6
IT and Global Development

6.1 Introduction

The increasing role of information technology (IT) in global development has a fascinating history, especially from the viewpoint of information systems (IS). In the following, we will briefly follow two parallel threads. The first one has transformed the interpretation of the concept of global development from a set of universal indicators to the one emphasizing local potential and strengths, conceptualized as the term of bottom of the pyramid. The other thread portrays the changing self-image of IS from that of analyzing the efficiency of a given information system from the perspective of the surrounding business or governance structure toward one that makes a positive change in the surrounding societal, even global, context.

Interestingly, the two threads have recently encountered each other at the crossroads of IT that—as an artifact—has remained the commonly accepted (although also widely debated, see Chapter 1) research focus for the discipline of IS.

The definition of global development depends on the factors that are used to measure it. In addition to the economically oriented gross national product, indicators such as human development index (HDI) that measures the level of well-being, based on factors such as education, life expectancy, and standard of living (Human Development Reports, 2012), are also used. In addition to objective approaches based on government statistics, interview-based barometers address development from the viewpoint of individual, subjective experiences, such as the World Happiness Report (World Happiness Report, 2012) by researchers from Columbia University.
HDI and WHR are examples of indices that measure the development from a human perspective. Sustainable development, however, takes also into account the environmental aspect. Encyclopedia of Earth (2012) as a member-administered website shows the manifold of ecologically oriented indicators for sustainable development.

Indicators and definitions are keys for analyzing the pace of global development. However, the crucial challenge is to make development take place. This requires setting goals of which the UN-defined Millennium Development Goals (2012) are the best known at the moment. The recent Rio + 20 Conference promoted the process toward sustainable development goals.

Complementary to the needs-focused MDGs, the bottom-of-the-pyramid approach (Prahalad, 2004) emphasizes the potential of the poor communities as a consuming market as well as a producer. This is a particularly interesting viewpoint from the IS perspective, since one of the main products for poor communities are mobile services—which also can market the communities’ own production to the global markets.

The other thread, parallel to the emerging theme and variations of global development, is the redefining scope of IS as a discipline. As coarse, objective and global development indicators are fabricated into more focused and detailed ones for limited contexts, to achieve better accuracy, IS research is getting more focused on improving the life of an individual user or a user community. For example, the generalizability of research outcomes seems to give way for their transferability, from one user or context to another one.

The traditional agenda of IS research has been to improve business and governance processes by technology (see, e.g., Nilsson et al., 1999). Over the years, the agenda reduced to analyzing the usability of off-the-shelf solutions in given user scenarios. The research has followed the positivist approach for explaining and predicting the usefulness of technology for the users’ performance and applied quantitative methods.

As neat and controlled the laboratory-oriented approach of IS might be, applying the natural science paradigm to the observed human users of technology, the agenda easily leaves IS in a disciplinary isolation, trapped by its own methodological dogmatism. Recently, this trend has been criticized by, for example, Walsham (2012) and Niederman and March (2012). Whereas Walsham calls for an interdisciplinary openness that allows IS to see whether ICTs make the world a better place, Niederman and March argue for a clear role for design science and action research that allows IS research to make a difference for the user communities.

The rebirth of IS as a world changing discipline—that would also inspire young talent and attract research funding—is built on the dual roots of the discipline. American pragmatism by Dewey and others emphasized the complementarity of thought and actions: what works is true. The continental critical theory, dating back to Marx, Nietzsche, and Freud, differentiates what is wrong from what is right. This academic heritage is the background of a discipline for a better, developing world, to be combined with the recent trends in IS research for design science and action research.

What many IS researchers (but not all; see the debates reported in Chapter 1) seem to agree upon is the role of IT artifacts as a focus of IS research. An IT artifact refers not only to a technical implementation, like product, service, application, or system, but can also be a construct, a model, or a method (Niederman and March, 2012). The consensus paves way to the very encounter of global development and IS.

For pragmatically oriented IS research, IT does not remain an observable artifact, but becomes an artifact for innovation. This is where IS meets with the challenges of global development: an innovation changes the existing practices, routines, and processes (Denning and Dunham, 2010). The agenda is also aligned with the expectations of and fears from IT in the context of developing countries. Critical approach is required to distinguish between the threats and opportunities where the affordances and limitations of IT take their users and the surrounding communities, even nations. For instance, the role of SMSs for improved communication is different before and after a revolution. For the oppressed and the ones in power, a simple SMS can carry a completely different meaning—even if the person in question is the same.
The global stage for the encounter of IS and universal development challenges is scaffolded by the recently emerged penetration of mobile technology, especially in the world’s poor communities. The poorest one billion people at the bottom of the pyramid are not only using innovations imported from outside, but actively bringing about novel applications for changing their everyday. The potential of the whole new user group is also a crowd for sourcing innovations (Howe, 2006). The innovations from the grassroots might not only be incremental but also radically change the conditions of life. From the organizational learning point of view, we talk about double-loop learning instead of single-loop learning (Argyris and Schön, 1978).

From the viewpoint of an IT user, development always takes place within an immediate, physically or virtually local context. This observation sets the background for the current article. Therefore, rather than portraying the global image of IT users (Dutta and Mia, 2011), we focus on how IS research can develop an ordinary life.

Following the newly emphasized interdisciplinary orientation of IS, we will deliberately move from the IS-centric discussion to the dialogue between IT and global development. This is the de facto agenda of the renewing, interdisciplinary IS: to analyze the encounter of people with IT and have technology make a positive change to the lives of these people.

