The Battle Ground Of Information System Development in Developing Countries: From Mutual Exclusion To Hybrid Vigor

Abstract:

The integration of software components into large-scale and complex information systems is a topic attracting interests from many information systems practitioners and researchers. Less attention is given to the intricate processes in which these infrastructures are built based on multiple independent software components. Introduced by different actors, these components are pushed onto a battleground of functional roles where components are replacing and blocking each other. Based on a case study of the development of an information infrastructure for communicable diseases detection, prevention, and control in Vietnam, we discuss ICT4D as one such battleground. This paper contributes by unpacking this case and conceptualizing its success as an example of the outcome of a strategy of hybrid vigor.

Keywords: information systems, information infrastructure, infectious diseases information systems

# Introduction

Information Systems are no longer designed and implemented in isolation, but are integrated with and extending already existing large-scale and complex information systems. How this situation influences the process of building successful information systems as parts of a larger whole has been a topic attracting interests from many information systems practitioners and researchers. Key aspects that make this context and the nature of the technology different from stand-alone information systems has been theorized as Information Infrastructure and their dynamics described by concepts of openness, shared, evolving, standardized, heterogeneous, and building on an installed base (Hanseth and Monteiro 1998, Hanseth 2010). The term “installed base” is coined by Hanseth (2010) as the existing “set of ICT capabilities and their users, operations and design communities”, and it also encompasses existing institutional and organizational components (Lanzara 2014).

Because this context is substantially different, it also requires novel approaches to design and implementation (Henningsson and Hanseth 2011). A key challenge working with a component of a larger whole is that control is distributed among different actors. Cultivation by using different means to influence other actors and changes in small steps towards a certain goal is an important technique espoused by both industry and academia (Hanseth 2010). The tenet of this approach advocates designing and building a system from what already exists rather than from scratch. In other cases where there is no (or little) installed base to utilize, one can incrementally build systems through the strategy of bootstrapping, referring to a process of producing and providing incentives to mobilize the first users of a technology and using this set of users as an installed for further development (Hanseth and Aanestad 2003).

Recently, approaches to the II design have been augmented by several researchers in a variety of ways. Sanner et al. (2014) propose grafting as a strategy to embed a component in the network of components by making sure this component can adapt well to the existing installed base through various social-technical adjustments. To that end, the choices of the right place and the right time are crucial for the success of the grafting. Other researchers have recognized the role of architecture in shaping II development trajectories, arguing that a resilient and open architecture is critical for II successful evolution (Grisot, Hanseth et al. 2014). Others have called for partial or complete devolution of control over IIs to maximize the innovation and creativity to enable success and sustainability of IIs (Ciborra 2000, Hanseth and Braa 2000, West and Dedrick 2000, Hanseth, Ciborra et al. 2001, Nielsen and Aanestad 2006, Tilson, Sørensen et al. 2012, Gregory, Beck et al. 2013, Sanner, Manda et al. 2014) .

Until now, most research on II focuses on finding innovative approaches that help cultivate IIs through various strategies such as bootstrapping and recently grafting. In such studies, constitutive elements or components are introduced to either extend, complement or be integrated with the existing network. Within the scope of this research, we employ the following definition by Szyperski et al. (2002) on software components:

“A software component is a unit of composition with contractually specified interfaces and explicit context dependencies only. A software component can be deployed independently and is subject to composition by third parties.”.

We believe this definition is generic and flexible enough to use in a variety of contexts. As the unit of evaluation of our research is IIs which are usually a complex information systems, a software component could be understood as a sub-system or module that have their own stakeholders (owner, development team, sponsor, advocator) and development agenda.

Till date, little research has been done on contexts where multiple components (software systems or modules) are deliberately brought in to replace existing ones and potentially block other alternatives and future attempts of replacements. One exception here is the work by Nielsen and Sæbø (2016), conceptualizing the interplay between and the strategy behind different and potentially competing software components as functional architecting. They distinguish three different strategies used by proponents of different software components and discuss them as charting, encroaching, and connecting. Extending the work of Nielsen and Sæbø, the focus in this paper is on situations where different software components meet and the established functional architecture is challenged and changed. What we will illustrate is that the consequence of these strategies can be severe clashes of old and new components. Conflicting strategies may lead to the replacement of existing components and potentially the whole network by a single new component.

We have a particular focus on the situation when multiple actors directly compete to serve the same functional needs and thus for their own existence. Thus, the aim of this research is about understand how components especially those are open-source based and relatively weak in terms of power can fight for their survival. We frame the goal of our research in the following research question: “What are strategies to avoid mutual exclusion and thus enhance the totality, openness, diversification, evolution, and innovation of information infrastructures when there are emerging new directly competing software components that offer overlapping functionalities and could substitute each other?”

To answer this research question, we undertook a multiple-case study consisting of 4 individual cases, all of which involved negotiation, development, and implementation of information systems for communicable disease monitoring in Vietnam between 2010 and 2015. We were directly involved in one of these initiatives and participated in general discussions and collaboration with stakeholders in the other three. Through the analysis of these cases, we discovered hybrid vigor as a different strategy for functional architecting, apart from the three strategies described by (Nielsen and Sæbø 2015) which are charting, encroaching, and connecting. We define hybrid vigor as a strategy that aims towards improving the totality of the functional attributes of an infrastructure. Our definition also includes three dimensions of hybrid vigor which we identify as follows: the politics of functional negotiation, the power of governmental agencies, and reconfiguration of the resulting infrastructure.

The remainder of the paper is organized as follows. The review of related research is provided in section 2 followed by the presentation of methods and approaches we used in this study. In section 4, we introduce the case of the epidemic reporting systems in Vietnam. Analysis and discussion are provided in section 5. We conclude the paper in section 6.

# Related research

Conceptualizations of large and complex information systems focus on complex networks or networks of systems that is comprised of heterogeneous components. Hanseth et al. characterizes them as a "shared, open, evolving, heterogeneous and evolving socio-technical systems” (Hanseth and Monteiro 1998). Appearing as a different genre of information systems, Information Infrastructures (II) cannot be built or constructed by conventional software engineering methods, methods developed for standalone and homogeneous systems (Somerville et al 2013).

A common argument in the II literature is that the complexity, uncontrollability, and the unintended consequences associated with IIs are the root causes for the failure of many large IT projects (Aanestad and Jensen 2011). Development and implementation of large scale and complex information systems demand new approaches which requires the perception of technology as more than individual tools (Tilson, Lyytinen et al. 2010). There have been many efforts to respond to this demand. One of them is the design theory developed by Hanseth and Lyytinen (2010), tackling dynamic complexity in the design for IIs by addressing two key challenges: bootstrap and adaptability. The theory delineates a list of design principles which guide designers on how to “generate attractors to propel users to adopt the IT capability so that its growth will reach a momentum” (ibid, p.8) and “guarantee that the II will grow adaptively and re-organize constantly with new connections between II components” (ibid, p.13).

Another important aspect of large scale and complex information systems is the lack of centralized control and as a consequence the nature of how they change. Cultivation is used to describe the process of actualization of an II in a piecemeal and incremental manner and based on what already exists, the installed base. II slowly evolves, emphasizing that due to its complexities, an II is hardly built or designed quickly, as it is not practically possible to ignore the legacy of: “historical accumulation of socio-technical arrangements around it” (Sanner, Manda et al. 2014, p.221).

Accepting the complex nature of IIs has led researchers to study the different strategies pursued by different actors within these large scale and complex networks. For example, Sanner, Manda et al. (2014) recently introduced the concept of grafting to describe a strategy where “local organizational goal-oriented information system innovations become viable extensions of shared and evolving information infrastructure” (ibid, p.221). Identifying the right moment and position to graft the scion which, in this case, is a novel software component to the rootstock (the installed base of software systems) is critical for success. The grafting strategy also entails effectively managing the relationships with stakeholders who are in control of parts of the existing installed base and the deliberate choice of rootstock to ensure congeniality. Another example is the concept of co-evolution, bringing the focus to how different technologies developed by different actors may intersect from time to time. Jansen and Nielsen (2005) introduced and applied the concept of co-evolution to analyze the parallel evolution of IIs using the case of two wireless communication platforms Universal Mobile Telephone System (UMTS) and Wireless Local Area Network (WLAN) in Norway. They concluded that the intra- and interlinkages of technologies, politics, interests, and user preferences within each II strongly influence the trajectory of the co-evolution. In a similar vein, based on the work of Benbya and McKelvey (2006), Shaw (2009) develops the co-evolutionary framework for managing the complexity of hospital management information systems development in resource-constrained settings. Yet another example is the literature on platforms, discussing the distribution and redistribution of control and innovative capacity between different actors and the technologies they control. For example, Tiwana, Konsynski et al. (2010) argue that the evolutionary dynamics of IIs is significantly influenced by the “coevolution of endogenous choices by platform owners and the dynamics of an ecosystem’s exogenous environment” (ibid, p.687). Further, several scholars including Nielsen and Aanestad (2006) and Hanseth, Ciborra et al. (2001) cleverly use the concept of devolution to describe a situation in which IIs might better evolve if centralized control is balanced with autonomy. The reduction of strict control on the II growth would permit “the distribution of resources, risks and abilities and willingness to innovate” (Nielsen and Aanestad 2006, p.185).

