WHITE PAPER
Intel® Cluster Ready
High-Performance Computing

Mass-Producing Your Certified Cluster Solutions

Use Intel® Cluster Ready processes to help you make the most of your engineering resources. Following the processes will enable you to sell several different configurations of clusters from each solution you design. By varying the hardware while maintaining the same software stack, you can gain a broader range of cluster products to sell without engineering each one "from scratch" (see Figure 1). As long as the software stack remains the same and works on each one of the hardware configurations, you can be confident that your clusters will interoperate with registered Intel Cluster Ready applications. Your customers can purchase your clusters with that same confidence, knowing they'll be able to get their applications up and running quickly on your systems.

Overview of Production-Related Activities

For original equipment manufacturers and platform integrators, the Intel Cluster Ready program can be divided into three types of activities: creating recipes, maintaining recipes, and copying or mass-producing recipes.

Creating and certifying recipes

A recipe is a repeatable method of building a cluster. It consists of key elements that you define: the hardware components, the software stack, and instructions for building software on the hardware. After you have created a specific build method, the next step is to certify it on specific hardware. The certification process includes building a cluster using the recipe, and then performing a series of technical steps to ensure that the cluster is Intel Cluster Ready–compliant. If the cluster passes, your recipe is certified. Having obtained certification for a recipe, you can use it to produce certified clusters for sale to your customers.

Maintaining recipes

Maintaining a recipe means keeping it current with market needs. Eventually, a cluster recipe software stack goes out of date and must be refreshed. It is likely that each of the software components in the stack will go out of date independently of the others. New or updated versions of the operating system and other components in the software stack may become available within weeks of the recipe certification. At some point, you will probably want to upgrade your recipe to the latest versions of the components.

Incremental updates to existing software components are allowed, but not additions of new software or major changes in existing software. For example, changing from Red Hat* Enterprise Linux* 5, Update 4 (RHEL 5U4) to RHEL 5U5 is an allowable refresh. The change can be made without recertification as long as the recipe continues to pass Intel® Cluster Checker health checks. However, a change from RHEL 4U8 to RHEL 5U5 would constitute a major software modification and would require recertification of the recipe.

Table of Contents

Mass-producing recipes

Manufacturing Intel® Cluster Ready solutions consists of taking customer orders matched to certified cluster recipes, and then producing and verifying systems based on the certified recipes specified in the orders. Certified recipes contain specific hardware components, but customers can order permutations of the hardware—for instance, more memory per node. The verification process accommodates a variety of hardware changes, or variances, without requiring the solution to be recertified. However, a cluster that contains variances from the recipe must still pass the Intel® Cluster Checker health test before it is released to the customer. Some of the Intel Cluster Checker settings may require modifications to reflect those variances.

The rest of this guide focuses on methods and suggestions for mass-producing instances of cluster recipes. It is intended to help you translate Intel Cluster Ready certified recipes into volume cluster sales.

Roles and Responsibilities for Certified Cluster Production

Roles and responsibilities for producing certified clusters can be separated into three functional areas and their corresponding teams of workers: engineering, sales, and manufacturing. Your company may give these groups different names. Also, multiple roles in the process may be handled by one department in your company.

Recipe engineering

Creating cluster recipes, certifying them, and keeping them current by means of software refreshes are all part of recipe engineering. Intel suggests that your engineering department establish a recipe development and deployment process encompassing these activities.

The engineering team maintains a set of refreshed and certified recipes that are ready for your company to use in fulfilling customer orders. If a software component is added or removed, triggering a recertification, your company's engineering team handles the recertification process. The

