Memo 134

Cloud Computing and the Square Kilometre Array

R. Newman (Oxford University)
J Tseng (Oxford University)

May 2011

www.skatelescope.org/pages/page_memos.htm

1 Executive Summary

The Square Kilometre Array (SKA) will be a next-generation radio telescope that has a discovery potential 10,000 times greater than anything available currently[24]. The SKA's scientific potential is due to its large combined antennae area and consequent ability to collect vast amounts of data—predicted to be many Exabytes[1] of data per year once fully operational [71, 46, 41, 44].

Processing this data to form standardised "data products" such as data cubes and images is a major challenge yet to be solved and conservative estimates suggest an Exaflop[2] computer will be needed to process the daily data [46, 45, 44, 71]. Although full production may not be until after 2020, such a computer would still be the top supercomputer in the world (even assuming another decade of Moore's Law-type technology improvements) and the daily data captured would still be impractical to store permanently [46].

Such challenges warrant examining all possible sources of computing power to mitigate project risks and ensure most effective use of project funds when construction begins.

This report was commissioned by ICRAR, iVEC, and Oxford University to examine whether aspects of *Cloud Computing* could be used as part of an overall computing strategy for the SKA. The dual aims of this 3 month project are therefore:

1. to examine the computing requirements for the planned SKA data processing pipeline in the context of the growing popularity of cloud computing; and
2. to develop one or more prototype applications to demonstrate the efficacy of Desktop Cloud computing in the context of the astrophysics problems related to the SKA.

It is expected that the SKA will be built in either South Africa or Western Australia, with the site chosen by 2012. Both sites are in advanced stages of building SKA *pathfinders*[3], but as two of the commissioning organisations (ICRAR and iVEC) represent the Australian candidate this report uses the Australian pathfinder "ASKAP" as the exemplar for examining SKA computing, and prototype cloud computing code was developed and tested during the Australian part of the work.

---

[1] Equivalent to 50,000 years of DVD quality video [26].

[2] A supercomputer 1000 times more powerful than the most powerful supercomputer today.

[3] Smaller systems designed to test technologies - see `http://en.wikipedia.org/wiki/Australian_Square_Kilometre_Array_Pathfinder`

2 SKA Data Processing Pipeline

As in other large scientific instruments such as the LHC[19], LOFAR[20] and LSST[18], the SKA data will be collected and then immediately processed and reduced into forms more convenient for storage, scientific analysis and/or reprocessing [43, 49]. Only dedicated parallel hardware will be able to cope with the initial computations (thousands of FPGA's or ASIC's), which start at the remote antenna sites where the signals will be digitised almost immediately after being captured [41, 43]. Digital data will then transported via fibre optic links in stages to progressively more central processing facilities where distilled data products will be derived and stored in a digital archive [46].

Although these data products will be orders of magnitude smaller than the raw data, and therefore suitable for online access, the archive will nevertheless grow by many petabytes per year [56, 54, 50]. Thus it makes sense to locate the archive relatively close to (and to be considered part of) the central SKA infrastructure [44, 54].

The ASKAP pathfinder project [4] serves as a useful template to examine the data pipeline plans for the SKA. Designed to be 1% the size of the full SKA, ASKAP consists of 36 dish antennas which will produce collectively a data rate of over 60 Terabits/sec (see [28, 43]). Correlation, beamforming and other processing steps will reduce this to less than 1 GByte/sec which will be then stored in a central archive in Perth. A 100 Teraflop computer has been built in the Pawsey HPC centre to manage the final data production steps [46, 48], and such a machine today ranks 87^{th} in the world [25]. The ASKAP data pipeline already pushes the current limits of data transport, processing and storage, thereby highlighting the significant technical challenges of building the full SKA pipeline even allowing for technology advances over the next decade [44].

Detecting astrophysical phenomena within these archived data products requires significant ongoing processing, and many will be examined many times over. Each investigation, which may be instigated by researchers almost anywhere in the world, is expected to require significant computing power [76]. For example, the Astropulse project aims to use distributed (volunteer) computing to reprocess data from the Arecibo telescope and Allen Telescope Array, and the latest processing algorithms easily utilise 100 teraFLOPS (TFLOPS) of computing power [73]. The data rate from ASKAP is many times that of the current Astropulse datasets[4], and the SKA many times more than that.

Experiences from other Astrophysics projects suggest the computational load of reprocessing initial data products from respective archives is substantial and grows significantly over time. The Sloan Digital Sky Survey team suggests over-provisioning computing and storage by a factor of 6 relative to initial "raw processing" estimates to account for this growth [53]. The project plan for the LSST also reflects the expectation that a large but difficult to estimate computational load will emerge once data

---

[4] The initial Astropulse datasets totaled only 43 TB.

is made available [57, 21]. This effect is not surprising, and indeed, once operational, the absence of increasing demand over time might be considered a project failure.

A comparison to the ATLAS experiment [5] at the Large Hadron Collider [19] is instructive. The online trigger and data acquisition system, consisting of both real-time components along with nearly 6000 CPU cores in 2300 PC's, is designed to reduce the data stream by a factor of 200,000 before the resulting 300 MB/sec is archived [40]. The problem of storing, processing, re-processing, and analysing this and other LHC data was recognized before it received its official go-ahead in 1996. In 1999, the concept of building a computer "grid" to share processing and storage resources to address this challenge was outlined, and the European Data Grid project started in 2001 to build a prototype [16]. Since then, follow-on projects EGEE, EGEE II, EGEE III, and EGI were successively funded by the European Union, and the LCG (LHC Computer Grid) was a major component of these projects. Since the LHC started large-volume data production in 2010, LCG has been coping well with the data processing load with over 100,000 CPU's [19, 61, 16] distributed between CERN and dozens of other participating institutions. Particle Physics researchers have access to the LCG and can use the resources to run their analysis jobs using a batch queue submission model - albeit a widely distributed one.