In many ways, the concepts of global development and IT seem slightly outdated from the mainstream of contemporary discussion. Development was on the global agenda at the turn of the millennium when the United Nations accepted the list of Global Development Goals (MDGs), to be reached by 2015. At about the same time, the expectations from IT were at their peak years. Within the cabinets of political decision makers, the challenges of global development and the opportunities of IT are still at the status as high as that of their politically correct promoters, but early adopters and change makers have already chosen alternative paths of uncertainty and new sources of inspiration, as the Arab spring has shown.

The intent of the chapter is to show that when juxtaposed with or cross-fertilizing one another, global development and IT can refresh, if not transform each other. In fact, global development, among other vast challenges of the humankind, offers a particularly interesting call for information technologists. But to comprehend this opportunity, one must go to the roots of both concepts, like for any renaissance.

What does global development mean from the viewpoint of IT? Who is the client of IT solutions that can launch the first step toward positive change and progress that the term development refers to? Is he/she a street kid sleeping in a wasteland next to a five star hotel? Or is he/she a president of a developing country? Or is it an anonymous group of bottom-of-the-pyramid customers that buy airtime for their text messages? From the IT perspective, there is nothing truly global in development, as there are very few, if any, universal solutions to all seven billion people. Development, although global as an abstract phenomenon and challenge, is local in contents and concrete customer requirements. Relatively, a poor community at the outskirts of a rich city may benefit from IT far more than a well-off landowner of a developing countryside. Therefore, rather than global development or a developing country, we will use terms such as local development or developing context, even a developing individual. From the IT perspective, a developing context is a challenge that technology can meet with information.

What, then, is IT from the aspect of a developing context? Customers in a developing context want to improve their lives by relevant, just-in-time, and functional information that requires technology. Far too often, these expectations are latent, unknown to those in the need of information, but likewise to potential solution designers. But when a street kid connects to an online supporter and can get regular food coupons as a text message, he can experience that the information works for him. Or when the president can promote equal access to learning basic literacy and numeracy skills, his/her nation can break the schooling barrier. And when a group of handicapped children in a remote Tanzanian city of Iringa get smart blocks for their rehabilitation (Lund, 2012), the information hits their bodies, not just minds. From the point of view of a developing context, IT is an opportunity beyond expectation.

The whole spectrum of information is relevant for a developing context: the information needs to exceed that of administrative databases or network protocols. Sensory data can be used to analyze...
the fertility of a soil or the muscular movements of a child with cerebral palsy; climate information can be refined to expand the knowledge of a small farmer toward that of an expert. In most developing contexts, the technologies should cope with and even make use of uncertain information. The concept of awareness with respect to information can be understood technically as in context-aware applications for unconsciously used data: a mobile device is aware of its user’s position and the user does not need to pay attention to that awareness. But awareness can also be used along the tradition of Gestalt therapy, as in tools that promote HIV and AIDS explicit awareness for the user—far from the unconsciousness-related connotation of the term. Digital storytelling, a natural platform for presenting information especially among illiterate users, would even require technologies that convey emotions. The diversity and originality of the demands for IT in developing contexts are indeed apparent already within the concept of information: its types, representation, visualization, and reliability. This starting point provides challenges for IT researchers at the fundamental level of information itself.

Table 6.1 exemplifies state-of-the-art research approaches in the intersection of IT and global development, categorized by the type of the IT artifact studied and whether the motivation is on global development as a challenge or IT as an opportunity. For implementations, or instantiations of constructs, models, and methods, the division between challenge and opportunity loses meaning, because the application-oriented instantiations are based on an explicit challenge, but portray either a design or an analysis (or both) of an IT opportunity that meets with the challenge. The table also illustrates the diversity of the research area. This is why this chapter sketches a scaffold for the research area, based on a conceptualization of computing as a context-oriented discipline.

Classically, computer science has been positioned in the crossroads of mathematics, natural science, and engineering (Wegner, 1976). Methodologically, it has inherited one main principle from each of the scientific parents, namely, abstraction, modeling, and design, respectively. When the area of computing has expanded from strict computer science, computer engineering, and software engineering, the human and social aspects have increasingly gained relevance also in the methods of computing. This has had consequences for identifying user scenarios, developing user experiences, understanding usability issues from a cultural viewpoint, and developing ethical principles for IT (for cultural usability, see, e.g., Kamppuri, 2011). The interplay of the diverse scientific traditions of IT becomes very challenging in developing contexts, which differ significantly from those that computing grew within. For an illiterate farmer, climate information on a mobile platform might be represented as a story. A customer will transform the traditionally Western interpretations of abstraction, modeling, and user experience.

### Table 6.1 Examples of Research Approaches on IT and Global Development by the Type of IT Artifact

<table>
<thead>
<tr>
<th>Type of IT Artifact</th>
<th>Focus on Global Development as a Challenge</th>
<th>Focus on IT as an Opportunity</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Construct</strong></td>
<td>Ethnocomputing (Tedre et al., 2006)</td>
<td>Emergence of ICT4D as a research area (Unwin, 2009)</td>
</tr>
<tr>
<td><strong>Model</strong></td>
<td>A posteriori analysis</td>
<td>Future-oriented agenda of IS (Walsham, 2012)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ICT4D business models (World e-Business Initiative [<a href="http://www.worldebusiness.org">http://www.worldebusiness.org</a>])</td>
</tr>
<tr>
<td><strong>Method</strong></td>
<td>Design-reality gap (Heeks, 2002)</td>
<td>Improving mobile interfaces (Medhi et al., 2011)</td>
</tr>
<tr>
<td><strong>Implementation</strong></td>
<td>Interventions in application areas such as health, governance, education, agriculture, banking, tourism, business</td>
<td></td>
</tr>
</tbody>
</table>

