



マルチグリッド環境下でのアプリケーションインターフェースに関する研究

高エネルギー加速器研究機構
計算科学センター



研究開発の動機と目的

・ 動機

- KEKでは、複数のグリッドミドルウェアを実際に運用し、アプリケーションのグリッド化を行っており、利用者の立場から、異ミドルウェア混在環境におけるグリッドアプリケーション開発の効率化の必要性を強く感じている

・ 目的

- 我が国で開発されたアプリケーションの世界への普及
- 同一アプリケーションソフトウェアを、複数のグリッドミドルウェア環境で利用可能にするためのAPIの開発を行う
- GRID環境のない大学研究室も含めて、世界的に共同して行うe-Scienceに参加していくための基盤を創る
- ワークフローが複雑でプログラム言語やスクリプト言語による条件分岐の記述が必要な分野向けのユーザ環境を開発する
- Localと複数のGrid上の資源(計算機、データなど)をシームレスに利用できる環境の構築を行う



サブテーマ1との関係

- 資源利用の普遍化(サブテーマ1)
 - GRID interoperability
 - 手元にあるアプリケーションが世界中の資源を利用できる
 - 既に存在するアプリケーションに有用
 - エンドユーザに易しい
- アプリケーションの普遍化(サブテーマ4)
 - 世界中のどこでも稼働するアプリケーションを開発したい
 - APIの開発が必要
 - 新たに開発するアプリケーションに有用
 - 特に長期間利用されるもの
 - アプリケーション開発者向け



開発の目標

- ・ End Userが研究室のPCから、データ検索、データ解析、シミュレーションなどをLocal資源、Grid上の資源を問わず利用できる。ここで資源とは、各種計算機、データ格納装置、データ類、データベースなど
- ・ どのGrid Middlewareでも同じユーザ環境で利用できる。
- ・ 複数のGrid Middleware下にある資源も必要に応じて、同時に利用できる。
- ・ 大学研究室では、Grid Middlewareに関するノウハウや保守を必要としない。
- ・ OGFですでに標準化を目指しているSAGAをベースに開発し、足りない機能は独自に開発する。 SAGA: Simple API for Grid Application
- ・ ファイルカタログ情報は、RNSをベースに検索する。
- ・ アプリケーション開発ユーザには、Python環境で見せる。



似て非なるコマンド

NAREGI

- naregi-signon
<https://nrg.host.net/portal>
- naregi-job-submit submit.xml
- naregi-job-status

gLite

- voms-proxy-init -voms VO
- glite-wms-job-submit --vo VO
-a hello.jdl
- glite-wms-job-status <jobid>



