Interoperable Cloud Networking for a Smarter Grid

Dave Hardin

GridWise Architecture Council
Invensys Operations Management
70 Mechanic St. Foxboro, MA 02035
david.hardin@ips.invensys.com

Keywords: dynamic pricing, cloud computing, service bus

"Make everything as simple as possible, but not simpler." - Albert Einstein[1]

Abstract

The smart grid will rely upon the ability to moderate consumption through market-based pricing and load-reduction signaling. This requires that pricing and reliability signals be distributed securely and in near real-time to very large numbers of automation systems in homes, building and industries of all types, distributed over a wide geographical area through a wide variety of existing network infrastructures.

Communications of this magnitude represents a significant challenge requiring compute and networking resources normally associated with large corporate data centers. This paper proposes that cloud computing technology has the functionality needed to provide the security, interoperability and performance required for large-scale smart grid applications at a significantly less cost than traditional data centers.

This paper will review and analyze the benefits and limitations of several networking architectural patterns in use by cloud computing providers. These patterns will be presented in the context of customer interactions and providing a consistent flow of actionable dynamic pricing information, along with reliability and curtailment signals, from ISO/RTO's (Independent Service Operators/Regional Transmission Organizations) through utilities and aggregators to residential, commercial and industrial consumers.

1. OVERVIEW

Smart grid is a National imperative [1] to upgrade and expand the electrical infrastructure in order to reliably and securely deliver power from a variety of fixed and variable energy sources to existing and new customers such as electric vehicles. The driver behind this effort is the need, supported by Congress, to protect National security by becoming less dependent upon foreign energy sources. This represents a challenge due to the real-time nature of electrical energy. The power grid must be maintained in a delicate balance with the energy supplied equaling the energy demand at all times. Today, the existing grid relies upon excess standby generation capacity to meet the requirement for reliable power delivery. And when it can't, the result is a blackout. In addition to standby capacity, the smart grid will rely upon distributed generation, future energy storage, advanced forecasting and the ability to moderate consumption through dynamic pricing and demand response load-reduction signaling based on market and grid conditions. Cloud networking technology can be applied throughout the electricity supply chain from monitoring distributed generation to providing customers with real-time information. This paper will focus on cloud networking as a technology to enable smart grid customers to make better energy decisions by providing "prices to people and devices".

2. "PRICES TO PEOPLE AND DEVICES"[2]

Energy pricing information, along with reliability and curtailment signals, form a supply chain from the regional ISO/RTO wholesale markets through local utility and aggregator retail markets to residential, commercial and industrial consumers and the devices residing within homes, buildings and industrial sites. This end-to-end pricing system starts with wholesale location-based marginal price feeds from ISO/RTOs that flow through utilities (and potentially aggregators) with retail price conversion. The retail pricing is then fed to consumer Energy Service Interfaces (ESI) [2] for monitoring and control. The price feed is converted at the ESI into a form compatible with end-use devices and appliances. Interoperability decisions are required at each level of the GridWise Architecture Council (GWAC) Interoperability Stack [3] as the pricing information flows from source to destination. This paper focuses on technical interoperability at the GWAC Technical Connectivity layers.

[1] http://www.quotedb.com/quotes/1360

[2] This term originated with GridWise Architecture Council members Dr. Lynn Kiesling and Alison Silverstein.

3. TECHNOLOGY CHALLENGES

Wholesale pricing information is currently available from ISO's (i.e. New England ISO[3], Midwest ISO[4], New York ISO[5]) as 5-minute location-based marginal price (LMP) values. Each LMP price is associated with a specific geographical location code and the semantics of the LMP values can vary between ISOs and RTOs. The 5-min LMP may represent the wholesale energy price for the last 5-minute period or for the next 5-min period.

This information must be filtered and transformed into actionable pricing data for large-numbers of customers. High performance, low-latency communications of this magnitude on a wide-area scale represents a significant challenge requiring compute and networking resources normally associated with large corporate data centers.

