

On the Performance Enhancement of the WISDOM Production Environment

BUI The Quang¹

Institut de la Francophonie pour l'Informatique (IFI)

Building D, alley 42, Ta Quang Buu Rd., Hai Ba Trung Dist, Hanoi, Vietnam

E-mail: buithequang@gmail.com

DOAN Trung Tung

Institut de la Francophonie pour l'Informatique (IFI)

Building D, alley 42, Ta Quang Buu Rd., Hai Ba Trung Dist, Hanoi, Vietnam

E-mail: dttung@gmail.com

NGUYEN Trung Hung

Institute of Natural Products Chemistry (INPC)

18 Hoang Quoc Viet Rd., Cau Giay Dist., Hanoi, Vietnam

E-mail: shortjean@gmail.com

PHAM Minh Quan

Institute of Natural Products Chemistry (INPC)

18 Hoang Quoc Viet Rd., Cau Giay Dist., Hanoi, Vietnam

E-mail: minhquanaries@gmail.com

NGUYEN Van Son

Institute of Natural Products Chemistry (INPC)

18 Hoang Quoc Viet Rd., Cau Giay Dist., Hanoi, Vietnam

E-mail: son.nguyen.pt@gmail.com

Emmanuel MEDERNACH

Laboratoire de Physique Corpusculaire, CNRS/IN2P3

24 avenue des Landais, BP10448, 63000 Clermont-Ferrand, France

E-mail: medernach@clermont.in2p3.fr

¹ Speaker

Vincent BRETON

Laboratoire de Physique Corpusculaire, CNRS/IN2P3
24 avenue des Landais, BP10448, 63000 Clermont-Ferrand, France
E-mail: breton@clermont.in2p3.fr

NGUYEN Hong Quang

Institut de la Francophonie pour l'Informatique (IFI)
Building D, alley 42, Ta Quang Buu Rd., Hai Ba Trung Dist, Hanoi, Vietnam
E-mail: nguyen.hong.quang@auf.org

PHAM Quoc Long

Institute of Natural Products Chemistry (INPC)
18 Hoang Quoc Viet Rd., Cau Giay Dist., Hanoi, Vietnam
E-mail: phamquoclong1960@gmail.com

LE Mai Huong

Institute of Natural Products Chemistry (INPC)
18 Hoang Quoc Viet Rd., Cau Giay Dist., Hanoi, Vietnam
E-mail: lehuong00@gmail.com

Abstract

The WPE (WISDOM Production Environment) is a job management system on the grid, which is successfully used in several virtual screening projects for malaria, such as WISDOM-I (2005) and WISDOM-II (2006). The experience using WPE showed however that there are still challenges to address such as agents massively killed when the Task Manager is empty, or the status monitoring module noticeably slowing down after a long execution time. We carried out tests to measure WPE performances and compare them to DIRAC's, a similar job management system. The obtained results confirm the problems encountered. To improve WPE performances, we discuss the reasons and propose corrective plans. This paper describes the scenario and results of the performance tests for the WPE before and after the corrections.

Keyword: grid computing, WISDOM Production Environment, DIRAC

The International Symposium on Grids and Clouds (ISGC) 2012
Academia Sinica, Taipei, Taiwan
February 26 – March 2, 2012

1. Introduction

The WISDOM Production Environment (WPE) platform, developed by the LPC Clermont-Ferrand (France) laboratory, was successfully used for the **Wide In Silico Docking On Malaria (WISDOM)** projects [12] that discovered new active inhibitors against malaria. In these projects, the authors used In Silico docking approach for drug discovery where computational chemistry software tools are used to evaluate the binding energy between a key protein and a ligand in order to select the ligands that dock to the protein active site. The WPE platform provided a layer between the users and the grid environment in order to hide its complexity. With this platform, users could easily use grid's resources for their calculation.

Despite its successful use for large scale data challenges [12], the WPE platform faces some limitations: (i) its Information Service slows down after a long execution time, (ii) pilot agents are killed for unknown reasons, (iii) the WPE platform stops when the Task Manager is empty for a long time.

In this paper, we present performance tests of the WPE platform in comparison to the DIRAC platform [9]. We also propose some corrections for WPE improvement and document their impact.

