catarina summarised complications during D3D-Porteus usage like this:

1. Some people couldn't boot from the USB on their laptop.

2. We couldn't write to the disk.

3. We couldn't have 2 usb ports open at the same time.

Both instructors had an overall positive attitude to D3D-Porteus as a RAW tool after the workshop.

> "Thank you for developing the USB stick - it worked like a charm [. . .]"
> – Catarina

> "[. . .] the download and install process of all the software would clearly take significantly more time [. . .] we were simply unprepared in terms of helping people find the boot menu." – Marcin

3.5 Other Instructor Comments

Instructor Marcin's summary of the workshop as a whole stresses the importance of better organization of all assembly steps, not just the software setup:

> "We identified 4 key missing elements: (1) complete step-by-step WRITTEN instructions (not just the cheatsheets/QC [*quality control, author's remark*] checklists, which helped but were not sufficient), as they would have assisted progress by freeing the instructors from being bottlenecks when questions arose, (2) short looping clips of videos (5 seconds or so for each individual step), not the 30 second-1 minute videos, which would allow a person to view a step repeatedly, instead of the hitting pause and play repeatedly, and finding the right location, in a longer video; (3) each person having those videos on their own computer - via a download prior to the workshop - so a person is in full control of the procedure. (4) With all these optimized documentation assets, we concluded that the proper way to have done this would be to help each other; which happened early on, but fell apart after 3 hours. This means that we all go through the steps together, and as soon as the first person finishes a certain milestone - they immediately get up and help others, and so forth - until EVERYONE is helping the last person to finish. This forces everyone to be an active collaborator. There was a lot of time when people were bottlenecked (waiting for instructors, not having written instructions). In this proposed approach - everyone would be active."

3.6 Boot Testing

The laptops we tested outside of the workshop are listed in Table 3.1. A larger list including contributions from the Porteus community and with further links is found at ref. [66].

Table 3.1: Laptops Configured to Boot D3D-Porteus Prior to Workshop

Laptop Model	Problem?	Button	Comment
Acer M5-581TG	No	F2/F12	F12 enters “boot menu”.
Asus g74s	No	Del	Mark USB in “boot override”.
Asus Zenbook UX32A	No	F2	Hold F2 while rebooting.
Dell Precision M6500	No	F12	
HP Pavilion zt3000	Yes	F10	Old. Works with 32-bit version.
Lenovo g580	No	F2	. . . or power with “Novo button”
Lenovo SL300	No	F12	
Lenovo Thinkpad SL510	No	F1	. . . or “Thinkvantage button”.
Macbook Air from 2011	No	Option	Press and hold while powering.
Dell XPS 13 from 2016	Yes	F12	Problem with graphical mode.

Discussion 4

4.1 Result Discussion

The results show that both OSE and RAW participants had initial difficulties. OSE had a slow Internet connection and troubles with writing to USB drives. Participants had troubles booting D3D-Porteus from USB drives.

We do not consider the size of D3D-Porteus to decrease copyability if served effectively, since downloading Printrun, Slic3r, Arduino IDE and OpenSCAD for Windows, OS X and Linux would have required a total download of about 450 MiB anyways. We also consider the USB permissions bug to be a special case that did not decrease D3D-Porteus' copyability, since the ISO image can be written to DVD or CD disks, or booted inside virtual machines, on most systems that suffer from USB bugs.

We think that participants' difficulties with booting D3D-Porteus are a bottleneck both to shortening RAWs and to maintaining copyability. The experienced boot limitations were associated with live USBs and live OSes in general, and not Porteus or D3D-Porteus in particular.[1]

We had problems with meeting the following two requirements for successful D3D-Porteus usage:

1. There must exist boot instructions and boot options that fit the majority of common laptop models.

2. Participants must be provided with these instructions and options.

We supplied 64-bit live USB drives as the only boot option. There existed instructions for booting from USB drives with D3D-Porteus for only six laptop models (see ref. [68] for the exact list), and the list was not used during the workshop. Comments in Sections 3.3 and 3.5 show that problems with organizing other assembly steps also hindered optimal and focused boot instructions.

Once booted, D3D-Porteus worked, but not optimally, since instructors were surprised by two behaviours:

- Changes they made to Marlin's configuration files were discarded upon reboot.

- Arduino IDE and Pronterface could not connect to the same printer simultaneously.

[1] There exist laptop models who can not boot Porteus easily due to Xorg bugs, Xorg configurations, Linux kernel versions, or missing device drivers. We did not observe problems of this kind in our experiment. See ref. [67] for our list of known laptop models with technical boot limitations. Also note that all laptops who can boot Porteus can also boot D3D-Porteus. This is because the D3D modules are loaded after boot time and are gracefully skipped if the laptop runs out of RAM.

These usage problems might have limited copyability, but we regard the booting problem as more limiting.

4.2 Method Discussion

The difference between the no-problem-rate of Table 3.1 (80 %) and the recorded boot success rates during the actual RAW (54 %) shows the strength of the full experiment trial-and-error approach. The insight that discarded file changes upon reboot surprised users, and that this actually slowed the workshop down could also have gone missed in an isolated experiment.

The downside of doing a full experiment was costs in time and money, which led to relatively few data points. The web survey also had a low response rate. This made many results into mere pointers that require additional research to confirm.

Even if OSE directed their marketing towards the general public, it is probable that many participants had knowledge and an interest in OSE's activities from before. This might bias the impression of copyability given in the web survey responses. Even though three of six respondents planned to host RAWs themselves, we do not believe that the arranged RAW would turn every second randomly chosen American into a potential RAW host.

We started this report with plotting multiple economical indicators, among them "labour's share of GDP". These indicators were used to describe the economical development of typical Americans. The solution we proposed of increasing copyability of production machinery would maybe not change these indicators directly, even if it spread virally and changed Americans' economy drastically. This is because of how GDP is measured and how labourer is defined as well as how their share is measured. An overview of the limits of GDP is available in ref. [69], and problems related to defining "labour's share" is available in ref. [70].

5 Conclusion

The research question was

> Can RAWs aimed at the general public be shortened without decreasing their copyabiliy by swapping the steps of downloading, installing and configuring software with booting a live OS with pre-packaged software?

Our answer is yes, this is certainly possible. With a configured live OS in place, difficulties with helping participants boot it is the major bottleneck to shortening software setup times even more.

Perceived copyability of the RAW seems to not have been reduced by D3D-Porteus since three participants intend to host workshops and no participants mention software as an obstacle to hosting.

5.1 *Further Work*

To make D3D-Porteus more useful, it must be made to work with Macbook Pro laptops, as two participants in our small sample had exactly this kind of laptop. One solution would be to include the boot manager rEFInd [71] on the USB drive. It is installable by running a single script in any OS X version prior to 10.11. A few future participants might also be helped by live DVDs and CDs as well as 32-bit versions.