© 2014 by Taylor & Francis Group, LLC
With its cross-cutting nature, IT is raising obvious expectations in developing countries. This far, the encounter between global development and IT has taken place in very large-scale projects. The approach goes back to the heroic, hall-of-fame type of understanding of global development and IT. Identify a major problem and attack it with well-trained troops carrying massive weaponry. Typical examples are extensive networking projects, teaching laboratories with hundreds of computers, or national programs for establishing community multimedia centers. A techno hero, trained outside the battlefield, is expected to plan and establish an extensive technical infrastructure to combat ignorance. But ignorance prevails as long as activities or services on the top of the infrastructure are not available. Plain walls or cables do not advance anyone else's progress than that of their constructor's. The projects are mostly funded by massive development aid programs whose benefits are getting more and more controversial and debatable (Moyo, 2009).

Recent concretizations of IT-flavored megaprojects are various programs with a focus on information or knowledge society and innovation; that is, services on the top of the technical infrastructure. The almost miraculous success of the global north, in particular natural resource poor Nordic countries, has been understood to be based on innovative designs and uses of IT. The Nordic road is an attractive progress story that most developing countries want to learn from and adapt to. In essence, national innovation system programs in developing countries aim at a level more advanced than building a plain technical infrastructure: they are building a national system of innovation or a whole knowledge society. While this goal setting looks at the first sight a bit more elaborated than just an extensive setup of material infrastructure, typical components of the innovation system might not take an intended beneficiary, like an individual farmer of a poor community, very far in their concrete development expectations. In order to succeed, incubation or funding services need to address the contents and links to the citizens' everyday life.

Like the devil is told to dwell in details, the challenge of twisting—or, indeed, distorting, see later—IT into a vehicle for progress is in individual encounters, at the very local level, at the bottom or base of the society's pyramid; although, interestingly, a truly networked society has a topology with neither top nor base. The contents and relevance in innovative IT designs for development can only be born at the grassroots.

Interestingly, the metaphors of the current IT streams reflect the ideological change from comprehensive, one-size-fits-all mega solutions toward personalized ones. At the time of ecosystems, cloud services, or grid computing, we are most excited of the individual components vitalizing the systems. Supercomputers do not raise the excitement of ubiquitous technology. In terms of the solutions, the perspective is changing from delivery-oriented electronic malls to piazzas where everyone can set up their small digital apartments, stages, and garages. With facebooks, twitters, and microcredits, individuals can make their expressions known, their chirps heard, and their talents realized. This is exactly what developing contexts need.

Developing communities can make use of IT-based innovations. However, it is essential to distinguish between various genres of innovation. The distinction between aiming at incremental or radical innovations (Henderson and Clark, 1990) has consequences for how IT takes up its role as an agent of change. The traditional mega(lomaniac) approach usually ends at incremental innovation in IT uses. It orients toward conservative changes: users live as earlier but with improvements. Text messages help life, but do not necessarily give value for money in terms of new challenges or alternative jobs. A radical innovation changes the rules of the game, transforms living conditions. A mobile endoscopic surgery unit with specialists consulting from overseas saves lives. A radical innovation, very often, is born in the periphery, behind the infrastructural services and out of the mainstream. We can talk about pockets of development by pocket technologies, in local and limited contexts.

A radical innovation at the grassroots—"radical" comes from the Latin word radix, for root—makes a creative use of IT as a source of innovation. But individual radical innovations need a way to the mainstream. The recent discussion on innovation emphasizes this dynamics between grassroots and periphery on one side and the mainstream on the other. Open innovation (Hagel and Brown,
Disciplinary Foundations and Global Impact

2008) refers to facilitating a movement where finding solutions to individual and open challenges can be outsourced to crowds: we talk about crowdsourcing (Howe, 2006). IT offers extensive opportunities for crowdsourcing by giving a face and voice to individuals, independently of their backgrounds. The potential of the young people’s energy and fresh ideas can become the radical solution to many of the globe’s problems.

One of the main lessons that every foreign expert in a developing context understands sooner or later is that a solution is viable only if it brings benefit to all those engaged in the project or initiative. The principle easily brings about jealousy, misunderstanding, and mistrust, especially in well-funded mega projects. Accusations of corruption or waste of resources start to emerge. However, the whole idea of technology is the benefit for people, and the participatory approach should treat funders, decision makers, designers, and users at an equal basis. At the same time, the nature of the expected benefits might vary significantly between diverse partners and stakeholders. However, demanding benefit for all is a useful starting point for IT development. An individual community in Chiure, northern Mozambique, built their multimedia center with their own money. In order to increase their benefit, they need connectivity, that is, infrastructure. The government will get goodwill for responding to this need. The benefit for all can only happen when all join in the effort, but everyone also gains.

Developing contexts offer several examples, such as mobile banking and e-health, of bending available technology into novel tools that have brought about prosperity and wealth by information and its communication. Whether already success stories or yet weak signals, they challenge IT professionals to familiarize themselves in the dynamics of development, so that they can proactively recognize latent demands for information within developing contexts. Innovative solutions, with their origin in developing contexts, will reshape the field of computing that has far too long become so conventional that IT studies face recruitment problems among the globe’s brightest young minds.

6.2 Underlying Principles

This chapter portrays the common ground for a developing context and IT that is designed and used in the context. IT is a cross-cutting technology that has potential to boost diverse areas of a developing society. The contextual challenges may divert the face of IT from the image that it is known of in developed countries. We will analyze the diverted image of IT from the four viewpoints of modeling, design, abstraction, and human aspects that we will call encounter. Although these perspectives were originally identified to position the discipline of computer science onto the map of sciences, they work well in setting up the stage for IT in a developing context.