The CERN experience suggests that plans to accommodate substantial processing power which will be demanded by users of the data archive should begin early on and be integral to the construction of other infrastructure. The LCG took over a decade to reach production level quality, consumed a substantial fraction of the overall budget and represented a significant project risk.

The SKA is in a similar place the LHC was in 2000, and thus it would seem prudent to consider how the data stored in the archive will be processed by researchers. The Pawsey supercomputing centre [48] is currently the only facility even partly dedicated to the SKA, and as outlined above, much of this processing will be consumed with continuous data production jobs.

As with the LHC, LOFAR, LSST, and other large collaborative projects, the SKA project plan, with some justification, expects researchers to export data and process it with their own resources. Indeed it is expected a tiered data distribution to the rest of the world similar to the model adopted at CERN will be used [55, 44]. While this approach may enable the SKA project to avoid budgeting for some of the processing requirements at this stage, it would be extrordinary[49] not to provide any computing facilities beyond basic data archiving as the rate of scientific discovery from the SKA data would be impeded significantly[5].

Fortunately, aspects of the increasingly popular paradigm of "Cloud Computing" can help mitigate these issues.

---

[5] Many researchers will find it difficult to obtain sufficient processing power themselves and/or suffer network limitations when downloading data from the central archive to other parts of the world.

3 Cloud Computing

Cloud computing is a recent computing paradigm shift where computing infrastructure purchasing, provisioning, housing, maintenance and even administration is outsourced to a specialist management company. The latter, exploiting economies of scale, takes on the capital expenditure of providing and running computing infrastructure and rents this to users as an operating expense. With many providers allowing flexible contracts with little or no tie-in, businesses can save major capital expenditure, financing and over-provisioning costs[6].

The cost savings made possible by Cloud Computing have created a new IT industry sector worth USD 68 billion in 2010, and it is projected to grow to over USD 140 billion by 2014 [14]. There is intense competition for capturing new customers as more businesses move internal IT over to cloud providers, and as the computer is seen as a basic commodity there is an equally intense pressure on prices.

To whatever extent the SKA project eventually provides facilities for processing archived datasets, datacentre-centric cloud computing offers low-risk options to address the isolated location of the archive relative to the bulk of Astrophysicists worldwide:

- Popular or recent data from the archive can be uploaded to the cloud as part of SKA production. Cloud storage makes it easy to find suitable space closer, in network terms, to the majority of users interested in the data, relieving the archive of repeated requests and enabling more efficient use of international bandwidth (and associated costs).
- Once in the cloud, data can be processed within it. Cloud processing offers many flexible ways to access computational resources, including renting HPC clusters by the hour as required, which consequently track improvements in hardware.

Although the datasets will be large by today's standards, ongoing advances in networking technologies mean that the price and time exporting petabyte files internationally will continue to fall and become routine within the timescale of the SKA. The CANARIE network recently (2010) upgraded to a 100 GBit backbone to connect several centres across Canada [10], and similar speed through an undersea cable has also been demonstrated [6]. With network technologies driving into the Terabit/sec range, fueled by ever increasing consumer and business demand, orders-of-magnitude improvements in local, national and international networks can be expected over the coming decade [77, 52, 22, 69].

The good news therefore is that the SKA project need not provide additional networking infrastructure beyond the data archive, and can simply make suitable budget

---

[6] Sudden extra needs can be accommodated within hours using cloud-based services; so called "cloud-bursting"

allocation for data transfer on standard commercial links. Individual researchers can expect to access the SKA archive remotely, but perhaps more importantly the multi-billion dollar cloud computing facilities worldwide will also be within easy reach.

The epitome of cloud provision is Amazon with its “Elastic Compute Cloud” (EC2) and “Simple Storage Service” (S3) [3]. In 2010, Amazon EC2 released their on-demand HPC instances which provide highly interconnected supercomputing power at an hourly cost[7], and as of Feb 2010 SGI has a similar offering[11]. In April 2010 Amazon opened a datacentre in Singapore and is “very likely” to have a presence in Australia in the near future [2, 27]. The Singapore facility could already handle the data storage of ASKAP and will easily handle the SKA data archive within the decade. Thus datacentre-centric cloud computing already provides the capability to store and process ASKAP datasets, and can be expected to be capable of doing the same for the SKA in the appropriate timeframe.

Amazon’s pricing model is based on actual CPU usage and network transfers into and out of the cloud; no additional charges are levied for hardware upgrades or the need to build further datacentre capacity.

Experiment	Size (TB)
High resolution spectral line	200
Survey spectral line medium resolution	330
Snapshot continuum some spectral information	30
High resolution long baseline	360

Table 1: Potential SKA Experiments and possible input data sizes [29]

Table 1 shows the data sizes of some hypothetical data processing tasks which might arise from the SKA. Uploading this data from the archive to the cloud (physically in Singapore, US or Europe) would take around 2-5 days on a 10GBit link (a few hours at 100GBit).

Following Amazon’s pricing model as at September 2010, uploading, storing and processing this data with 1000 EC2 “High CPU” Linux spot instances would cost USD 70,000 per month. This estimate assumes ongoing production of experiment runs, so the 200 TB is stored continuously and 160 CPU hours of processing (on all 1000 nodes) is performed every month.

