Grid Computing Workloads: Bags of Tasks, Workflows, Pilots, and Others

Alexandru Iosup and Dick Epema

Parallel and Distributed Systems Group, Delft University of Technology, Mekelweg 4, 2628CD Delft, the Netherlands
Contact info: A.Iosup@tudelft.nl, D.H.J.Epema@tudelft.nl

***Abstract*—In the mid 1990s, the grid computing community promised the "compute power grid," a utility computing infrastructure for scientists and engineers. Since then, a variety of grids have been built world-wide—for academic purposes, for specific application domains, for general production work. Understanding the workloads of grids is important for the design and tuning of future grid resource managers and applications, especially in the recent wake of commercial grids and clouds. This article presents an overview of the most important characteristics of grid workloads in the past seven years (2003-2010). Starting from the data collected by the authors in the Grid Workloads Archive, this study focuses on four main axes of characterization: system usage, user population, general application characteristics, and characteristics of grid-specific application types. The utilizations of grids vary widely, but are stable in the long term. Although grid user populations range from tens to hundreds of individuals, a few users dominate each grid's workload both in terms of consumed resources and of number of jobs submitted to the system. Real grid workloads include very few parallel jobs but many independent single-machine jobs (tasks) grouped into single "bags of tasks."**

I. INTRODUCTION

The vision of the grid as a computing and data platform that is ubiquitous, uninterrupted, and has uniform user access–in this sense, similar to the power grid–was formulated in the mid 1990s [3]. Grids such as the EGEE (global, but mostly EU-based), the Open Science Grid (OSG, world-wide, but mostly USA-based), Teragrid (USA), Naregi (Japan), Grid'5000 (France), and the DAS (the Netherlands), have grown to serving tens to hundreds of scientists. These grids are used for many application areas, such as physics, bioinformatics, earth sciences, life sciences, finance, space engineering, etc. Grids have strengthened the change in science and engineering of complementing theory and experimentation with computational and data-intensive discovery [2], [6]. This article discusses grid workloads and their evolution between 2003 and 2010.

Grids are collections of resources ranging from clusters to supercomputers. Many types of jobs have been tried on grids, from sequential to parallel, from compute-intensive to data-intensive, and from massive coordinated applications to bags of independent tasks. A typical grid-based experiment requires the repeated execution of a computational task on different sets of input parameters or data; thus, many grid workloads are dominated by applications with a bag of tasks structure. The grid resource providers and the grid resource consumers (the users) are often different entities. The grid resource providers decide on the resource management policies, and provide only minimal, generic job management services. To simplify management, Virtual Organizations (VOs) group administratively users or resource providers.

Understanding the characteristics of entire grid workloads is important in evolving and tuning existing grids, and in the design and development of new grid resource management solutions. This article reviews grid workloads along four main axes: system usage (we look at utilization and task arrivals), user population (number of users and VOs), general application characteristics (CPU, memory, disk, and network), and characteristics of grid-specific application types (presence, structure, etc.).

Our analysis is based on grid workload traces collected from over fifteen real grids. The traces have been kindly provided by grid owners or users; some of these traces are publicly available via the Grid Workloads Archive (GWA) [13]. Table I summarizes two properties of the studied traces, duration and system size. The values illustrate the breadth of our study: time-wise, nine of the traces

TABLE I

SUMMARY OF THE PROPERTIES OF THE STUDIED TRACES. THE ⋆ SIGN MARKS THAT THE TRACE ONLY REPRESENTS A PART OF THE SYSTEM. GRP AND USR ARE ACRONYMS FOR NUMBER OF GROUPS (VOS) AND OF USERS, RESPECTIVELY.

