Scalable Services for Digital Preservation

R. Schmidt, C. Sadilek, R. King

Austrian Research Centers GmbH - ARC, Digital Memory Engineering,
Donau-City-Str. 1, Vienna, Austria

e-mail: rainer.schmidt@arcs.ac.at

Abstract
Due to rapid changes in information technology, digital data, documents, and records are doomed to become uninterpretable bit-streams within short time periods. Digital Preservation deals with the long-term storage, access, and maintenance of digital data objects. In order to prevent a loss of information, digital libraries and archives are increasingly faced with the need to electronically preserve vast amounts of data while having limited computational resources in-house. However, due to the potentially immense data sets and computationally intensive tasks involved, preservation systems have become a recognized challenge for e-science and e-infrastructures. In this paper, we argue that grid and cloud technology can provide the crucial technology for building scalable preservation systems. We introduce a strategy for utilizing cloud infrastructures that are based on platform virtualization as a scaling environment for the execution of preservation workflows. We present recent developments on a Job Submission Service (JSS) that is based on standard grid mechanisms (JSDL [1], HPCBP [7]) and capable of providing large clusters of virtual machines. Finally, we present experimental results that have been conducted on the Amazon EC2 and S3 utility cloud infrastructure.

1 Introduction

Due to rapid changes in information technology, a significant fraction of digital data, documents, and records are doomed to become uninterpretable bit-streams within short time periods. Digital Preservation deals with long-term storage, access, and maintenance of digital data objects. In order to prevent a loss of information, digital libraries and archives are increasingly faced with the need to electronically preserve vast amounts of data while having limited computational resources in-house. However, due to the potentially immense data sets and computationally intensive tasks involved, preservation systems have become a recognized challenge for e-science and e-infrastructures [4]. In this paper, we argue that grid and cloud technology can provide the crucial technology for building scalable preservation systems. We introduce a strategy for utilizing cloud infrastructures that are based on platform virtualization as a scaling environment for the execution of preservation workflows. We present recent developments on a Job Submission Service (JSS) that is based on standard grid mechanisms (JSDL [1], HPCBP [7]) and capable of providing large clusters of virtual machines. Finally, we present experimental results that have been conducted on the Amazon EC2 and S3 utility cloud infrastructure.
conducted using the Amazon Simple Storage Service (S3) and Elastic Compute Cloud (EC2) services (AWS) [19]. The EU project Planets [9] aims to provide a service-based solution to ensure long-term access to the growing collections of digital scientific and cultural assets. Within this project, the Interoperability Framework (IF) provides the technical environment for integrating preservation services, meta-data, and archival storage elements. Components that perform preservation actions often rely on preinstalled tools (e.g., a file format converter) that are wrapped by a service interface on the lowest-layer. The Planets IF workflow engine implements a component-oriented enactor that governs lifecycle operation of the various preservation components, such as instantiation, communication, and data provenance. Distributed preservation workflows are conducted from high-level components that abstract the underlying protocol layers. A crucial aspect of the preservation system is the establishment of a distributed, reliable, and scalable computational tier. A typical preservation workflow may consist of a set of components for data conversions, storage, quality assurance, and data model manipulations and may be applied to millions of digital objects. In principle, these workflows could be easily parallelized and run in a massively parallel environment. However, the fact that preservation tools often rely on closed source, 3rd party libraries and applications that often require a platform-dependent and non-trivial installation procedure prevents the utilization of standard HPC facilities. In order to efficiently execute a preservation plan, a varying set of preservation tools would need to be available on a scalable number of computational nodes. The solution proposed in this paper tackles this problem by incorporating hardware virtualization, allowing us to instantiate sets of transient system images on demand, which are federated as a virtualized cluster. We present a Job Submission Service that is based on standard grid mechanisms and protocols, and utilized as the computational tier of a digital preservation system. Jobs are capable of executing data-intensive preservation workflows by utilizing a MapReduce [5] implementation that is instantiated within a utility cloud infrastructure. The presented system relies on the Planets Interoperability Framework, Apache Hadoop [18], and a JSS prototype providing a grid middleware layer on top of the AWS cloud infrastructure. The paper is organized as follows: Section 2 provides an overview on cloud computing, data grids, and digital preservation. Section 3 presents the Job Submission Service and experimental results. Section 4 concludes the paper.

