A new approach to scaling high availability: High Availability Architecture HA² combines scalable HA features across architectures in a new way for multiple levels of system complexity

By Mathias Renner

High availability (HA) and system scalability are often required as applications are converted to network-centric architectures. However, software development and frequently incompatible standards may hinder a timely HA solution. This article addresses these problems and introduces High Availability Architecture, HA², as Force Computers' solution to streamline long-term systems development.

The need for scalability

High availability (HA) and system scalability have become commodity features as more and more applications are converted to network-centric architectures. Because ease of application migration between platforms is necessary, scalability is a common requirement. For example, the trend can be observed in the converged telecom/datacom next-generation network (NGN) market and applications, and is equally obvious with the success of the PICMG 2.16 CompactPCI Packet-Switching Backplane standard within the CompactPCI sphere. NGN deployments range from large Central Office core network equipment to edge network and cost sensitive customer premises equipment – hence, the requirements for migration from smaller to larger overall system sizes, different standard compliances, different availability, and different price points. To address these design needs efficiently, OEMs decided open standard based systems offer the best time-to-market and interoperability for sourcing and price.

When evaluating the impact on scalability of high availability systems, multiple standards need to be covered to address different functionalities. However, those standards have not often been defined to coexist with one another and can only be applied on a selective basis. When adopting scalability requirements for different levels of HA features, different system architectures or complexity and overall-size, typically the framework of open standards is overburdened and OEMs needs to make significant investments to maintain its software on those very different platforms.

And this is the problem: Software impact and too many – often incompatible – standards, do not leverage the advantages of open standards when applied to an HA solution. This article addresses these problems and discusses one possible solution.

The OEM-friendly approach for the best use of standards and interoperability of standards while maintaining time-to-market and reuse of software building blocks is to define a super standard as the key reference for system design. The High Availability Architecture (HA²) as proposed by Force Computers does just this.

The following article is structured to further introduce the scope of scalability with the dilemma of open standards, introduce HA², show the scalabilty of system size and HA and cost covered by HA² as well as introduce its applicability to control and transport plane applications. It will also define several system views, highlighting various aspects of highly available service architectures, specifically an application view, an operator view, high-availability platform properties and a middleware-oriented view, all of them leading to complementary requirements for an HA²-compliant platform. Also, a failover example will describe how each of these properties contributes to a system's capability of tolerating partial failures. Finally, it presents an example of a Force system adhering to the HA² standard, providing customers with the mentioned advantages of a consistent and long-term architectural thinking.

Dimensions of scalability and conflict of standards

Depending on an OEM's manufacturing customs, there is an affiliation to certain building practices, such as slim-line stackable systems vs. blade card servers. There is a good reason for this, because stackable 1U servers have lower entry costs than blade card systems and on the other side, for larger configurations, blade card systems provide higher densities of processing power and cost advantages than do stacked 1U systems. Even among blade card systems there are differences for front I/O and rear I/O preferences, which depend on the I/O intensity of the system and the given service concept.

All these factors make the HA open standards challenge fairly transparent, because OEMs, especially concerning NGN equipment, need to reuse their applications on any given system implementation whether it is PICMG 1.x, PICMG 2.x with or without PCI bus support (see Figure 1).

HA² meets this challenge and defines an abstract system architecture that incorpo-

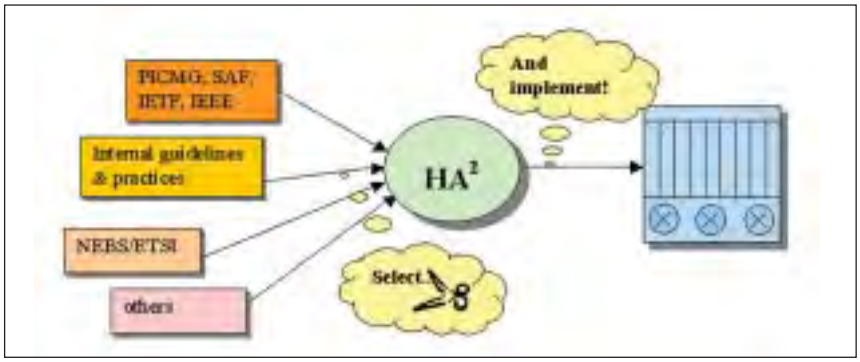

Figure 1. HA² combines aspects of various external and internal standards

rates open standards and defined interfaces to allow for interoperability and reuse of building blocks – especially HA software and OEM applications.