Trace ID	Source System		Number of ... in System	
(GWA Index)	*Name (Country, Type)*	Period	Sites	CPUs
GWA-T-1	DAS-2 (NL, academic)	02/05-03/06	5	400
GWA-T-2	Grid'5000 (FR, academic)	05/04-11/06	15	2,500+
GWA-T-3	NorduGrid (EU, academic/production)	05/04-02/06	75+	2,000+
GWA-T-4	AuverGrid (FR, production)	01/06-01/07	5	475
GWA-T-5	NGS (UK, production)	02/03-02/07	1	400+
GWA-T-6	LCG, RAL cluster (UK, production)	05/05-01/06	$1^{\star}$	880
GWA-T-7	GLOW (US, production)	09/06-01/07	$1^{\star}$	1,400+
GWA-T-8	Grid3 (US, academic/production)	06/04-01/06	29	2,200+
GWA-T-9	TeraGrid-1, ANL cluster (US, production)	08/05-03/06	$1^{\star}$	96
GWA-T-10	SHARCNET (CA, production)	12/05-12/06	10	6,828
GWA-T-11	EGEE/LCG (EU, production)	11/05-12/05	220+	24,000+
GWA-T-12	Condor U.Wisc.-Madison (US, production)	10/06-11/06	5	2,100+
GWA-T-13	TeraGrid-2, NCSA cluster (US, production)	05/06-01/07	$1^{\star}$	1,000
GWA-T-14	DAS-3 (NL, academic)	07/06–10/08	5	544
GWA-T-15	Austrian Grid (AT, academic/production)	09/06-10/07	8	250
GWA-T-16	Austrian Grid 2 (AT, academic/production)	05/07-11/07	8	250
GWA-T-17	TeraGrid-2, NCSA cluster (US, production)	01/10-05/10	$1^{\star}$	930

are long-term (one year of operation or more) and thirteen are medium-term (six or more months); size-wise, the traces we study have been collected from several large (2,000 CPUs or more) grids, including EGEE, Grid'5000, Grid3 (the precursor of OSG), and NorduGrid. The traces also include examples of system replacement (DAS-2 was phased-out and replaced with DAS-3, traces GWA-T-15 and GWA-T-16 represent the replacement of the job manager), system evolution (traces GWA-T-13 and GWA-T-17 have been taken in the same system with a 3.5 years interval), and detailed/coarse views of the same system (for example, for EGEE the traces GWA-T-6/GWA-T-11, respectively).

II. GENERAL WORKLOAD CHARACTERISTICS

Grid workloads exhibit a number of features that we summarize below; more information can be found in our previous studies [8], [13].

System utilization is either very high or very low. The long-term average grid utilization ranges from very low (10-15% in the research grids DAS and Grid'5000) to very high (over 85% in parts of the LCG, in Condor U.Wisc.-Madison, and in AuverGrid, which are all production grids). The short-term utilization can be very high, and every grid investigated in this work has experienced week-long overloads (full-capacity utilization and excess demand) in their existence. Load imbalance between grid sites and submission spikes happen often [10].

Workload Size: hundreds of users, many tasks. Table II summarizes the size characteristics of the grid workloads. A single grid cluster can provide over 750 CPU Years per year (the RAL cluster in LCG), whereas a single user VO can consume over 350 CPU Years per year in combined use (the ATLAS VO in Grid3). The number of jobs completed per day in grid systems is on average over 4,000 jobs/day in LCG's RAL cluster, and 500 to 1,000 for Grid3 and DAS2. While the number of hourly job arrivals is in general small, the number of jobs running in a grid can spike to over 20,000 per day for a single cluster (for example, in DAS-2 and the LCG RAL cluster traces), and to over 20,000 per hour for a whole grid (SHARCNET).

Population: a few users contribute most to the workload. In general, even the largest grids are used by a few tens of organizations and by several hundreds of users. Less than ten users, often less than five, dominate the workload of the grid, both in terms of number of jobs submitted to the grid and of consumed resources, as exemplified in Figure 1.

Submission Patterns Grid workloads exhibit strong time patterns, including seasonal, work day, and hourly. Most grids are less used during holidays, week-ends, and middle-of-day hours. Many academic grids are overloaded during the periods preceding major conferences. The submission behavior of individual user varies greatly among users, but the top users have often replaced irregular (manual) submission with tools that submit jobs periodically.

Grids vs. Parallel Production Workloads In comparison with the clusters and low-end supercomputers of the end-1990s and beginning-2000s,

TABLE II

SUMMARY OF THE CONTENT OF THE STUDIED TRACES. THE ⋆ SIGN MARKS THAT THE TRACE ONLY REPRESENTS A PART OF THE SYSTEM. GRP AND USR ARE ACRONYMS FOR NUMBER OF GROUPS AND OF USERS, RESPECTIVELY. THE COLUMN "ARRIVALS" LISTS FOR EACH SYSTEM THE AVERAGE NUMBER OF ARRIVALS PER HOUR. THE COLUMN "SPIKE" LISTS FOR EACH SYSTEM THE MAXIMUM NUMBER OF JOBS RUNNING DURING A DAY.