For a single researcher, with a more limited budget, a one-time run of 200TB on only 100 machines over 3 days would cost USD 11,000.

For a larger collaboration collectively sharing data and processing resources, there might be a need for 1PB storage and 1000 processing nodes. The cost of continuous processing and storage of this data would be approximately USD 225,000 per month (and a yearly payment of USD 455,000). These figures do not factor in volume

---

[7] Amazon HPC on-demand delivering 41 TFLOPS, at USD 1400 per hour `http://www.hpcinthecloud.com/features/Amazon-Adds-HPC-Capability-to-EC2-98349769.html`

discounts or negotiated services available from Amazon's competitors, which would reduce the price significantly.

4 Desktop Cloud Computing

Another aspect of cloud computing, which is fundamentally about sharing physical computing resources efficiently to minimise wastage, is that of a "Desktop Cloud". Similar to the idea of volunteer computing, a desktop cloud is deployed over a wide range of potentially idle desktop computers in domestic or institutional premises. It differs from volunteer computing in that commercial work is embraced, and thus the contributor might earn a reward for the CPU time provided. A desktop cloud also differs from an "Enterprise Cloud" in that it implies computers from different institutions and/or companies are to be used together rather than remaining hidden behind the corporate firewall [15, 51, 38, 60, 37]. The usefulness of a desktop cloud is driven by two key requirements:

1. A large number of underutilised computers, and
2. A good network connection to many of these.

Astrophysics has been a keen user of distributed volunteer computing from the earliest days of the Internet, where personal desktop computers were still relatively underpowered and network connections slow. The original SETI@Home project began in 1999 and when completed in 2005 had achieved the largest computation in history [23]. This success led to the development of the BOINC platform for distributed computing [7], spawning several scientific computing projects including several ongoing Astrophysics searches, *e.g.*, SETI@Home (version 2), Einstein@Home, and Astropulse [73, 12, 23]. The combined BOINC pool of computers grows by 100,000 new donors every year and currently releases over 5 petaFLOPS of computing power; much more than that of the world's top supercomputer.

The success of the volunteer model is due to the rapid changes in both aspects above:

1. There are over 1.2 billion personal computers worldwide in 2010, and over 300 million are manufactured every year. Most computers are idle most of the time (90% or greater [72, 65]), and with multi-core systems now standard, the amount of processing power going to waste is growing rapidly as most PC usage does not require all cores simultaneously.
2. There are over 1.5 billion people connected to the Internet, over 300 million connected with broadband (high-speed) links, and this number increases by millions per year.

The unused computing power worldwide is therefore substantial, currently more than 100 times the combined computing power of the world's top 500 supercomputers, and is growing at a rapid rate. The fair price for using these resources, assuming they are not volunteered, is the data transport cost plus the incremental electricity charge for running a CPU core at 100% rather than idle. By any set of assumptions this cost is minimal, at least 10 times cheaper than any other means of accessing CPU time, because the capital and idle running costs are borne by the owner.

The desktop cloud concept makes even more sense in an academic setting, where they have also been called "Campus Grids" [74, 1, 72, 75]. A typical University will have relatively limited geographic extent and with many high volume network users can easily justify installing and maintaining high speed local area networks. Universities will also provide relatively modern desktop PCs for staff and will typically have rooms of additional (shared) PCs for student laboratories and teaching purposes.

Thus a typical University with 20,000 staff and students might be assumed to have 10,000 computers deployed around campus in offices and labs. If only 5000 of these are assumed to be less than 2 years old (and therefore typical of the average processing power reported by the BOINC project[7]), their combined CPU power would be 50 TFLOPS, *i.e.*, a supercomputer ranked 160^{th} in the world. Linking these machines via a high-speed LAN makes this power readily available and so a campus grid can often provide researchers with more much power than any local HPC centre [72, 75].

Perth has four local universities already equipped with high speed networks, but perhaps just as important is the fact that all four campuses (including the site of the ASKAP and proposed SKA archive) are connected via the iVEC network with 10GBit links [17]. With a combined population of 100,000 staff and students, an assumed 40,000 relatively modern desktop computers could be combined into a 400 TFlop supercomputer, equivalent to the 16^{th} in the world and four times the power of the Pawsey HPC centre [48]. As the infrastructure is already in place, the cost of releasing this power is modest to insignificant (as reported in [72, 9, 32, 74]). It is therefore clear that a suitably provisioned desktop cloud in Perth could dramatically increase the rate at which SKA data is turned into discoveries.

5 A Prototype Desktop Cloud

Although a desktop cloud represents the cheapest way of obtaining large amounts of processing power, not all tasks map well onto a distributed network of computers.

Searching for pulsars in a tiny section of the sky often requires data taken only from that small area. Thus searching the sky requires independent processing of tiny subsamples of the entire sky dataset, and distributed computing can be deployed very effectively in this case. Conversely, simulating galactic evolution since the Big Bang requires accurate modeling of gravitational interactions between large numbers of virtual masses, and thus data may need constantly to be exchanged between compute nodes as the simulation evolves. Distributed computing would be much less effective in this case, and a dedicated HPC cluster with high-speed network interconnect would perform the task much more quickly. Astrophysicists have both types of challenges

[44, 73, 76], so it makes sense to use both types of ICT infrastructures: desktop clouds preferably because of the low cost and high throughput, and HPC facilities otherwise.

