

ISSUES IN OPEN HARDWARE

Standardisation of Practices in Open Source Hardware

Jérémy Bonvoisin*, Jenny Molloy†, Martin Häuer‡ and Tobias Wenzel§,||

Standardisation is an important component in the maturation of any field of technology. It contributes to the formation of a recognisable identity and enables interactions with a wider community. This article reviews past and current standardisation initiatives in the field of Open Source Hardware (OSH). While early initiatives focused on aspects such as licencing, intellectual property and documentation formats, recent efforts extend to ways for users to exercise their rights under open licences and to keep OSH projects discoverable and accessible online. We specifically introduce two standards that are currently being released and are seeking early users and contributors, the DIN SPEC 3105 and the Open Know-How Manifest Specification. Building on this, we reflect on challenges around standardisation in the OSH community and on relevant areas for future development such as an open tool chains, modularity and hardware-specific interface standards.

Note: Correspondence can be addressed equally to Jérémy Bonvoisin (j.bonvoisin@bath.ac.uk), Jenny Molloy (jcm80@cam.ac.uk) and Tobias Wenzel (tobias.wenzel@embl.de).

Keywords: standard; specification; compliance; certification; discoverability

Introduction

Open Source Hardware¹ (OSH), often simply referred to as Open Hardware, is an emergent phenomenon applying to physical products the alternative approach to traditional intellectual property (IP) protection that has been developed in Open Source Software (OSS) through decades of practice. As such, OSH now faces the same questions as the OSS community in its early days: can the open source development of physical products lead to sustainable, functional products and economical development? In spite of its unorthodox approach to IP, OSS has been the foundation of a highly innovative billion-euro economy and forms the basis of most digital products in daily use. The most iconic example of a successful open source software project is Linux. Since 2005, more than 13,594 people working for more than 1,340 companies have been involved in the development of the 22 millions lines of code constituting the Linux kernel. Since 2009, the development has been supported by a steady number of more than 200 companies (Corbet and Kroah-Hartman 2016). In 2019, Red Hat, one of the

companies contributing most development resources to the Linux kernel and flagging themselves as “provider of enterprise open source solutions”, acquired a 3.4 billion \$US revenue (Wonderlick and Walas 2019). These figures paint a picture of OSS as a concept that made its way from an alternative to a mainstream practice in a successful economic sector. They also underline that the complexity of OSS solutions can be comparable with those of proprietary software (approximated by the number of lines of code).

For the much younger phenomenon of OSH however, adoption by businesses remains low. The production of OSH has been mostly limited to non-commercial sectors such as grassroots communities, hobbyist markets, NGOs and academia (Troxler 2016). The complexity of hardware products developed under open source product development settings and released under open source licenses hasn’t reached the level of those developed in proprietary settings. For example, in the automotive industry, it is common practice that development teams record over 20,000 CAD file changes a month (Audi, personal communication). A previous study by the authors investigated the number of file changes observable in the repositories of 105 OSH projects hosted on GitHub selected for representing the high end of OSH product complexity available. The maximum number of CAD file changes observed for each project over their complete lifetime was 7522, with a median value of 123 and average value of 509 (Bonvoisin et al. 2018). These figures indicate that “the level of maturity of Open Source Hardware (OSH) remains far lower than that of Open Source Software (OSS)” and raises the question for several stakeholders, including policy makers,

* Department of Mechanical Engineering, University of Bath, GB

† Department of Chemical Engineering and Biotechnology, University of Cambridge, GB

‡ Fraunhofer Institute for Production Systems and Design Technology (IPK), DE

§ Institute for Biological and Medical Engineering, Schools of Engineering, Medicine and Biological Sciences, Pontificia Universidad Católica de Chile, CL

|| European Molecular Biology Laboratory (EMBL), DE

Corresponding author: Tobias Wenzel (tobias.wenzel@cantab.net)

of whether “Open Source Hardware [will] follow the path of its sibling [OSS]” (European Commission 2019).

When considering whether OSH will follow the same development path as OSS and whether an iconic project like Linux will ever emerge in OSH, the relative youth of the phenomenon has to be taken into account. With a 30-year delay compared to OSS, OSH is still a concept in the process of forming a consistent identity and a set of commonly-accepted best practices. This stands in contrast to the extensive and mature set of standards governing OSS. These are both *de facto*, i.e. adopted in practice by a community but not officially endorsed by a Standard-Setting Organisation (SSO), and *de jure*, which is defined by European standardisation legislation as “a technical specification, adopted by a recognised standardisation body, for repeated or continuous application, with which compliance is not compulsory” (European Union 2012). Standards can change their status, for example HTML began as a *de facto* standard but was formally maintained by the World Wide Web Consortium (W3C) from 1996 and became an international standard (ISO/IEC 15445:2000) in 2000.

The process of defining and standardising OSH is made difficult by the multifactorial and maybe ill-defined nature of *openness* (as discussed in Bonvoisin and Mies 2018 and; Balka, Raasch, and Herstatt 2014) and the consequent difficulty to draw a straight line between the practices referred to as OSH and other more or less related practices. A previous study from the authors published in this journal showed that hardware originators tend to interpret the concept of OSH differently when it comes to sharing hardware documentation (Bonvoisin et al. 2017). This spectrum of interpretations is further diluted by the “openwashing” discourses from industrial practice, whereby openness is considered to be a trending concept, incentivising companies to identify with a set of principles they may not fully practice (Murillo 2017). They therefore deploy use of “open” language when they do not meet what would be considered a community norm or standard for OSH. While degrees of freedom are certainly necessary for a new phenomenon to emerge, agreement on practices and definitions is also required to give that phenomenon momentum and maturity.

This article reviews the contribution of two major new OSH standards: the DIN SPEC 3105 and the Open Know-How Manifest Specification 1.0. We provide background on existing standards and describe the process of creating these new initiatives. We then identify gaps in OSH standardisation efforts, recommending further initiatives to support OSH in its transition from a niche to a more mainstream position within hardware industries.