4. APPLICATION REQUIREMENTS

Some relevant application requirements that stand out include:

- The need to use standard data models and communication encoding technologies.
- The need to be compatible with as many relevant standards as practical.
- The need to be compatible with the existing Internet and broadband infrastructure.
- The need to scale quickly and economically.
- The need to provide access and cyber security that is compatible with existing firewalls.
- The need to provide low-latency communications.
- The need to provide highly available communications.
- The need to provide rapid time-to-market along with acceptable costs.

5. TIMELINESS AND SCALABILITY

Of all variables that determine how and when data needs to be transferred and consumed, "time" is arguably the most important.

Within the context of communicating dynamic pricing and demand response information from the wholesale markets to the retail markets, the following characteristics must be considered:

- When does the information change?
- When is the information received?
- How long is the information valid, accurate and useful?
- Is there enough time to act upon that information before it becomes of historical interest?
- How much effort is needed to get the information distributed in time to do something with it?

In the "prices to people and devices" application scenario, every five minutes pricing data needs to:

- flow from the balancing authority's wholesale price data store to the utilities or "load-serving entity",
- be transformed into retail prices,
- flow from the utilities to aggregators,
- be transformed into consumer prices,
- flow from the aggregator (or utilities) to all consumers through the Energy Services Interface and
- be transformed into device signals that can be viewed and perform actions.

This results in two, three or more distinct data transfers. The sooner the data arrives at the customer's Energy Services Interface, the more time is available to perform beneficial control actions.

One general rule-of-thumb is to allow 10% of the time interval to be used for data transfer latency leaving 90% of the time interval available for performing actions. A change will then be detected at the ESI with 90% of the time still available until the next update. Using this rule, an hourly change requires a 6 min maximum latency, a 10 min change requires a 1 minute latency and a 5 min change requires a 30 sec. latency. For two data transfers, 15 second maximum latency per transfer is required and, for three data transfers, 10 second maximum latency is required.

The transfers between the wholesale market, utilities and aggregators involve relatively small numbers of messages but sending dynamic pricing data to 100,000 (or 1,000,000) consumers means that 100,000 (to 1,000,000) messages need to be sent in under 10 seconds. Scalability and performance at this magnitude requires special attention to system architecture.

6. SYSTEM ARCHITECTURAL CONCEPTS

6.1. Separation of Concerns

A layered communication stack provides separation between the distinct functions that must be performed. This permits different behaviors to be injected into the layers of the stack without impacting other layers. Independent layers thus provide a high degree of flexibility and permit a

---

[3] http://www.iso-ne.com/

[4] http://www.midwestiso.org

[5] http://www.nyiso.com

communications stack to adapt to change by accommodating alternate behavior and functionality.

Using this layered stack approach, the "what" that is transmitted can be separated from the "how" it is transmitted. This permits new innovations in the way that the information is transmitted while preserving what is transmitted. The cell phone and Internet industries have clearly demonstrated the benefits of this approach. HTML and voice payloads are transmitted through an ever broadening array of transport technologies while at the same time preserving the semantics and structure of the information transferred.

The GWAC Stack clearly differentiates these layers. A high-degree of systems interoperability can be achieved through the alignment of information at the semantic and syntactic levels (GWAC Stack Level 3-4) while enabling innovation at the technical connectivity layers (GWAC Stack Level 1-2).

Advanced communications technology can be utilized to transport standardized semantic information models in standardized syntactical formats thus achieving interoperability while simultaneously supporting emerging communication technology. This will be a normal smart grid scenario moving forward. While an argument can be made that this prevents true plug and play interoperability, it still reduces the "distance to integrate" [3] to low-cost systems integration at the technical communications transport level.

