



HAL
open science

Software Platforms for Inclusive Innovation

Terje Aksel Sanner, Petter Nielsen

► **To cite this version:**

Terje Aksel Sanner, Petter Nielsen. Software Platforms for Inclusive Innovation. 15th International Conference on Social Implications of Computers in Developing Countries (ICT4D), May 2019, Dar es Salaam, Tanzania. pp.218-230, 10.1007/978-3-030-18400-1_18 . hal-02285315

HAL Id: hal-02285315

<https://inria.hal.science/hal-02285315v1>

Submitted on 12 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Software Platforms for Inclusive Innovation

Terje Aksel Sanner¹ and Petter Nielsen²

^{1,2} Department of Informatics, University of Oslo, Norway
terjeasa@ifi.uio.no

Keywords: inclusive innovation, software platform, development, ICT4D

Abstract.

Software platforms present novel opportunities for innovation across heterogeneous settings, users and areas of use. We report from the case of the Health Information System Programme (HISP) that started out in post-apartheid South Africa more than two decades ago. The programme centres on the development of an open source software – called DHIS2 – primarily for decentralized public health management. Today, DHIS2 is a software platform with a significant global footprint. We contribute to literature on innovation for development, by identifying and examining processes of inclusive innovation pertaining to the longitudinal development of DHIS2. We find that a combination of long-term capacity building and knowledge sharing, consensus-based decision-making, and a modular platform architecture facilitates inclusive innovation. However, short term and project-oriented funding limits the sharing and scale-up of local innovations while the size of the venture and the heterogeneity of actors moderates inclusion in the development of core components of the platform.

1. Introduction

Digital services, applications and content can be reused and recombined and increase in breadth and value with the number of people involved in their innovation and consumption [1, 2]. Commercial platforms, such as Apple iOS, Android, Baidu and Alibaba have demonstrated an unprecedented capacity for innovation. These platforms mediate between two or more distinct groups of actors, such as app developers, content producers and consumers; or advertisers, buyers and sellers [3]. In relation to software platforms, what matters is not only technology [4], but also an ecosystem of platform stakeholders [3] and associated value creation mechanisms [5]. Yet, the platform architecture is key to a software platform's capacity to innovate. The separation of reusable generic components (platform core) from particularized services, applications and content (platform extensions) through stable interfaces facilitates distributed innovation at minimal coordination and transaction costs [3].

The rise of commercial platform ecosystems poses important questions for ICT4D research. How, if at all, can similar dynamics be cultivated around platforms for global public good? And how does the emergence of platform ecosystems shape intermediation between users and producers of innovations for development [6, 7]? Heeks et al. argue that it is currently unclear how different stakeholder relationships should best be aligned for innovation sharing and inclusive innovation in development [8]. Inclusive innovation refers to inclusion in terms of both processes and outcomes. Innovations can

be inclusive in their intended outcomes, such as improving the livelihood of the poorest of the poor, and in the participatory process of reaching those outcomes. Intermediation challenges arise with inclusive innovation at a large scale, which has recently been highlighted as an ICT4D research priority [7, 9]. In this paper, we explore inclusive innovation primarily as a process and at a large scale by asking, *how can inclusive innovation be facilitated by a software platform for global public good?*

We begin to answer this question through a longitudinal case study of software platform innovation in the Health Information System Programme (HISP). HISP is a global health information systems strengthening initiative coordinated from the Department of Informatics at the University of Oslo (UiO) in Norway. HISP aims to overcome barriers to equitable, data-driven and decentralized public health administration in developing countries. Historically, development activities in the context of health information systems strengthening have favoured short bursts of funding and technical assistance to implement standardized solutions along vertical silo structures such as HIV/AIDS, Tuberculosis and malaria rather than cross-fertilization and sharing of innovations between programmes [10, 11]. To mitigate challenges to scale up of local innovations, HISP engages in the cultivation of an international knowledge and experience-sharing network in tandem with the participatory development, implementation and use of the open source District Health Information Software (DHIS2) [12].

