ALERT-2
Requirements Specification

Version 1.0
January 16, 2008

Timothy J. Salo
Salo IT Solutions, Inc.
1313 5th Street SE
Minneapolis, MN 55414-4504
salo <at> saloits <dot> com
612-605-6896

This material is based upon work supported by the Department of Commerce under contract number DG133R-07-CN-0175. Any opinions, findings, conclusions or recommendations expressed in this publication are those of the author and do not necessarily reflect the views of the Department of Commerce.

Contents

1. Overview

This document identifies the requirements for the ALERT-2 protocols, a suite of next-generation wireless communication protocols for use in automated flood warning systems (AFWS). The ALERT-2 protocols will provided an enhanced alternative to the original ALERT and IFLOWS protocols (although many networks will continue to use these legacy protocols for the foreseeable future). These new wireless communications protocols will offer increased functionality and will facilitate the development of new capabilities for automated flood warning systems. They will use the new modem technology that has recently been developed by Blue Water Design LLC, but can easily be adapted to use other transmission media.

This document focuses on "user requirements", high-level requirements that specify the services that the ALERT-2 protocols should provide or other externally visible behaviors that the ALERT-2 protocols should exhibit. The requirements documented here were culled from numerous sources:

- Slides, notes, and discussions from the October 25, 2006 session “ALERT into the Future” moderated by Ilse Gayl, which was part of the 2006 Southwestern Association of ALERT Systems (SAAS) conference, which was held in Overland Park, Kansas October 23 – 25, 2006[1];

- Sessions from the 2006 SAAS conference and 2007 National Hydrologic Warning Council (NHWC) conference, which was held June 11 – June 14, 2007 in Savannah, Georgia;

- Proceedings from other SAAS, NHWC, and Alert Users Group (AUG) conferences;

- Archives of the Yahoo! Flood Warning and Flood Warning Systems (Floodsystems) group; and

- Meetings, phone calls, e-mails, and conversations with members of the automated flood warning systems community, including flood warning system vendors, flood warning system operators, users of flood warning information, and Federal agency officials.

To the best of the author’s understanding, and with some qualifications, the requirements for the ALERT-2 protocols documented here reflect the consensus of the AFWS community. Dozens of copies of the first version of this document were downloaded; many of these copies were presumably read or at least skimmed. Several written review comments were received. The author met with or called numerous members of the AFWS community to discuss their perspectives on the material presented here. Most of the people with whom the author spoke agreed with most of the user requirements included in this document. However, while many had strong opinions about a few requirements (e.g., the ability to manage an AFWS system over the network) they often didn’t have similarly strong feelings about *how* those requirements should be met or about what technologies should be employed to provide the desired capabilities. Also,

[1] Notes taken during this session by Ilse Gayl and Timothy J. Salo are included as Appendix A and Appendix B, respectively, of this document.

there are several people who have been active in the long-standing effort to develop a successor to the original ALERT protocol who appear to disagree with the general thrust of this document. Specifically, these people appear to believe that the next-generation AFWS protocol should be a modestly enhanced version of the original ALERT protocol, even if this approach doesn't meet many of the users' requirements documented here.

This document is available on the ALERT-2 Protocol Development project Web site (http://www.alert-2.com/). Please send any questions, comments, or suggestions you might have about this material to the author.

This document is a product of the *ALERT-2 Protocol Development* project, an SBIR Phase I contract awarded to Salo IT Solutions, Inc. (SaloITS) by the National Oceanic and Atmospheric Administration (NOAA). Of course, any opinions, findings, conclusions or recommendations expressed in this publication are those of the author and do not necessarily reflect the views of NOAA or of the Department of Commerce.

2. The Original ALERT and IFLOWS Protocols

The ALERT protocol and IFLOWS protocol are wireless communications protocols used in automated flood warning systems. Both protocols were developed by the National Weather Service in the 1970s, the ALERT (Automated Local Evaluation in Real Time) protocol on the West Coast [Burnash 1983], [Burnash 1984], [Clark 1983] and the IFLOWS (Integrated FLood Observing and Warning System) protocol on the East Coast [Scawthorn]. The objective of the ALERT and IFLOWS protocols is to provide real-time data for automated flood warning systems.

Automated flood warning systems have traditionally relied on real-time rainfall and river level sensors to provide data for flood forecasting models. Figure 1 illustrates the components and configuration of a typical ALERT or IFLOWS network. Remote nodes include one or more sensors, often a rainfall sensor (e.g., a tipping bucket rain gauge) and/or a river level sensor (which employ a variety of technologies). The remote nodes send sensor data to a base station using radio frequency (RF) transmissions. Generally, sensor data are transmitted as they become available (e.g., in response to a bucket tip). Repeaters are used to extend the geographic extent of the network when remote nodes are out of direct radio range of the base station.

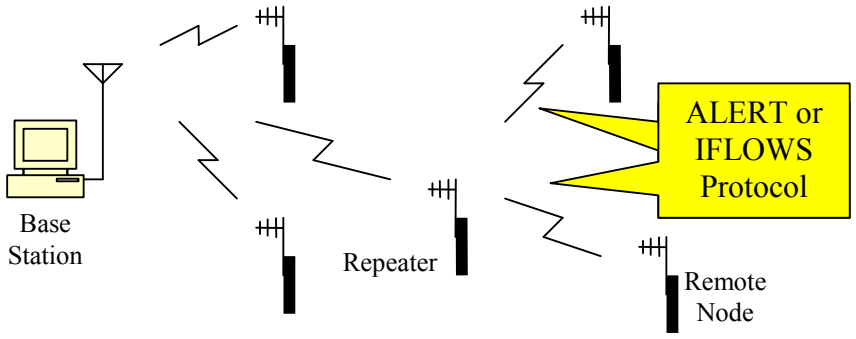


Figure 1. ALERT Network Components.

