Cloud Computing for Graphic Arts Printing

Lee C. Moore, Barry G. Gombert; Xerox Corp.; Webster, NY/USA

Abstract

Cloud Computing offers new opportunities to enhance digital print and variable data workflows. The combination of Software-as-a-Service (SaaS) with a scalable Infrastructure-as-a-Service (IaaS) has the potential to offer great benefits to the graphics arts value chain. Cloud computing can increase accessibility to new markets, provide higher levels of automation, and help to reduce operational costs.

In this paper we will describe ways in which digital printing workflows can harness the power of the cloud.

Introduction

Technical Drivers

In this section, we discuss the current technical environment as well as anticipated trends.

Moore's Law is changing – the maximum speed of an individual CPU is reaching physical limits. The way systems are becoming faster is through adding additional processors (horizontal scaling). To utilize additional processors efficiently, work must be executed in parallel.

On the network side of things, technology continues to increase transmission speeds at an astonishing rate. Network speed is projected to increase rapidly for a number of years. This speed increase is expected to occur for both short-haul and long-haul networks.

Cloud computing is in daily use for a variety of applications. While there is some debate as to what exactly constitutes a cloud, the concept of scalable computing, infrastructure-as-a-service (IaaS), and software-as-a-service (SaaS) are common elements. Commercially available clouds include services such as Amazon Web Services (AWS), Rackspace's Cloud Servers, and Microsoft's Azure Compute. Open-source systems exist as well. Examples include: OpenStack and Eucalyptus.

Business Drivers

Most print shops in North America have under twenty-five employees. Often they cannot afford a full-time IT staff. By moving to a cloud deployment, local IT support can be simplified.

Opportunity

The value chain for digital printing can benefit from cloud computing. Benefits gained in the print product lifecycle could start as early as the graphical design step with CrowdSourced design services such as 99designs[1]. With an increase in web-based design services, electronic submission of production print is also probable. Once a job is submitted, it is possible to construct a workflow that will perform the necessary prepress operations in the cloud. When prepress operations are completed, the print-ready file can be transferred into the shop for physical operations.

In addition to the above model, cloud can also be used to supplement local resources. Local compute resources can provide processing for typical demands but peak periods could be fulfilled "on-demand" by utilizing "cloud bursting" to external services.

By moving to a cloud model, print shops can expand their service offerings and adopt new models for business. The benefits include:

- Paying for only what they use—and consequently minimize fixed IT infrastructure costs.
- Lower the barrier to the adoption of non-impact printing.
- Shops can be positioned to easily try and offer new services (variable data, e-media, packaging …) without long term commitments.

Architecture

Technical Background

The key technical challenge is to map production print shop workflows on to a highly scalable cloud computing infra-structure. Let us make a few observations…

One of the most scalable design patterns is the Master/Worker pattern. In this pattern, a master process creates "work items" for worker processes to pick up and perform. After a worker successfully processes a work item, the result is posted and another item is picked up. This design pattern offers high scalability because an unbounded number of workers can be started and they will process in parallel. If the problem can be divided neatly into disjoint chunks, then the processing of work can efficiently be scaled to meet demand. This is "divide and conquer" style that is used by Seti@Home, Folding@Home, and other systems.

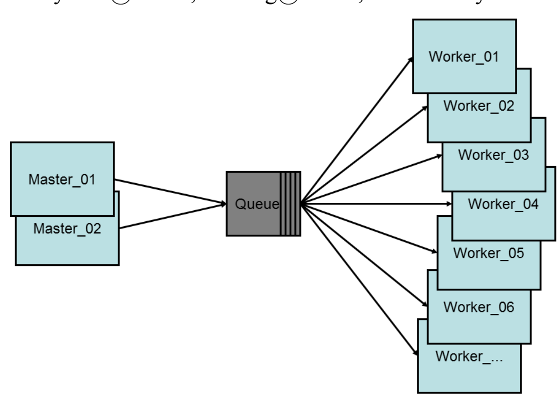

Figure 1: Master/Worker Design Pattern

[1] http://www.99designs.com – a crowd sourced graphic design marketplace

JDF

For a number of years, the printing industry (especially vendors of printing equipment) has been developing a standard entitled simply the "Job Definition Format" or JDF. JDF provides a common language that supports the lifecycle of a print job providing for the creation, command, control, and configuration necessary to enable automation and facilitate job production. JDF presents a "batch-sequential" model of processing. It allows for the composition of arbitrary sequences of elemental processes in a sequential as well as parallel processing mode. JDF workflow is expressed through a dataflow processing model.