Boot configuration workload is multiplied with the number of different boot procedures found on different laptops. This is an Achilles heel of live portable storage media as a solution. To get away from handling boot configuration, we would need to make D3D-Porteus into a program running inside any OS. That is, we would need to run D3D-Porteus in a virtual machine.

A bundle of bioinformatics software called DNALinux [72] runs inside virtual machines created by a proprietary program called VMware Workstation Player. The VirtualBox program that was used to take the screenshot in Figure 2.1 has a FLOS base package that might be suitable to boot and run D3D-Porteus, at the cost of having to install VirtualBox.

The Docker [73] software is a promising FLOS alternative to full virtual machines, using more lightweight "containers". Docker needs to be installed and an image file must be explicitly loaded in order to use it. Docker is in beta for Mac and Windows at the time of writing.

Live OSes is a general purpose technology, so harvesting its potential requires adjusting the process in which it is used. One obvious potential is including step-by-step manuals, demonstration videos and all sorts of documentation and multimedia that participants need during an assembly workshop, not only for the mechanical assembly but also for the electronics and software parts.

It would be possible but not optimal to use the included material from another system. Booting the live system at the beginning of the RAW would give participants both the material and the time to get used to the controlled live OS environment. A design and test of a live system for such software-first assembly workshop usage would be interesting future work.

Live OSes could also provide communications channels. They could be implemented as simply as web links on the desktop, and build upon existing platforms, such as forums, social networking sites, wikis and chat programs. Such a system connected to the Internet would enable distributed and remote first-line support. Testing strategies for promoting a positive group dynamics, and remote support using the live OS as a tool would also be interesting future work.

A third potential gain from live OS usage could be avoiding Internet dependence. This could make workshop locality an easier and cheaper problem for hosts. It would also eliminate the risk of wasting time on Internet connectivity problems. It would be interesting to measure these effects on cost and time usage and find examples of situations where Internet independence would be relevant.

One aspect of D3D-Porteus usage that we did not investigate was how to best treat system changes. Since the whole system lives in RAM, we have to decide what and when to save anything to disk. The current default choice is to never write automatically to disk, which most users will experience as "nothing is saved, system is restored upon reboot". This has the advantage that unintentionally broken systems can be trivially repaired. USB systems also get independent of the underlying file system (FAT requires special saving mechanisms), and their usage get almost identical to non-writeable CD/DVD systems.

5.2 Author's Last Words And Recommendations

D3D-Porteus could offer a simplification of RAW hosting and RepRap usage if developed further. Since Porteus lack pre-compiled Computer-Aided Drafting (CAD) packages, we would prefer to re-base D3D-Porteus upon another distro. A cousin of Porteus called Nemesis is based upon Arch Linux and migt allow us to keep our `xzm` modules from D3D-Porteus.

The live OSes are a flexible packaging format. Since a host need to give participants informations anyways, it might as well be digital and directly bootable. Participants who prefer to use Docker should find a Docker image[1] on their USB drive (or DVD or CD). To keep development workload down, creation of this Docker images should be automated. Preferrably, a script should extract the relevant files from from the live

[1] It might be technically preferrable to use multiple images with linked containers, which is ok as long as the user experience is kept very simple.

ISO file or the live OS repo, and build a Docker image in one step. The Docker image should contain RAW software only, and not the complete live OS.

It would be more accessible if configuration files were stored in an uncompressed state rather than in an `xzm` module. In Porteus, two uncompressed packages are supported by default, called `changes` and `rootcopy`. Using one of these for configuration files would work, but a more elegant solution would be to make D3D-Porteus accept arbitrary uncompressed packages alongside the compressed ones.

This paper focused on the live OS but would have been more fruitful if workshop plans were considered as a whole. Since some participant groups fell behind others we do not know the exact times when they started to try booting the live OS or for how long they tried. The fact that some participants missed the oral instructions suggests that a measure of organization/chaos on the workshop day would have been useful to get compareable data. The live OS could also have helped mechanical assembly and electronics wiring if we had booted it earlier and had it loaded with instructions.

The practice of instructing mechanical assembly, then wiring, then software is found in all common RepRap assembly manuals, including the Folgertech 2020 Prusa i3 build manual. This impacted how we used D3D-Porteus. A rationale and rigorous terminology for this pedagogical practice and its alternatives would have helped us understand our own work better.

Appendix A
Acronyms

3D Three-Dimensional

BIOS Basic Input/Output System

CAD Computer-Aided Drafting

CD Compact Disc

D3D Distributive 3D printing enterprise

DVD Digital Versatile Disc

EFI Extensible Firmware Interface

FAT File Allocation Table

FLOS Free, Libre and Open source

GDP Gross Domestic Product

GNU GNU's Not Unix (recursive acronym)

GUI Graphical User Interface

HTTP Hypertext Transfer Protocol

IDE Integrated Development Environment

LZMA Lempel–Ziv–Markov chain Algorithm

MOST Michigan Tech Open Sustainability Technology

OS Operating System

OSE Open Source Ecology

PDF Portable Document Format

RAM Random-Access Memory

RAW RepRap Assembly Workshop

TAILS The Amnesic Incognito Live System

UEFI Unified Extensible Firmware Interface

USB Universal Serial Bus

Appendix B
The Clone Wars Project

Between August 2013 and up until time of writing (May 2016), a group of Spanish RepRap users under a project called Clone Wars, have registered "genealogy data" of 551 related RepRap 3D printers [74, 75]. The dataset includes usernames of owners, which lets us create a people-centered tree structure, similar to that in Figure 1.2.

The data shows a person-to-person structural virality of 5.0[1], and the tree of RepRap part transfers is shown in Figure B.1. There are 279 person nodes, 197 of them leaf nodes. The largest distance (shortest path between two connected nodes) is 12. The 7 most active transmitters supplied 89 others with RepRap parts, forming distinct clusters in Figure B.1. The dataset contains 276 person-to-person transfers.

[1] This is comparable to a perfect and full binary tree of depth 5 or roughly the median structural virality of a widespread tweet [21].

