X-SENSE: Sensing in Extreme Environments

Jan Beutel, Bernhard Buchli, Federico Ferrari, Matthias Keller, Lothar Thiele, Marco Zimmerling
Computer Engineering and Networks Laboratory
ETH Zurich, 8092 Zurich, Switzerland
Email: {beutel,bbuchli,ferrari,kellmatt,thiele,zimmerling}@tik.ee.ethz.ch

***Abstract*—The field of Wireless Sensor Networks (WSNs) is now in a stage where serious applications of societal and economical importance are in reach. For example, it is well-known that the global climate change dramatically influences the visual appearance of mountain areas like the European Alps. Very destructive geological processes are triggered or intensified, affecting the stability of slopes and possibly inducing landslides. Up to now, however, the interactions between these complex processes are poorly understood. To significantly advance the knowledge of these interactions, we advocate the use of wireless sensing technology as a new scientific instrument for environmental monitoring under extreme conditions. Large spatio-temporal variations in temperature and humidity, mechanical forces, snow and ice coverage, and unattended operation play a crucial role in long-term, high-altitude deployments. Despite these challenges, we argue that in order to reach the set out goals it is inevitable that WSNs be created as a high-quality scientific instrument with known and predictable properties, rather than as a research toy delivering average observations at best. In this paper, we present key techniques for achieving highly reliable, yet resource-efficient WSNs based on our longstanding experience with productive WSNs measuring permafrost processes in the Swiss Alps.**

I. INTRODUCTION

X-SENSE (successor to PERMASENSE [2]) is a joint, interdisciplinary project that has the ambitious goal of developing wireless sensing technology as a new scientific instrument for environmental monitoring under extreme conditions in terms of temperature variations, humidity, mechanical forces, snow coverage as well as unattended operation that are needed for long-term deployments.

In 2008 we started to deploy Wireless Sensor Networks (WSNs) to collect environmental data related to permafrost at the Matterhorn and Jungfraujoch. Each deployment consists of around 20 sensor nodes, a base station, and extra equipment such as weather stations and video cameras to remotely monitor on-site operation. To this date, after more than two years of continuous operation, we have collected about 100 million data samples. The extensive experience gained with all aspects of operating these WSNs provides the X-SENSE project with the background necessary to extend the work performed, and develop a set of environmental sensors for the observation of ground-based terrain movement in high-alpine regions. Further leveraging familiarity with ultra low-power sensors used for monitoring steep bedrock permafrost [5], we are currently designing a set of more capable sensors. With respect to rather simple sensors developed for PERMASENSE, which display constant, and low sampling and data rates, the X-SENSE nodes are targeted at more complex sensors, having higher, and variable data rates, require user-interaction, or in-network data fusion.

Figure 1. The Matterhorn field site.

In this paper we review our past experience with designing the existing low data-rate sensor network. Keeping the X-SENSE goals in mind, we present key findings, and discuss elements of the design methodology used. The paper concludes with an outlook on ongoing and future work in which we try to circumvent the process of continuously refining and optimizing a target architecture in numerous iterations until a satisfactory result is achieved. Instead, by means of larger-scale prototypes deployed in the field, we try to assess parameters that influence architectural and procedural decisions. By doing so, the prototypes are exposed to the conditions of the target application site and yield representative sensor, and behavioral trace data. The sample data produced during the design process is used to assess the reaction of system design decisions, tradeoffs, and parameterization, and drive early steps in data analysis – all in all critical insights that are unlikely to be satisfied by the use of formal models alone.