Trace ID	Source System	Number of Observed					
(GWA Index)	*Name (Country)*	Jobs	GRP	USR	CPU Time	Arrivals	Spike
GWA-T-1	DAS-2 (NL)	602K	12	332	69y	71	19,550
GWA-T-2	Grid'5000 (FR)	951K	10	473	128y	-	
GWA-T-3	NorduGrid (EU)	781K	106	387	2,444y	28	7,953
GWA-T-4	AuverGrid (FR)	404K	9	405	278y	46	823/h
GWA-T-5	NGS (UK)	632K	1	379	270y	23	4,994/h
GWA-T-6	LCG, RAL cl. (UK)	1.1M	25	206	751y	-	22,550
GWA-T-7	GLOW (US)	216K	1⋆	18	120y	-	6,590
GWA-T-8	Grid3 (US)	1.3M	1⋆	19	240y	-	15,853
GWA-T-9	TeraGrid-1, ANL cl. (US)	1.1M	26	121	-	-	7,561
GWA-T-10	SHARCNET (CA)	1.2M	-	412	3,782y	127	22,334/h
GWA-T-11	EGEE/LCG (EU)	188K	28	216	54y	504	1,638/h
GWA-T-12	Condor U.Wisc.-Madison (US)	765K	-	-	22,370y	-	-
GWA-T-13	TeraGrid-2, NCSA cl. (US)	140K	-	-	222y	-	-
GWA-T-14	DAS-3 (NL)	2.0M	12	333	-	-	-
GWA-T-15	Austrian Grid (AT)	141k	-	-	152d	-	-
GWA-T-16	Austrian Grid 2 (AT)	46k	-	-	41d	-	-
GWA-T-17	TeraGrid-2, NCSA cl. (US)	28K	-	83	-	-	-

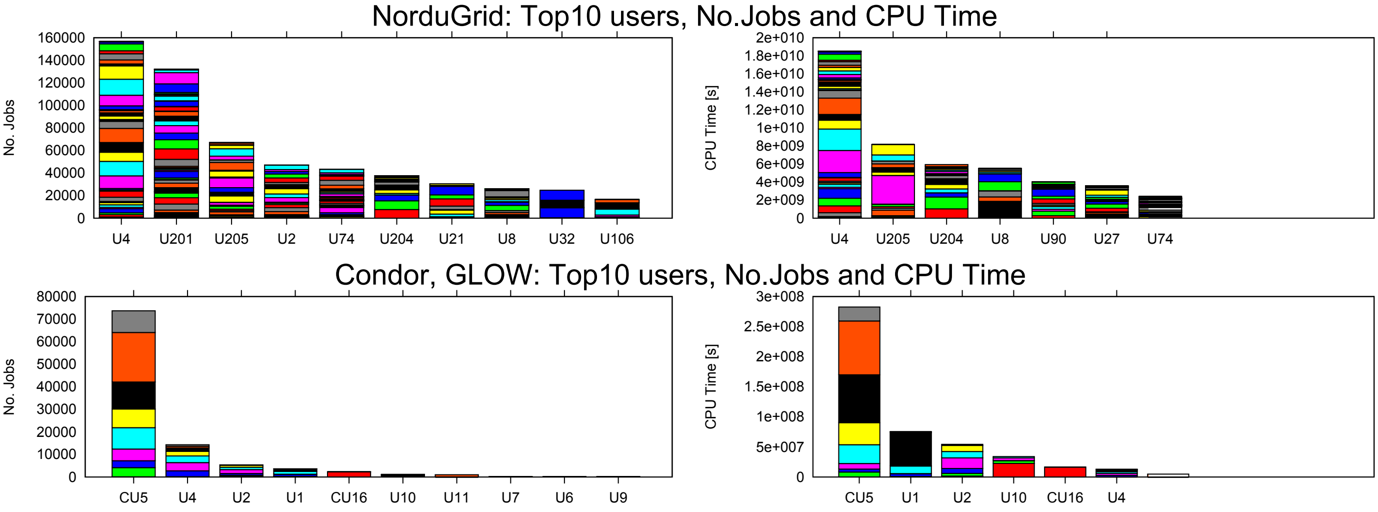

Fig. 1. The number of submitted jobs (left) and the consumed CPU time (right) by user per system: NorduGrid (top) and Condor GLOW (bottom). Only the top 10 users are displayed for each system. The horizontal axis depicts the user's rank. The vertical axis shows the cumulated values. Weekly consumption is shown as blocks of different shade (color); larger blocks denote weekly demand surges.

