A power line communication stack for metering, SCADA and large-scale domotic applications

Filipe Pacheco[1], Maksim Lobashov[2], Miguel Pinho[3], Gerhard Pratl[4]
Instituto Superior de Engenharia do Porto
Porto, Portugal
{[1]ffp|[3]lpinho}@dei.isep.ipp.pt

Vienna University of Technology
Institute of Computer Technology
Vienna, Austria
{[2]lobashov|[4]pratl}@ict.tuwien.ac.at

Abstract—**This paper describes the communication stack of the REMPLI system: a structure using power-lines and IP-based networks for communication, for data acquisition and control of energy distribution and consumption. It is furthermore prepared to use alternative communication media like GSM or analog modem connections. The REMPLI system provides communication service for existing applications, namely automated meter reading, energy billing and domotic applications. The communication stack, consisting of physical, network, transport, and application layer is described as well as the communication services provided by the system. We show how the peculiarities of the power-line communication influence the design of the communication stack, by introducing requirements to efficiently use the limited bandwidth, optimize traffic and implement fair use of the communication medium for the extensive communication partners.**

Keywords-Power line communication, remote metering, SCADA, communication protocol stack, communication architecture, bandwidth sharing.

I. INTRODUCTION

The REMPLI project aims at creating a communication infrastructure that is suitable for data acquisition (e.g. reading of utility meters in private households), control of metering and SCADA devices and domotic applications. REMPLI is based on a power-line communication system (PLC), which is designed within the project. We focus on low-bandwidth and long-distance data transmission over low-voltage (LV) and medium-voltage (MV) power lines with redundant and alternating communication paths (meshed and switched power line). The upper level of REMPLI network is an IP (Internet Protocol) backbone, which connects PLC segments to metering, SCADA, and domotic systems.

This paper presents the architecture of PLC communication, including network, transport and application layers with focus on the specific solutions for a large-scale deployment. A detailed description of the REMPLI system can be found in [1] and [2].

II. REMPLI COMMUNICATION

The REMPLI PLC network spans from private apartments to a secondary (MV/LV) transformer station and then further to a primary (MV/HV) transformer station. These two segments are coupled using PLC *Bridges* (we write the terms *Bridge*, *Node*, and *Access Point* starting with a capital letter, since these terms refer to the components that are designed in the REMPLI project). From the application point of view Bridges are transparent: they forward data up- and downwards. *Nodes*, installed in households (or any other energy consumption location), communicate directly to the local metering and SCADA equipment. Another device, the *Access Point*, terminates communication over power line at the primary transformer station. The upper side of the Access Point is connected directly to the IP backbone. Application servers at the utilities communicate to Access Point(s), typically, in a request-response mode. Access Points send these requests via PLC, over Bridges if necessary, to Nodes. Responses from Nodes are transported back to the servers. The system also supports non-request/response communication: Nodes can generate alarm messages, which are delivered to an appropriate server without explicit request.

The REMPLI project does not attempt to develop new software for utilities. Instead, the REMPLI system transparently tunnels protocols that are understood by application servers and metering/SCADA devices. The majority of metering and SCADA protocols are not session-oriented, meaning that one request results in one response without any prior connection setup.

The Communication stack at both Nodes and Access Points is conceptually symmetric and includes the following layers (Figure 1):

The **physical layer** implements power line medium coupling. Design of the physical layer is outside the scope of this paper.

The **network layer** implements a master/slave time division network with basic error recovery and short-distance routing mechanisms, and is responsible for maintaining information about whether a Node is currently active in the network ("logged in") or not.

The **transport layer** supplements the network layer with inter-network routing, fragmentation, request/response pairing, address translation and reverse-channel (Node to Access Point) alarm service. The current transport layer that uses PLC communication may be replaced by new

This work is being partly supported by the EC within project REMPLI (NNE5-2001-00825) and by FCT (CISTER UI 608)

implementations supporting other communication mediums like GSM links, phone line modem, etc…

Finally, the **application layer** carries out two tasks:

- conversion of metering/SCADA protocols into a form that is suitable for transmission over PLC (including reverse process on the other side);
- multiplexing of data from different application servers and their respective sets of equipment over the same PLC channel.