In the following two sections, we review relevant research on inclusive innovation and software platforms respectively. This motivates our discussion on software platforms for inclusive innovation. In section four, methods, we present the case background and our approach to data collection and analysis, which is followed by our case description in section five. Section six, provides the discussion part of the paper, while section seven reflects on the papers contributions and implications for further research.

2. Related research: inclusive innovation

Innovation for development has been studied in terms of its intended purpose, process and impact under labels such as inclusive, frugal, grassroots, pro-poor and bottom-of-the-pyramid innovation. Here, we use the term ‘inclusive innovation’, which appears to have been coined by TheWorld Bank to describe “knowledge creation and absorption efforts most relevant to the needs of the poor” [13]. Recently, ICT4D scholars have highlighted that inclusive innovations need to be inclusive not only in outcome, by satisfying the basic needs of the underserved, but also in process, through participatory production of the innovative means to reach those outcomes [14, 15]. The participatory dimension of inclusive innovation aims to bridge traditional dichotomies between users and producers [6, 7], the supply side and the demand side [8] and align top-down standardization with local innovation [7, 9]. As highlighted by [15], inclusion in innovation is also a matter of degree, with inclusivity ranging from inclusion in consumption and use of innovations; to inclusion in innovation development processes; to inclusion in defining and shaping the very structures through which innovations are produced.

Despite the recent interest in inclusive innovation in development research in general and ICT4D in particular, the potential for software platforms to facilitate inclusive innovation at a large scale has received scant attention. Braund and Schwittay provide an exception through their study of how Kiva, a web-based micro loan service, was able

to scale inclusive innovation by complementing standardization with diversification and local customization throughout its multinational network [16]. However, the Kiva study also highlights that increasing scale challenges governance of distributed innovation processes to meet idiosyncratic needs. Attempts to grapple with such challenges include absorption of local sites of learning and innovation into the core of the network and the cultivation of intermediaries who can actively support innovations [7].

A key interest in recent studies of inclusive innovation has been the role of “innovation intermediaries”, for instance in diffusing and scaling innovations [15]. The role of innovation intermediaries is particularly important in large-scale and complex innovation networks involving heterogeneous settings, users and areas of use. In such networks, intermediaries enable innovation not just through knowledge brokering between those who can redesign technology and those who understand how the technology needs to be used [7, 8], but also through direct involvement in technology appropriation and customization [17].

3. Innovation in software platform ecosystems

Tiwana defines a *software platform* as a “software-based product or service that serves as a foundation on which outside parties can build complementary products or services” [3]. However, a *software platform ecosystem* constitutes not only software modules and the interfaces between them [4], but also different groups of stakeholders that come together to generative and derive value from the platform [3]. Hence, a software platform ecosystem can be understood as a particular type of innovation network. For instance, Android and Apple iOS facilitate innovation by enabling interaction between consumers and third party application developers. Both the technical software platform architecture and the platform ecosystem needs to be carefully governed to facilitate innovation and create value [5].

Technically, software platforms comprise three key elements: core components with low variability (e.g., data storage and data access), complementary components with high variability, and stable interfaces / boundary resources between them [18]. A key aspect with a software platform is to facilitate an economy of scale of the lean and reusable core, while further development of services, content and applications can be performed by distributed third party developers with the help of boundary resources [19], such as standardised Application Programming Interfaces (APIs), Software Development Kits (SDKs), documentation and software licenses. The boundary resources allow software platforms to be open for innovation across heterogeneous settings, users and areas of use [19, 20]. Local innovation and tailoring can take place at a large scale, at the “app level”, without intimate knowledge of the inner workings of the reusable platform core (valuable ignorance) and without direct involvement from the platform core developers. If the functionality of a platform extension, such as an app, is used extensively across settings, it can potentially be absorbed into the platform core so that everyone using the platform can reuse and enjoy it.

4. Method

We report from an interpretive qualitative case study [21, 22] that builds on both authors' decade long participation in software implementation, project management, evaluation and capacity building associated with the District Health Information Software 2 (DHIS2). Observation notes and interview transcripts generated through our participation in the HISP network are our primary sources of data. The first author has been closely involved with implementation research on mobile phone-based reporting of public health data from health facilities from 2009 – 2015 [10], while the second author has coordinated initiatives to democratize requirements gathering and feature selection in the global software development process. The second author has also contributed to initiatives to use DHIS2 in new domains and contexts such logistics management in Kenya and Uganda and civil registration and vital statistics in Tajikistan [23].