Building and testing algorithms on HPC systems is beyond the scope and budget of this study. However it was possible to deploy and test some processing tasks on a prototype desktop cloud in Perth. The NereusV open source desktop cloud technology, developed at Oxford University, was chosen as the basis of this testbed. NereusV has several advantages over other similar middleware, but the comparison to BOINC is perhaps the most relevant in the Astrophysics context:

	BOINC	NereusV
Security	Runs native code. Donors must trust the code.	Runs all code within the Java sandbox - no code is trusted.
Deployment	Client executable must be downloaded and installed/run by the user.	User clicks a link in a web-page; no download/install procedure.
Portability	Different code needed for different hardware.	Runs Java code on the portable JVM - write once run anywhere.
Privacy	Data stored on local disk, using space and potentially viewable.	No data stored on disk, only restricted network IO is allowed.

Perhaps the most important of the above aspects is security; the NereusV clients use the Java “Sandbox” to contain data processing software within a limited execution environment. Sandboxes are increasingly recognised as being essential for large desktop grids/clouds to protect donor machines from unwanted effects of running untrusted, third-party processing code [36, 80, 70, 39, 35, 47, 31, 34, 30]. The Java sandbox is the original Internet sandbox, developed in 1995, and has been protecting client machines from unvetted code on the Web ever since.

The Java sandbox also restricts the NereusV client, and any processing code therein, to outbound TCP connections back to the original server only. Consequently if a computer can start the Nereus client in the first place, then all subsequent networking will pass through any firewalls along the same route and no additional network configuration is needed.

Launched in 2009 at JavaOne, NereusV is gaining popularity in both academia and industry as a easy, open, flexible, secure tool for massively parallel computations

[62]. A Nereus deployment consists of a large number of Nereus clients which connect to a few Nereus servers. Via their Nereus server clients then connect to one or more application servers which are responsible for supplying data and associated processing code. Nereus servers can optionally delegate control of their clients to a 3^{rd} party management system, and in this deployment the NereusCloud system was used (see `http://www.nereuscloud.com`). Clients can be instructed to download data processing code using account control on NereusCloud, so multiple runs of data processing can be started, stopped and rerun at will.

Two Nereus servers were installed, one each at WASP (UWA) and ICRAR (Curtin). Clients were connected to a server based on lowest number of network hops, and in total around 200 clients were loaded from multiple locations over UWA, Curtin and iVEC. The server hardware requirement is minimal, any modern dual core machine with 2GB ram will suffice, and in this instance a Mac Mini and Linux Desktop PC were used[8]. Application servers were started and stopped as required using servers at ICRAR (UWA and Curtin) and, on occasion, an individual user's own desktop PC.

6 Applications on the Nereus Desktop Cloud

6.1 Galaxy Simulation Viewing (Dr Allan Duffy)

Although simulating galaxy evolution in the early universe is in a class of problems which work well only on traditional HPC clusters, computationally demanding post processing can often be done in parallel. In this application the simulation results are rendered to produce frames for a movie, with each frame independent from the others, so the problem maps well onto the distributed model.

Several "runs" of 950 frames were performed with each frame "job" taking 68MB input data, between 5 minutes and 3 hours CPU time and producing 25 MB output data. Runs were completed between 8 and 24 hours (wall-clock), a significant improvement to the 3-4 days necessary on conventional compute cluster resources. In this case the speedup was due to the larger number of machines in the Nereus desktop cloud which were, in many cases, more powerful than a single node in the HPC cluster.

6.2 Gravity Wave Filter Selection (Dr Linqing Wen)

A reliable and robust method to find the optimal parameters for detecting gravity wave signals in noisy data is to generate artificial gravity wave signals, corrupt them with suitable noise and then test the ability of the software to detect the hidden signals. The initial run consisted of approximately 200000 simulations each taking 16MB data input, 10-15 minutes CPU time, and producing a few bytes of output. Execution proceeded at 20000-40000 per day for several days, during which time algorithms

---

[8] A typical server can manage the load from at least 100 clients, and will often cope with 1000.

were refined and corrections applied immediately. Currently the team is working on final code which will then be run for several million simulations. The results will feed directly into the detector software of international gravity wave detectors.

6.3 Source Finding (Dr Martin Meyer)

Examining sky survey data is an obviously parallel data processing challenge. Sky survey data can be broken up into small pieces representing a single "pixel" of the sky (at the resolution of the survey), and each piece can then be processed in parallel to determine whether any interesting phenomena such as pulsars are present [76]. In this case the goal was to test the accuracy and reliability of pulsar detection software, so ASKAP simulated sky survey data and the detection algorithm taken from Boyce[33] were used.

The 3.5GB input data was divided into 500 pixels of 5-7MB data files. Each file required 5 minutes CPU time with negligible data output. Runs were completed in $\approx$ 3 hours, and further source finding algorithm(s) are being investigated for testing in due course. Eventual ASKAP cubes are expected to be 4.5TB/day, which could be processed easily in less than 24 hours with a few hundred desktop machines.

6.4 Summary of Example Applications

Several versions of the code were run over the 8 week engagement period in Perth, and subsequently remotely from Oxford, and all new code was distributed to client workers as necessary via the Nereus system. No further actions were required on the client machines themselves to redeploy application software.

Client machines used were distributed around UWA, iVEC and Curtin over a wide range of computer types. At one extreme, Nereus was installed on the worker nodes of a standard HPC Linux cluster outside the control of an already installed batch processing manager. NereusV ran at low priority and thus did not interfere with the normal processing of the cluster, but made it more efficient by using idle time for productive work automatically. At the other extreme the Nereus client ran on normal office PCs, and the users noticed no effects of having it processing in the background.

