Bringing Context to the Underserved: Rethinking Context-Aware Design to Bridge the Digital Divide

Summit Shrestha
summitshrestha@gatech.edu
College of Computing
Georgia Institute of Technology
Atlanta, GA, USA

Josiah Hester
josiah@gatech.edu
College of Computing
Georgia Institute of Technology
Atlanta, GA, USA

Ashutosh Dhekne
dhekne@gatech.edu
College of Computing
Georgia Institute of Technology
Atlanta, GA, USA

Umakishore Ramachandran
rama@gatech.edu
College of Computing
Georgia Institute of Technology
Atlanta, GA, USA

Alex Cabral
acabral30@gatech.edu
College of Computing
Georgia Institute of Technology
Atlanta, GA, USA

Abstract

The limited and highly variable resource dynamics of underserved communities, each with their own unique needs and values, underscore the need to integrate a context-aware approach when designing for these settings. Context-aware computing has long been a fundamental aspect of ubiquitous and pervasive systems, yet its application in Information and Communication Technologies for Development (ICT4D) remains limited. Existing context-aware approaches are predominantly designed for resource-rich environments and privileged communities, often failing to account for the unique constraints and dynamics of underserved populations. In this paper, we advocate for a paradigm shift in ICT system and service design to serve not only the privileged but also the underserved. Through the lens of two real-world case studies, we illustrate the contextual challenges faced by underserved communities and validate the design goals of our proposed framework by grounding them in real-world constraints, needs, and potential outcomes. Drawing upon existing literature and insights from the case studies, we first redefine context in ICT4D as a dynamic interplay of *situated location*, *community needs*, and *limited resources*, emphasizing a community-centered perspective. Building upon this definition, we conceptualize a more **community-context-aware** ICT4D design and propose enabling technologies for integrating *community-in-the-loop methodologies*, *efficient resource allocation mechanisms*, and *context-aware service resiliency and adaptability strategies* to enhance ICT services in resource-limited settings. By introducing a more context-aware approach to ICT4D, this paper aims to foster inclusivity, mitigate information inequity, and contribute to bridging the digital divide. Our work lays the foundation for future research on inclusive, resource-efficient, and community-driven context-aware ICT solutions.

COMPASS '25, July 22–25, 2025, Toronto, ON, Canada

ACM ISBN 979-8-4007-1484-9/2025/07
https://doi.org/10.1145/3715335.3735497

CCS Concepts

• **Human-centered computing** → *HCI theory, concepts and models*; *Collaborative and social computing theory, concepts and paradigms*.

Keywords

Information and Communication Technologies for Development (ICT4D), Context-aware Systems, Community Networks, Resource-scarce environments

ACM Reference Format:
Summit Shrestha, Josiah Hester, Ashutosh Dhekne, Umakishore Ramachandran, and Alex Cabral. 2025. Bringing Context to the Underserved: Rethinking Context-Aware Design to Bridge the Digital Divide. In *ACM SIGCAS/SIGCHI Conference on Computing and Sustainable Societies (COMPASS '25), July 22–25, 2025, Toronto, ON, Canada.* ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3715335.3735497

1 Introduction

In a rural classroom nestled in the foothills of Nepal, a solar-powered server hums quietly in the corner. It holds a world of knowledge: Khan Academy lessons, science simulations, storybooks, all accessible without the internet. For many children here, it is their only digital learning tool. But now that lesson plans have changed and new exams are scheduled, the content on the server is months out of date. Teachers cannot update it, the field team has not arrived, and cloudy weather has left the solar panels struggling to power the device. The students wait—learning from outdated lessons. This is not a failure of infrastructure. It is a failure of awareness, of systems designed without regard for their context. In this paper, we argue that to meaningfully serve underserved communities, Information and Communication Technologies for Development (ICT4D) must adopt a fundamentally different approach: one that places context, including location, community need, and resource constraints, at the center of system design.

Context-aware computing [63] has long been a fundamental aspect of ubiquitous and pervasive systems, enabling applications to dynamically adapt to their environment. Yet, its application in ICT4D remains underexplored. Existing systems are mainly designed for resource-rich environments, assuming continuous internet access, advanced infrastructure, and high computational power, conditions that do not hold in many resource-limited developing communities [33, 68]. As a result, Information and Communication Technology (ICT) solutions frequently fail to meet the needs of marginalized populations, reinforcing the digital divide and limiting access to education, healthcare, and economic opportunities.

The major difference in adopting a context-aware design for ICT4D from the existing context-aware systems lies in the "situatedness" of the receiving end users' side. Existing context-aware systems from the ubiquitous computing standpoint are designed with an individual user owning multiple different technological devices (resources) offering different user context. However, ICT4D solutions are designed with *community* in mind where the communities are generally underserved or resource-constrained. This brings differences in 1) what "context" matters the most from a resource-limited community's perspective and 2) how a more "context-aware" approach to ICT4D design can better serve such underserved communities with the resources they have in the best way possible.

A key challenge in ICT4D is the lack of community-context-awareness and context-adaptability in existing solutions, which do not consider dynamic socio-economic, infrastructural, and environmental contexts. For example, communities relying on offline educational repositories struggle with content updates and data collection due to limited connectivity (Section 4.1). Similarly, initiatives for community redevelopment in economically constrained regions face high infrastructure costs, making traditional cloud-based solutions impractical (Section 4.2). These challenges highlight the need for ICT systems that are contextually aware, meaning they are capable of adapting to local constraints, community priorities, and available technological resources.

In this paper, we argue that the current paradigm of context-aware system design must be rethought to effectively serve underserved populations. We redefine context in ICT4D as a dynamic interplay of three core factors: 1) *situated location*, 2) *community needs*, and 3) *limited resources*. This definition shifts the focus from individual-centric context awareness to a community-driven perspective, where ICT systems must be designed to integrate and adapt to the lived realities of marginalized populations. Building on this reconceptualization, we propose design strategies and technologies to improve the reliability and accessibility of ICT4D, thereby promoting information equity and contributing to improved quality of life in resource-limited underprivileged communities.

The realities surfaced through our case studies (Section 4), such as the failure of automated updates, intermittent server uptime, and community requests for prioritized content, do not merely illustrate the problem. They directly inform the design principles and architecture of our proposed framework. In particular, they reveal the need for systems that can dynamically adapt to resource variability, integrate local decision-making, and support context-aware prioritization. For example, the inability to perform content updates due to erratic power and weather patterns directly motivates the need for runtime resource-awareness and intermittent computing. Similarly, community requests for topic-specific updates and prioritization inform our community-in-the-loop mechanisms. These are not abstract design ideas; they are grounded responses to real, recurring needs. Drawing from these insights, we propose a framework that operationalizes *community-context-awareness* in ICT4D systems through design strategies for adaptability, inclusion, and resilience. We further detail this framework and the enabling technologies it draws upon, structured around the key contextual dimensions in Section 5.

To this end, the contributions of this paper are:

- Through a review of existing literature on context-aware computing and ICT4D, we first highlight the strengths of context-aware systems in enhancing user understanding and service delivery. We then examine their limitations in ICT4D use cases and propose a redefinition of context for ICT4D, shifting the focus from individual-based computing to community-driven context-awareness.
- Building on existing literature and real-world challenges in community-based applications identified through case studies (e.g., education, sustainable redevelopment), we propose a conceptual framework for community-context-aware ICT4D. This framework enables users to set service priorities and policies, automates resource optimization with self-adaptability, and enhances service resiliency and accessibility in resource-limited settings.

Section 2 outlines the background on context-aware systems and summarizes prior ICT4D research. Section 3 redefines context for resource-limited communities in ICT4D. Section 4 examines two real-world case studies and demonstrates how a context-aware approach can address their challenges. Section 5 conceptualizes design goals and enabling technologies for context-aware ICT4D. Section 6 explores design challenges and future research directions, and Section 7 concludes the paper.

2 Background and Related Works

In this section, we go over the background on context-aware systems, its applicability in the case of ICT4D scenarios, the notion of "context" in terms of ICT4D, current ICT4D scenarios and solutions, and the need for a context-aware system design for supporting ICT4D services.

2.1 Context-Aware Systems

2.1.1 Definition of "Context". Context-aware systems form the backbone of ubiquitous and pervasive computing, empowering applications to adjust dynamically based on environmental, social, and user-specific data. Schilit et al. [63] provided the foundational insights into designing systems that actively sense and react to their surroundings where they outlined "context" as comprising of three primary dimensions: "where you are (location), who you are with (nearby entities), and what resources are nearby (available resources)". They introduced the idea of context-aware applications, categorizing them into: proximate selection (nearby objects for easier interaction), automatic contextual reconfiguration (dynamic changes in system configurations), context-triggered actions (automatic execution of predefined actions based on context), and contextual information and commands (enabling users to query and interact with contextual data).

A widely accepted and commonly used definition of context is provided by Dey and Abowd [1] which states: "Context is any information that can be used to characterize the situation of an entity, where an entity is a person, place, or object relevant to the interaction between a user and an application." According to them, context-aware systems refer to applications or frameworks that utilize context to adapt their functionality dynamically and are able to: extract and interpret raw contextual data from sensors and other sources, reason about high-level abstractions of context, and adapt services and applications to better align with the user's needs and the environment. This definition provides a general framework applicable to a wide range of domains, from mobile and pervasive computing to the Internet of Things (IoT).

2.1.2 Dimensions of "Context". Over time, the dimensions of "context" has broadened to include a wide array of attributes. A recent survey [56] on context-aware systems categorized "context" into different dimensions, where context refers to diverse categories of information that characterize an entity's situation, encompassing physical, temporal, relational, and abstract aspects. Context can be primary (e.g., identity, location, or time) or secondary (e.g., social connections derived from identity) and is structured by the Five W's: Who, What, Where, When, and Why (e.g., a student writing an exam at 10 a.m. in a hall). It spans computing, user, physical, and time contexts, such as network connectivity, user profiles, environmental conditions, and temporal factors like the time of day. Context can be static (unchanging, like a person's name) or dynamic (changing, like location), as well as physical (e.g., temperature) or logical (e.g., GPS coordinates). Further categories include physical and cultural contexts, where physical aspects involve measurable features (e.g., location) and cultural aspects relate to user preferences (e.g., payment methods). Context may be sensed, static, profiled, or derived, based on data origin or evolution, and classified as global (e.g., room occupancy) or local (e.g., an object's location). It also encompasses individual, location, time, activity, and relational contexts, such as human characteristics, physical or virtual locations, specific tasks, or social relationships. Additionally, context may be active (current, like a patient's blood pressure) or past (historical, like medical records), and direct (sensor-derived) or indirect (inferred from other data). These categorizations reflect the complex and interdependent nature of context, enabling systems to adapt dynamically to users and environments.

2.1.3 Methods for Context-Awareness. Prior works have explored and defined various methodologies and ontologies for context-aware systems in regards to context acquisition, context representation, context processing architectures, context distribution, and context-based actuations [32, 55, 56]. Dey and Abowd [1, 20] proposed a framework with reusable components such as widgets, aggregators, and interpreters to simplify the acquisition and use of context in applications, and focused on separating sensing mechanisms from application logic. The literature describes context modeling as reusable ways to model context gathering sources or mechanisms such that they abstract different data sources from the system and the higher-level applications. In that regard, researchers have explored different ways in representing and storing environmentally gathered contexts such as using data structures as key-value pairs, object-oriented approach, tree and graph based representations, etc [9, 43]. Moreover, other works have explored the possibilities of using various architecture for context-aware systems, among which the middleware architecture is the most popular and commonly used due to its ability to abstract and easily integrate various context sources with less burden to the underlying system or application developers and ensure modularity and scalability [9, 13, 26, 58].

From our mobile phones possessing many different sensors (accelerometers, GPS, gyroscopes, camera, etc.) to various things (printers, refrigerators, mirrors, etc.) becoming smarter, the domain has found rich ways for gathering contextual information and make contextually informed decisions. Given this, today's context-aware system designs encompass multiple domains (involving wireless networking, IoT and sensing, system designs, and human-computer interaction) that possess the ability to take contextually relevant decisions or actions based on the information acquired and processed from various different sensing sources.