The extant II literature mainly discusses the introduction and the (re-)combination of innovative systems, subsystem, and components into existing social-technical arrangements and how these constituents shape and being shaped by the social, technical, and institutional contexts. There is a dearth of research that clearly discusses the overlap and duplication of the new and the incumbent software components, their competition and potential substitution. An important exception to this is the work of Nielsen and Sæbø conceptualizing the interplay between and the strategy behind different and potentially competing software components as functional architecting. They distinguish three different strategies when software components made for one domain is moving into another, such as for example when a corporate accounting system is also offering functionality for human resources management. Charting is one strategy where a software component is moving into a new domain by meeting an unmet functional need. Connecting is another strategy where software components from different domains are connected to leverage on the benefits of complementary functionality. Finally, encroaching is described as a strategy where a software component is moving into a new domain by offering functionality in direct competition with functionality already provided by other components.

Health information systems in developing countries, as in our case study, are notoriously incomplete, unreliable, obsolete and of poor quality (Heeks, Mundy et al. 1999, Braa and Hedberg 2002, Braa, Monteiro et al. 2004, Haux 2006, Heeks 2006, Braa, Hanseth et al. 2007, Garde, Hullin et al. 2007). Although there has been substantial investment from local governments and international donors, the situation does not seem to be improved but sometime get worse due to the lack of effective coordination and technical competency as well as poor and perhaps corrupt governance. In many cases, poor coordination results in wasteful overlap and duplication of investment in software systems because donors and government agencies support the development of disparate systems that provide exactly the same functionalities for the same setting. In practice, they make these systems as direct competitors.

Braa, Monteiro et al. (2004) raise another problem in public health systems which they call the “all or nothing dilemma”. There is a common need to scale an ICT innovation to full coverage, i.e. all districts in a province or all provinces in a country, to make it useful to health managers. But typically, different systems are implemented in an uncoordinated and fragmented fashion and there is no single system that meets the full coverage criterion. These systems are at risk of being substituted by new systems that will cover a larger area. At the same time, developing countries are commonly too poor to afford the substitution of all directly competing systems.

While the II literature pictures the development and evolution of infrastructure as emergent and out of limited central control, there is still room for intervention. Based on his work on the history of the electrical system in the US, conceptualised as a large scale and socio-technical system, Hughes coined the term “system builder” (Hughes 1979, Hughes 1987) . Hughes showed how this electrical system was not only technical, but also was related to and based on various institutions, manufacturers and investors. Such systems cannot satisfactorily be treated in isolation from organisational, political and economic matters, but must be integrated with their context to work and to grow. To facilitate this is the key role of the “system builders” – the creators of large-scale and socio-technical systems. “System builders” preside over technological projects from the concept and preliminary design stages through research, development and deployment. In order to do so, they need to cross disciplinary and functional boundaries and become involved in funding and political stage-setting. The core competency of the “system builders” thus lies in their ability to integrate heterogeneous physical, human and organisational components into a working and goal-oriented system: “… to force unity from diversity, centralization in the face of pluralism, and coherence from chaos.” (Hughes 1987). According to Hughes, “system builders” should have a holistic focus and see the entire system, rather than only its components. Through control and management, and with attention to the interconnection between the system’s different components, “system builders” believe that the system will not evolve and grow without someone viewing it as a coherent system. Inspired by the concept of “system builders”, we conceptualize the hybrid vigor strategy to capture and understand situations in which new systems are introduced and live side-by-side by the existing and what is needed to make them do so.

As we have discussed earlier, clearly there is a gap in II literature which deals with directly compete software components. So far, II researchers have rarely provided discourses on the issues of co-existence of direct compete software components and solutions to address that challenge. One exception includes the case of asymmetric integration reported by Sahay, Monteiro et al. (2009). In their study, Sahay, Monteiro et al. (2009) criticize that integration is not merely technical issues but involves political and institutional interests of different stakeholders. Even though the integration of different software components could contributes to the evolution of IIs, it is hard for more powerful actors to easily accept new components. That raises the issue of asymmetric integration, i.e. unbalanced power distribution. The solution is the smaller actor must configure their systems to match with the more dominant ones. However, Sahay, Monteiro et al. (2009) do not highlight the issue of the direct competing software components which offer overlapping functionalities. In such cases, a system can be easily substituted by other systems backed by more powerful stakeholders such as governmental agencies, international donors etc.

We borrow the term hybrid vigor from biology where it is used to describe the improved or increased function of any biological quality in a hybrid offspring (Shull 1948). Hybrid vigor occurs through cross-breeding of plants or animals from the same species (although in different breeds) or from the same genus.

We decided to use a biology term (hybrid vigor) in the context of II development for several reasons. First, the development of II has many characteristics similar to the growth and evolution of biology ecosystems (Manikas and Hansen 2013). The II development is incremental as it happens in baby steps, complex as it involves many components and depends on internal and external factors, evolutionary as it is gradually upgraded in terms of functionalities and complexities (Henningsson and Hanseth 2011) . Second, II researchers have been extensively using terms borrowed from biology to describe the development of IIs. For example, Hanseth and Monteiro (1998) uses the term evolution, i.e. evolution of IIs, to define the development of IIs as a natural and biological process in which the II development is similar to the growth of biological entities. They also compare this process with potatoes’ growth. Hanseth (2010) adopts the term cultivation which is an agricultural process in which seeds or trees are planted to propose a strategy of building IIs. Sanner, Manda et al. (2014) uses the term grafting, a biological technique that helps transplant parts of one tree to another to refer to the process of adoption, transfer, and implementation of technology invented in one place to the others, usually from North to South.

All in all, there is some limitations on the use of these biological concepts given the dynamic nature of II development. The concept of evolution is relatively generic and descriptive, i.e. it is mainly about how the II grows but not about how to design II to enable its healthy growth. The concept of cultivation has been criticized for its lack of precision in capturing specific goal-oriented organizational interventions (Sanner, Manda et al. 2014). Therefore, we still need additional concepts in other to unpack the complexities of II development. As such, we propose the use of hybrid vigor to further understand the II development process better. Hybrid vigor is a strategy used under circumstances in which direct competition between software components exists, and where certain actors enable the combination of components that offer almost identical functionalities. Hybrid vigor is the strategy of actors understanding the potential of and pursuing the opportunities which the cross-breed of rival software components can yield the evolutionary information infrastructure.

In situations of direct competition, existing components and actors are in risk of being sidelined and replaced. In some instances, there will be only one “winning” component, which is best adapted to the environment, while all other fade away. The hybrid vigor strategy is based on the assumption that such a situation will be suboptimal. In particular, IIs will evolve more dynamically and support innovation and flexibility to a much larger extent if competing components can find a way to co-exist and collaborate instead of mutually excluding each other. In such a case, the strengths and weaknesses of each component will be combined and complemented which in turn foster the flexibility and the potential for expansion. In regard to functional architecting, the three strategies developed by Nielsen and Sæbø (2015) do not sufficiently capture the dynamics in which key system builders, both governmental and non-governmental agencies, regulations and politics are shaping this context. Our contribution is the conceptualization of hybrid vigor as the fourth strategy to deal with functional architecting in such contexts. Based on our case study of the building of several communicable diseases reporting system in Vietnam, we further unpack the hybrid vigor strategy by identifying and articulating its various dimensions.

# Research Method and approach

This research is based on multiple-case studies (Stake 2013). First, we followed the design and deployment of the Vietnamese epidemic notification system (ENS) by the Administration of Medical Services (VAMS) in response to a fatal measles outbreak in Vietnam at early 2014. It killed hundreds of children. Two of the authors of this paper were active participants in this effort on both managerial and technical levels. Second, we examined three other initiatives pursued in parallel with VAMS by three other actors and with more or less the same goal: to support data collection, reporting, and monitoring of communicable diseases in Vietnam. These three cases include the electronic communicable disease system (eCDS) backed by the General Department of Preventive Medicine (GDPM), the new version of eCDS developed by a state-owned telecom company (VCom), and the communicable disease dashboard system supported by an international NGOs (referred to as iNGOs). Apart from separately studying these four cases, we have also focused on their intersections and the collaboration between the different actors coordinated by the iNGOs in leveraging opportunities that a joint effort could generate.