Definitions and dimensions of Open Source Hardware

The evolution of OSH definitions and standards can best be understood by reviewing what *hardware* meant for the older field of OSS and for early OSH developers. Examples of OSH presented in the seminal book by Alicia Gibb (2014) “Building Open Source Hardware, DIY Manufactur-

ing for Hackers and Makers” are mostly electronic boards. In the early days of OSH, “hardware” was mainly understood as *electronic* hardware: the hardware on which OSS would run. In line with this, Mota’s (2015) depiction of the early days of the open hardware movement describes several attempts to set standards that did not stand the test of time as the word hardware in OSH acquired a broader meaning, encompassing diverse technologies beyond electronics, such as mechanics or textile. The first of these was the Open Hardware Certification Program, announced in 1997, which understood openness as “availability of documentation for programming the device-driver interface of a specific hardware device” (Perens 1997). Over a decade later the Open Source Hardware and Design Alliance (OHANDA) formed in 2009 with the ambition to create a “certification model and a form of registration” (Neumann 2011). An original contribution of OHANDA was to rewrite the four freedoms of the Free Software Definition (Free Software Foundation 2019) by changing the term “software” into “device”. While this was a conceptual leap, one limitation was that the freedoms did not explicitly state the possibility of making or replicating the device. Within a year, this limitation was overcome in the first version of the currently widely accepted definition of OSH, maintained by the Open Source Hardware Association ([Open Source Hardware Association 2016](#)), which arose following the “Opening Hardware” workshop in New York in March 2010 and has now reached version 1.0 (OSHOWA 2020). The “Open Source Hardware Statement of Principles 1.0” and “Open Source Hardware (OSHW) Definition 1.0” cover the different dimensions of Open Hardware as identified by Balka (2011): transparency (right to study), accessibility (right to modify and distribute), and replicability (right to make and sell).

Within this broad definition, OSH has gradually evolved over the last decade into a field that now spans machines and electronic boards, 3D printing as well as machine tools (predominantly desktop machine tools), vehicles (predominantly bicycles), robots, medical equipment, electricity production, agricultural machines, toys and games, optical products, musical instruments, clothing (Bonvoisin et al. 2016), and even materials, wetware, DNA, modified biological cell lines and so on.

The definitions and initiatives listed above cover some but not all dimensions that are frequently cited in the OSH community as key to open hardware success (Murillo et al. 2019; Murillo, Molloy, and Dosemagen 2017; Bij et al. 2013; Heikkinen et al. 2020), as we will further discuss in this article. These dimensions include modularity (to enable modification and reuse), interoperability (ability to combine different OSH parts), licencing, and openness through the entire product lifecycle (Oberloier and Pearce 2018; Boeshaghi et al. 2019; Gavras 2018). The definitions nevertheless guide various ongoing standardisation initiatives, including the DIN SPEC 3105 and the Open Know-How Manifest Specification 1.0 which are the focus of this article. In order to better contextualise those two initiatives, we will first introduce some other key tools that build on the definitions above to create standards

for licensing, documentation and recognition of OSH projects.

Licences, certification and documentation structure

Generally speaking, definitions enable alignment of various standards, provide a benchmark for compliance and generate boundaries for a field. For example, the OSHWA-hosted Open Source Hardware Statement of Principles and OSHW Definition provided important touchstones for OSH actors developing standardised open hardware licensing options such as the CERN-OHL (Powell, 2015).

These documents can also be used as a means to formally identify hardware that meets the community definition of open source hardware. Since 2018 OSHWA provides an OSH self-certification scheme based on transparent licensing of parts and products. As would be expected given the shared history and common originators of the initiatives, the OSHWA certification is also built around the OSHW Definition. As a self-certification process that is broadly focused on licensing and aspects of intellectual property, the OSHWA certification does not contribute to standardisation of other dimensions of OSH such as modularity, interoperability, or the particularly active area of standardisation of documentation formats. OSHWA members may review the quality of documentation of submitted projects as part of the certification process but the criteria used are not formalised, leaving a gap for further standards and certification development. The same approach is adopted by the *Journal of Open Hardware* to review hardware metapapers with scientific criteria in mind. While both processes are based on internal guidelines and checklists, they are neither formally aligned nor universal.

There are numerous *de facto* and *de jure* standards for hardware documentation. For example, oManual is an XML standard for documentation and manuals that was approved as the IEEE 1874 Standard for Documentation Schema for Repair and Assembly of Electronic Devices in 2013 (oManual 2020; IEEE 2014), and subsequently integrated into a commercial documentation platform.² oManual is also compatible with the more general Darwin Information Typing Architecture standard and had the goals of enabling documentation portability through an XML schema as well as readability through the integration of annotated images and a linear narrative. However, it was not designed specifically for OSH. Its linearity and the fact that all current implementations are proprietary limit the standard's use in the OSH community.

This has led to initiatives such as DocuBricks,³ a documentation standard devised specifically for OSH and with a focus on modularity and interoperability. One goal of the DocuBricks format is to encourage functional explanation of modules (called “bricks”) by explicitly naming them in a project. A brick can contain files and instructions, and represent the top-level project, a hardware module or a separate aspect of the documentation such as detailed guidance on how to modify design files. Importantly, the modularity of DocuBricks enables licence interoperability for conflicting share-alike licenses. This is achieved

by defining the licence as a property of each documentation brick separately. Where the licence is not exclusively defined in the top-level brick, a certain sub-brick in the documentation (e.g. forked from another project “X” or with a software licence) is interpreted as the derivative of licenced work X instead of all bricks collectively, and can be presented alongside a brick with conflicting licence “Y” in the overall project documentation. DocuBricks is also based on an XML schema, but is incompatible with the IEEE 1874 Standard due to the divergence in design goals. There are further, more recent initiatives working towards their first beta-release such as GitBuilding,⁴ Hardocs⁵ and likely others not yet known to the authors. Both GitBuilding and Hardocs aim to integrate Open Hardware documentations with Git-based repositories allowing version control.

While these different documentation tools address different community needs, there are similarities that would allow for standardisation e.g. use of the same terminology such as “part” or “step”, and schema compatibility to allow inter-conversion between different documentation formats. Schema conversion compatibility would allow content to be hosted on different kinds of repositories and edited in different ways according to user preference, which might differ e.g. between the original team and a forking team. Work to build a consensus has just started between GitBuilding and DocuBricks and invites wider participation.⁶ Major documentation hosting platforms used by OSH projects do not currently use a documentation or content standard, which may either be due to a lack of development effort or due to the preferences of the largely hobbyist-focused target group.