Existing works in context-aware computing systems are designed and work well for the **resource-rich** and **privileged** environments (with advanced sensing and almost infinite cloud resources) but fall short when considering **resource-constrained** or **underprivileged** environments. The major reason is the assumption of today's context-aware computing applications on the pervasiveness of sensing technologies, 24/7 access to connectivity and computing resources, and the differing importance of each "context" type when considering **resource-constrained** or **underprivileged** environments. This will be further discussed in the upcoming subsections.

2.2 ICT4D and Existing Solutions for the Underserved

Information and Communication Technology for Development (ICT4D) is a global initiative aimed at promoting equitable access to digital technologies to bridge the digital divide and foster economic and social development, particularly in marginalized and resource-constrained underserved developing communities [7]. Prior works in ICT4D for underserved communities with access to limited resources can be majorly classified into six different categories: 1) communication infrastructure and connectivity; 2) data and information dissemination; 3) adaptable systems and services; 4) people and community supported designs; 5) accessibility supported designs; and 6) contextual design frameworks. Below, we briefly describe existing advancement in each of these categories.

2.2.1 Communication Infrastructure and Connectivity. A key pillar of Information and Communication Technologies for Development (ICT4D) is connectivity and networking infrastructure. Numerous studies have explored extending connectivity to underserved areas with environmental or infrastructural constraints, including using free-space optics, LoRa networks, non-terrestrial networks, and dynamic spectrum access like TV White Spaces, which shows promise for rural regions [2, 34, 40, 53]. However, high costs for client-side devices make this option impractical for low-income populations, despite its current use in broadband services.

Satellite-based internet services, such as Starlink, have gained popularity but are not yet widely pervasive, resulting in inconsistent signal strength and connectivity, with limited studies on their reliability [75]. Additionally, the high cost of satellite networks raises concerns about their accessibility for low-income users [42].

Dynamic and ad-hoc wireless mesh networks have also been explored and show potential for providing connections to underserved communities [54, 78].

Innovative community-driven cellular networks offer a promising solution. These networks involve the community in building, deploying, managing, and maintaining the system. With training, communities can deploy networks using low-cost hardware (under $1,000) and open-source software[1], providing coverage for more than two miles [29]. Such networks have been embraced by underserved areas for their sustainability and affordability [37, 65].

2.2.2 Data and Information Dissemination. In the context of ICT4D, researchers have explored various mechanisms for collecting data from rural and resource-constrained areas with limited or no Internet access. Two common approaches for data collection and analysis are offline and hybrid data collection [12], which are both widely used in surveying and data collection tools. Offline data collection involves: 1) gathering data locally and 2) physically transferring it using flash drives between locations. In contrast, hybrid mechanisms involve: 1) collecting data offline, 2) storing it locally, and 3) syncing with servers when Internet access becomes available.

Information dissemination involves sending latest data to locally deployed applications with limited Internet access, making direct online data retrieval unfeasible. In such cases, similar offline and hybrid syncing strategies have been used, employing data ferrying [23] or syncing when Internet access is available [17, 21]. Moreover, delay/disruption tolerant networks (DTNs) provide a "store-and-forward" approach for transferring data across limited, unreliable, and intermittently connected nodes with high throughput and lower latency [77].

Although hybrid mechanisms provide a practical solution for addressing limited connectivity, current syncing methods typically rely on detecting Internet availability and then transmitting all data, without considering available network bandwidth or power capacity (especially in battery- or solar-powered systems) at that moment. This can lead to network congestion or power depletion, resulting in synchronization failures, data accumulation, and eventually making updates unfeasible under the current hybrid approach.

2.2.3 Adaptable Systems and Services. Although networking, data, and information aspects have received significant attention and contributions in the domain of ICT4D, the adaptability requirements of systems and services have not been widely explored. Existing works focus on building dedicated platforms and technologies to deliver essential services and content in underserved regions. Cloudlet-based micro-data centers [30] provide localized data storage and computational power in rural and remote areas, addressing infrastructural limitations and unreliable power and internet connectivity. Similarly, the BlendNet platform [47] leverages a "hub-and-spoke" model with satellite and edge technologies to facilitate offline and low-cost digital content distribution through intermediaries such as local shops. The BASS platform [64] considers the network bandwidth as context for optimizing service deployment in community wireless mesh networks, addressing bandwidth variability.

Existing service adaptability examples from cloud service platforms include websites [2] offering text-only versions of their pages and YouTube providing dynamic video streaming resolutions that adapt to available network bandwidth. Moreover, existing research efforts have also enabled support for service adaptability. Jun et al. [39] performed a large-scale evaluation of Google's Accelerated Mobile Pages (AMP) and their impact on web performance in constrained settings, showing it can drastically improve mobile page load speed – e.g. yielding 60% lower Speed Index (visual load time) than equivalent non-AMP pages given its strict page simplification, lazy-loading of content, caching via a CDN, and Google's search pre-rendering. However, this speed boost comes with hidden costs: AMP's aggressive prefetching of content imposes significant data overhead (averaging over 1.4MB of extra data per search) unbeknownst to users. This finding is critical for design in low-resource contexts, where data is often limited or expensive – an apparently "faster" experience may actually consume more bandwidth, undermining its benefit for underserved users. This clearly portrays technology designs today being silently intended towards the resource-rich environment while abruptly failing for the resource-scarce ones. Such trade-offs highlight the need for frameworks to balance performance gains with resource costs, motivating our work's context-aware approach.

Another recent example is MAML (Mobile Application Markup Language), a web framework designed for the Global South that pre-simplifies web content for low-end phones [52]. Modern websites are often bloated with complex scripts and heavy media, which makes them nearly unusable over slow networks or on cheap smartphones. MAML flattens webpage structure and strip non-essential code, yielding dramatic performance gains (loading pages tens of seconds faster than even Google's AMP in poor network conditions). This optimization acknowledges the resource constraints in underserved settings and adapts mainstream services accordingly. Similarly, Newman et al. [50] developed ScaleUp system that improved above-the-fold load times by 19% (median) simply by zooming-in on pages under slow networks (hiding some content below the fold), and a crowdsourced trial with 1,000 users confirmed corresponding gains in perceived user experience. Such results illustrate that slow or lossy connections demand tailored solutions – a one-size-fits-all web design fails when latency, loss, or bandwidth differ drastically. Furthermore, the Siskin chrome extension [71] leverages the ubiquity and capabilities of web browsers for distributed content caching and sharing within a local network requiring minimal configuration and no additional hardware, reducing overall costs. These small feature integrations on the application services side help ensure functionality even with limited resources (e.g., network bandwidth), although they often compromise on quality or performance, have limited adaptability capabilities, and are not designed with community-in-mind.

2.2.4 People and Community Supported Designs. Beyond addressing technological gaps, ICT4D systems must align with the sociocultural values and traditions of target communities, fostering acceptance and long-term sustainability [28]. Because these services are intended for individuals or groups within these communities,

[1] https://github.com/uw-ictd/colte

[2] https://lite.cnn.com/, https://text.npr.org/

understanding their perspectives and socio-cultural context is crucial for creating systems that are not only usable but also culturally acceptable and responsive. In some cases, even when technology is readily available, people may be hesitant to fully adopt it or utilize it as designed for because it is misaligned with their values [38, 48]. Communities often share common beliefs and values. Thus, designing systems and services that account for each community's unique context, considering factors such as location, people, and social values, can significantly enhance their usability and impact.

Few existing works include the implementation of socio-culturally sensitive systems and services. For example, Wiki Katat and Taewaloni integrate Indigenous worldviews into digital platforms, and Radio Tosepan and FiDO's caching systems prioritize local content relevance and community engagement [27, 48, 80]. Similarly, Kasadaka enables a self-contained, voice-based platform that hosts and develops spoken information services tailored to non-literate populations [8]. Other works emphasize community-driven technological decisions such as the need of a community-driven network congestion management platform - a participatory tool for network measurement and coordination, enabling communities to challenge official broadband data and advocate improved infrastructure [38].

Another thread of ICT4D deals with ethical design – ensuring technologies respect local practices and rights. Ahmed et al.'s study of mobile phone sharing in Bangladesh reveals how Western assumptions of personal device use break down in multi-user contexts, leading to serious privacy challenges [4]. People share phones out of economic necessity and cultural habit, but this compromises personal data privacy and introduces power imbalances (e.g. men inspecting women's phones). The finding is an ethical wake-up call: ICT4D systems must account for communal use and consent, rather than assuming individual ownership and privacy norms as found in rich-resource environments. As a solution, they proposed Nirapod [5] which implements a "tiered privacy" model that lets users create separate modes or accounts on a shared phone – one for personal data and another for shared use. It shows how incorporating local context (like shared usage patterns) into design is crucial in ICT4D scenarios to protect vulnerable users.

Researchers have also highlighted misalignments that arise when designers or decision-makers overlook marginalized communities' contexts. Saha et al. [59] reported that top-down development planning often leads to tensions and disconnects with low-income communities, and they urge more inventive approaches to "bridge the gap" between high-level decision-makers and local voices. Moreover, marginalized groups exercise agency in how they engage with technology. They may even practice technological refusal, actively resisting or repurposing technologies that do not fit their values or needs – a form of contesting sociotechnical systems that underscores the need for more contextually appropriate, community-centered solutions. Innovative methods have emerged to empower community voices. In one study, disadvantaged women farmers in rural Bangladesh participated in participatory video production to articulate their needs and lived experiences [60], allowing them to "speak" in a medium comfortable to them, surfacing issues outsiders had overlooked. The resulting insights informed development programs in ways that purely top-down data could not. Such examples show that when communities can convey context in their own terms, even to the point of refusing ill-suited technologies - the interventions can be better tailored. Our approach builds on these lessons by embedding community feedback loops in the design.

Moreover, prior work emphasizes that effective ICT4D must leverage human infrastructure – the networks of people, social processes, and local knowledge that underpin technological systems in underserved settings. Sambasivan and Smyth [62] first introduced human infrastructure as an analytical lens to understand how shared social norms, information flows, and community practices enable ICT access in low-income communities. For example, Dell et al. [19] revealed how ICT deployments in NGOs and development organizations face myriad infrastructural, social, cultural, and linguistic challenges in practice, underscoring that technology alone cannot succeed without aligning to local human support structures. Real-world cases like Cuba's ElPaquete offline Internet make this clear: instead of purely technical networks, ElPaquete is "populated and sustained by a human infrastructure", wherein community distributors, local couriers, and social agreements deliver up-to-date digital content through an entirely community-run system [22]. Sultana et al. [72] highlighted the socio-cultural barriers (e.g. patriarchal norms) facing rural Bangladeshi women and the need to design around existing community support networks. These studies show that successful interventions depend on community actors (family, peers, local experts) who fill gaps left by sparse technical infrastructure. These works lay a foundation for viewing ICT4D systems not just as technical artifacts but as socio-technical ecosystems. Our approach builds on this foundation. By designing context-aware ICT4D solutions that explicitly integrate community actors ("community-in-the-loop") and local resource dynamics, we extend the human infrastructure concept to more adaptive, system-level designs that treat community context as a first-class component.

2.2.5 Accessibility Supported Designs. Another critical yet under-addressed facet of ICT4D is accessibility for people with disabilities in low-resource communities. Approximately 80–90% of people with disabilities worldwide live in developing regions [76], where they face acute shortages of assistive technologies and accessible services. For example, only an estimated 5–15% of mobility-impaired individuals in low-income countries have access to an appropriate wheelchair [10], and the vast majority lack reliable access to digital assistive tools (e.g. screen readers in local languages) or even basic internet connectivity.

Numerous solutions have been explored – albeit in resource-rich settings – to assist people with disabilities. For instance, VizWiz [11] introduced a smartphone-based service for blind users to receive nearly real-time answers to visual questions by sending photos to remote crowd workers. This approach leverages ubiquitous connectivity and cloud resources to dynamically provide information on demand. Similarly, DarkReader [83] is a context-aware screen reader enhancement that reacts to device power conditions: by truly turning off the display when a blind user does not need it, DarkReader preserved 24–52% of smartphone battery life with no loss of usability. These innovations illustrate the power of context-aware adaptation (e.g., human-in-the-loop assistance, adaptive power management) in improving accessibility. However, they presume consistent internet access, modern devices, and ample power – conditions often unavailable in underserved environments.