6.2. Informational Semantic and Syntactic Interoperability

The NIST Interoperability Framework V1.0 Priority Action Plans [2] are addressing the need for a standardized dynamic pricing information model with standard data encoding. This will be the information model and data format used to transfer pricing and event information between systems on the smart grid. These standards are in the process of being developed and should take into consideration the following: 1) XML (Extensible Markup Language) is a standard modern data encoding technology that is well supported by all viable software systems, including embedded systems and 2) web service interfaces developed to transport these XML documents should provide a messaging-style (or Document/Literal) SOAP interface as opposed to an RPC/Encoded-style interface. The issue that remains then is how this information should be communicated at scale.

6.3. Technical Communication Patterns and Interoperability

Communication patterns involve the concepts of "pull" and "push" data transfer. The benefits and tradeoffs associated with these two patterns determine when they should be applied to an application.

Pull communications is data transfer initiated by the receiver of the data and is often referred to as request/response. The receiver sends a request to the sender (potentially in the form of a Uniform Resource Locator or URL) and the sender responds with data, or an error message if the request could not be satisfied. This pattern creates a time skew unless the receiver knows when the data is available and therefore when to request an update. This time skew is of little impact if the data is slow-changing and the client does not request updates often. This is the case with "normal" web browsing. The pull pattern provides good error detection, failures are detected within a timeout window and interactions are simple. Ask for something, get something. A lot of mileage has been traveled with this very basic pattern. Problems arise however when you want to be notified when data changes. This requires polling often but usually getting the same response. One work around is to use a low-impact and efficient "change flag" to signal that the data has changed and to "come get it!". Even this approach becomes very inefficient as the number of requests increase and the time between polls decreases.

"Push" communications represents data transfer initiated by the sender of the data and is referred to as publish/subscribe. The receiver subscribes to the data and the publisher sends that data when changes occur, or a time interval elapses. A typical example is the stock market. Data changes when it wants to, not when you want it to. Nothing happens, until something happens. Would it make sense to constantly pickup a phone to see if someone is calling you or is it better and more efficient to wait for the phone to ring?

7. POTENTIAL TECHNOLOGY SOLUTIONS

7.1. Advanced Metering Infrastructure (AMI)

Some newer smart meters have the capability of distributing limited pricing information and demand response signals to residential customers. Deployments of these meters are already underway by some utilities. Many existing smart meters were installed for the purposes of remote disconnect and meter reading and were not designed to provide advanced functionality such as pricing and demand response signaling. In addition, AMI communication channels are often bandwidth limited and meters are costly to support, upgrade and maintain. Embedded devices differ from general purpose computing systems in that they are purpose-built to be rugged and stable over a long lifetime. This differs from general purpose computing which is lower cost, has a shorter lifetime but can easily change and adapt. A natural migration of technology occurs from general purpose devices to special purpose devices as technology matures and stabilizes.

Many high-end residential, buildings and industrial facilities already have Internet connectivity that can be leveraged and used for interaction with smart grid dynamic pricing and demand response signaling.

In these situations, alternative solutions, such as Internet connectivity, should be analyzed and evaluated.

7.2. Enterprise Integration Patterns and Protocols

Within enterprises, the use of "Service Bus" middleware (i.e. Enterprise Service Bus or ESB) integrating application services using service-oriented architectural (SOA) patterns is commonplace. [4][5][6] This approach is being widely adopted by utilities as they upgrade their information technology infrastructure to integrate newer smart grid functionality. Service buses provide the messaging "glue" that permits a service to find and efficiently use other services through a directory or registry structure. Performance and security are both top priorities. ESB's provide both pull and push data transfer with low-latency along with integrated access security covering authentication, authorization, privacy, integrity and non-repudiation using public and private key technology. Publish/subscribe notifications use "push" technology for performance and scalability.