The data analysis builds on both authors' re-examination of previously collected case study material. We jointly processed the empirical data through the development of data displays, including timelines and conceptual drawings. This allowed us to group volumes of data in relation to major shifts in user involvement in DHIS2 development, implementation and capacity building. We then identified candidate narratives that met three criteria. First, the authors had access to detailed and longitudinal data to support the narrative. Second, the narrative illustrated changes to participation in innovation processes pertaining to software development, implementation and capacity building. Third, the narrative illustrated the relationship between the emergent software platform architecture and large-scale participation in innovation. Out of the candidate narratives identified, we chose to include two narrative vignettes in this paper as they illustrate the relationship between the software platform architecture and inclusive innovation. Vignette one, about DHIS2 Academies; illustrate how different modes of inclusive innovation are deeply intertwined with capacity building activities in the HISP network. Vignette two, describes RCAT, an initiative to democratize DHIS2 platform development across a large and diverse community of stakeholders.

5. DHIS2, from software product to software platform ecosystem

The DHIS software started out as an MS Access-based desktop application to support tasks of public health management in three pilot districts in Cape Town, South Africa, in 1994, after the fall of apartheid. Emancipation was a central political theme, both in terms of equitable health service provision to all, irrespective of background and 'race', and in terms of empowering local health authorities to provide health services in the ways they saw most appropriate. Early versions of DHIS evolved through participatory prototyping with local district staffs [24]. Over the next few years, DHIS saw a relative success in South Africa and emerged as the national health management information system. DHIS slowly gained a foothold in other countries along with the cultivation of a network of stakeholders including universities, ministries of health, global regulatory agencies like WHO, international development funders, and the establishment of regional implementation organizations such as HISP South Africa and HISP India [12].

In 2004, due to growing computer literacy and increasing access to the Internet by many potential DHIS end users, work started on the open source and web-based DHIS version 2 (DHIS2). In the remainder of the paper, we focus on the years after the launch of DHIS2 in 2006, as this is when the traditional participatory software development

techniques, based on prototyping and close personal interaction between developers and users, met with challenges of scale in terms of the number and distribution of heterogeneous settings, users and uses. Currently, ministries of health, NGOs and donors in more than 80 countries use DHIS2 as a health management information system. There are currently more than 30,000 end users of DHIS2 in Bangladesh alone.

DHIS2 has application programming interfaces (APIs) and a number of configurable modules (bundled apps), which enables customization across domains. Through the APIs, the core functionality of the platform can be extended by custom apps that meet specific needs. Dedicated DHIS2 app stores exist for distribution of both web applications and Android applications. Although a cadre of DHIS2 experts are located throughout Africa and Asia, they primarily work with short-term assignments in large donor funded implementations of the software. Few incentives are available for local DHIS2 experts to explore directions not directly relevant to the international donors. There are examples where DHIS2 is appropriated and used in other domains than health such as road safety in Tanzania, the new year accident reporting system in Vietnam [25] and the civil registration and vital statistics system in Tajikistan [23]. While these innovations successfully support local needs, they remain isolated branches of DHIS2 that are not taken up elsewhere in the HISP network.

The global HISP network comprises national and/or regional organizations, many of which have been established by the more than 40 information systems PhDs that have graduated with the HISP UiO research programme. The national and regional HISP organizations provide day-to-day support to national ministries and engage in local DHIS2 capacity building through DHIS2 Academies. To date, more than 90 DHIS2 Academies have been conducted with more than 4500 trainees on DHIS2 analytics, system configuration and app development. For example, a DHIS2 web applications development academy was held in Dar es Salaam, Tanzania in November 2017.