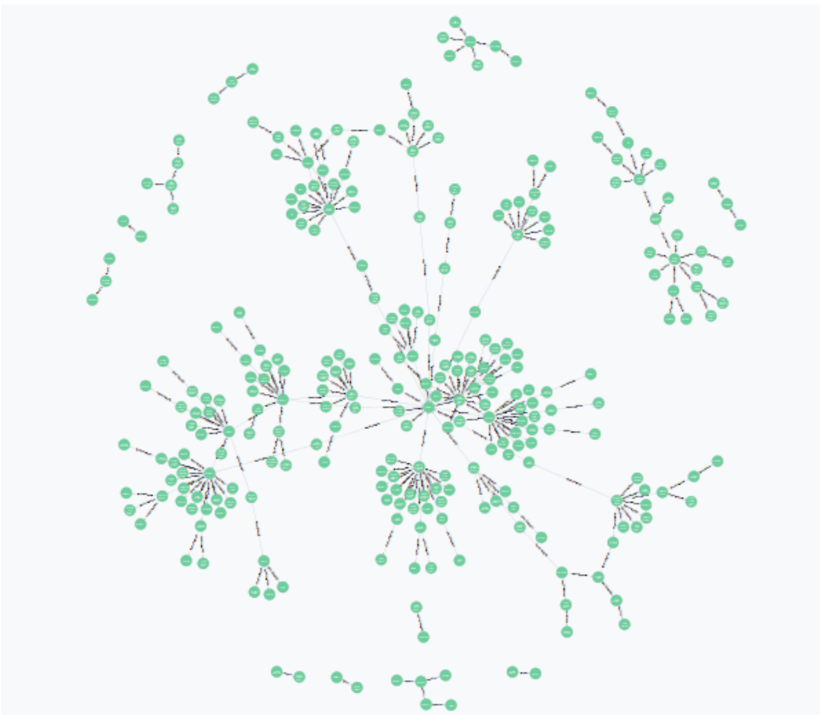

Figure B.1: Tree-structure showing how RepRap users in the Clone Wars project have transfered 3D printing abilities to each other. Each node represents a person, and edges are transfers of machine parts. Data source: [74].

Code and data behind this analysis is found at ref. [76].

Appendix C
RepRap Assembly Workshop Standards

Data presented in Table C.1 was collected by web searching for each individual data point. Sources were pictures, videos, download pages, and build instructions from RAWs, documented on the web.

Blank field means no conclusive data was found. Data generally describes workshop plans and not outcomes. That is, we do not know how many printers were working at the end of each RAW. We know that one host (Pumping Station One) did not intend to reach the stage of test printing during their RAW.

Some hosts had hosted multiple workshops. The majority workshops accepted 2-3 participants per machine and prices were almost always paid per machine, and not per participant.

The *Designer-Instr?* column tells if a designer of the used model were among the instructors. Josef Prusa co-instructed at least three workshops that used Prusa designs.

Mendel and Prusa designs are popular RAW models. Orca and Prusa are based on Mendel, while i3 Berlin, Bcn3dand Graber are based on Prusa designs.

Host software are programs for sending commands to RepRaps from PCs or laptops. The table shows two innovative host software approaches. The MOST lab have developed their own coherent software suite including a host software interface that can be displayed by web browsers [77]. i3 Berlin is host software-independent by having controller hardware on the printer itself.

Firmware Uploaders are programs who install programs on RepRap microcontrollers. The table shows innovation from the same two RAW hosts. The MOST lab uses Franklin Server (who is also their host software) in place of Arduino IDE. i3 Berlin trades away Arduino IDE installation procedures by using Cura both as a slicer and as a firmware uploader.

Only three one-day workshops were found. One of them (Pumping Station One) only taught mechanical assembly. The other two used non-Prusa designs and were as expensive as many 2- and 3-day workshops. Longer RAWs typically included introductions to theoretical aspects of 3D printing, and sometimes introductions to 3D modelling software.

The data in Table C.1 is found in spreadsheet format at ref. [78].

Table C.1: RAW Plans Data Collected By Web Search

Host	Model	Duration	Designer-Instr?	Price/machine
Garage-lab	Prusa i3	2 days	Yes	€850
Garage-lab	Orca v0.43	10-16 h, 1 day	Yes	€1090
MOST lab	Athena & MOST delta	4 days	Yes	
Humboldt	Prusa i2	4 days		
Pumping Station One	Prusa	$\geq$ 8 h, 1 day		\$300–\$400
Ohm Base Hackerspace	Any	11 h		\$0
Medialab Gdansk	Mendel	3 days		
Daan Uttien, Bart Meijer	Beagle, different sizes	1 day	Yes	€460–€799
Fablab Berlin	Prusa i3	2 days		€800
i3 Berlin	i3 Berlin	2 days, 18 h	Yes	€1345–€1545
hive76	Mendel	3 days		\$1200
Poti-Poti	Prusa i3 or SmartRap	20 h		€460
Voxel Factory	Prusa i2	2 days		
Fau Fablab, Aachen	Prusa Mendel	3 days	Yes	
RepRapBcn	Prusa Mendel or Bcn3d	3 days	Only Bcn3d	€740–€990 + VAT
Bcn3d	Bcn3d+ or Bcn3dR	3 or 2 days	Yes	€995 or €685
Media Computing Group Aachen	Prusa Mendel		Yes	€700
Botbuilder.net	Prusa i3	18 h, 2 days		\$999
Hedron Makerspace	MOST delta	24 h, 3 days		\$1000
ProtoSpace Utrecht	Ultimaker Original	2.5 days		€1795
Ballarat Hackerspace	Prusa i3	12–16 h, 4–5 days		\$900
Hackerspace Ffm	Prusa Mendel	3 days		
Workshop RepRap Recife	Graber Z35		Yes	2500–3500 BRL

Table C.1: (Continued)

Host	Year	Host Software	Firmware
Garage-lab	2012		
Garage-lab	2013		
MOST lab	2014	Franklin Server (web browser interface)	Franklin Firmware
Humboldt	2013	Pronterface	
Pumping Station One	2011	No software, only mechanical assembly	
Ohm Base Hackerspace	2013		
Medialab Gdansk			
Daan Uttien, Bart Meijer			
Fablab Berlin	2013	Pronterface	
i3 Berlin	2015	Controls on printer	Marlin
hive76	2011		
Poti-Poti	2014		
Voxel Factory	2012		
Fau Fablab, Aachen	2011		
RepRapBcn	2013		
Bcn3d	2016	Repetier Host	Marlin
Media Computing Group Aachen	2011	ReplicatorG/Repsnapper	
Botbuilder.net	2014	Pronterface	Marlin
Hedron Makerspace	2014	Pronterface, Repetier host, Octoprint	Marlin
ProtoSpace Utrecht	2015	Pronterface	
Ballarat Hackerspace	2016		
Hackerspace Ffm	2011	Bolt v0.3	Sprinter
Workshop RepRap Recife	2015	Repetier	

Table C.1: (Continued)