7.3. Internet Patterns and Protocols

Outside the enterprise, the Internet and WWW provide a great platform to distribute information on a global scale using standard request/response "pull" web protocols such as HTTP (Hypertext Transfer Protocol) with HTML (Hypertext Markup Language), SOAP (originally Simple Object Access Protocol) web services and REST (Representational State Transfer) web services. Google and other web data appear to be very comprehensive and covers almost everything. But the data is updated slowly and much of the data becomes obsolete and "rots on the vine" waiting for an Internet "crawler" to visit and update the data using a "pull" pattern. These technologies were designed to allow asynchronous access to data when a user desires that data, not when the data actually changed.

RSS/Atom syndication protocols were developed to provide this notification capability and allow users to subscribe to data content such as news feeds and blogs. The Atom provides a generalized XML metadata container that enables diverse content to be transferred between systems in a standard way. It's widely used and well-designed.

A major limitation to these syndication protocols is that they actually use a pull mechanism internally to poll for data changes every hour or so. Different clients receive changes at different times. Satisfactory results can be obtained if changes occur slowly and timing jitter doesn't have any affect on the system but polling doesn't scale to large numbers of users or to data that changes fast. Why is this required? Why can't scalable push technology be used? Unfortunately, the need for tight Internet security has locked down firewalls. For very good reason, most firewalls don't allow any inbound ports to be open and only allow limited outbound ports to be open.

The other problem is a limited IPv4 (Internet Protocol version 4) address space that requires network address translation (NATs). The use of NATs impedes direct IP endpoint addressability which is needed for an external system to send a push notification to another system. These issues have impeded the ability to use scalable push technology in the Internet. It should be noted that several attempts have been made to standardize publish/subscribe web service protocols (i.e. WS-Notification, WS-Eventing, WS-Events, WS-Event Notification) but none have reached critical mass due, in part, to the above limitations.

In order to overcome these limitations, Internet technology companies have developed the concept of "messaging relays" in order to implement instant messaging and chat channels along with other peer-to-peer networking applications such as Skype, Napster and others. These services rely on outbound client connections to server resources and have proven to be scalable and performant.

8. PROPOSED TECHNOLOGY SOLUTION

Enterprise service buses provide the push communications and performance needed but only within a private network that interconnects systems within a protected corporate boundary. The Internet provides the very broad, global communication access needed but doesn't provide standard push notification. Both have adopted similar security technology. Neither solution satisfies the requirements for delivering high-performance notifications on a global scale.

It is proposed that:

- The requirements for communicating and distributing dynamic pricing can be fulfilled by combining the open accessibility of Internet communications with the performance and low-latency of an Enterprise Service Bus based on a messaging-relay architecture. This is essentially an Enterprise Service Bus in the Internet. One that can be accessed easily, widely and securely, paid for as used and expanded as needed.
- The information transferred should utilize the standard Atom Syndication Format as it provides an ideal metadata container to host the NIST standard dynamic pricing and demand response information payloads.

Microsoft .NET Services Bus[6] cloud networking platform provides one implementation of this architecture and will be used as an illustrative example. The WCF (Windows Communication Foundation) [7] "event relay" channel provides peer-to-peer publish/subscribe connectivity

[6] http://www.microsoft.com/azure/servicebus.mspx

through a directory structure based on Uniform Resource Locators or URL's. This technology coupled with the Microsoft claims-based Access Control Service (ACS) provides a scalable infrastructure that can technically support the secure delivery of dynamic pricing information including 5-minute RTP (Real Time Pricing) from the wholesale markets to consumers as well as low-latency demand response event notification. Feasibility prototype results indicate that an average message propagation latency of under 3 seconds can be attained using publish/subscribe push messaging. [8]

8.1. Interoperability

Internet service bus technology is considered by most software vendors to be proprietary and a competitive advantage. As such, the technology relies upon vendor-provided tools for integration into systems and products instead of using standards-based wire-protocols. The Microsoft ".NET Services" service bus supports interoperability by permitting applications to be written in either a .NET language (i.e. C#, VB) or Java[7]. In either case, a service bus communication stack needs to be installed on all platforms. These are limitations typical of emerging communications technology. By separating the messaging payload content from the messaging transport infrastructure, the impact of incorporating proprietary service bus technology on interoperability can be minimized resulting in a balance between standardization and innovation.