Vignette 1: DHIS2 Academies, 2010-present

This vignette concerns the DHIS2 Academies and illustrates synergies between large-scale representation of users in software requirements gathering and top down standardization through capacity building. Since 2010, national and regional DHIS2 organizations, with coordination from HISP UiO, have arranged a number of workshops, called DHIS2 Academies. Academies are held regularly in Africa, Asia and Latin America, and have diversified to cater for training needs of software implementers and customizers at different experience levels. The Academies mainly consist of training sessions, but they also provide an arena for national and regional DHIS2 implementers to keep up to date with recent software features, share experiences, discuss technical challenges, build professional relations, and give feedback to core developers directly.

During a DHIS2 Academy in Mombasa, Kenya in 2012, a group of 12 specialists from different countries in east Africa discussed requirements for mobile phone-oriented features in DHIS2 for three hours. Regarding a feature that had been difficult to implement across multiple contexts, a DHIS2 developer shared his positive sentiment.

To create new functionality, we really need a proper use case, a real one, so we can communicate. We cannot just [gesture showing something coming out of the air]. We have been thinking

about this [feature] for the longest time, from the very beginning, but we never found good examples of what would be beneficial for the health worker, as we have discussed today.

In particular, generic functions that have local variations in workflow have been difficult to develop without discussions about specific examples of use, with one example being integrated disease surveillance and response (IDSR). IDSR is the process for discovering and responding to disease outbreaks. IDSR has a number of WHO guidelines, but effective disease identification and response still requires local adaptation of ‘best practices’. During another Academy session in Mombasa, a discussion on usage of SMS alerts to IDSR response teams opened the requirement scope to include many other non-IDSR functions that could benefit from SMS alerts, such as SMS reminders to health workers to improve routine health data reporting. One of the DHIS2 Academy coordinators thanked the group by saying:

This is very good. We have been thinking about some of these cases, but very narrow. It is clear that it is a whole area with different [...] it has more than one use case.

The academy discussions mentioned above helped to clarify design features that despite international standards and guidelines had local variations and idiosyncrasies. During the same session, a coordinator tried to uncover local variations by saying:

I am wondering about IDSR. When there is an outbreak, to which extent is the process based on these structures processes versus people picking up the phone, coordinating on the phone, and breaking out of the protocol.

A long discussion followed, where participants shared their views on the organizing of IDSR in their respective countries. In addition to providing valuable input to the DHIS2 development team, participants commented that it was useful to discuss use across nations. Interestingly, despite the consensus during the session on a number of DHIS2 software requirements to support IDSR, these features were not implemented in the software until 2014, when a large international NGO paid for their development as part of its DHIS2 adoption. This highlights the challenge of balancing (bottom-up) inclusive innovation with the prioritisation of software requirements in a large venture.

At an Expert Academy in Oslo in 2015, with more than 80 participants from ministries of health, international NGOs, independent DHIS2 consultants, and experienced country implementation representatives, a session was held on the DHIS2 software development process and how various stakeholders could participate in it. Although the DHIS2 development team at HISP UiO would continue to make final decisions about standard release features, the software development coordinator listed three ways for DHIS2 community members to propose new features: 1) write on the developer mailing list, 2) participate at DHIS2 Academies or 3) write and upload complete requirement specifications on an issue-tracking platform (i.e., Jira). These three modes of participation are not equally accessible to all users and organizations that rely on DHIS2 in their daily activities. Furthermore, the lead developer presented six qualitative criteria employed to prioritize functionality requests: perceived usefulness to the wider DHIS2 community, perceived software development workload, perceived benefit beyond what is already possible with the software, interdependencies with other functions and modules (risk), expected level of participation and feedback from the requesting party, and availability of earmarked funding. Based on these criteria, it is not surprising that well-

funded late adopters of the software such as international NGOs have become increasingly influential in shaping the DHIS2 software development roadmap.

International NGOs are able to invest in time with core developers and project managers at HISP UiO to make sure their requirements are realistic, well understood and correctly formatted. Furthermore, these stakeholders can hire independent DHIS2 consultants to implement their specific in house requirements directly.