2 Background and Related Work

2.1 Cloud Computing and Science

The demand for storage and computational power of scientific computations often exceeds the resources that are available locally. Grid infrastructures, services and remote HPC facilities can provide a viable solution for scientists to overcome these limitations. However, many applications require dedicated platforms or need time-consuming adaptations in order to utilize a remote resource. Virtual machine technology provides software that virtualizes a physical host
machine, allowing the deployment of platform-independent system images. The deployment of virtual computer instances is supported by a virtual machine monitor, also called a hypervisor. Cloud systems are consumable via Internet-based services offering IT-technology in the form of applications, hosting platforms, or access to computer infrastructures. Amazon’s EC2 and S3 services, one of the most prominent commercial offerings, allow to rent large computational and storage resources on-demand. EC2 is based on the Xen hypervisor and allows to prepare and deploy virtual system instances that suit individual application needs. S3 provides access to a global, distributed, and replicated storage system. A detailed evaluation of Amazon’s compute, storage, and coordination (SQS) web services and their suitability for scientific computing is given in [14] [17]. Deelman et al. provides a cost-based analysis of utilizing the Amazon cloud infrastructure for scientific computing [6]. A proof-of-concept study that runs a complex nuclear physics application on a set of virtual machine nodes is presented in [15]. The Nimbus workspace cloud provides a service to scientific communities allowing the provisioning of customized compute nodes in the form of Xen virtual machines that are deployed on physical nodes of a cluster [3]. A study that compares differences of grid and cloud systems based on the EGEE project [10] for grid and Amazon’s EC2 and S3 services for cloud is given in [11].

2.2 Data Grids and Digital Preservation

A computational grid is a hardware and software infrastructure that provides high-end computational capabilities [12]. Data grids [20] focus on the controlled sharing and management of large data sets that is distributed over heterogeneous sites and organizations. One important aspect is the storage of data in a reliable, distributed, and replicated way. Digital libraries focus on the creation, discovery, and publication of digital collections. Digital preservation and archiving deals with the management and treatment of large data stores in order to preserve their content over time. Preservation archives are systems that implement long-term preservation managing data integrity and technological evolution. This includes migrating digital objects to new technologies, maintaining their relationships and preservation metadata. Data grids can be used as the underlying technology to implement digital libraries and distributed preservation archives [16]. Computational grid systems provide a complimentary technology and are often combined with data grids. For example, the EGEE project [10], currently the world’s largest production grid, provides large quantities of distributed CPUs and petabytes of storage. A survey of initiatives that focus on the integration of emerging technologies like digital libraries, Grid, and Web services for distributed processing and long-term preservation of scientific knowledge is given in [13]. The JSS presented in this paper, provides a Grid service for digital libraries and preservation archives that allows to utilize third party tools based on customized virtual clusters and data intensive computation mechanisms.
3 A Cloud-enabled Job Submission Service

3.1 Motivation

In the context of grid computing and data grids, digital preservation archives are systems that can preserve the output of computational grid processes [16]. An important issue in the context of preserving existing digital content is the process of deriving metadata from digital assets like file collections in order to extract significant semantic information for their preservation (e.g. format characterization). Decisions in preservation planning [2] rely on information generated by algorithms and tools for identification, characterization, feature extraction, and comparison. Analyzing digital entities typically relies on sequential, third party libraries and tools that are not supported by scientific parallel and grid systems. Therefore, we propose a service that employs clusters of customizable virtual nodes in order to overcome these restrictions. The IF JSS implements a grid service that provides access to large numbers of individually tailored compute nodes that can process bulk data based on data-intensive computing mechanisms and that is integratable with computational and data grid systems.