8.2. Security

Cyber-security and access security are both very critical within a service-bus architecture. Cyber-security issues are outside the scope of this paper but it should be noted that all major providers of cloud-based computing resources are very active in implementing protection against cyber attacks.

Access security must permit only clients with proper credentials to access service bus resources on an as-needed and as-allowed basis. This includes providing user authentication and authorization along with message integrity and privacy. The Microsoft Access Control System (ACS) provides access security based upon a set of claims encoded using SAML (Security Assertion Markup Language) tokens encrypted using X.509 certificates and conforming to WS-Trust and other security standards. SAML provides a standard mechanism for describing and transmitting security information.

[7] http://www.jdotnetservices.com/

8.3. Cost Analysis of "Leasing" an Internet Service Bus

The following cost analysis uses costing information for the Microsoft .NET Services platform released by Microsoft.[8] It only addresses costs associated with actual messaging.

8.3.1. "Customer" Subscription Costs

The costs associated with price subscription will vary depending upon the number of subscribers and the frequency of price updates.

Number of 2KiloByte Messages Per Consumer: 365 Days/Year * 24 Hours/Day * 12 Messages/Hour = 105120 Messages/Year

Message Cost Per Consumer: $0.15 * 10-5 * 105120 = $.1577/Year

Bandwidth Cost Per Consumer: $0.15 outbound / GB * 2 Kilobyte/Message * 10 -6 GB/Kilobyte * 105120 Messages = $0.0316/Year

Total Cost Per Consumer = $0.1577/Year + $0.0316/Year = $0.1893/Year

Case #1: 10,000 consumers = $1893/Year

Case #2: 100,000 consumers = $18,930/Year

Case #2: 500,000 consumers = $94,650/Year

Case #2: 1,000,000 consumers = $189,300/Year

8.3.2. "Utility" Publication Costs

The costs associated with price publication will vary depending upon the number of pricing signals that need to be published. This will be based on the geo-location mapping between wholesale regions/nodes and retail regions. Within a region, ISO/RTOs manage several thousand wholesale pricing zones distributed over several utilities. For purposes of estimation, it is assumed that each utility will manage 1000 to 2000 retail pricing zones.

Number of 2KiloByte Messages per Publisher: 365 Days/Year * 24 Hours/Day * 12 Messages/Hour = 105120 Messages/Year

Message Cost Per Publisher: $0.15 * 10-5 * 105120 = $.1577/Year

Bandwidth Cost Per Publisher: $0.15 outbound / GB * 2 Kilobyte/Message * 10 -6 GB/Kilobyte * 105120 Messages = $0.0316/Year

Total Cost Per Publisher = $0.1577/Year + $0. 0316/Year = $0.1893/Year

Case #1: 1000 publishers = $189/Year

[8] http://www.microsoft.com/azure/pricing.mspx

Case #2: 2000 publishers = $ 378/Year

For comparison, these costs can be balanced against the capital and expense costs of building and supporting a modern high-availability data center that provides the resources, connectivity and technology needed to provide the equivalent communications capability.

9. CONCLUSIONS AND ACTIONS

It is proposed that current cloud computing technology has the functionality needed to provide the security, interoperability and performance required for selected near-real-time, large-scale smart grid applications at a significantly less cost than traditional data centers. These applications include the communications of dynamic pricing and demand response information.