Most variants of the ALERT and IFLOWS protocols employ a four-byte (32-bit) packet that is transmitted asynchronously at 300 bits-per-second (bps) using frequency-shift keying (FSK) modulation. The format of the five most common variants (ASCII, Binary, Enhanced ALERT and Enhanced IFLOWS) are shown in Figure 2 below [Anonymous], [HydroLynx], [National Weather Service]. Several less-well-documented, vendor-specific variants have also been developed.

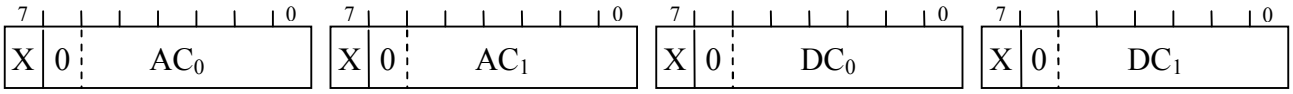
The original ALERT/IFLOWS protocol, the ASCII message format, is shown in Figure 2a. This message format is simply four decimal digits encoded in ASCII; the first two characters identify the sensor and the second two characters are the sensor data value. Receivers ignore the high-order bit of each byte (the ASCII parity bit). Because these ASCII characters are limited to decimal digits (i.e., the characters “0” – “9”, which are encoded in ASCII as 0x30 – 0x39) bit six of each byte is always zero.

The Binary message format, illustrated in Figure 2b, transmits information as binary values, rather than as ASCII character strings. This extends the range of sensor addresses to 0 - 8191 and the range of the sensor data values to 0 – 2047. Receivers can determine whether a message uses the ASCII message format or the Binary message format by examining bit six of each byte (i.e., a zero indicates the ASCII message format is being used, while a one indicates that the Binary message format is being used).

A Wind message format is also defined, and is shown in Figure 2c. Receivers can identify messages that use this format because the two high-order bits of the first two bytes are 01, while the two high-order bits of the second two bytes are 11.

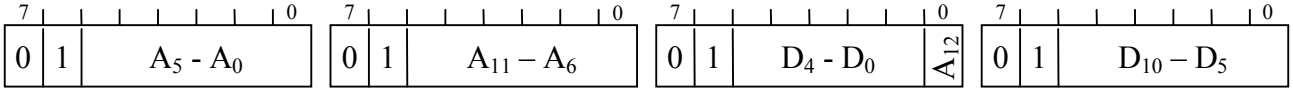
Two enhanced binary formants have been specified; both contain a six-bit cyclical redundancy check (CRC) field that detects many transmission errors. The Enhanced ALERT message format reduces the size of the sensor address field to include a one-bit battery status field, while the Enhanced IFLOWS message format does not. Receivers can identify messages that use one of the Enhanced message formats because the two high-order bits of the first byte are 11. Apparently, receivers have to be preconfigured to know whether a particular remote station is using the Enhanced ALERT message format or the Enhanced IFLOWS message format. Additionally, [Anonymous] states that the Enhanced IFLOWS message format "requires that messages from sensors with an address below 100 conform to the [ASCII message format] rule that data values range from 0 – 99." It goes on to say that the Enhanced ALERT message format "allows wind sensors to substitute gust information for the CRC bits." Receivers must be preconfigured to know whether a particular remote station is using this field for a CRC or for wind gust information.

Additional message formats have been defined. See, for example, [Slouber] and [Futuretech].



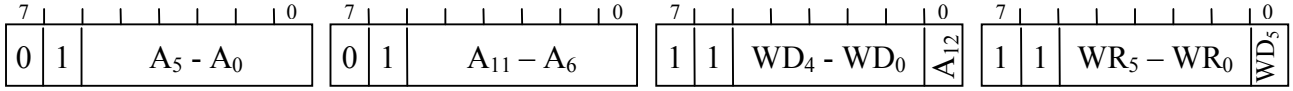
X – Ignored on receive (ASCII parity bit)
AC_0, AC_1 – Source address character 0 and 1 (low-order digit, high-order digit)
DC_0, DC_1 – Sensor data character 0 and 1 (low-order digit, high-order digit)

Figure 2a. ASCII Message Format.



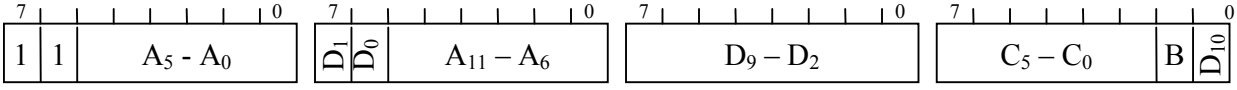
A_{12}-A_0 – Source address (13 bits)
D_{10}-D_0 – Sensor data (10 bits)

Figure 2b. Binary Message Format.



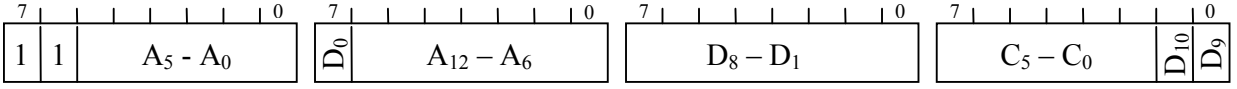
A_{12}-A_0 – Source address (13 bits)
WD_5-WD_0 – Wind direction (6 bits)
WR_5-WR_0 – Wind run (5 bits)

Figure 2c. Wind Message Format.