A recent trend has been the development of third-party apps by international NGOs, who use DHIS2 as an internal management information system. Many of these apps could be useful to other organisations such as ministries of health. However, the broader HISP network has not taken up most such custom apps and many of them are not shared on the DHIS2 app stores. In part, this is because sharing generates additional responsibilities and costs associated with making app functionality generic and maintaining them over time beyond the single organization's needs. These and other challenges to large-scale inclusive innovation informed HISP UiO to implement an explicitly democratic requirement specification process, albeit with a limited number of country representatives, which we describe in the second vignette.

Vignette 2: RCAT – Democratizing requirements selection (2017 - present)

Every new release (quarterly) of DHIS2 centres on extending the platform core and the bundled apps with generic functionality relevant across different implementation sites and organizations. However, HISP UiO has increasingly committed to contractual obligations to meet software requirements directly funded by international NGOs and donors. In 2017, HISP UiO established an initiative to deal with a concern that requirements from NGOs and donors were prioritized over early adopters of DHIS2 such as ministries of health with limited technical capacity and funds. On behalf of ministries of health in less developed economies, national and regional HISP-groups mediate DHIS2 software requirements based on active involvement in implementation and customization projects. While the HISP-groups have technical expertise, maintain strong relations with HISP UiO and legitimately represent ministries of health in the countries they are working, they have repeatedly expressed concerns that their software requirements are neglected. This was the starting point for establishing the DHIS2 Roadmap Country Advisory Team (RCAT).

The second author initiated RCAT based on three key concerns raised and reiterated by the HISP-groups over several years. First, there is a lack of transparency in the process of defining the DHIS2 software development roadmap and prioritizing requirements. Second, requirements prioritized in the roadmap are often postponed just before a new release without any discussion. Third, the impression among the HISP-groups was that HISP UiO had made a promise to involve them more actively into the roadmap process and commit to their particular needs. A reflection document summarized these concerns and suggested the aim of RCAT to “assure that country requirements are considered for the DHIS2 roadmap”, and proposed that “representatives (1-2 people) from each HISP group will offer advice on priorities before the work starts on release 2.28”.

The virtual RCAT team was initiated with a Skype meeting with representatives from HISP-groups in Nigeria, India, Vietnam, Rwanda, Uganda, West Africa, South Africa and Bangladesh. Several issues were raised during the first meeting. First, the

mandate of RCAT was discussed with a particular focus on what kind of commitment HISP UiO would give to prioritized requirements. Second, questions were asked whether this process also would involve the delegation of certain software development tasks to the different regional/national HISP-groups. Third, the representatives from the HISP-groups raised the issue of requirements specified in Jira commonly being postponed and the lack of an overview over the origin of prioritized requirements (i.e., NGOs, donors or ministries of health).

Based on the initial meeting and the RCAT coordinator (second author) discussing with the HISP UiO team, RCAT decided to organize work around requirements described in Jira. Based on one-to-one discussions with RCAT team members and email conversations with the whole team, a seven-step process of defining a prioritized list of requirements was outlined: 1) New requirements are shared and vetted on the DHIS2 user mailing list before suggested as improvements in Jira. 2) Each RCAT member group provide a list of relevant issues from Jira for the next release. 3) The coordinator compiles a common list of issues with all relevant issues. 4) Each RCAT member group give each issues on the common list a priority from zero (lowest) to three (highest). 5) The HISP UiO software developer team scrutinize the prioritized list with a focus on software dependencies and resource demand and provide change suggestions. 6) Based on the developers' feedback, RCAT compiles a final prioritized list of requirements. 7) HISP UiO considers the final list and decides which issues to put on the roadmap.

In practice, the member groups entered their prioritized requirements in a dedicated template (stage 2). The requirements had a short textual summary and a reference to Jira (Jira Issue Key). The number of requirements entered by each RCAT member group ranged from zero to 31. Some requirements were new, but the majority already existed in Jira. Some of the new requirements were sketchy and not properly described. When compiled, the list consisted of 69 requirements. The RCAT member groups then prioritized the 69 requirements on a scale from zero to three. Only two of the HISP-groups returned a prioritized list. One of them prioritized three requirement, while the other prioritized 33 as top priority (3), 27 as priority level 2 and 4 as priority level 1. The RCAT coordinator did not find this ranking very useful because 60 requirements were highly ranked based on input primarily from one group. The HISP UiO developers could not prioritize all 60 requirements. More importantly, the requirements did not fairly represent the needs of the different member countries.