Approximately 60% of the Nation's electrical system is serviced by ISOs/RTOs and most utilize LMP (Locational Marginal Pricing) real-time wholesale pricing.[9] Over fifty percent (50%) of households have broadband Internet access and AT&T has committed to providing 100% coverage within 5 years.[10] Given this widespread availability of LMP pricing along with large, and growing, access to broadband, a reasonable approach moving forward involves the following steps:

- Define pricing information standards. This work is currently underway as NIST Priority Action Plans for Common Pricing Model, Common Scheduling Mechanism and Standard DR Signals. [2]
- Validate and deploy pricing information using cloud networking and other Internet technologies to computer-based customer systems. This would fuel the development of ESI software for residential, commercial and industrial users. Validation must address the friction points and impedance mismatches that exist in the pricing value chain at upper layers of the GWAC Stack. [3] These need to be identified and remedied before wide-scale application of a technical solution is feasible.
- Move system functionality to embedded products on a selective basis, such as smart meters and Energy Service Interfaces (ESI).

Advanced Internet technologies, like Microsoft's .NET Services service bus, can provide a valuable vehicle for expediting the implementation of smart grid technologies. It provides a mechanism for distributing pricing and demand response information to the large population of consumers that have Internet access.

The architectural approach presented achieves a high-degree of systems interoperability through the alignment of information at the semantic and syntactic levels (GWAC Stack Level 3-4) while enabling innovation at the technical connectivity layers (GWAC Stack Level 1-2).

Communications technology will continue to evolve and the smart grid must be capable of leveraging the benefits that new technology provides while maintaining interoperability through standardized semantic information models in standardized syntactical formats.

Moving forward, it is the desire of the author that the many cloud service providers work together to develop standards for advanced high-performance publish/subscribe interfaces in the same way that SOAP and REST have been standardized and gained wide acceptance.

Biography

Dave Hardin is a Staff Engineer and System Architect for Invensys Operations Management specializing in system and software architecture for industrial and energy management systems. He has more than 25 years of experience designing, constructing and managing process control and production management systems with 15 years as Engineering Team Leader of the automation modernization program at the (now) Valero oil refinery in Delaware. Dave serves as a member of the DOE-supported GridWise Architecture Council and co-chair of the NIST Industrial-to-Grid Domain Expert Working Group. He also serves as a member on the NERC Smart Grid Task Force, OPC Foundation's Technical Advisory Council and Unified Architecture Working Group. Hardin graduated from the University of Delaware with a Bachelor of Electrical Engineering. He is a Registered Professional Engineer (DE/MD), a Project Management Professional (PMP), an IEEE Certified Software Development Professional (CSDP) and Microsoft Certified Application Developer (MCAD).

References

[1] Energy Independence and Security Act 2007, Title XIII, http://energy.senate.gov/public/_files/RL342941.pdf

[2] *NIST Framework and Roadmap for Smart Grid Interoperability Interoperability Standards Release 1.0*, 2009, Office of the National Coordinator for Smart Grid Interoperability, National Institute of Standards and Technology, U.S. Department of Commerce,

---

[9] http://www.nrgstream.com/tu_data_index.htm

[10] www.att.com/Common/about_us/public.../100_Pct_Broadband.pdf
http://www.networkworld.com/news/2008/072408-broadband-penetration-gartner-study.html

www.nist.gov/public_affairs/releases/smartgrid_interoperability.pdf

[3] *GridWise Interoperability Context-Setting Framework V1.1,* 2008, GridWise Architecture Council, http://www.gridwiseac.org/pdfs/interopframework_v1_1.pdf

[4] E. Thomas. 2009 *SOA Desi gn P atterns.* ISB N 0-13-613516-1.

[5] D.A.Chappell, 2004, *Enterprise S ervice Bus* , ISB N 0-596-00675-6

[6] G. Hohpe, B. Woofe 2004 *Enterprise Integration Patterns, Designing, Building and Deploying Messaging Patterns,* ISBN 0-321-20068-3

[7] J. Lowy, 2009 *Programming W CF Services*, 2nd Ed. ISBN 978-0-596-52130-1

[8] D. Hardin, A. Savateyev, *Joint Invensys/Microsoft SmartGrid .NET Service Bus Dy namic Pricing Demo*, PDC 2009