*Technology Article*

radisys®

# Evaluating COTS vs. Emerging Fully-Integrated Platforms

*By Mike Coward, VP of Strategy & Innovation*

The challenges of developing telecom infrastructure have been a topic of keen discussion for the past decade. Increasingly complex applications and higher bandwidth requirements coupled with shorter product lifecycles and smaller development teams have pushed the telecom industry to the brink.

For as long as these challenges have been part of the industry dialogue, the commercial off-the-shelf (COTS) model has been touted as the solution. If the problem is smaller development teams, buying blades makes intuitive sense; less development work should mean that a smaller team can successfully build a system from off-the-shelf components. If the problem is meeting stringent time-to-market requirements, again, it follows that buying blades should accelerate the overall product schedule as the blades are, in theory, fully debugged and ready to deploy into the application.

The COTS model is built on the assumption that development is difficult and integration is easy. Given this assumption, the COTS model makes sense: outsource the development of blades, chassis, and software components to a set of vendors that specialize in the components, and do a "quick" integration of the selected pieces. This model is appealing to system architects who like to choose from a wide variety of blade vendors and look for the best-of-breed on each component, and appeals to the executive sponsors who like the idea of building products without all those pesky development engineers.

However, the core assumption of COTS is wrong; development is straight-forward and predictable, while integration is messy, time-consuming, and highly unpredictable. This is the dirty little secret of COTS that has resulted in the delay or cancellation of many projects.

The point of integration is to ensure the various components work together and to resolve the incompatibilities. Until the integration task starts, it is not known whether any issues that surface will be minor and easily addressed or whether major issues will arise, thereby requiring substantial redesign, workarounds, or even re-opening the component selection process.

When building a project schedule, it is difficult to include integration activities given the huge range of possible durations. For this reason, many project managers include it in the schedule with a short duration and either keep their fingers crossed that any issues will be quickly and easily resolved ” or exclude integration from the schedule entirely. After all, if all of the components are standards-compliant, there shouldn't be any issues, right?

Experienced integration engineers know that solving integration problems frequently involves getting detailed technical support from the designers of the components and often requires resolving differing interpretations between the developers of various components in the system.

This challenge often exists even when the integration engineer is in the same company as the development engineers, because the COTS model divides engineering teams in exactly the wrong place and separates integration engineers and development engineers into different companies rather than housing them under one roof with a common set of priorities.

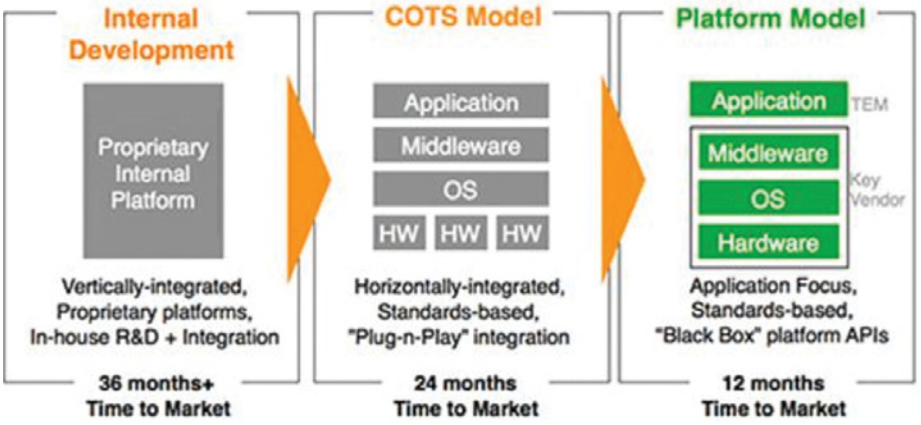
Internal Development
Proprietary Internal Platform
Vertically-integrated, Proprietary platforms, In-house R&D + Integration
36 months+ Time to Market
COTS Model
Application
Middleware
OS
HW
HW
HW
Horizontally-integrated, Standards-based, "Plug-n-Play" integration
24 months Time to Market
Platform Model
Application
TEM
Middleware
OS
Hardware
Key Vendor
Application Focus, Standards-based, "Black Box" platform APIs
12 months Time to Market

**Figure 1.** *Evolving NEP development approach*

In this case, it can be impossible to get a quick answer from the vendor company, and the ability to share information may be hampered by corporate IP protection policies, differing project priorities between the two companies, or in a many cases, a desire to “protect” the development engineers from the customers and shield them behind layers of field engineers and customer support systems.

Finally, the COTS model often puts system integrators at cross purposes with the blade vendors. During the integration process, it is often the case that the best way to resolve an integration issue, or to make the platform more consistent and “user friendly”, is to make a change to the underlying product, either to change some hardware implementation or add some software feature.

This is in direct conflict with the COTS model of aiming to sell a single standard product to multiple customers” because as soon as customization is required, many of the benefits of COTS start to disappear.

Mindful of these challenges, the largest network equipment providers (NEPs) have embraced COTS in a very specific fashion that highlights the challenges they see in COTS and provides an instruction manual for smaller vendors (Figure 1 above).

The major NEPs have created common platform teams that build application-ready platforms using blades sourced both from third-party vendors and internal engineering groups.

These common platform teams shield the NEP application teams from the pains of COTS integration, and they update the common platform on regular intervals with releases that incorporate new blades and new software functionality. These releases are put through a suite of integration testing that is intended to wring out any incompatibilities between blades, BIOS versions, and software components.