The rest of the paper is organized as follows:

- In section 2, we introduce the WPE and DIRAC platforms
- In section 3, we present the performance test scenario, discuss the results and propose some improvements.
- Finally, in section 4 we conclude on performance tests and present some perspectives

2. WPE and DIRAC platforms

2.1 The WPE platform

The WPE platform [2], developed by the LPC Laboratory in Clermont-Ferrand, France, was successfully used in the WISDOM-I (2005) and WISDOM-II (2006) projects as a layer between users and grid resources. It facilitates the massive submission of jobs to the grid. The WPE platform is built on top of gLite middleware [5] (see *Figure 1*).

The WPE user interface makes use of the command line rather than a graphical interface. Client has to communicate with the system through Linux-like command lines. The upload of tasks and download of results are processed by written commands. As a result, a beginner will have significant difficulty to get accustomed to this system. The WPE platform was specifically

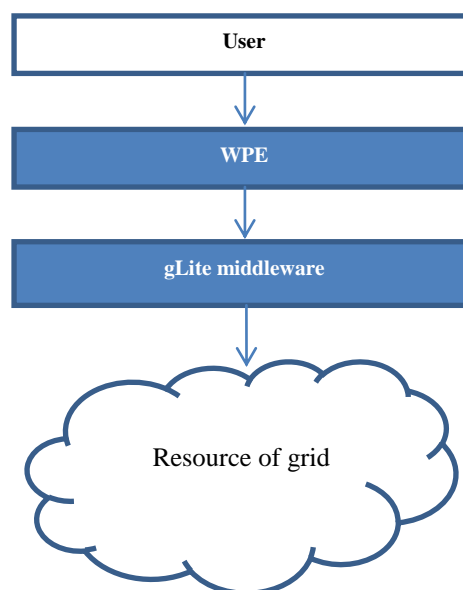


Figure 1: WISDOM Production Environment

built for the WISDOM project and therefore its use was restricted to a few people for specific experiments (WISDOM - I and II).

The WPE platform is composed of three modules, as illustrated on figure 2.

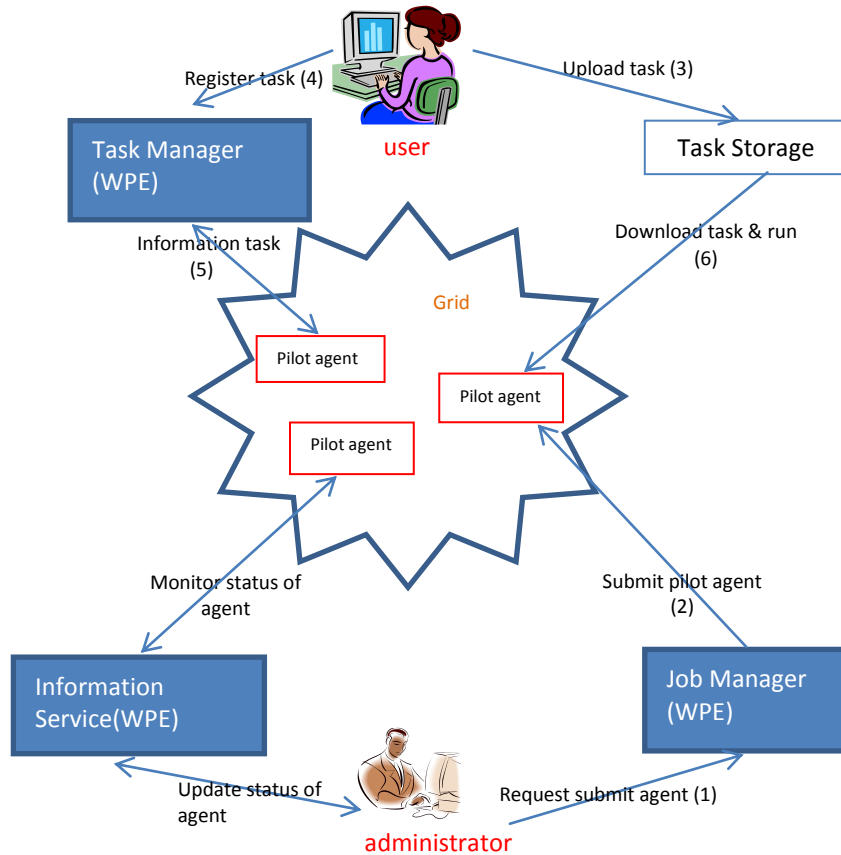


Figure 2: Structure of WPE

First, the “**Job Manager**” module receives administrator requests and submits pilot agents to the grid. Second, the “**WISDOM Information Service (WIS)**” module controls the information of pilot agents on the grid, for example its status or the Computing Element on which the pilot agent is submitted. Pilot agents on the grid update always their status information on WIS. Third, the “**Task Manager**” module manages tasks created by the user and their status.