II. SYSTEM ARCHITECTURE

WSNs for environmental monitoring applications [1], [8], [9] typically employ a three-tiered system architecture. On the first tier, a set of low-power sensor nodes constitutes the heart of the system. These sensor nodes form an ad-hoc wireless network, jointly relaying sensed data over multiple hops to one or more sink nodes (base stations), which comprise the second tier. Sink nodes are usually equipped with more powerful

978-3-9810801-7-9/DATE11/© 2011 EDAA

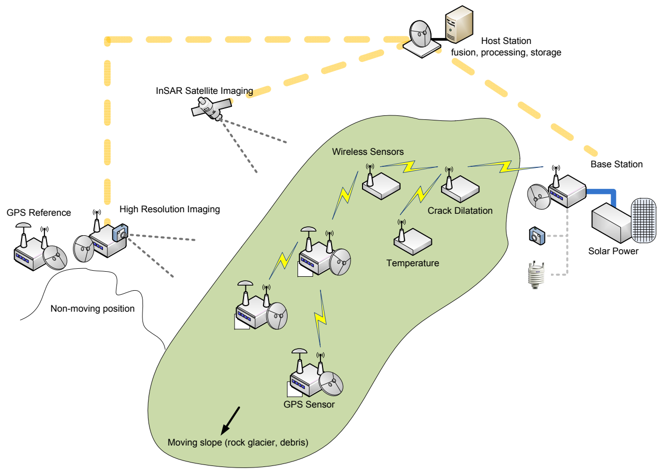


Figure 2. A multitude of sensor types as well as the fusion of information from different time and space scales are used for the detailed analysis of rock glacier terrain movement in the Swiss Alps.

hardware and connect the remote WSN to the Internet. Finally, on the third tier, servers and additional backend infrastructure provide storage and processing capabilities, and interfaces for users to interact with the network and perform maintenance tasks.

We have adopted this architectural approach in the X-SENSE project. In particular, the Shockfish TinyNode [4], a 16-bit micro-controller platform with a low-power ISM-band radio, is used as the sensor node core. It is augmented by the custom-built sensor interface board (SIB), which integrates power supply, a number of in-system monitoring components such as temperature, humidity, voltage, and current monitors [2], and extended data storage (SD card). The complete node electronics is boxed in a solid enclosure to protect from water penetration and moderate rock fall.

The sensor nodes run a customized version of the Dozer [3] multi-hop networking protocol, which allows collection of data (sampled every two minutes) with an average power footprint below 200 μA. A base station (see Figure 8), composed of a more capable Embedded Linux platform, aggregates all data on the field site. Using a local buffering and synchronization mechanism, data are relayed to a database server and published on our web servers, see Section IV. While there are several options for long-haul point-to-point communication, including cellular networks, satellite phones, and custom wireless links, a directional WiFi to link on-site base stations to access points located in the valley appeared to be superior in terms of energy efficiency, bandwidth and reliability.

New challenges found in the X-SENSE project ask for instrumentations at higher sampling and data rates than the previous PERMASENSE data acquisition architecture was designed for. The new architecture, shown in Figure 2, evolved through extension of the existing architecture. The former first tier is no longer implemented using low-power sensor nodes alone; instead, it now also includes more powerful devices, capable of processing large amounts of data (*e.g.*, raw images taken with a high-resolution camera). As an alternative to only

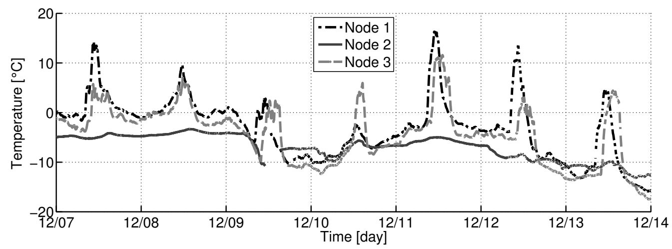


Figure 3. Temporal and spatial temperature variations.

using low-power ISM-band communication on a deployment site, a WiFi link supporting high data rates now enhances the first tier. System extensions further include numerous integration efforts: Existing ISM-band communication now also acts as a management plane, allowing the control (*e.g.*, switching on and off) of complex and power-hungry devices such as GPS sensors. To facilitate the integration of more complex sensors, base and host stations need are extended with additional control logic. Moreover, the backend data management must be provisioned to cater both old and new classes of sensors.