Collecting data of the first case was done in parallel with the process of development and implementation of ENS in which the two first authors directly participated. The other three cases were conducted between late 2015 and early 2016initated by the two first authors being invited to take part in the joint effort led by iNGOs in integrating the existing communicable diseases reporting systems. In this particular case, our research can be seen as an action research with a mix between software prototyping and problem solving for organizational change (Davison, Martinsons et al. 2021). We also take into considerations the recommendations on maintaining ethics when practicing action research (Davison, Martinsons et al. 2022).

Using the case study approach, case selection is significant (Merriam 1998). While partially being opportunistic and based on our access to the field, our selection of cases to include in this paper was primarily based on our aim of understanding the battleground unfolding related to disease surveillance systems in Vietnam. The strengths of these cases combined is partly access and partly richness. On the one hand, our long-term involvement and access in the health care sector in Vietnam in general and the domain of communicable disease monitoring in particular, was crucial for a rich and longitudinal study which is useful to observe the evolutional trajectory of IIs. Second, extreme case selection method which “selects a case because of its extreme value on the independent (X) or dependent (Y) variable of interest” (Seawright and Gerring 2008, p.301) was also the case, supporting our aim of theorizing hybrid vigor: “extremes or ideal types typically define theoretical concepts” (Henfridsson and Bygstad 2013, p.914). We do believe that cases which simultaneously involved four separate attempts to build systems with similar functionality are rare.

In gathering data for the study, we followed Benbasat, Goldstein et al. (1987) who emphasize the need for using multiple sources of data to increase the reliability of the research. In addition, careful planning before collection and meticulous record keeping during and after collection were actively applied to utilize time spent on sites and avoid loss of precious data. Our methods of data collection included participative observation, interviews, focus groups, and archival records. First, participative observation was applied when we were engaged in collaborative efforts coordinated by the iNGOs. Our exposure to events such as meetings gave us chances of "absorbing and noting details, actions, or subtleties of the field environment” (Benbasat, Goldstein et al. 1987, p.374). In total, we attended four different meetings organized by the iNGOs out of which one lasted for a whole day. In these meetings, demonstrations of different systems were done, giving us insights into the architectural and functional design of them. This was critical for us to understand the issue of direct functional. Also, the participative observation method was used extensively in the case in which the first and second authors of this paper were involved to develop and implement. This involvement lent us a great chance to clearly observe the political and technical negotiating processes. Being involved too closely with the case we study can cause threat of biases which we are fully aware and deliberately try to avoid. We adopted data triangulation techniques and purposely sought neutral and alternative explanations for the same phenomena. The third author had a particular role in asking the critical questions needed.

Second, interviews and focus groups were used for data collection. Interviews were conducted to get broader information relating to the systems that we were not directly involved and to enrich and discuss the data that we had collected from other sources. This also helped ruling out contradictions and inconsistencies. Formally, we conducted eight interviews. Each interview lasted about half an hour to an hour. The first group of informants included the staffs in charge of operating the eCDS at General Department of Preventive Medicine (GDPM) (4 interviews). The second group consisted of managers and developers working for VCom in the project that entailed the redevelopment of the eCDS system being used at GDPM (4 interviews). All the interviews were recorded and transcribed for analysis purpose. Apart from that, numerous informal interviews were conducted by authors interacting with iNGOs, GDPM, and VCom. These informal interviews provided details that complemented the data we gathered from other sources. To collect collective views and experiences and beliefs of the participants from GDPM, iNGOs, and VAMS, we also used focus groups discussions during the four meetings organized by iNGOs. The purpose of each meeting, happened between October 2015 and January 2016, was varied depending on time and participants. However, it pivoted on making sure that all stakeholders understand the status quo of each system and creating opportunities for interest expression and plan discussion.

Third, the archival records method was also used to collect data that provided general information about legal issues around communicable diseases administration in Vietnam. Email exchanges, project reports, and proposals were other types of archival records used. The list of key archival records is described in table 1.

|  |  |  |
| --- | --- | --- |
| # | Names of documents | Authors, issuers |
| 1 | Circular 48/2010/TT-BYT, guiding communicable disease reporting processes and procedures. | MoH |
| 2 | Circular 54/2015/TT-BYT, modifying Circular 48/2010/TT-BYT | MoH |
| 3 | The implementation plan of the electronic Communicable Diseases Systems in 63 provinces. | GDPM |
| 4 | Official dispatch 2615/BYT-KCB dated May 13th, 2014, requesting daily reporting of measles on the online system. | MoH |
|  | The ADB Report on the Preventive Health System Support Project http://www.adb.org/projects/3so4348-013/main#tabs-0-1 | ADB |
| 5 | Presentations at the meetings | iNGOs, GDPM, VAMS |
| 6 | Emails exchanges (~ 200 threads) | Authors and stakeholders |

The data collection process highly influenced by our prior hypotheses, assumptions, and knowledge about the research topic was conducted in an iterative manner. That means that data analysis was performed simultaneously with the data collection process. In general, our coding procedures started with predefined codes and was gradually augmented by codes emerged from the collected data. Through a number of reflections and rereading the materials, larger themes were identified, which was an important step in finding answers to the research questions. Initially, we traced the competition amongst software systems and early conceptualized hybrid vigor as a strategy that helps reconcile and neutralize conflicts between systems’ proponents. Examining the strategy we gradually discovered multiple layers of its dynamics and delineate several aspects to which we referred as dimensions of hybrid vigor.

As part of the analysis process, the cases description is presented in the following session.

# Four Initiatives Addressing the communicable diseases reporting

In this section, we present the four initiatives and systems implemented to address the need of data collection, reporting, and monitoring of communicable diseases in Vietnam. The cases are described in a chronological order.

# The Electronic Communicable Disease Reporting System (eCDS)

The preventive medicine system of Vietnam can be described as "a passive system" which was mainly responsible for data collection and reporting rather than epidemic control and prevention. A dearth of skilled staff and adequate infrastructure and the lack of standards and protocols in epidemic reporting and monitoring were identified to be the root causes for such weaknesses. In 2008, the Vietnam Nation Assembly enacted the Law on Communicable Diseases Prevention (03/2007/QH12). This was followed by the Circular 48/2010/TT-BYT (shortly called Circular 48) issued by MoH on December 31st 2010. Both of the documents provided a legal frame to legitimize the reporting of communicable diseases, making it a routinized and compulsory protocol.

In 2012, the Asian Development Bank (ADB) and the World Health Organization (WHO) sponsored the Vietnamese General Department of Preventive Medicine (GDPM) to build an information system to support data collection, reporting, and monitoring of communicable diseases through two projects: the Preventive Health System Support Project (PHSS) and the Vietnam Avian and Human Influenza Control and Preparedness Project (VAHIP). The information system was based on a web-based software, developed on Microsoft .NET Web Form platform, and used Microsoft SQL Server as the database management system. A local company was awarded the contract to build the software. After some months, the software was completed and ready for a pilot. Seven provinces representative for the different regions of the country were selected as pilot sites. The goals of the pilot were to evaluate the software and identify necessary infrastructural conditions required for successful national implementation. The implementation team was comprised of staff from GDPM, PHSS, VAHIP, WHO, as well as regional epidemic control institutes. Training sessions on using the software were organized for doctors and health workers who were in charge of preventive medicine at provincial and district levels. Continuous support to users was given by GDPM and the implementation team. The pilot phase ended with positive outcomes and most of the goals were achieved. All districts in the pilot provinces had used the system for reporting the communicable disease data on weekly and monthly basic. The software provided most of the required functionalities needed for communicable disease reporting as mandated by Circular 48. Users from various levels (ward clinics, district health center, provincial preventive medicine centers, regional epidemic control institutes, and GDPM) could access the same system for data entry, validation, processing and analysis. This design facilitated the collaboration across levels and ensured interoperability of data coming from various sources.

The success of the pilot further attracted the interest of donors in scaling up the implementation across the country. ADB continued their support by sponsoring the implementation of the system (training, support, evaluation) in 45 provinces in total. WHO provided support for servers, hosting, and other hardware services. They also sponsored the implementation, monitoring and evaluation of the system in 3 provinces. The nationwide scaling up process also received support from Centers for Disease Control and Prevention (CDC, the United States) to implement the system in seven provinces. GDPM covered the implementation in the last eight provinces of Vietnam by their budget.