Transparency through enforceable content – the DIN SPEC 3105

So far, we can see that efforts to shape the contours of what “open source hardware” means have primarily focused on legal aspects of licensing. This has been supported in parallel by efforts to improve the availability of resources to reproduce and build upon OSH, mainly focused on documentation format. While these efforts contributed to frame and practically support OSH development practices, they left open the question of documentation *contents*. This aspect is addressed by the DIN SPEC 3105, extending the OSHW Definition to establish a clear OSH identity based on sharp definitions and enforceable compliance i.e. precise and objective criteria delineating what OSH is from what it is not (Bonvoisin et al. 2017). Such criteria are required for OSH practitioners to establish a consistent public discourse about OSH and the general sense of trust required by businesses to take up the concept and scale it up to a wider level.

DIN SPEC 3105 “Open Source Hardware” is a standardisation effort initiated by Open Source Ecology Germany e.V. and officially launched by the German Standardisation Organisation DIN e.V. (Deutsches Institut für Normung e.V.) April 2019 (DIN 2019). The project was led under DIN's “Publicly Available Specification” procedure (PAS) delivering a public document that can “be used as a basis

for a full standard". It involves "smaller, more agile working groups" and is an "effective marketing instrument [...] widely accepted by customers and potential partners alike". While its formulation is also community-led, DIN SPEC 3105 is the first OSH standardisation effort that is on the path to becoming a *de jure* rather than *de facto* standard.

The standard contains two parts:

- DIN SPEC 3105-1 "Open Source Hardware – Requirements for documentation" (DIN 2020a; Arndt et al. 2020) aimed at delivering an unambiguous definition of the term Open Source Hardware based on objective and enforceable criteria.
- DIN SPEC 3105-2 "Open Source Hardware – Community-based assessment" (DIN 2020b; Arndt et al. 2020) builds upon the definitions provided by DIN SPEC 3105-1 to define requirements for an assessment procedure for OSH products based on reviews by OSH community members—emulating the model of peer-review used in scientific publishing.

DIN SPEC 3105-1: Requirements for documentation

The starting point of the DIN SPEC 3105-1 is "extending the OSHW Definition 1.0, which itself extends the Open Source Definition" by setting concrete requirements regarding the content of the information to be disclosed.

The specification therefore makes significant original contributions to clarify the concept of OSH and addresses several pitfalls of previous standards:

- It breaks down the requirement made by the OSH Definition 1.0 to enable "anyone [to] study, modify, distribute, make, and sell the design or hardware" into clear, actionable ways to meet that requirement. Therewith, it defines four "Rights of Open Source" transposing the "four essential freedoms" stated in the Free Software Definition (Free Software Foundation 2019) and links them with concrete requirements in terms of documentation content. These rights are the right to study, to modify, to make, and to distribute.
- It acknowledges that OSH is not only a matter of licensing but also a matter of documentation content. It differentiates between *granting* the four rights of open source and enabling others to more effectively *exercise* these rights.⁷ Many people have successfully exercised rights to reproduce and build upon OSH products but the DIN SPEC 3105-1 addresses the issue that existing standards have limited guidance on what documentation should contain in order to ensure compliance.
- It acknowledges that the content of the documentation to be shared depends on the hardware technologies embedded in the product under consideration (e.g. electronic vs mechanical hardware). To account for these dependencies, the specification only dictates generic requirements in terms of documentation contents. These contents are in turn specified by so-called "Technology-specific Documentation Criteria" which

are specific for each technology.

- It acknowledges that the content of the documentation also depends on the audience targeted by the documentation. It is not practicable to allow "*anyone* [to] study, modify, distribute, make, and sell the design or hardware" since not everyone has the same capacity to read and make use of product documentation. Instead, documentation is required to address a defined group of *recipients*. A documentation must provide at minimum the information that specialists in the relevant field of technology would require to exercise the four rights of Open Source Hardware. This is analogous to the requirement in patent law that applicants must provide sufficient information for implementation by those "skilled in the art". Projects can of course exceed this minimal expectation of documentation detail and completeness and benefit wider users by making this aim explicit.
- It adopts a product life cycle perspective, considering that "making" a product is not only about manufacturing it, it is also about using, maintaining, updating and processing it at end-of-life. It defines documentation as information allowing to operate all activities belonging to the product life cycle, from raw material extraction to end-of-life. Therewith, it removes the arbitrary and artificial barrier imposed by previous standards between production and the rest of the life cycle.

For a brief overview of the DIN SPEC 3105 contents, see Supplementary Table 1 "Definitions from DIN SPEC 3105".

DIN SPEC 3105-2: Community-based assessment

DIN SPEC 3105-2 intends to frame the application of the DIN SPEC 3105-1 in practice by defining a dedicated assessment procedure of OSH products wishing to be certified as compliant. This procedure can be implemented by any willing conformity assessment body, who will play a role akin to that of a scientific journal editor: to moderate a constructive discussion between authors and reviewers and take a certification decision based on the convergence of this discussion. The underlying assumption is that, for the specific context of OSH, peer-certification is a more trustable assessment system than self-certification and more practical than third-party certification. While self-certification does not provide enough transparency to check compliance, third party certification creates dependency on certification bodies and often involves costly processes. By vesting responsibility and decision making power in the OSH community who understands the social and technical norms of OSH, the trust that OSH developers hold in the credibility and validity of the assessment and certification may be higher.

Open access and collaborative future development

The first release of DIN SPEC 3105 was published in July 2020. It specifies requirements for technical documentation and compliance beyond currently existing definitions and standards. Therewith, it clarifies the meaning of OSH and contributes to identity building in the OSH

community. This unavoidably goes hand in hand with higher barriers to adoption. The future will tell whether this standard found the right balance between rigour and flexibility and to what extent OSH development communities are ready to invest in documentation efforts to comply with the ideals of open source.