The second challenge for an open standard based HA system is to tailor the HA features to the actual technical and commercial requirements of the OEM. This requires scalability of the cost-associated HA features and will result in an attractive system price because it reflects the customer requirements. Also the scalability of HA features and price is fully addressed by HA².

High availability architecture – The holistic basis for Service Availability

High availability is a term that is being used quite often these days in the embedded systems world. However, most discussions covering this topic focus on individual hardware aspects – such as hot-swap capability, system management busses or hardware reliability – or on individual software aspects such as cluster management software or system management middleware. In order to create the basis for Service Availability at really high levels, i.e. 5-nines (99.999 percent) or even 6-nines (99.9999 percent), it is necessary to take an holistic approach: only if all pieces and building blocks of the architecture are designed from the beginning to fit to each other, will the entire system achieve such high Service Availability levels.

HA² stands for "High Availability Architecture" and represents a new standard for "vertical" system solutions – from the core hardware technologies up to the higher levels of software. This standard is defined by Force Computers but basically driven by carrier grade telecom requirements. HA² is a specification that not only outlines current and future integration levels of Force Computers products, but also intends to promote a new holistic level of thinking between equipment providers and their customers.

HA²'s purpose is to secure the huge value of telecommunications software applications over a long period. This is achieved by a well-defined platform architecture that is expected to survive the short-lived technologies and components. Combined with the company's commitment to offer state-of-the-art equipment based on this architecture telecom OEMs will be able to quickly and easily migrate to new technologies and performance levels.

Technically speaking, HA² describes a computer platform with scalable performance, functionality size and availability. Acknowledging the fact that the paradigm of computing nodes communicating over an IP-based network is a successful and long-standing setup, the key technologies in today's "incarnation" of HA² are based on a redundant IP network for application and management communication; an IPMI- and SNMP-based platform management and a high availability framework in combination with standard operating systems to run system services. Mechanical equipment practices are driven by ETSI and NEBS requirements for Central Offices and include PICMG-compliant blade architectures as well as *stackable* computers. While these technologies represent *today's* state of the art, no doubt this will change in the future. The philosophy behind HA² is therefore not limited to the current set of technologies but remains open to incorporate upcoming standards and technological developments.

Platform independent framework

HA² is designed to be a framework of specifications, standards and well-established practices on various levels characterized through standardization bodies and consortia like IETF, IEEE, PICMG, ETSI, NEBS, ITU-T, and others. HA² products shall stay within this framework. It is the intention to limit the degrees of freedom to a consistent set of architectural building blocks. This approach offers better transparency and is able to avoid end-of-life situations. Deriving a consistent set of these architectural requirements can only come from a set of well-defined user cases describing the majority of situations over the life of the system.

The operator view

It is the operator's motivation to provide resources to the application by administering, provisioning, and managing the underlying platform. The operator is aware of the finite performance and reliability of available resources and can therefore manage a collection of resources to deliver sufficient performance and reliability to his application. (All following items in italic style describe conceptual parts of the HA² specification.)

An *operator* in the HA² context is a collection of software policies in combination with a user interface. Manual intervention is typically limited to system upgrade and reconfiguration as well as maintenance of defective components.

The operator view is very much a hardware-oriented view. Figure 2 shows an example of a physical system instance with a *platform controller* responsible for monitoring and controlling all hardware components within his domain.

The conceptual *platform controller* does not run any hardware management policies like fail-over in case of a hardware failure, rather, it provides the interface to a higher *platform management* instance. The interface between *platform controller* and *platform management* is defined as Simple Network Management Protocol (SNMP). The *platform controller* is typically accessed via the Ethernet backbone but also offers a private access for outband management if the network is not available.