3.2 Service Infrastructure

Our current implementation provides interfaces that support the BES base case specification and accepts JSDL documents that are compliant with the HPCBP profile see Fig. 1. The Job Submission Service (JSS) is a stand-alone web service deployed in a Java EE Web Container. It is secured using HTTPS and SSL/TLS for the transport-layer and WS-Security for the message-layer. The JSS is capable of executing a job within the Amazon EC2 service and makes use of the Amazon S3 storage infrastructure. The virtual nodes within EC2 use the S3 REST API as Data Transfer Service to access input and output files.

3.3 Service Components

The service consists of the following core components: Account Manager, JSDL parser, Session Handler and Execution Manager. In order submit a request to JSS, username and password have to be provided that match a previously created account for the institution that utilizes the service. The individual accounts, utilization history, and potentially billing information are maintained by the Account Manager component. As HPCBP is used as the web service profile, JSDL documents are used to describe the individual job requests which need to be mapped to physical resources by the resource manager. The JSDL parser component validates the XML document and creates an object structure that serves as input for the Execution Manager. A Session Handler maps service requests based on activity identifiers to physical jobs and keeps track of their current status (e.g. pending, running, finished, failed).
3.4 Experimental Results

3.5 Preliminary Considerations

The experiments were carried out were aimed to give a quantitative evaluation of utilizing a virtual, cloud-based infrastructure for executing digital preservation tools. Four dimensions were analyzed and compared to sequential executions on local execution environments: the execution time, the number of tasks, the number of computing nodes, the physical size of the digital objects to migrate. As performance metrics we calculate Speedup and Efficiency [8] as formally described in equations $S_{s,n}$ (1) and $E_p$ (2).

\[
S_{s,n} = \frac{T_{seq,s,n}}{T_{p,s,n}} \quad (1)
\]

\[
E_p = \frac{S_{s,n}}{p} \quad (2)
\]

where:
- $s$ - is the physical size of the digital objects to migrate,
- $n$ - is the number of tasks,
- $p$ - is the number of computing nodes,
- $T_{seq}$ - is the sequential execution time,
- $T_{p}$ - is the execution time with $p$ computing nodes.
3.6 Experiment Setup

We used the Amazon Elastic Compute Cloud (EC2) as cloud infrastructure with up to 150 cluster nodes running a custom image based on RedHat Fedora 8 i386 with the relevant migration tool preinstalled. For all experiments, a simple workflow was implemented that migrates one file collection into a new collection of a different format using a set of command-line tools (e.g. ps2pdf). The default instances used provide one virtual core with one EC2 Compute Unit which provides the equivalent capacity of a 1.0-1.2 GHz 2007 Opteron or 2007 Xeon processor. The digital objects to migrate were stored on Amazon’s Simple Storage System (S3) due to performance considerations. We experienced an average download speed from S3 to EC2 of 32.5 MB/s and an average upload speed from EC2 to S3 of 13.8 MB/s. To parallelize the given tasks across the virtual cluster we used the Apache MapReduce implementation Hadoop in version 0.18.0.

3.7 Measurements and Results

For the experiments shown in Fig. 2 we executed all computations on a constant number of five virtual nodes. The number of migration tasks was increased using different sized digital collections to compare the execution time within the Amazon cloud (EC2) to the sequential local execution time (SLE) on a single node with equal capacity to one EC2 node. Fig. 2 focuses on the intersection points of the corresponding EC2 and SLE curves. These points identify the thresholds above which the parallel execution within EC2 is faster than the sequential execution on a single node. The results outside the bounding box of Fig. 2 including Speedup and Efficiency are shown in Tab. 1. For the experiments shown in Fig. 3 we held the number of tasks constant and increased the number of computing nodes to evaluate the effects on scalability. The values for Speedup, Efficiency and execution time were monitored based on the sequential local execution time of the given job. Speedup increases with the number of nodes as Efficiency decreases due to parallelization overheads. The results are shown in Tab. 2.