In addition, the common platform team often develops layers of software that provide services such as diagnostics, blade configuration, firmware updating, and software distribution. These services allow the application teams to quickly port new applications onto the platform, and the overall approach is to simplify the tasks for the application teams, allowing them to focus on the needs of the application and not on the nuances of low level blade and shelf manager interactions.

It is interesting to contrast this approach with the approach to COTS attempted by many startup NEPs. Most telecom startups have some new application in mind, and have a small group of developers with deep expertise in that application and the end market.

Mindful of the cost and time-to-market impacts of developing a proprietary platform, they embrace open COTS platforms and buy into the rhetoric suggesting that all the components will interoperate without issue and that the bulk of their efforts can focus on selecting the best-of-breed hardware and software components, quickly integrate them, and then work on the application. Many of these companies assign one person as the "integration engineer", or even worse, assume that the application developers can do it in their spare time.

The reality is often sadly different; the integration effort soaks up more and more time and effort from the engineers working on it, diverting resources from the and the effort quickly expands to include tracking action items with multiple vendors and managing a schedule dependent on four or five separate vendors, each of whom announces a slip every month or two, leading to schedule juggling on a weekly basis.

Learning from the largest equipment vendors, one solution is to build an internal platform team, staffed with strong architects who can sort out the intricacies of how the platform should work, and equipped with strong vendor management skills. However, most small NEPs can't afford a common platform team with 10 or 20 people, let alone the 100+ found in the largest equipment manufacturers. Where does that leave them?

These industry challenges aren't magically going to vanish; the requirements to get complex applications into the market quickly with minimal development teams are real requirements and returning to the model of developing the project completely in-house isn't a practical solution.

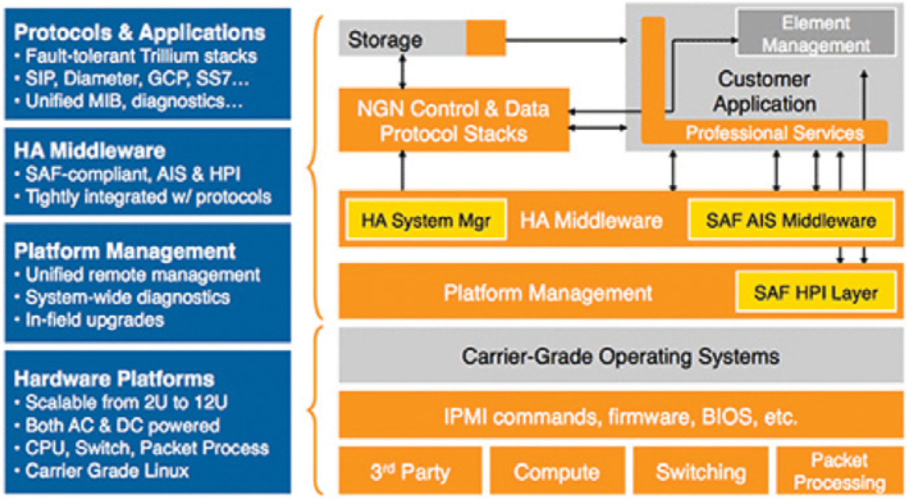
Protocols & Applications
• Fault-tolerant Trillium stacks
• SIP, Diameter, GCP, SS7…
• Unified MIB, diagnostics…
HA Middleware
• SAF-compliant, AIS & HPI
• Tightly integrated w/ protocols
Platform Management
• Unified remote management
• System-wide diagnostics
• In-field upgrades
Hardware Platforms
• Scalable from 2U to 12U
• Both AC & DC powered
• CPU, Switch, Packet Process
• Carrier Grade Linux
Storage
Element Management
Customer Application
NGN Control & Data Protocol Stacks
Professional Services
HA System Mgr
HA Middleware
SAF AIS Middleware
Platform Management
SAF HPI Layer
Carrier-Grade Operating Systems
IPMI commands, firmware, BIOS, etc.
3rd Party
Compute
Switching
Packet Processing

**Figure 2.** *FlexTCA Fully Integrated System*

Fortunately, a solution is now available in the form of pre-integrated systems that include hardware, software, and reference applications.

These fully-integrated systems include a set of components that have been tested together and tuned to operate as a system, as well as a set of value-add software that includes platform management, high availability middleware, unified management, and protocol stacks. A block diagram for a representative pre-integrated system is shown Figure 2 above.

Pre-integrated systems offer the best time-to-market available. As a rule of thumb, applications built on pre-integrated systems tend to reach the market within a year, while systems built on COTS components often 24 months and proprietary platforms as much as four years. Pre-integrated systems have been called the third phase of telecom infrastructure development. Pre-integrated systems based on open standards leverage the benefits of COTS, but offer a compelling solution to the challenges posed by the COTS model. Pre-integrated systems removes the risk of integration and schedule uncertainty that plagues projects using the COTS model, while delivering on the accelerated time-to-market and reduced development costs that pushed customers to the COTS model in the first place. Given these advantages, it looks like the third phase of telecom is here to stay.

## Corporate Headquarters

5435 NE Dawson Creek Drive
Hillsboro, OR 97124 USA
503-615-1100 | Fax 503-615-1121
Toll-Free 800-950-0044
www.radisys.com | info@radisys.com

February 2010