III. CHALLENGING PHYSICAL ENVIRONMENTS

The X-SENSE system is designed to operate autonomously over several years. However, the environmental conditions at high-altitude areas of more than 3,500 meters are extremely severe. The possibilities of lightning, avalanches, periods of prolonged ice or snow cover, and strong solar radiation represent significant system reliability challenges.

Alpine environments experience extreme climatic variations. In winter, temperatures can drop below -20 °C, while the radiation from direct sunlight during a summer day can heat up a system in an enclosure to +60 °C. Depending on the locations of nodes, temperature variations show a distinctive pattern and therefore, affects their operation in significantly different ways. For example, sensors on the north side, or those buried in snow, are less exposed to temperature dynamics than sensors on the south with direct exposure to sunlight. Figure 3 shows an example of the temporal and spatial variability of ambient temperature experienced by three sensor nodes over a week. During the day, temperature readings among different nodes can vary by more than 20 °C; at some nodes it can even change by this amount within a few hours.

System components need to be carefully designed and tested in order to ensure a reliable operation under such harsh conditions. Temperature and humidity stress electrical components, leading to phenomena like spurious short-circuits, memory errors, degradation of battery capacity, additional current consumption of electrical components, and decreasing accuracy of sensors and analog-to-digital converters (ADC). These changes in electrical properties can have a direct impact on network lifetime and data quality. Moreover, the end-to-end performance of the system is affected by all components, ranging from data generation at the sensors to analysis tools applied to diverse datasets. The main challenge is thus to ensure a proper interaction among these components, so that the whole communication and processing chain from sensors to the data storage results in a reliable and predictable functionality in terms of data quality, data yield and network lifetime.

In the following, we address the most important challenges presented by alpine environments. We further discuss our design choices to counteract these issues, and briefly explain the underlying testing methodology.

A. Clock drift

Temperature, power quality, and system design choices affect the stability of oscillators that provide the heartbeat of digital systems. Instability of the local clock subsystem has several detrimental consequences. For instance, the accuracy of an ADC may be affected, which can lead to lower quality of the sensed data. Furthermore, performance of other digital components, such as processors and memories may also decrease due to the clock degradation. Spatial and temporal variations of temperature lead to different clock drifts for different nodes. Time synchronization is thus required to make nodes share a common knowledge of time to permit energy efficient communication.

In order to evaluate the resilience of a system to harsh environmental conditions, it is inevitable to perform a continuous transition from simple system tests under ideal environmental conditions, via extensive test under conditions that are as close as possible to the expected deployment environment (outdoor or climate chamber), to the final deployment at the field sites. Such a continuous process has been installed and used in the described project, see also Section III-D.

For example, in order to create artificial temperature variations, and test system components under extreme conditions, we make use of a climate chamber. First, we evaluate the *clock drift*, that is the deviation of the oscillator frequency from its nominal value as a function of temperature. With temperatures ranging between -30 °C and +40 °C, the frequency of a crystal drifts from its nominal value to a maximum of -70 parts per million (ppm). We exploit the temperature sensors equipping each node to correct oscillator frequency errors based on the findings during the climate chamber experiments. The processor periodically triggers a reading from the temperature sensors, whose readings are used to compensate for the estimated clock drift. Tests with the climate chamber show that clock drift is significantly reduced with this approach, and always lies within bounds of ± 5 ppm. This is important, as more stable clocks allow significant reduction of the guard time duration, which are used by the wireless communication protocol to tolerate possible synchronization errors among sensor nodes. As a result, the amount of time the wireless radio is turned on is drastically reduced, leading to a higher energy efficiency of the system.