grids exhibit similar resource consumption, more completed jobs per day, higher spikes in the number of concurrently running jobs, and can reach much higher utilizations. Specifically, parallel production environments (*PPEs*) offer 50 to 1,300 CPU Years per year, have on average less than 500 jobs completed per day, spikes of 300 to 5,400 jobs, and utilization often in the mid-60%s (these results hold for each individual parallel production environment trace in the Parallel Workloads Archive (http://cs.huji.ac.il/labs/parallel/workload/).

III. GENERAL JOB CHARACTERISTICS

This section characterizes the jobs present in grid workloads, regardless of their application domain or structure; more in-depth studies on this topic are [13], [23], [7]. Table III summarizes the average and standard deviation of the number of processors allocated to jobs, the job runtime, and the memory consumption of jobs. Figure 2 depicts for selected

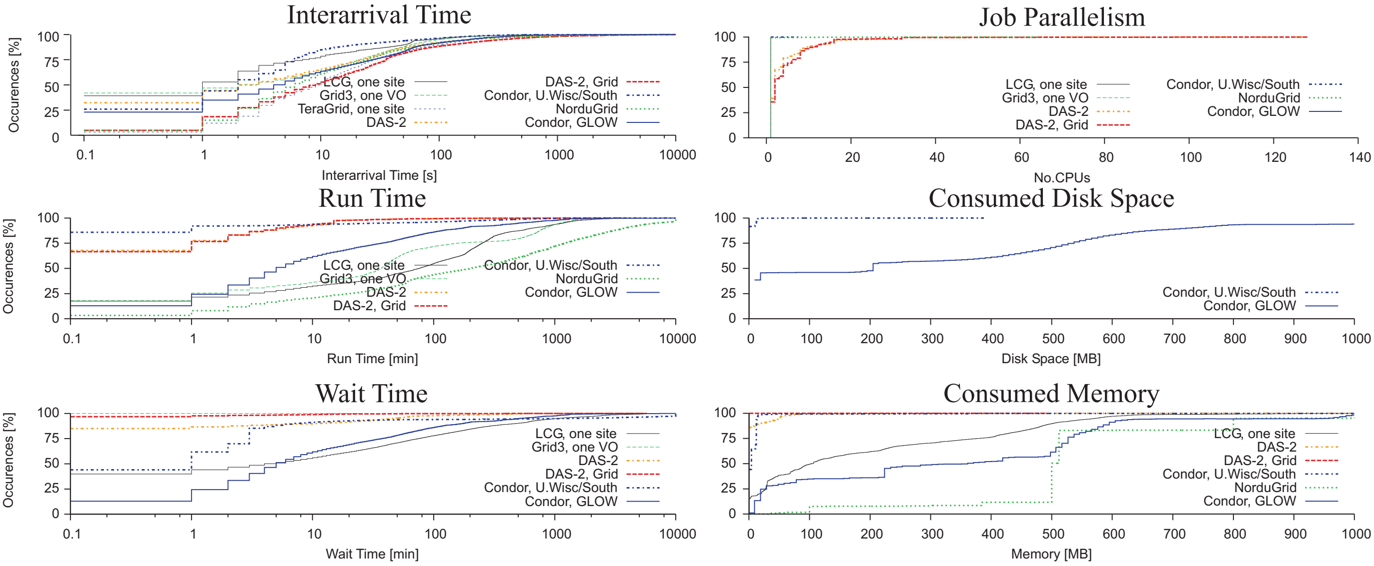

Fig. 2. CDFs of the most important job characteristics for NorduGrid, Condor GLOW, Condor UWisc-South, TeraGrid, Grid3, LCG, DAS-2, and DAS-2 Grid. Time-related characteristics in logscale.

TABLE III
SUMMARY OF JOB CHARACTERISTICS FOR THE STUDIED TRACES.
IN PARENTHESES THE STANDARD DEVIATION.