The 10Gbit links across Perth between UWA, Curtin and iVEC were instrumental to the smooth running of the systems, as indeed were the 1GBit links around the various buildings in each campus. In particular, transferring 68MB of input data for the first application from the application server to a desktop PC in another campus was observed to take approximately 3 seconds, leaving plenty of bandwidth free for other clients to get their input data from the application server. The Nereus servers, in ICRAR and WASP, required no reboots during any run and were able to keep pace easily with the network loads placed on them by managing the clients' connections to the application server(s).

Each of the three applications was adapted to work within the NereusV environment with approximately 2 days' software engineering time. Typical changes to the original code were to convert the source from C/C++/FORTRAN into Java, and to modify the I/O operations to/from the filesystem to network requests (HTTP GET and POST back to the application server).

In each case, the converted code was tested for accuracy and performance, owing to the common misconception that Java code is slower and/or less accurate than C/C++/Fortran. In each of the above applications the converted Java code produced results well within acceptable precision[9] and actually ran 10-30% *faster* than the original C/C++/FORTRAN. Only when the Gravity Wave code was modified to include calls to the FFTW Fourier Transform library[13] did the C version perform better (the JTransforms pure Java fourier transform library was used in this case to mimic the FFTW calls [79]). Even here the performance penalty was at most 30%, so the overall throughput with the Nereus cloud (with many more computers) was still much higher.

These applications demonstrate that

1. astrophysics data processing can be converted into Nereus applications with sometimes minimal software engineering effort;
2. the speed of Java versions is very competitive with native code, and distributed dormant computing power can be harnessed effectively;
3. the high-speed networks found in a typical academic setting make accessing distributed resources a compelling mechanism to release a substantial research capability; and
4. deploying the NereusV desktop cloud within existing institutional IT infrastructure is simple and easy to manage, and caused no issues for the IT administration staff.

7 Public Outreach

Public engagement is an increasingly important aspect of large research programmes such as the Square Kilometre Array (SKA) [67]. Aside from the general benefits of scientific awareness and education that arise from such engagement, such activities are essential for the continuation of their public support.

The term"Citizen Science" [67] has emerged to identify those aspects of projects which can be accessed and in some cases actively contributed to by an inspired yet lay audience. By engaging the public, these activities provide important validation

---

[9] Converted Java code often used Double instead of C/Fortran Float as the speed difference was not significant and the higher accuracy of Java Double made it a worthwhile tradeoff.

of government support, and in some cases the public contributions have been critical to the viability of an otherwise daunting research project [68, 66].

A desktop cloud infrastructure, as in other astrophysical data reprocessing projects [73, 12, 23], can provide substantial extra computing resources but becomes even more compelling when combined with an active public outreach activity. ICRAR already has dedicated outreach staff who are currently planning the extension of the desktop cloud built in this study to a broader public.

NereusV is particularly suited to public distribution due to the ease of installation, maintenance and management:

- **Awareness:** advertise on-line and in print to encourage visitors to an SKA sponsored project website.
- **Participation:** join by clicking on a link - no separate install, elevated administrative privileges, or left over data on the donor machine. Join and leave at will without notice, no need to register clients centrally.
- **Security:** All code runs within the Java sandbox, no access to the local filesystem and all network traffic restricted through the single Nereus server. No installed files/data are left cluttering up the donor's computer.
- **Flexibility:** Project code runs identically on all systems, and new versions can be pushed out from central project servers within minutes.
- **Cost:** NereusV is open source GPLv2 and therefore free.

Planning to use large amounts of donated computing time from the public, government or industry requires careful planning. However previous projects have documented key lessons [68, 63] and based on the long tradition of Astrophysics and a positive public image of science in Australia, a significant contribution from outside academia should be possible. Indeed, as at September 2010, 38,000 Australian volunteers are actively donating computing time to the BOINC desktop cloud ranking Australia 8^{th} in the world and 1^{st} in the world per person [8] in relation to that US led initiative.

Education has been in recent years Australia's 3^{rd} most valuable export [64], so in light of this growing importance it might be expected that there had been a commensurate growth in computing resources to support this expanding research and student base. Unfortunately Australia's share of supercomputing power worldwide has *decreased* over recent years, owing mainly to the rapid advances of other nations relative to Australia [25, 48]. The petaFLOPS capable Pawsey centre [48], being built in part to support the ASKAP pathfinder goes some way to redress the balance, but a large gap still remains [25].

Against this background, the Australian government has been developing one of the most ambitious national ICT infrastructure plans in the world—the **National**

Broadband Network [22]. This AUD 37 billion project started at the end of 2010 and when complete will connect 93% of homes and businesses with a network delivering at least 100MBit/sec connection speed. Australia's personal computers could therefore form the most powerful computing resource in the world if only **2.5%** of owners were persuaded to join a public desktop cloud.

Such a resource would make an invaluable contribution to Australia's research and education sector, as well as Australian industry in general.

8 Conclusions and Recommendations

The Square Kilometre Array represents an exciting opportunity for a generation of Astrophysicists and the wider public. The ambitious plan pushes many aspects of technology, including the need for extraordinary amounts of digital data processing—estimated to require, in time, the largest aggregation of computing power ever conceived for a single project.