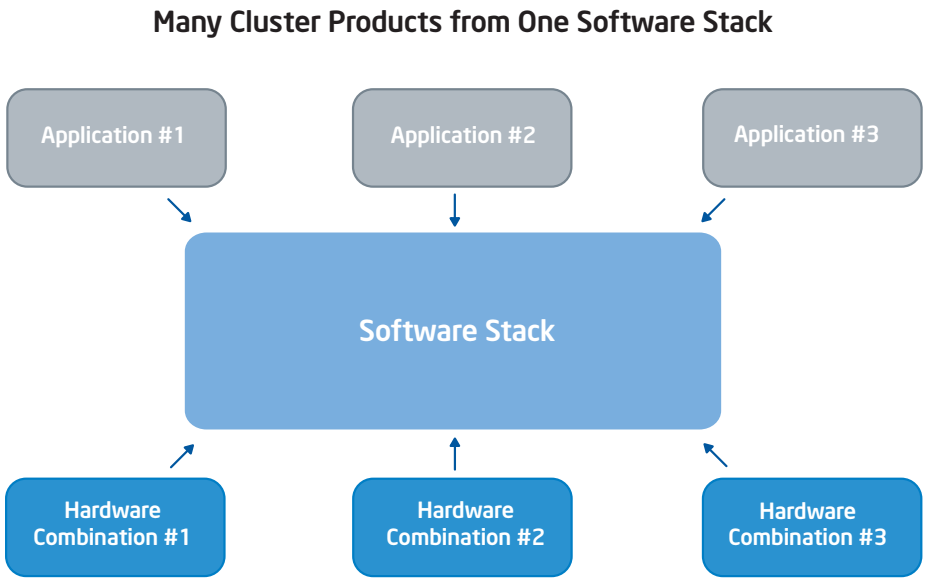

Figure 1. Many cluster products from one software stack

Once you have certified a cluster recipe, as long as the software stack stays the same, you can change some of the underlying hardware to create different configurations.

engineering team should also manage the Intel® Cluster Checker fingerprint files that are used to verify that a certified cluster matches a recipe. Your engineers must change the fingerprint files whenever software components are added, removed, or refreshed.

Selling Intel® Cluster Ready Solutions

Intel® Cluster Ready lets you maximize the return on engineering investment by enabling a certify-once, sell-many environment. Customer orders must map to cluster solutions that are covered under a certification and only allow deviations that do not require a new or updated certification at deployment. The recipes certified by engineering have set software stacks and hardware components. The software stack must remain the same as the certified recipe to ensure there is no additional integration work or certification needed when deploying the order (see Figure 2). The hardware components are allowed to vary as long as the software stack remains unaffected and will still pass all verification checks at deployment.

The marketing and sales teams play a key role in the ability to mass-produce Intel Cluster Ready solutions. Certified solutions need to be advertised as a complete solution—software stack on the hardware—and not as parts requiring the customer to determine the best method for putting them into a solution. Customer orders should always start with the software stack locked to match the certified recipe and only allow common hardware variances that can cover a wide range of solutions. Some common hardware variances that are allowed without breaking a certification include different processor frequencies, more memory, or a larger number of nodes. These hardware variances may need adjustments to the Intel Cluster Checker configuration files at deployment, but this effort is much less time than full certification, and customer orders can go directly to your manufacturing process without any engineering involvement. See *Intel® Cluster Ready Allowed Hardware Variances* for more information on hardware components that can vary without forcing a new certification.

Manufacturing processes must have access to the exact recipe instructions and resulting Intel Cluster Checker input files that correspond to the certified solutions you sell. When a salesperson submits a customer order, it must indicate which certified recipe to use and include any variances in the hardware requested by the customer—enabling the customer order to go immediately into manufacturing and follow a standard procedure that should speed up the deployment of the order. The certified recipe instructions provide the blueprint to repeat the original solution, and Intel Cluster Checker gives you a way to verify that you have built a copy and are providing a working system to your customer.

Not every customer order is a good match to an Intel Cluster Ready certified solution. Customers that require a highly customized solution with very specific requirements for the operating system, provisioning software, or specialized hardware components may not fit any certified recipe you currently have. Although these systems can be Intel Cluster Ready, they probably need to be fully certified, which is an engineering function that can take longer than deploying an existing certified recipe.