As of June 2014, all provinces had implemented the system at district and provincial levels. Out of them, 11 provinces had the system down to the ward level. For provinces that did not implement the system at ward level, district PMCs had to enter data from paper-based reports sent by ward clinics. This nationwide implementation thus had the potential to provide full coverage of timely and sufficient data about communicable diseases on the country level. However, this was not the case. The appendices of Circular 48 regulated reporting templates that health facilities (HF) and preventive medicine centers (PMC) must use to report communicable diseases data. These templates could be divided into three categories: aggregate data, case-based data, and outbreak. Routinely, data were reported in weekly, monthly, and yearly basis. For fatal and severe communicable diseases classified in Group A by Circular 48, reports must be sent immediately for timely intervention. Apart from upward reporting flow i.e. district health centers report to provincial health centers and peer reporting flow i.e. district hospital report to district health center, there was a feedback flow in which upper-level PMCs send lists of confirmed cases to lower-level PMCs for verification and intervention purposes. However, due to the dearth of adequate skills and equipment in PMCs at both district and provincial levels, most of the communicable diseases confirmed cases were diagnosed and discovered by hospitals only. This design was the Achilles heel of the preventive medicine system in Vietnam, making the collected data relentlessly insufficient, inaccurate, and delayed. It means that PMCs are in need of infectious disease data for their intervention however they could not actively and directly collect such data. Instead, they have to rely on hospitals. Delay in reporting data from hospitals is such a serious issue that PMCs often have to send their staff to some big hospitals and stay permanently just for data collection purpose. As a result, data from other hospitals were sometimes missing or lately reported.

# The Epidemic Notification System (ENS)

During new Lunar New Year of 2014, the accidents and injuries reporting system of the Vietnam Administration of Medical Services (VAMS), an electronic reporting system tracking daily admission and emergency of all hospitals in the country, noted several cases of measles from a mountainous province in Northern Vietnam. The length and joy of the festival caused everybody overlook the significance of the harbinger of a severe epidemic. Two months later, the epidemic became very serious with hundreds of deaths and thousands of infection cases. However, public and media were not aware of the severeness of the situation. The director of a Hanoi-based pediatrics hospital had to invite journalists to visit his hospital to directly observe the seriousness of the epidemic, hoping that media could help to change the public attitudes toward the disease. Witnessing many fatalities of measles in child patients, one pediatrics doctor posted an emotional status on Facebook expressing his helplessness on fighting against the disease. Coincidently, this post was seen by his friend who was the Deputy Prime Minister in charge of the health care sector. The Deputy Prime Minister decided to visit the hospital in person to get a closer look at the situation. The increasing number of measles patients admitted to hospitals made overcrowding a serious issue. Two or three patients had to share a sickbed. The workspace of staff was also allocated for inpatients. Respirators were mobilized from other wards in the same hospital for measles treatment. After the visit of the Deputy Prime Minister, the media were full of alarming reports about the measles epidemic, causing panic across the country.

A key challenge in this situation was that no agencies or authorities could provide the exact number of fatalities, infected and admitted cases. The official source of MoH reported only 25 deaths while data from a single hospital indicated nearly 3 times this number. It became obvious that Vietnam did not have appropriate and effective information systems to support the monitoring of communicable diseases. At that time, the eCDS backed by GDPM was still in its pilot phase. By its original design, eCDS was primarily designed to provide detailed data required for controlling a measles outbreak. For example, it supported the collection of the total number of infection cases and fatalities but did not support the tracking of data such as the number of inpatients with serious complications. This number, however, was critical for efficient allocation of scarce resources such as highly skilled doctors and respirators.

In an effort to mitigate the harshness of the situation, an official at VAMS discussed with the first author about the feasibility of reusing the annual accidents and injuries reporting system (AIRS) to build a daily reporting system to monitor the measles epidemic that was ravaging across the country. Prior to the implementation of AIRS, two of the authors had supported MoH in a number of systems such as health professionals licensing system, hospital inventory, and patient complaint system. All these systems belonged to a bigger infrastructure that covered most of examination and treatment related activities within MoH. A team was immediately formed to work on the extension. To speed up the development process, the team decided to reuse most of functionalities such as data entry forms and data dictionary management from an open source platform for health care already implemented in Vietnam, namely DHIS2. After about two weeks of intensive work, the team completed the system with basic functionalities for data entry and analysis.

The success of previously implemented systems easily convinced leaders of MoH to approve the rollout plan. An official dispatch was sent to all provincial health departments and hospitals, requesting them to use the system for reporting measles and other infectious diseases on a daily basis. Being aware of the urgency of the epidemic, most of the health facilities consented to the request and started to report the data immediately. Just a few days later, data related to measles diseases of the whole country were collected and synthesized through the system, playing an vital role in further controlling the outbreak. Compared to the eCDS, the system developed by VAMS provided data in greater detail. With regard to the case-based data, the VAMS system did not only collect basic information about infectious and fatal cases but also monitor the treatment progress of measles inpatients. The dataset included additional data elements such as the number of discharged or transferred patients, the number of lab tests, the number of patients in critical conditions etc. In total, there were more than 20 additional data elements ENS compared to eCDS. This gap legitimated having an additional system which focused on collecting clinical data of communicable diseases. Figure 1 shows an example of a number of cases admitted daily for measles collected by the VAMS system in five months between May and October 2014 from all hospitals in the county

# The new Electronic Communicable Disease System (the new eCDS)

VCom is a state-owned giant corporation originally operating only in the telecom sector. VCom has ventured into foreign markets and gained considerable success. Its international presence includes countries in Asia, Africa, and Central America. Recently, VCom has gradually expanded its business scope into other areas like software and services as the growth in the telecom market has slowed down. In the software sector, VCom has ambitious plans to develop enterprise resource planning (ERP) solutions, e-Commerce, and e-Government-related products. In 2015, after many rounds of discussion, VCom and MoH signed an agreement to collaborate in development and implementation of ICT solutions for healthcare in the period between 2015 and 2020. Following the agreement, VCom has worked with various departments within MoH to conduct situation analysis and propose plans to either improve or redevelop the systems being used in these departments. Apart from a few systems, most of the existing systems have therefore been substituted, developed or redeveloped using VCom' architecture and platform. The systems under the VCom's plans include the Official Dispatch Managements System, Electronic Communicable Disease Reporting System, Patient Complaint System, Electronic Insurance Claims and Payment System, Vaccine and Immunization Tracking System, and Online Health Professionals Licensing System.

The new eCDS is part of the holistic collaboration framework between VCom and MoH in applying ICTs to strengthen the health care sector. Prior to the agreement between VCom and MoH, two communicable disease systems had been built and put to use separately by VAMS and GDPM. After carefully examining the status quo, the VCom technical team decided to develop a completely new system to substitute the system used by GDPM (eCDS). To motivate this decision, the technical team made the following arguments. First, by virtue of the modification of Circular 48, the eCDS must be changed accordingly in order to continue functioning. Second, according to a team member, the eCDS used archaic .NET technologies (WebForm) which should be upgraded to more modern frameworks like Model-View-Controller (MVC). Furthermore, several design flaws in the eCDS could pose security risks and affect performance of the system. The redevelopment of the system also received support from the leaders of GDPM. At first, the team estimated to complete the work in only three months. However, due to many technical and communication challenges, the team spent significantly longer time than what they had planned. Ultimately, after one year, the team completed the first release with basic functionalities and sought approval to pilot the new system.

Apart from replicating all functions offered in the eCDS, the team also built additional features such as the one that automates the case-based entry task. The technical team was highly aware that most of the communicable disease cases were discovered and diagnosed by hospitals so that this feature could by reducing manual data entry increase the timeliness and coverage of reported data. According to the design, this automatic process would pull data from the epidemic notification system run by VAMS through either web services or files. In the latter case, a spreadsheet-based template file is used to make case-based data interchangeable. To implement the data exchange plan, several technical meetings were organized with representatives from GDPM and VAMS. However, the actual progress of the collaboration was slow due to the lack of adequate resources from VAMS to support for the reuired upgrade of their system.

# The communicable disease data warehouse (CDDW) for Emergency Operations Center (EOC)

In 2014, Vietnam joined the Global Health Security Initiative (GHSI), a US-led program aiming at boosting global cooperation to detect, prevent, and rapidly response to infectious disease threats. The initiative was also joined by other 25 countries and key international agencies like such as the World Health Organization (WHO), the Food and Agriculture Organization (FAO), and the World Organization for Animal Health (OIE). With the support of the U.S. Centers for Disease Control and Prevention (U.S. CDC), Vietnam has implemented a demonstration project to improve the laboratory systems and develop an emergency operations center (EOC). However, in order to run EOC successfully, it is critical to have an operational data warehouse for infectious diseases. To that end, experts from CDC and the implementing partner PATH has actively worked with local stakeholders including GDPM, VAMS, and regional public health institutions to seek necessary resources and define processes, tools, and technologies required for the actualization of a data warehouse. The significance of this was emphasized by a CDC team member as follows:

“The Data Warehouse and visualization platform will provide opportunities for historical intelligence (analysis of data in different time periods and trends) and springboard rapid disease outbreak detection and prediction. This will further aid public health resource allocation, program planning and increased communication and collaboration at all levels nationally and internationally” (Pre-meeting materials)

In a meeting between CDC and VAMS to discuss the development of the infectious disease data warehouse, VAMS welcomed the support from GHSI and asked CDC and its partners for help in upgrading the current epidemic notification system used at VAMS. A Vice Head of VAMS also recommended the CDC team to contact the first author of this paper for further discussion about the collaboration. Following that, a short meeting was arranged. During the meeting, a technical expert from CDC asked the first author detailed questions about the existing electronic patient records system in Vietnam and suggested to reuse existing systems to avoid duplication of case-based data entry.

