SPECIFICATION FOR AN OPEN PROTOCOL FOR INTEROPERABILITY

Prepared By

Allen Walker, President

Walker Systems Corporation

PREAMBLE

A substantial amount of communication and activity is occurring in the HVAC industry regarding system inter-operation or Interoperability. This specification is the author's definition of an English protocol that can be simply implemented by any vendor. Because of its simplicity, its application to other languages is obvious.
With the birth of the Internet, existing simple and elegant solutions for data exchange have been adopted. The ease of use of these solutions by the masses was solely responsible for the Internet's explosive growth. By adopting these same standards along with the rest of the world, the HVAC industry can easily specify interoperability in a way that gives control to the building owners.

Current interoperability protocols, although perhaps well intentioned, fall short of achieving their goals. With the power of the Internet, and convergence of technologies, interoperability needs to extend to simple communication between any devices or systems. For a building this must include management systems such as hotel management and reservations, maintenance management systems, Elevators, guest services such as security and booking systems, tenant personal control systems, CCTV systems, plant control systems etc. As well, any small third party system or any previously installed systems should be easily integrated into this communication schema. For the manufacturers of these systems to all agree to implement a constrained, and complex protocol is unthinkable.

What is needed is a way of implementing communications between systems that is simple and flexible enough to quickly encompass any specific interoperable requirement without carrying any extra cost or complexity, and one that compliments our current ability to communicate via the Internet.
The communication technique should allow manufacturers to simply and efficiently adapt their products, either future, or installed "legacy products", to communicate so that owners can expand existing facilities with it and largely avoid the expense of replacing "non compliant" devices. If the protocol is simple, many devices will be able to become compliant with only a firmware change. Almost all digital systems already have one thing in common, an operator interface, generally English like, that communicates with the operator or as a diagnostic. By adapting this interface to support a standardized communication language (in English), these systems will be instantly able to communicate in any interoperable schema and also be ready to connect via the most standard of all communication mediums, the Internet.

The standard of Internet communications is HTML, which is English language with some formatting interspersed in it such as Font, color etc. This is what will be used between all intelligent devices.

Contrary to what some may understand, the Internet/Intranet using TCP/IP technology is the most widely used and inexpensive LAN available today and will grow to pervade everything. Eventually, most devices, even down to the Roomstat will have some form of Internet connectivity on board ---- and an IP address. Intercommunication systems not based on this will simply not survive. The most common device connective mechanism today is a serial port. Devices that can connect or Bridge a serial port to the Internet (or Ethernet) abound, in fact, that is the way most of us connect to the Internet with our PC's.

THE SPECIFICATION –

Benefits to the Owner and Consultant

The enclosed specification will allow device-to-device interoperability for any two intelligent devices. It gives a number of advantages for the specifier and owner:
For the Specifier/Engineer:

-It substantially reduces or removes the learning curve for understanding Interoperability while substantially increasing the flexibility and adaptability and expandability of the Interoperable system.
-It allows simple modification of the Interoperable functionality between the Interoperating systems at any time without necessarily involving the assistance of the Vendor.
-It allows the specifier to define the actual interoperable messages being passed between systems.
-It allows Interoperability to be defined specifically in terms of what information is being transferred between the control systems using simple English statements. These statements, which use the assigned descriptors of the points being passed then become the actual transactions messages that the systems pass back and forth between them
-It allows simple verification of the operational interface using any 'dumb terminal' device such as a PDA.
-Rather than having to relay on the individual vendors to affect the interoperability, the specifier can write the interoperable commands using simple English, then require the vendors to implement them.
-It allows the defacto specification of Internet or Intranet connection between systems using a simple Bridge.
-It allows any strategy command to be passed back and forth between Interoperating systems ie "Start heating strategy for rainy Tuesdays after a long weekend".

For the Owner.

-Allows simple Interoperability that does not eliminate any vendor. By specifying Interoperability in terms that anyone can implement quickly the owners opens the playing field to all contenders, giving him more competitive bids.
-The owner is no longer held to ransom by one, or a few vendors.
-No additional cost for interoperability.
-Once this is implemented in his system --- any other vendor can easily plug into his systems at any point and at any time in the future.
-Owner is able to select suppliers on a value only basis.