We further conduct experiments with the climate chamber to analyze how temperature affects the timing behavior of important operations executed by a sensor node. This may include, but is not limited to data acquisition (DAQ) from a sensor, and storage of data into the flash memory. Figure 4 shows results from a 14-hour experiment in the climate

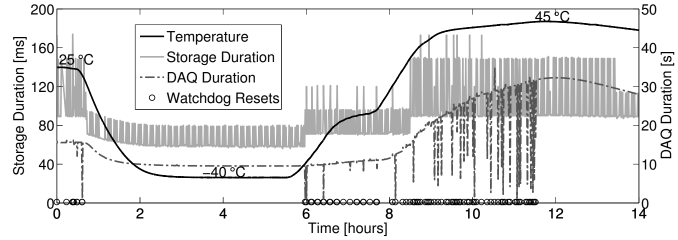


Figure 4. Effects of temperature variations on the duration of operations.

chamber. Starting with room-temperature (25 °C), we decrease the temperature to roughly -40 °C over the period of an hour, after which we leave it stable for four hours. Then we increase it to +45 °C, and let it stay at that value for four more hours. The time required to store data into the flash memory indicates a strong dependency on temperature, and varies between 70 ms and 150 ms. Moreover, the time required to acquire a value from a sensor has a strong temperature correlation, and ranges between 10 and 30 seconds. When there is no compensation for the clock drift, strong temperature variations lead to system malfunctioning: Figure 4 shows several system resets caused by the watchdog timer. These temperature-related pre-deployment experiments prove to be extremely important; they allow proper tuning of timing dependent operations executed by the processor. Without this a-priori knowledge, the temperature-related effects might be underestimated and possibly lead to system malfunction.

B. Real Batteries and Power Draw

Sensor nodes in the field are powered by real batteries, not by steady laboratory power supplies (*e.g.*, USB cords). Since batteries deplete faster under cold conditions, this necessitates the selection of suitable cells based on the expected peak current consumption, temperature constraints, and lifetime requirements [2], [8]. Furthermore, the ultra low-power consumption on the order of μAs is close to the self discharge of common batteries, which complicates the estimation of remaining capacity and lifetime. Considering the deviation of battery models from reality due to these influences, batteries affect both power quality and yield that the system can expect.

In addition to hardware considerations, we have to ensure that our networking protocols operate correctly and at an acceptable energy-efficiency. To this end, we have developed a comprehensive testing environment including indoor and outdoor WSN testbeds [11], a supportive testing methodology and framework [13], and techniques based on formal analysis to automatically identify incorrect behavior using power measurements [12].

For instance, Figure 5 plots power traces obtained from a single node during one of our daily, automated test runs. In a disconnected setting, illustrated in Figure 5, the node repeatedly turns on its radio $\langle 1 \rangle$ to scan for beacons from other nodes. As it never receives any beacon, it steadily reduces the listening time to the minimum $\langle 2 \rangle$. Every two minutes $\langle 3 \rangle$ the node powers up the SIB, reads out the sensors, switches off the SIB, and writes a record of the sampled data to the external

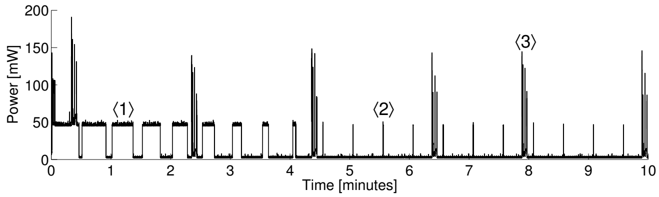


Figure 5. Disconnected operation of an isolated node.

SD card. The node never attempts to send any beacons or data packets unless it discovers neighboring nodes.

Such detailed power traces are essential in finding power leaks due to software bugs. Moreover, they demonstrate the (correct) protocol behavior under exceptional conditions, such as when a node is disconnected from the rest of the network.