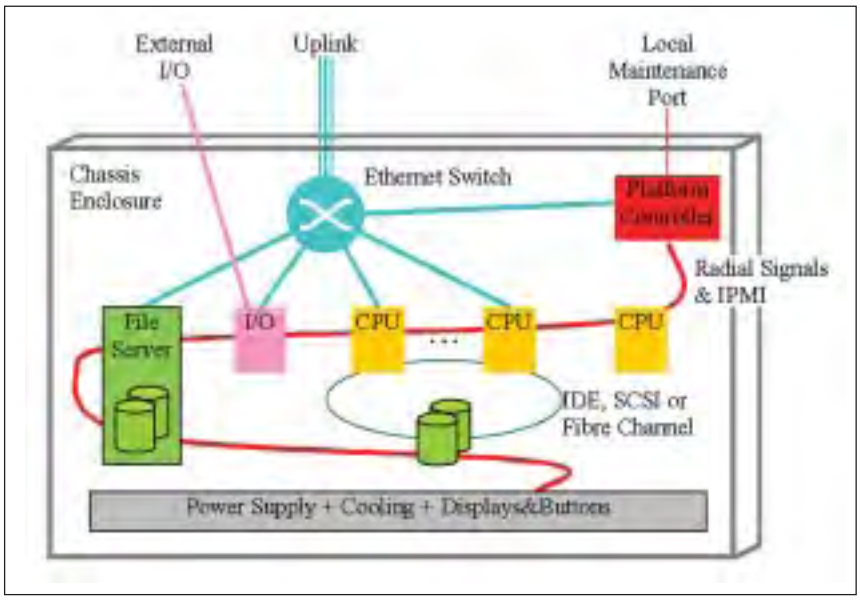

Figure 2. A chassis implementation example

The *platform controller* uses *radial signals* to power up individual hardware *modules*, to check health of the *modules* and to power down individual *modules* if needed. *Radial signals* are used for reliability reasons.

Beside this mission-critical function, the *platform controller* shall assist in monitoring and administration of this system. An out-band bus (I2C or RS-485) with IPMI protocol on top gives the *platform controller* access to all *modules* of this system. Via IPMI, the *platform controller* can read the inventory data of the individual *modules* and can read and modify the configuration.

Architectural consequences

While the presented views clarify the use of a system from different angles, it is the duty of the system architect to condense all of the considered use cases – and thus both the application and the operator view into a single architecture. Two vehicles can help in this quest: dedicated measures on the platform level to secure *availability*, and an orchestrating and integrating *middleware* layer mediating and providing this functionality to the application.

Availability aspects

Uninterrupted availability of *services* at most times is one of the key requirements of the carrier grade market. Systems supporting this requirement are called highly available. Several standard methods and practices promoted by HA^2 to scale and adapt the levels of availability required by an HA^2 instance. Most notably, redundancy of critical components and maintainability of defective components within the running system play a major role, as does the isolation of defects in so-called fault domains.

Middleware aspects

Platform management, network management, and *cluster management* were already used as terms to summarize the key tasks of a high availability management layer. The *HA middleware* is the building block that implements this functionality. It is placed between a standard OS and the application, and is seen as an operating system independent horseshoe, which can be extended with regard to functionality (see Figure 3).

Required functions are:

- A system-wide communications abstraction
- A unified single-system image
- A system-wide policy-driven fault-management
- A standardized high availability API to a scaleable cluster manager
- A well-defined local and remote management

Scalability in size and HA features

The architecture as previously described defines a frame that shall not be violated by specific product instances. However, it enables someone to scale availability from non-redundant implementations up to 5-nines and beyond. The reference implementations listed here define a minimum feature set – still within the frame – to specifically address high availability requirements:

1. A *stack* of non-redundant computers connected to two redundant network planes plus a simple HA middleware. This "simple" HA middleware is able to detect insertion, removal and crash of individual computers and switch fabrics just via standard IP-network protocols. For example, this is a cost-efficient solution for small applications – with requirements for two to four CPU cards – where savings are based on a low entry cost for stacked systems and little HA software support.
2. Same as (1) plus extensions to the middleware. These extensions provide inventory data of the boxes and provide deeper hardware diagnostics such as power supply and temperature sensor input to the HA middleware.
3. Same as (1) plus a non-redundant out-band management bus (ICMB) to collect inventory data, read sensors and remotely power up and power down individual boxes. Each ICMB node implements a hardware watchdog to isolate the node from the bus. Depending on implementation, this configuration may also offer a post mortem access for failure analysis.
4. A PICMG 2.16 system with two redundant network planes and a single *platform monitor* to read the status of field replaceable units (FRUs) that are not connected to the network (PSU, fan-tray, temperature sensors). A defective *platform monitor* does not affect the application and can be replaced in a running system. A *platform monitor* may be installed twice for redundancy. Due to the limitation to pure monitoring, both examples of the *platform monitor* can run simultaneously and do not require any fail-over procedure. It's accepted in a non-redundant implementation that connections to system *modules* break down until replacement of the defective *platform monitor.*
5. A PICMG 2.16 system with two redundant network planes and two redundant *platform controllers.* The *platform controllers* implement the functionality of the *platform monitor* as described for (4). The *platform controller* makes further use of the PICMG hot-swap signals to detect insertion, removal and health of each board and to power up boards and power them down. The *platform controller* signals via *chassis*