2.2 The DIRAC platform

The DIRAC (Distributed Infrastructure with Remote Agent Control) platform is a complete Grid solution for a community of users needing access to distributed computing resources [9]. DIRAC forms a layer between a particular community and various

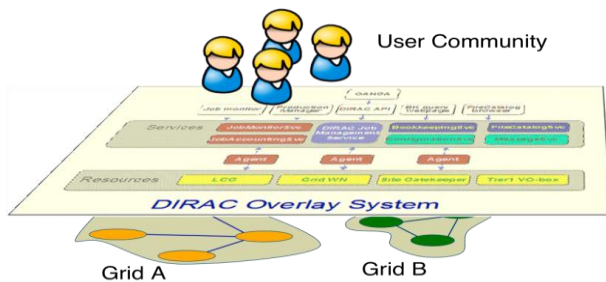


Figure 3: Use of DIRAC (Source: <http://diracgrid.org>)

computing resources to allow optimized, transparent and reliable usage (see *Figure 3*).

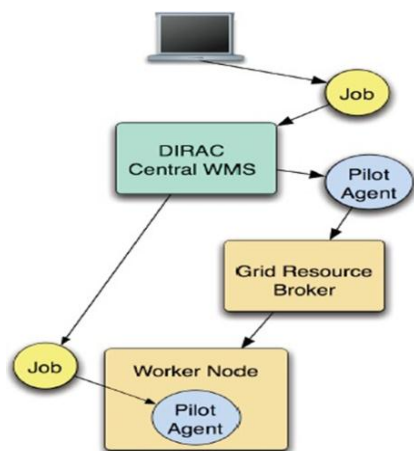


Figure 4: Job submission of DIRAC

The DIRAC platform allows deploying pilot agents on the Worker Nodes as regular jobs using the standard grid scheduling mechanism. It forms a distributed Workload Management system and reserves the resources for immediate use (see *Figure 4*).

Once started on the Worker Node, the pilot agent performs some checks upon the environment: it measures the CPU performance, disk and memory space, and also installs the application software.

If the Worker Node is OK the user job is pulled from the central DIRAC Task Queue and executed.

2.3 Comparison between the WPE and DIRAC platforms

The WPE and DIRAC platforms have the same goal, namely to help users easily communicate with the grid environment, submit, control and monitor their jobs. However, their architectures are quite different. Table 1 shows the differences between the two platforms.

WPE platform	DIRAC platform
First, a pilot agent is submitted to the grid. Then, the user creates a task in the Task Manager. (<i>Pull model</i>)	First, the user creates a task. Then, a pilot agent is submitted to the grid for executing this task (<i>Push model</i>)
One WPE pilot agent executes many tasks	One DIRAC pilot agent executes one task

Table 1: Differences between the WPE and DIRAC platforms

3. Performance tests

3.1 Application used

The AUTODOCK application was used for our test. This is a suite of automated docking tools, designed to predict how small molecules, such as substrates or drug candidates, bind to a receptor of known 3D structure [11]. In the WISDOM-I and WISDOM-II projects, AUTODOCK was used for docking on the grid. For each docking, 3 to 5 minutes of computing time was needed on one PC. Docking computations are particularly well fitted to grid infrastructures because they are heavily parallel: the results of different dockings are independent from each other. As such if a task fails, it doesn't influence the results of other tasks. This makes it easy to divide docking computations into many small dockings and execute them simultaneously on different computers on the grid.

In our performance tests, the WPE and DIRAC platforms were used to deploy AUTODOCK on the EGI-INSPIRE [13] Biomed Virtual Organization. The DIRAC platform is compared to the WPE platform in some tests to show that problems of submission and stability

of the pilot agents are not only caused by the status of the grid infrastructure, but also by the conceptual design of the WPE platform.

3.2 Goal 1: measure the performance of task execution on the WPE & DIRAC platforms

3.2.1 Scenario

The chosen amounts of AUTODOCK tasks to execute were 100, 500, 1000, 5000 and 10.000. The numbers of pilot agents submitted by the WPE platform were fixed to 200, 500 or 1000. We then measured the total time for finishing all tasks. For each selected number of tasks and pilot agents, we repeated the performance tests 5 times and calculated the average total time.