In contrast, customers that provide only a list of required applications for their solution are easily mapped to your existing cluster recipes. These customers are focused on getting the correct hardware configuration that meets their needs based on the types of applications and workloads they run. These are the types of variances allowed under a certified

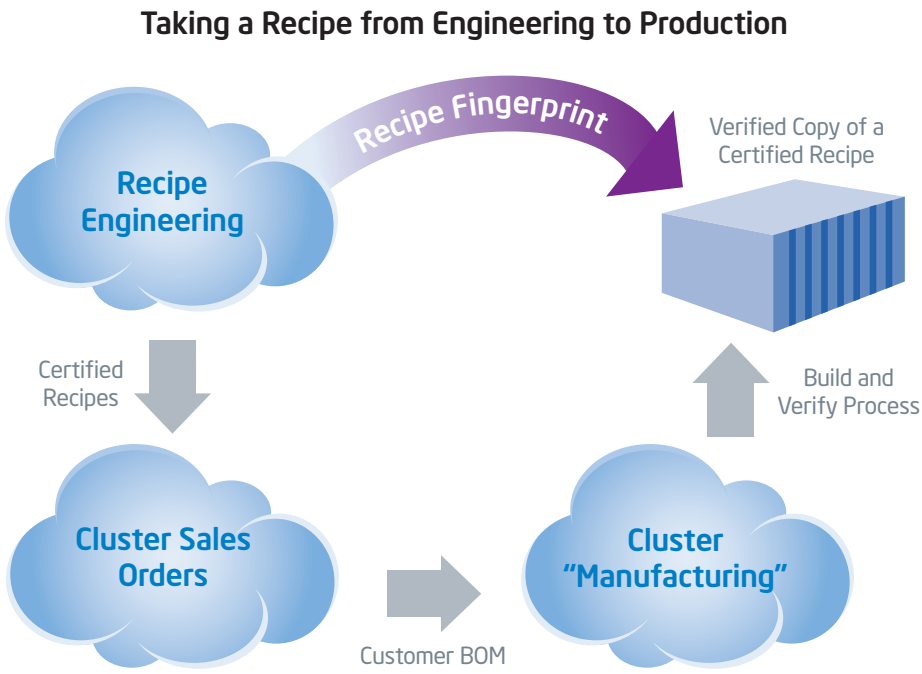

Figure 2. Taking a recipe from engineering to production

The hardware and software portions of a certified cluster both originate with a recipe in the engineering department but take different paths to the end product. The hardware can be altered by the sales team, while the software must match.

solution—ones that relate to the specific optimizations in hardware configurations that improve performance or enable a given workload for a given application. Understanding these points is critical for enabling volume production of clusters with Intel® Cluster Ready.

Manufacturing clusters

Once an order has been matched to a recipe, your company's manufacturing team uses that recipe and any variances noted by the sales team to build the cluster or clusters for the customer. The manufacturing team is responsible for obtaining the necessary recipe documentation, software, and files to build the ordered cluster. As the cluster is built, the team will swap-in any variant hardware components specified by the sales team.

Your company's manufacturing team or field installers have the additional responsibility of adjusting Intel® Cluster Checker to account for the variances between what was originally certified and what the customer has ordered. Team members should be trained to make the proper modifications to the relevant Intel Cluster Checker test module configurations. Intel Cluster Checker version 1.5 enables some variances to be automated. See the *Intel Cluster Checker User's Guide* for more information on those features.

In limited circumstances, the manufacturing team itself may make hardware changes. For instance, when building multiple servers based on the same recipe during a cluster production run, the team may run out of a part from one vendor—a memory component, for example—and may have to substitute an equivalent part from another vendor. This is usually an allowable variance, depending on the specific hardware involved. The team will have to modify the Intel Cluster Checker test module configuration to reflect the change.