	Overall job characteristics				
System	Size [CPUs]	Runtime [s]		Memory [MB]	
GWA-T-1	4.3 (6.3)	370	(3,938)	46	(346)
GWA-T-3	1.1 (1.0)	89,274	(284,300)	200	(307)
GWA-T-4	1.0 (0.0)	25,186	(40,780)	296	(343)
GWA-T-5	1.4 (2.8)	2,925	(17,908)	39	(226)
GWA-T-6	1.0 (0.0)	14,599	(28,641)	195	(206)
GWA-T-7	1.0 (0.0)	4,705	(14,488)	332	(276)
GWA-T-8	1.0 (0.0)	13,797	(25,201)	*n/a*	
GWA-T-10	1.5 (6.2)	31,964	(117,088)	81	(466)
GWA-T-11	1.0 (0.0)	8,971	(32,833)	*n/a*	

grid workloads the cumulative distribution functions (CDFs) associated with various job characteristics. Both the table and the figure indicate the **high variability of grid job characteristics**.

Not all the grid workload traces we use in this study contain information about all the characteristics. In particular, only few contain memory, I/O, and network-related related information; for I/O and network we use the Condor-based system traced in GWA-T-12, for which we analyze independently five data subsets coming each from a traced resource pool. Subsets t1 and t2 comprise mostly engineering and computer science jobs, respectively; subsets t3, t4, and t5 comprise exclusively high-energy physics (HEP) jobs of different characteristics.

Mostly conveniently parallel jobs. Grid workloads exhibit little intra-job parallelism, in contrast to PPEs, they are dominated by loosely coupled jobs (see Section IV). In many grid workload traces there exist no parallel jobs, that is, jobs that require more than a single node to operate. Most of the grids workload traces in which parallel jobs are present are academic grids; the exceptions are SHARCNET and TeraGrid, which run scientific applications as parallel jobs. Even for the few grids that do run parallel jobs, the job parallelism is low: mostly under 32 processors per job for grids (maximum 800 for SHARCNET and 128 for the others). These small parallel job sizes match well the parallel workloads of early-2000s PPEs. Although few grid workloads comprise parallel jobs, where they occur, they consume a majority of the grid's provided CPU time.

Job runtime: several hours. Grid jobs require in general multiple hours to complete, with per-grid averages ranging from about one hour to about a day. The jobs typical to High-Energy Physics (HEP) have been specifically designed to be processed in around twelve hours on low-end machines, with low variability in processing time; thus, many run for six-seven hours on the high-end grid nodes [7]. The DAS-2 and DAS-3 grids have been designed to promote the use of small, interactive jobs, which explains their outlier average job runtime of 370s. Although the averages are relatively long, most

TABLE IV
AVERAGE I/O PER JOB IN CONDOR-BASED GRIDS.

T-12	I/O [KOps]				I/O Traffic [MB]		
part	Total	Rd	Wr	Wr %	Total	Rd	Wr %
t1	28	18	6	20%	469	174	63%
t2	957	770	187	20%	144	114	21%
t3	904	881	23	3%	161	130	19%
t4	13,058	9	13,049	100%	389	33	92%
t5	11,128	8	11,121	100%	330	31	91%

TABLE V
AVERAGE NETWORK USAGE PER JOB IN CONDOR-BASED GRIDS.

T-12	File Transfer [MB]				Remote Sys. Calls [MB]			
part	Total	In	In / Out %		Total	In	In / Out %	
t1	10,865	8,259	76%	24%	28	16	59%	41%
t2	1,736	1,542	89%	11%	71	28	40%	60%
t3	1,938	1,738	90%	10%	77	32	42%	58%
t4	1,043	653	63%	37%	6	6	100%	0%
t5	671	432	64%	36%	44	40	91%	9%

grids workloads contain large numbers of much shorter or much longer jobs. Notably, Figure 2 shows that in many grids a quarter of the jobs have a runtime of 2 minutes or less.

Memory requirements: modest, with the exception of HEP jobs. Grid jobs require on average tens to hundreds of MBs of memory. Most HEP jobs require machines with at least 2GB memory per processor, although in practice they may use less. On average, production grid jobs require more memory than academic grid jobs. The CDF of the memory consumption in Figure 2 shows the existence of preferred memory consumption sizes; the NorduGrid trace has a distribution mode around 500MB.