Furthermore, disability scholarship emphasizes interdependence, where technology use is entwined with social support networks and community infrastructures, especially in resource-constrained environments. A study of wheelchair users in an informal settlement found mobile phones became an accessibility bridge only through shared use and help from others, underscoring the "human infrastructure" that makes technology workable in low-resource contexts [10]. By accounting for collaborative use patterns, shared devices, and community information flows, the authors argue technologies can better amplify their impact in low-resource settings. Designing for such contexts thus requires moving beyond an individual-centric lens to a more social and community-oriented one.

2.2.6 Contextual Design Frameworks. Several recent works have proposed contextual design frameworks tailored to specific underserved groups or technologies. For instance, Cruz et al. [16] co-designed wearable devices with low-income immigrant communities, distilling a research agenda called "Equityware," which emphasizes building wearables around those communities' needs, comfort, and resource constraints. This framework targets unique barriers (e.g. safety, cost) faced by marginalized users of wearables and demonstrates how contextualized co-design can make technology more accessible. Similarly, Greenlee et al. [25] worked with Native American partners to develop an environmental sensing framework for remote conservation efforts. They confirmed a five-stage "microclimate sensor lifecycle" and highlighted design priorities like field usability (the "cost of experience") and community engagement in sensor deployments. Such domain-specific frameworks show how tailoring HCI principles to a particular population or technology – from undocumented Latine residents and wearable health trackers to Indigenous land stewards and IoT sensor networks – can yield deeply relevant design insights. Moreover, other scholars have focused on population-specific challenges: Almohamed et al. [6] present a conceptual framework for refugee resettlement technologies that foregrounds cultural background, displacement-related stressors, and social resources in the host community. Each of these frameworks offers valuable guidance within its niche – be it wearables for marginalized urban users, community-run environmental IoT systems, or digital services for refugees – and underscores the importance of context in design for underserved and marginalized groups.

Despite the value of these targeted frameworks, they remain inherently narrow in scope. Each is largely confined to a single community or technology, addressing one slice of the broader digital divide. In fact, some researchers argue for culturally specific design practices in lieu of generalized approaches [27]. Such highly localized methods ensure cultural resonance but are hard to generalize beyond their original context. This is where our work diverges and contributes novelty. We aim to complement these siloed models with a generalized, adaptable framework that can be applied across diverse underserved contexts. In contrast to one-size-fits-one solutions, our framework abstracts common principles drawn from multiple communities and technologies to guide context-aware design in low-resource settings. In doing so, we answer calls in the literature for a more cross-cutting approach to equitable technology design, bridging the gap between bespoke solutions and a unified theory of context in ICT4D. By synthesizing lessons across domains, we strive to provide practitioners a flexible blueprint that retains local relevance (through community-in-the-loop adaptation) while offering broadly applicable strategies. This positions our work as a general-purpose context-aware ICT4D framework (detailed in Section 5) that complements, rather than replaces, existing specialized frameworks – unifying their insights and extending them to new domains and populations.

2.3 A Review of Context from ICT4D Perspectives

Numerous studies in ICT4D, discussed earlier, emphasize the critical role of understanding and incorporating the "context" of the environment or community for which ICT solutions are designed and implemented. This context can vary significantly across different communities due to differences in geographical, social, political, organizational, cultural, economic, religious, and infrastructural factors. For instance, one community may have more limited access to computing devices and internet bandwidth than another, while a community with a predominantly younger population may prioritize educational services over a community with an older demographic. These contextual differences shape the accessibility, adoption, and effectiveness of ICT solutions in different regions.

A recent survey [57] examines various perspectives on context in ICT4D, categorizing them into "psycho-social contexts"—such as social, cultural, demographic, cognitive, emotional, and linguistic factors—and "structural contexts", which include systemic, politico-legal, environmental, economic, financial, and historical aspects. These contexts range from cognitive (e.g., learning styles, literacy levels) to politico-legal (e.g., institutional frameworks, governance, policies, regulations) to emotional (e.g., frustration, empowerment), among others, as summarized in Table 1.

The breadth of context highlights the inherent vagueness in defining "context" for ICT4D and underscores the need for multidisciplinary expertise to address its complexities, further leading to the emergence of fields such as HCI4D [14] and UbiComp4D [61].

To this end, there is a need to refine the definition of "context" when designing ICT4D systems due to two major reasons: 1) current context-aware systems (Section 2.1) do not consider community and socio-economic factors and assume resource-rich environments where individuals generally own many devices; and 2) existing taxonomies of context dimensions for ICT4D, such as that shown in Table 1, are often overly broad, lack the measurability and structure necessary for effectively representing or modeling each unique community, and make it impossible to integrate context-awareness in their current form. Thus, in designing a context-aware system for ICT4D scenarios–one that can be measured, modeled, and integrated for context-awareness–requires refining and reorganizing the existing taxonomy of context for ICT4D so that contexts are not only defined qualitatively, but quantitatively represent each unique community and are addressable at the system level. This calls for a redefinition of the existing taxonomy of context for ICT4D scenario, which is further detailed in Section 3.

This work draws inspiration from existing context-aware computing and system designs in ubiquitous computing, while acknowledging their limitations in addressing the unique contextual dimensions of ICT4D. It builds on an understanding of these contextual

Table 1: Summary of Context Dimensions for ICT4D [57]

Dimension	Category	Key Aspects
Psycho-Social	Social	Community participation, marginalized groups' inclusion
	Cultural	Cultural norms, values, and practices affecting ICT adoption
	Demographic	Variability in gender, age, and education across user groups
	Cognitive	User learning styles, literacy levels
	Emotional	Emotional impact of ICT use, such as frustration or empowerment
	Linguistic	Language barriers in ICT use
Structural	Systemic	Infrastructure, technology readiness, geographical challenges
	Politico-Legal	Institutional frameworks, governance, policies, and regulations
	Environmental	Physical environment, exposure to natural risks
	Economic & Financial	Affordability and access to financial services
	Historical	Legacy systems and their impact on technology adoption

perspectives from an ICT4D standpoint and proposes potential design goals and enabling technologies to better align with and serve the needs of resource-constrained underprivileged communities.

3 Redefining "Context" for ICT4D

Given the limitations of the "context" definition in both existing context-aware computing systems and ICT4D perspectives, this section refines and redefines context specifically from the standpoint of context-aware ICT4D design.

As highlighted in earlier literature, ICT4D solutions adopt a *situatedness* approach in designing and deploying technological solutions that align closely with the conditions of the executing environment. Based on prior ICT4D works discussed in Section 2.2, three major questions arise when designing ICT4D solutions: 1) Where is the solution being designed to operate? 2) What do the people in these locations genuinely need? and 3) What resources are actually available to them? The first question addresses the locational attributes, the second focuses on the users' information service needs, and the third examines the technological resources accessible to the community. The answers to these questions vary across both space (different communities) and time (different periods within the same community), leading to dynamic and context-specific ICT4D environments. Therefore, solutions designed for ICT4D scenarios must: 1) ensure they are relevant to the end users' needs, and 2) ensure they function effectively within the constraints of the local environment and available resources.

To this end, we redefine context for ICT4D design as the dynamic factors that influence the availability, accuracy, and quality of ICT services for a specific community at any given time. A community-centric context plays a critical role in determining the most appropriate solutions to address its needs at that moment. We broadly categorize a community's context into three primary dimensions: geographical attributes, technological resources, and unique user-to-application requirements, which encompass a range of social, economic, and technological factors. Based on these dimensions, the existing literature on ICT4D can be divided into three major "contexts," within which additional subcontexts can be further categorized. Table 2 provides a summary of the various contextual dimensions, along with their corresponding subcontexts and relevant attributes/ properties.

3.1 Situated Location as Context

This category of context encompasses the location-awareness of specific communities or individual users within a community. Such locational context can be further subdivided into geographical and environmental contexts. Geographical context refers to place-based properties, including terrain type, altitude (e.g., above sea level), and the density of buildings or forests, providing an initial framework for assessing the community and determining suitable technological solutions. For instance, a region situated at a higher altitude with dense forests may require a different connectivity mechanism than a more open and flat area.

Environmental context involves the dynamic factors of a location, such as changing climatic and weather conditions, which can indirectly affect the availability of technological resources. For example, on a rainy day with limited sunlight, areas relying on solar-powered solutions may experience intermittent service disruptions. Furthermore, the movement of entities (technological resources) and actors (community users) over time results in shifts in their specific locations (in terms of current longitude and latitude), introducing the mobility context.

3.2 Community Needs as Context

One of the most critical aspects to consider for the success of any ICT4D solution is the ability of technological systems to understand and align with the needs and economic conditions of the target community. Differences in service needs across communities are influenced by various factors, which can broadly be categorized into social and cultural contexts.

Table 2: Proposed "Context" Definition for ICT4D

Context	Subcontext	Attributes/ Properties
Situated Location	Geographical	terrain type, altitude
	Environmental	climate, weather, temperature
	Mobility	longitude, latitude (user and resource)
Community Needs	Social	demographics, population density
	Cultural	culture, religion, language, beliefs
	Economical	income, expenditure limit on ICT
Limited Resources	Compute	CPU, memory, disk, storage
	Network	internet and intranet bandwidth, signal-to-noise ratio (SNR)
	Power	battery level, charging mode, device uptime, temperature

Social contexts encompass factors such as a community's demographics or population density, which can significantly impact the type and level of granularity required in the services provided. For instance, a community with a predominantly younger population may prioritize educational services over entertainment services, with a focus on specific educational content tailored to their needs.

Cultural contexts refer to the unique mental models of each community, which are closely tied to their culture, religion, language, and belief systems. These factors influence the acceptability and perceived value of information services. For example, a community might prefer services in their local language or collectively decide to block certain services that conflict with their cultural beliefs. Similarly, economic contexts play a significant role in shaping community needs, such as preferring services that adhere to specific constraints on power or network consumption.

By incorporating these community-driven needs as contextual factors, ICT systems and services empower users to actively participate in and influence the design and functionality of the underlying technological solutions, ensuring their relevance and sustainability.

3.3 Limited Resource as Context

This contextual perspective emphasizes the technological dimension while integrating socio-economic and locational factors. A critical consideration in ICT4D scenarios is the limitation of technological resources available to a community, which often exhibit dynamic behaviors. Basic technological resources accessible to underserved communities, such as computational resources (e.g., CPU, memory), network resources (e.g., internet connectivity, bandwidth), and energy resources (e.g., battery capacity, device uptime), are intricately linked to their economic and environmental conditions.

For instance, internet availability in such communities largely depends on population density and the number of potential subscribers, as internet service providers (ISPs) are incentivized to establish base stations only when a sufficient subscriber base ensures a high return on investment. However, underserved communities, often characterized by low-income levels and sparse populations, are unable to afford high-speed internet services, resulting in limited internet bandwidth and restricted access to internet-based resources and services.

Additionally, the usage and availability of these resources are dynamic and continuously evolving, necessitating adaptive approaches to resource and service provisioning. This resource variability is influenced by environmental and mobility contexts discussed earlier. For example, extreme weather conditions can degrade available internet bandwidth and reduce the uptime of solar-powered devices, disrupting access to time-sensitive services. Furthermore, mobility and user access patterns across different locations introduce variations in resource usage, impacting the availability of shared resources within a community.