Quantity	Item	Manufacturer	Model
32	Intel® server board S5520UR	Intel	S5520UR
32	Intel® Server Chassis	Intel	SR1600URSASBPP
32	Intel® HDD Backplane	Intel	ASR1600PASBP
32	2 Intel® Xeon® Processors	Intel	Intel Xeon Processor X5660
32	6x 2GB DDR3 PC3-10600	Micron	MT18JSF25672PDZ-1G4D1
32	500Gb SATA Hard Disk Drive 3 Gbs	Seagate*	Barracuda* ST3500320NS
32	InfiniBand* Host Channel Adapter	Mellanox	AXX1BDDRPT (MT25204)

Table 1. Partial example of the bill of materials from an Intel® Cluster Ready certified recipe

Quantity	Item	Manufacturer	Model
16	Intel® server board S5520UR	Intel	S5520UR
16	Intel® Server Chassis	Intel	SR1600URSASBPP
16	Intel® HDD Backplane	Intel	ASR1600PASBP
16	2 Intel® Xeon® Processors	Intel	Intel Xeon Processor X5660
16	12x 2GB DDR3 PC3-10600	Micron	MT18JSF25672PDZ-1G4D1
16	750Gb SATA Hard Disk Drive 3 Gbs	Seagate*	Barracuda* ST3750330NS
16	InfiniBand* Host Channel Adapter	Mellanox	AXX1BDDRPT (MT25204)

Table 2. Partial example of the bill of materials from a customer order

Scenarios for Performing Intel Cluster Checker Testing

Your manufacturing team loads the Intel Cluster Checker tool onto each cluster during the software build. The Intel Cluster Checker testing must be run at some point following the build and prior to transferring control of the cluster to the customer, but the time and place is up to your company. Scenarios for conducting the required testing include the following:

- The cluster manufacturing team performs the testing at the manufacturing site
- Your company's field installation team runs the testing at the factory or the customer's site

Component	Allowed Variance
Processor	Within the same family: frequency, cache size, cores
Memory	Vendor, speed, size
Storage Device	Vendor, speed, size, type
InfiniBand	Vendor, speed, omission
Server Board	Revision (chipset stepping)
Node Count	Up to 2N

Table 3. Allowed typical hardware variances

- Intel® Cluster Checker runs automatically when the customer starts up the cluster for the first time—this option must require no additional Intel Cluster Checker configuration by the customer

You may have another scenario that better fits your company's business model. The key is to ensure that the testing takes place before the cluster is released to your customer. Otherwise, the cluster may not interoperate with Intel® Cluster Ready applications, and your ability to use Intel Cluster Checker to troubleshoot your customer's cluster may be compromised.

Determining if Intel Cluster Checker Requires Modifications

Regardless of who is assigned to perform cluster testing, your manufacturing team and field installers should be proficient in reviewing variances and making any necessary Intel Cluster Checker configuration modifications. Your team members can determine whether modifications are needed by comparing the hardware portion of the original, certified recipe to the hardware required in the customer order.

Each Intel Cluster Ready recipe is certified with a specific bill of materials (BOM) that is defined by the recipe engineering team. An example is provided in Table 1.

A customer order may have a similar BOM but with allowed variances. For example, in Table 2, the customer has ordered twice the memory and faster hard disk drives, but only half the nodes. Variances are shown in red.

In this example, the variances in the customer order are allowed. However, Intel Cluster Checker is configured for the original, certified recipe BOM. If your team performs testing with the original configuration in place, your customer's cluster will not pass. The new customer BOM alters the performance of the cluster, so your manufacturing team or field installers must change the Intel Cluster Checker performance settings.

The Quantity column in the customer order BOM shows the customer has ordered half the number of nodes that were in the original recipe. The rule is that up to 2N nodes are allowed, where N is the number of nodes recorded for the original certified cluster. A certified 32-node recipe means that this 16-node cluster is considered certified, as would any cluster up to 64 nodes. Because the change in node count affects cluster performance, it also requires configuration changes.

Allowable Hardware Variance Categories

A variety of hardware variances are allowed, including but not limited to the examples shown in Table 2. A list of typical variances is provided in Table 3.

Allowed variances that require changes to Intel Cluster Checker are categorized based on the effects they produce. Allowed variances can cause changes in:

- Performance
- Devices tested (some tests may need to be omitted)
- Expected characteristics (for example, total available memory)

Each category of variance requires specific changes to the Intel Cluster Checker test module configurations. The next few sections describe these categories in more detail and provide examples of the corresponding configuration changes. Please note that this guide does not re-create the Intel Cluster Checker manuals that describe how to modify each individual configuration. The Intel Cluster Checker manuals are the authority for instructing how to adjust the configuration file. Intel recommends that you keep a copy of the manuals available for easy reference. Manuals are included with Intel Cluster Checker in the document directory.