I/O requirements: modest, with the exception of HEP jobs. Many grid jobs are compute-intensive and have in general modest I/O requirements. Table IV summarizes the I/O consumption for five subsets of the GWA-T-12 trace, one for each resource pool in the system. The total number of operations and the total I/O traffic averaged by grid jobs are higher than for typical scientific applications [23]. The variability of observed values remains high. The sizes and rates of various I/O operations exhibit pronounced modes, which means that system designers can optimize for the common cases. The high fraction of Writes, from all I/O operations, may make caching difficult. HEP jobs put more stress on the I/O system than other grid jobs; their characteristics are [7]: about 2.2PB of data processed per year by a single experiment, at about 65MBps; mean file size 300MB, with about 5% of the files are larger than 1GB; and each job accesses on average over 100 files. In Figure 2, sub-figure "Consumed Disk Space" exemplifies the difference between HEP and engineering jobs in a Condor-based environment–the Condor-based GLOW environment's HEP jobs have much larger input.

Network requirements: generally modest. Although there are few tightly coupled parallel jobs in grids, network traffic may be required to transfer the input and output files to/from the processing nodes, and to manage the remote execution of jobs. Table V summarizes the job network consumption for the same five subsets of the GWA-T-12 trace we used for the I/O analysis (see also Table IV). The input varies widely among these subsets. The input represents over 60% of the file traffic, in all traces. The traffic used for remote system calls is much lower than for files; the fraction of output traffic ranges here from 0 to 60% of the total traffic.

IV. BAGS-OF-TASKS

Bags of Tasks (BoTs) are loosely coupled parallel jobs in which a set of tasks are executed to produce a meaningful, combined result. In many grid workload traces, information about the job-to-BoT mapping is missing. Identifying BoTs in such traces may be made more difficult by BoT managers; for example, many BoT managers delay the submission of tasks to ensure that a limited number of tasks are concurrently running in the grid, so tasks belonging to the same BoT become grid jobs with different submission times. When job-to-BoT mapping information is missing from the trace, we identify BoTs with a method [11], [14] that groups jobs submitted by the same user, according to their relative arrival time.

BoT submissions dominate the grid workloads, by number of tasks and consumed resources. Table VI summarizes the presence of BoTs in a number of selected grids. In most grid traces, BoT submissions account for over 75% of the tasks and of the consumed CPU time; BoTs are often responsible for over 90% of the total CPU-time consumption. The average number of tasks per BoT ranges for the different grid traces investigated here from 2 to 70, with most averages between 5 and 20.

TABLE VI
SUMMARY OF BOT PRESENCE IN GRID TRACES.

Trace ID	Observed BoTs	Percentage From Total	
		Jobs	CPUTime
GWA-T-1	57k	92%	78%
GWA-T-2	26k	85%	30%
GWA-T-3	50k	94%	90%
GWA-T-6	43k	95%	95%
GWA-T-7	13k	95%	96%
GWA-T-8	302k	94%	98%
GWA-T-10	16k	93%	92%
GWA-T-11	5k	96%	97%
GWA-T-12	135K	94%	96%
GWA-T-13	68K	96%	86%

A **model for grid BoTs** [14] that captures well the highly variable data observed in many grid traces can focus on **four aspects: the submitting user, the BoT arrival patterns, the BoT size, and the intra-BoT (individual task) characteristics**. The probability of a grid job to be submitted by a specific user is well modeled by a Zipf distribution. The BoT inter-arrival time is best modeled by a Weibull distribution. The size of the BoTs is best modeled by the Weibull distribution for most systems. The average BoT task runtime is best modeled by the Normal distribution for a majority of systems. Last, the variability of the runtimes of BoT tasks is best fit by a Weibull distribution for most systems.

V. WORKFLOWS, PILOTS, AND OTHERS

While grids are already supporting (small) bags of tasks, the performance of the generic job and resource management services provided by grids can be improved through user- and application-specific tools and policies. Motivated by high rates of system [12], [15] and middleware [9] failures, high job management overhead[1], and slow detection of job failures [5] the grid community has built tools and mechanisms for improved execution and coordination of specific types of jobs in grids. We review in the following three such mechanisms.

Grid Workflows: very large and long-running, or small and short-running. Grid workflows are jobs with a graph structure where the nodes are grid computing and grid data transfer tasks, and the edges are dependencies between the tasks; more details can be found in a recent overview of the current status of grid workflow engines [2]. A common engineering workflow would consist of pre-processing, simulation, and post-processing steps, each consisting of several tasks.