One aspect that will potentially play a role in adoption is the public availability of the standard. The DIN SPEC 3105 is the first German standard to be published under an open license (CC-BY-SA 4.0) and to implement an open standardisation process. Viewing and using the standard is possible without charge or permission from DIN e.V. or any further restrictions apart from copyleft and DIN's trademark protection. While the first published version is the result of a conventional "closed" standardisation process as otherwise led by DIN, an open standardisation process has been put in place that provides a mechanism for the OSH community to adjust the standard to their needs in future versions. The standard (and its possible derivatives) can be field-tested by users and then adjusted by collectively submitting a new version to DIN e.V. in order to update the official DIN SPEC 3105 release. This process aims to 1) attract and resolve much more feedback compared to the conventional standards process, 2) point out gaps where new standards are needed and 3) provide a fast and flexible way to test and adjust standards before an official release. Creating an open format at the outset for OSH is significant because in OSS, telecommunications and information technology, closed standards have been contentious (DeNardis 2011) and closed formal standards have been posited to create barriers for implementation and inhibit an open and inclusive business-friendly ecosystem (Lundell, Gamalielsson, and Katz 2018).

Discoverability through Metadata: the Open Know-How Manifest Specification

While DIN SPEC 3105 addresses the content of OSH documentation, including a minimal set of metadata for a documentation release, it remains platform-agnostic and does not adopt any formal metadata and data schema. The question of metadata schema is addressed by another initiative bringing an additional layer of standardisation to facilitate sharing Open Hardware documentation: the Open Know-How Manifest Specification.

The Open Know-How Manifest Specification is an initiative from the Open Know-How Working Group, who focus on standardised protocols for the exchange of OSH related information online. The initiative began with the observation that OSH requires new networks of tools and information to ensure that users can find and access appropriate hardware designs, local physical tools to make them, and potentially customers. It was established in 2019, bringing together around 20 contributors from academia, NGOs, companies and OSH community groups with the objective of issuing standards to address three levels of sharing OSH documentation "Know-How":

1. Discoverability, i.e. the ability to find the documentation regardless of where it resides on the internet, in part through making documentation accessible to

web crawlers;

2. Portability, i.e. the ability to share documentation on different platforms or to transfer it from one platform to another;
3. Platform interoperability, i.e. the ability to update, relate and join documentations across different online repositories and documentation formats.

The first standard released by the working group addresses discoverability and is called the Open Know-How Manifest Specification 1.0⁸ (Open Know-How Working Group 2019). The working principle of this standard is creating an additional file, called a "manifest", for each OSH project documentation, which contains metadata relative to the documentation and to other files commonly shared alongside technical documentation, e.g. license terms and the descriptive "readme" file.

The standard delivers a template formatted in YAML markup language (Ben-Kiki, Evans, and dot Net 2019) which is both machine-readable and easy to modify for non-expert users. The manifest file covers the following metadata:

- basic information such as contact person, licence, language and the locations of (i.e. links to) documentation, bill of materials and project homepage;
- descriptive information such as intended use, keywords and development stage;
- locations of (links to) some more advanced information such as risk assessments, quality control, or maintenance protocols;
- the ability to make basic relationships between OSH items more transparent e.g. by using "version", "variant-of" and the relationship term "derivative-of" to indicate that the design is derived from another and providing a link to that documentation.

This content makes manifest files fully compatible with the requirements from DIN SPEC 3105. For a more detailed overview of the fields defined by the Open Know-How Manifest Specification 1.0, see Supplementary Table 2, "OKH manifest fields".

The manifest file may be written by hand by the authors or it may be auto generated by documentation repositories if the corresponding documentation is sufficiently structured, e.g. using the DocuBricks format (note that currently only projects using one licence per project are fully compatible with the manifest).

The manifest promotes discoverability of OSH because its machine-readable format enables search engines and web crawlers to index the metadata wherever it is located on the internet. The manifest file links to the documentation and can thus be placed in the documentation repository or in another dedicated location. The project has set a goal of 500 OSH projects adopting the Open Know-How Manifests by end of 2020 and is actively recruiting projects to publish a manifest and index it in their demonstrator, which currently lists around 400 projects.⁹

Like DIN SPEC 3105, the standard has been a collaborative effort. It has open channels on the open standard

editing platform StandardsRepo¹⁰ where feedback and improvements can be suggested by raising an issue or proposal.

The proposed series of standards from the Open Know-How Group aims to build an “Internet of Production” for OSH, a network enabled by open exchange protocols to locally share and access OSH documentation, as well as needed tools and information to use it. It is the vision of the Group that the creation of appropriate standards will provide a basis for such essential infrastructure for OSH which ensures that diverse projects and platforms can exchange information effectively, and to avoid potential vendor lock-ins for the online infrastructure on which the community relies.

Discussion and outlook

Together with existing definitions and *de facto* standards, the initiatives described above i) further define standards for OSH documentation content; and ii) promote discoverable, interoperable metadata. This provides a basis for building transparent certification processes and trust of OSH amongst a wider audience. Nonetheless, the initiatives still only partially address the multifaceted concept of hardware openness as encountered within OSH communities of practice. As Gavras (2018) highlighted: both licensing terms and documentation contents are not “structural properties of the source code”; they are not “inherent properties” of a given piece of hardware that is a candidate for the OSH label. How open a piece of hardware is in terms of design—for example how interoperable with other systems it is or how it facilitates diagnosis and repair—is not addressed by any of the standards cited here but has been considered for specific technology areas within the OSH community. For example, the Open Design Manifesto (Kadushin 2010) defines “open design” as design that is “produced directly from files by CNC machines and without special tooling.” In this case openness is considered in terms of product architecture and in the context of voluntary limitations on the means of production. This raises questions on whether there are inherent properties of hardware that make it more or less amenable to standardised “open” practices and to what extent there are gaps that are not addressed by existing efforts. We attempt to summarise these properties below.