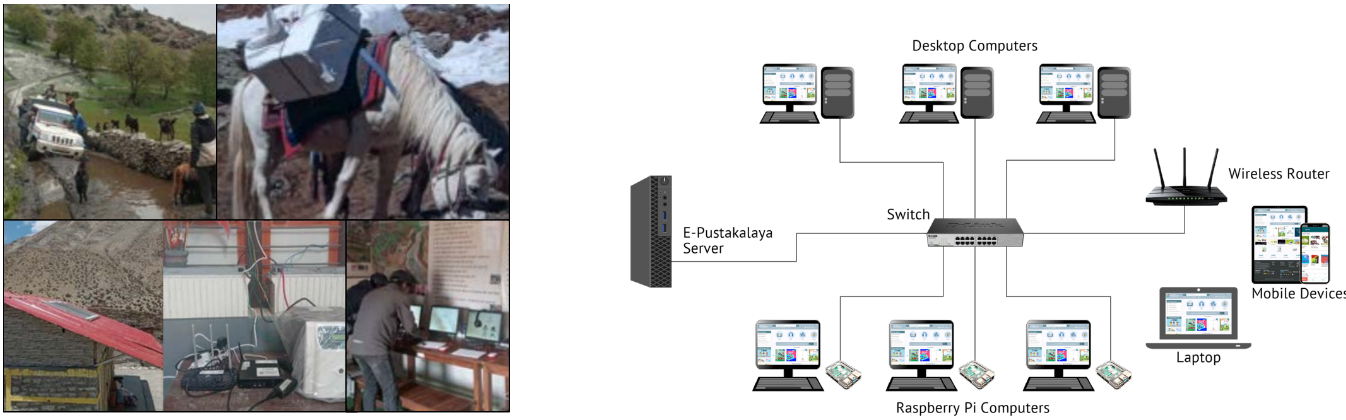
4 Exemplar Use Cases: Real-World Case Studies

In this section, we present two real-world scenarios to better illustrate their existing problems and requirements for context-aware ICT4D design.

4.1 Case I: OER Content Updates and Data Collection in the Global South

4.1.1 Background. Open Educational Repositories (OERs) are collections of educational resources, such as Khan Academy videos, PheT Simulations, and YouTube videos, bundled into a single server and deployed in environments such as rural schools and community centers with limited or no internet access [41, 69] (represented by the "limited resources" context in Table 2). Examples of such deployments include Internet-in-a-box, Kolibri, and Kiwix [35, 70]. These efforts provide critical access to information for underserved communities, yet significant challenges remain in updating content and understanding usage patterns in resource-limited settings.

In many developing countries, OERs are used to support education in rural areas. A real-world example is OLE Nepal, a non-profit organization aiming to bridge the educational divide through technology. OLE Nepal [49] has developed open-source interactive educational software, accessible via web browsers on mobile



(a) OLE Nepal's OER Server Deployment: Servers reach rural schools through difficult terrains where they are deployed locally, powered by solar panels

(b) OLE Nepal's Current OER Server Deployment Architecture: Uses a local network and low-powered devices but has no or limited connectivity to the Internet

Figure 1: Case for OLE's OER Service Deployment [49]

phones or desktop devices. They also offer a digital library containing documents, novels, audiobooks, video lessons, and other learning materials, including content from Khan Academy, Wikipedia, OpenStreetMap, and PheT Simulations[3]. These resources have been deployed in more than 1,500 rural schools and community libraries across Nepal, impacting over 400,000 students [49]. Figure 1a illustrates the deployment of an OER server powered by solar panels in rural schools across Nepal's challenging terrains, while Figure 1b depicts the current server deployment architecture implemented by OLE Nepal. Such solar powered servers are directly tied to their "situated location" contexts, such as weather conditions, causing dynamic server uptime behaviors and further impacting the "limited resources" contexts.

4.1.2 Existing Problem. Despite the success of making educational resources accessible, updating content presents significant challenges. Offline servers either need to be physically transported back to the organization's headquarters, or field teams must visit each school to perform updates. In the first scenario, servers must be carried over difficult terrains to the capital, where staff manually push updates—a process that can take weeks, during which students may lose access to vital learning content. In the second scenario, the cost and logistics of sending field teams to each school make regular updates impractical. As a result, students often rely on outdated content, which may become inaccurate or irrelevant over time, calling into question the validity of the information provided.

Additionally, OLE Nepal lacks a way to gather data on how these educational tools are used, limiting their ability to prioritize updates based on unique community contexts (represented by "community needs" in Table 2). The absence of data collection mechanisms, combined with unreliable internet and intermittent power (often reliant on solar energy), complicates the timing and execution of content updates. Even with good internet access, servers may not be powered on consistently, exacerbating the problem.

[3]https://pustakalaya.org/en/

4.1.3 Potential Solution: Tackling this challenge could be achieved by a well-designed "resource-aware" and "priority-aware" content update and content caching mechanism. First, the system needs to be aware of its available resources at runtime, which includes available CPU, memory, inter-network and intra-network bandwidth, and power. As these resources, in addition to being physically limited, also change dynamically at runtime due to internal (colocated tasks running on the same device from different user or user groups affecting the available CPU, memory, and storage capacity at runtime) and external (environmental such as weather, temperature, occlusions, etc. impacting the network signal strength and charging abilities for solar-powered devices) factors, acquiring correct states of technological resources at each resource-specific granularity is necessary. Second, the community end users (in this case, the students or teachers) should be provided with easy-to-use interfaces for prioritizing content updates, for example to prioritize content based on the topics included in upcoming lectures. If priorities are not provided, the underlying system should intelligently prioritize the contents based on historical and prediction-based approaches. The first solution takes into account the "situated location" and "limited resources" contexts, while the second solution integrates the "community needs" contexts, which altogether forms a context-aware approach to designing content update solutions.

Fulfilling all these requirements independently for each application (for example, digital library system, interactive learning platform, etc.) requires a system that can abstract relevant community contexts from lower-layered hardware infrastructures (devices, sensors) for *resources* and higher-layered application services for *priorities*. Moreover, it also needs to work well with the intermittent behaviors caused by the lack of connectivity or power due to *locational* contexts such as environmental factors. Considering these three contexts (situated location, limited resources, and community needs as proposed in Table 2), the system would automatically decide on when and what task to schedule for updates in the optimal way possible with best-effort and with proper chunking, checkpoint, and store-and-forward strategies, further detailed in Section 5.3.

4.2 Case II: Technology Integration for Sustainable Community Redevelopment in the Global North

4.2.1 Background. Although developed countries situated in the Global North generally have broader Internet connectivity, connectivity can vary greatly in different locations (due to "situated location" and "community economy" context), with rural, racial minority, and low-income communities often facing poor or no connectivity [51, 84]. This can be prohibitive for communities looking to invest in development projects that rely on broadband access, as the infrastructure is costly.

One such project is the "Henderson School Alumni Association Trust (HSAAT)". HSAAT aims to revitalize the old abandoned Henderson High and Elementary School property in Jackson, Georgia in the United States, and turn it into a tech-driven center for community and workforce development[4]. The formation of HSAAT and the alumni's ongoing commitment to restoring the space is partially driven by the historical significance of the site. Established in the 1950s, Henderson School initially served as an all-Black public school before later opening doors for the general public of all colors, symbolizing a significant chapter in the community's history. Unfortunately, the school closed in 2010, and the building and surrounding spaces were abandoned, falling out of government maintenance. Fig. 2a portrays the current condition of the abandoned space without proper infrastructure support, no Internet connectivity, and manually controlled limited power supply for cost savings. This highlights the intertwined relationship between "community context" (such as their economies and needs), "situated location", and "limited resources".

4.2.2 Existing Problem. One challenge local community groups like HSAAT often face in the early stages of such projects, particularly when seeking grants, is overcoming economic barriers (characterized by the "community" context). These barriers prevent them from establishing advanced technological infrastructure, such as servers and good Internet connectivity (resulting in limited availability of resources or "limited resources" context), due to the high costs involved at the early stages while funding is still being secured. Although it would be beneficial to initiate community engagement and development activities early on to gain support and feedback, the initial investment required for this is often prohibitive. While groups may cover some early expenses through their own collections, this approach is typically unsustainable for the long term, especially as the project continues to pursue grant funding.

4.2.3 Potential Solution. In such scenarios, a low-cost, locally deployed OER servers (considering the community's economies) with services prioritized (Fig. 2b) according to the *community's needs* could be a game changer. This approach eliminates the need for costly servers and high Internet expenses, making it a far more sustainable solution for revitalization projects. Examining the costs involved, only a small, low-powered computing device, costing between $100 to $500 depending on the community's needs, would be required, avoiding large initial infrastructure investments and eliminating any Internet subscription fees. Additionally, power costs would remain minimal due to the use of low-powered system. Moreover, the community users can be provided with an easy-to-use interface to input cap for the power usage/ power costs as well as prioritize times for the system availability. The system based on such *community needs* and current state of *resources* (compute and power usage) can integrate low-powered solar-operated sensors to activate the servers or WiFi access points only when necessary, much like automated lighting systems in buildings. The idea is to use the limited resources available to a community in the most efficient and cost-effective way possible by better understanding the community's context and by putting community users in the driver's seat for control. Integrating a context-aware approach, which considers the community's socio-economic and technology requirements through low-powered sensing and locally deployed low-powered computing infrastructure, offers sustainable support for the project's long-term goals.

[4] https://www.hendersonrepurpose.com/

5 Context-aware ICT4D Design: Enabling Technologies and Challenges

Motivated by an in-depth review of existing literature and insights from real-world case studies (Section 4), we identify an urgent need for a more "community-aware" approach to existing context-aware computing systems in the design of ICT4D systems and services. These case studies directly exposed the limitations of existing ICT4D systems in dealing with dynamic community contexts and constrained resources. As such, they not only motivate but also shape the formulation of our design principles by exemplifying how context-aware features—such as content prioritization, runtime adaptability, and community-in-the-loop control—can be practically implemented and beneficial. Aligning with our definition of context (Section 3), such an approach should:

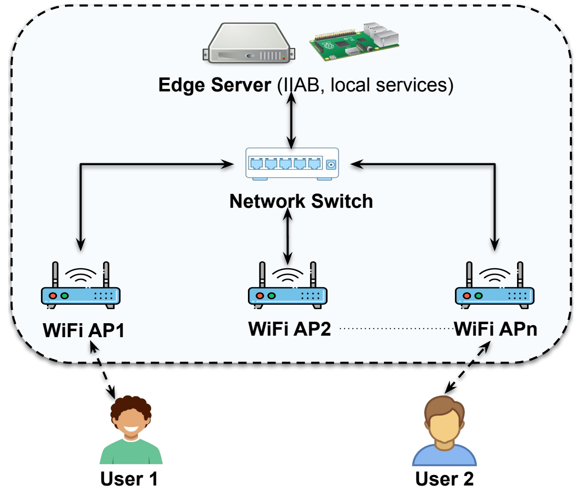
- **Comprehensively understand the community's needs and constraints**: Understand the community's internal needs and preferences for information services, the technological resources available for running these services, and both community-specific and external factors that influence resource and service availability.
- **Adopt a "community-in-the-loop" methodology**: Actively involve the community in mapping their needs to the available resources in the most efficient way possible.
- **Learn and adapt to dynamic contexts**: Develop systems that can learn from and self-adapt to the evolving "contexts" both within and across communities.
- **Enhance access and reliability**: Improve access to ICT services and increase their reliability.

In this section, we outline the key principles and enabling technologies of context-aware ICT4D design, consolidating context inclusivity, resource efficiency, system resiliency, service adaptability, and extensibility into a unified framework that prioritizes community participation and empowerment.

5.1 Design for Context Inclusivity

We posit that context-aware designs for ICT4D require the contexts being considered to be more inclusive and community-centric. This can be enabled by using both manual and automated methods, where manual methods use a community-in-the-loop approach to

(a) Current Condition of the Abandoned School Space: The building lacks proper infrastructure, has no Internet connectivity, and has limited power supply to cut down costs



(b) Current Local Edge Deployment Architecture: Uses low-powered Raspberry Pi and Wi-Fi access points

Figure 2: Case for HSAAT's Low-powered OER Service Deployment

acquire less frequently changing community policies and priorities while automated methods use plug-and-play interfaces for acquiring more frequently changing locational and resource contexts.

5.1.1 Community-in-the-loop Policies and Priorities. To integrate the context of community needs and preferences, the development of easy-to-use human-in-the-loop interfaces for community users is an essential component of context-aware ICT4D. This component directly aligns with the human-computer interaction (HCI) aspects of ICT4D, where interfaces must be localized to empower community users to drive services based on their specific needs. Factors such as adapting interfaces to local languages and incorporating intuitive visual representations enhance accessibility, making them user-friendly for diverse communities, including low-literate and differently-abled users. Although HCI research can further contribute through participatory and co-design methods to make these interfaces more community-oriented, this component focuses on the functionalities that such interfaces should enable. These functionalities can be broadly categorized into two key areas: user-driven service priorities and rule-based community policies. As seen in Case I (Section 4.1), the lack of user-driven content prioritization for OER systems led to inefficient update cycles and outdated content. This motivates the need for community-prioritized service configurations as part of our proposed design.