Performance-Related Modules

Intel® Cluster Checker modules that *may* need to vary

Test Module	Memory	Network Device	Node Count	Processor	Storage Device
core_frequency				■	
hdparm					■
hpcc	■	■	■	■	
imb_pingpong_intel_mpi		■			
memory_bandwidth_stream	■			■	
mflops_intel_mkl				■	

Figure 3. Performance-related modules

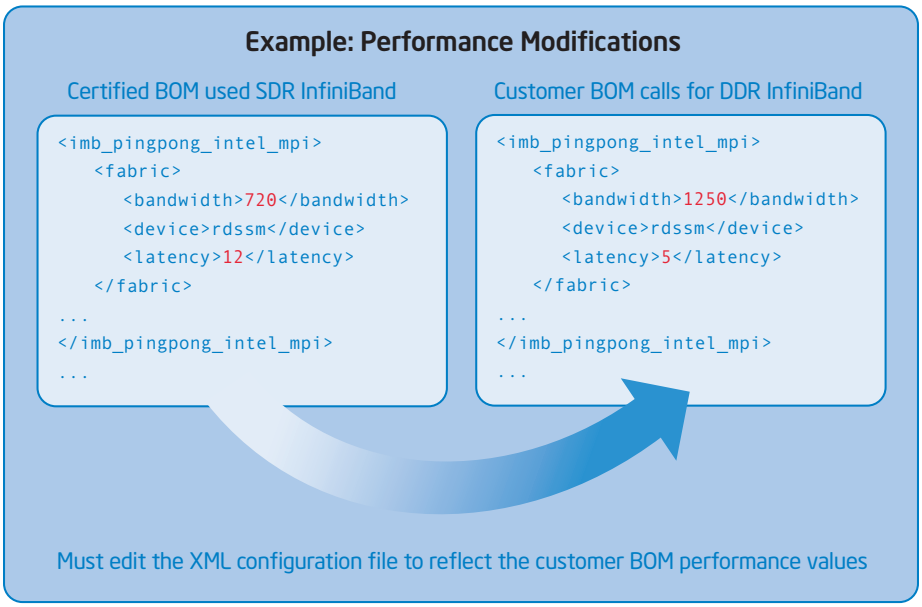

Figure 4. Example of performance-related Cluster Checker modifications

Performance Variances and Cluster Checker Modifications

Changes in memory, network devices, node count, processors, and storage devices are examples of allowable variances that affect cluster performance. These changes require configuration adjustments to Intel® Cluster Checker, and your team will need to adjust different configurations depending on the specific hardware variances involved. Performance-related modules that may need to be modified are shown in Figure 3.

To consider a specific performance-related example, suppose that the InfiniBand* hardware for a particular customer order is different from the certified bill of materials. The certified BOM used SDR InfiniBand, but the customer BOM calls for DDR InfiniBand. The two components have different performance thresholds: with the SDR configuration, Intel Cluster Checker will test to ensure that the bandwidth is at least 720 MB/s and the latency is no more than 12 microseconds, while DDR should use values of 1250 and 5, respectively. Your manufacturing or field installation team member must edit the XML configuration file in Intel Cluster Checker to reflect the customer BOM performance values. In this example, the team member would access the imb_pingpong_intel_mpi section of the XML configuration file and change the bandwidth and latency values (Figure 4).

Device Omission Variances and Cluster Checker Modifications

You may need to eliminate a network component that is part of the original recipe but is not wanted by the customer. For example, suppose one of your company's certified recipes includes InfiniBand and Ethernet, but the customer orders Ethernet only. Removing network fabric hardware from a solution is allowed, but your team must modify Intel Cluster Checker accordingly. Several modules are affected by omitting InfiniBand hardware (Figure 5). Note that the drivers and other software for the InfiniBand network fabric must remain installed as required in the certified recipe. If you removed both the hardware and software, you would trigger a recertification.