Modularity

Modularity is one of the inherent properties of hardware which is often mentioned as a key enabler in hardware openness. The appetite for modularity in OSH communities is reflected in the lego-like design style of some iconic projects like Openstructures, Wikihouse or XYZ Cargo-Bikes, and which is more generally characteristic of the DIY movement which shares overlapping values with the open source movement. One of the arguments behind the importance of modularity in open source is that it is “a form of task decomposition” (Dafermos and Söderberg 2009) and therefore influences organisation of production and development. For Kostakis and Papachristou (2014), modular product design is one of the key condi-

tions for the emergence of commons-based peer production. Product structure and organisational structure of the development team also go hand in hand, as stated in the often-cited “mirroring hypothesis”, according to which “organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations” (Conway 1968). MacCormack (2012) empirically confirmed that modular design is characteristic of the FOSS development model in general and that products developed by loosely-coupled organizations are more modular than those developed by tightly-coupled organizations. It is reasonable to expect such correspondence between product and organisational structure in OSH. The influence of product structure on the product development process is of importance considering that the very concept of open source is generally interpreted as a product development model, and not only as an intellectual property management scheme (see for example Gacek and Arief 2004; Raasch and Herstatt 2011; Moritz, Redlich, and Wulfsberg 2018).

However important the concept of modularity may be, it might not be sensible to integrate it as an enforceable aspect of the OSH Definition, partly because there is no commonly accepted definition of what it means for a product to be “modular” and of how to quantify product modularity (see a discussion of this question in Bonvoisin et al. 2016). Nonetheless, there is certainly value in collating further guidance for practitioners on how to leverage modular product design for reaching means of distributed development and production as well as product maintainability and upgradability, among others. Enabling a modular documentation of hardware may also support practitioners in designing modular products. Some Open Hardware sharing platforms already allow to define modules that build up into a full documentation through a hierarchical structure, e.g. Wikifactory, and XML specifications like DocuBricks were predicated on the need for modules and submodules that could be assembled into different configurations.

Interoperability

Another property of hardware that is not reflected in the standards discussed above is interoperability, or the ability for a product to interface with others in an ecosystem. Interoperability is necessary to enable modular design and to realise some of the most commonly cited advantages of OSH. There, however, are complications including interoperability within general OSH standards, which is why we consider it outside the scope of this review. For example, while the standards described in this article cover in principle all Open Hardware projects, interoperability relies on a network of technology-specific standards for communication, software and data exchange between different components and devices, in addition to physical interoperability of components such as dimensions and thread size.

One example of such a technology specific standard is the BioBrick (Shetty, Endy, and Knight 2008). It is a standard in the area of synthetic and molecular biology for the

physical composition of genetic parts. The BioBrick can be counted towards technology specific Open Hardware standards since it is concerned with physical material (DNA), even if classified as wetware and not classical electro-mechanical hardware. It specifies BioBrick “parts” as functional genetic sequences that use a defined set of sequences to interface with BioBrick “vectors” as the structural DNA backbone used to handle DNA for cloning. The interface sequences that flank the genes are recognised by a defined set of restriction enzymes, which are needed to assemble and transfer parts in a universal manner.

Open components

Instead of generalistic standards, recommendations could be made to adopt domain-specific open technical standards in OSH designs where possible. However, the term “Open Standards” also has multiple meanings (Larrouche 2014). Some, like the DIN SPEC 3105, i) arise from an open community process; ii) may be accessed and read by anybody and iii) may be used by anybody without royalty payments, which is aligned with OSH values and practices whereas others are open solely in one of these areas. Moreover, most OSH designs are not using open technical specifications at every level of their design. In reality, many OSH designs make use of common but proprietary electronics architectures and interfaces e.g. Bluetooth. Hence the new CERN OHL v2 licences introduce the concept of “available components” to describe an exception involving “parts, sub-assembly, library or code” that are available with “sufficient rights and information” and/or “as part of the normal distribution of a tool used to design or Make the Product.” (CERN 2020).

The availability of open components at all levels of OSH design may change in the future: a major area of growth and funding in open hardware are open Instruction Set Architectures (ISAs) such as RISC-V,¹¹ where the RISC-V Foundation manages the standard specifications and RISC-V software is managed by respective open source software projects. These initiatives allow users to design interoperable microprocessors that can be hard-wired or programmed (for example in field programmable gate arrays – FPGAs). Open ISAs increase opportunities for OSH projects to adopt open approaches at multiple levels of design but existing OSH definition and standards do not address the desirability or enforceability of using open ISAs as and when that becomes realistic.

OSH Development Tools

Beyond the interoperability of the hardware itself, interoperability of product development data formats is a long-lived and tenacious issue that greatly limits efforts to open up hardware development processes. The closedness of Computer-Aided Design (CAD) output file formats is an aspect that plagues data sharing in proprietary product development and also affects OSH. Standardisation efforts such as the CAD file format STEP (ISO 1994) brought practical but limited solutions to this problem. It is unlikely that the OSH movement will play a driving role in solving this complex issue among the incumbent proprietary

software providers, as they have established markets and benefit from a degree of vendor lock-in with major corporate customers. However, a workaround to this deadlock is the development of FOSS for hardware development, such as mechanical or electronic CAD software. Applications such as FreeCAD, OpenSCAD and KiCAD are gaining users and functionality in these respective fields. While there is still a long way towards a fully open source toolchain competing with the capabilities of proprietary software, great progress has been made in the last years and this may need to be considered in future standards.

Sharing materials and tools

Another important aspect of Open Hardware is accessing material and tools required to build or reuse respective products. This is a problem unique to hardware, as the hardware itself cannot simply be downloaded for use on largely interoperable personal computers or on single machines able to handle all aspects of digitally manufacturing most hardware. How these materials and tools can be more effectively shared is an important question also for the participation of resource poorer areas in the world, where local solutions might have an even larger impact, but e.g. the density of community workshops such as Maker Spaces and FabLabs are lower, as these are currently mainly concentrated in the US and Europe.¹² A related initiative to Open Know-How, is the Open Know-Where initiative still in its formation, which aims to help the community in locating tools or manufacturers for open tools close to their location in a distributed manner.