Cloud computing has emerged as an essential component of ICT worldwide, and already the combined computing infrastructure in "Cloud" datacentres is possibly the only source of conventional HPC facilities on the scale required that could be accessed at reasonable price and timescale. As detailed in this report, beyond the initial data processing and archive storage for ASKAP, cloud computing providers can supply today the processing power necessary for generating the scientific research results. Renting existing resources lowers overall project risk, eliminates the need to build extensive datacentres and can obviate other HPC capital investments by converting them into recurrent expenditure. Cloud providers also upgrade the hardware over time, again eliminating vendor lock-in, support, and hardware refresh costs from a long term project such as the multi-decade SKA. It is recommended that the SKA community engage with cloud computing vendors, especially those which provide HPC capabilities, to make sure sufficient infrastructure is available for use by the community, and ultimately to negotiate volume discount pricing for these resources.

The vast data processing needed for Astrophysics over the coming decade will only be possible by efficient use of multi-core, distributed computing [58]. Such ICT architectural shifts demand new software with new algorithms tailored to exploit them [58, 43, 59, 42, 50]. The rapid adoption of cloud computing has and will continue to influence the design of basic computing "boxes" and the HPC facilities they operate in, but the equally rapid advances in network technology should not be overlooked [42]. High speed wide-area networks make distributed computing more sensible, even for applications traditionally assumed to work only within a single HPC centre [78], and as shown in this report and elsewhere many Astrophysical data processing tasks can be adapted efficiently to use distributed ICT infrastructure [73, 23]. Desktop Cloud computing, representing many "small" computers connected over a surprisingly good WAN (such as the NBN), could provide a substantial contribution [50, 44].

It is recommended that while developing better parallel algorithms for HPC resources, how desktop clouds can be used alongside should also be examined seriously. It is also recommended that the SKA community help build a shared desktop cloud resource, starting with a few SKA institutes, on which to deploy and test these applications as part of the SKA construction. The combination of Moore's-Law increases of the average PC and the unique network upgrade being deployed by the Australian National Broadband project could become a world-leading computing infrastructure to complement the ambitious Astrophysics vision that is the SKA.

References

[1] Advanced research computing at cardiff universtiy. `http://www.cardiff.ac.uk/arcca/index.html`.

[2] Amazon launches apac cloud. `http://www.zdnet.com.au/amazon-launches-apac-cloud-339302775.htm`.

[3] Amazon web services. `http://aws.amazon.com/`.

[4] Askap - australian square kilometre array pathfinder. `http://www.atnf.csiro.au/SKA/`.

[5] Atlas experiment. `http://atlas.ch/`.

[6] Bell Labs breaks optical transmission record, 100 Petabit per second kilometer barrier. `http://www.physorg.com/news173455192.html/`.

[7] Boinc - berkley open infrastructure for network computing. `http://boinc.berkeley.edu/`.

[8] Boinc all project stats. `http://www.allprojectstats.com/showproject.php?projekt=0`.

[9] Campus grid special interest group. `http://wikis.nesc.ac.uk/escinet/Campus_Grids`.

[10] Canarie. `http://www.techeye.net/hardware/canada-launches-100-gigabit-network`.

[11] Cyclone - HPC Cloud Computing. `http://www.sgi.com/products/hpc_cloud/cyclone/`.

[12] Einstien at home. `http://einstein.phys.uwm.edu/`.

[13] Fftw. `http://www.fftw.org/`.

[14] Gartner worldwide cloud services market report. `http://www.gartner.com/it/page.jsp?id=1389313`.

[15] Grid computing. `http://en.wikipedia.org/wiki/Grid_computing`.

[16] Highlights of cern history. `http://public.web.cern.ch/press/Journalists/CERNHistory.html`.

[17] iVEC the hub of advanced computing in western australia. `http://www.ivec.org/`.

[18] Large synoptic survey telescope. `http://www.lsst.org/lsst`.

[19] The lhc - large hadron collider. `http://lhc.web.cern.ch/lhc/`.

[20] Lofar. `http://www.lofar.org/`.

[21] Lsst reference design. `http://lsst.org/files/docs/LSST-RefDesign.pdf`.

[22] NBNCo Ltd. `http://www.nbnco.com.au/`.

[23] Seti@home classic: In memoriam. `http://setiathome.berkeley.edu/classic.php`.

[24] Ska telescope - home. `http://www.skatelescope.org/`.

[25] Top 500 supercomputer sites. `http://www.top500.org/`.

[26] Wikipedia - exabyte. `http://en.wikipedia.org/wiki/Exabyte`.

[27] Amazon web services mulls australian data centre. `http://www.computerworld.com.au/article/344809/amazon_web_services_mulls_australian_data_centre/`, April 2010.

[28] ASKAP Technologies: Computing. `http://www.atnf.csiro.au/projects/askap/computing.html`, July 2010.

[29] P Alexander, J Bregman, and A Faulkner. SKA data flow and processing. In A Torchinsky, A van Ardenne, T van den Brink-Havinga, van Es A, and A Faulkner, editors, *WIDEFIELD SCIENCE AND TECHNOLOGY FOR THE SKA*, pages 119–123, 2009.

[30] Rasmus Andersen and Brian Vinter. Performance and portability of the sciby virtual machine. *Enabling Technologies, IEEE International Workshops on*, 0:252–257, 2010.

[31] David Anderson. Boinc: A system for public-resource computing and storage. In *Fifth IEEE/ACM International Workshop on Grid Computing (GRID'04)*. IEEE, 2004.

[32] Hugh Beedie. Campus grids business cases. `http://wikis.nesc.ac.uk/_escinet/files/c/c7/August_2010_Business_Plan.pdf`, August 2010.

[33] P. J. Boyce. Gammafinder: a java application to find galaxies in astronomical spectral line datacubes. Master's thesis, School of Computer Science, Cardiff University, 2003.