To produce a shorter and a more representative list of requirements, the RCAT coordinator asked the groups to rank their top five requirements (1-5). Seven groups engaged in this ranking. Four groups prioritized five requirements; one prioritized two and one prioritized six. The most active group prioritized 23 requirements related to the four different countries they were currently working in. In total, 28 requirements were ranked, but only four requirements were ranked more than once. When the developer team at HISP UiO were presented with the 28 ranked requirements and an explanation of the process, they opted to look into the long list of 69 requirement. They marked one as not a requirement, 4 as already solved, and 28 as having incomplete specification in Jira. Finally, 35 requirements were completely specified. The developer team did not commit to address any of the requirements beyond promising to do their best.

The version 2.28 of DHIS2 was released in October 2017. Out of the 69 requirements, 10 were implemented. While RCAT established a mechanism for the member groups to communicate and prioritize requirements on behalf of ministries of health and their end users, its effects were limited. First, the initiative did not bring much transparency to the way in which the developer team prioritize requirements from different user groups. Second, the issue voiced by the national and regional HISP-groups that the developers at HISP UiO fails to deliver on ‘promised’ functionality persisted. RCAT made this accountability challenge visible as requirements remained unresolved.

The RCAT process was in itself challenging. Some of the member groups remained passive, a few contributed with their priorities and one were extremely active. This imbalance in participation could in part be a result of the different experiences and competencies of the individual representatives. The lack of overlap between the priorities from the different groups also made it easy for HISP UiO developers to downplay their overall importance. The lack of overlap can both be due to a lack of willingness to make compromises or simply that the needs of the countries were too divergent. The RCAT initiative continues, but the establishment of several DHIS2 product manager roles has recently been established to supplement it. The product managers are responsible for engaging with different DHIS2 thematically oriented communities – to reflect the diversity of interests and needs – and to help translate specific community needs into software requirements.

6. Discussion

DHIS2 brings together different groups of stakeholders, such as Ministries of Health, local NGOs, international donors, consultants and software developers. Stakeholders interact with the platform differently and have different needs and motives, albeit with the majority of interests geared towards public health administration. Innovations extending the software platform are often inclusive in outcome, as they facilitate data driven decision making towards equitable health service provisioning. For instance, with access to timely and accurate statistics on the utilization of essential health services, such as child immunization and antenatal care, service providers can design interventions to reach the underserved. The DHIS2 software development has a long history of being inclusive also in terms of process through user involvement in solution design. With the rapid growth of the venture, HISP and the DHIS2 software platform serves as a fruitful arena to problematize large scale inclusive innovation [14, 15].

In principle, the logistical challenges to large scale inclusive innovation [7, 9] may be mitigated by a software platform architecture as it allows idiosyncratic solutions to emerge at the platform fringes through app development [26]. Inclusive innovation can be more open ended at the platform fringes, where standardisation and generic features are less of a concern and both technical and organizational dependencies are fewer. Local needs associated with a growing number of heterogeneous users and uses can be met at different levels of the platform architecture through either custom apps (specific), bundled apps (configurable) or core software development (generic).

In practice, innovation in the DHIS2 ecosystem often constitute idiosyncratic local customization, rather than development of generic core functionality or configurable apps that can be bundled with the software releases and adapted across settings and

programs. While actors at the fringes of the DHIS2 platform ecosystem have access to the increasingly ubiquitous Internet, digital devices, services, and content, they remain somewhat disconnected from the core of the HISP network where innovative ideas and features are likely to attract the resources and capacities required to scale innovations across settings [17]. For instance, the yearly DHIS2 Expert Academy in Oslo is an important venue for sharing innovations with the wider community and influencing the overall direction of the venture. Yet, with increasing scale, inclusion through representation in the development of the core software platform features becomes further and further removed from specific local needs and requirements.