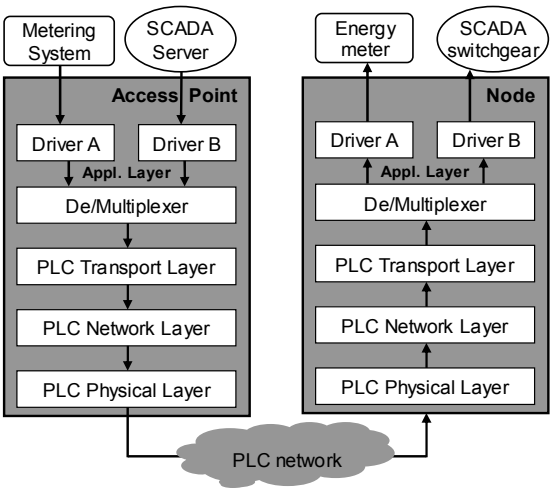

Figure 1. PLC communication stack

The communication stack at the Bridge is similar. However, in case no SCADA or metering devices are installed at the MV/LV transformer, the application layer is not used by utility applications. MV and LV parts of the Bridge share the same transport layer, which performs bi-directional data forwarding.

III. Network and Transport Layers

These two layers provide communication functionality to the Application Layer. Services available at the Access Point side include: the Send Unicast Request with or without response, the Retrieve Alarm Message, the Retrieve Status Service and the Node Login/Logout Notification.

Services available at the Node side include: the Receive Request/Send Response, the Send Alarm Message, the Set Status Service and the Access Point Login/Logout Notification.

Communication services in each PLC network level are handled by the Network Layer, the Transport Layer takes care of end-to-end communication (see Figure 2).

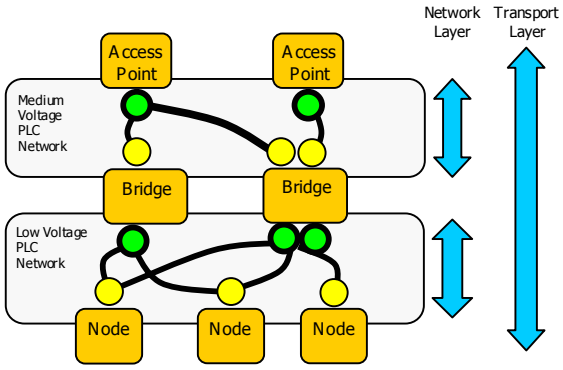

Figure 2. Network Layer and Transport Layer communication services

The Network Layer uses a master-slave time slot scheme and multiple masters may share the medium either using different slots in each cycle or using separate frequency bands. The minimum time slot cycle is 3 slots and typically 2 master networks sharing the same frequency band will use 4 slots cycles: 2 for each master (see Figure 3). The drawback of multi-master networks using different frequency bands is that a slave cannot be connected to several masters at the same time in this situation. In this case a slave can "switch" masters when needed but this process will take considerable time.

The "connection" process is fully automated by the Network Layer: each master device has a list of authorized slaves (using unique serial numbers) and will periodically scan for new devices. The authorized slaves will connect to the designated master or masters and will automatically adjust themselves to the masters' frequency bands and other communications parameters. If a slave detects a lost connection to all its masters it will try to connect to other masters scanning all the frequency bands.

This solution enables easy setup of very large networks simply configuring parameters on the easily accessible (and small number) Access Points. It also automates the installation of new Nodes or Bridges: the system manager simply has to configure the new serial numbers in the Access Point and after a while the new Node or Bridge will be available on the network.

In terms of data communication services the Network Layer implements not only basic "unconfirmed send" and "send confirmed" services but also an automatic poll system (not only to check slave connectivity, but also to find new devices) and a slave-based repeater mechanism to increase the coverage of the master/slave network (at the expense of additional delays). Optionally it can be configured to do automatic retries on the confirmed service exchanges.