3.2.2 Results and analysis

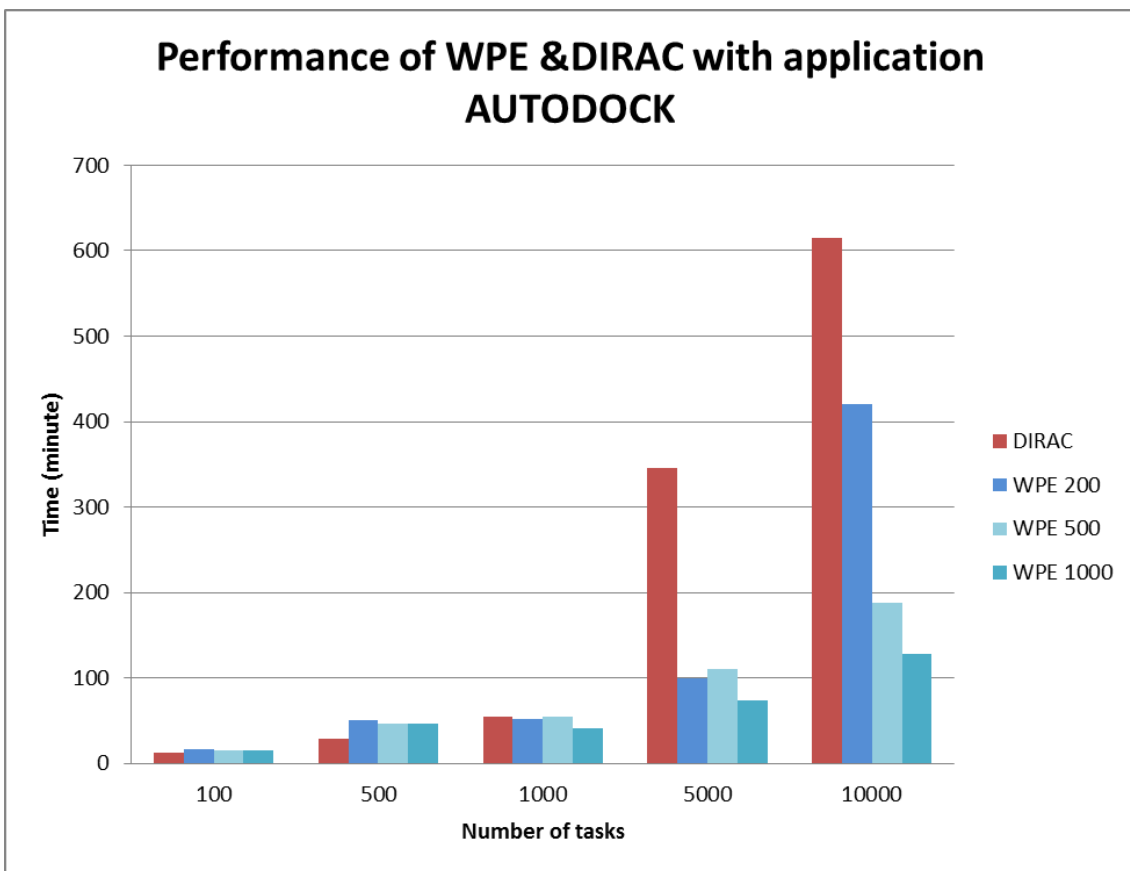


Figure 5: Total execution time for tasks on the WPE & DIRAC platforms

Figure 5 shows the results of the performance tests. In this graph, WPE 200/500/1000 stands for the submission of 200/500/1000 pilot agents on grid by the WPE platform. When the number of tasks is smaller than 1000, the total time for finishing all tasks is shorter on the DIRAC platform. This is due to the fact that the DIRAC platform submits pilot agents faster than the WPE platform because of its selection algorithm and submission method.

On the other hand, when the number of tasks is larger than 1000, the total time for finishing all tasks is greater on the DIRAC platform, because WPE pilot agents can execute many tasks while DIRAC pilot agents execute only one task.

3.3 Goal 2: To study the WPE platform shut down

3.3.1 Scenario

Experience shows that after using WPE for a day, systems do not react to new incoming tasks, which forces the user to restart the submission procedure. The cause of this problem could be that the Task Manager remains empty for a long time, so WPE is stopped (for example at night, no user submits docking tasks, so there are no tasks in the Task Manager). We used the following scenario to test this hypothesis: we started WPE first, and pushed tasks continuously to the Task Manager. We then stopped pushing tasks for about 1 hour and pushed again new tasks to the Task Manager.