- **User-Driven Service Priorities:** In resource-limited environments where compute, memory, storage, and network resources are constrained, allocation to individual community users must be equitable while allowing for personalized service priorities at a fine-grained level. For instance, an individual user could assign different priority levels to specific webpages within the same website or web application; teachers can assign a higher priority to contents based on the topics included in upcoming lectures (useful for cases as seen in Section 4.1). Similarly, priorities can also be established at a group level, where nearby users (identified through locational context or mutual agreement) form a group and collectively define service priorities. This enables efficient resource utilization by pooling resources for shared tasks. For example, a group of users might collectively request a movie download and enjoy it together, sharing their allocated resources to reduce individual resource consumption while meeting their shared needs.
- **Rule-Based Community Policies:** Community-wide policies can be implemented using simple rule-based conditional logic, such as *if-then-else*, integrated into easy-to-use drag-and-drop interfaces. These policies, established through community consensus, allow for shared resource management based on collective viewpoints and beliefs. Such policies may be coarse-grained, addressing resource allocation at the community or group level. For example, a policy might prioritize traffic for certain services, such as educational (Section 4.1) or health-related applications, during periods of limited bandwidth. In such cases, lower-priority traffic, such as individual YouTube streams, could be temporarily deprioritized to ensure critical services are maintained. This approach enables community-level traffic shaping and congestion control mechanisms, ensuring shared resources are utilized efficiently and in alignment with community priorities. This concept aligns with the findings from [38].

Thus, these policies and priorities can be divided into three hierarchical levels: user, group, and community, where "*community policies > group priorities > user priorities*". Moreover, within groups, groups with more members could be afforded higher priority than smaller groups (*"group_of_5_users > group_of_3_users"*).

5.1.2 Plug-and-Play Context Adapters. While the less frequently changing community needs context can be acquired through a manual community-in-the-loop approach, there is also a critical need to capture *frequently* changing community contexts. For example, real-time environmental conditions and available resources provide essential context and necessitate a more automated approach to context acquisition. Technologies that help to acquire these contextual elements include low-cost sensing mechanisms (e.g., GPS, temperature sensors, light sensors), low-bandwidth web services (e.g., weather APIs for specific longitude and latitude coordinates), low-level hardware utilization APIs, or embedded sensing mechanisms in the hardware.

There is no strict requirement to integrate specific context sources, as the choice of context acquisition method may vary across communities. This variability highlights the need for managing multiple adapters for different sources. Consequently, an easily integrable plug-and-play interface is essential for facilitating context gathering from various sources. Such plug-and-play context adapters can be useful for integrating low-cost sensing mechanisms in the scenarios discussed in Section 4.2.

- **Containerization [36]:** Context source adapters can be packaged into containers along with their respective codebases, dependencies, expected data formats, and data parsers.
- **Digital Twins [67]:** A digital twin creates a virtual representation of a physical entity, enabling continuous context monitoring. For instance, in the case discussed in Section 4.2, a twin can be created for traditional power control mechanisms using basic low-cost sensors, without requiring advanced, expensive replacements with newer controllers, thus supporting the community's socio-economic context.

5.2 Design for Resource Efficiency

Given the resource limitations in underprivileged communities, one important design goal for context-aware ICT4D is the mapping of resources to tasks with varying priorities at runtime such that potentially limited resources are used in the most efficient and resourceful manner. Some technologies that would enable meeting such design goals are briefly described below.

5.2.1 Dynamic Resource Allocation across the Edge-Cloud Continuum. Almost all information services we use today are powered by the cloud, creating the need for a stable internet connection to access them. Additionally, each service has its own Quality of Service (QoS) requirements, such as internet bandwidth and latency, to make the service usable. For instance, a typical video conferencing call requires high-speed internet connectivity (latency under 150 ms) and significant bandwidth to ensure smooth video transitions with minimal packet loss (ideally less than 1%) [73]. In rural, resource-limited areas with either complete disconnection or limited connectivity (e.g., 2.5G/3G networks), such cloud-based services often fail to function altogether (case in Section 4.1).

A relatively new paradigm, known as Edge Computing, offers an alternative to cloud computing by enabling services to be hosted closer to end users. While edge computing is predominantly used today to reduce latency for time-sensitive applications such as autonomous driving and metaverse applications [66], we argue that its principles are also well-suited for serving disconnected or resource-constrained populations with essential day-to-day services. The widely accepted four-layered architecture of edge computing, comprising mist, edge, fog, and cloud layers, provides an ideal framework for underserved regions. ICT services can be hosted at any one of these layers or decomposed and distributed across multiple layers in a hybrid approach, depending on the community's priorities and the availability of runtime resources. This flexibility enables edge computing to dynamically adapt to the situated and evolving contexts of underserved communities. In both case studies, resource limitations (locally deployed solar-powered systems in Section 4.1 and cost-constrained computing in Section 4.2) underscore the importance of edge-cloud tradeoffs and resource allocation strategies that are both context- and priority-aware.

For example, in a community with limited connectivity, high-priority services can be hosted locally on low-cost hardware, while low-priority services can be accessed from higher layers when connectivity becomes available, which is useful for periodic OER content updates discussed in Section 4.1. Furthermore, by adopting a multi-tenancy model similar to cloud computing, edge devices situated at community-specific locations (schools or community centers), can share resources, including compute, memory, storage, network, and power. Each community member (e.g., a student) or group of users (e.g., a class or a family) can be allocated a portion of these resources to meet their needs. Thus, leveraging the edge-cloud continuum as the foundational system for hosting application services allows communities to better manage limited and intermittent resource availability while still meeting their service requirements.

However, one major challenge lies in the dynamic nature of resource allocation within a community-aware edge-based system. Unlike the static resource allocation models typical in cloud computing, resource availability at the edge may shift based on community priorities. For instance, an individual user's allocated resources might be reallocated to support a group voting for a high-priority task. This dynamic, community-driven approach to resource allocation and task scheduling requires active involvement of the community to guide the system's operation. Designing such systems with community participation as a core component introduces new complexities that are not present in traditional cloud computing and necessitates innovative approaches to system co-design and governance with the community.

5.2.2 Dynamic and Hybrid Connection Mechanisms. The connectivity aspect of ICT4D facilitates communication between entities and actors within a community. Since connectivity can be influenced by the locational context both across and within a community, it is essential to consider the available connectivity mechanisms and dynamically select the most suitable option at runtime based on the community's specific service needs. A community with access to multiple connectivity mechanisms, such as limited cellular internet (e.g., 2.5G/3G), an intranet supported by local Wi-Fi mesh networking, and local file transfers enabled by low-cost hardware (e.g., Bluetooth, mmWave, UWB), can leverage these mechanisms individually or in a hybrid manner to optimize resource usage.

For instance, as a potential solution for the case discussed in Section 4.1, limited internet connectivity could be used solely for periodic content updates and refreshes, while the local intranet serves user requests. In scenarios where users are in close proximity but intranet traffic is congested, a hybrid approach could be adopted. In such cases, webpage content might be loaded via the intranet while larger media files are transferred asynchronously using alternative wireless methods such as Bluetooth or Wi-Fi Direct.

This dynamic and hybrid approach to serving ICT application services, based on the community's connectivity context, promotes more efficient utilization of available resources. However, realizing such an approach requires either application services to be explicitly designed to support dynamic, hybrid connectivity mechanisms or the implementation of middleware that can manage and handle these mechanisms seamlessly, as further discussed in Section 6.1.

5.2.3 Distributed Caching Mechanisms. Instead of repeatedly requesting identical services over limited and congested internet connections, service requests can be fetched once and cached locally, allowing community users to rely on the cached results rather than duplicating requests for the same service within a specific timeframe. Integrating such caching mechanisms at pervasive edge gateways like community routers [44] has shown significant improvements in cache hit rates and latency for browsing web services.

5.3 Design for Resiliency

The locational context of a community, specifically its environmental factors (e.g., harsh weather and climatic conditions) and mobility patterns, can significantly impact the availability of limited resources (i.e., compute, network, power). Additionally, higher-priority community service requests may necessitate termination of lower-priority tasks to free up resources, resulting in service disruptions and intermittent behavior in ICT service availability. Moreover, resource-limited communities are often powered by intermittently available energy sources, such as solar panels or energy harvesting mechanisms (soil-powered, wind-powered, etc.), or rely on grid systems that have significant scheduled downtime, directly impacting device availability and, consequently, service uptime. In such cases, service requests may enter a continuous try-and-fail loop, wasting time and valuable resources without completing useful work.

This issue is particularly severe for long-running service requests, such as large file transfers (Section 4.1), or computationally intensive tasks like training machine learning models. These tasks can monopolize critical resources only to be terminated prematurely, requiring the process to restart entirely. Addressing these challenges is crucial for resource-limited settings to ensure both resilient and reliable access to ICT services. Mechanisms that can play a pivotal role in mitigating these issues include:

- **Delay-tolerant networking (DTN) and Cooperative computing:** These strategies work opportunistically and collaboratively across nodes, by either storing data shards locally across networks and forwarding the useful or needed bits only when its possible–such as when the receiving node or its resources become available. Examples of this in the literature include exploring opportunistic cooperative data offload [45], message ferrying to slowly but surely get data (and the computation it enables) where it needs to be [85], and approaches like Serendipity, that enable mobile devices to use remote computational resources available in other mobile systems in a large local network environment. Each of these approaches explores how to be resilient to disruptions, how to share resources intelligently, and how to opportunistically take advantage of good environments, all to speed up computing and conserve energy. Numerous other methods around computational offloading in mobile edge computing are described in this survey [46]. Of course, these techniques were not made for ICT4D contexts, and require a level of care, and engineering, to be practicable in (for example) the use cases we present.
- **Intermittent Computing:** While the prior set of technologies focus on shared computing, data, and memory across multiple devices, intermittent computing [3] is concerned with ensuring that local computing is highly efficient in the face of disruptions. These approaches range from hardware approaches to federate energy to ensure backup energy is available for key tasks [15, 31], software stacks that enable just-in-time checkpointing of unmodified code with low overhead [74] and full-stack operating systems [82], including for intensive machine learning [24], to networking approaches that enable Bluetooth LE and other low power protocols to work despite interruptions in power of unknown lengths [18]. Each of these approaches ensure that tasks assigned are completed without wasting valuable energy repeating portions of a task. These techniques are complementary to the framework in this paper, and could enable (if extended to application) progress as long as real-time community contexts and constraints are considered, and inform the adaptation.

As one example of a system that benefits from these paradigms, consider a solar-powered system that experiences higher energy availability during daylight hours but lower bandwidth due to congested networks during working hours. These contrasting resource availability patterns highlight the need for dynamic runtime adaptability to balance service availability and reliability. Addressing these scenarios requires capturing real-time community contexts and implementing adaptive mechanisms that can respond intelligently to evolving resource and connectivity conditions. The disruptions experienced in Case I (Section 4.1) due to unreliable solar power and bandwidth fluctuations make clear the need for mechanisms such as intermittent computing and delay-tolerant networking to maintain reliability.

Using a context-aware ICT4D approach and integrating intermittent computing and DTN strategies would improve service reliability and availability. However, achieving this integration poses several challenges. These include: 1) capturing and managing execution states dynamically to minimize task interruption (dynamic checkpointing of program execution states) and 2) breaking down content intelligently for efficient storage and transfer, particularly in low-connectivity scenarios (dynamic content chunking).

5.4 Design for Service-level Adaptability and Extensibility

The majority of the content types accessed over the internet today can be categorized into four main groups: text, image, audio, and video. The bandwidth usage for these content types largely depends on the size of the data being transmitted. Furthermore, modern websites and web services are inherently multimodal, delivering outputs composed of multiple content types simultaneously. For resource-limited communities, browsing such multimodal ICT services using traditional synchronous, completely internet-based approaches may be infeasible and result in service failures.