In contrast, the level of inclusion in innovation processes are profound at the early stages of conceptualization of new features in new domains such as IDSR described in the first vignette. Depending on their perceived importance by stakeholders close to the core of the HISP network, such efforts can involve articulation of the software requirements by experienced DHIS2 core software developers. In other cases, a local software implementer or customizer, associated with one of the HISP groups, can spearhead an initiative to accommodate particular local needs. In either case, boundary resources such as APIs, SDKs, issue tracking software and requirement synthetization efforts such as RCAT both enable and constraints the work of innovation intermediaries.

Innovation intermediaries, such as the HISP UiO software developers who travelled extensively to engage in workshops and conduct in-country DHIS2 Academies in the early days of DHIS2, play a crucial role in maintaining a balance between processes of top down standardization (i.e., software appropriation and customisation) and the diffusion and scaling of bottom up innovations. With the increasing scale and scope of the venture, it is no longer feasible for the core software developers primarily situated in Oslo to participate in the majority of DHIS2 Academies and engage in discussions about the details of particular local needs. Consequently, the role of mediating requirements into the software development roadmap and the organizing of DHIS2 Academies has transferred to the national and regional HISP groups. Professionals associated with the HISP groups can either engage in software customization and app development directly or mediate requirements towards the centre of the network when technical flexibility or their capacity to approach the problem through local innovation is exhausted.

In relation to development of the generic core and bundled apps, mobilization of community representatives in the platform ecosystem is important to ensure representation of different user groups and organisations and to facilitate transparent negotiations of requirements. As our case study illustrates, with the sheer size and diversity of the HISP network, this can be challenging to facilitate in a democratic way even when the venture has a long tradition for participation in solution design. A global public good software platform for inclusive innovation, as we see it, depends on a modular software platform architecture, stable and well-defined boundary resources in addition to meticulous capacity building efforts, including training on software customization and app development, to empower innovation intermediaries in the platform ecosystem. The DHIS2 platform, the DHIS2 app store, the DHIS2 Academies, RCAT and other venues for innovation intermediation have evolved in tandem and helps balance top down standardization with bottom up innovation.

7. Conclusion

Making software platform innovations generic and available across contexts (i.e. scale-up) is often not perceived as a winning strategy within the confines of the current donor-driven and project-oriented funding structures that dominate global health. The strong compartmentalization of funding and activities into vertical health programs and silo structures (e.g., HVI/AIDS, TB, malaria) further solidify these dynamics. In short, the structures and time frames that encapsulate information system projects in global health often constrain local project managers and project participants' flexibility to innovate (see e.g.[27, 28]). Interestingly, our experiences from global health echoes, at least partly, the experiences around the web-based micro loan service Kiva, where local partner institutions faced challenges in terms of obtaining adequate local capacities and knowledge to comply with technological standardization initiatives aimed at efficiency and economies of scale in the multinational network [16].

International NGOs and donors provide short bursts of technical assistance or hire consultants to solve particular problems based on programmatic solutions. Such processes are often inclusive in outcome, but not in process. Furthermore, initiatives often fail to facilitate sharing and scale-up of innovations in the broader network of stakeholders and across settings with variable resources and capacities. This hampers the cultivation of a sufficiently strong cadre of innovation intermediaries that can resolve tensions between generic features and local issues and requirements. There is a need to identify ways to cultivate networks of innovation intermediaries around global public good software platforms, such as DHIS2 to facilitate inclusive innovation at a large scale within the confines of extant development industry value creation mechanisms.