3.3.2 Results and analysis

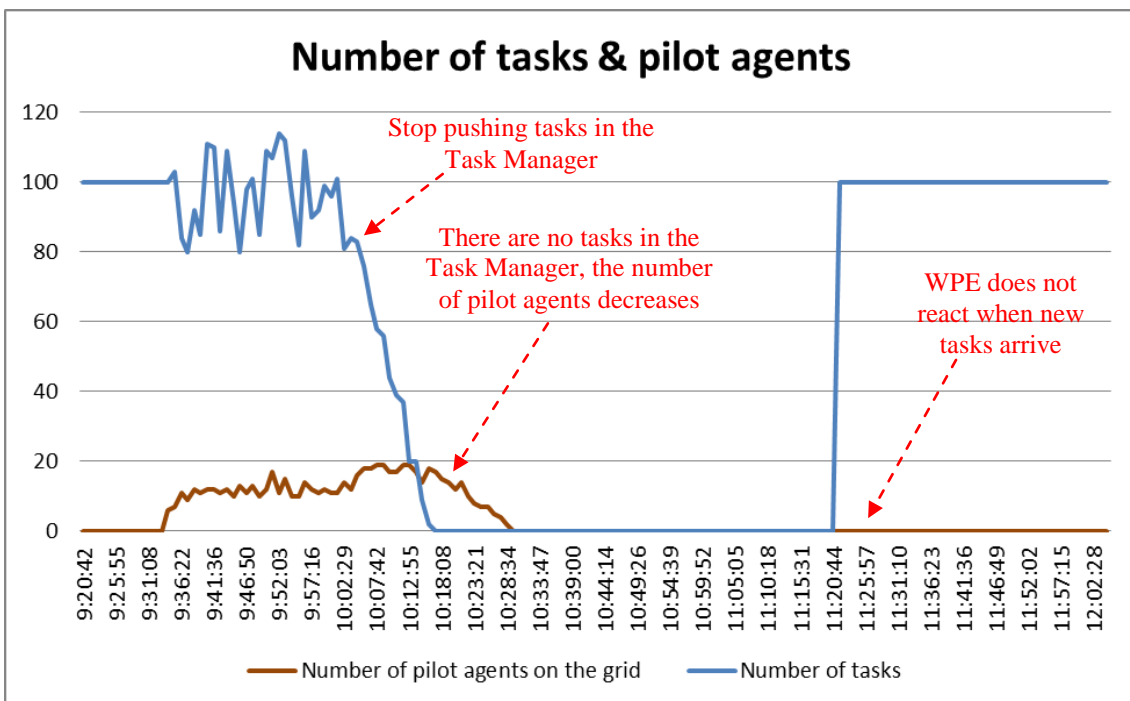


Figure 6: Number of running agents on the grid and number of tasks in the Task Manager

Based on the results illustrated on Figure 6, we observe that: the number of pilot agents is reduced when there are no tasks in Task Manager. This reduction is normal because a pilot agent is programmed to wait for a period TIMEOUT (20 minutes), after which if there are no tasks, the agent is automatically terminated. However, the problem is that when new tasks arrive, WPE must resubmit agents to execute these tasks. There are no new pilot agents on the grid when new tasks arrive. The defaulting mechanism for resubmitting agents is due to incorrect information in the WISDOM Information Service (WIS). Some agents do not keep updated status information on WIS because they are suddenly killed. According to WIS data, their status is still “**running**”. Therefore, WPE cannot resubmit these agents normally.

3.3.3 The new pilot agent resubmission module

a. Mechanism of action

We have designed a new pilot agent resubmission module. Rather than letting agent updates their status on WIS, this module retrieves status information directly using gLite command. This solves the problem of resubmission caused by incorrect information in WIS.

This mechanism also allows the number of pilot agents on the grid to adapt to the number of tasks in the Task Manager. When there are no tasks in the Task Manager, the pilot agents on the grid will automatically terminate after the TIMEOUT period, thereby reducing the amount of pilot agents on the grid to release grid resources. These killed agents will be resubmitted into the grid after a period T . If there are tasks in the Task Manager, the pilot agents will exist on the grid until all the tasks are done. So when there are tasks in the Task Manager, the number of pilot agents on the grid will increase.