In such scenarios, adopting asynchronous and adaptive approaches for browsing common content and media types, driven by runtime resource availability and community needs, can significantly improve service success rates. Some potential approaches include:

- **Text-only browsing with "*-to-text" adaptations:** When internet bandwidth and local compute resources are constrained, other content types (e.g., images, audio, and video) can be converted into text-based representations. For example, images can be described textually, and audio or video can be accompanied by subtitles or transcripts. This approach enables users to access key information while minimizing resource usage.
- **Adaptive quality control based on bandwidth availability:** Content quality can be dynamically adjusted to match available bandwidth. For instance, images and videos can be converted to grayscale or lower resolutions, and adaptive bitrate streaming can be used for audio and video content [79]. These adjustments ensure that users can still access services, albeit with reduced quality.
- **Adaptive content partitioning and distributed content aggregation:** Content delivery can be optimized by partitioning content and leveraging both internet bandwidth and local compute resources. For example, an image or video could initially be sent in a low-resolution or black-and-white format along with its metadata. The original quality could then be reconstructed using local compute resources in a distributed manner. This approach balances network load with local processing capabilities.
- **Generative AI-based content reconstruction:** Lightweight generative AI models can be used to minimize bandwidth requirements while maintaining content integrity. For example, critical features of an image can be extracted and transmitted over a low-bandwidth connection. These features can then be used to reconstruct the image from scratch using generative AI models deployed on local edge devices. This distributed approach leverages local compute resources to deliver high-quality outputs despite limited connectivity.

By providing application developers with tools to implement such custom service-level adaptation features, developers can tailor services to meet the specific needs of resource-limited settings. Easy-to-integrate programming models for developers can facilitate the creation of community-context-aware application services, while also offering extensibility to enhance the built-in adaptation features with service-specific policies.

Fig. 3 illustrates a potential annotation-based programming model that supports various mechanisms for retrieving media content based on real-time context. For example, as shown in Fig. 4, during instances of low-bandwidth connectivity, a large media image can be retrieved using the `getGrayScaledAndMetadata()` function, which delivers a grayscaled image along with its coloring metadata. This approach reduces bandwidth usage by approximately

```
@Retention(RetentionPolicy.RUNTIME)
@Target(ElementType.FIELD)
public @interface ContextAwareImage {
    String getAltText();
    String getCompressed();
    String getGrayScaledAndMetadata();
    String getAdjustedScale();
    ...
}
@Target(ElementType.FIELD)
public @interface ContextAwareAudio {
    String getTranscribedText();
    ...
}
@Target(ElementType.FIELD)
public @interface ContextAwareVideo {
    String getSubtitlesText();
    String getFrameChangeCoordinates();
    ...
}
```

Figure 3: Context-aware annotations to support various mechanisms for retrieving media content enabling service adaptability features

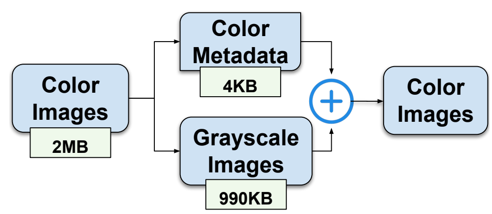


Figure 4: Example of an adaptive image fetching process where a grayscaled image and its coloring metadata can be sent separately during limited bandwidth scenarios and later be combined to get the colored image locally

half, with the coloring process performed client-side. This method is particularly advantageous during periods of bandwidth limitations when adequate power supply is available. However, in scenarios where both bandwidth and power are constrained, the system can adapt by opting for alternative methods, such as `getAltText()` (a text-based image description) or `getAdjustedScale()` (scaled-down versions of the image), to ensure optimal service delivery for community users based on real-time context.

Integrating such approaches empowers application services to actively participate in and contribute to context-aware design. By doing so, it promotes equitable access to ICT services for underserved communities, addressing their unique needs and constraints.

5.5 Key Takeaways and Quickstart Guide

Integrating the above principles to ICT4D design would enable a more context-aware approach to designing ICT system and services that better understands and adheres to the unique and dynamic contexts of resource limited underserved communities, thus improving ICT service accessibility and helping to bridge the digital divide. Given that underserved communities are characterized by limited resources, situated locations, and unique needs, designing ICT systems and services for them requires that key designs be community-context aware. These steps were directly informed by real-world failures and constraints observed in our case studies, where the absence of community-tailored context-awareness resulted in poor service sustainability. Below we briefly outline actionable steps that serve as a quickstart guide for ICT4D designers to start applying context-aware ICT4D design principles.

(1) **Context Identification & Filtering:** Identify important contexts for each community through workshops, surveys, interviews, and participatory design methods [38]. As community needs may differ across different communities, co-designing with the community helps in identifying and prioritizing their unique needs.
(2) **Context Acquisition & Analysis:** Determine possible ways to acquire the identified contexts, either through hardware or software solutions. Hardware-based context acquisition involves integrating low-cost sensors to gather "situated location" contexts, such as environmental conditions (e.g., weather, temperature, etc.). Software-based solutions, on the other hand, include satellite imagery analysis and service-level data analysis. Satellite imagery analysis helps gather "situated location" contexts (e.g., available infrastructures such as communication towers, solar panels, etc.), while service-level analysis provides insights into service resource usage, access patterns, and community contexts in terms of user policies and priorities.
(3) **Context Modeling:** Model the acquired context in a way that is extensible–changes to the context structure (e.g. context metadata) should be easy to integrate. Using flexible data stores without strict relational structures such as key-value (KV) stores enables context extensibility.
(4) **Contextual Reconfiguration:** Enable community users to reconfigure the system based on shared values or policies, supporting "community-in-the-loop" design capabilities.
(5) **Contextual Adaptation:** Ensure that the system can automatically adapt to changing environmental, runtime resource, or community contexts. This can be achieved using a middleware system design (Section 6.1) with an inherent context monitoring component and feedback loop mechanisms to observe contextual changes, quickly determine an optimal balanced state, and trigger necessary adaptations.
(6) **Context Presentation and Interface:** Abstract the underlying technical details while presenting an easy-to-use interface for end users to retrieve information or reconfigure service policies. The design of such user-facing interfaces should focus on reducing cognitive overhead and the technical burden on community users. Design strategies may include integrating support for local languages, voice-based interfaces, and simple interfaces with limited visual elements for reconfiguration.
(7) **Context-based Triggers:** Integrate triggers based on contextual reconfigurations or sudden changes in conditions that require recording and notifying ICT4D designers for manual intervention and further co-design.

6 Discussion and Future Works

Context-aware ICT4D design presents three main challenges: 1) the implementation of such systems, 2) adaptation to unexpected rapidly changing conditions, and 3) ensuring the sustained use of context-aware systems. In this section, we discuss potential solutions and strategies to address these challenges.

6.1 Middleware Architectures

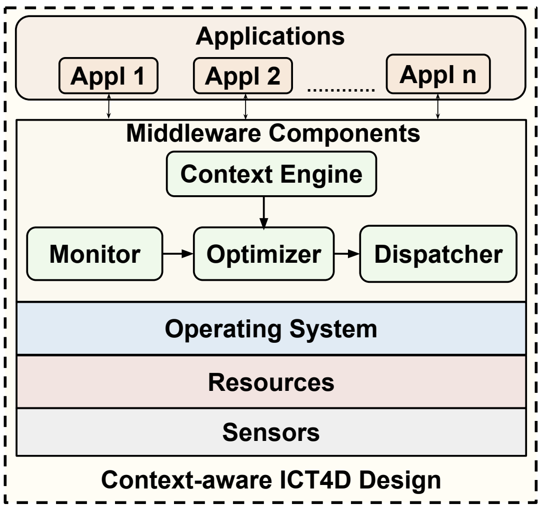
The implementation of a context-aware system for ICT4D that adheres to the design goals stated in Section 5 and integrates with existing technologies requires careful consideration. ICT4D systems are often designed for and/or tailored to the specific needs of a single community, but designing systems optimized for individual communities is often time- and resource-intensive, impractical, and lacks scalability.

These challenges highlight the need for innovative design mechanisms that can capture and adapt to a community's contextual nuances with minimal effort. Such mechanisms must intelligently balance "automated" approaches with "community-in-the-loop" methodologies. *Middleware* architectures integrated with existing edge-based systems deployed at local communities present a promising solution for achieving "community-context-awareness" in a more generalizable and scalable manner. Middleware solutions provide flexibility, facilitates seamless integration with application services, and optimizes service provisioning decisions, thereby enabling sustainable and efficient ICT4D solutions.

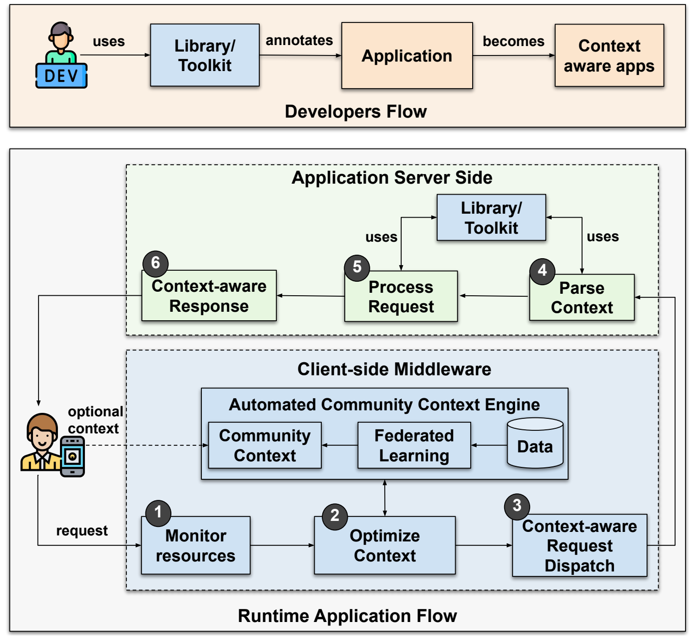
Figure 5a illustrates a potential middleware-based architecture for integrating context-awareness, while Figure 5b details its overall workflow for a dynamic content adaptation use case (Design 5.4). The middleware architecture sits on top of existing sensors, hardware, and operating system, whereby it acquires lower-level resource contexts using the *Monitor* adapter, and integrates and processes community needs context using the *Context Engine*. The system periodically combines community's needs context with its historical service usage data and applies learning mechanisms (such as federated learning) to better understand service needs. As executing such learning mechanisms is costly in terms of time and resource usage, its execution period is decided dynamically by the *Optimizer* based on the optimal resource availability and environmental contexts. During service requests at runtime, these optimized context (both runtime resources and currently recommended community needs) are used as input to the *Dispatcher* which adds these contexts to the service request. On the application/ service side, using extensible programming models and service-level adaptations provided as library/ toolkit (as proposed in Section5.4), these contexts can be further parsed by the service and service-level adaptations can be applied while sending back a response.

6.2 Asset-based Design

This work draws inspiration from **asset-based** community design [81], where the design considers the community's available technological resources as the core asset and aims for its optimal usage to fulfill the community's burning ICT needs. However, we adopt a broader definition of assets and consider infrastructure that promotes development such as roads, community centers, and libraries as additional community assets. Integrating such infrastructure into our context-aware design for ICT4D further complicates the design and requires a way to map the presence of such infrastructure to the community's needs and their technological resource



(a) Context-Aware Middleware Design



(b) Service-level Adaptation Workflow

Figure 5: Middleware Architecture and Service Adaptation Workflow

allocation. For example, a context-aware system in a community with well-maintained roads and transportation infrastructure to nearby community centers or village stations that are equipped with computers may enforce users to travel there for large file transfers rather than utilize limited intranet link bandwidth.

Although such scenarios could be managed by a system that relies on a community's policies with rule-based mechanisms, directly mapping the physical and geographic infrastructural assets of each community at real-time could enhance the design. One way to achieve this is to utilize crowd-sourced mapping and open-data platforms such as Google Maps and regulatory satellite imagery to identify the presence of external infrastructures, such as roads, hospitals, and community centers. Local government websites and datasets such as census reports can provide further insights on internal infrastructures, such as the number of personal computers in community centers. Moreover, there is an opportunity for researchers and HCI practitioners to design community-friendly interfaces for documenting such infrastructural assets in a manual, participatory way.