C. Transient Links and Disconnected Operation

Highly variable link quality is a common phenomenon in wireless systems, and can cause transmission errors and bursty packet loss [14]. In addition, snow and ice cover on antennas can lead to long-term network disconnects, and so do rock fall, avalanches, riming, permanent water damage, vandalism or theft. As a result, the system design must account for phases during which parts of the network are incapable of communicating with, or unreachable by neighboring nodes. Besides external influences, such exceptional situations may also arise due to node reboots caused by software bugs [7] or brown, and black-outs. Depending on the severity, nodes may be disconnected for only a few minutes (*e.g.*, during a reboot) or several months (*e.g.*, until the covering snow melts).

Since the system design can generally not prevent external influences from causing a node to be disconnected, it needs to prepare for such situations by taking into account the consequences. For example, if a node detects that it can no longer communicate with its parent, the routing or topology control protocol first needs to commence searching for an alternative parent among the node's neighbors. If this fails, and the node discovers itself completely disconnected, it then needs to store the precious sensor readings locally (*e.g.*, on a SD card) to prevent packet queue overflows. The size of this memory must be large enough to accommodate all data generated during a disconnected phase, defined by the requirements specification (in our case six months). Given the sampling rate and the size of each sample, one can calculate the minimum memory size needed to cater for reliable data logging during phases of disconnected operation.

In general, long term disconnections lead to large deviations of absolute local clock readings which poses additional difficulties to correctly time-stamp and to correctly order sensor data. For example, packets are not timestamped at the source node in X-SENSE. Instead, the packet dwell times at intermediate nodes along the routing path to the sink are accumulated as the packet travels through the network. Using the total traveling time and the time of reception, the sink computes the packet's generation time at the source node. These accumulated time values are subject ot large absolute inaccuracies in case of long term disconnections. In addition, if a node gets disconnected for several months, the dwell times of packets held by that node may exceed the maximum value reserved in the packet header. In such a case, packets that are stored longer cause overflows, and the sink is no longer able to determine the packet generation time.

D. Transition from Lab to Deployment Site

To verify correct system behavior during disconnects, we regularly simulate isolated nodes in our lab, as discussed in Section III-B. Additionally, we have installed an outdoor testbed of sensor nodes on the roof of the university building, exposing the WSN to all kinds of weather phenomena, approaching the conditions at the deployment site. The impact of extreme temperature variations not experienced in our outdoor testbed are examined with climate chamber experiments (see Section III-A).

However, despite the efforts to replay the system interactions on the deployment site in our lab environment, the transition from one to the other is still a challenging task. For example, a disconnected scenario on one of our deployment site, caused by radio interference, went undetected until we discovered that we no longer received any data from the network nodes. Analysis revealed that, since nodes were temporarily unable to communicate with the base station, the output power of the Global System for Mobile Communication (GSM) antenna was set too high for the nodes to communicate with one another, or the base station via the low-power radio. After disabling the GSM, the nodes resumed delivery of data. This unfortunate interplay was not detected prior to deployment only because we did not test the full system beforehand; while the WSN-related components had been tested individually, the seamless coexistence with GSM had not. Consequently, we decided to always test the full system by enhancing our local rooftop and indoor testbeds to replicate our deployments as close as possible.

IV. DATA ACQUISITION AND MANAGEMENT

Artifacts on data retrieved from sensor nodes are common in WSNs [1], *e.g.*, due to the intrinsic unreliability of wireless communication. Concerns such as clock drift, incorrect temporal system behavior, packet duplication, and packet loss decrease the overall quality of the sensor data. In Section III, we discuss how harsh environments aggravate these issues, and how reliable system operation can be achieved by proper design and testing. However, the data retrieved from the network will still contain artifacts. Meeting qualitative and quantitative data requirements therefore becomes a challenging task. Domain experts ask for data that 1) can be associated with a unique origin, 2) provide a reliable acquisition time, 3) come from a properly operating sensor, and 4) are in chronological order with respect to acquisition time.