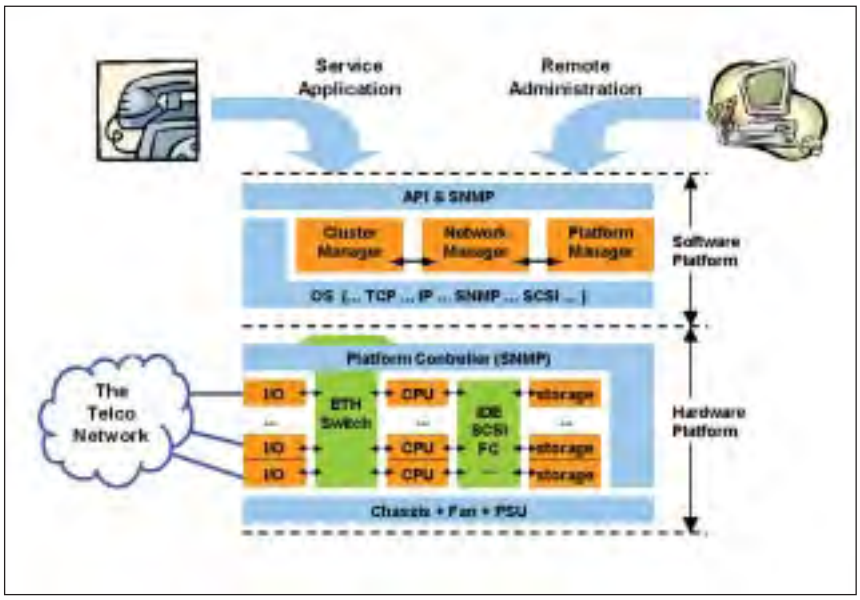

Figure 3. A system block diagram

LEDs for each individual FRU if it is "in-service" or "out-of-service." If "out-of-service" an *FRU* may be safely removed.

In a PICMG 2.16 environment an IPMI connection from the *platform controller* to all node boards is not mandatory but a recommended extension to (4) and (5). IPMI offers outband access to inventory data and sensors. IPMI within a PICMG 2.16 system is non-redundant and shall therefore never be used for life-critical control purposes.

So, based on the super standard HA² as a selected choice of open standards for architecture and interfaces, it is possible to define a virtually unlimited number of configurations that are based on the same or interoperable building blocks to facilitate cost-efficient HA solutions, and will meet the customer's availability and manageability requirements by selecting only required features. This offers OEMs an efficient and best time-to-market service.

Applicability to control and transport plane applications

Revolutionary high density server – 112 SBCs per 40U rack

As a real-world example of an HA² implementation, Force's Centellis CO 25000 is a revolutionary product, in that it is designed for server and Control Plane applications in the NGN requiring high CPU density. With typical computer racks – in datacom and ETSI telecom – providing 600mm depth, the Centellis CO 25000 is designed at a depth of 255mm with no need for rear I/O connectivity. Designed for back-to-back mounting, it allows for two systems – 14 SBCs each per 9U or eight systems easily fitted in a 40U rack, which equates to 112 SBCs. This compares to 40 SBCs per 40U using stackable 1U systems. The revolutionary concept of a 255mm deep and NEBS Level 3 design makes the Centellis 25000 unmatched in density and cost efficiency.

The Centellis CO 25000 realizes the desired high availability requirements in a PICMG 2.16-compatible "Ethernet-in-a-box" design using the established CompactPCI 6U form factor for SBCs and I/O cards. (See Figure 4 for a picture of the system loaded with Intel processor-based SBCs.)

The Centellis CO 25000 supports an advanced platform controlling and management infrastructure, one of the important features of the HA² specification (Figure 5). This enables the previously established operator view, primarily the *platform controller* and parts of the *platform controller* functionality, to provide base availability mechanisms within the hardware platform and network infrastructure *fault domain*.

Based on radial signals according PICMG 2.1 (hot swap), the platform management controller in the combined service processor/switch board (SP-SW) will monitor and control the node slots, power supplies, fan tray, and the maintenance panel. This enables the detection of HA²-compliant *FRU* states such as *present*, *fault*, and *failure* and provides capabilities to interface to remote management software through SNMP. With each PICMG 2.16-compliant fabric blade integrating switch as well as platform controller and management functionality, automatic fail-over for platform control and management itself is handled autonomously in an active/standby fashion.