A few months later, the first author was invited to the PATH office in Hanoi to attend a series of technical meetings of the health information system (HIS) leads from a number of MoH agencies, implementing partners, and donors to foster data sharing and communication between the systems and plan for the development and implementation of the data warehouse. Three important processes were identified during the meeting: identifying a data source for the data warehouse, defining a list of essential data elements, and consenting to health information exchange mechanisms needed for the data warehouse operations. The meetings resulted in a number of decisions that stimulated the collaboration among stakeholders. First, to quickly produce some useful results, the focus was given to building tools to analyze and visualize the collected data on graphs and maps, supporting the usage of data for decision making. Second, training and other capacity building activities would be organized to prepare adequately skilled personnel for operating EOC.

# The current relationships among the different infectious diseases systems

In this section, we move beyond the different systems and describe their relationships. The issuance of the Circular 54 and the emergence of new systems such as the eCDS backed by VCom and infectious diseases data warehouse advocated by iNGOs, have challenged the status of the two existing software systems: eCDS and ENS. In response, both eCDS and ENS have been proactively reconfigured to avoid being encroached and potentially excluded from the ecosystem.

Depending on its roles, resources, and control, the different systems have changed according to the strategic actions of its proponents:

* VCom leveraged its role as a strategic partner of MoH to problematize the qualities of the existing eCDS and at the same time proposed to offer a new and better replacement in terms of functionality, performance, and security.
* GDPM realized that they did not have resources and competence to upgrade its eCDS to comply with the Circular 54. Therefore, their decision was made to abandon their existing eCDS system and instead impose control over the functional design of the new eCDS delivered by VCom. The specification was simply that the new eCDS must offer the same functionality as the existing eCDS.
* VAMS has its major influence on hospitals and not on health centers like GDPM has. Infectious diseases data mainly come from hospitals which are better equipped and have competent human resources for case confirmation and diagnosis. However, Circular 54 mandates hospitals to report with eCDS. To sustain its role in the ecosystem, VAMS decided to extend its ENS by adding functionality that allows exporting infectious disease cases to Excel files in a format compatible with the importing functionality in the new eCDS. VCom has in its long-term plan the ambition to enable data interchange between these systems via standardized means such as web services.
* iNGOs comes as part of an international treaty between Vietnam and GHSI. This partnership allows iNGO to flexibly select the point of intervention and phase out the system development. In the first phase, they decided to reuse data collected by the existing eCDS and the infectious diseases data warehouse. At the same time, there is a plan for collaboration between iNGOs and VAMS in upgrading the ENS to be able to directly acquire data from existing electronic medical records systems in hospitals to automate data collection process and improve data quality.

The following table summarizes the relationships among the different software systems before and after the configuration:

|  |  |  |  |
| --- | --- | --- | --- |
| System | System proponent (owner) | Functionality | Changes after the configuration |
| Existing eCDS | GDPM | Capture aggregate infectious data from ward levels on weekly, monthly, and yearly basic.  Capture case-based data of critical infectious diseases. Capture detailed data of outbreaks Basic data visualization on graphs and maps. | +Temporarily feed data to infectious diseases data warehouse +A functional reference for the new eCDS |
| New eCDS | VCom | Capture aggregate infectious data from ward levels on weekly, monthly, and yearly basic.  Capture case-based data of critical infectious diseases. Capture detailed data of outbreaks Basic data visualization on graphs and maps Extended data set to cover reporting requirements of the Circular 54 Allow importing case-based data in Excel files exported from hospitals | +Import case-based data from ENS and other Electronic Medical Records System (EMRs) +Will feed data to CDDW after fully replacing the existing eCDS |
| ENS | VAMS | Daily capture of aggregate infectious data.  Capture case-based data of all infectious diseases.  Basic reporting | +Feed data to CDDW +Export case-based data to the new eCDS. |
| CDDW | iNGOs | Historical intelligence (analysis of data indifferent time periods and trends)  Springboard rapid disease outbreak detection and prediction. | +Retrieve case-based and aggregate data from eCDS and ENS for dashboard and business intelligence |

# Analysis and Discussion

In this section, we first use and develop the concept of hybrid vigor to understand the competition between software systems. We unpack and analyze dimensions of hybrid vigor to find out how hybrid vigor as a strategy has been cleverly leveraged by various actors to completely change the course of the battle.

# The battle amongst systems that offer identical functionality

The overlap and duplication of investment in health information systems in developing countries is widespread and commonly attributed to the poor coordination of international donors and local governments (Sahay, Monteiro et al. 2009). The data from our cases are contesting this widespread perception. The motivation and interplay of stakeholders’ interests have shaped the birth and development trajectories of four seemly overlapping efforts in building separate systems for infectious disease reporting. Together they make up a ecosystem delivering what is needed for disease surveillance in Vietnam.

The development and implementation of the eCDS by GDPM was the first attempt aiming at providing a holistic solution for infectious disease reporting and monitoring. It was based on significant investment and support from donors and the authorities. However, during the measles outbreak in early 2014 when it was supposed to provide timely and accurate data for epidemic control, the system was not ready due to its delay in implementation. Furthermore, the rigid design of eCDS did not allow switching the data collection frequency from weekly to other period types such as daily. It also lacked important data elements such as number of inpatients who need respirators and the functionality needed to keep track of the progress of inpatients with complications. This created a need for a system supporting the collection of measles related data from inpatients at hospitals. The epidemic notification system (ENS) by VAMS managed to rapidly fill the missing piece of functionalities that eCDS could not provide. This is what Nielsen and Sæbø (2016) describe as charting. The ENS was not built from scratch. It was an extension of a larger holistic infrastructure used at VAMS to manage various activities such as medical licensing, hospital quality, patient complaints etc. The two systems (eCDS and ENS) had some overlap in functionalities and collected data but they also provided distinct features and data that could complement each other.

The introduction of the new eCDS developed by VCom with the aim to replace the eCDS is another example demonstrating that overlap and duplication are not a result of poor coordination. It was a well-planned intervention through which interested stakeholders exercise their influence and power to challenge the established network of communicable diseases reporting. The intervention was initiated by a holistic and strategic agreement between MoH and VCom, followed by a series of situation analyses and recommendations related to the existing information infrastructure at MoH. The new eCDS was a fully replacement of the old eCDS and offered identical functionality. In fact, GDPM had requested VCom to build a new system which was as similar to the old system as possible to avoid confusing the users. However, to convince all involved stakeholders to consent to their plan, VCom cleverly inscribed innovative features on the upgrade version such as automatic importing data from hospital electronic medical records systems. This liberated users from daunting manual data entry, thus attracting them.

The data warehouse for communicable diseases coordinated by the iNGOs is yet another example that reflects the vigilance of international donors when approaching the health information systems in a developing country like Vietnam. The long-term intervention of 5 years was broken down into piecemeal phases with specific goals. The iNGOs also did a holistic situation analysis through which they were fully aware of the existence of overlapping systems used for communicable diseases reporting and monitoring. Based on their longitudinal and rich experiences with fragmented and uncoordinated efforts in many developing countries, they were very cautious to avoid any action that could lead to further fragmentation. Based on this, iNGOs selected to intervene related to the weakest part of the information infrastructure which was identified to be the data visualization and dashboard.

# Direct Competition and Mutual Encroaching

The four systems were introduced at different points in time and reflected the interests and intentions of multiple different stakeholders. The different systems offered overlapping functionality and they were all appearing as being able to be scaled to substitute each other. Unlike in a common marketplace where multiple identical products or services can freely compete and be used by a variety of customers, the preventive health sector in Vietnam at the ministry level plays a role of being the sole client. Thus, there is hardly a chance for two overlapping systems to coexist in such a setting and direct competition will result in the exclusion of one of the systems.

The mutual exclusion demonstrated between the new and old eCDS was inevitable when it was favored broadly by social, political, and technical conditions. Politically, the substitution of the old eCDS by the new version had received consent from the top management of GDPM and MoH. And technically, the VCom team had cleverly pointed out the need for the full-blown development of the new version. A more intricate process unfolded when the eCDS by GDPM and the ENS by VAMS were built to solve different issues but at the same time offering similar functionality. They also served two different sets of users: hospitals and preventive health centers. At the same time, they could both be extended to be used by both sets of users. Initially iNGO only focused on building a central data warehouse and a dashboard module that allowed visualizing communicable diseases data via charts and maps. The data warehouse and dashboard system consumed data collected through existing systems such as eCDS or ENS. However, once the data warehouse and dashboard became fully functional, it would not be too difficult for them to evolve to cover the data collection as well.