In X-SENSE, we have to deal with data that originate from three distinct classes of sensors: weather stations, visual sensors, and geophysical sensors. These classes generate different data streams that vary in several orders of magnitude with respect to the size of a sample, the sampling rate, and the

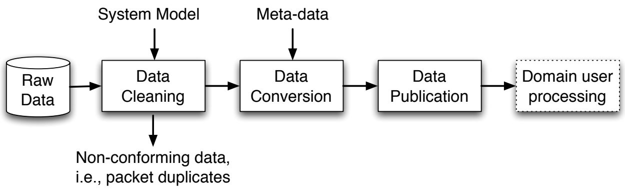


Figure 6. Collected raw data is firstly cleaned by involving a formal model of the data generation and transmission system. Conversion functions, calibration data and other meta-data are then used for converting raw readings to SI values.

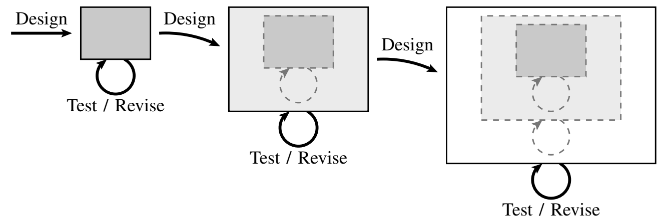
sampling window. For instance, the size of one high-resolution image taken once per hour is in the order of 10 MB, whereas a sensor node only generates five packets every two minutes, totaling in a few tens of bytes.

Before domain experts can analyze the data, extensive cleaning, conversion, and mapping operations need to take place. On a high level, data handling consists of four tasks: data collection comprises all steps for transporting raw output data from a sensing system to the backend database server. The data stream is then cleaned from artifacts introduced by the transmission system, *i.e.*, packet duplicates, erroneous readings, etc. Third, raw values, *e.g.*, ADC readings, are converted into physical units. Converted and cleaned data with verified and known properties are published on the public data front-end (http://data.permasense.ch/). Domain experts are then able to use this publicly available data and apply further filtering and aggregation steps that suit their applications.

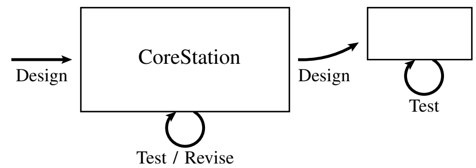
The goal of data cleaning is to generate a stream that contains duplicate-free data that are in the correct chronological order of data acquisition and contain information about the acquisition time, e.g. a time interval that provides safe bounds. Additionally, data that does not conform to system specifications is removed. We achieve this cleaning by analyzing application packet headers, such as sequencing information and dwell time in the network. A formal model allows us to verify packet streams, remove packet duplicates, and reconstruct the correct chronological order.

In addition, a large amount of meta-data is needed to allow tracing and converting sensor data. This includes conversion functions, calibration data, mappings of analog channels, unique device identifiers, and other information. Using this meta-data, collected data is cleaned in multiple dimensions. A stream-based data backend allows the repetition of conversion and filtering steps multiple times by applying filters and algorithms on raw data. Raw sensor data is of high value and never removed or modified; the results of the conversion process are stored in databases.

Gathering data under the extreme influences of the environment may be hard, but handling and managing data requires just as much attention. Having collected about 100 million data samples from the WSN deployments, even mundane problems such as server performance and storage issues arise. Experience with real-world deployments have shown that a reliable and scalable back-end is extremely important to sustain a level of quality that enables geoscientists to derive or validate geological models, and hence provide insight into high-alpine microclimates.



(a) Conventional design approaches (waterfall-model) increase system complexity and feature-sets iteratively (and possibly cyclical).



(b) The proposed approach employs a powerful, generic node (see Figure 8) that provides validation data for system design based on real deployment data.

Figure 7. (a) Conventional design approach, in which a design step represents addition of new features; (b) Proposed design approach, in which superfluous features are removed.