In a developing context, the principles of construction and ethnocomputing (Tedre et al., 2006) will shape the four perspectives on computing and, hence, IT artifacts (Figure 6.1). Unlike the mainstream of the ICT4D (ICT for development) (Unwin, 2009) movement that is mostly interested in analyzing the impacts of the transfer of available IT into developing contexts, we take seriously the constructive agenda of computer science (Sutinen and Tedre, 2010). This means that the theoretical emphasis shifts from a conservative and controllable evaluation toward radical and risk taking art of creative design, development, construction, and implementation. A developing context will divert its challenges to artifacts, including products and services that are not yet even dreamed of among the IT users or developers of the global north.

In order to construct, a developer needs to find a comfort zone in his/her workshop. In developing contexts, not only the information technologies but also their related concepts are of alien origin. Similar problems have aroused in other fields, like learning mathematics. A set of ethnomathematics or ethnoarts have answered to the question: ethnomathematics, ethnomusic, ethnobotany, and recently ethnocomputing. Whereas ethnomathematics (Gerdes, 2007) is interested in different representations of mathematical concepts, ethnocomputing emphasizes that everyone should be able to enter the world of IT from his/her own background. For example, programming courses
might start from contextually important problems rather than from the conceptology of variables and statements, based on the Western science.

## 6.2.1 Constructive Modeling in Context

The modeling perspective to IT refers to its roots in natural sciences. The key idea of a model is a set of variables and a function that maps the variables to predict the phenomenon that the model is assumed to portray. In IT, the predicted phenomena cover a wide spectrum of automated processes from the behavior of algorithms up to the user experience of applications. Estimating the performance of an algorithm is an example of a quantitative model, explaining the experience of using, say, a social media platform might require a qualitative approach. For IT artifacts in the developing contexts, the most interesting phenomena to be explained, predicted, or interpreted are the **functionality** and **relevance** of the systems and services in physically, culturally, economically, or politically demanding conditions. Functionality denotes the extent to which the evaluated technology works according to its expectations that can vary throughout the modeling process. Relevance measures issues like the benefit from, added value, cost efficiency, or robustness of, and improved quality of life promoted by the technology in a given context.

Along the lines of modern science education, the empirical paradigm calls for modeling that is a learning process based on observations in the perceived surrounding reality, rather than previously derived, existing theories. The contemporary pedagogical approach is contradictory to the earlier one where a lesson started from the teacher’s solo introduction to a theory that the students needed to learn by heart; the students’ own activities with applications and examples were secondary to the teacher’s presentation. The old approach believes in localizing the commonly approved standards in students’ minds, the modern one in students building their own insights from their own observations. When measuring the learning outcomes, the first approach emphasizes global standards, the latter the contextual relevance of what has been learnt.

Often, the transfer of information technologies to a developing context is analogous to the outdated pedagogical model. Technologists seem to believe largely in massifying, delivering, or generalizing the so-called best practices: technologies that have shown to work in a similar context are assumed to be ready for others to adapt to, or learn from. This understanding materializes in countless seminars, workshops, conferences, and courses that call policy makers to learn by heart IT systems, services, or solutions that have shown to be functional or relevant by the models developed elsewhere. One of the
key ideas of computing—modeling, that is, building a model—has been replaced by imitating, that is, following the given model.

In fact, an empirical learning process sets the foundation for a process of building a contextual model for understanding the functionality and relevance of a given IT in a given context. In essence, an empirical modeling process always involves trial and error: the measures to understand the functionality and relevance of the technology are iterated by the feedback from the context. The iterative modeling process goes hand in hand with its parallel of participatory design. Whereas the design process results in artifacts, also in iteration, the modeling process iterates the criteria against which the functionality and relevance of the artifacts are measured. The variables determining the functionality and relevance of the technology vary from context to context. Hence, the criteria for functionality and relevance cannot be given beforehand, or standardized. For example, a mobile climate service for a user group of illiterate farmers in an electricity-secure context requires criteria different from those of a similar service to industrial farmers in extremely remote places: the criteria for both functionality and relevance are different.

The analysis mentioned earlier shows that the modeling process is a natural part of the construction of an IT in a given, developing context. It allows developers to tackle real challenges by making use of the opportunities of IT in a bottom-up, empirical, organic way. The modeling process turns feedback from observations into criteria for improving the functionality and relevance of the systems and services to be designed.

Somewhat surprisingly, robotics gives an opportunity to derive a potential user’s own understanding of a computational process in a concrete way in a developing context. For instance, Lego robots have been used in IT education to promote self-confidence with technology. Programming robots opens the black box of IT in a way that sparks the creativity of local designers (Sutinen and Vesisenaho, 2006).

Quality assurance is a standard part of the modern design and construction process of an IT system or service. However, in many cases a certified production process is only assessed by superficial quantitative factors or, even worse, by the outcomes that are called performance indicators. For example, in an educational process a certified primary schooling process might count the percentage of children that have taken IT lessons at school, but pay no attention to the competences they have at the time of graduation, or the way they are using computers 1 year after finishing the school. The quality assurance factors and indicators are normally set before the process, thus resembling the outdated theory-driven science education approach.

The definition of quality as the relevance of the contents (Machili, 2012) shifts the focus from the process to the contents at its each step. While the empirically derived, emerging model of an IT system or service focuses on functionality and relevance issues, each modeling iteration improves the quality of the system or service as understood as the relevance of the contents of the system or service.