Once you omit a network fabric, the configuration XML file must be adjusted to ignore modules or subtests that are specific to that fabric. In this example (Figure 6), your team member would access the cluster section in the XML file and add the tags to exclude the specific InfiniBand test modules, such as ipoib, openib, and subnet_manager. Then, in

Omission-Related Modules

Intel® Cluster Checker modules affected by omitting InfiniBand* hardware from a recipe

Test Module	Omit InfiniBand
hpcc	
imb_collective_intel_mpi	
imb_message_integrity_intel_mpi	
imb_mpi_rt	
imb_mpi_rt_internode	
imb_pingpong_intel_mpi	
intel_mpi_internode	
intel_mpi_testsuite	

Figure 5. Omission-related modules

the test section of the XML file, your team member would comment out or remove the InfiniBand device (e.g., rdssm) entries from the network fabric tests.

Nominal Variances and Cluster Checker Modifications

Many components of clusters can vary in terms of size or characteristics, and differences also can exist between nodes when components in the cluster are similar but not identical. Components that may have nominal variances include memory, processors, server boards, and storage devices. For nominal variances, a variety of Intel® Cluster Checker modules must be modified depending on the specific hardware involved (Figure 7).

To illustrate, suppose the certified recipe BOM has 6 GB of memory per node and your customer BOM calls for 8 GB of memory per node. The XML configuration file must be edited to reflect the customer BOM component characteristics (Figure 8). Your team member would access the system memory section of the XML configuration file and change the value corresponding to the total amount of physical memory from 6 GB, or 6154464 bytes, to 8 GB, or 8388608 bytes.

Consider another memory variance, this time affecting a group of nodes. Suppose you are using the same certified recipe, which has 6 GB of memory per node, but your customer order calls for the first four compute nodes to have 8 GB of memory and the last four to have 4 GB. Your team member would first create two node groups in the nodelist file and then make appropriate setting changes in the XML configuration file.

In this example (Figure 9), the team member has placed the 8 GB nodes into a group named "bigmem," and then accessed the system memory section of the XML configuration file to specify that nodes in the "bigmem" group should have 8 GB, or 8388608 bytes, of total physical memory. The other nodes have the default amount of memory: 4 GB, or 4194304 bytes.

Example: Omission Modifications

Specific InfiniBand* test modules are excluded in the XML file

InfiniBand devices are commented out or removed from fabric tests

```
<cluster>
...
   <exclude_module>ipoib</exclude_module>
   <exclude_module>openib</exclude_module>
   <exclude_module>subnet</exclude_module>
...
   <test>
      <imb_collective_intel_mpi>
<!--     <fabric>
            <device>rdssm</device>
         </fabric> -->
         <fabric>
            <device>sock</device>
         </fabric>
         <path>/opt/intel/impi/3.1/</path>
      </imb_collective_intel_mpi>
...
```

Figure 6. Example of omission-related Cluster Checker modifications

Nominal Variance-Related Modules

Intel® Cluster Checker modules that *may* need to vary based on nominal variances in the cluster

Test Module	Memory Size	Memory Type or Vendor	Processor Variance	Server Board Revision	Storage Device Size
available_disk					■
core_count			■		
cpuinfo			■		
dmidecode		■		■	
system_memory	■				

Figure 7. Nominal variance-related modules

These are some of the hardware variances you can make to your Intel® Cluster Ready recipes to meet the requirements of your cluster customers. The corresponding changes required in the Intel® Cluster Checker test module configurations are straightforward and should be easy for members of your company's manufacturing or field installation teams to make.

The benefit of this system of variances and modifications is that it lets you leverage your engineering resources by multiplying the types of clusters you can produce from each certified recipe. For example, the ability to omit a fabric allows a single certified recipe providing both InfiniBand and Ethernet networks to be used to build an Ethernet-only or InfiniBand-only cluster. The result: more sales from the same engineering effort. It is hoped that this guide will help your company use allowable variances to obtain maximum benefit from manufacturing Intel Cluster Ready systems.