References

1. Yoo, Y., Boland Jr, R.J., Lyytinen, K., Majchrzak, A.: Organizing for innovation in the digitized world. *Organization science*. 23, 1398–1408 (2012)
2. Nambisan, S., Lyytinen, K., Majchrzak, A., Song, M.: Digital Innovation Management: Reinventing innovation management research in a digital world. *Mis Quarterly*. 41, (2017)
3. Tiwana, A.: Platform ecosystems: aligning architecture, governance, and strategy. Newnes (2013)
4. Van Schewick, B.: Internet architecture and innovation. MIT Press (2012)
5. Autio, E., Thomas, L.: Innovation ecosystems. *The Oxford handbook of innovation management*. 204–288 (2014)
6. Chataway, J., Hanlin, R., Kaplinsky, R.: Inclusive innovation: an architecture for policy development. *Innovation and Development*. 4, 33–54 (2014)
7. Foster, C., Heeks, R.: Nurturing user–producer interaction: inclusive innovation flows in a low-income mobile phone market. *Innovation and Development*. 4, 221–237 (2014)
8. Foster, C., Heeks, R.: Conceptualising inclusive innovation: Modifying systems of innovation frameworks to understand diffusion of new technology to low-income consumers. *The European Journal of Development Research*. 25, 333–355 (2013)
9. Heeks, R., Foster, C., Nugroho, Y.: New models of inclusive innovation for development. *Innovation and Development*. 4, 175–185 (2014)

10. Sanner, T.A., Manda, T.D., Nielsen, P.: Grafting: Balancing Control and Cultivation in Information Infrastructure Innovation. *Journal of the Association for Information Systems*. 15, 220–243 (2014)
11. Mudaly, T., Moodley, D., Pillay, A., Seebregts, C.J.: Architectural frameworks for developing national health information systems in low and middle income countries. In: *Enterprise Systems Conference (ES)*, 2013. pp. 1–9. IEEE (2013)
12. Braa, J., Monteiro, E., Sahay, S.: Networks of action: Sustainable health information systems across developing countries. *MIS Quarterly*. 28, 337–362 (2004)
13. Dutz, M.: *Unleashing India's innovation: toward sustainable and inclusive growth*. The world bank (2007)
14. Papaioannou, T.: How inclusive can innovation and development be in the twenty-first century? *Innovation and Development*. 4, 187–202 (2014)
15. Heeks, R., Amalia, M., Kintu, R., Shah, N.: Inclusive innovation: definition, conceptualisation and future research priorities. *IDPM Development Informatics Working Papers*. (2013)
16. Braund, P., Schwittay, A.: Scaling inclusive digital innovation successfully: the case of crowdfunding social enterprises. *Innovation and Development*. 6, 15–29 (2016)
17. Roland, L.K., Sanner, T., Sæbø, J.I., Monteiro, E.: P for Platform. *Architectures of large-scale participatory design*. *Scandinavian Journal of Information Systems*. 29, (2017)
18. Baldwin, C.Y., Woodard, C.J.: *The architecture of platforms: a unified view*. Harvard Business School Finance Working Paper. (2008)
19. Ghazawneh, A., Henfridsson, O.: Balancing platform control and external contribution in third-party development: the boundary resources model. *Information Systems Journal*. 23, 173–192 (2013)
20. Gawer, A.: Platform dynamics and strategies: from products to services. *Platforms, markets and innovation*. 45, 57 (2009)
21. Walsham, G.: The emergence of interpretivism in IS research. *Information systems research*. 6, 376–394 (1995)
22. Walsham, G.: Doing interpretive research. *European Journal of Information Systems*. 15, 320–330 (2006)
23. Sahay, S., Nielsen, P., Latifov, M.: Grand challenges of public health: How can health information systems support facing them? *Health policy and technology*. 7, 81–87 (2018)
24. Braa, J., Hedberg, C.: The struggle for district-based health information systems in South Africa. *The information society*. 18, 113–127 (2002)
25. Nguyen, T.N., Nielsen, P.: From Routine to Revolt: Improving Routine Health Data Quality and Relevance by Making Them Public. In: *International Conference on Social Implications of Computers in Developing Countries*. pp. 548–558. Springer (2017)
26. Olleross, X.: The lean core in digital platforms. *Technovation*. 28, 266–276 (2008)
27. Winschiers-Theophilus, H., Zaman, T., Yeo, A.: Reducing white elephant ICT4D projects: a community-researcher engagement. In: *Proceedings of the 7th International Conference on Communities and Technologies*. pp. 99–107. ACM (2015)
28. Sanner, T.A., Sæbø, J.I.: Paying Per Diems for ICT4D Project Participation: A Sustainability Challenge. *Information Technologies & International Development*. 10, pp–33 (2014)