Modeling answers to the question “Why does a system or service function in a relevant way?” “Why” is also the main question of any natural science inquiry to the observed phenomena in the surrounding physical world.

### 6.2.2 Constructive Design in Context

The design approach in computing stems from the engineering legacy of the discipline. In developing contexts, engineers have typically been invited as constructors and builders implementing designs that have their origins outside the context. Hence, from the design point of view, engineering has been neither constructive nor contextual. This applies also to most IT systems and services, with exceptions like mobile banking where Kenya is among the global pioneers of novel designs.

Participatory design is a movement within design that calls for ordinary users to join a design process. Culturally, the approach is of Nordic origin: it is a design incarnation of the political egalitarianism of the Nordic countries. In this approach, participation does not separate designers from the rest of the society into an own class of ivory tower experts, but invites them to learn from and struggle with laymen...
for innovative designs. Participatory design can also be understood as a consumer movement: consumers are seen the best experts of not only the everyday realities but also the criteria for the functionality and relevance of the solutions to be designed. As a bottom-up or base-of-the-pyramid-up movement, we can call participatory design contextualized or contextualizing design.

The inherent dilemma of participatory design—obvious particularly in developing contexts—is that the users are not aware of the opportunities of technology and for this or other reasons do not have imagination for novel, out-of-everyday designs. This is also reflected in Steve Jobs’s comment that radical designs can never be achieved by listening to users’ needs. This is certainly true if the users’ imagination is not empowered. But designers can also be intellectually lazy or tied to their own ethnocentrism. A latent obstacle making designers reserved for a contextualized approach is the common attitude that developing contexts should use affordable and tested second-hand technologies. This attitude is based on the understanding that every technology has its predestined sequence of steps that every culture should go through. Hence, typewriters should be used before text processing, landline telephones before mobile phones, dull phones before smart phones, and smart phones ahead of tablets. As we know of several examples of developing contexts leapfrogging the predestined steps, the idea of predetermined sequences of technical developments does not hold.

Because of the challenges described previously, participatory design requires close dialogue between professional and laymen designers, as to discover the latent demands for relevant designs and to match them with the opportunities that the advancing, not-yet-available or not-yet-designed technology provides. A design dialogue is a reciprocal learning process between the professionals and the laymen. The professionals need to learn the resources and requirements of the future users of the to-be-designed technologies. The laymen have to learn the principles of technology and get hands-on experience with it to the extent that they can use their imagination to ideate what is possible. A fundamental prerequisite for a mutual learning process—that might take several years—is trust among the professional and laymen designers, because the users might be shy to explicate their still very vague ideas. A fast and politically correct exchange of system specification forms is but a remote image of a committed dialogue for sustainable designs.

One of the ancestors of the participatory or contextual design approach is the Bauhaus movement from the Germany of the 1920s. The Bauhaus movement was a counter reaction to the success story of industrial design of the contemporary England. As a movement, Bauhaus emphasized interdisciplinary and explorative design communities for functional solutions. A concurrent parallel of the Bauhaus communities are living laboratories that accommodate technology developers to implement and analyze a design process in situ, within a real-life context. As their best, living laboratories are pockets of development that radiate their successes and enthusiasm to the surrounding region that—particularly now at the time of social media—might easily extend its physical limitations. A living laboratory might be a street studio that gathers street vendors to share the ideas of their business strategies with designers that are creating context-aware applications to city dwellers. A creative design milieu is an environment that features the characteristics for a novel and fresh design.

Unlike the bottom-up approach to constructive design in context, many international development aid funders are stuck in the outdated top-down ideology, leading to strictly hierarchical planning. The planning scheme with its follow-up monitoring mechanisms is called the logical framework (AusGuideline, 2005), logframe for short, and it is, in theory, based on a temporal logic model for understanding the dependencies between various components of a project. While the logframe can be used for projects that follow a linear sequence of predefined steps, like those of building a house, it is unsuitable for bottom-up technology design projects with—hopefully—several simultaneous and unexpected inventions, stages, and iterations. The many simultaneous threads of a contextual design project make it impossible to represent the project as a hierarchy. Structured as a hierarchy, the logframe does not allow nor support links between the leaves of the tree-typed structure. However, for an innovative design project of an IT artifact, the most exciting opportunities are based on combining components distant from each other.
Despite all its weaknesses, logframe can be used as a structure for planning a technology design project. The reason is that it is closely aligned with needs-based design. An expatriate designer, whether an individual or a group of experts, is tempted to understand a user community from the viewpoint of their assumed hierarchy of needs: the needs hierarchies can be easily identified outside the user community as a desk research exercise or plainly copied from sources like the MDGs. By the logframe approach, the community can easily be imposed by a structure completely unfamiliar to them. The serious shortcoming of the logframe approach is that it does not fit into a reality with a diversified structure, different from but more valid than its Western, hierarchical representation. Indeed, the diversion is in the viewers’ eyes: the hierarchical representation might be equally diverted from a viewer representing a developing context as a developing context can seem chaotic to a Western professional designer or planner. In any case, the logframe is unable to accommodate unexpected, emerging threads that a contextual design process is known of. We conclude that as a design tool, the logframe is based on the outdated waterfall model. The modern agile development should replace it especially in the developing contexts.

In essence, participatory design is aligned with the strengths-based rather than needs-based orientation for design expectations. Unlike the needs-based approach based on the undisputable lists of given targets, the strength-based approach starts from the given resources. While the needs-based approach leads to a simplistic and linear design process consisting of a sequence of predetermined steps, the strengths-based approach requires explorative and creative screening of resources that are not included in any given wish list of ideal contents: the resources might be hidden and unexpected. Whereas the needs-based approach is based on the negatively flavored list of what a community does not have, the strength-based approach is positive and based on what is available.