It is difficult to identify the presence of workflows in most grid workload traces used in this study—at this level of tracing there is little data concerning workflows. In a recent study [1], five scientific workflows covering astronomy, earth sciences, and bioinformatics are shown to have sizes of tens to tens of thousands of tasks; the same authors have reported cases of even larger instances. The sums of task runtimes in these workflows is from hours to weeks, which makes workflows equivalent to long-running grid jobs. Engineering workflows can be very different from scientific workflows, as we have shown recently [16] based on the GWA-T-15 and GWA-T-16 traces. For these workflows, the average number of tasks per workflow is in the low tens, with 75% of the workflows having fewer than 40 tasks, and 95% of the workflows having fewer than 200 tasks. The average graph level (shortest path from start to completion) is between 2 and 4, and in GWA-T-16 over 80% of the workflows have at most two levels. Tasks in these engineering workflows can be very short, with over 75% of the tasks taking less than 2 minutes to complete. An alternative explanation for the small sizes of engineering workflows is that the most common grid workflow schedulers, i.e., Condor's and Globus's, incur high overheads when managing large or complex workflows [22].

Pilot Jobs: BoTs with *many* tasks. For performance and reliability reasons, pilot jobs install the user's own job management system on the resources provisioned from the grid, then execute through this system a stream (bag) of tasks coming from the user. Common pilot job tools are Condor (through its glide-in features), DIANE [18], glideCAF or glideinWMS [19], Falkon [17], and GridBot [20]. For pilot jobs, a common performance metric is throughput, defined as the number of tasks completed per second (tps); the Falkon system has achieved [17] a throughput of about 500tps vs Condor's 0.5tps and PBS's 0.4tps, in the same grid environment.

There currently exists no study of a pilot job workload. With pilot jobs, grid systems may record jobs that are running for days or even weeks; in reality, such jobs run streams of short tasks that may

---

[1] In EGEE around 2007, half of the submitted jobs waited more than five minutes to be deployed, due to high execution overhead [4].

take each a few minutes up to about an hour. One pilot job system, GridBot, has been used [20] to execute through pilot jobs the workload of a *real* bioinformatics community: hundreds up to millions of tasks per pilot job (stream), with about 4,000 tasks on average per pilot job; 0.5 CPU years per pilot job; the average task runtime is 15 minutes for a medium-sized pilot job, and 30 seconds to 5 minutes for small-sized pilot jobs; a large pilot job may execute over 2M tasks of 20–40 minutes each, requiring over 100 CPU years.

Others: co-allocated and malleable jobs One of the first new grid mechanisms for user-specific resource management to be designed were co-allocation [21], that is, the simultaneous allocation of resources from different grid clusters or even sites for a single grid job, and malleable allocation, that is, the dynamic allocation and de-allocation of resources for a single grid jobs. No study exists of the actual use of either mechanism in real grid workloads, but in Grid'5000 there exist only about 6,000 co-allocated (parallel) jobs, or under 2% of the jobs recorded in the trace.

VI. CONCLUSION

Understanding grid workloads is important for tuning existing grids and for designing the grids of the future. In addition, when grid workloads are moved to clouds, which may very well happen as high-performance computing centers are currently installing private clouds for their user communities, their understanding may also drive the design and tuning of clouds. This article has reviewed the characteristics and evolution of grid workloads between 2003 and 2010, along four axes: system usage, user population, general application characteristics, and grid-specific application types. From the many grid application types, the article has focused on bags of tasks, which have been the most common grid application during the observed period, and workflows and pilot jobs.

Grid workloads are very different from the workloads of other environments, and in particular from the workloads of parallel production environments such as supercomputers and large clusters. In particular, grid workloads are dominated by bags of independent, multi-hour tasks, which can lead to very high system utilizations over long periods of time.

Only the future can tell how the evolution of grid workloads will continue. Will inter-dependent many-task jobs become daily scientific tools and dominate the workloads? Will job runtimes decrease towards minute-long jobs, as suggested by today's pilot jobs? Will parallel jobs see a resurgence with the increase in the number of multi-core grid nodes? And, perhaps most importantly, will the worlds of grids and clouds move closer or even merge, with more diverse workloads for the resulting systems?

REFERENCES

[1] S. Bharathi, A. Chervenak, E. Deelman, G. Mehta, M.-H. Su, and K. Vahi. Characterization of scientific workflows. In *Workshop on Workflows in Support of Large-Scale Science (WORKS08)*, pages 1–11, 2008.