[34] R. P. Bruin, T. O. H. White, A. M. Walker, K. F. Austen, M. T. Dove, R. P. Tyer, P. A. Couch, I. T. Todorov, and M. O. Blanchard. Job submission to grid computing environments. *Concurr. Comput. : Pract. Exper.*, 20(11):1329–1340, 2008.

[35] Ali Raza Butt, Sumalatha Adabala, Nirav H. Kapadia, Renato J. Figueiredo, and José A. B. Fortes. Grid-computing portals and security issues. *J. Parallel Distrib. Comput.*, 63(10):1006–1014, 2003.

[36] Franck Cappello, Samir Djilali, Gilles Fedak, Thomas Herault, Frédéric Magniette, Vincent Néri, and Oleg Lodygensky. Computing on large-scale distributed systems: Xtrem web architecture, programming models, security, tests and convergence with grid. *Future Gener. Comput. Syst.*, 21(3):417–437, 2005.

[37] Andrew Chien, Brad Calder, Stephen Elbert, and Karan Bhatia. Entropia: architecture and performance of an enterprise desktop grid system. *J. Parallel Distrib. Comput.*, 63(5):597–610, 2003.

[38] Xingchen Chu, Krishna Nadiminti, Chao Jin, Srikumar Venugopal, and Rajkumar Buyya. Aneka: Next-generation enterprise grid platform for e-science and e-business applications. In *E-SCIENCE '07: Proceedings of the Third IEEE International Conference on e-Science and Grid Computing*, pages 151–159, Washington, DC, USA, 2007. IEEE Computer Society.

[39] Walfredo Cirne, Francisco Brasileiro, Nazareno Andrade, Lauro Costa, Alisson Andrade, Reynaldo Novaes, and Miranda Mowbray. Labs of the world, unite!!! *Journal of Grid Computing*, 4:225–246, 2006.

[40] The ATLAS Collaboration. Expected performance of the atlas experiment: detector, trigger and physics, December 2008.

[41] J. Cordes, D. Campbell, N. Director, F. Lo, and N. Director. The Square Kilometre Array. `http://science.nrao.edu/A2010/rfi/SKA.pdf`, 2009.

[42] T. Cornwell and B Glendenning. Software development for the Square Kilometre Array. `http://www.skatelescope.org/PDF/memos/50_cornwell.pdf`, June 2004. SKA Memo Series.

[43] T. Cornwell and B. Humphreys. SKA Exascale Software Challenges. `http://www.skatelescope.org/PDF/memos/128_Memo_Cornwell.pdf`, October 2010. SKA Memo Series.

[44] Tim Cornwell. Askap industry technical briefing. `http://www.atnf.csiro.au/SKA/ASKAP_Computing_July09.pdf`, July 2009.

[45] Tim Cornwell. Climbing mount exaflop - computing for the askap and the square kilometre array. `http://www.atnf.csiro.au/people/tim.cornwell/presentations/askapcurtin2009.pdf`, September 2009.

[46] Tim Cornwell. Mount exaflop. `http://www.atnf.csiro.au/people/tim.cornwell/publications/MountExaflop.pdf`, September 2009.

[47] Attila Csaba Marosi, Peter Kacsuk, Gilles Fedak, and Oleg Lodygensky. Using virtual machines in desktop grid clients for application sandboxing. Technical Report TR-0140, Institute on Architectural Issues: Scalability, Dependability, Adaptability, CoreGRID - Network of Excellence, August 2008.

[48] CSIRO. The pawsey high performance computing centre for ska science. `http://www.aph.gov.au/house/committee/pwc/pawseycentre/subs/sub001.pdf`.

[49] M. de Vos, A.W. Gunst, and R. Nijboer. The LOFAR telescope: System architecture and signal processing. *Proceedings of the IEEE*, 97(8):1431–1437, 2009.

[50] P. Dewdney and J. et al bij de Vaate. SKA Phase 1: Preliminary System Description. `http://www.skatelescope.org/PDF/memos/130_Memo_Dewdney.pdf`, November 2010. SKA Memo Series.

[51] Marcos Dias de Assuncao, Krishna Nadiminti, Srikumar Venugopal, Tianchi Ma, and Rajkumar Buyya. An integration of global and enterprise grid computing: Gridbus broker and xgrid perspective. In *Proceedings of the 4th International Conference on Grid and Cooperative Computing*. Springer-Verlag, Berlin, 2005.

[52] G.J. Eilenberger, S. Bunse, L. Dembeck, U. Gebhard, F. Ilchmann, W. Lautenschlaeger, and J. Milbrandt. Energy-efficient transport for the future Internet. *Bell Labs Technical Journal*, 15(2):147–167, 2010.

[53] J. Gray and A.S. Szalay. Where the rubber meets the sky: Bridging the gap between databases and science. *Arxiv preprint cs/0502011*, 2005.

[54] J.C. Guzman and B. Humphreys. The Australian SKA Pathfinder (ASKAP) software architecture. In *Proceedings of SPIE*, volume 7740, page 77401J, 2010.

[55] Peter Hall. The ska overview and update. `http://www.skatelescope.org/PDF/WA_industry.pdf`, 2006.

[56] Ben Humphreysl. Askap computing - askaic technical update. `http://www.atnf.csiro.au/SKA/ASKAP_Computing_BH_July09.pdf`, July 2009.

[57] Z. Ivezic, JA Tyson, R. Allsman, J. Andrew, R. Angel, T. Axelrod, JD Barr, AC Becker, J. Becla, C. Beldica, et al. LSST: from science drivers to reference design and anticipated data products. 2008.