There are some aspects of sharing materials that are not related to intellectual property rights but to contractual arrangements called Material Transfer Agreements (MTAs) which are very common in transfer of biological and chemical material between academic institutions and to companies. The *de facto* standard MTA prohibits redistribution or commercial use (Rodriguez 2005). Moreover, it prevents sharing even if the material is not protected by patent or other intellectual property. The Open Material Transfer Agreement (Kahl et al. 2018) is the first material transfer agreement that fulfills minimal OSH community requirements, enabling recipients to exercise all of the freedoms associated with OSS and OSH. This is an example of the type of legal and community-building work that may be needed to facilitate the expansion of OSH to replace current institutional norms. Any solution needs to take into account the specific needs of material providers, which in this case meant that not having an MTA was not an option as organization and a contract of some form with legal clauses that limit their liability.

Hosting standards

Another future question is where to host community standards. Currently, most community-born standardisation initiatives are hosted by a community organisation website or the website of the development community without formal affiliation. In the future, it might be that standards organisations such as DIN open up further towards hosting community discussion and standards

releases, and also the IEEE standards association considers making its platform more amenable to less formally organised communities for this purpose (IEEE 2020). The experience gained in establishing the DIN SPEC 3105 will not only inform DIN e.V.'s decision on further opening of their standardisation processes but provides a practical example to other standards bodies of how they might effectively collaborate with open source communities using open approaches.

Conclusion

We have outlined how recent standardisation initiatives have refined the concept of Open Source Hardware beyond the previous focus topics of definitions, licences, and documentation formats. In particular, the new standard DIN SPEC 3105 provides enforceable criteria for documentation—the “source code” of hardware—validated in a community-based assessment process. With this approach, it aims to improve the recognisability of OSH and builds a bridge connecting industry, science and the worldwide Open Source Hardware community. The Open Know-How Manifest Specification 1.0, the other recent standard we highlight in this review, specifies a file or “manifest” to be created in addition to documentation, which is machine readable and contains project metadata and references to the documentation. This file makes OSH documentations discoverable online independently of their host repository. Importantly, both initiatives have brought in various stakeholders in the Open Hardware community to contribute to their development and facilitate adoption. Both initiatives are open for anyone's active contribution for future versions.

We have highlighted two further important dimensions of Open Hardware as modularity and interoperability, including software and data aspects. These dimensions are, however, more difficult to generalise and to evaluate in terms of enforceable criteria. They are more amenable to standardisation in their respective OSH sub-communities at the sector/domain level. Hopefully for future standards, we will see increased activity from such communities, and the spread of this type of discussion from the global community to subject groups. This process may involve subject journals which are already increasingly publishing Open Hardware manuscripts and Learned Societies could play an active role in standardisation efforts.

An open infrastructure for OSH designs is another key enabler for users to effectively exercise the rights granted under OSH licences. However, in particular the CAD toolchain for hardware design is beyond the exclusive control of the OSH community and unlikely to become more interoperable soon. A good way forward seems to be the continued support of Open Source CAD software. A consistent and powerful open toolchain would likely also change the criteria of current OSH standards towards more stringent policies. Open infrastructure may furthermore extend to cover network-access to physical tools and platforms for community standards development itself.

Licensing & rights	✓	Stable
Documentation formats	↑	In transition
Documentation contents Discoverability		
Process openness	✗	Not covered
Hardware openness		

Figure 1: Aspects of hardware openness covered by standards and standard-setting initiatives. The gear logo, “Open Source Hardware Logo” by Macklin Chaffee, is used under CC BY-SA 4.0; all text has been removed from the original and the colour has been changed to light grey.

Figure 1 summarizes which aspects of openness in OSH are covered by standards or standard-setting initiatives. Aspects of licensing are covered by the largely acknowledged OSWHA-hosted OSHW Definition and significant compatible license have emerged (e.g. CERN OHL v2). Some other aspects mentioned in this article did not reach such a large consensus so far but can be considered in a transition stage. Aspects of documentation contents and discoverability have been introduced by the standards introduced in this paper. These standards have just been released and their adoption within the OSH community cannot be predicted for now. Different and partly competing propositions have been made to formalize documentation formats (e.g. oManual, DocuBricks, Hardocs, GitBuildng). While there is no consensus or unique solution here, solutions are already at hand to support practices. Other aspects of openness in OSH are so far not addressed in existing standard. This is the case of hardware openness (including modularity, interoperability and usage of open components) and process openness (CAD tools and formats as well as general guidance for open source hardware development).

To conclude, we shall be reminded that all the covered initiatives are essentially voluntary open source communities, trying to make Open Hardware development easier and better while averting “Open Washing” or commercial lock-ins. These initiatives rely on our support by using, referencing and citing them. They also rely on feedback in order to find the right balance between efficient requirements and creative deregulation in a fast changing field. Larger initiatives such as OSWHA, GOSH and institutions such as DIN e.V. can facilitate sustainable maintenance of standardisation initiatives, which enable growth of OSH in return. While we are working on further standards, we should remind ourselves that the community grows only if we build on each other's work, and that this development is only inclusive when underrepresented communities are taken on board, so that newly developed standards work for all. We certainly need further growth in participation, efficiency, and adaptation to fulfill our vision of ubiquitous Open Hardware products and large open project collaboration in the image of our sibling Open Source Software, which enjoys a 30-year head start.

Notes

- ¹ In this article, we use the term Open Source Software/Hardware as a way to refer indifferently to open source and free software/hardware. This article does not intend to enter the debate about the differences between the *open source* and the *free* development philosophies.
- ² <https://www.dozuki.com/>, accessed 12/04/2020.
- ³ <https://www.docubricks.com>, accessed 12/04/2020.
- ⁴ <https://gitbuilding.io>, accessed 12/04/2020.
- ⁵ <https://goscommons.github.io/hardocs-guide>, accessed 02/07/2020.
- ⁶ <https://opensource.ieee.org/hardware-documentation>, accessed 02/07/2020.
- ⁷ To give simple illustrations of the difference between granting a right and enabling people to exercise this right: you may have the right to fly to the moon but may not have the effective possibility to do so; you may be entitled by law to tax benefits but may not have the capacity to overcome the administrative hurdles to claim these benefits.
- ⁸ <https://app.standardsrepo.com/MakerNetAlliance/OpenKnowHow/wiki>, accessed 12/04/2020.
- ⁹ https://github.com/OpenKnowHow/okh-search/blob/master/projects_okhs.csv, accessed 12/04/2020.
- ¹⁰ Now called Barbal <https://barbal.co>, accessed 12/04/2020.
- ¹¹ <https://riscv.org/>, accessed 15/04/2020.
- ¹² As observable from the data from <https://www.atlasofinnovation.com>, accessed 12/04/2020.