For More Information

You can find more information about the program on the Intel Cluster Ready Web site at www.intel.com/go/cluster.

For technical details, see the program documentation, including the Intel Cluster Ready Specification and *Intel Cluster Ready System Certification Procedure* manual, available online at the same location.

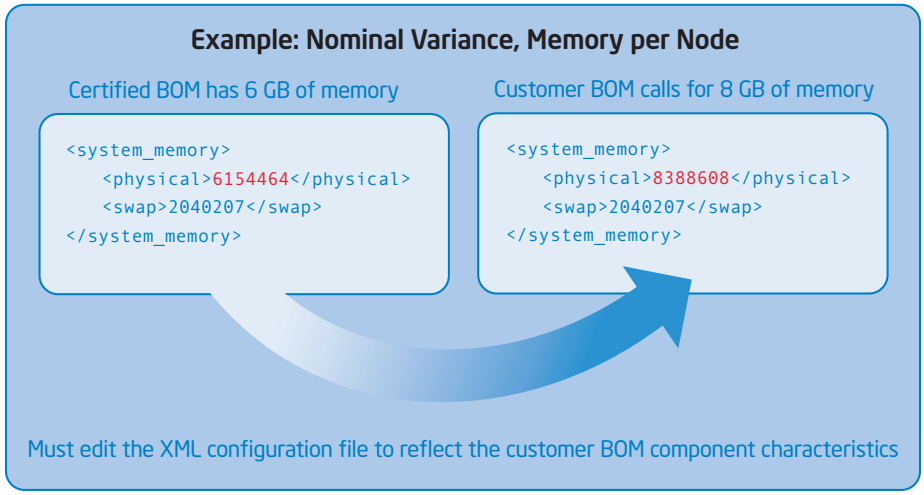

Figure 8. Example of Cluster Checker modifications for per-node memory variance

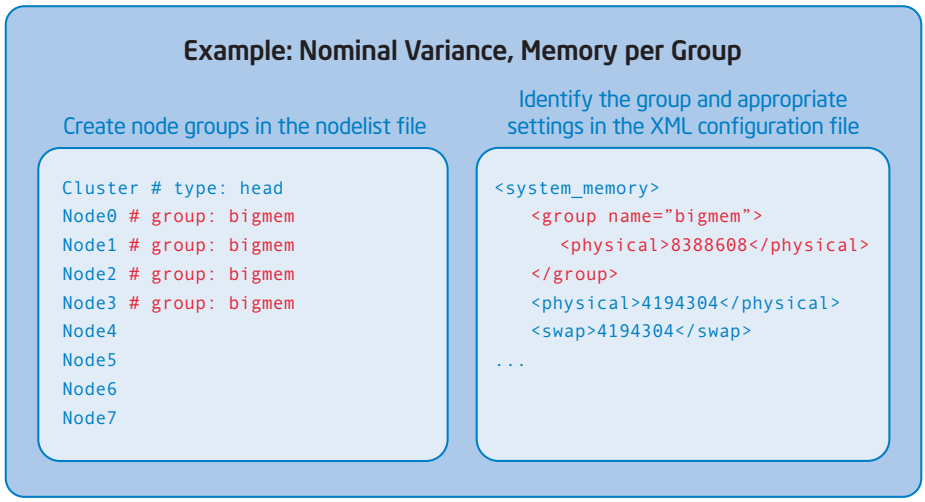

Figure 9. Example of Cluster Checker modifications for group memory variance

Collecting Intel Cluster Ready Data

Intel requests that participating companies collect Intel Cluster Ready solutions deployment data on a voluntary basis. This is a way for you to communicate your requirements to help Intel improve the Intel Cluster Ready architecture, tools, and processes. No information is collected about your specific customers and the process is designed to be quick and easy. Please e-mail cluster@intel.com to take part in this data collection or to request more information.

Printed in USA 0310/BJS/TDA/XX/PDF Please Recycle 320193-003US