V. Designing for Extremes

In this section, we briefly address the design space of WSNs for environmental monitoring, and discuss how we envision its exploration. The ambitious long-term goal is to devise systematic design principles, applicable to system development for diverse application scenarios that require highest possible data quality and yield, while maintaining system controllability, and observability at lowest possible cost. Because of the early stage, and space considerations, we will only be able to point into the direction of our investigations.

Conventional Design Flow. The prevalent design paradigm followed by many WSN research communities has a strong computer science background. On a high level, this approach consists of incremental, iterative design stages that transform a specification into a marketable product (see Figure 7(a)). However, we have observed that this approach may not be optimal for hardware/software co-design with many, highly stochastic parameters because of three main factors. First, missing system parameter specifications due to insufficient knowledge of the target environment require assumptions that may not apply, and therefore lead to complicated and costly re-design phases. Second, continuously extending a system – if at all possible – is complicated and likely increases development cost while reducing system efficiency. Finally, limited visibility into embedded systems severely complicates debugging and design verification, making the design stages prone to errors. Therefore, a top-down (in terms of complexity) design process (see Figure 7(b)), versus the conventional approach, may lead to more efficient design flows [10] that allow cost-efficient design of optimized systems, featuring only the functionality needed to serve the purpose.

Revised Design Flow. Experience with all aspects of operating

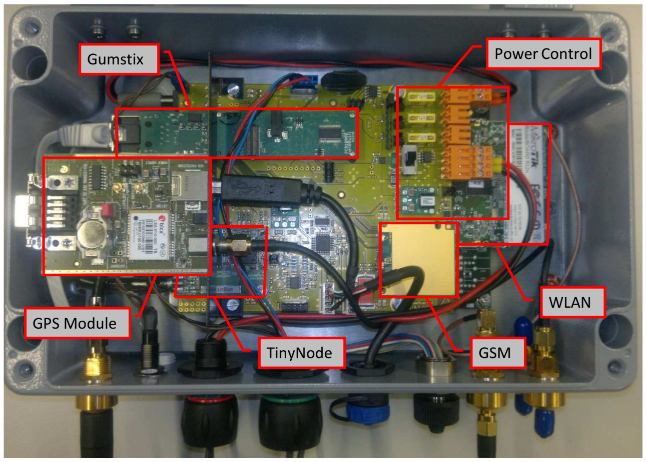


Figure 8. *CoreStation:* The motherboard provides extensive power control, system monitoring, and interface circuitry (*e.g.*, to a GPS module). Interfaces to several communication links (WiFi, GSM modem, TinyNode radio) allow remote observation and control.

heterogeneous WSNs in extreme environments has shown that the deployment site dynamics dominate design choices. To circumvent this complication, and expose remote observability of both the environment and the system under real-world conditions, we argue that as a first step in designing new nodes, few, but feature-rich nodes should be deployed. Such an over-provisioned test-deployment then allows to observe, experiment, and learn on-site. Not only does this give the possibility to draw from actual deployment experience and clearly, and close to exhaustively, define the design possibilities within the application's requirement specification, but also permits profiling of system performance before the first application-specific prototype is even built. Hence, continuous hardware refinement, as is necessary with the conventional approach, can be largely omitted. The system's "unlimited" resources provide the researchers with experimentation opportunities beyond what is possible in artificial lab, or even outdoor testbed simulations. Having such an on-site testbed can bring forth insights into diverse environmental interactions with the equipment that could otherwise only be provisioned for with approximations. Leveraging the knowledge gained by domain experts and system designers over an exploration period could lead to fewer optimization steps in which experience directs design choices (see Figure 7(b)).

To facilitate such an investigative approach, we use an over-provisioned "node", a tried and tested, highly flexible *CoreStation*, of which a block-diagram is shown in Figure 8. This platform, together with a user configurable software framework, permits design space exploration of low-power network nodes, specialized XL sensors (GPS, Camera, etc.), or base stations under real-world conditions.