6.3 Effective Partnerships and Community Co-design

Environmental and runtime contexts might change rapidly (within minutes) which can be addressed through integrated sensing and technologies that enable adaptability. Similarly, evolving community-needs can be addressed by incorporating reconfigurable capabilities, such as community-in-the-loop policies. However, unpredictable and rapidly changing conditions–such as those caused by political instability, power dynamics, or natural calamities–may arise . While adapting to such rapidly changing conditions can be partly handled through community-in-the-loop reconfiguration by redefining and establishing new policies, these situations may require careful consideration. To address such scenarios, the design should include built-in features that allow communities to easily identify and communicate such needs to the ICT4D designers. Depending on the specific impact of these changing scenarios, which will vary across communities, both the community and ICT4D designers can then come together to further introduce new policy engines that best address these rapid and unexpected conditions. Such policy engines can also be designed as modular extensions, enabling other communities facing similar scenarios to reuse them effectively, promoting reusability and sustainability.

A core tenet of our approach to context-aware design is the importance of designing for communities. Therefore, it is essential to ensure that context-aware systems incorporate community preferences and accommodate for sustained use based on community knowledge and resources. This requires intentional implementation and maintenance of such solutions, demanding equal participation from local community members, technical experts, community partners, local and international project partners (such as INGOs, NGOs), economists, research partners, etc. Approaches that would enable a more sustainable context-aware ICT4D design include: integration of open-source technology development practices allowing for others to contribute towards better software maintainability, participatory workshops and co-designing with the community to train them on how to best benefit from the enabled services, community trainings for debugging and maintaining community technological infrastructure, etc.

Developing and deploying systems using such a context-aware ICT4D approach should be further explored and its impact on long-term, sustained use by the target communities needs to be studied.

6.4 Future Work

This work highlights several opportunities for future research to enable the realization of context-aware ICT4D. The concept of "community-in-the-loop" policies and priorities remains underexplored and requires significant contributions from both systems and HCI research domains. Designing systems that balance individual, group, and community needs while incorporating dynamic, community-driven decision-making processes presents unique challenges. Developing intuitive interfaces and robust frameworks to support this collaborative approach will be critical for advancing ICT4D in underserved communities.

Additionally, the resource overhead and practicality of plug-and-play approaches, as described in Section 5.1.2 require further investigation. The chosen approach should balance acceptability in terms of resource constraints while aligning with the community's preferences and needs. Future research should focus on identifying solutions that are both efficient and adaptable to the specific contexts of underserved communities.

7 Conclusion

This position paper re-envisions context-aware computing for ICT4D by transitioning from an individual-focused model to a community-centric approach. We redefined context in ICT4D as the interaction between location, community needs, and resource availability, highlighting the importance of adaptability in resource-constrained environments. By examining real-world case studies, we illustrated how context-aware design can enhance digital inclusion and sustainability. Our proposed framework incorporates community participation, flexible resource management, and resilient service adaptability and extensibility strategies to improve ICT accessibility. However, challenges such as scalability, efficient implementation and deployment, and participatory design remain. Future research should explore adaptive solutions and collaboration to ensure long-term, community-driven ICT advancements. This work contributes to narrowing the digital divide by making technology more accessible and equitable for underserved communities.

Acknowledgments

This research was supported by the National Science Foundation through grants RISE-2209226, and CNS-2145584. We would also like to acknowledge support by the Alfred P. Sloan Foundation, VMware, Google, and Catherine M. and James E. Allchin. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or other supporters. We thank Dr. Ruthie Yow, Archit Sengupta, and Dipesh Rawat for their assistance with the HSAAT case study. We are grateful to our organizational and community partners from both OLE Nepal and HSAAT for their continued support. Finally, thanks to Eric Greenlee, Regan Maharjan, and Pawan Dhakal for their support and valuable insights.

References

[1] G Abowd. 1999. Towards a better understanding of context and context-awareness. *Springer google schola* 2 (1999), 23–32.
[2] Kebiru Abu and Stephen U Ufoaroh. 2019. Estimation and application of TV whitespaces for rural broadband connectivity (Internet Access). *Am. J. Eng. Res* 8 (2019), 19–28.
[3] Saad Ahmed, Bashima Islam, Kasim Sinan Yildirim, Marco Zimmerling, Przemysław Pawełczak, Muhammad Hamad Alizai, Brandon Lucia, Luca Mottola, Jacob Sorber, and Josiah Hester. 2024. The internet of batteryless things. *Commun. ACM* 67, 3 (2024), 64–73.
[4] Syed Ishtiaque Ahmed, Md Romael Haque, Jay Chen, and Nicola Dell. 2017. Digital privacy challenges with shared mobile phone use in Bangladesh. *Proceedings of the ACM on Human-computer Interaction* 1, CSCW (2017), 1–20.
[5] Syed Ishtiaque Ahmed, Md Romael Haque, Irtaza Haider, Jay Chen, and Nicola Dell. 2019. " Everyone Has Some Personal Stuff" Designing to Support Digital Privacy with Shared Mobile Phone Use in Bangladesh. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.
[6] Asam Hamed Abbas Almohamed, Reem Talhouk, and Dhaval Vyas. 2022. Towards a conceptual framework for understanding the challenges in refugee re-settlement. *Proceedings of the ACM on human-computer interaction* 6, GROUP (2022), 1–27.
[7] Rahul Awati. 2023. What is ICT4D (Information and Communications Technologies for Development)? — techtarget.com. https://www.techtarget.com/whatis/definition/ICT4D-Information-and-Communications-Technologies-for-Development. [Accessed 18-01-2025].
[8] André Baart, Anna Bon, Victor de Boer, Francis Dittoh, Wendelien Tuijp, and Hans Akkermans. 2019. Affordable voice services to bridge the digital divide: Presenting the Kasadaka platform. In *Web Information Systems and Technologies: 14th International Conference, WEBIST 2018, Seville, Spain, September 18–20, 2018, Revised Selected Papers 14*. Springer, 195–220.
[9] Matthias Baldauf, Schahram Dustdar, and Florian Rosenberg. 2007. A survey on context-aware systems. *International Journal of ad Hoc and ubiquitous Computing* 2, 4 (2007), 263–277.
[10] Giulia Barbareschi, Ben Oldfrey, Long Xin, Grace Nyachomba Magomere, Wycliffe Ambeyi Wetende, Carol Wanjira, Joyce Olenja, Victoria Austin, and Catherine Holloway. 2020. Bridging the Divide: Exploring the use of digital and physical technology to aid mobility impaired people living in an informal settlement. In *Proceedings of the 22nd International ACM SIGACCESS Conference on Computers and Accessibility*. 1–13.
[11] Jeffrey P Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, et al. 2010. Vizwiz: nearly real-time answers to visual questions. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*. 333–342.
[12] Patrick Loola Bokonda, Khadija Ouazzani-Touhami, and Nissrine Souissi. 2020. A Practical Analysis of Mobile Data Collection Apps. *International Journal of Interactive Mobile Technologies* 14, 13 (2020).
[13] Alvin TS Chan and Siu-Nam Chuang. 2003. MobiPADS: a reflective middleware for context-aware mobile computing. *IEEE Transactions on software engineering* 29, 12 (2003), 1072–1085.
[14] Marshini Chetty and Rebecca E Grinter. 2007. HCI4D: HCI challenges in the global south. In *CHI'07 extended abstracts on Human factors in computing systems*. 2327–2332.
[15] Alexei Colin, Emily Ruppel, and Brandon Lucia. 2018. A reconfigurable energy storage architecture for energy-harvesting devices. In *Proceedings of the Twenty-Third International Conference on Architectural Support for Programming Languages and Operating Systems*. 767–781.
[16] Stefany Cruz, Alexander Redding, Connie W Chau, Claire Lu, Julia Persche, Josiah Hester, and Maia Jacobs. 2023. EquityWare: Co-designing wearables with and for low income communities in the US. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–18.
[17] Haixiao Dai, Phong Lam Nguyen, and Cat Kutay. 2024. Offline collaborative learning approach for remote Northern territory students. *Interactive Technology and Smart Education* 21, 1 (2024), 67–82.
[18] Jasper De Winkel, Haozhe Tang, and Przemysław Pawełczak. 2022. Intermittently-powered bluetooth that works. In *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services*. 287–301.
[19] Nicola Dell, Trevor Perrier, Neha Kumar, Mitchell Lee, Rachel Powers, and Gaetano Borriello. 2015. Digital workflows in global development organizations. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. 1659–1669.
[20] Anind K Dey, Gregory D Abowd, et al. 2000. The context toolkit: Aiding the development of context-aware applications. In *Workshop on Software Engineering for wearable and pervasive computing*, Vol. 278. Citeseer.
[21] Reinald Adrian DL Pugoy, Consuelo DL Habito, and Roberto B. Figueroa Jr. 2016. Hybrid online/offline mobile solutions for accessing open educational resources in areas with poor internet connectivity. *Asian Association of Open Universities*

Journal 11, 2 (2016), 182–196.

[22] Michaelanne Dye, David Nemer, Josiah Mangiameli, Amy Bruckman, and Neha Kumar. 2018. The human infrastructure of El Paquete, Cuba's offline internet. *Interactions* 26, 1 (2018), 58–62.

[23] Gaël Fromentoux, Arnaud Braud, and Xavier Marjou. 2012. Scenario to Serve Remote Areas in Emerging Countries with the Village Internet Service Station. In *e-Infrastructure and e-Services for Developing Countries: Third International ICST Conference, AFRICOMM 2011, Zanzibar, Tanzania, November 23-24, 2011, Revised Selected Papers 3*. Springer, 1–13.

[24] Graham Gobieski, Brandon Lucia, and Nathan Beckmann. 2019. Intelligence beyond the edge: Inference on intermittent embedded systems. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*. 199–213.

[25] Eric Greenlee, Blaine Rothrock, Hyeonwook Kim, Ellen Zegura, and Josiah Hester. 2024. "The Devil You Know": Barriers and Opportunities for Co-Designing Microclimate Sensors, A Case Study of Manoomi. *ACM Journal on Computing and Sustainable Societies* 2, 3 (2024), 1–30.

[26] Tao Gu, Hung Keng Pung, and Da Qing Zhang. 2005. A service-oriented middleware for building context-aware services. *Journal of Network and computer applications* 28, 1 (2005), 1–18.

[27] Carlos Guerrero Millan, Bettina Nissen, and Larissa Pschetz. 2024. Cosmovision Of Data: An Indigenous Approach to Technologies for Self-Determination. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–13.

[28] Susan Hansen. 2019. *Culturally tailored ICT4D: designing with communities, culture and context*. Ph. D. Dissertation. University of Technology Sydney (Australia).

[29] Shaddi Hasan, Mary Claire Barela, Matthew Johnson, Eric Brewer, and Kurtis Heimerl. 2019. Scaling Community Cellular Networks with {CommunityCellularManager}. In *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*. 735–750.

[30] Sven Helmer, Claus Pahl, Julian Sanin, Lorenzo Miori, Stefan Brocanelli, Filippo Cardano, Daniele Gadler, Daniel Morandini, Alessandro Piccoli, Saifur Salam, et al. 2016. Bringing the cloud to rural and remote areas via cloudlets. In *Proceedings of the 7th Annual Symposium on Computing for Development*. 1–10.

[31] Josiah Hester and Jacob Sorber. 2017. Flicker: Rapid prototyping for the batteryless internet-of-things. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*. 1–13.

[32] Jong-yi Hong, Eui-ho Suh, and Sung-Jin Kim. 2009. Context-aware systems: A literature review and classification. *Expert Systems with applications* 36, 4 (2009), 8509–8522.

[33] Ahmed Imran. 2023. Why addressing digital inequality should be a priority. *The Electronic Journal of Information Systems in Developing Countries* 89, 3 (2023), e12255.