A_{11}-A_0 – Source address (12 bits)
D_{10}-D_0 – Sensor data (11 bits)
B – Battery Status (1 bit)
C_0-C_5 – CRC (6-bit, polynomial = $X^6 + X^4 + X^3 + 1$)

Figure 2d. Enhanced ALERT Message Format.



A_{12}-A_0 – Source address (13 bits)
D_{10}-D_0 – Sensor data (11 bits)
C_0-C_5 – CRC (6-bit, polynomial = $X^6 + X^4 + X^3 + 1$)

Figure 2e. Enhanced IFLOWS Message Format.

2.1 Limitations of the Original ALERT and IFLOWS Protocols

Numerous limitations of the original ALERT and IFLOWS protocols have been identified, many of which are discussed below. The notes from the October 25, 2006 session "ALERT into the Future" held at the 2006 Southwestern Association of ALERT Systems (SAAS) conference may offer additional insights into the concerns of many ALERT system users [Gayl 2006b], [Salo 2006].

2.1.1 Poor RF Channel Efficiency

The original ALERT protocol makes poor use of the available RF channel capacity [Nelson 2001], [Roark 1999]. It transmits data at 300 bps [Anonymous], [National Weather Service][2], although much higher data rates (e.g., 9,600 bps) are possible on the 12.5 kilo-Hertz (kHz) RF channel that is used [Roark 1999], [Roark 2000], [Roark 2003a], [Roark 2006], [Van Wie 2003]. Furthermore, the original ALERT protocol has a very long preamble, as long as 200 msec, while the transmission of the four-byte ALERT packet requires only an additional 133 msec [Roark 1999], [Roark 2007b]. The new ALERT modem being developed by Blue Water Design LLC transmits data at 4,800 bps, which should provide a significant performance improvement.

2.1.2 High Packet-Loss Rates

Users have persistently complained about high packet-loss rates during major rain events. Van Wie estimated that over 80% of the messages transmitted through one repeater were lost during a major rain event in the Denver area, although a suggested reconfiguration of the network would have reduced this loss rate to 55% [Van Wie 2007]. Another user reported that his network typically successfully transfers only about 80% of the messages that are transmitted by the remote nodes [Salo 2006].

These high packet-loss rates are largely due to collisions, rather than to transmission errors. The original ALERT protocol has no facility to avoid collisions between stations that are competing for simultaneous use of the RF channel. Rather, each node transmits data without regard to any external factors, such as whether another node might be transmitting at the same time. If two or more stations transmit at the same time, both packets are generally corrupted and are usually lost. This strategy is known as the "pure ALOHA protocol". Unfortunately, the maximum throughput of the pure ALOHA protocol is about 18% of the available bandwidth, (i.e., the bandwidth used by the successfully received packets is at most about 18% of the total available bandwidth)[3]. As the rate at which packets are transmitted increases, collisions between packets increase; if the rate at which packets are transmitted is high enough, all packets are lost due to collisions and no packets are successfully received. This is undoubtedly the cause of the large number of lost packets that are experienced in ALERT networks during major rain events.

[2] Note that [National Weather Service] states that the transmission speed can be either 300 or 1,200 bps, although it is not clear that the 1,200 bps speed is used in practice.
[3] See, for example: Tanenbaum, Andres S., *Computer Networks 4e*, Prentice Hall PTR, 2003, pp 251-254.

Because the original ALERT protocol is a pure ALOHA protocol, it has a theoretical maximum throughput of approximately 32 packets per second[4]. The new ALERT modem being developed by Blue Water Design should (in the absence of other protocol changes) increase the maximum theoretical throughput of the original ALERT protocol to approximately 88 packets per second[5] (assuming that the new modem requires 123 msec to transmit a packet [Roark 2007a], [Roark 2007b]). This represents a nearly three-fold performance improvement over the original ALERT protocol.

2.1.3 Poor Error Detection Capabilities

The most common variant of the original ALERT protocol, the binary message format, has extremely limited error detection capabilities. Transmission errors can corrupt a packet by changing the value of one or more bits. However, a receiver is unable to detect most of these errors[6]. As a result, corrupted messages can, and often are, forwarded to the applications. See, for example, [Roark 1999], [Roark 2000], [Roark 2003a], [Roark 2003b], [Roark 2004], and [Slouber]. The new modem being developed by Blue Water Design employs forward error correction (FEC) technology, which should reduce the rate at which corrupted packets are passed to applications.

2.1.4 Limited Address Space

The address field in the original ALERT protocol is 13 bits long, and therefore the protocol supports a maximum of 8191 sensors in a single geographic area. This limit is generally viewed as undesirable. See, for example, [Roark 1999], [Roark 2000], [Roark 2003a], [Roark 2003b], and [Roark 2004].

2.1.5 Limited Sensor Value Range

The sensor data field in the original ALERT protocol is 11 bits long, and so the protocol can transport sensor data values of 0 – 2047. This range is inadequate for some newer, higher-resolution sensors, and is generally viewed as undesirable. See, for example, [Roark 1999], [Roark 2000], [Roark 2003b], and [Roark 2004].

2.1.6 Small, Fixed Message Format

An ALERT message uses four eight-bit bytes. The most common variant of the protocol divides these 32 bits into a 13-bit source address field and an 11-bit sensor data field, plus eight bits of overhead. This small message size makes it extremely difficult to add any new fields to the ALERT message. For additional comments on this topic, see [Roark 2003a], [Roark 2003b], and [Roark 2004]. The error-correcting modem being developed by Blue Water Design enables larger packets to be used with the ALERT-2 protocols.