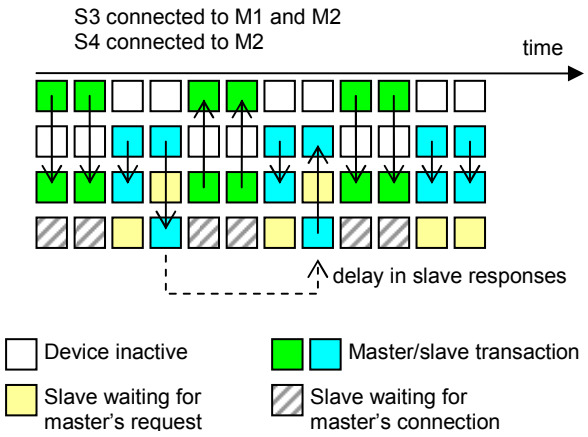

Figure 3. Multiple master network

The timings of the Network Layer services are handled by an internal dispatcher service. The dispatcher manages the traffic from the 3 different priorities (only 2 on slaves) queues, the polling cycles and additional confirmed requests to retrieve data from slaves that have pending information on their queues.

Access Points act as masters in the network; Nodes are slaves. Bridges have a Slave Network Unit at the Medium Voltage (MV) side and a Master Network Unit at the Low Voltage (LV) side.

The Transport Layer extends the fixed-size master/slave packet service of the Network Layer to the Request/Response end-to-end services required by the Applications: for this task it will fragment the data at the source and rebuild it at the destination. Since the MV PLC and the LV PLC may have different frame sizes, Bridges have to store and join/split fragments as needed.

Fragmentation in this system must be aware of two limiting factors in terms of header data:

- *the limited data payload available in each frame.* This implies that absolute fragment numbering schemes are not acceptable and an offset based system is implemented.
- *the high error rate that may occur in certain periods of time.* This implies that any fragment identification method must be robust enough to survive the loss of several fragments.

To solve these issues our basic implementation uses 6 bits for fragment identification and 6 bits for packet identification. Depending on the final field conditions this numbers may be adapted to a particular deployment.

The system must also handle huge packets in-transit on the Bridges. Since several large packets cannot be stored in the limited Bridge memory we use temporary buffers to store fragments as they are received, and try to forward the data as soon as possible. As soon as the forwarded data is delivered the temporary buffers are discarded. There is also a system to confirm correct delivery of the fragments of large packets even if the original application service is non-confirmed. When a delivery failure is detected the rest of the packet is discarded at the Bridge and this information is forwarded to the origin.

The Transport Layer also does automatic address translation and route discovery using as little information as possible after the basic device information is exchanged. When a Network level connection is established, the Transport Layer is informed, so that it can build a map of available paths (and respective Network Layer addresses) from each Access Point to each Node. The map of available paths also includes information about each link quality (delay/error rate).

The starting point for this process is a table configured in each Access Point with information about each device: node address (used by application drivers), unique serial number and Bridges that can be used to reach the device. Then the Access Point Transport Layer transfers the unique serial numbers to the local Master Network Layer. When a Bridge connection is detected the Access Point Transport Layer sends special packets with the list of connection data for the nodes that are authorized for that particular Bridge. The Bridge Transport Layer transfers this information to its local Master Network Layer. When a device is connected do a Bridge, the Bridge Transport Layer first allocates a Bridge ID to the new device and sends this Bridge ID and the node Unique Serial Number to the Access Point. When the Access Point wants to send information to the node it will include the Bridge ID on the Transport Layer header. The Bridge Transport Layer forwards periodically the table with link quality of the nodes in a particular Bridge to all the connected Access Points so the latter can have an estimation of current network link status.

Some of these services depend on reliable delivery of the Transport Layer's own data packets. The Transport Layer has special internal services that ensure trustworthy transfers of data not only from Masters to Slaves but also on the reverse direction.

Finally, the Transport Layer must do its best to guarantee a fair use of the network paths and to enforce application requests priorities in a meshed dynamic network with variable link capabilities and requests. The links change not only due to physical reconfiguration of the network (power distribution switching devices) but also due to external electromagnetic interference (e.g. motors, welding machines, etc). This task is accomplished using an internal scheduler that analyzes the current requests and the available path map (see Figure 4).

The Transport Layer handles redundancy (see Figure 5) for a particular Access Point – Node data transfer. When a packet is sent from an Access Point the Transport Layer on the source checks the available paths and selects the most appropriate for the given moment. To limit the complexity of the system all the fragments of a request (and the

possible response) follow the same route. This has the additional benefit that for most requests the routing calculations are done in the Access Points, the devices with more computing resources in the system.