The length of the period T to resubmit the terminated agent is important. If T is too short, there will always be many pilot agents while there are no tasks in the Task Manager. If T is too long, the system will be slow to react to new tasks. After trying the new pilot agent resubmission module for several T values, we found $T = 30$ minutes to be reasonable on the EGI-INSPIRE Biomed Virtual Organization.

b. Result

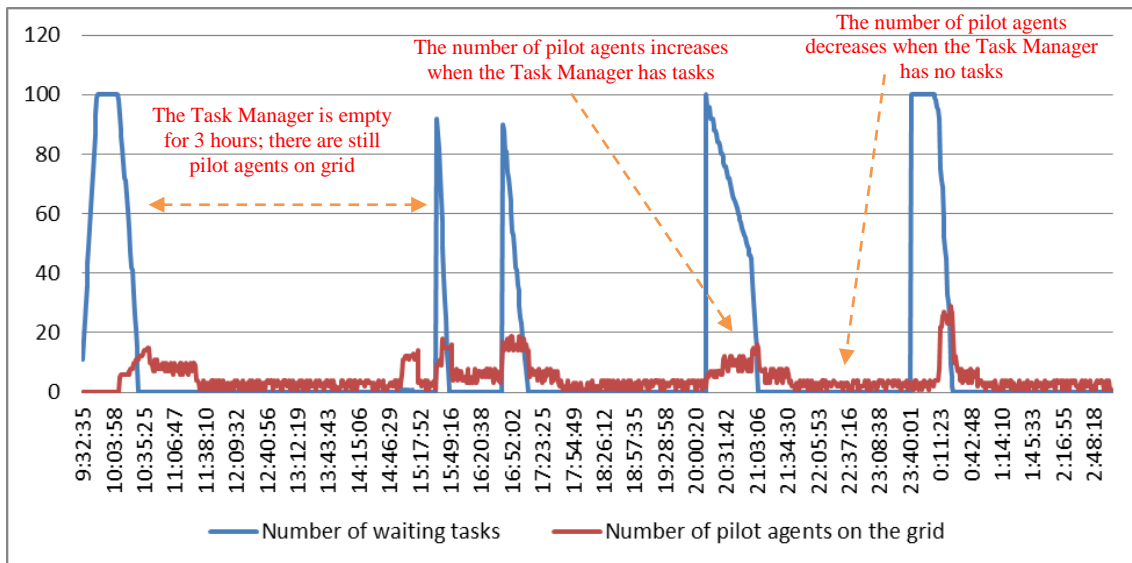


Figure 7: After adding the pilot agent resubmission module (value $T=30'$)

After running the pilot agent resubmission module with the resubmission period $T = 30'$ (see Figure 7), even when the Task Manager was left empty for 3 hours, pilot agents were still available when new tasks were pushed to the Task Manager.

Moreover, if many tasks are pushed to the Task Manager, the number of pilot agents on grid increases to execute them. On the other hand, if there are no tasks in the Task Manager, the number of agents on the grid decreases to free grid resources. So with this module, WPE is able to better adapt to the number of tasks in the Task Manager.

3.4 Goal 3: To find out if the WISDOM Information Service slows down after a long execution time

3.4.1 Scenario

First, we restarted the WISDOM Information Service (WIS). Next, we logged the time for executing a pilot agent’s query to the WIS, and calculated the average time everyday thereafter. This test was repeated for one and a half month.

3.4.2 Results and analysis

Figure 8 shows the obtained results. Execution time of a pilot agent’s query gradually increases every day showing that the WIS slows down after a long execution time.