Device Hardware Configuration

For simplicity's sake, lets require that all systems intercommunicating do it via a serial port as a minimum so that this port can be bridged to the Internet or other serial ports as desired.. In most modern control systems, a panel or node allowing serial communication into its database should be very inexpensive. For devices such PDA that do not have a serial port, USB to serial converters, (or their equivalent) abound.
Devices that have an Ethernet port on board for Internet communications should also have a serial port so that local testing and communication via the Protocol standard can be achieved. In this case the serial port would copy or repeat the data transmitting to and from the Internet for this panel.

We are going to require flexible database mapping between systems done in a way that the system programmers and commissioners can modify data being transferred at will as they evolve the control strategy involving the various interoperating systems.

Example of transaction using the Interoperable language.

General command format is:

"Destination : Command/Response : Originator"

Where:
Destination ---is the destination the command packet is directed to.
Command/Response ---is the data being passed which would be a command or a response to a command.
Originator --- is the station issuing the packet.

"ALLSTATIONS : ! OAT = 25 deg C : EMCS Master

Lighting System : ! START ZONE1 : EMCS MASTER

EMCS MASTER : * ZONE1 = On : Lighting Systems

OPEN PROTOCOL for INTEROPERABILITY (OPI)

Serial Port.

The EMCS shall have an RS232 serial port operating at a minimum of 9600 baud with one start bit, no parity, and 2 stop bits. The port will be able to auto configure its baud rate up to 128K Baud.

The port will be dedicated to interoperable communications and be capable of being connected via a 'party line' connection I.e. it will detect collisions and stop transmitting when someone else talks. The port should be able to connect to the Internet via an Internet Bridge. The software will allow multiple concurrent command packets be handled.

The serial port will execute and respond to interoperation commands within two seconds. It will use the printable ASCII character set, case dependant. The EMCS vendor shall supply a single serial port for each Interoperable link and the appropriate bridge where Internet/Intranet connection is required.

The system shall have the capability of expanding to at least 8 individual and concurrent interoperable links. Where Intranet/Internet connectivity is required and implemented these concurrent links can be via the same bridge and IP address.

System Names/Group name/Broadcast Name.

1. The EMCS system shall allow the assignment of an English name (***EMCSname***) of up to 20 characters by which it shall be referred to in all interoperation communications. It shall use this name for either sending or receiving commands and will not respond to any command without ***EMCSname*** as its address other than to global commands.
2. **Global Commands**.
   a) The EMCS shall allow the assignment of up to three additional group names (***GROUPname***) for the purpose of segregating the system into different groups such as all lighting controllers or all card access systems.. The EMCS will react to all commands issued to any of its assigned groups in the same way it will respond to a global command as below.
   b) The EMCS shall allow the assignment of a single broadcast name, ***"ALLSTATIONS".*** This name will be used for all broadcast messages such as 'set time' or outdoor air temperature etc.
3. **Response to Global Commands**. The EMCS shall not respond to Global or Group commands but shall keep a log of the last 3 global commands for verification if that is required.

System Interoperable requirements

It is the general intent of this specification that all points in each EMCS system be available for monitoring and control via the interoperable serial port attached to the system. As a minimum, the commands listed below should be implemented within the Vendor's product. The owne,r prior to final acceptance, will verify correct operation. If the EMCS is to connect to the Intranet/Internet, the owner will be responsible for providing the connecting the EMCS system to the building LAN / WAN.

Definitions:

Interoperability Port The serial port described above through which all interoperability communications takes place.

EMCSname The owner-assigned name for the vendors system. The owner must be able to enter or change this name at will using a high or specially assigned password. Alternately the Vendor can assign a backdoor password for the system that will allow use of the "set name" command below to set these names via the Interoperability Port.

GROUPname1, GROUPname2, GROUPname3, Global group names assigned to the vendor's system by the owner. The owner must be able to enter or change these names at will using a high or specially assigned password. Alternately the Vendor can assign a backdoor password for the system that will allow use of the "set name" command below to set these names via the Interoperability Port.

DESCRIPTOR--- the descriptor or label assigned to a point or variable within the vendor's EMCS system. These should be the point labels assigned during the installation or commissioning

COMMAND----- a string representing the command required. All commands except global commands should result in a response. A command string will consist of a leading character followed by the body of the command. The leading character will tell the global sense of the command.

"?" --- Indicates a question or request for data.
"!" --- Indicates a command or assertion.
"*" --- Indicates a response.

Examles:

"? Give_value_of" requests value of data point.
"! Set_value_of" sets the value of data point.
"* OAT=72 Deg" response to command.
"* Operation_confirmed" ---- response to command.