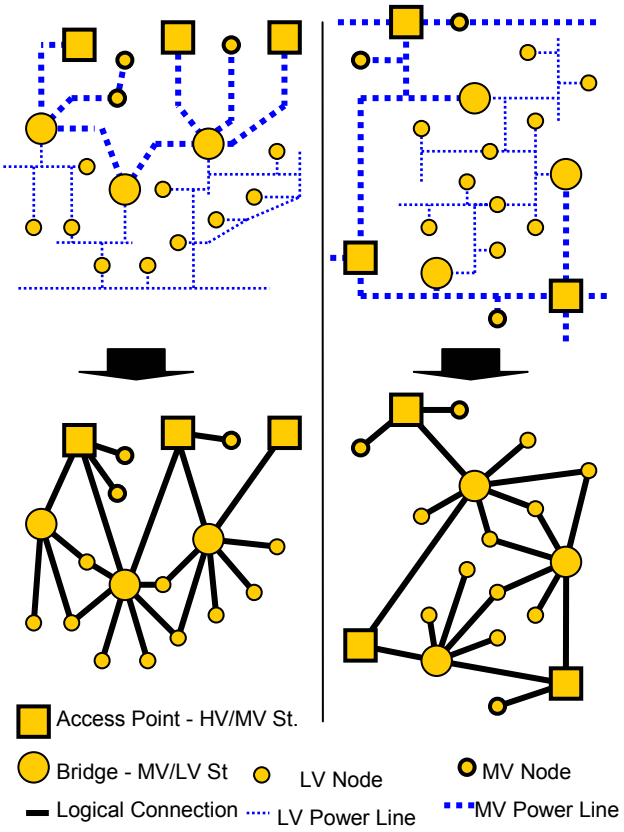

Figure 4. Examples of geographical network layout with power and logical connections

In terms of routing packets from the Access Point to a Node are processed by Bridges simply reading the routing information from the transport layer header and replacing it with backtrack information if needed (for requests with response). To limit the usage of data in the transport layer header all this routing information is used using (low-bit length) local indexes that are only valid for a particular master/slave combination.

The application priorities processing algorithm is configurable and a system may have strict priority serving (i.e. higher priority queues are completely served before lower priority queues) or have a minimum bandwidth share for each priority class (i.e. higher priority queues are served until they are empty or lower priority queues with guaranteed bandwidth are not serviced yet). If needed the Transport Layer's application priority system may be completely overridden by applications and requests are mapped directly to the Network Layer's priorities (2 priorities in the slaves, 3 priorities in the masters). Since Bridges must also handle packed priorities (as well as Node responses), the priority information must be included in the Transport Layer header of most packets.

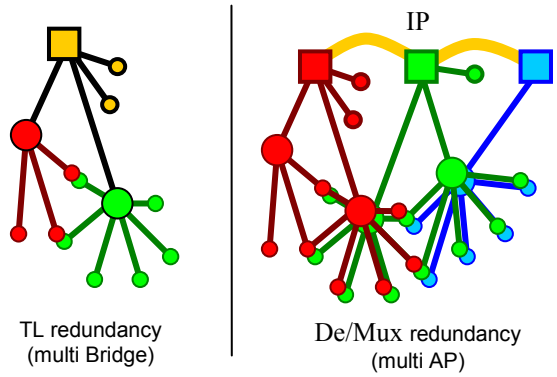

Figure 5. Transport Layer redundancy and Driver De/Mux redundancy

When it comes to spontaneous information from the Nodes, there are two alternative services each with its own characteristics. The Status Services is a low-bandwidth process that enables Nodes to update a local status multi-bit field and this information will be transmitted over the network to all the Access Points to whom the Node is connected. This service is a non-confirmed service and uses very few network resources. For critical Node to Access Point information transfer the Alarm Service guarantees fast delivery of the data to at least one Access Point. This service uses a lot of network resources and so it should be used only in exceptional situations.