# Unpacking the dynamics of hybrid vigor

The analysis of the case so far reveals that it is not only about competition and substitution. That is more a story of different stakeholders playing the game of negotiation and strategic reconfiguration in response to the changes ignited by others.

# The politics of negotiation of functional roles between the different systems

While competing to play functional roles, our empirical data revealed that all the different stakeholders pursued strategies of collaboration, negotiation and the establishment of mutual dependencies. Each stakeholder in our case was clearly dependent on at least one other stakeholder whose interests in turn relied upon another. For instance, GDPM needed supports from VCom to make their system comply with Circular 54. VCom in turn depended on VAMS for their needs of timely and complete data, while VAMS needed expertise and supports in improving the ENS. In addition, iNGOs relied on GDPM for data to operate the EOC’s data warehouse. The circular interdependencies among stakeholders are illustrated in Figure 5.

In the complex network of socially and technically heterogeneous actors with diverse interests as seen in this case, the interdependency among stakeholders plays an important role in stabilizing and sustaining the whole network. If one system could not provide any useful functionality to the network, it would probably be excluded quickly by other software components taking their functional role.

This functional interdependency is crucial for negotiations between stakeholders to take place. A change in strategy of one actor thus will affect other actors, and thus their strategies need to be adjusted accordingly. For example, when the VCom introduced functionality to import case-based data from Excel files in its new eCDS, VAMS provided corresponding functionality to export data to Excel files that can be used in the new eCDS. Second, when iNGOs came to know that eCDS and ENS are in place and function well as communicable disease data collection tools, it has changed the approach to focus on data visualization. The two examples clearly demonstrated that the reconfiguration strategy of an actor heavily depended on the dynamics of others’ strategies and vice versa. For example, if VAMS rejected to participate in the functional boundary delineated by other actors, the eCDS might encroach upon the ENS and the ENS would be gradually replaced.

Certainly, the balance of mutual interests as analyzed in our case would not last forever. It is a result of a complex and ongoing process involving a myriad of negotiations and interessement and among stakeholders (Callon 1984). Thus, what is presented in Figure 5 is merely a snapshot of an ever-evolving network. Future actions and new actors will trigger changes that lead to the formation of newer version of the same network or completely new networks.

# The power of governmental agencies in shaping functional roles

In our cases, both GDPM and VAMS are governmental actors. They have the power to define and influence policies which favor certain systems. However, the level and scope of influence greatly depend on the functional domains that they are in charge of. For example, GDPM has the greatest control over issuance of policies, development and implementation of information infrastructure related to infectious diseases in Vietnam while according to the division of departments within MoH, VAMS is responsible for all activities related to administration of medical services at all hospitals and clinics in the country. This authority allowed VAMS to deploy ENS to respond to the measles outbreak without having to wait for GDPM to intervene. As most of the diagnoses of infectious diseases are conducted in hospitals and clinics, VAMS is the only agency that has access to the nationwide and near-real-time infectious diseases data.

Despite being unable to control the functional architecting, the two non-governmental stakeholders VCom and iNGOs had their own strategy to actively respond to the functional plan that had been sketched by VAMS and GDPM. iNGOs gained their legitimacy through the Global Health Security Initiative. iNGOs was assigned to work with local stakeholders to actualize an infectious data visualization and dashboard system as part of the US aid package in strengthing Vietnam’s ability to early detect, prevent, and control infectious diseases. As the data entry part has been covered by other systems, iNGOs shifted their focus to the data analysis part. This choice thus was significantly shaped by the powerful governmental actors. VCom maneuvered necessary political support and grasped the opportunity brought by the issuance of Circular 54 to redefine its functional role. The circular introduced various changes into the templates, structure, and flows of reports used in communicable diseases, thus requiring an update on the eCDS. This gave legitimacy to the VCom’s plan on the replacement of the eCDS by a newly developed system.

In summary, governmental agencies are important actors playing key roles in shaping functional boundaries between systems via their legal means such as decrees, circulars, decisions etc. In our case, we observed three different instances were Circular 54 played an important role:

1) It trigger a functional reconfiguration process by putting pressure on all the involved actors, forcing them to change strategy to maintain their role. Prior to that, both the ENS and eCDS existed as separated systems without any interaction.

2) It redefined functional boundaries between systems

by requiring hospitals to use eCDS to report data to health centers. This requirement posed a threat of exclusion to ENS. Before this circular was issued, there was a clear functional division between eCDS and ENS: eCDS being used in health centers and ENS in hospitals. However, Circular 54 redrew this boundary when eCDS was supposed to encroach upon ENS. 3) It promoted one system over the others by favoring the eCDS backed by GDPM and by doing so undermining the role other systems. This triggered the takeholders to cleverly act within the predefined functional boundaries to protect and ensure their interests.

# The co-configuration of individual systems and the re-configuration of the resulting ecosystem

As argued early, direct competition poses the threat of exclusion. However, threat and opportunity often go hand in hand (Hillson 2011) and higher risk usually entails higher potential gains. We could see the density of direct competitors in the sphere of communicable diseases in Vietnam as an opportunity to not only combine strengths of each individual system but also leverage the power of collaboration to engender a better II. There are a number of possible trajectories of the II evolution in directly competitive environments.

For example, in our case study, the mutual exclusion scenario would occur when one of the four systems encroach the functional roles and replace all others. In the case of hybrid vigor, however, by inheriting some or all the strengths from each individual system, the resulting system offered much better functionality compared to any individual one alone. The superior quality of the resulting system further triggered more adoption and in turn helped attract more and more support.

The functional co-configuration which was an indirectly result of the negotiation processing between systems’ proponents helped individual systems neutralize the tensions and decrease the risk of exclusion. As a result, the whole ecosystem has been transformed in an evolutional trajectory, i.e. becoming better in the sense of providing more useful and diverse functionality.

# What is exactly Hybrid Vigor strategy?

The interplay between different software components is also in the topic of the software ecosystem literature. Missing a shared definition (Manikas and Hansen 2013) the common approach is to view software ecosystems as a complex environment in which multiple software components exist and interact with a software platform (the host). Both the platform and the components on top of it are interwoven in a web of symbiotic relationships. The components depend on the platform to function and become useful while the platform needs components to offer value to the users and grow. The platform plays a critical role to the existence of its components while other components easily can be replaced and substituted. In a systematic literature review, Manikas and Hansen (2013) have identified three innate attributes of a software ecosystem, one of which is pointing to the existence of a central and common software, referring to the software platform that all other modules or components are based upon to function. From our long-term engagement with Health Information Systems in developing countries, we have observed many cases in which the interplay of multiple software systems takes place in a considerably different way. In our case, there is not one central or common software element.

In this paper, we deal with software ecosystems in which similar or peer systems compete to meet functional needs. These systems offer to a large extent identical and overlapping functionality. As in any ecosystem, to survive, each system must align and constantly realign to coexist with others. In this process, each system reconfigures itself either by reducing or expanding its functionality. A central driving force behind the evolution of the software ecosystem is in our case the legal control of governmental agencies. These agencies have in many ways the same central role as platforms in other ecosystems. We have seen that changes can be triggered by different factors such as a change in the existing environment (new policy or new requirements) or by new actors joining the ecosystem. Such changes in policy, requirement and composition of actors trigger intricate negotiating processes, and will typically change the ecosystem in the favor of some actors and at the cost of others. Nielsen and Sæbø (2015) have identified three general strategies which actors can employ to (re-)configure their functional roles in such a game, namely connecting, charting, and encroaching. Connecting refers to a situation in which different systems offering different functionality are integrated in a complementary fashion. Charting refers to a strategy where a system extends to serve unmet functional needs and freely operates with no competition. This is different from an encroaching strategy where a system is challenging other systems by introducing competing functionality with the aim to overtake their functional roles.

The three strategies discussed by Nielsen and Sæbø (2015) are relevant to understand the strategic moves of different actors and the evolutional trajectory of the ecosystem. But they are not sufficient to capture the dynamics of our case in terms of; 1) the politics involved in the negotiation between the functional roles of the different systems; 2) the power and the central role of governmental agencies in shaping functional roles; and 3) the configuration of the resulting ecosystem.

We can see clearly that the three concepts connecting, charting, and encroaching by Nielsen and Sæbø (2016) do not really offer a full analytical power in understanding our case. Even the concept of asymmetric integration by Sahay, Monteiro et al. (2009) does not give a good hint to follow the dynamics of overlapping and substitutions between direct competing systems as both of the systems discussed in their study offer relatively separate functionalities, i.e. collect aggregation data and detailed name-based data. We thus argue that the use of new concept hybrid vigor is necessary in analyzing the emerging empirical data.