Additional Files

The additional files for this article can be found as follows:

- **Supplementary Table 1.** Definitions from DIN SPEC 3105. DOI: <https://doi.org/10.5334/joh.22.s1>
- **Supplementary Table 2.** OKH manifest fields overview. DOI: <https://doi.org/10.5334/joh.22.s2>

Acknowledgements

Jérémy Bonvoisin and Martin Häuer are supported by the European Union's Horizon 2020 research and innovation programme under grant agreement no. 869984. Jenny Molloy is supported by a Shuttleworth Foundation Fellowship. Tobias Wenzel is supported by the European Commission Marie Curie Fellowship EI3POD grant no. 664726 at the EMBL and the Fellowship Freies Wissen (3rd cohort) by Wikimedia Germany and partners.

Competing Interests

Jérémy Bonvoisin, Jenny Molloy and Tobias Wenzel are editors of the Journal of Open Hardware. Tobias Wenzel leads the DocuBricks initiative. Jérémy Bonvoisin, Martin Häuer and Tobias Wenzel are part of the DIN SPEC 3105 and Open Know-How working groups. None of the authors draw any financial advantage in their participation in these initiatives.

References

- Arndt, Felix, Jérémy Bonvoisin, Tobias Burkert, Lukas Schattenhofer, Jerry de Vos, Fabian Flüchter, Martin Häuer, et al. 2020. "DIN SPEC 3105 – Community Version – Release 0.10.0," June. DOI: <https://doi.org/10.5281/zenodo.3921841>
- Balka, Kerstin. 2011. *Open Source Product Development – The Meaning and Relevance of Openness*. Edited by Hans Dietmar Bürgel, Diana Grosse, Cornelius Herrmann, Hans Koller, and Martin G. Möhrle. Gabler Verlag.
- Balka, Kerstin, Christina Raasch, and Cornelius Herstatt. 2014. "The Effect of Selective Openness on Value Creation in User Innovation Communities." *Journal of Product Innovation Management* 31 (2): 392–407. DOI: <https://doi.org/10.1111/jpim.12102>
- Bij, E. van der, M. Arruat, M. Cattin, G. Daniluk, J. D. Gonzalez Cobas, E. Gousiou, J. Lewis, et al. 2013. "How to Create Successful Open Hardware Projects – About White Rabbits and Open Fields." *Journal of Instrumentation* 8 (12): C12021–C12021. DOI: <https://doi.org/10.1088/1748-0221/8/12/C12021>
- Bonvoisin, Jérémy, Tom Buchert, Maurice Preidel, and Rainer G. Stark. 2018. "How Participative Is Open Source Hardware? Insights from Online Repository Mining." *Design Science* 4. DOI: <https://doi.org/10.1017/dsj.2018.15>
- Bonvoisin, Jérémy, Friedrich Halstenberg, Tom Buchert, and Rainer Stark. 2016. "A Systematic Literature Review on Modular Product Design." *Journal of Engineering Design* 27 (7): 488–514. DOI: <https://doi.org/10.1080/09544828.2016.1166482>
- Bonvoisin, Jérémy, and Robert Mies. 2018. "Measuring Openness in Open Source Hardware with the Openo-Meter." *Procedia CIRP, 6th CIRP Global Web Conference – Envisaging the future manufacturing, design, technologies and systems in innovation era (CIRPe 2018)*, 78 (January): 388–393. DOI: <https://doi.org/10.1016/j.procir.2018.08.306>
- Bonvoisin, Jérémy, Robert Mies, Rainer Stark, and Jean-François Boujut. 2017. "What Is the 'Source' of Open Source Hardware?" *Journal of Open Hardware* 1 (1): 18. DOI: <https://doi.org/10.5334/joh.7>
- Booeshaghi, A. Sina, Eduardo da Veiga Beltrame, Dylan Bannon, Jase Gehring, and Lior Pachter. 2019. "Principles of Open Source Bioinstrumentation Applied to the Poseidon Syringe Pump System." *Scientific Reports* 9 (1): 1–8. DOI: <https://doi.org/10.1038/s41598-019-48815-9>
- CERN. 2020. "CERN OHL Version 2 – An Introduction and Explanation."
- Conway, Melvin E. 1968. "How Do Committees Invent." *Datamation* 14 (4): 28–31.
- Corbet, Jonathan, and Greg Kroah-Hartman. 2016. *Linux Kernel Development, How Fast It Is Going, Who Is Doing It, What They Are Doing and Who Is Sponsoring the Work, 25th Anniversary Edition*. The Linux Foundation.
- Dafermos, George, and Johan Söderberg. 2009. "The Hacker Movement as a Continuation of Labour Struggle." *Capital & Class*, January. DOI: <https://doi.org/10.1177/030981680909700104>