As a constructive design approach in context, the participatory design process makes IT an agent of change in the design community and beyond it. The design process makes its milieu a cradle for fresh designs. Because of its designed products, the cradle starts to attract people from outside. What was a laid back community from the viewpoint of technology has a potential to transform to an origin of truly new technologies.

A properly organized participatory design process is an answer to the How question of constructing functional and relevant IT.

### 6.2.3 Constructive Abstraction in Context

The role of abstraction as a key approach in computing goes back to the mathematical roots of the discipline. Mathematics forms the foundation of any automation and is seen in the formal representation and language of computational processes, and the interrelations and functional dependencies of the formal entities. While many of the formal structures of computing, such as database theory, live their own, independent mathematical life, they originate in the real world problems and their formal representation, like that of occupying flights or arranging a census. As a reminder of their real-life background, formal structures have physically descriptive names such as trees for certain types of data structures or divide and conquer for a class of algorithms.

While modeling and design as threads of computing were almost straightforward to interpret as underlying principles of constructing IT artifacts in a developing context, abstraction requires more elaboration. The most natural approach is to analyze abstraction from the viewpoints of representation and language.

Ethnostudies and ethnoarts, in this case particularly ethnocomputing, refer to the ways that culturally central principles, ideas, or forms of expression have been used as entry points to understanding or practicing a certain field of science or art. Ethnocomputing suggests that the language of computing can be learned from one’s own culture, not necessarily following the standard steps of a universal curriculum. In practice, an ethnocomputing approach to learning programming in a developing context can take place by constructing programs for gadgets made of physical blocks. Ron Eglash’s monograph
African Fractals (1999) gives an example on how fractal, self-iterative, or recursive expressions repeat in West African architecture and how they can be used in (learning) computing.

Contrary to the common belief, mathematics is not *per se* a discipline of a theoretical or deductive character, at least when created. According to Polya (1957), mathematicians work as inductive experimenters, not deductive theorists, and this should be more transparent in how mathematics is taught. From the viewpoint of abstraction in IT, Polya’s message translates to IT explorations, which are inspired also by cultural contexts, along the lines of mathematical adventures by de Guzman (1986).

The idea of abstraction calls for unprejudiced theory and concept formation in the developing contexts earlier unexposed to IT, based on identifying the computational interpretations and formulations of their real-life problems. Coping with tolerance, uncertainty, and diversity, whether in natural languages or cultural representations, can form starting points for indigenous abstraction. In many contexts, storytelling might be an appropriate representation for even the core of computing, like programming.

Indigenous abstraction can serve as a counterattack to the conceptual counterpart of technology transfer, namely, concept transfer. Almost any IT conference in developing countries seems to repeat the global slogans, such as the quest for community multimedia centers or integrating IT in school curricula. While innocent and sometimes even beneficial initiatives in themselves, transferred concepts seldom have carried their original meaning to the audiences of their importers.

Abstraction names and positions the inner mechanics of IT. Thus it generates the navigation or mapping system for IT. Each cultural context might need to (and, indeed, needs to) identify IT using its own representations. Abstraction answers to the What, Where, and When questions related to a technology that needs to be in the comfort zone of its users.

### 6.2.4 Constructive Encounter in Context

We call the human perspective to computing an *encounter*, to emphasize that the development and use of IT takes place in an encounter that always involves people, whether as intellectuals searching for inventions, nerds enjoying the fascination of computational devices, or ordinary users eager to improve their everyday living.

Much of the research and practice of the users’ encounter with IT in developing contexts sees people as objects using the technology. This is called an evaluative approach of ICT4D. The objects need to be understood, their needs analyzed, and IT solutions either modestly bent accordingly or, what happens also in developed countries, the encounter simplified by training the objects to use technologies that are far from their everyday needs and unnecessarily hard to learn and master. Rather than empowering people in their everyday life, the technology makes them its slaves. We call the evaluative view of IT encounter an *objectifying* view of users.

A constructive encounter between IT and its users in their context sees people as subjects who want to influence on their situation and make a difference. It is thus aligned with the participatory design approach. However, instead of engineering solutions, the encounter approach focuses on the *human rights perspective*. It promotes aspects like digital rights, ownership, access, and collaboration within diverse user groups. We call the constructive view of IT encounter a *subjectifying* view of users.

Whereas the objectifying view evaluates the extent to which users can *access* information by technological systems and services, the subjectifying view is interested in the opportunities that people can use technology for creating, delivering, and promoting the full spectrum of information: data, knowledge, and awareness. Access of information is usually identified with digital rights. The latter view emphasizes the *ownership* and sharing of information and tools that allow people to process, generate, elaborate, and modify information. Let us consider an example from the area of e-government. The objectifying view evaluates the e-government services from the perspective of government as a service provider and taxpayers as service users. The subjectifying view is interested in the construction of digital platforms that facilitate the citizens to make their voice heard.
Disciplinary Foundations and Global Impact

From the human rights perspective, the objectifying view is interested in how human rights set standards to information services and access thereof. Hence, the services are analyzed against the given human rights. According to the subjectifying view, the focus is on how people can use IT to promote human rights, even those yet unheard of. Awareness is of a particular interest to human rights, especially from the viewpoint of a developing context. Awareness has two connotations. Awareness can be unconscious and latent or conscious and explicated. As latent, awareness can be understood as a smooth and transparent access to information, and it is based on human rights in the objectifying sense. Explicated awareness is close to emancipation by information, and thus, it promotes human rights in the subjectifying sense. For instance, an emancipatory HIV and AIDS awareness campaign allows its participants to share their stories of coping with the disease. Interestingly, as awareness, also context has both the unconscious or transparent and explicit or manifest connotations. In order for IT to contribute to the development of a context, the members of the context need to be fully aware of that they belong to it.

