Blade Server DSM Analysis

Lisa Curhan

As a product engineer for Volume Products operations at Sun, I act as an organizational interface between hardware and software design groups and the manufacturing organization. The engineers in my group need to be product generalists and to know enough about both the hardware and software architecture to design good test processes for manufacturing and act as information conduits between the external manufacturers and engineering. We also try to influence the hardware and software design for ease of manufacture and test. Therefore we have a high interest in the architecture of the system, and how the system development process evolves.
Outline

- What is a Blade Server?
- High Level Object-Process Diagram
- Organizational Resources
- Project Challenges
- Component Relationship Diagrams
- Initial Component DSM
- Partitioned/Torn Component DSM
- Present Organizational DSM
- Comparison of Organization and Architecture
- Project Management Implications
- Conclusions
What is a Blade Server?

As you can see, the appearance of a blade server chassis looks like a library shelf full of books. Each “book” is an independently functional computer system. The chassis supplies power, networking capability and system management in addition to the blade modules. The idea behind this form for a server is increased density and expandability. The challenges are cooling, interconnecting the system and management. This type of system has proven popular for financial institutions and forecasts are that the market will grow. This kind of system was actually popularized by startup companies (eGenerra and RLX). The blade project in development is a distant cousin to this one, which is in production.
This very high-level object-process diagram shows how the hardware is fairly well modularized by major process, but the software and firmware effort is spread over many processes. The split between software and firmware seems arbitrary, but it is a traditional organizational division.
The software effort for this project is spread all over the globe, and there are 8 to 10 software managers involved. The hardware effort has several groups involved, but most of the resources are in Massachusetts. This makes software coordination a complex effort for the project managers. Communication between geographically distant and organizationally distant groups is somewhat constrained, which can lead to rework and delay.
This is a very generic representation of the software components on the blade showing interactions with the hardware and each other. Even in this very simplified form you can see that more than one level of the software interacts with the hardware. I’ve portrayed the hardware as a single block because it’s basically all coupled internally.
Here is a similar chart of the system controller. You can see it has monitoring functions and its upper levels communicate externally to the chassis. The SC talks to both the blades and the switch. A complication of this project is that all the hardware is basically coupled due to both electrical and mechanical interactions. I haven’t bothered to show a hardware diagram because you’ll see the hardware on the DSM as a highly coupled block. Also, I’m more interested in the hardware/software relationships which usually aren’t as closely examined.
The switch module has many layers of software to handle communications both internally and externally to the chassis. Notice that it handles communication between blades.

It may not be a good commentary on our project documentation, but I had to build block diagrams similar to these myself from interviewing. There was no single document that showed these relationships.
The initial DSM is ordered by firmware and software associated with each major functional board, with the hardware as a block at the end; this is somewhat similar to the organizational arrangement of this project. It is actually not a bad clustering, except that the hardware and software interactions are clearly not taken into account. This clustering is roughly analogous to the project team structure.
I had a lot of trouble with the partitioning and tearing of this DSM, as there are so many interactions. This attempt is not ideal, but does present some interesting opportunities. The upper left hand corner of the green square presents a cluster involving SC and Switch firmware that is overlapped with the SC and switch part of the hardware cluster (first purple square inside the green square). This shows us that the hardware cluster, which is tightly coupled, is also coupled with a set of key firmware and software applications. You can note this again by observing the “torn” hardware elements on the right hand side of the green square. I would have preferred that the blade self test turned out to be blocked with the blade hardware at level zero, but it is in the larger square due to the fact that the coupling between hardware components is even stronger than that between firmware and hardware. The larger purple square under the hardware group is what I call the “fault management” cluster. It shows that the fault management software is tightly coupled to several other pieces of code. The last purple group on the right/bottom is an “external communication” cluster. This involves code that works to communicate between systems.

I think the main advantage of this partition is that the large group in the middle has few items outside of it that it depends on but which are not inside the square.
This organizational DSM was not based on a team communications survey. Instead I applied a few simple rules to the organizational structure and which components were assigned to the people in different parts of the organization. I wanted to see where communications might be hampered by organizational and geographical barriers. So I used the following rules for assumed interaction:

0 = most interaction for groups with the same director at the same site
1 = groups within the same site with different directors
2 = groups with the same director at different sites
blank = groups with different directors at different sites

This probably exaggerates the known problem with geographical and organization distance (especially geographical distance) but it shows where there may be issues clearly (as blank or pink spaces within the outlined project teams.)
This comparison of an organizational DSM arranged in the same order as the functional partitioned blocking shows that the fault management function has some challenges with regard to geographical and organizational impediments to communication.
Conclusions

- Present teams may work as long as hardware group acts as integration team to software activities. They must remain involved and software teams, especially blade software, must stay on top of hardware issues.
- Switch and SC teams may want to form some working groups to deal with fault management issues and other linkages or form a cross-functional fault management team.
- Blade and Switch Software teams have some geographical and organizational communication challenges to deal with.
- Fault management function has a lot of communication challenges as well.

Other conclusions:

This sort of project would be most effective when architecture has been discussed, but project teams are not firmed up. It usually unwise to re-arrange teams in mid-project.

I need better graphics software for presentations to deal with images like this!

PSM32 student version not sufficient for a project of this size. Also needs better import/export functions.
Project Extension

- Get software that can work with larger datasets to look at real 70x70 relationship matrix.
- Extract better information on strength of functional relationships from project team.
- Pursue issue of insufficient platform-level software architecture documentation with team. I think a minimal investment in documents would save the team time over the course of the project.

I was worried that my component mergers needed to fit in a 40x40 matrix to use the available software led to some missing information or even false conclusions. For example, the geographical and organizational challenges may be even greater that shown because there were some merged functional component groups in this component list which are actually divided between geographical locations.
Questions?