Borrowing from biology and genetics, we deliberate on what we see as a fourth strategy for the reconfiguration of software in a wider ecosystem, namely hybrid vigor. Hybrid vigor is in biology the superior qualities emerging from the crossbreed of different plants or animals. As an analogy, hybrid vigor is in case of software ecosystems the emerging superior functional attributes of an ecosystem over single systems. Where the connecting strategy discussed by Nielsen and Sæbø (2015) reflects an approach towards data sharing and complementary functionality, our concept of hybrid vigor focuses on a strategy towards strengthening the functional attributes of the ecosystem as a whole. We base this concept on an underlying assumption that the existence of multiple software components is a strength and a source for innovation (Sahay, Monteiro et al. 2009).

We now go into a deeper analysis to fully unpack the potential of the hybrid vigor concept in examining and studying the development and evolution of IIs and thus software ecosystems in general. As we believe that the hybrid vigor concept provides analytical power in understanding and probably predicting the trajectories and dynamics of complex apparatus such as IIs and software ecosystems, we now go deeply in the core of the concept and linking our findings with the origin and context of the concept. We will focus on 3 main layers of the hybrid vigor concepts: 1) cross-breeding of different software genres, 2) specificalities of vigor in the context of IIs, and 3) underlying generativity mechanism of hybrid vigor in IIs. We trust that this rigor is necessary to really makes a good sense of using this concept.

# Cross-breeding of different software genres

As we discussed earlier, in biology, hybrid vigor occurs when cross-breeding of plants or animals from the same species (although in different breeds) or from the same genus. When applying the hybrid vigor concept in the context of IIs and software ecosystem, we thus also emphasizes the significance of combining software components that are closely related. For example, all the software components discussed in our cases provide functionalities to support managing, controlling, and monitoring infectious diseases. To what extends, the level of relatedness is required to enable hybrid vigor is still a question that we haven’t been sure about. The other pre-requisite is these software components must exhibits diversify in terms of design, functionalities, architecture, technology stack, stakeholders etc. For example, in our cases, the four software components are distinct as summarized in Table 2:

|  |  |  |  |
| --- | --- | --- | --- |
| Components | Stakeholders | Technology stack | Type of software |
| Existing eCDS | GDPM | .NET Web-form | Case-based |
| New eCDS | VCom | .NET, MVC | Case-based |
| ENS | VAMS | Java, Open source software | Case-based |
| CDDW | iNGOs | Java, Open source software | Aggregation-based |

The quintessence of hybrid vigor comes with the cross-breeding of different breeds of the sample species or same genus. It is hard to map exactly what are breeds and genus from biology to IIs. However, we suggest that breeds and genus in IIs could be understood as software components that do not offer exactly similar functionalities and/or using same technologies. This diversity we believe as the source of generativity of vigor in a resulting IIs. We will revisit this point when we discuss the underlying mechanisms of hybrid vigor in IIs.

# Specificalities of vigor in the context of IIs

In biology, hybrid vigor is seen as improved or increased function of any biological quality in a hybrid offspring. Biological quality can include any of the following: stature, biomass, and fertility etc. In the context of IIs, what can be counted as II quality of a hybrid II? We could borrow a list of key quality of a software system that are often used in software engineering disciplines and II literature. We then discuss them using on our empirical data. The six quality includes functionality, UX and UI, scalability, resilience, evolution (Sahay, Monteiro et al. 2009), and innovation (Grisot, Hanseth et al. 2014).

* More complete set of functionalities

Although both the new eCDS and ENS offer basic reporting functionality, they are not powerful enough to really act as an intelligent analytic tool that end-users can use to quickly identify and predict the emergence of new outbreaks and a probability of a pandemic. CDDW is developed by a specialist team with strong background and expertise in disease and epidemic modeling could provide end-users many functionalities that the new eCDS and ENS couldn’t do. For example, analysis of data indifferent time periods and trends and rapid disease outbreak detection and prediction are two set of functionalities that are completely missing in the new eCDS and ENS.

* Improved UX and UI (graphical user interface and user experience)

While UI design is sometime controversial, we believe that the combining of different software components would yield a better user experience. In our case, hospital users are already accustomed to the flow and UI of ENS, it will entail another learning cycle if they must learn and use another software. More seriously, introducing eCDS into hospitals would require hospital users to double entry, i.e. using 2 software components to enter same piece of data. This will significantly ruin any user experience. Hence, combining software will produce better user experience.

* Better scalability

Sahay and Walsham (2006) argues that in IIs scalability should not be seen as merely technical problem but should involve other aspects such as institutional, political, human resources etc. From that perspective we can see that the resulting II in our case exhibits some better traits for scalability. In Vietnam health care system, there are 2 relatively separate streams: public health centers and hospitals. While the eCDS (both old and new) has their strength in supporting public health centers, they struggle to work with hospitals. The status quo in hospitals in Vietnam is that nearly all of them are using some level of electronic patient records. Introducing eCDS into hospitals would create extra burden for hospital staff and generate fragmentation problems. By combining the two software components (eCDS, ENS), we could gain a resulting II that is already scaled to both public health centers and hospitals which yields a full coverage of data collection (Braa, Monteiro et al. 2004).

* Higher resilience

Modern software architecture best practices highlight the needs of micro-service or modular design. While this would create extra communication and integration, the overall system will achieve higher resilience in terms of failure, i.e. only partial failure. The deployment of the three software components by different stakeholders in different hardware infrastructure could maintain higher resilience of the II to confront with any failures such as hardware failure, power outages, network interruption etc.

* Faster innovation and better evolution

For a long time, II researchers constantly seek for strategies and mechanisms to boost innovation and enable evolution of IIs (Henfridsson and Bygstad 2013, Grisot, Hanseth et al. 2014). Our empirical data show that crossbreeding software components open up many opportunities for innovation. For example, using the ENS software to collect data directly from hospitals is a fine measure to ensure data quality and data coverage. Hospitals also the place that have advanced resources that could diagnose and confirm a positive case. Putting data collecting software tool into hospital is really a novelty approach in increasing data quality. The other example involves the use of intelligence tool for data analytics with software component that could bring in the capabilities of identifying and predicting an epidemic (CDDW).

# Underlying generativity mechanisms of hybrid vigor in IIs

In its original context, the mechanism of hybrid vigor is still very controversial. However, biology researchers agree to some extends that genetic mixing is the driving force for hybrid vigor. In IIs, we could derive similarities based on our empirical data. First, crossbreeding software components from different genres would allow more choices among overlapping and redundant software functionalities. Through that process, a better feature could be negotiated to replace the poorer one. For example, both ENS and eCDS are designed with reporting functionalities. When CDDW is integrated, it would replace reporting parts of both ENS and eCDS. Second, crossbreeding software components would help components complement each other. For example, ENS is accepted and functioning well in hospital environment, it would allow seamless integration into the overall epidemic II for Vietnam without any resistance from hospital users. All in all, opting dominant features and complementing each other are two underlying mechanisms for hybrid vigor in IIs.

# The resulting II for epidemic control in Vietnam and who were the followers of hybrid vigor strategy?

The current status of the infectious disease reporting system in Vietnam can be seen as the outcome of a hybrid vigor strategy with components successfully reconfigured to collectively (hybrid) offer superior (vigor) functionality:

* The old eCDS has, despite being replaced, still acted as a functional reference for the new eCDS.
* The ENS continues to play its role in the ecosystem by providing clinical data of infectious cases to the new eCDS electronically.
* The CDDW relies on data from eCDS and ENS to provide functionality for data visualizing and analysis. As the process of (re-)configuring is contingent, the evolution of software ecosystems may take many different trajectories. What we have seen is that software components which gains strong political favor may get the required backing to expand to encroach upon other components and the functionality they offer. Other components are thus marginalized and potentially excluded. While this new configuration can offer the needed functionality, it will at the same time undermine further innovation and new cycles of reconfiguration based on the hybrid vigor strategy. Actors pursuing an encroaching strategy with the result of excluding other systems may also experience political repercussion as there are vested interests in any software component. Depending on the political climate, the power of the involved actors and endogenous configurability of components, hybrid vigor can be based on different configurations.

As in any other strategy, hybrid vigor must be consciously followed by certain stakeholders in order to make it work. From our discussion, we could easily point out that VAMS and iNGOs are the two actors that have deliberately followed the strategy. We could argue that as they have been in a minor legitimate role in comparison with GDPM and VCom regarding the arena of infectious diseases data, hybrid vigor is vital to the existence of their systems. In whatever explanation, there is a need of having such “system builders” and VAMS and iNGOs have been playing that heroic role. This argument also contributes to the debate on the role of centralized and legitimate control of governmental agencies in II development. While their role is very important, it does not mean that roles of other actors should be ignored. It means that the course of a battle could be easily changed even its epilogue seemed to be early determined. Future research can further explore the scenarios that did not happen in our case, thus were not discussed: What is the likelihood of the resulting infrastructure in cases of a) governmental agencies pursuing the hybrid vigor strategy b) None of the actors following it.