As an indication that MDGs are aligned with the objectifying view of IT users, the MDGs do not refer to ICTs promoting human rights while they cover, for example, environmental sustainability and global partnership that can set standards for evaluating IT access from the human rights perspective. The UN declaration of human rights (2012) is equally vague in how IT can promote human rights, see relevant articles 17 (right for one's property), 19 (information), and 21 (public service). In general, in developing contexts IT has a significant role in promoting dialogue between conflictive human rights, for instance, those of environmental awareness and the rights of the owner of a mine (http://en.wikipedia.org/wiki/Human_rights#Human_rights_and_the_environment).

The Arab Spring in 2011 showed that IT is in particular a human rights tool for the young population. The challenge is in how IT can help them to join forces to tackle development challenges in their contexts.

The encounter aspect of IT answers to the Who questions—who, by whom, and whose—of IT.

6.2.5 Toward Life Computing

The underlying principles of IT and global development answer the fundamental challenges of making IT a tool for development: Why does it work (modeling)? How does it work (design)? What works, where and when (abstraction)? Who is the subject of IT (encounter)? The answers set up a scene where a diversity of experts and users make use of observation and measuring instruments for modeling; shared workbench for design, various languages, representations, and navigation tools for multilayered understanding; and refresh their human rights for active encounters. As understood in this way, IT is in the center of human activity for a better world, and it can be called life computing.

6.3 Impact on Practice

Even if the landscape of IT within developing contexts is complex and sporadic, it is easy to observe several instances where it has dramatically changed not only the way the discipline of computing is understood by professionals but also the conditions of its users. An obvious example is the mobile phone revolution among the poor people: people at the grassroots are using technology for everything from getting a job up to making a revolution.

Due to the diversity of IT uses in developing contexts, this section can but portray pockets of the impacts that IT has had on people’s lives. The perspective is that of individual technologies, their designers, and users.

6.3.1 Impact on the Discipline

We will go through the foreseen impact of IT and global development on the discipline of computing by the underlying principles of modeling, design, abstraction, and encounter.
Modeling emphasizes the empirical orientation in the context. The interdisciplinary orientation that is required for IT to make a difference in a given context strengthens the role of humanities in the discipline.

Participatory design generates unconventional crosses between disciplines. Environmental challenges in developing contexts and the demand for renewable local energy connect IT to forestry. Opportunities of modern endoscopic surgery allow specialized pediatric surgeons to use highly advanced technology even in remote areas: this sets demands for using IT for learning modern surgery in real contexts. IT can support a farming community throughout the lifetime of a crop from purchasing the seeds until selling the harvest: successful mobile service design requires collaboration between meteorology, agriculture, anthropology, sociology, and IT.

In the area of abstraction, digital storytelling generates novel representations for IT. The contextual orientation transforms usability toward cultural usability, and replaces the idea of localizing universal services to contextualization that starts from the requirements of a given context.

Encounter emphasizes ethics and an anthropological orientation throughout the discipline. For instance, the anthropological concept of awareness transforms standard user expectations to those promoting the metacognition of the user’s behavior. The subjectifying orientation of the actor calls for open platforms, for multiple contributions, on an equal basis. This is foreseen in the open source movement that paved the way to the culture of tweets and micromovies.

The general trend common to all the impacts above leads toward a more contextualized discipline that is capable to absorb inspiration from the challenges. The trend poses requirements for an adaptive and flexible curriculum that interconnects to other disciplines.

### 6.3.2 Impact on the Profession

We will go through the foreseen impact of IT and global development on the profession of computing by the underlying principles of modeling, design, abstraction, and encounter.

Empirical modeling as an integrated thread of a software design and implementation process requires that the criteria for functionality and relevance are iterated throughout the process. This demands that IT professionals are able to replace given criteria lists by those created together with the users.

The participatory design projects in developing contexts change the landscape of IT consulting. Emerging areas cover areas like mobile banking, mobile learning, IT for health, and IT education that take place in the context. An IT professional changes from a person supporting available practices by conservative solutions toward a change maker. For example, mobile learning focuses on informal learning settings rather than conventional schooling.

Abstraction calls IT professionals to listen to the local voices at the bottom of the pyramid, instead of following the traditional top-down approach of various needs-lists. IT professionals turn into language experts and need to interpret the hidden demands with novel IT representations.

For encounter, IT professionals need to design technologies that promote their users’ human rights. These cover applications that support interfaith and intercultural dialogue, conflict resolution, and crisis management. Natural language processing and text tools form key techniques for these applications.

In general, the impact from IT in the context of global development has transformed the IT profession toward diverse job profiles. The competences required from IT professionals reflect increased sensitivity and ethical orientation.

### 6.4 Research Issues

The impact of IT for global development is still at its infancy. At the same time, IT can significantly gain from novel challenges and solutions within the developing contexts. Therefore, it is crucial that IT is developed within developing contexts where an interdisciplinary and multi-stakeholder collaboration renews and rejuvenates both technology and the development that it shall serve. This means that mere
evaluation of existing technology in various developing contexts is not enough; proactive and engaging
design and implementation are required.