[58] A. Kemball, J. Cordes, J. Lazio, and D. Backer. Technology development for computational radio astronomy: 2010-2020. *RADIO ASTRONOMY*, page 2020, 2010.

[59] AJ Kemball and TJ Cornwell. A Simple Model Of Software Costs For The Square Kilometre Array. *Experimental Astronomy*, 17(1):317–327, 2004.

[60] Derrick Kondo, Gilles Fedak, Franck Cappello, Andrew A. Chien, and Henri Casanova. Characterizing resource availability in enterprise desktop grids. *Future Gener. Comput. Syst.*, 23(7):888–903, 2007.

[61] F. Malek, F. Hernandez, and F. Chollet. A worldwide grid infrastructure for the four LHC experiments. 2010.

[62] Rhys Newman and Ian Preston. Nereus-v: Massively parallel computing of, by and for the community. `http://developers.sun.com/learning/javaoneonline/j1sessn.jsp?sessn=TS-5136&yr=2009&track=nextweb`.

[63] O. Nov, O. Arazy, and D. Anderson. Crowdsourcing for science: understanding and enhancing SciSourcing contribution. In *ACM CSCW 2010 Workshop on the Changing Dynamics of Scientific Collaborations*, 2010.

[64] Australian Bureau of Statistics. International students in south australia. `http://www.abs.gov.au/AUSSTATS/abs@.nsf/Lookup/1345.4Feature%20Article1Dec%202009`.

[65] Z. Patoli, M. Gkion, A. Al-Barakati, W. Zhang, P. Newbury, and M. White. How to build an open source render farm based on desktop grid computing. *Wireless Networks, Information Processing and Systems*, pages 268–278, 2009.

[66] J. Raddick, CJ Lintott, K. Schawinski, D. Thomas, RC Nichol, D. Andreescu, S. Bamford, KR Land, P. Murray, A. Slosar, et al. Galaxy Zoo: An Experiment in Public Science Participation. In *Bulletin of the American Astronomical Society*, volume 38, page 892, 2007.

[67] M.J. Raddick, G. Bracey, K. Carney, G. Gyuk, K. Borne, J. Wallin, S. Jacoby, and A. Planetarium. Citizen Science: Status and Research Directions for the Coming Decade. *AGB Stars and Related Phenomenastro2010: The Astronomy and Astrophysics Decadal Survey*, 2010:46P, 2009.

[68] M.J. Raddick, G. Bracey, P.L. Gay, C.J. Lintott, P. Murray, K. Schawinski, A.S. Szalay, and J. Vandenberg. Galaxy Zoo: Exploring the Motivations of Citizen Science Volunteers. *Astronomy Education Review*, 9:010103, 2010.

[69] K. Roberts, D. Beckett, D. Boertjes, J. Berthold, and C. Laperle. 100G and beyond with digital coherent signal processing. *Communications Magazine, IEEE*, 48(7):62–69, 2010.

[70] Sriya Santhanam, Pradheep Elango, Andrea Arpaci-Dusseau, and Miron Livny. Deploying virtual machines as sandboxes for the grid. In *WORLDS'05: Proceedings of the 2nd conference on Real, Large Distributed Systems*, pages 7–12, Berkeley, CA, USA, 2005. USENIX Association.

[71] Jean Schneider. Ska telescope to provide a billion pcs' worth of processing. `http://www.computerworld.com.au/article/319128/ska_telescope_provide_billion_pcs_worth_processing_updated_/`, September 2009.

[72] D. Sepulveda and S. Goasguen. The Deployment and Maintenance of a Condor-Based Campus Grid. *Advances in Grid and Pervasive Computing*, pages 165–176, 2009.

[73] A. Siemion, J. Von Korff, P. McMahon, E. Korpela, D. Werthimer, D. Anderson, G. Bower, J. Cobb, G. Foster, M. Lebofsky, et al. New SETI sky surveys for radio pulses. *Acta Astronautica*, 2010.

[74] Ian C. Smith. Towards a greener condor pool: adapting condor for use with energy-efficient pcs. `http://wikis.nesc.ac.uk/_escinet/files/e/ea/Power_save.pdf`.

[75] P.M. Smith, T.J. Hacker, C.X. Song, and D.P.C. Center. Implementing an Industrial-Strength Academic Cyberinfrastructure at Purdue University. In *Second Workshop on Desktop Grids and Volunteer Computing*, 2008.

[76] R. Smits, M. Kramer, B. Stappers, DR Lorimer, J. Cordes, and A. Faulkner. Pulsar searches and timing with the SKA. *A&A*, 493:1161, 2009.

[77] G. Veith, E. Lach, and K. Schuh. 100 Gigabit-per-second: Ultra-high transmission bitrate for next generation optical transport networks. *Comptes Rendus Physique*, 9(9-10):1002–1011, 2008.

[78] K. Verstoep, J. Maassen, H.E. Bal, and J.W. Romein. Experiences with fine-grained distributed supercomputing on a 10g testbed. In *Cluster Computing and the Grid, 2008. CCGRID'08. 8th IEEE International Symposium on*, pages 376–383. IEEE, 2008.

[79] Piotr Wendykier. Jtransforms. `http://sites.google.com/site/piotrwendykier/software/jtransforms`.

[80] Ming Zhao, Jian Zhang, and Renato J. Figueiredo. Distributed file system virtualization techniques supporting on-demand virtual machine environments for grid computing. *Cluster Computing*, 9(1):45–56, 2006.