[4] (1000 ms / (200 + 133) ms/packet) * 60 sec/min * 18% = 32.4 packets/minute
[5] (1000 ms / (123) ms/packet) * 60 sec/min * 18% = 87.8 packets/minute
[6] A receiver can detect some errors, such as illegal values in the two high-order bits of each byte or ill-formed characters, such as those that are missing a stop bit.

2.1.7 Limited Protocol Extensibility

The ALERT protocol contains no mechanisms to permit the protocol to be gracefully extended or evolved [Roark 2003a]. As can be seen from the descriptions of the various ALERT message formats, vendors have introduced new message formats by using various tricks that enable a receiver to [usually] determine which message format is being used. However, even with these tricks, it is effectively impossible to make major extensions to the protocol. The new modem being developed by Blue Water Design offers an opportunity to deploy a new ALERT protocol that includes features that make it possible to easily add new capabilities in the future.

2.1.8 Monolithic Protocol

The ALERT protocol is monolithic, in the sense that it doesn't embody a notion of protocol layering, (e.g., identify the functions that should be performed at each protocol layer and clearly define the interaction between protocol layers) [Roark 1999]. Clean protocol layering has been repeatedly demonstrated to enhance the quality of protocol designs, simplify the implementation of protocols, and facilitate the enhancement and evolution of the protocols.

2.1.9 Integrated Physical Layer

The details of the physical layer (the transmission media) are an integral part of the specification of the original ALERT protocol [Roark 1999]. However, some users have wanted to use alternative transmission media, such as satellite links [Van Wie 2004] or other, existing, available transmission facilities [Allan].

2.1.10 Missing Sensor Data Descriptions

The original ALERT protocol message formats include no information about the type of sensor that originated the data or the units in which the sensor data are expressed. Rather, the sensor type and data units must be manually configured for each sensor in the network [Roark 2003b].

2.1.11 No Two-Way Capability

The original ALERT protocol is one-way protocol, in which the remote stations only transmit data and the base station only receives data. There is no provision for two-way communications, such as enabling the base station to communicate with a remote station. At least one vendor implemented a primitive two-way protocol, although this capability was never reflected in the protocol standard [Slouber].

The desirability of a two-way communications capability, in addition to the existing one-way communications capability, was discussed at the 2006 SAAS session about the future of the ALERT protocol [Salo 2006]. Some capabilities of considerable interest to ALERT system operators, such as the ability to configure nodes remotely or to download log files from remote nodes, require a two-way communications capability.

2.1.12 Nonexistent Network Security Mechanisms

The ALERT protocol contains no security mechanisms that ensure the integrity of received data. In an era of heightened concern about homeland security, critical infrastructure such as environmental monitoring networks ought to include features that protect sensitive data, (even if those features aren't universally deployed or enabled).

2.2 Efforts to Update the ALERT and IFLOWS Protocols

Efforts have been underway for nearly a decade to develop a successor to the original ALERT protocol. A committee was formed in mid-1999 that was presumably charged with developing a next-generation ALERT protocol [Roark 1999]. In a September 2000 note, Chris Roark and Don Van Wie reference a panel discussion about the future of the ALERT protocol that was held at the May 2000 ALERT Users Group meeting [Roark 2000]. In their note, Roark and Van Wie also describe experiments that compared the performance of a 4800 bps data modem with that of the original 300 bps ALERT modem. A detailed report on the feasibility of a higher-speed physical layer protocol was published by Roark and Van Wie in February 2003. The results of this study were presented at the October 2003 meeting of the National Hydrologic Warning Council (NHWC) [Van Wie 2003]. Sessions about efforts to update the ALERT protocol were also presented at the NHWC meeting in October 2003 [Roark 2003] and the Southwestern Association of ALERT Systems in October 2004 [Roark 2004]. In July 2005 NOAA awarded a Phase I Small Business Innovation Research (SBIR) contract to Blue Water Design LLC, a company owned by Chris Roark. The project resulted in a preliminary design for a 4,800 bps modem that included a forward error correction (FEC) capability. A prototype of this modem was recently developed by Blue Water Design LLC under a contract with the ALERT Users Group.

In July 2007 NOAA awarded a Phase I SBIR contract to Salo IT Solutions, Inc. to work on the next generation of the ALERT protocol. This document, the "ALERT-2 Requirements Specification", and the "ALERT-2 Protocol Specification" document are being developed under this contract.

3. ALERT-2 User Requirements

This section identifies the user requirements for the ALERT-2 protocols. These are high-level requirements that specify the services that the ALERT-2 protocols should provide or other externally visible behaviors that the ALERT-2 protocols should exhibit.

3.1 Functionality Requirements

The ALERT-2 protocols must:

- **Provide a real-time, best-effort, datagram (i.e., single packet) service for transmitting data from remote nodes to a base station**. This is the service provided by the original ALERT protocol.

3.2 Performance Requirements

The ALERT-2 protocols should:

- **Provide enhanced throughput.** The ALERT-2 protocols should ensure that a larger number of messages can be transmitted per hour than is possible with the original ALERT protocol. They should ensure that at least TBD messages per hour can be successfully transmitted on a single RF channel. The new modem being developed by Blue Water Design LLC promotes this goal.

- **Ensure better channel utilization.** The ALERT-2 protocols should ensure that channel utilization of at least TBD percent can be achieved, where utilization is measured as the number of bits of link-layer payload data successfully received compared to the raw bandwidth. The new modem being developed by Blue Water Design LLC may promote this goal, at least to some extent.