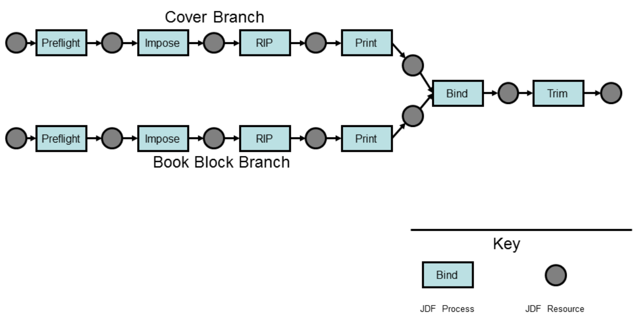

Figure 2: a JDF Workflow for printing a book using Processes and Resources and the dataflow linking between them.

In the JDF reference architecture, Process nodes in the dataflow description are executed by JDF Devices. Each JDF Device includes a work queue in front of it. This is depicted in the following figure:

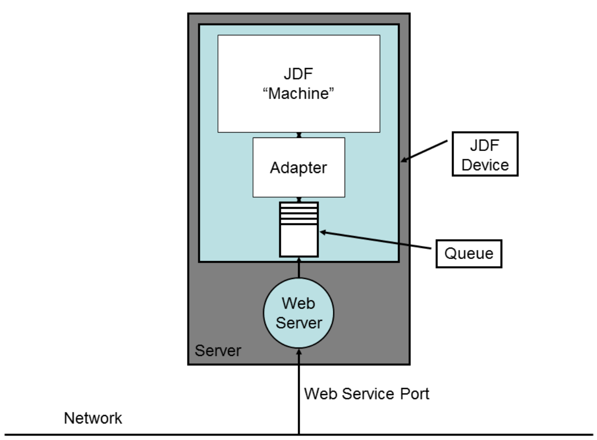

Figure 3: JDF/JMF Reference Model for network services

A JDF Controller takes a JDF description and determines what to do next. Typically it will determine which JDF Processes are eligible for execution and assign the JDF to a corresponding JDF Device or another JDF Controller. A JDF Controller is analogous to a workflow orchestrator.

One helpful feature of JDF is its philosophy of separating the flow of control from data. In the cloud it is important to not move data any more than is necessary.

Technical Insight

Combining the above, it is possible to create mappings from the JDF standard to the Master/Worker processing design pattern described previously. The queue that is embedded in the JDF Device is pulled out and is made a replicated and reliable service on the cloud. The JDF Devices are embodied in the worker virtual machines. They then pull work out of the queue service associated with their JDF Device. Instances of the same JDF device pull from the common work queue and execute the work as before. As many replicas as needed may be assigned to the same queue.

After execution, the job is typically returned to the JDF Controller queue for routing to the next JDF Device.

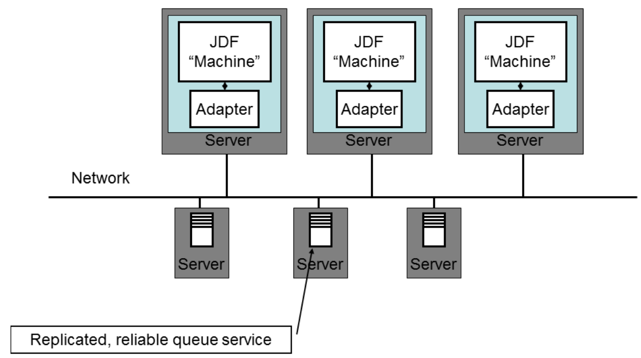

Figure 4: single replicated JDF queue feeding multiple machines

Parallelism can come in a variety of ways. At the top level, independent jobs can be executed in parallel. Within a job, there may be elements of natural parallelism—for example, the cover and book block "legs" of the workflow shown in Figure 2. Finally, documents may be similarly partitioned by pages for page-parallel processing or even by color separation.