# Conclusion

The findings and the discussions this paper challenges the pervasive yet simplistic view on information infrastructure that largely ignores the intricacy of the development of its software components and their overlaps, duplication, and potential substitution. Our study has offered a closer look at the information infrastructures as software eco-systems and their how they evolve. We have in particular unpacked the interplay and dynamics of directly competing components struggling to survive throughout the course of evolution. Theoretically, by introducing the concept of hybrid vigor as a strategy for functional architecting, the study provides new insights into the evolutional trajectories of information infrastructures as competing environment – as battlegrounds. Three dimensions of hybrid vigor (political negotiation, power of governmental agencies, and functional reconfiguration) have been identified to characterize the strategy and offer a better tool to analyze threats and opportunities imposed by crossbreeding directly competing components. We also deep dive into the core of the hybrid vigor concept by unpacking various aspects of the concept (cross-breeding, specificities of vigor, and mechanisms of hybrid vigor) and linking it back to its original context where it is stemmed from, i.e. biology. We answer the research question that we ask at the beginning that hybrid vigor is the strategy to follow when dealing with situations when rival software components pose threats of encroaching each other.

Our findings have several implications for both theory and practice. The tension of direct competition is in many cases inevitable, and it is in these situations important to adopt elastic and flexible strategies that can leverage and combine strengths of separate yet competing components, rather than encouraging a fight until only the last man is standing. As a consequence of the hybrid vigor strategy, the resulting ecosystem will support innovation and become more responsive to any contingency and change in the environment. These findings could go beyond the health care domain and be useful in analyzing the same phenomena in other areas such as business, social networking etc.

# References

Benbasat, I., et al. (1987). "The case research strategy in studies of information systems." MIS quarterly: 369-386.

Aanestad, M. and T. B. Jensen (2011). "Building nation-wide information infrastructures in healthcare through modular implementation strategies." The Journal of Strategic Information Systems 20(2): 161-176.

Benbya, H. and B. McKelvey (2006). "Using coevolutionary and complexity theories to improve IS alignment: a multi-level approach." Journal of Information Technology 21(4): 284-298.

Braa, J., et al. (2007). "Developing health information systems in developing countries: the flexible standards strategy." Management Information Systems Quarterly 31(2): 381-402.

Braa, J. and C. Hedberg (2002). "The struggle for district-based health information systems in South Africa." The information society 18(2): 113-127.

Braa, J., et al. (2004). "Networks of action: sustainable health information systems across developing countries." Management Information Systems Quarterly 28(3): 337-362.

Braa, J., et al. (2004). "Networks of action: sustainable health information systems across developing countries." Mis Quarterly 28(3): 337-362.

Callon, M. (1984). "Some elements of a sociology of translation: domestication of the scallops and the fishermen of St Brieuc Bay." The Sociological Review 32(S1): 196-233.

Ciborra, C. (2000). From control to drift: the dynamics of corporate information infastructures. Oxford, Oxford University Press.

Davison, R. M., et al. (2021). "Research Perspectives: Improving Action Research by Integrating Methods." Journal of the Association for Information Systems 22(3): 1.

Davison, R. M., et al. (2022). "The ethics of action research participation." Information Systems Journal 32(3): 573-594.

Garde, S., et al. (2007). "Towards sustainability of health information systems: how can we define, measure and achieve it?" Studies in health technology and informatics 129(2): 1179.

Gregory, R. W., et al. (2013). "Control balancing in information systems development offshoring projects." MIS quarterly 37(4): 1211-1232.

Grisot, M., et al. (2014). "Innovation of, in, on infrastructures: articulating the role of architecture in information infrastructure evolution." Journal of the Association for Information Systems 15(4): 197-219.

Hanseth, O. (2010). "From Systems and Tools to Networks and Infrastructures—from Design to Cultivation: Towards a design Theory of Information Infrastructures." Industrial Informatics design, Use and Innovation. IGI Global, Hershey, PA: 122-156.

Hanseth, O. and M. Aanestad (2003). "Design as bootstrapping. On the evolution of ICT networks in health care." Methods of information in medicine 42(4): 384-391.

Hanseth, O. and K. Braa (2000). "Who's in control: Designers, managers-or technology? Infrastructures at norsk hydro." From Control To Drift: The dynamics of corporate information infrastructure: 125-147.

Hanseth, O., et al. (2001). "The control devolution: ERP and the side effects of globalization." ACM Sigmis Database 32(4): 34-46.

Hanseth, O. and K. Lyytinen (2010). "Design theory for dynamic complexity in information infrastructures: the case of building internet." Journal of Information Technology 25(1): 1-19.

Hanseth, O. and E. Monteiro (1998). "Understanding information infrastructure." Unpublished book, http://heim. ifi. uio. no/~ oleha/Publications/bok. html.

Haux, R. (2006). "Health information systems? past, present, future." International journal of medical informatics 75(3-4): 268-281.

Heeks, R. (2006). "Health information systems: Failure, success and improvisation." International journal of medical informatics 75(2): 125-137.

Heeks, R., et al. (1999). "Understanding success and failure of healthcare information systems." Healthcare Information Systems: Challenges of the New Millennium: Challenges of the New Millennium: 96.

Henfridsson, O. and B. Bygstad (2013). "The generative mechanisms of digital infrastructure evolution." MIS quarterly 37(3): 907-931.

Henningsson, S. and O. Hanseth (2011). "The essential dynamics of information infrastructures."

Hughes, T. P. (1979). "The electrification of America: the system builders." Technology and Culture 20(1): 124-161.

Hughes, T. P. (1987). "The evolution of large technological systems." The social construction of technological systems: New directions in the sociology and history of technology: 51-82.

Jansen, A. and P. Nielsen (2005). "Theorizing convergence: Co-evolution of information infrastructures." Scandinavian Journal of Information Systems 17(1): 4.

Manikas, K. and K. M. Hansen (2013). "Software ecosystems–a systematic literature review." Journal of Systems and Software 86(5): 1294-1306.

Merriam, S. B. (1998). Qualitative Research and Case Study Applications in Education. Revised and Expanded from, ERIC.

Nielsen, P. and M. Aanestad (2006). "Control Devolution as Information Infrastructure Design Strategy: A case study of a content service platform for mobile phones in Norway." Journal of Information Technology 21(3): 185-194.

Nielsen, P. and J. I. Sæbø (2015). "Three Strategies for Functional Architecting: Cases from the Health Systems of Developing Countries." Information Technology for Development: 1-18.

Nielsen, P. and J. I. Sæbø (2016). "Three Strategies for Functional Architecting: Cases from the Health Systems of Developing Countries." Information Technology for Development 22(1): 134-151.

Sahay, S., et al. (2009). "Configurable politics and asymmetric integration: Health e-infrastructures in India." Journal of the Association for Information Systems 10(5): 399-414.

Sahay, S. and G. Walsham (2006). "Scaling of health information systems in India: Challenges and approaches." Information Technology for Development 12(3): 185-200.

Sanner, T. A., et al. (2014). "Grafting: Balancing Control and Cultivation in Information Infrastructure Innovation." Journal of the Association for Information Systems 15(4): 220-243.

Seawright, J. and J. Gerring (2008). "Case selection techniques in case study research a menu of qualitative and quantitative options." Political Research Quarterly 61(2): 294-308.

Shaw, V. (2009). AComplexity INSPIRED APPROACH TO CO-EVOLUTIONARY HOSPITAL MANAGEMENT INFORMATION SYSTEMS DEVELOPMENT, University of Oslo Norway 21.

Shull, G. H. (1948). "What is" heterosis"?" Genetics 33(5): 439.

Stake, R. E. (2013). Multiple case study analysis, Guilford Press.

Tilson, D., et al. (2010). "Research commentary-digital infrastructures: the missing IS research agenda." Information systems research 21(4): 748-759.

Tilson, D., et al. (2012). Change and control paradoxes in mobile infrastructure innovation: the Android and iOS mobile operating systems cases. System Science (HICSS), 2012 45th Hawaii International Conference on, IEEE.

Tiwana, A., et al. (2010). "Research commentary-Platform evolution: Coevolution of platform architecture, governance, and environmental dynamics." Information systems research 21(4): 675-687.

West, J. and J. Dedrick (2000). "Innovation and control in standards architectures: the rise and fall of Japan's PC-98." Information systems research 11(2): 197-216.