Our over-provisioned node of course has its price. The powerful hardware architecture imposes high development, and component cost. Furthermore, the lack of application-specific optimizations also affects the power budget; despite extensive software control over power-hungry components, the energy requirements of our *CoreStation* are orders of magnitude higher when compared to standard low-power nodes. This implies that component duty-cycling, energy scavenging, and appropriately dimensioned buffers are an absolute necessity. Finally, due to increased size, weight, and energy requirements of the system, deployment cost increases correspondingly. However, the main challenge of our proposed approach lies in substantial redesign (and related verification) to map resource and software to the final target platform. Despite the overhead, and potential difficulties arising in the translation process, initial experimentation with specialized, resource intensive sensors have produced insights that can facilitate definition of design principles and system models for new and improved sensor node architectures that permit sensing in extreme environments.

VI. Conclusions

More than a decade ago, researchers envisioned "smart dust" as networks of numerous tiny, cheap sensor nodes [6]. We observe instead that applications such as early warning systems of natural hazards (*e.g.*, rockfalls or landslides) require feature-rich, dependable node platforms, delivering correct and reliable data over long periods. Our experience stems from developing WSN systems for environmental monitoring under extreme conditions. To cater for the arising challenges, we propose a novel design approach that targets highly optimized, reliable systems based on extensive on-site design exploration.

References

[1] G. Barrenetxea, F. Ingelrest, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "SensorScope: Out-of-the-box environmental monitoring," in *ACM/IEEE IPSN*, 2008.

[2] J. Beutel, S. Gruber, A. Hasler, R. Lim, A. Meier, C. Plessl, I. Talzi, L. Thiele, C. Tschudin, M. Woehrle, and M. Yuecel, "PermaDAQ: A scientific instrument for precision sensing and data recovery under extreme conditions," in *ACM/IEEE IPSN*, 2009.

[3] N. Burri, P. von Rickenbach, and R. Wattenhofer, "Dozer: ultra-low power data gathering in sensor networks," in *ACM/IEEE IPSN*, 2007.

[4] H. Dubois-Ferriere, R. Meier, L. Fabre, and P. Metrailler, "TinyNode: A comprehensive platform for wireless sensor network applications," in *ACM/IEEE IPSN*, 2006.

[5] A. Hasler, I. Talzi, J. Beutel, C. Tschudin, and S. Gruber, "Wireless sensor networks in permafrost research - concept, requirements, implementation and challenges," in *NICOP*, 2008.

[6] J. M. Kahn, R. H. Katz, and K. S. J. Pister, "Next century challenges: Mobile networking for smart dust," in *ACM/IEEE MobiCom*, 1999.

[7] M. Keller, J. Beutel, A. Meier, R. Lim, and L. Thiele, "Learning from sensor network data," in *ACM SenSys*, 2009.

[8] K. Martinez, J. K. Hart, and R. Ong, "Environmental sensor networks," *IEEE Computer*, vol. 37, no. 8, pp. 50–56, August 2004.

[9] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler, "An analysis of a large scale habitat monitoring application," in *ACM SenSys*, 2004.

[10] A. Vonivento, L. P. Carloni, and A. Sangiovanni-Vincentelli, "Platform based design for wireless sensor networks," in *Springer Mobile Networks and Applications*, 2006.

[11] M. Woehrle, J. Beutel, R. Lim, A. Meier, L. Thiele, C. Walser, and M. Yuecel, "Poster abstract: The FlockLab testbed architecture," in *ACM SenSys*, 2009.

[12] M. Woehrle, K. Lampka, and L. Thiele, "Exploiting timed automata for conformance testing of power measurements," in *FORMATS*, 2009.

[13] M. Woehrle, C. Plessl, J. Beutel, and L. Thiele, "Increasing the reliability of wireless sensor networks with a distributed testing framework," in *EmNets*, 2007.

[14] J. Zhao and R. Govindan, "Understanding packet delivery performance in dense wireless sensor networks," in *ACM SenSys*, 2003.