- **Support larger networks.** The ALERT-2 protocols should support networks that include at least 1023 nodes.

- **Support more sensors.** The ALERT-2 protocols should not limit the number of sensors that an individual node can support or that a network can support (although applications and application protocols may impose limits on the number of sensors that they can support).

- **Ensure minimum latency.** The ALERT-2 protocols should ensure that the network latency for time-critical traffic is less than TBD seconds, as measured between the time that a remote station has data to transmit and the time that those data are received by the base station.

3.3 Reliability Requirements

The ALERT-2 protocols should:

- **Reduce or eliminate packet loss due to collisions.** The ALERT-2 protocols should be able to prevent, at least at certain times or for certain types of data, packets from being lost because more than one node tries to transmit at the same time. The new modem being developed by Blue Water Design LLC will help reduce packet loss due to collisions, but will generally not eliminate this packet loss.

- **Detect and discard packets that contain transmission errors.** The ALERT-2 protocols should prevent damaged packets from being forwarded to applications or otherwise being processed. The new modem being developed by Blue Water Design LLC should achieve this goal.

- **Minimize the number of packets that are lost as a result of transmission errors, collisions, or congestion.** The ALERT-2 protocols should, optionally and when desired by the application, make additional efforts (e.g., retransmitting packets) to ensure that that packets are successfully received.

3.4 Extensibility Requirements

The ALERT-2 protocols should:

- **Not limit the types of sensors that AFWS products can support.**

- **Simplify the implementation of new functionality in AFWS products.** The ALERT-2 protocols should facilitate the development by vendors of new functionality for AFWS products. Additionally, the ALERT-2 protocols should simplify the addition of new features to the ALERT-2 protocol.

In order to simplify the implementation of new products, services or features, the ALERT-2 protocols should:

- **Provide a reliable datagram service**. The network should employ techniques (e.g., acknowledgements and retransmissions) to ensure that messages are successfully received by the destination node. The transmitting node can select whether this feature is used for a particular packet. This service should support the transmission of data between any two nodes in the network.

- **Provide a reliable stream-oriented service.** The network should employ techniques (e.g., sequence numbers, acknowledgements, and retransmissions) that ensure that a sequence of messages is received in the correct order and without loss by the destination node. The transmitting node can select whether this feature is used for a particular set of packets. This service should support the transmission of data between any two nodes in the network.

3.5 Network Administration and Management Requirements

The ALERT-2 protocols should:

- **Reduce the labor required to deploy, configure, upgrade, and manage AFWS networks and systems.** Wherever practical, the ALERT-2 protocols should minimize or even eliminate the need for physical access to remote nodes and the need for manual configuration.

- **Support remote network management.** The ALERT-2 protocols should enable an ALERT-2 network to be managed remotely, typically from a base station. Specifically, the need to physically visit a remote node to manage the network (e.g., to upgrade or reconfigure the software in the remote node or to retrieve a log file) should be eliminated.

- **Permit passive base stations.** The ALERT-2 protocols should enable additional base stations to passively monitor the traffic on an ALERT-2 network.

- **Support automatic base station fail-over.** The ALERT-2 protocols should ensure that an available back-up base station automatically assumes responsibility for an ALERT-2 network, without the need for human intervention, in the event that the primary base station fails.

- **Support multiple, independently administered networks per channel.** The ALERT-2 protocols should permit multiple, independently administered networks to share a single RF channel (where a network is a base station and the remote nodes that forward sensor data towards that base station). The coordination required between the independently administered networks should be minimized.

- **Simplify deployment of new versions of the ALERT-2 Protocol.** It should be possible to deploy new versions of the ALERT-2 protocol incrementally. Specifically, it should be possible to deploy a new version of the ALERT-2 protocols one node at a time in an ALERT-2 network, rather than upgrading all of the nodes at the same time.

3.6 Interoperability and Compatibility Requirements

The ALERT-2 protocols should:

- **Support transmit-only remote nodes.** The ALERT-2 protocols should operate, perhaps with a significant loss of functionality, in networks in which remote nodes can transmit packets, but can not receive packets.

- **Ensure interoperability between implementations and vendors.** The ALERT-2 protocol specification should be written with the clarity and level of detail necessary to ensure that products that conform to the specification will be assured of interoperating with each other.

- **Share an RF channel with the original ALERT protocol.** However, significant ALERT-2 functionality may not be available in mixed ALERT/ALERT-2 networks. This capability is the responsibility of the new modem being developed by Blue Water Design LLC.

3.7 Transmission Media Requirements

The ALERT-2 protocols should:

- **Operate with the new modem being developed by Blue Water Design LLC.**

- **Be easily adaptable to other transmission media, such as satellite links, other wireless transmission media, or other available transmission facilities.** Furthermore, it should be possible to use multiple transmission media within an ALERT-2 network.

3.8 Energy Conservation Requirements

Energy conservation is an important objective in many ALERT-2 networks, because many remote nodes are powered by batteries that are recharged by solar panels. The ALERT-2 protocols should:

- **Permit remote nodes to sleep.** The ALERT-2 protocols should permit remote nodes to sleep for long periods of time, although base stations and routers may be expected to be active and prepared to receive and transmit packets all of the time.

- **Operate with limited computational power and storage capacity.** The ALERT-2 protocols should not require remote nodes to have substantial computational power or storage capacity.