Host	Slicer	CAD Program	Firmware Uploader	Source
Garage-lab			Arduino IDE	[79]
Garage-lab				[80]
MOST lab	Slic3r & Cura	OpenSCAD	Franklin Server	[81, 82, 77, 83]
Humboldt			Arduino IDE	[84]
Pumping Station One	No software	No software	No software	[40]
Ohm Base Hackerspace				[85]
Medialab Gdansk				[86]
Daan Uttien, Bart Meijer				[87]
Fablab Berlin				[88, 89]
i3 Berlin	Cura and Kisslicer		Cura	[37, 90]
hive76				[91]
Poti-Poti				[92, 93]
Voxel Factory				[94]
Fau Fablab, Aachen			Arduino IDE	[95]
RepRapBcn	Slic3r	NetFabb	Arduino IDE	[96]
Bcn3d	Slic3r	NetFabb	Arduino IDE	[97, 38]
Media Computing Group Aachen	Custom FiveD/Tonokip		Arduino IDE	[98]
Botbuilder.net	Slic3r			[99]
Hedron Makerspace	Cura	Meshmixer	Arduino IDE	[100]
ProtoSpace Utrecht				[101, 102]
Ballarat Hackerspace				[103]
Hackerspace Ffm	Skeinforge	OpenSCAD	Arduino IDE	[104]
Workshop RepRap Recife				[105]

Appendix D
Approximating the Number of Prusa Machines

A very rough approximation can be made based on only two data sources; 3D Hubs and Thingiverse. The strength of their data sets is that they are self-reported by 3D printer users, so that home-copied machines are as likely as branded machines to get counted.

Model Name	Count
Prusa i3	2 352
Ultimaker 2	2 065
Replicator 2	1 412
Zortrax M200	845
Replicator 2x	817
RepRap	724
Ultimaker 1	666
Form1+	658
FlashForge Creator Pro	624
Printrbot Simple Metal	491
Makerbot Replicator 5th Gen	441
Da Vinci 1.0	431
Robo 3D printer	384
Mendel Prusa	348
Rostock MAX	339
Prusa i3 Hephestos	335
Makergear M2	308
Other	16 898
Total	30 138

Table D.1: 3D printers registered on `3dhubs.com` as of May 2016, sorted by model. Source: [106]

Many owners of 3D printers register their machines on `3dhubs.com`, who release monthly data on model number and distribution on `3dhubs.com/trends`. The model numbers of May 2016 are presented in Table D.1. Assuming that they are representative, these numbers suggest that 10 % of all desktop 3D printers are either Prusa i3, Mendel Prusa or Hephestos Prusa i3.

The popular 3D model sharing web site Thingiverse claims on their website (on 10 May 2016) that they have 867 690 "community members". We can use this number to estimate the number of 3D printers worldwide, including old, broken and unused machines by assuming that most historical 3D printer owners are Thingiverse community members and most of those who don't own a 3D printer have never register an account on Thingiverse. This assumption is obviously not perfect since one can create an account without owning a 3D printer or even own several 3D printers without creating an account. On the other hand, Thingiverse is widely used within the 3D printing community and has been since its launch in 2008.

The error caused by competing 3D model sharing sites is expected to be small. Alexa is a company who ranks web pages based on estimated unique visitors and page views [107]. It ranks `thingiverse.com` as the 2 956'th most popular website on the Internet. The nearest competing 3D printing specific 3D model sharing site is `youmagine.com`, which ranks at 93 568'th place. `alexa.com` was visited on 11 May 2016.

To check the Thingiverse based estimate, we can use numbers from the Wohlers Report 2016 [108]. It estimates that ca 580 000 3D printers under $5000 were sold before 1 Jan 2016, with 278 000 of them in 2015 alone and with doubling numbers every year from 2012 to 2015. This trajectory gets us to 780 000 machines around the time this is written (10 May 2016). Wohlers' numbers concern the number of 3D printers *sold*, a process that many home-copied RepRap machines never formally go through.

Thingiverse user count and 3D Hubs statistics suggests ca 87 000 Prusa i3, Hephestos Prusa i3 and Prusa Mendels combined worldwide. It is surprising that the Prusa i2 does not show up in 3D Hub's statistics since its popularity at its peak was comparable to the peaks of Prusa Mendel and the current Prusa i3.

Appendix E
Porteus

Basic Configuration

This section briefly comments the configuration choices listed in Table 2.1.

Extensible Firmware Interface (EFI) and its successor Unified Extensible Firmware Interface (UEFI) are interfaces between OSes and computer firmware that affect booting. Most laptops from 2011 or later support both EFI/UEFI and the older Basic Input/Output System (BIOS) interface, but many newer laptops are unable to boot OSes without EFI/UEFI support. A Porteus image with EFI support still also supports BIOS, so the EFI option only increases portability.

The desktop environment Xfce is a simple desktop environment, providing simple windows, and a simple start menu. It was the smallest available pre-packaged GUI with a compressed size of only ca 28 MiB, which was ca 10 MiB smaller than the second smallest pre-packaged GUI, LXQt. Timezone and keyboard layout was set to suit workshop participants in Missouri, USA. Firefox and open source video drivers were chosen because they gave the most free software among the preconfigured alternatives.

Modules

Porteus' modules allow users to handle files and directories with logical operations. The most common operation is called *activate*.[1] It corresponds to a logical union of the package and the root directory, as shown in Figure E.1. The reverse operation, logical difference with root, is called *deactivate*.[2]

[1] Other GNU/Linux systems call similar operations *install*.

[2] Other GNU/Linux systems call similar operations *uninstall* or *remove*.

Modules usually contain one program each, so `activate` and `deactivate` do some common install operations automatically. These are often called *activation/deactivation hooks* in other GNU/Linux package systems and include updating desktop icons, shared library links and various system caches.

The command `dir2xzm` compresses a directory into a module that can be handled by `activate` and `deactivate`. It uses the Lempel–Ziv–Markov chain Algorithm (LZMA) and the squashfs file system for compression. `dir2xzm` is rather slow because LZMA is slow, but it reaches a high level of compression compared to other popular

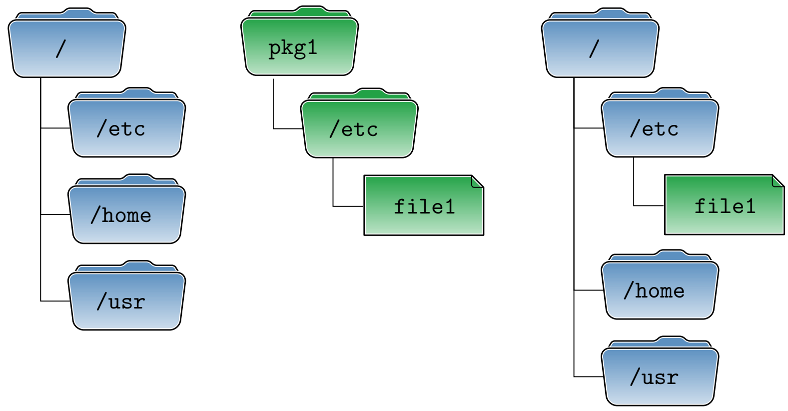

Figure E.1: Installing or activating the module `pkg1`. The leftmost tree is a randomly chosen part of Porteus' directory hierarchy. The middle tree (green) is the exact directory hierarchy found in a package called `pkg1`. The rightmost tree shows the effect of activating the package.

compression algorithms such as the Huffman algorithm [109]. Both the reverse operation, `xzm2dir` and `activate` are fast because LZMA decompression is fast.