- DeNardis, Laura. 2011. *Opening Standards: The Global Politics of Interoperability*. MIT Press. DOI: <https://doi.org/10.7551/mitpress/8066.001.0001>
- DIN. 2020a. *DIN SPEC 3105-1:2020-07, Open Source Hardware – Part 1: Requirements for Technical Documentation*. Beuth Verlag GmbH. DOI: <https://doi.org/10.31030/3173063>
- DIN. 2020b. *DIN SPEC 3105-2:2020-07, Open Source Hardware – Part 2: Community-Based Assessment*. Beuth Verlag GmbH. DOI: <https://doi.org/10.31030/3173062>
- European Commission. 2019. “Workshop about the Future of Open Source Software and Open Source Hardware.” <https://ec.europa.eu/digital-single-market/en/news/workshop-about-future-open-source-software-and-open-source-hardware>.
- European Union. 2012. “Regulation (EU) No 1025/2012 of the European Parliament and of the Council of 25 October 2012 on European Standardisation.”
- Free Software Foundation. 2019. “The Free Software Definition – Version 1.165.” <https://www.gnu.org/philosophy/free-sw.en.html>.
- Gacek, C., and B. Arief. 2004. “The Many Meanings of Open Source.” *IEEE Software* 21 (1): 34–40. DOI: <https://doi.org/10.1109/MS.2004.1259206>
- Gavras, Kosmas. 2018. “OPEN SOURCE BEYOND SOFTWARE: RE-INVENT OPEN DESIGN ON THE COMMON'S GROUND.” *Journal of Peer Production*, no. 13.
- Gibb, Alicia. 2014. *Building Open Source Hardware: DIY Manufacturing for Hackers and Makers*. 1 edition. Upper Saddle River, NJ: Addison-Wesley Professional.
- Heikkinen, I. T. S., H. Savin, J. Partanen, J. Seppälä, and J. M. Pearce. 2020. “Towards National Policy for Open Source Hardware Research: The Case of Finland.” *Technological Forecasting and Social Change* 155 (June): 119986. DOI: <https://doi.org/10.1016/j.techfore.2020.119986>
- IEEE. 2014. “IEEE Standard for Documentation Schema for Repair and Assembly of Electronic Devices.” *IEEE Std 1874-2013*, January, 1–42. DOI: <https://doi.org/10.1109/IEEESTD.2014.6712032>
- IEEE. 2020. “About IEEE SA Open.” <https://standards.ieee.org/initiatives/opensource/index.html>.
- ISO. 1994. “ISO 10303-1:1994 Industrial Automation Systems and Integration – Product Data Representation and Exchange – Part 1: Overview and Fundamental Principles.”
- Kadushin, Ronen. 2010. “Open Design Manifesto.” <https://www.ronen-kadushin.com/open-design-manifesto>.
- Kostakis, Vasilis, and Marios Papachristou. 2014. “Commons-Based Peer Production and Digital Fabrication: The Case of a RepRap-Based, Lego-Built 3D Printing-Milling Machine.” *Telematics and Informatics* 31 (3): 434–443. DOI: <https://doi.org/10.1016/j.tele.2013.09.006>
- Lundell, Björn, Jonas Gamalielsson, and Andrew Katz. 2018. “On Challenges for Implementing ISO Standards in Software: Can Both Open and Closed Standards Be Implemented in Open Source Software?” Chapter. *Corporate and Global Standardization Initiatives in Contemporary Society*. DOI: <https://doi.org/10.4018/978-1-5225-5320-5.ch010>
- MacCormack, Alan, Carliss Baldwin, and John Rusnak. 2012. “Exploring the Duality between Product and Organizational Architectures: A Test of the ‘Mirroring’ Hypothesis.” *Research Policy* 41 (8): 1309–1324. DOI: <https://doi.org/10.1016/j.respol.2012.04.011>
- Moritz, Manuel, Tobias Redlich, and Jens Wulfsberg. 2018. “Best Practices and Pitfalls in Open Source Hardware.” In *International Conference on Information Theoretic Security*, 200–210. Springer. DOI: https://doi.org/10.1007/978-3-319-73450-7_20
- Murillo, Luis Felipe R., J. Molloy, and S. Dosemagen, et al. 2017. “Global Open Science Hardware Roadmap.”
- Murillo, Luis Felipe R. 2017. “When the ‘Open Wash’ Comes with ‘Open Everything’ – Paris Innovation Review.” <http://parisinnovationreview.com/articles-en/when-the-open-wash-comes-with-open-everything>.
- Murillo, Luis Felipe R., Pietari Kauttu, Laia Pujol Priego, Andrew Katz, and Jonathan Wareham. 2019. *Open Hardware Licences: Parallels and Contrasts: Open Science Monitor Case Study*. Publications Office of the European Union.
- Oberloier, Shane, and Joshua M. Pearce. 2018. “General Design Procedure for Free and Open-Source Hardware for Scientific Equipment.” *Designs* 2 (1): 2. DOI: <https://doi.org/10.3390/designs2010002>
- OManual. 2020. “About OManual.” <https://www.omanual.org/about.php>.
- Open Know How Working Group. 2019. “The Open Know-How Manifest Specification Version 1.0.” <https://app.standardsrepo.com/MakerNetAlliance/OpenKnowHow/src/branch/master/1>.
- Open Source Hardware Association. 2016. “Open Source Hardware (OSHW) Statement of Principles 1.0.” <http://www.oshwa.org/definition/>.
- Raasch, Christina, and Cornelius Herstatt. 2011. “Product Development in Open Design Communities: A Process Perspective.” *International Journal of Innovation and Technology Management* 08 (04): 557–575. DOI: <https://doi.org/10.1142/S021987701100260X>
- Rodriguez, Victor. 2005. “Material Transfer Agreements: Open Science vs. Proprietary Claims.” *Nature Biotechnology* 23 (4): 489–491. DOI: <https://doi.org/10.1038/nbt0405-489>
- Shetty, Reshma P., Drew Endy, and Thomas F. Knight. 2008. “Engineering BioBrick Vectors from BioBrick Parts.” *Journal of Biological Engineering* 2 (1): 5. DOI: <https://doi.org/10.1186/1754-1611-2-5>
- Troxler, Peter. 2016. “Offene Freie Technik: Was Ist Offene Freie Technik (Open Source Hardware)?” *Die Welt Reparieren Open Source Und Selbermachen Als Postkapitalistische Praxis*, 85–92. DOI: <https://doi.org/10.14361/9783839433775-006>
- Wonderlick, Stephanie, and Sarah Walas. 2019. “Red Hat Reports Fourth Quarter and Fiscal Year 2019 Results.”

How to cite this article: Bonvoisin, J, Molloy, J, Häuer, M and Wenzel, T. 2020. Standardisation of Practices in Open Source Hardware. *Journal of Open Hardware*, 4(1): 2, pp. 1–11. DOI: <https://doi.org/10.5334/joh.22>

Published: 19 August 2020

Copyright: © 2020 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.



Journal of Open Hardware is a peer-reviewed open access journal published by Ubiquity Press.

OPEN ACCESS 