All this Transport Layer features combined results in services for the Application Layer that provide:

- Unconfirmed request service and Request/Response service that map easily to several application scenarios
- Node-to-Access Point Alarm service with guaranteed delivery and high priority
- A (small byte count) Status Service that provides a simple and effective way to keep track of Node status
- Variable length data transfer services, exceeding the memory capacity of the foreseen devices
- Low network overheads for most used requests
- Large network support: up to a thousand Nodes per Bridge, hundreds of Bridges per Access Point, and thousands of Nodes per Access Point
- Plug&Play operation: Nodes can simply be connected to the network and the Transport Layer will establish a connection to the authorized devices

IV. Application Layer

The application layer, available at both Node and Access Point (and Bridges, if required), consists of:

- De/multiplexer (one component at every entity);

- protocol drivers (one driver per protocol, both at Node and Access Point side).

Protocol Drivers at the Access Points interface with utility software over IP. At the Node side protocol drivers implement the interface to the equipment, namely meters and SCADA devices. There is one driver per protocol; multiple homogenous devices, such as M-Bus meters, can be connected to the same driver. Depending on the application, different standardized protocols are implemented (using Drivers), namely IEC 60870 series, EN 62056 (also known as IEC1107), and EN 1434-3 commonly known as M-Bus.

The De/Multiplexer merges streams of messages, received from different drivers (these are requests from servers, responses from equipment and internal traffic between the drivers). The multiplexed stream is transmitted to a receiver device, where the De/Multiplexer distributes it, so that a target driver receives only the data that was designated to it. This requires transmitting additional piece of information (1-byte destination driver address) in every packet, which allows for up to 255 sender/recipient drivers at every node and Access Point. This concept is somewhat similar to port numbers in the TCP/IP suite.

In the REMPLI system drivers, implementing tunneling of a particular protocol, talk strictly to each other. For example, all drivers for an IEC 60870 protocol may be assigned the same address (say, 10) on all Access Points and Nodes. This means, that communication always occurs between *pairs* of drivers: a driver at the Access Point sends packets to its siblings at the Nodes; responses are delivered back to the same address. In fact, apart from sharing PLC bandwidth, different protocol drivers do not influence each other at all; communication between pairs is transparently multiplexed onto the same medium without "intervention" from drivers themselves.

Apart from multiplexing communication, the De/Multiplexer at the Access Point implements one more function: if an Access Point cannot reach target Node, the De/Multiplexer can re-route request from a driver (on behalf of an application server) through another Access Point. Re-routing is done over the IP backbone (see Figure 5), which links all Access Points to the application servers and "horizontally", between each other.

In a switched PLC network, where a given communication path between Access Point and Node pair can terminate at any time, every possible path to the Node should be covered by a separate Access Point (PLC Master). Redundant paths in a meshed PLC are handled by the Transport Layer, as described above. PLC switching is handled by the Application Layer. Application servers and drivers do not need to be aware of duplicate and alternating communication paths. A server can connect over IP to any Access Point. Its requests are re-routed via other Access Points whenever necessary.

In a meshed network special attention is paid to transmitting multicast and broadcast requests. Since a single Access Point may not be able to reach all Nodes, such messages have to be re-routed by De/Multiplexer to several, or even all Access Points, which altogether provide complete coverage of the addressed nodes. Since every Access Point then sends the request over PLC independently, target Nodes can receive duplicate messages. These are filtered out by De/Multiplexers at the Nodes.

Since the De/Multiplexer and the underlying PLC stack isolate communication between different driver pairs, the system is easily extendable to support tunneling of future protocols that are not anticipated at the project design stage.

V. CONCLUSIONS

This paper presented the communication stack used in the REMPLI system, which provides communication service for existing utilities applications. This paper focuses on the architecture of the network, transport and application layers with focus on the specific solutions for large-scale systems. We presented how the design of the communication stack is influenced by the particular requirements of the power-line communication, particularly to provide efficiency in the use of the available bandwidth, optimizing traffic and guaranteeing fair use of the communication medium.

VI. REFERENCES

[1] A. Treytl, T. Sauter, G. Bumiller. "Real-time energy management over power-lines and Internet"; *Proceedings of the 8th International Symposium on Power-Line Communications and its Applications*, vol. ISPLC'04 no. 8, March 2004, pp. 306-411.

[2] G. Pratl, M. Lobashov. "Remote Access to Power-Line Networked Nodes: Digging the Tunnel"; WFCS 2004 *5th International Workshop on Factory Communication Systems, Vienna, Austria*; Sept. 2004; p. 323-326.