Both `activate` and `deactivate` can be applied through the terminal or by double-clicking modules in the file browser. `dir2xzm` and `xzm2dir` can be applied through the terminal or by right-clicking modules or directories in the file browser. Porteus modules are named with a `.xzm` file extension.

Appendix F
Web Survey

Below is pasted a PDF printout of the web survey that participants responded to.

3D Printer Workshop - Followup Survey

Thank you for participating in a 3D Printer 1 Day Build Workshop by Open Source Ecology. This survey is intended to gather learnings on the workshop, so that it can be improved in the future. Further, Torbjorn Ludvigsen - remote collaborator from Umeå University in Sweden - is using this data for his Master's Thesis (http://bit.ly/1U6wbM7) - exploring the 3D Printer as a Distributive Enterprise.

This survey takes 12 minutes to complete. Your name and email is optional if you'd like to keep your answers confidential. Results will be published openly for learning purposes - for other potential workshop organizers. You can view the responses by clicking on the Survey Results link below the survey.

1. **What is your name**
   optional

   ....

2. **What is your email address?**
   optional

   ....

3. **What is your Facebook and LinkedIn page?**
   So we can connect to you.

   ....

4. **What did you like about the workshop?**

   ....

   ....

   ....

   ....

   ....

5. **What did you not like about the workshop?**

6. **What is the most important thing that you learned?**

7. **What are your improvement suggestions?**

8. **Will you be hosting a workshop in the future?**
Mark only one oval.

- Yes
- No
- Maybe

9. **If you will be hosting a workshop, what support do you need to make that happen?**

10. **If you will be hosting a workshop, what do you consider to be the biggest challenge that you will have to address to run a successful workshop?**
Venue? Marketing? Your skill set? Assistace? Part sourcing? Time commitment? Etc.

....

....

....

....

....

11. **How would you rate your instructors? Did you feel you got sufficient support? What was missing?**

....

....

....

....

....

12. **How did you feel during the workshop day? Rate your overall enthusiasm/enjoyment throughout the day.**
Mark only one oval per row.

	very low	low	neutral	high	very high
8 AM	◯	◯	◯	◯	◯
12 AM	◯	◯	◯	◯	◯
1 PM	◯	◯	◯	◯	◯
4 PM	◯	◯	◯	◯	◯
5 PM	◯	◯	◯	◯	◯
6 PM	◯	◯	◯	◯	◯
8 PM	◯	◯	◯	◯	◯
10 PM	◯	◯	◯	◯	◯

13. **What was your most favorite part of the workshop, and why?**

....

....

....

....

....

14. **What was your least favorite part of the workshop, and why?**

....

....

....

....

....

15. **Were participants able to help each other out? Why/why not, and in what ways?**

....

....

....

....

....

16. **Will you contact participants or instructors again after the workshop?**

....

....

....

....

....

17. **What adjustment of workshop arrangement would let you connect to participants and instructors more easily?**
    Some people like to talk while building slowly, others prefer assembly efficiently to free up the lunch break. Some love structured introductions, others prefer unorganized coffee breaks. Some like small groups other like big ones etc.

....

....

....

....

....

18. **What was your most challenging part of the assembly, and why?**

....
....
....
....
....

19. **Would you try to fine tune/fix your 3D printer by yourself if needed?**
Mark only one oval.

- ◯ Yes, the workshop gave me that self confidence
- ◯ Yes, but I already had that self confidence prior to the workshop
- ◯ No
- ◯ Don't know
- ◯ Other: ....

20. **What link in your 3D printing tool-chain do you feel are most fragile (most likely to break in ways that you're unable to debug and repair)?**
Mark only one oval.

- ◯ Mechanics: Straightness, flatness, movement smoothness, etc
- ◯ Electronics: Wire connections, replacement part availability, etc
- ◯ Software: Ability to convert 3D models to 3D printer instructions, ability to connect to 3D printer
- ◯ Don't know
- ◯ Other: ....

Tools, support and economic feasibility

21. **If you were to host a D3D workshop, what tools would you need to acquire first?**

....
....
....
....
....

22. **What would you estimate as the cost of tools? Is that prohibitive in terms of cost?**

23. **Do you know who you would ask for help with your preparations?**

24. **How much revenue would you need to make in order to have the interest in organizing a workshop?**

25. **In what other ways than hosting workshops would you consider for generating revenue with your 3D printer?**

26. **Please share any other comments or suggestions.**

Appendix G
Economic Outcome Of Workshop

The workshop generated a net revenue of \$4000 that were divided so that three OSE hosts, two of which acted as instructors and one who participated in preparations, earned \$1333 each. Time investment per host is difficult to approximate because all hosts had different sets of previous knowledge. A very rough table of time investment per host is given in Table G.1. Assuming that the approximation is correct within ± 12 h for all three hosts, each of them made \$29–\$61 per hour.

Economical investments prior to the workshop are listed in Table G.2. The pricing scheme used is listed in Table G.3.

Activity	Recurring?	Time consumption
Familiarization with kit	No	12 h
Placing orders	Yes	2 h
Pre-assembly of parts	Yes	4 h
Work hours on workshop day	Yes	10 h
Post Workshop Support	Yes	6 h

Table G.1: Approximate time investment per OSE host

Item	Recurring?	Total price
3D printer hardware	Yes	$3046
Tools	No	$150
USB drives	Yes	$50
Unused spare parts	No	$20
Room/space	Yes	Donated
Fuel to get there	Yes	$30
Lunch	Yes	Donated

Table G.2: Economical investments prior to workshop

Product	Description	Price
Early Registration	Assembly and ownership of 3D printer	$599
Registration		$699
Assistant	Assembly, not ownership of 3D printer	$150
2-for-1 Discount	2 participants, no second 3D printer	$0
Group rate	Negotiated via email	<$699
Remote Participation	Assembly guidance. No 3D printer	$300
True Fans	A 25 % discount for OSE sponsors	

Table G.3: Pricing scheme of workshop

Acknowledgements

The red pencil and fine tuner behind this text is my wife, Maja. Thanks for reviewing five versions of text and a thousand versions of thoughts, straightening them with your human logic and structure. Thanks for showing me the love towards oneself and others that makes writing and engineering worthwhile in the first place. Thanks for your patience.