One of the great features of JDF is the separation of Product intent Descriptions (customer oriented) of jobs from the Process Descriptions. The Product Description says what the customer wants and the Process Description says how to make it. This capability to flexibly map the customer request to specific process descriptions provides a means to inject parallelism. For the cloud, we wish to express parallelism, when possible, so as to take advantage of the parallel compute resources.

This separation of Intent from Process also provides the ability to define a workflow that is closely matched to a given shop's capabilities. Automated planning tools and techniques can be applied to ensure the most efficient use of available resources.

Deployment Architecture

Because of a print shop's firewall, the cloud does not have visibility to the devices in the print shop. Instead, the work must be pulled into the shop by an on-site agent that bi-directionally communicates between the local devices and the compute cloud.

Unlike the ability to scale compute resources, the network bandwidth to the shop is fixed and must be managed to ensure optimal utilization.

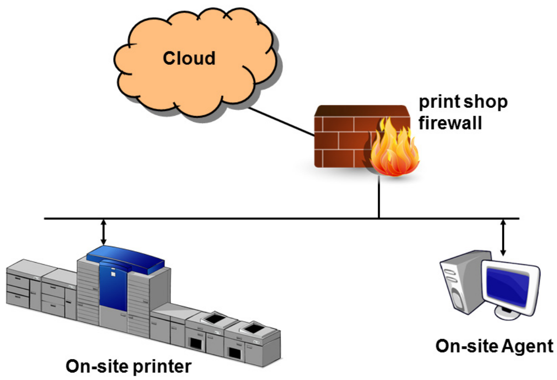

Figure 5: On-site Agent Architecture

Scaling

To achieve the benefits of a "pay-as-you-go" cloud architecture, it is important to scale the number of virtual machines up and down as work ebbs and flows. Figure 6 shows a scalable prepress system acquiring resources as the backlog of customer jobs increases. As jobs are received, the backlog becomes larger and more virtual machines are requested to fulfill the demand. Eventually, the virtual machines process the backlog and will be released at the appropriate time.

Figure 6: Scalability of prepress jobs in the cloud

Challenges

Once a distributed processing approach is taken, the issue of node or network failure must be addressed. Should a node in the cloud infrastructure fail, that node must be detected and the work it was doing must be assigned elsewhere.

Large scale computing means that we cannot manage the system infrastructure manually. We must have in place automatic processes to manage the lifecycle of activities on the cloud.

Partial products from the execution of a job will require a file system-like capability. If this system is distributed, then the execution should be planned to occur at a place that has efficient access to the resources that will be required.

While network and computer advances have made more power available, the graphics arts community is also seeing an increase in average document size.

Conclusion

The application of large scale computing to production printing has the potential to enhance the capabilities of print shops. We have shown a number of techniques for utilizing the cloud to meet the needs of printers. We feel that the promise of JDF is well matched to cloud computing, with careful adjustment. As we work through the challenges, we see great opportunity for the graphic arts industry.

References

[1] JDF Specification 1.4a, (CIP4, 2009)

[2] Armbrust, et al, "Above the Clouds: A Berkeley View of Cloud Computing", Technical Report No. UCB/EECS-2009-28 (2009) Electrical Engineering and Computer Sciences, University of California at Berkeley

[3] L. Moore, B. Gombert, Custom Book Publishing and Production Using XML, JDF, and PDF, International Conference on Digital Production Printing and Industrial Applications, (2005)

Author Biography

Lee Moore received his BS in computer science from the University of Maryland, College Park (1978) and a MS in computer science from the University of Rochester (1981). Since then he has worked in the research division of Xerox Corporation in Webster, NY USA. His worked has focused on distributed systems and imaging.

Barry Gombert received his BS in electrical engineering from Valparaiso University. His work at Xerox has focused on software systems.