With these capabilities, the Centellis 25000 platform management architecture is modeled exactly to follow the High Availability Architecture and supports, for example, the fail-over scenario exemplified in the previous section.

So although the Centellis CO 25000 is a high-density design, it meets all requirements also for a full HA implementation of 99.999 percent or better service availability.

Figure 4. The Centellis 25000 – 9U high-density server platform – with 14 server blades

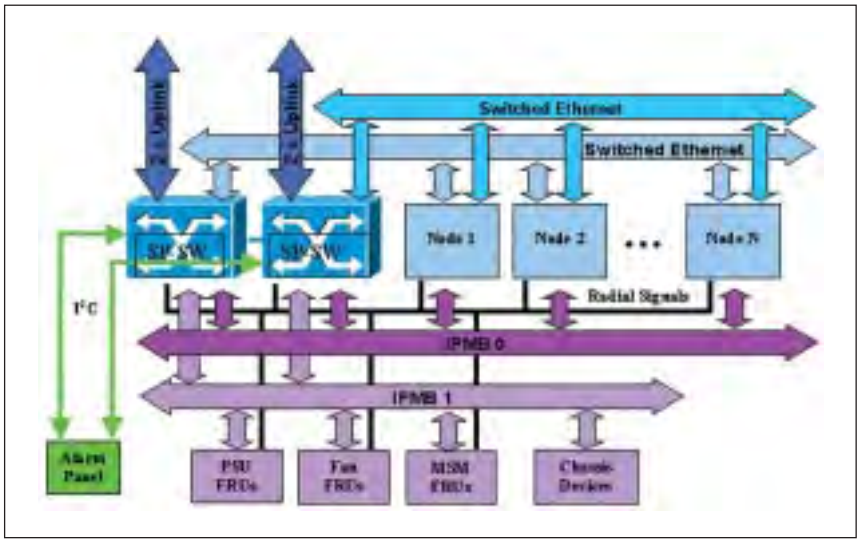

Figure 5. Centellis 25000 platform management architecture supporting HA²

Transport plane HA platforms

Within the same HA² framework, the Centellis CO 21000-12U offers PICMG 2.16 compliance packaged to support rear I/O, typically required in Transport Plane applications (see Figure 6). Sample applications for this are media gateways or wireless 3G radio node controllers, which have a large number of and varying types of interfaces.

A key difference of Transport Plane applications is the more heterogeneous system architecture compared to Control Plane equipment. Here it is important that a vast number of different I/O cards be easily integratable in the HA framework. Also, to this aspect of HA solutions, HA² offers different defined choices, to select the most efficient implementation.

Figure 6. Centellis CO 21000 – 12U platform for control and transport plane applications

Also of prime importance is the ability to route data traffic for VoIP, VoATM, or RNC over the PICMG 2.16 Ethernet backplane. Force has a patent-pending technology to efficiently address this, too – enabling 2.16 for even more transport plane applications.

Conclusion

This article has introduced the *High Availability Architecture, HA²*, as Force Computers' attempt to streamline long-term systems development. In this respect, HA² represents an umbrella standard consisting of various interrelated specifications for the "highly available collection of IP-connected nodes" paradigm used in a carrier-grade environment. By doing so, HA² provides longevity of design decisions while keeping cross-architectural portability intact. In detail, HA² focuses on various use cases making sure that application, operator, availability, and system designer requirements are met.

The Centellis 25000 is Force's initial HA² offering for high-density servers in a line of current and future products aimed at the carrier-grade market and featuring advanced platform management.

Mathias Renner *is System Marketing Manager at Force Computers located in Munich, Germany, responsible for marketing of strategic system programs and new system technologies. He joined Force in 1992 and held various positions in Product Marketing and Project Management. His expertise is in telecom and NGN markets and technologies. Mathias Renner holds degrees in Electrical Engineering from TU Munchen and Business Administration from FU Hagen.*

For more information, contact Mathias at:
Mathias Renner
Force Computers GmbH
Prof.-Messerschmitt-Str. 1
D-85579 Neubiberg/München
Tel: (0 89) 608 14-0
Fax: (0 89) 608 14-353
E-mail: Mathias.Renner@force.de
Web site: www.force.de