Thanks to Marcin Jakubowski and OSE for the inspiration and the power to act. Thanks for supervising me with great enthusiasm, optimism and real world insight.

Thanks to Martin Rosvall for showing me aspects of research and good writing that stimulated a more joyful process.

Thanks to Ragnar Seton for proofreading and pointing out Docker.

Thanks to Andreas Ekberg Sjöstedt for help with project planning.

You are all sources of copyable peace and kindness.

Bibliography

[1] OECD. *GDP per capita and productivity levels.* URL: http://dx.doi.org/10.1787/data-00686-en (visited on 05/02/2016).

[2] US. Bureau of the Census. *Real Median Personal Income in the United States [MEPAINUSA672N].* URL: https://research.stlouisfed.org/fred2/series/MEPAINUSA672N (visited on 05/02/2016).

[3] University of Groningen and Davis University of California. *Share of Labour Compensation in GDP at Current National Prices for United States [LABSHPUSA156NRUG].* URL: https://research.stlouisfed.org/fred2/series/LABSHPUSA156NRUG (visited on 05/02/2016).

[4] Council Of Economic Advisers. "Economic Report of the President". In: *Washington, DC: US GPO, Annual. US Council Of Economic AdvisersEconomic report of the President* (2015). Table B-15: Hours and earnings in private nonagricultural industries.

[5] Dale W Jorgenson. "Information technology and the US economy". In: *The American Economic Review* 91.1 (2001), pp. 1–32.

[6] Dale W Jorgenson et al. "Raising the speed limit: US economic growth in the information age". In: *Brookings papers on economic activity* 2000.1 (2000), pp. 125–235.

[7] Dale W Jorgenson, Mun S Ho, and Kevin J Stiroh. "A retrospective look at the US productivity growth resurgence". In: *FRB of New York Staff Report* 277 (2007).

[8] Erik Brynjolfsson and Lorin M Hitt. "Beyond computation: Information technology, organizational transformation and business performance". In: *The Journal of Economic Perspectives* 14.4 (2000), pp. 23–48.

[9] Ian Dew-Becker and Robert J Gordon. *Where did the productivity growth go? Inflation dynamics and the distribution of income.* Tech. rep. National Bureau of Economic Research, 2005.

[10] William D Nordhaus. "Do real-output and real-wage measures capture reality? The history of lighting suggests not". In: *The economics of new goods.* University of Chicago Press, 1996, pp. 27–70.

[11] Louis-Philippe Rochon and Sergio Rossi. "Inflation targeting, economic performance, and income distribution: a monetary macroeconomics analysis". In: *Journal of Post Keynesian Economics* 28.4 (2006), pp. 615–638.

[12] Thomas Piketty and Emmanuel Saez. *A theory of optimal capital taxation*. Tech. rep. National Bureau of Economic Research, 2012.

[13] Arthur S Alderson and Francois Nielsen. "Globalization and the Great U-Turn: Income Inequality Trends in 16 OECD Countries1". In: *American Journal of Sociology* 107.5 (2002), pp. 1244–1299.

[14] Jonathan Lazar et al. "Severity and impact of computer user frustration: A comparison of student and workplace users". In: *Interacting with Computers* 18.2 (2006), pp. 187–207.

[15] Jonathan Lazar, Adam Jones, and Ben Shneiderman. "Workplace user frustration with computers: An exploratory investigation of the causes and severity". In: *Behaviour & Information Technology* 25.03 (2006), pp. 239–251.

[16] Katie Bessiere et al. "A model for computer frustration: The role of instrumental and dispositional factors on incident, session, and post-session frustration and mood". In: *Computers in human behavior* 22.6 (2006), pp. 941–961.

[17] Adrian Bowyer. "Wealth without money". In: (2004). URL: `http://www.reprap.org/wiki/BackgroundPage` (visited on 05/04/2016).

[18] Edward A Sells. "Towards a self-manufacturing rapid prototyping machine". PhD thesis. University of Bath, 2009.

[19] Rhys Jones et al. "RepRap–the replicating rapid prototyper". In: *Robotica* 29.01 (2011), pp. 177–191.

[20] Richard Stallman. "The GNU operating system and the free software movement". In: (1999).

[21] Sharad Goel et al. "The structural virality of online diffusion". In: *Management Science* 62.1 (2015), pp. 180–196.

[22] Ben T Wittbrodt et al. "Life-cycle economic analysis of distributed manufacturing with open-source 3-D printers". In: *Mechatronics* 23.6 (2013), pp. 713–726.

[23] Wade Aitken-Palmer. "A market-based approach to 3d printing for economic development in Ghana". In: (2015).

[24] John L Irwin. "Evaluation of RepRap 3D Printer Workshops in K-12 STEM". In: *age* 26 (2015), p. 1.

[25] Gary Hodgson. "Interview with Adrian Bowyer". In: *RepRap Magazine* 1.1 (2013), p. 10.

[26] Alexander Hars and Shaosong Ou. "Working for free? Motivations of participating in open source projects". In: *System Sciences, 2001. Proceedings of the 34th Annual Hawaii International Conference on*. IEEE. 2001, 9–pp.

[27] Ruth Maria Stock, Pedro Oliveira, and Eric Hippel. “Impacts of Hedonic and Utilitarian User Motives on the Innovativeness of User-Developed Solutions”. In: *Journal of Product Innovation Management* 32.3 (2015), pp. 389–403.

[28] Karim R Lakhani and Eric Von Hippel. “How open source software works:“free” user-to-user assistance”. In: *Research policy* 32.6 (2003), pp. 923–943.

[29] Yulin Fang and Derrick Neufeld. “Understanding sustained participation in open source software projects”. In: *Journal of Management Information Systems* 25.4 (2009), pp. 9–50.

[30] Erik de Bruijn. “On the viability of the open source development model for the design of physical objects. Lessons learned from the RepRap project”. In: *Unpublished Diss.(MSc Thesis). Tilburg University* (2010).

[31] Mathilde Berchon. *Interview with Josef Prusa.* URL: `https://ia600801.us.archive.org/17/items/3D-printing-podcast-prusa-mendel/02-3DPS-josefPrusa-reprap.ogg` (visited on 05/15/2016).

[32] *Marlin 3D Printer Firmware.* URL: `https://github.com/MarlinFirmware/Marlin` (visited on 05/12/2016).

[33] *Arduino.* URL: `https://www.arduino.cc/` (visited on 05/12/2016).

[34] *Slic3r – G-code generator for 3D printers.* URL: `http://www.slic3r.org/` (visited on 05/12/2016).

[35] *Printrun.* URL: `http://www.pronterface.com/` (visited on 05/12/2016).

[36] *OpenSCAD – The Programmers Solid 3D CAD Modeller.* URL: `https://www.openscad.org/` (visited on 05/12/2016).