In a way, the relation between technology and its users in developing contexts reminds that of digital
natives and digital immigrants in other contexts. There is a difference, though: instead of differentiating
between users from the viewpoint of their technology use history, we can measure the extent to which
technology itself has been exposed to or rooted in a particular user context. Technology can be either
context native, that is, born—designed and implemented—in a given context, or context immigrant, that
is, transferred from its origin to an alien place to serve. Using this terminology, most of the technolo-
gies used in developing contexts are context immigrants. The key research challenge is how to design
context native IT.

Like the impact on IT discipline and profession, the research challenges can be categorized in mod-
eling, design, abstraction, and encounter. Contrary to the issues of a practical nature, research looks,
though, for radical rather than incremental changes. In the area of language learning, an incremental
innovation allows students to learn Portuguese by their mobile phones during their holidays in Lisbon.
A radical innovation lets Mozambican secondary school students to earn airtime by chatting with stu-
dents that are becoming teachers in the Portuguese language. The role of radical innovation and cre-
ativity is crucial for the research-based creation of context native technology, which can also be called
indigenous technology.

Empirical modeling requires that the measures for the functionality and relevance of context native
technology be rethought. This results in out-of-box, fresh models as criteria for successful technologies. To
measure the success of context immigrant technologies, it is normally enough to adjust the weights of the
measured variables, like in the case of the mobile Portuguese learning environment those related to the
user behavior. A set of completely new variables needs to be introduced for evaluating the functionality
and relevance of the user scenarios of children teaching adults from the global south to the global north.
To measure the impact of context native technology requires qualitative indicators, and thus, methodolo-
gies based on the formative approach, like artifact contests and focus user group barometers.

Strength-based or resource-based design of context native technologies can be easily juxtaposed with
the needs-based design of context immigrant technologies. On one hand, the first is based on how to
use the existing resources in a creative way for matching the demands and expectations of the context;
both the resources and the demands need to be explored without prejudices or available lists. Needs-
based design, on the other hand, makes use of existing, straightforward lists of what is lacking. Research
on creative design milieu is an example of a task that results in context native technologies. This requires
designing a relevant IT curriculum for the particular developing context.

From the perspective of abstraction, research in non-linear information representations and non-
structured media is of a particular importance for context native technologies. In a developing context,
digital storytelling requires workbenches and studios for diverted stories. An interesting trade-off takes
place between abstraction and concretization: a multilayered representation supports both of them.

As of the research for enriching the encounters between users and their context native technologies,
a completely new research area in the intersection of human rights and information technologies is
envisaged. The key question is how IT can promote human rights by strengthening the ownership in
information in all its forms, including awareness.

An example of a possible research initiative is microstudies on context native IT. Microstudies is an
equivalent of microcredits in the domain of education. Microstudies enable a bottom-up study process
where each study—performed using a mobile device—gives a narrow competence in a user-requested
area. The acquired competences are stored as a digital portfolio. When a student has reached a critical
set of studies, she can apply for a certificate.

The rationale of microstudies is in the massive and fast-changing educational landscape of a develop-
ing context. Educational demands are hard to forecast, mobile technology is available, and people need
fast track for IT competences required at jobs. At the same time, the culture recognizes traditional,
formal aspects of education, like certificates.
The exemplified research project will be carried out following the strength-based design in a given context. Representation of the learning process is a digital, interactive, multimedia-based learning story, shared by social mobile media. A focus group representing potential users identifies indicators for the functionality and relevance of microstudies.

We can summarize the challenge of context native technology with the alternative concepts of garage computing, bottom-of-the-pyramid computing or life computing. The research toward context native technology takes place in developing technology pockets, that is, a periphery far away from centers of conventional technology.

6.5 Summary

The era of fast global development sets an attractive stage for IT to return to its passionate and radical roots. Developing context native technologies will require young talents that the developing contexts are full of. The emphasis is on designing fascinating solutions that only later might find their satisfied customers. The carrying force of the challenge is the belief that fresh ideas grow in peripheries.

Acknowledgments

In addition to the review of the key academic and industry literature, this chapter reflects the lessons learned by the author during his 2 years’ work as the Chief Technical Advisor in STIFIMO, a program of cooperation in Science, Technology, and Innovation between Finland and Mozambique in 2010–2014. The job offered a unique opportunity to discuss issues related to IT and global development with diverse stakeholders and also with people who did not have a clue of how science, technology, or innovation could benefit individual Mozambicans—mostly, the latter were expatriates. The opinions do not necessarily reflect those of the Ministry of Science and Technology of Mozambique or of the Finnish Ministry for Foreign Affairs. For the research part of the work, the author is indebted to the partial support from the Academy of Finland.

Further Information

http://ictlogy.net
http://www.ict4d.org.uk, for links http://www.gg.rhul.ac.uk/ict4d/links.html
e-Government for Development: http://www.egov4dev.org/

References


**Initiatives**

Bridges.org: [http://www.bridges.org](http://www.bridges.org)
Global e-Schools and Communities Initiative: [http://www.gesci.org](http://www.gesci.org)
Mobile Technology for Social Impact: [http://mobileactive.org](http://mobileactive.org)
One Laptop per Child: [http://one.laptop.org](http://one.laptop.org)
World e-Business Initiative: [http://www.worldebusiness.org](http://www.worldebusiness.org)

**Journals**


**Conferences**

*IFIP WG 9.4 Social Implications of Computers in Developing Countries*: [http://www.ifipwg94.org/conference-tracks](http://www.ifipwg94.org/conference-tracks)
*International Development Informatics Association*: [http://www.developmentinformatics.org](http://www.developmentinformatics.org)
*InfoDev*: [http://www.infodev.org](http://www.infodev.org)