DESTINATION --- a string representing the assigned name of the system to which a command is being sent.

ORIGINATOR --- a string representing the assigned name of the system from which a command is being sent.

Command Structure

Interoperability commands will consist of three fields separated by Colons and terminated by a carriage return and line feed <CRLF>. Optionally, a checksum <CK> can be included. The checksum <CK> will be a two number string just prior to the carriage return which is the low order remainder of the sum of the characters in the string up to but not including the checksum. Software can use the checksum to verify the integrity of the data stream or ignore it with no failure. Software that does not use the checksum should ignore it. The checksum will be the characters between the last colon and the <CRLF>,

General format of command:

DESTINATION: COMMAND STATEMENT: ORIGINATOR: <CK><CRLF>

General format of response:

DESTINATION: RESPONSE STATEMENT: ORIGINATOR: <CK><CRLF>

Example 1:
"Master" measures the value of outdoor temperature and uses the label "OAT".
System 1 = "GYM"
System 2 = "Master"
The Gym request the value of outdoor temperature.

MASTER: ?Give_Value_of OAT: GYM: <CK><CRLF>

The Master responds

GYM: *OAT=71 Deg.f: MASTER:<CK><CRLF>

EXAMPLE 2:
Master commands Supply Fan6 to enter occupied mode.
The system with Supply Fan 6 is the gym and it the occupied state is controlled by a variable called FAN6-OCCUPIED.

GYM : !Set_Value_Of FAN6-OCCUPIED=ON : MASTER <CK><CRLF>

Response

MASTER : *FAN6-OCCUPIED=ON : GYM <CK><CRLF>

Author's note

This command requires a detailed knowledge of the process and strategies of both systems so a flexibility mechanism is needed. A protocol that allows an "as needed" functionality for the system implementer will work. In a general sense, this would be difficult. But for a specific application (which is always the case) this is quite simple. The question is does the protocol provide only the tools, or does it provide pre-processed solutions. If the later is the case, then you are stuck of
The problem with most Interoperable protocols today is that they want to define the control strategies and give the solutions where what is needed are just the tools. The owner or consultant should define the control strategy, and then use the communication tool, OPI, to implement the types of communication he requires. This gives complete flexibility and control to the Owner and the ability to change strategy at will.

CONCLUSIONS

A simple language implementation for Interoperability such as Walker's OPI specification will reduce the cost of Interoperability and gives complete control of Interoperable solutions to the owners and specifiers.
Walker recommends that vendors be required to implement OPI for the following reasons.

1. It is by far the most cost effective and flexible Interoperable technology.

2. It can be retrofitted onto any existing control system that uses programmable memory ie by a firmware upgrade.

3. Because it is verbose and logical it can easily be embedded in any Internet carrier such as HTML.

4. It is simple and low cost to implement making development time man hours instead of man years.

5. It gives owners and specifiers the ability to explicitly define interoperable command exchanges required in various vendors products right in the specifications and also defines a method by which an owner or specifier can test and verify vendors command exchanges. For clarity this test method can also be stated in the specifications.

6. Walker's ASCII protocol has been used for 10 years and has about 5 to 10 different products that have interfaced to it. The protocol is so simple that Walker's customers normally implement it by themselves when third party interfaces are required. Third party vendors have usually done the interface in a matter of hours or days.

7. OPI is a simpler, more general, and more verbose format of the ASCII protocol above and will be even simpler for third party vendors to implement.

8. OPI is so verbose that third party manufacturers can publish exact OPI command summaries that their product will respond to. These are so simple that they can be publish right in the manufactures documentation normally shipped with the product.

9. Walker is now offering a number of products that follow the OPI standard and has generated substantial interest from several large vendors in the Middle East including Teletrol and Johnson.

10. OPI is so simple that specifiers can use precise OPI command sequence within their specification documents and RFQ's to clearly state what interoperable sequences will be required in the final system implementation. As well the specification can detail the exact method of testing for compliance. This will leave no uncertainty or guesswork and will guarantee the owner gets what he pays for.

11. The time to implement any OPI command within any Walker SAC is a few minutes. Other manufactures will be pleased to implement OPI because it is simple and allows them to publish and offer simple interoperability and Internet connectivity for their products.

12. Because OPI is so simple it will level the playing field, allowing any vendor to compete on an RFQ independent of whether or not he has previously implemented OPI because his implementation time will be measure in hours, not months or years.

Please contact me if you have any questions

Al Walker
awalker@walkersys.com
www.walkersys.com