3.9 Security Requirements

The ALERT-2 protocols should:

- **Provide optional features that can be independently enabled that ensure the integrity of, prevent the disclosure of, verify the source of, and prevent the replaying of data transported by an ALERT-2 network.**

3.10 Intellectual Property Requirements

The ALERT-2 protocol should:

- **Have freely available, complete protocol specification.** The ALERT-2 protocol specification should be available to any vendor or other party that wishes to implement it, or for any other reason.

- **Permit implementation without paying fees.** Vendors should be free to implement the ALERT-2 protocols without paying for the right to implement or use the protocol.

- **Offer an open-source implementation.** There is a strong desire within the ALERT community to have an open-source implementation of the ALERT-2 protocols available.

4. Bibliography

[Allan] Allan, Rand. "A possible solution for the next generation ALERT system", e-mail sent to the Yahoo! Floodsystems group. August 13, 2002. < http://tech.groups.yahoo.com/group/Floodsystems/message/114>

[Anonymous] Anonymous. "Enhanced IFLOWS Format (EIF) Specification", undated. <http://www.afws.net/supportsite/iflows/enhanced_iflows_format.htm">

[Burnash 1983] Burnash, Robert J. C. "Design considerations for an operational real-time hydrologic data system for small computer systems". Paper presented at the International Technical Conference on Mitigation of Natural Hazards through Real-Time Data Collection Systems and Hydrological Forecasting, Sacramento, California, 19–23 September 1983.

[Burnash 1984] Burnash, Robert J. C. "The Meaning and Challenge of Real-Time Data and Analysis Systems to Future Public Service Programs" presented at the Fifth Conference on Hydrometeorology, Tulsa Oklahoma, October 17, 1983, reprinted in Bulletin of the American Meteorological Society, April 1984.

[Clark 1983] Clark, Robert A., Robert J. C. Burnash, and Ira Bartfeld.. "ALERT: A National Weather Service program for a locally-operated real-time hydrologic telemetry and warning system". Paper presented at the International Technical Conference on Mitigation of Natural Hazards through Real-Time Data Collection Systems and Hydrological Forecasting, Sacramento, California, 19–23 September 1983.

[DIAD 2000] DIAD Corp. "ALERT real-time weather monitoring and flood warning", September 20, 2000. <http://www.onerain.com/includes/pdf/whitepaper/ALERTintro.pdf>

[Francis] Francis, Stephen. "Working Toward a New ALERT Protocol". c. December 2001. <http://www.alertsystems.org/sf.html>

[Futuretech] Futuretech Electronics. Telemetry Systems Overview. <http://www.futuretech.com.au/Telemetry_Overview_R10.pdf>

[Gayl 2006a] Gayl, Ilse. "SAAS Alert Protocol Discussion – 25 October 2006", slides from the "ALERT into the Future" session held October 25, 2006 at the 2006 Southwestern Association of ALERT Systems (SAAS) conference held in Overland Park, Kansas October 23 – 25, 2006.

[Gayl 2006b] Gayl, Ilse. "ALERT2 SAAS 2006: Next Steps", 11/5/2006 (Notes from ALERT-2 discussions at SAAS). Attachment A to this document.

[HydroLynx] HydroLynx. *SCADALYNX 50386 Data Collection Unit: Operating Manual*, part number A1027711-10. 2007. <http://www.hydrolynx.com/_manuals/50386DCU.pdf>

[National Weather Service] National Weather Service. *National Weather Service Manual 10-942: Flood Warning Systems Manual*. November 29, 2005. Appendix F. <http://www.nws.noaa.gov/directives/sym/pd01009042curr.pdf>

[Nelson 2001] Nelson, Rob. "Discussion on a Proposed New ALERT Protocol", e-mail sent by Rand Allen to the Yahoo! Floodsystems group. December 6, 2001. <http://tech.groups.yahoo.com/group/Floodsystems/message/99>

[Salo 2006] Salo, Timothy J. "Notes From October 2006 ALERT-2 Discussions". These notes are included as Appendix B of this document.

[Roark 1999] Roark, R. Chris and Don Van Wie. "The ALERT Protocol: Evolving for the Next Millennium", *ALERT Transmission*, Summer 1999 issue. <http://www.onerain.com/includes/pdf/whitepaper/ALERTprotocol.pdf">

[Roark 2000] Roark, Chris and Don Van Wie. "Update on efforts toward a new ALERT protocol", September 2000. <http://www.onerain.com/includes/pdf/whitepaper/ALERTupdate.pdf">

[Roark 2003a] Roark, Chris and Donald G. Van Wie. "A new ALERT Protocol: Feasibility Study of a New Air Interface and Physical Layer Packet Definition for the ALERT User Community", February 2003. <http://www.afws.net/supportsite/iflows/pdf/ALERT%20Physical%20Layer%20Feasibility%20Study%202003.pdf">

[Roark 2003b] Roark, R. Chris and Don Van Wie, "Alert Protocol Workshop", October 2003. <http://www.stormwatch.com/2003nhwc/s6b_VanWie.pdf">

[Roark 2004] Roark, R. Chris and Don Van Wie. "ALERT Protocol Update" in *Proceedings of the 15th Conference and Exposition of the Southwestern Association of ALERT systems (SAAS)*, October 18 – 20, 2004, Mesa Arizona. <http://156.42.96.39/SAAS04/Session_9.1.pdf>

[Roark 2006a] Roark, Chris. "New ALERT Protocol SBIR Technical Report", January 2006. <http://www.afws.net/supportsite/iflows/pdf/New%20ALERT%20Protocol%20SBIR%20Technical%20Report.pdf">

[Roark 2007a] Roark, Chris. "Status of New ALERT Air Interface Development and First Field Testing" June, 2007.