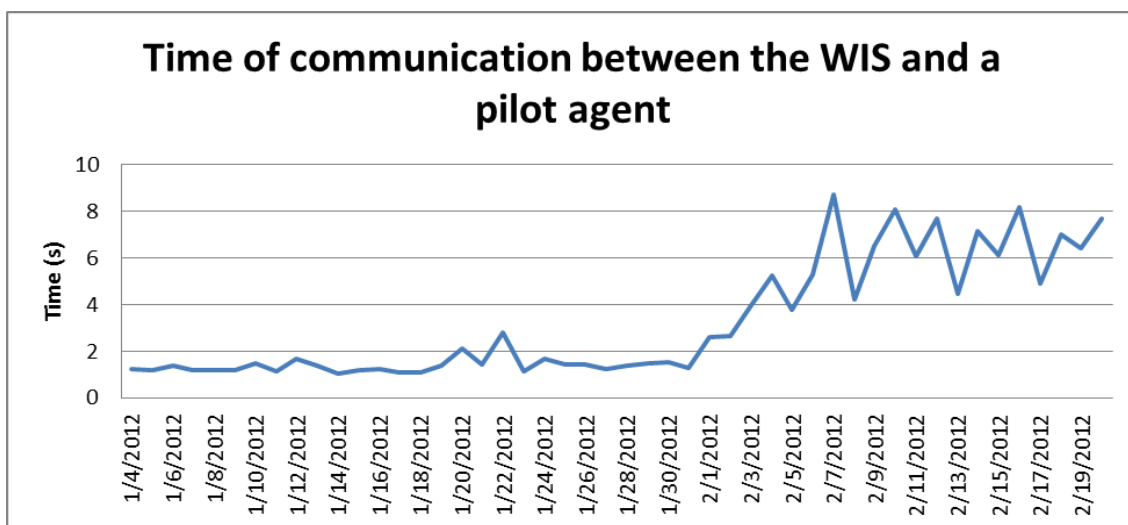


Figure 8: Time to process a pilot agent’s query to the WIS

The cause of this behavior could be the amount of data stored in the WIS. When a new agent is submitted to the grid, agent-related information is stored in the WIS. After a long execution time, the database grows and this introduces a delay in response time. Our WIS module uses the old version of AMGA with known issues for handling large volume of data.

3.5 Goal 4: To measure agent submission performances for the WPE & DIRAC platforms

3.5.1 Scenario

The WPE platform was used to submit 100/200/500/1000 agents to the grid and log the number of agents successfully submitted to the grid. The tests were repeated 5 times for each configuration. The same scenario was applied to the DIRAC platform.

3.5.2 Results and analysis

Figure 9 shows that 98% of the agents were submitted successfully to the grid by the DIRAC platform, against only 25% for WPE agents. The DIRAC platform submission scheme is better than the WPE platform one.

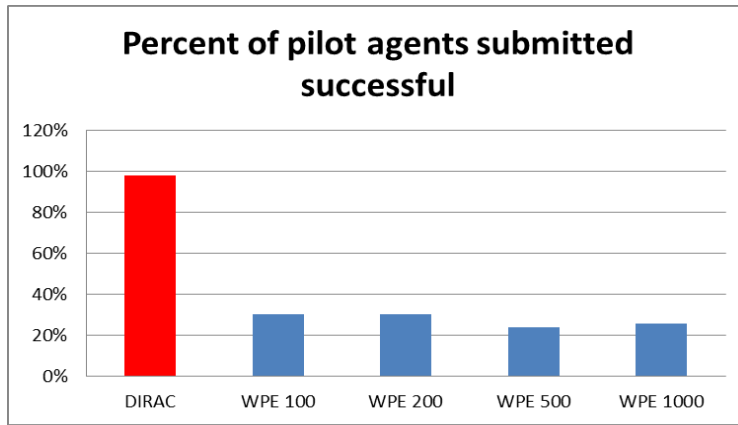


Figure 9: Rate of successful pilot agent submission

3.6 Goal 5: To measure agent stability on the grid

3.6.1 Scenario

Using the WPE platform, we submitted 100/200/500/1000 agents to the grid and log the number of pilot agents killed when running. Each case was run 5 times, from which we calculated the average. The same test was done on the DIRAC platform.

3.6.2 Results and analysis

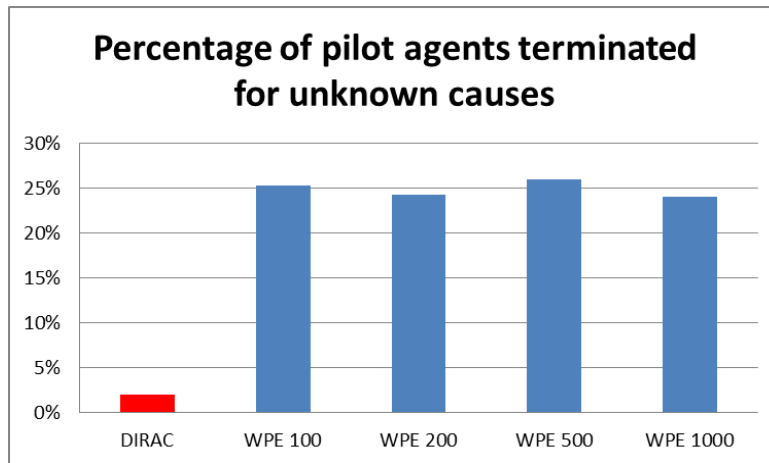


Figure 10: Stability of pilot agents on the grid

Figure 10 shows that only about 2% of pilot agents submitted with the DIRAC platform are killed. But about 25% of the pilot agents submitted by the WPE platform terminate for unknown causes. So the stability of DIRAC pilot agents is significantly better than those on the WPE platform.