[2] E. Deelman, D. Gannon, M. S. Shields, and I. Taylor. Workflows and e-science: An overview of workflow system features and capabilities. *Future Generation Comp. Syst.*, 25(5):528–540, 2009.

[3] I. T. Foster, C. Kesselman, and S. Tuecke. The anatomy of the grid: Enabling scalable virtual organizations. *IJHPCA*, 15(3):200–222, 2001.

[4] T. Glatard, D. Lingrand, J. Montagnat, and M. Riveill. Impact of the execution context on grid job performances. In *CCGRID*, pages 713–718, 2007.

[5] T. Glatard and X. Pennec. Optimizing jobs timeouts on clusters and production grids. In *CCGRID*, pages 100–107, 2007.

[6] T. Hey, S. Tansley, and K. Tolle. *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research Press, 2009. [Online] Available: http://research.microsoft.com/en-us/collaboration/fourthparadigm/contents.aspx.

[7] A. Iamnitchi, S. Doraimani, and G. Garzoglio. Filecules in high-energy physics: Characteristics and impact on resource management. In *HPDC*, pages 69–80, 2006.

[8] A. Iosup, C. Dumitrescu, D. H. J. Epema, H. Li, and L. Wolters. How are real grids used? the analysis of four grid traces and its implications. In *GRID*, pages 262–269, 2006.

[9] A. Iosup, D. H. J. Epema, P. Couvares, A. Karp, and M. Livny. Build-and-test workloads for grid middleware: Problem, analysis, and applications. In *CCGRID*, pages 205–213, 2007.

[10] A. Iosup, D. H. J. Epema, T. Tannenbaum, M. Farrellee, and M. Livny. Inter-operating grids through delegated matchmaking. In *SC*, page 13, 2007.

[11] A. Iosup, M. Jan, O. O. Sonmez, and D. H. J. Epema. The characteristics and performance of groups of jobs in grids. In *Euro-Par*, pages 382–393, 2007.

[12] A. Iosup, M. Jan, O. O. Sonmez, and D. H. J. Epema. On the dynamic resource availability in grids. In *GRID*, pages 26–33, 2007.

[13] A. Iosup, H. Li, M. Jan, S. Anoep, C. Dumitrescu, L. Wolters, and D. H. J. Epema. The grid workloads archive. *Future Generation Comp. Syst.*, 24(7):672–686, 2008.

[14] A. Iosup, O. O. Sonmez, S. Anoep, and D. H. J. Epema. The performance of bags-of-tasks in large-scale distributed systems. In *HPDC*, pages 97–108, 2008.

[15] D. Kondo, B. Javadi, A. Iosup, and D. H. J. Epema. The failure trace archive: Enabling comparative analysis of failures in diverse distributed systems. In *CCGRID*, pages 398–407, 2010. [Online] Traces: http://fta.inria.fr.

[16] S. Ostermann, A. Iosup, R. Prodan, T. Fahringer, and D. Epema. On the characteristics of grid workflows. In *Proc. of the CoreGRID Workshop on Integrated Research in Grid Computing (CGIW'08)*, pages 431–442, Apr 2008. Crete, GR. ISBN: 978-960-524-260-2.

[17] I. Raicu, Y. Zhao, C. Dumitrescu, I. T. Foster, and M. Wilde. Falkon: a fast and light-weight task execution framework. In *SC*, page 43, 2007.

[18] D. Sarrut and L. Guigues. Region-oriented ct image representation for reducing computing time of monte carlo simulations. *Med. Phys.*, 35(4), 2008.

[19] I. Sfiligoi. glideInWMS: a generic pilot-based workload management system. *Journal of Physics: Conference Series*, 119(6), 2008.

[20] M. Silberstein, A. Sharov, D. Geiger, and A. Schuster. Gridbot: execution of bags of tasks in multiple grids. In *SC*, 2009.

[21] O. O. Sonmez, H. H. Mohamed, and D. H. J. Epema. On the benefit of processor coallocation in multicluster grid systems. *IEEE Trans. Parallel Distrib. Syst.*, 21(6):778–789, 2010.

[22] C. Stratan, A. Iosup, and D. H. J. Epema. A performance study of grid workflow engines. In *GRID*, pages 25–32, 2008.

[23] D. Thain, J. Bent, A. C. Arpaci-Dusseau, R. H. Arpaci-Dusseau, and M. Livny. Pipeline and batch sharing in grid workloads. In *HPDC*, pages 152–161, 2003.