<table>
<thead>
<tr>
<th>Number of tasks (n)</th>
<th>Object Size (s) [MB]</th>
<th>SLE exec. time [min]</th>
<th>EC2 exec. time [min]</th>
<th>Speedup</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>0.07</td>
<td>26.38</td>
<td>8.03</td>
<td>3.28</td>
<td>0.67</td>
</tr>
<tr>
<td>100</td>
<td>7.5</td>
<td>152.17</td>
<td>42.27</td>
<td>3.60</td>
<td>0.72</td>
</tr>
<tr>
<td>1000</td>
<td>7.5</td>
<td>1521.67</td>
<td>342.70</td>
<td>4.44</td>
<td>0.88</td>
</tr>
<tr>
<td>100</td>
<td>250</td>
<td>523.83</td>
<td>156.27</td>
<td>3.36</td>
<td>0.67</td>
</tr>
<tr>
<td>1000</td>
<td>250</td>
<td>5326.63</td>
<td>1572.73</td>
<td>3.37</td>
<td>0.68</td>
</tr>
</tbody>
</table>

Tab. 1: Results outside the bounding box of Fig. 2 including Speedup and Efficiency
Fig. 2: Execution time for an increasing number of migration tasks and 5 constant computing nodes with the focus on the intersection points of the EC2 and local execution time.

![Graph](image1.png)

Fig. 3: Execution time for 1000 constant migration tasks using an increasing number of computing nodes to evaluate Speedup and Efficiency for the given job.

![Graph](image2.png)

<table>
<thead>
<tr>
<th>Number of nodes (p)</th>
<th>EC2 execution time [min]</th>
<th>( S_{n,t} )</th>
<th>( E_p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>36.53</td>
<td>0.72</td>
<td>0.72</td>
</tr>
<tr>
<td>5</td>
<td>8.03</td>
<td>3.28</td>
<td>0.66</td>
</tr>
<tr>
<td>10</td>
<td>4.82</td>
<td>5.48</td>
<td>0.55</td>
</tr>
<tr>
<td>50</td>
<td>1.68</td>
<td>15.67</td>
<td>0.31</td>
</tr>
<tr>
<td>100</td>
<td>1.03</td>
<td>25.53</td>
<td>0.26</td>
</tr>
<tr>
<td>150</td>
<td>0.87</td>
<td>30.44</td>
<td>0.20</td>
</tr>
</tbody>
</table>

Tab. 2: Results of experiments shown in Fig. 3 based on the sequential local execution of the given job (n=1000, s=0.07 MB) which took 26.38 minutes.

4 Conclusions and Outlook

Utility clouds provide a novel paradigm for the access and on-demand utilization of remote computational resources and distributed storage infrastructures. In this paper, we presented a grid service for digital preservation system that allows to utilize third party tools based on virtual clusters and data intensive applications. We examined scalability and described experimental results that have been conducted on the Amazon’s EC2 and S3 infrastructure. Future work will deal with developments on a research cloud infrastructure and distributed resource management.

Acknowledgements. Work presented in this paper is partially supported by European Community under the Information Society Technologies (IST) 6th Framework Programme for RTD - Project IST-033789.
References

1. A. Anjomshoaa, F. Brisard, M. Drescher, D. Fellows, A. Ly, S. McGough, D. Pul-
4. Digital Preservation Coalition. Digital curation: digital archives, libraries, and e-sci-
org/graphics/events/digitalarchives.html.
6. Ewa Deelman, Gurmeet Singh, Miron Livny, Bruce Berriman, and John Good. The cost of doing science on the cloud: the montage example. In SC ’08: Proceed-
ings of the 2008 ACM/IEEE conference on Supercomputing, pages 1–12, Pisc-
12. I. Foster and C. Kesselman. The Grid: Blueprint for a New Computing Infra-
20. Srikrum Venugopal, Rajkumar Buyya, and Kotagiri Ramamohanarao. A tax-