[37] laydrop. *Downloads.* URL: `https://www.i3berlin.com/support/downloads/` (visited on 05/18/2016).

[38] BCN3D. *Plus Downloads.* URL: `https://www.bcn3dtechnologies.com/en/plus-downloads/` (visited on 05/18/2016).

[39] Bas Wijnen. *Franklin RepRap driver.* URL: `https://github.com/mtu-most/franklin` (visited on 05/20/2016).

[40] Sacha De’Angeli. *Build Your Own 3D Printer Workshop: June 18.* URL: `https://pumpingstationone.org/2011/05/3d-printing-reprap-mendel-prusa-build-june-18/` (visited on 05/18/2016).

[41] Dawn Field et al. “Open software for biologists: from famine to feast”. In: *Nature biotechnology* 24.7 (2006), pp. 801–804.

[42] *Meikian Live – Debian for makers.* URL: `http://meikian.eu/` (visited on 05/09/2016).

[43] Robert Winkler. “MASSyPup—an ‘Out of the Box’solution for the analysis of mass spectrometry data”. In: *Journal of Mass Spectrometry* 49.1 (2014), pp. 37–42.

[44] Tatsuyoshi Hamada et al. "KNOPPIXMath: a live system for enjoying mathematics with computer". In: *ACM Communications in Computer Algebra* 42.3 (2009), pp. 175–176.

[45] Fumikazu Konishi et al. "Improving the research environment of high performance computing for non-cluster experts based on knoppix instant computing technology". In: *Euro-Par 2006 Parallel Processing*. Springer, 2006, pp. 1169–1178.

[46] *TAILS The Amnesic Incognito Live System*. URL: `https://tails.boum.org/contribute/design/` (visited on 05/10/2016).

[47] John L Hennessy and David A Patterson. *Computer architecture: a quantitative approach*. Elsevier, 2011.

[48] *Windows 7 system requirements*. URL: `http://windows.microsoft.com/en-us/windows7/products/system-requirements` (visited on 05/11/2016).

[49] *Windows 10 specifications*. URL: `https://www.microsoft.com/en-US/windows/windows-10-specifications` (visited on 05/11/2016).

[50] Chengjun Wang. "Enhancing Flash Lifetime in Secondary Storage". PhD thesis. Auburn University, 2011.

[51] Simona Boboila and Peter Desnoyers. "Write Endurance in Flash Drives: Measurements and Analysis." In: *FAST*. 2010, pp. 115–128.

[52] Marcin Jakubowski. *Open Source Ecology Mission Statement*. URL: `http://opensourceecology.org/about-overview/` (visited on 05/20/2016).

[53] Cameron Colby Thomson and Marcin Jakubowski. "Toward an Open Source Civilization:(Innovations Case Narrative: Open Source Ecology)". In: *innovations* 7.3 (2012), pp. 53–70.

[54] *Distributive Enterprise*. URL: `http://opensourceecology.org/wiki/Distributive_Enterprise` (visited on 05/12/2016).

[55] *Distributive Enterprise Experiments*. URL: `http://opensourceecology.org/distributive-enterprise-experiments/` (visited on 05/12/2016).

[56] *OSE Wiki, Category:D3D*. URL: `http://opensourceecology.org/wiki/Category:D3D` (visited on 05/12/2016).

[57] *Build Porteus*. URL: `http://build.porteus.org/` (visited on 05/11/2016).

[58] *D3D-Porteus, porteus.cfg, Line 38*. URL: `https://github.com/tobbelobb/D3D-Porteus/blob/master/boot/syslinux/porteus.cfg#L38` (visited on 05/12/2016).

[59] *Cheatcodes – what they are and how to use them*. URL: `http://www.porteus.org/tutorials/26-general-info-tutorials/117-cheatcodes-what-they-are-and-how-to-use-them.html` (visited on 05/12/2016).

[60] Torbjørn Ludvigsen. *Live D3D ISO – Install.* URL: http://opensourceecology.org/wiki/D3D_Live_ISO#Install (visited on 05/12/2016).

[61] *OSE Wiki, Category:D3D.* At commit 3e664e4. URL: https://github.com/tobbelobb/D3D-Porteus (visited on 05/12/2016).

[62] *Porteus-D3D-Workshop-x86_64-v2.iso.* URL: http://vitana.se/opr3d/tbear/iso/Porteus-D3D-Workshop-x86_64-v2.iso (visited on 05/26/2016).

[63] Torbjørn Ludvigsen. *3D Printer Workshop – Followup Survey.* URL: https://docs.google.com/forms/d/136AIOvtZ9ste7SKgrEkQlrcxanXyZDaiAwDbZ98fLQQ/viewform (visited on 05/12/2016).

[64] Marcin Jakubowski. *Live ISO Creation.* URL: https://docs.google.com/presentation/d/1gZRXzfFK03bW6OEkkI-8VKIgZnziD9hu268y8tQPDHE/edit#slide=id.p (visited on 05/23/2016).

[65] Marcin Jakubowski. *3D Printer Workshop Survey – Comments.* Mar. 25, 2016. URL: http://opensourceecology.org/wiki/3D_Printer_Workshop_Survey#Marcin (visited on 05/12/2016).

[66] Torbjørn Ludvigsen et al. *D3D Live ISO, Comuter Models Verified to Boot.* URL: http://opensourceecology.org/wiki/D3D_Live_ISO#Computer_Models_Verified_to_Boot (visited on 05/23/2016).

[67] Torbjørn Ludvigsen et al. *D3D Live ISO, Comuter Models Not Booting D3D Porteus.* URL: http://opensourceecology.org/wiki/D3D_Live_ISO#Computer_Models_Not_Booting_D3D_Porteus (visited on 05/23/2016).

[68] *Computer Models Verified to Boot, as it looked on March 17 2016.* URL: http://opensourceecology.org/w/index.php?title=D3D_Live_ISO&oldid=137664#Computer_Models_Verified_to_Boot (visited on 05/26/2016).

[69] Leonardo Palumbo. "A post-GDP critique of the Europe 2020 strategy". In: *Procedia-Social and Behavioral Sciences* 72 (2013), pp. 47–63.

[70] Alan Krueger. *Measuring labor's share.* Tech. rep. National Bureau of Economic Research, 1999.

[71] Roderick W. Smith. *The rEFInd Boot Manager.* URL: http://www.rodsbooks.com/refind/ (visited on 05/23/2016).

[72] Sebastian Bassi and Virginia VC Gonzalez. "DNALinux virtual desktop edition". In: (2007).

[73] *An open platform for distributed applications for developers and sysadmins.* URL: https://www.docker.com/ (visited on 05/23/2016).

[74] Github user algspd. *Clone Wars, May The Clone Be With You.* URL: http://maytheclonebewithyou.com/ (visited on 05/30/2016).