[34] Muhammad A Imran, Marco Zennaro, Olaoluwa R Popoola, Luca Chiaraviglio, Hongwei Zhang, Pietro Manzoni, Jaap van de Beek, Robert Stewart, Mitchell Arij Cox, Luciano Leonel Mendes, et al. 2024. Exploring the Boundaries of Connected Systems: Communications for Hard-to-Reach Areas and Extreme Conditions. *Proc. IEEE* (2024).

[35] Nabeel Janjua, Yan Li, and Manoj Thomas. 2017. Developing information and communication Technologies for education in Haiti. *VA Engage Journal* 5, 1 (2017), 3.

[36] Asad Javed et al. 2016. *Container-based IoT sensor node on raspberry Pi and the Kubernetes cluster framework*. Master's thesis.

[37] Matthew Johnson, Jenny Liang, Michelle Lin, Sudheesh Singanamalla, and Kurtis Heimerl. 2021. Whale watching in inland indonesia: Analyzing a small, remote, internet-based community cellular network. In *Proceedings of the Web Conference 2021*. 1483–1494.

[38] Matthew William Johnson, Esther Han Beol Jang, Frankie O'Rourke, Rachel Ye, and Kurtis Heimerl. 2021. Network Capacity as Common Pool Resource: Community-Based Congestion Management in a Community Network. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1 (2021), 1–25.

[39] Byungjin Jun, Fabián E Bustamante, Sung Yoon Whang, and Zachary S Bischof. 2019. AMP up your mobile web experience: Characterizing the impact of Google's accelerated mobile project. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–14.

[40] Santosh Kawade and Maziar Nekovee. 2012. Is wireless broadband provision to rural communities in TV whitespaces viable? A UK case study and analysis. In *2012 IEEE International Symposium on Dynamic Spectrum Access Networks*. IEEE, 461–466.

[41] Toru Kishida and Y Fukukara. 2014. Open Online Learning Environment utilizing OER contents. OCW Consortium Global Conference, Ljubljana, Slovenia.

[42] Rinchen Kyi and Neeraj Laishram. 2022. Concerns, implications and Possible Barriers to Licensing: A Legal and Operational Analysis on Setting up Starlink in India. *Jus Corpus LJ* 3 (2022), 575.

[43] Xin Li, Martina Eckert, José-Fernán Martinez, and Gregorio Rubio. 2015. Context aware middleware architectures: Survey and challenges. *Sensors* 15, 8 (2015), 20570–20607.

[44] Zhengquan Li, Summit Shrestha, Zheng Song, and Eli Tilevich. 2024. Edge Cache on WiFi Access Points: Millisecond-Level App Latency Almost for Free. In *2024 IEEE 44th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 601–612.

[45] Zongqing Lu, Xiao Sun, and Thomas La Porta. 2017. Cooperative data offload in opportunistic networks: From mobile devices to infrastructure. *IEEE/ACM Transactions on Networking* 25, 6 (2017), 3382–3395.

[46] Pavel Mach and Zdenek Becvar. 2017. Mobile edge computing: A survey on architecture and computation offloading. *IEEE communications surveys & tutorials* 19, 3 (2017), 1628–1656.

[47] Apurv Mehra, Vishali Sairam, and Kashish Mittal. 2024. BlendNet: An Assisted Digital Distribution Platform for Underserved Populations. In *Proceedings of the 7th ACM SIGCAS/SIGCHI Conference on Computing and Sustainable Societies*. 1–17.

[48] Carlos Guerrero Millan, Bettina Nissen, and Larissa Pschetz. 2024. Cosmovision of data: An indigenous approach to technologies for self-determination. In *CHI'24: Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. ACM, 1–22.

[49] OLE Nepal. 2024. OLE Nepal — olenepal.org. https://olenepal.org/our-stories/. Accessed: 2024-10-16.

[50] James Newman, Robert Belson, and Fabián E Bustamante. 2019. Scaling Up Your Web Experience, Everywhere. In *Proceedings of the 20th International Workshop on Mobile Computing Systems and Applications*. 87–92.

[51] Beatriz Palacios Abad, Elizabeth Belding, Morgan Vigil-Hayes, and Ellen Zegura. 2024. Mending the Fabric: the Contentious, Collaborative Work of Repairing Broadband Maps. *Proceedings of the ACM on Human-Computer Interaction* 8, CSCW2 (2024), 1–24.

[52] Ayush Pandey, Matteo Varvello, Syed Ishtiaque Ahmed, Shurui Zhou, Lakshmi Subramanian, and Yasir Zaki. 2025. MAML: Towards a Faster Web in Developing Regions. In *Proceedings of the ACM on Web Conference 2025*. 727–739.

[53] Bipun Man Pati, Raju Chohan, Raunak Ghimire, Arun Kunwar, and Anup Shrestha. 2024. An Assessment Study on TV White Space in Nepal Using Low-Cost Spectrum Analyzer. In *2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS)*. IEEE, 1–7.

[54] Alex Pentland, Richard Fletcher, and Amir Hasson. 2004. Daknet: Rethinking connectivity in developing nations. *Computer* 37, 1 (2004), 78–83.

[55] Charith Perera, Arkady Zaslavsky, Peter Christen, and Dimitrios Georgakopoulos. 2013. Context aware computing for the internet of things: A survey. *IEEE communications surveys & tutorials* 16, 1 (2013), 414–454.

[56] Preeja Pradeep and Shivsubramani Krishnamoorthy. 2019. The MOM of context-aware systems: A survey. *Computer Communications* 137 (2019), 44–69.

[57] Sonel Pyram. 2024. Future directions for context in ICT4D: A systematic literature review. *Information Development* (2024), 02666669241248149.

[58] Anand Ranganathan and Roy H Campbell. 2003. A middleware for context-aware agents in ubiquitous computing environments. In *ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing*. Springer, 143–161.

[59] Manika Saha, Tom Bartindale, Sharifa Sultana, Gillian Oliver, Dan Richardson, Shakuntala Haraksingh Thilsted, Syed Ishtiaque Ahmed, and Patrick Olivier. 2022. Commissioning Development: Grantmaking, Community Voices, and their Implications for ICTD. In *Proceedings of the 2022 International Conference on Information and Communication Technologies and Development*. 1–18.

[60] Manika Saha, Tom Bartindale, Delvin Varghese, Stephen Lindsay, Dan Richardson, Syed Ishtiaque Ahmed, and Patrick Olivier. 2023. Community voice as data: affordances of participatory videos for international program development. In *Proceedings of the 2023 CHI conference on human factors in computing systems*. 1–16.

[61] Nithya Sambasivan, Nimmi Rangaswamy, Ed Cutrell, and Bonnie Nardi. 2009. Ubicomp4D: infrastructure and interaction for international development–the case of urban indian slums. In *Proceedings of the 11th international conference on Ubiquitous computing*. 155–164.

[62] Nithya Sambasivan and Thomas Smyth. 2010. The human infrastructure of ICTD. In *Proceedings of the 4th ACM/IEEE international conference on information and communication technologies and development*. 1–9.

[63] Bill Schilit, Norman Adams, and Roy Want. 1994. Context-aware computing applications. In *1994 first workshop on mobile computing systems and applications*. IEEE, 85–90.

[64] Manasvini Sethuraman, Anirudh Sarma, Netra Ghaisas, Adwait Bauskar, Ashutosh Dhekne, Anand Sivasubramaniam, and Kishore Ramachandran. 2024. BASS: A Resource Orchestrator to Account for Vagaries in Network Conditions in Community Wi-Fi Mesh. In *Proceedings of the 25th International Middleware Conference*. 131–144.

[65] Spencer Sevilla, Matthew Johnson, Pat Kosakanchit, Jenny Liang, and Kurtis Heimerl. 2019. Experiences: Design, implementation, and deployment of CoLTE, a community LTE solution. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–16.

[66] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. 2016. Edge computing: Vision and challenges. *IEEE internet of things journal* 3, 5 (2016), 637–646.

[67] Summit Shrestha, Zhengquan Li, Khairul Mottakin, Zheng Song, and Qiang Zhu. 2023. From Tight Coupling to Flexibility: A Digital Twin Middleware Layer for the ShakeAlert System. In *2023 IEEE/ACM Symposium on Edge Computing (SEC)*. IEEE, 313–318.

[68] Samiksha Shukla, Kritica Bisht, Kapil Tiwari, and Shahid Bashir. 2023. Comparative study of the global data economy. In *Data economy in the digital age*. Springer, 63–86.

[69] Samiksha Shukla, Arundhati Kolahal, MM Padmakumar, Lija Jacob, and Jossy P George. 2022. Open Educational Resources: The Future of Learning. In *Handbook of Research on the Global View of Open Access and Scholarly Communications*. IGI Global, 288–313.

[70] Shani Evenstein Sigalov, Anat Cohen, and Rafi Nachmias. 2023. Open educational resources for literacies, diversity, equity and inclusion: The case of integrating wikidata into higher education. In *Open Educational Resources in Higher Education: A Global Perspective*. Springer, 279–306.

[71] Samuel Sudar, Matt Welsh, and Richard Anderson. 2018. Siskin: Leveraging the Browser to Share Web Content in Disconnected Environments. In *Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies*. 1–7.

[72] Sharifa Sultana, François Guimbretière, Phoebe Sengers, and Nicola Dell. 2018. Design within a patriarchal society: Opportunities and challenges in designing for rural women in bangladesh. In *Proceedings of the 2018 CHI conference on human factors in computing systems*. 1–13.

[73] Zoom Support. 2024. Accessing meeting and phone statistics — support.zoom.com. https://support.zoom.com/hc/en/article?id=zm_kb&sysparm_article=KB0070504. [Accessed 2025-01-15].

[74] Milijana Surbatovich, Naomi Spargo, Limin Jia, and Brandon Lucia. 2023. A type system for safe intermittent computing. *Proceedings of the ACM on Programming Languages* 7, PLDI (2023), 736–760.

[75] Luca Turchet and Paolo Casari. 2024. The Internet of Musical Things Meets Satellites: Evaluating Starlink Support for Networked Music Performances in Rural Areas. In *2024 IEEE 5th International Symposium on the Internet of Sounds (IS2)*. IEEE, 1–8.

[76] Aditya Vashistha, Edward Cutrell, Nicola Dell, and Richard Anderson. 2015. Social media platforms for low-income blind people in India. In *Proceedings of the 17th International ACM SIGACCESS conference on computers & accessibility*. 259–272.

[77] Athanasios Vasilakos, Yan Zhang, and Thrasyvoulos Spyropoulos. 2016. *Delay tolerant networks*. CRC press Boca Raton, FL, USA.

[78] Davide Vega, Llorenc Cerda-Alabern, Leandro Navarro, and Roc Meseguer. 2012. Topology patterns of a community network: Guifi. net. In *2012 IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 612–619.

[79] Nanavath Venkatesh Naik and Kasa Madhavi. 2024. Adaptive QoE-based video delivery by optimizing the bitrate of the video course using network bandwidth and user bandwidth. *Concurrency and Computation: Practice and Experience* 36, 16 (2024), e8114.

[80] Morgan Vigil-Hayes, Elizabeth Belding, and Ellen Zegura. 2017. FiDO: A Community-based Web Browsing Agent and CDN for Challenged Network Environments. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–25.

[81] Marisol Wong-Villacres, Carl DiSalvo, Neha Kumar, and Betsy DiSalvo. 2020. Culture in action: Unpacking capacities to inform assets-based design. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–14.

[82] Yilun Wu, Byounguk Min, Mohannad Ismail, Wenjie Xiong, Changhee Jung, and Dongyoon Lee. 2024. {IntOS}: Persistent embedded operating system and language support for multi-threaded intermittent computing. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*. 425–443.

[83] Jian Xu, Syed Masum Billah, Roy Shilkrot, and Aruna Balasubramanian. 2019. DarkReader: bridging the gap between perception and reality of power consumption in smartphones for blind users. In *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility*. 96–104.

[84] Whitney E Zahnd, Nathaniel Bell, and Annie E Larson. 2022. Geographic, racial/ethnic, and socioeconomic inequities in broadband access. *The Journal of Rural Health* 38, 3 (2022), 519–526.

[85] Wenrui Zhao, Mostafa Ammar, and Ellen Zegura. 2004. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*. 187–198.