[Roark 2007b] Roark, Chris, personal communication, December 31, 2007.

[Scawthorn] Scawthorn, Charles. "Modeling Flood Events in the US" in the *Proceedings of the EuroConference on Global Change and Catastrophe Risk Management*, IIASA, Laxenburg, Austria, June 6-9, 1999.

[Slouber] Slouber, James. "Advanced Warning Systems for Flood Prone Areas" in *Proceedings of the 15th Conference and Exposition of the Southwestern Association of*

ALERT systems (SAAS), October 18 – 20, 2004, Mesa Arizona. <http://156.42.96.39/SAAS04/Session_10.1.pdf>

[Van Wie 2003] Van Wie, Don and R. Chris Roark. "A new ALERT Protocol: Feasibility Study of a New Air Interface and Physical Layer Packet Definition, October 2003. <http://www.stormwatch.com/2003nhwc/s6b_Roark.pdf">

[Van Wie 2004] Van Wie, Don and Michael Harter. "Case Study of a Hybrid ALERT-Satellite System" in *Proceedings of the 15th Conference and Exposition of the Southwestern Association of ALERT systems (SAAS)*, October 18 – 20, 2004, Mesa Arizona. <http://156.42.96.39/SAAS04/Session_6.2.pdf>

[Van Wie 2007] Van Wie, Don. "Understanding the Channel Capacity of Large ALERT Flood Detection Networks", June, 2007.

A. SAAS 2006 ALERT Protocol Discussion Notes – Ilse Gayl

ALERT2 SAAS 2006: Next Steps

Contents

Working group

Ilse Gayl (chair), David Haynes, Don Lawrence, Dave Leader, Rob Nelson, Tim Salo, Jim Slouber, Don Van Wie, George Wilkins

Introduction

The purpose of this document is to create a framework and path for the next steps toward defining ALERT2. The SAAS 2006 discussion was a preliminary step toward this, and this document emerged from that discussion.

The purpose of the SAAS discussion was to make progress on the content or "business" layers of the "new ALERT protocol." The new protocol has been several years in process at the lower air interface layers. Unfortunately we did not have present the primary people who worked on that.

The discussion was wide-ranging and constrained simply by a series of assumptions (see Basic assumptions, below). Issues considered outside the discussion domain were put in the parking lot – they are extremely important issues and were simply not part of what we were addressing that day. I have incorporated those as "link" issues into my suggestions for our efforts going forward.

We identified the working group, named the protocol and created actions to be completed next. What follows are my suggestions for how to achieve those goals. These are draft suggestions and I am looking for input from my colleagues.

Chair's suggestions for next step

Based on the outcome of our discussion I pulled out four focus areas that I think can be addressed by our working group in order to get to the next step:

1. Message types – basic types, transition and extensibility
2. Encoding strategies – codes for what we need to know
3. Usage/administration – who maintains global codes, how; policing
4. Link issues – what ALERT2 does and what lies outside

I think focusing on smaller areas primarily will make the overall task more doable. I suggest sub-groups of our working group each adopt one of these areas for further definition. Regardless of which area is being worked on there will be interaction required with the others.

Summary of 25.Oct.2006 SAAS discussion

We named the protocol ALERT2
We agreed to have working drafts by the 2007 NHWC meeting in Savannah of the following:

- Message types design
- Data elements design
- Timetable for prototype

Basic assumptions we used for discussion

- Lower layers will support higher speed, higher accuracy data
- Outcome will be higher capacity network
- Remaining lower layer issues will be addressed separately – parking lot (below)

Primary needs from current ALERT base station users

- Steve Waters – Maricopa County, AZ
  - more available sensor IDs
  - more data received, target 99%
  - more valid data
  - polling option would be nice
  - change repeater pass list remotely
- Rob Nelson – Roseville, CA
  - more data received
- Dan Miller – Overland Park, KS
  - engineering units; sensor type
  - shorter message lengths (time)
  - location not critical
  - built and implemented SOON.
- Keith LeJeune – Harris County, TX
  - more valid, quality data
  - not concerned about location or sensor type
  - repeater can be programmed remotely

Draft components to be developed into requirement – not exhaustive, but a start and what came up at the meeting

- Backwards compatibility a given – i.e., transition path
- Global database of station IDs
- Location as ID?
- Sensor vs. Station ID
- Detect missing reports for any sensor – sequence numbers?
- Parameter codes for sensor types
- Who's responsible for maintaining parameter codes?
- How many bytes for parameter code?
- Engineering values sent
- Sending vs. knowing engineering units
- Sending metadata optional for local system – packet flag
- Metadata location(s) remote base Internet – receive base stations hit central repository to translate incoming data?
- Options for security/encryption
- Option for 2-way communication – control, remote reprogramming. Polling?
- If we use ACK how do we handle broadcast?
- Multiple RF channel use – polling on one, broadcast on another?
- Policing of bandwidth use – who?
- Less than a minute = real time for this group
- Video – not now

ALERT2 SAAS 2006: Next Steps
Ilse Gayl Page 3 11/5/2006

Parking lot issues – fundamental, and not discussed at meeting

- Transmitted data validation done at lower level
- Contention issues of concern
- ALOHA vs. slotted ALOHA or other
- CPU, power requirements – are they getting too high?