[75] Github user algspd. *Frontend and backend to ather CloneWars clone data.* URL: https://github.com/algspd/CWData (visited on 05/30/2016).

[76] Torbjørn Ludvigsen. *clone-analysis-scripts.* URL: https://github.com/tobbelobb/clone-analysis-scripts (visited on 05/31/2016).

[77] Bas Wijnen et al. "Free and open-source control software for 3-D motion and processing". In: *Journal of Open Research Software* 4.1 (2016).

[78] Torbjørn Ludvigsen. *RepRap Assembly Workshops Plans Data Collection Through Web Search.* URL: http://opensourceecology.org/wiki/File:Assembly_workshops_data_collection_through_web_search.ods (visited on 05/18/2016).

[79] Garage Lab. *Workshop: Wir bauen einen RepRap-3D-Drucker.* URL: http://garage-lab.de/Veranstaltung/workshop-wir-bauen-einen-reprap-3d-drucker/ (visited on 05/18/2016).

[80] Garage Lab. *Orca v0.43 3D-Drucker Bauworkshop.* URL: http://garage-lab.de/Veranstaltung/orca-v0-43-3d-drucker-bauworkshop/?instance_id=2498 (visited on 05/18/2016).

[81] Joshua Pearce & others. *Athena Build Overview, Assembly.* URL: http://www.appropedia.org/Athena_Build_Overview#Assembly (visited on 05/18/2016).

[82] Joshua Pearce & others. *Delta Build Workshop.* URL: http://www.appropedia.org/Delta_Build_Workshop (visited on 05/18/2016).

[83] Bas Wijnen. *Franklin.* URL: http://www.appropedia.org/Franklin (visited on 05/18/2016).

[84] Andreas Geyer and Justin Tuttle. *Humboldt 3D printer workshop.* URL: http://www.appropedia.org/Humboldt_3D_printer_workshop (visited on 05/18/2016).

[85] Ohm Base Hackerspace. *RepRap Assembly Workshop.* URL: https://www.facebook.com/events/506141799484843/ (visited on 05/18/2016).

[86] Maciej Wojnicki. *RepRap workshop in Gdańsk, Poland.* URL: https://www.flickr.com/photos/mananasoko/8003192761/ (visited on 05/18/2016).

[87] Bart Meijer & Daan Uttien. *Workshop Beagle Building (Delta 3D printer).* URL: https://hackerhotel.nl/workshops/#Daan%20Uttien (visited on 05/18/2016).

[88] Wolf Jeschonnek. *Build your own 3D printer.* URL: https://web.archive.org/web/20130610080737/http://www.fablab-berlin.org/de/post-mit-bild/ (visited on 05/18/2016).

[89] Fab Lab Berlin. *Workshops.* URL: https://web.archive.org/web/20150530032520/http://www.fablab-berlin.org/events/ (visited on 05/18/2016).

[90] laydrop. *I3 BERLIN WORKSHOP.* URL: https://www.i3berlin.com/product/i3-berlin-workshop/ (visited on 05/18/2016).

[91] Jordan Miller. *Build Your Own Open Source 3D Printer.* URL: http://www.hive76.org/build-your-own-open-source-3d-printer (visited on 05/18/2016).

[92] Poti-Poti. *The Workshops.* URL: http://www.poti-poti.org/the-workshops/ (visited on 05/18/2016).

[93] Poti-Poti. *DIY: 3D PRINTER.* URL: http://www.poti-poti.org/diy-3d-printer-reprap/ (visited on 05/18/2016).

[94] François Lahey. *Build your own 3D printer WORKSHOP!* URL: https://www.voxelfactory.com/blogs/news/6601539-build-your-own-3d-printer-workshop (visited on 05/18/2016).

[95] ypnos. *Bericht vom RepRap-Workshop in Aachen.* URL: https://fablab.fau.de/node/33 (visited on 05/18/2016).

[96] reprapbarcelona. *Quieres una impresora 3D? Ven a Barcelona en febrero y móntatela tu mismo (PLAZAS AGOTADAS).* URL: https://reprapbcn.wordpress.com/tag/workshop-reprap/ (visited on 05/18/2016).

[97] BCN3D Technologies. *Assembling Workshop BCN3D.* URL: https://www.bcn3dtechnologies.com/en/workshop/1026-2/ (visited on 05/18/2016).

[98] Christoph Schwaeppe. *RepRap Prusa Mendel 3D Drucker Masterclass.* URL: https://hci.rwth-aachen.de/masterclass_3dprinter_18022011 (visited on 05/18/2016).

[99] BotBuilder 3D Printer Assembly Workshops. *Documentation.* URL: http://botbuilder.net/documentation/ (visited on 05/18/2016).

[100] Hedron Hackerspace. *Build a 3D Printer.* URL: http://hedron.technology/3dprinterfabworkshop/ (visited on 05/18/2016).

[101] Frank Visser. *Protospace Utrecht @ £mfw13 doing a 3d printing workshop.* URL: http://www.mediafutureweek.nl/protospace-utrecht-mfw13-doing-a-3d-printing-workshop/ (visited on 05/18/2016).

[102] Fablab Utrecht. *Masterclass Ultimaker.* URL: https://web.archive.org/web/20160422064732/http://protospace.nl/workshops/masterclass-ultimaker (visited on 05/18/2016).

[103] Ballarat Hackerspace. *3D Printing Workshop – Build Your Own.* URL: https://ballarathackerspace.tidyhq.com/public/events/4195-3d-printing-workshop-build-your-own (visited on 05/18/2016).

[104] Hackerspace Ffm. *3D Drucker für Wöhlerschule*. URL: http://www.hackerspace-ffm.de/wiki/index.php?title=3D_Drucker_f%C3%83%C2%BCr_W%C3%83%C2%B6hlerschule (visited on 05/18/2016).

[105] Workshop RepRap. *Workshop RepRap Pe Em Recife*. URL: http://workshopreprap.com/ (visited on 05/18/2016).

[106] *3D Printing Trends May 2016*. URL: https://www.3dhubs.com/trends (visited on 05/10/2016).

[107] *How are Alexa's traffic rankings determined?* URL: https://support.alexa.com/hc/en-us/articles/200449744-How-are-Alexa-s-traffic-rankings-determined- (visited on 05/11/2016).

[108] T.T. Wohlers and Wohlers Associates. *Wohlers Report 2016: 3D Printing and Additive Manufacturing State of the Industry Annual Worldwide Progress Report*. Wohlers Associates, 2016. ISBN: 9780991333226.

[109] E Jebamalar Leavline and D Asir Antony Gnana Singh. "Hardware implementation of LZMA data compression algorithm". In: *International Journal of Applied Information Systems (IJAIS)* 5.4 (2013), pp. 51–56.