The difference in the stability of WPE and DIRAC pilot agents could be the result of different mechanisms of action. WPE pilot agents are able to run an unlimited number of tasks so that their execution time could exceed allowed time on the CE. In this case, WPE pilot agents could be stopped by the CE. On the contrary, DIRAC pilot agents execute only one or a fixed

number of tasks. As a result, the execution time of DIRAC pilot agents on the CE is under control and the administrator can set the number of tasks to avoid pilot agents being killed by the CE.

4. Conclusion & perspectives

4.1 Conclusion

Performance tests demonstrate the issues below about the WPE platform:

- When the WPE platform is down for a long time, it is not able to resubmit agents
- The WISDOM Information Service slows down after a long execution time
- WPE pilot agents are not stable
- WPE pilot agent submission to the grid is less effective than DIRAC's

We fixed the WPE platform by implementing a resubmission module for pilot agents. This fix resolved the problem of the WPE platform going down when the Task Manager is empty for a long time.

4.1.1 Perspectives

The module WIS slows down after a long execution time. The WPE platform uses the AMGA database version 1.9.0 so we could use the latest version of the AMGA database for WIS.

We propose to combine the WPE and DIRAC platforms where WPE pilot agents would be controlled by the DIRAC platform. As such, we can use the DIRAC powerful agent submission scheme and use WPE pilot agents that can execute many tasks.

Another improvement could be to set up communication between the Task Manager and the agent resubmission module. If there is no task in the Task Manager, the period for resubmission is long whereas if there are many tasks in the Task Manager, the period for resubmission is reduced.

Acknowledgements

The authors acknowledge support of FVPPL International Associated Laboratory. The authors are thankful to Niobe Haitas for proof reading the document. Computing resources from France Grilles and EGI were used through the Biomed and France-Asia Virtual Organizations.

References

- [1] D. S. Goodsell, G. M. Morris, A. J. Olson, *Automated docking of flexible ligands: applications of AutoDock*, Journal of Molecular Recognition, Volume 9 Issue 196, pages 1–5, January/February 1996
- [2] V. Kasam, J. Salzemann, M. Botha et al., *WISDOM-II: screening against multiple targets implicated in malaria using computational grid infrastructures*, Malaria Journal, vol. 8, no. 1, article 88, 2009

- [3] E. Huedo, R. Moreno-Vozmediano, R. S. Montero and I. M. Llorente, *Architectures for Enhancing Grid Infrastructures with Cloud Computing*, Computer Communications and Networks, pages 55-69, 2011
- [4] P. Andretto et al., *The gLite workload management system*, Journal of Physics: Conference Series, Volume 119, part 6, 2008
- [5] E. Laure et al., *Programming the Grid with gLite*, Computational Methods in Science and Technology, pages 33 – 45
- [6] Trung Tung Doan, Hong Quang Nguyen, Ana Lucia Da-Costa, Yannick Legre, Aurelien Bernard, Lydia Maigne, Jean Salzemann, David Sarramia, Vincent Breton, Thanh Hoa Le, Duc Hung Le and Johan Montagnat, *A new flexible workflow on the Grid for monitoring H5N1*, The International Symposium on Grids and Clouds and the Open Grid Forum, Academia Sinica, Taipei, Taiwan, 2011
- [7] Jacq N., Salzemann J., Jacq F., Legré Y., Medernach E., Montagnat J., Maaß A., Reichstadt M., Schwivhtenberg H., Sridhar M., Kasam V., Zimmermann M., Hofmann M., Breton V., *Grid enabled virtual screening against malaria*, J. Grid Computing 6, 29–43 (2008)
- [8] EMBRACE Grid, [Online], Available: <http://www.embracegrid.info/page.php>
- [9] DIRAC Grid, [Online], Available: <http://www.diracgrid.org>
- [10] N. Santos and B. Koblitz, *Distributed Metadata with the AMGA Metadata Catalog*, Workshop on Next-Generation Distributed Data Management, HPDC-15, Paris, France, June 2006
- [11] Autodock, [Online], Available: <http://autodock.scripps.edu/>
- [12] WISDOM project, [Online], Available : <http://wisdom.healthgrid.org/>
- [13] EGI-Inspire, [online], Available: <http://www.egi.eu>