Preliminary focus areas for identified issues – content is not exhaustive but these are the issues that came up at the meeting, rearranged

Message types

- Backwards compatibility a given – i.e., transition path
- Sending vs. knowing engineering units
- Sending metadata optional for local system – packet flag
- Metadata location(s) remote base Internet – receive base stations hit central repository to translate incoming data?
- If we use ACK how do we handle broadcast?
- Options for security/encryption

Encoding strategies

- Global database of station IDs
- Location as ID?
- Parameter codes for sensor types
- How many bytes for parameter code?
- Engineering values sent

Usage/administration

- Who's responsible for maintaining ID database, parameter codes?
- Multiple RF channel use – polling on one, broadcast on another?
- Policing of bandwidth use – who?
- Less than a minute = real time for this group

Link issues, including much of the parking lot. These issues affect the business layers because they involve the assumptions we retain or reject about the lower communication layers. There is more work to be done here, and one of our sub-groups will take it on.

- Option for 2-way communication – control, remote reprogramming. Polling?
- Transmitted data validation done at lower level
- Contention issues of concern
- ALOHA vs. slotted ALOHA or other

ALERT2 SAAS 2006: Next Steps
Ilse Gayl Page 4 11/5/2006

- CPU, power requirements – are they getting too high?

Action requested by 1.Dec.2006

I suggest the following sub-group assignments:

Message types: Dave Leader and Rob Nelson
Encoding strategies: David Haynes and Ilse Gayl
Usage/administration: Don Lawrence and George Wilkins
Link issues: Tim Salo, Jim Slouber and Don Van Wie

Please work within your sub-group between now and then to create an outline of the work required for your focus area. Please send it to the group when you are ready – can be sooner than deadline!

This is also the time and place to comment on the focus areas I have drafted. Please feel free to clarify what I have proposed or to propose a modified approach. I'm not sensitive and I just want to get it done!

B. SAAS 2006 ALERT Protocol Discussion Notes – Timothy J. Salo

These notes are from the "ALERT into the Future" session held October 25, 2006 at the 2006 Southwestern Association of ALERT Systems (SAAS) conference held in Overland Park, Kansas October 23 – 25, 2006. The session was moderated by Ilse Gayl. These notes were taken by Timothy J. Salo and were transcribed in their raw form with additions identified by square brackets.

Business Layer Discussion [Transcription of Ilse's slide]

- More sensors per network
- More elaborate data – reduced metadata at base station
- Scope: regional vs. global information
- Backward compatibility / interoperability
- More frequent data reports
- More sensor types / extensibility
- Security

"Better way of validating data on the fly", e.g., CRC
Dave L. – FEC
AZ – 700 ALERT stations
only get 80% of packets, has been decreasing
Brief discussion of time slotting
Globally unique IDs
Location (how should lat/lon be encoded)
Jim: likes acknowledgements, but don't require it
Ilse: So, we want both broadcast telemetry and acknowledged transmissions?
Engineering unit data
Jim: a laudable goal to send engineering units
Ilse: Zero-configuration installation?
Bob (?): access metadata via Internet
USGS: parameter codes (the guy who arrived late...)
James (OneRain): How should new type codes be defined? NHWC?
AZ: Some reservations may not want to disclose data
Q: is this permissible re FCC? Sovereign nation?
Security: is encryption required? Authentication?
Should location be the ID? What if multiple stations/sensors per location?
What should the unique sensor ID be based on?
Identify station independently of sensors?
James: some want two-way communications
send video... "That was discussed and rejected"
Ilse: Since we are doing it today, we should support optional two-way communication
Dave: "We shouldn't exclude two-way communication"

Ilse's slide "Identified Issues" [Transcription of Ilse's slide]

- Data validation done at lower level
- Contention issues of concern
- ALOHA vs. slotted ALOHA

Globally unique identifiers?
Location as identifiers
How would acks work, if used
Data in engineering units
How are data units passed
Where is metadata? Remote? Base station? Internet?
Do we use parameter codes? How many bytes?
How are parameter codes managed?
Security & encryption
Sensor ID vs. Station ID
Should we support video
Protocol should not exclude two-way communication
How is two-way communication managed (bandwidth)
Should reports have sequence numbers?
Discussion of operation under heavy load
Ability to poll for missed data
Discussion of supporting multiple, passive base stations
Mention of multiple channels or time slice channel
Concern about power consumption? CPU?
A one-way protocol and a two-way protocol
Remote configuration "don't go to field with laptop"

Base Station Needs

Steve; AZ Need more sensor IDs
More valid data
Need to receive 99% of data, not 80%
More than 8 bits of data, but not crucial
Would like to poll some stations
Would like to reconfigure (repeater) remotely
Chris: What is "real-time"
Steve: Under a minute is probably enough
Jim: others may have different answers

Rob: More data received (less contention)

Dan: Engineering units
ID type of sensor
Should shorten message length
Lat/lon not critical (at least in every message)

Keith: More quality data, more valid data
Not concerned about location or sensor type
Remote repeater programmability – want this soon

Chris: Backward compatibility & migration

[unknown]: Send metadata only occasionally; not in every ALERT message

Chris: Remotely change parameters
"One of the biggest expenses of managing [an] ALERT network is rolling a truck"

Actions by Savannah

Assign Roles

Chair	David Haynes, Don Vie Wie (maybe)
Role Definition	Ilse, Jim, Dave
Candidates	
Select	Don Lawrence
Working Resources	David Haynes
What and How	Ilse Gayl
How to get broad input	Dave Leader
	Jim Slouber
Ilse and Dave will summarize and e-mail	Don Van Wie
	Rob Nelson
Bulletin Board – will be created	George Wilkins
Rob & John	Tim Salo

By Savannah, committee will have a target date for a prototype (definite timetable)

[The name of the new protocol will be] ALERT2