JDL

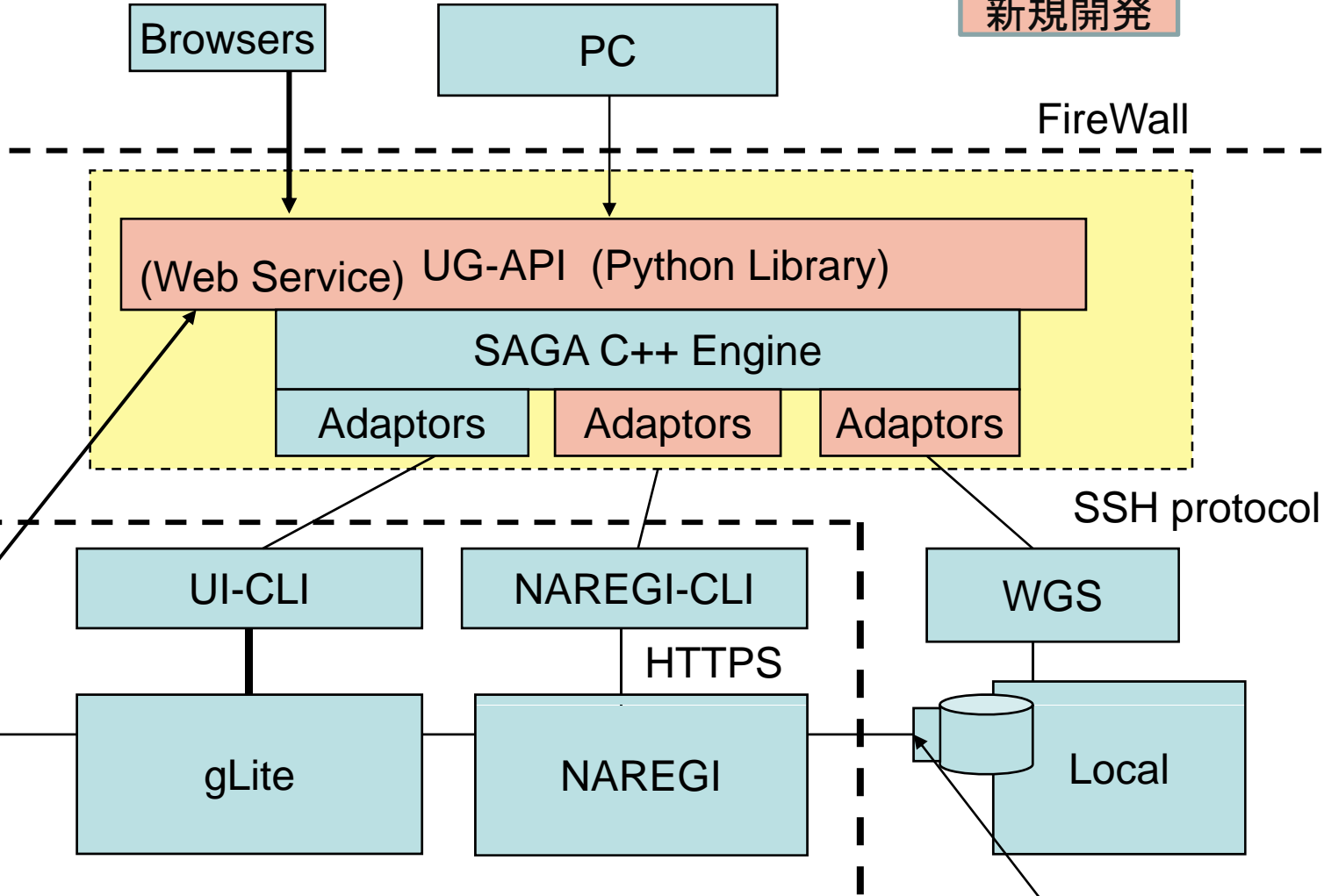
- {Executable = "/bin/sh"; Arguments = "hello.sh"; StdOutput = "std.out"; StdError = "std.err"; InputSandbox = {"hello.sh"}; OutputSandbox = {"std.out", "std.err"};



Architecture

既存
新規開発

End users
“Non-GRID”



2009/3/5

Fire Wall

DSI to
SRB, iRODS, Gfarm

7



SAGA

- Simple API for Grid Applications
 - OGFで標準化
 - GFD71;GFD90
- ローカルジョブスケジューラ、GRID、クラウドなど統一的に扱うことを目指す
- Job,file,



SAGAの例

ローカルジョブ

- 01: // Submitting a simple job and wait for completion
- 02: //
- 03: saga::job_description jobdef;
- 04: jobdef.set_attribute ("Executable", "job.sh");
- 05:
- 06: saga::job_service js; emo
- 07: saga::job job = js.create_job ("rte.host.net", jobdef);
- 08:
- 09: job.run();
- 10:
- 11: while(job.get_state() == saga::job::Running)
- 12: {
- 13: std::cout << "Job running with ID: "
- 14: << job.get_attribute("JobID") << std::endl;
- 15: sleep(1);
- 16: }

NAREGI

- 01: // Submitting a simple job and wait for completion
- 02: //
- 03: saga::job_description jobdef;
- 04: jobdef.set_attribute ("Executable", "job.sh");
- 05:
- 06: saga::job_service js;
- 07: saga::job job = js.create_job ("naregi::nrg-saga.host.net", jobdef);
- 08:
- 09: job.run();
- 10:
- 11: while(job.get_state() == saga::job::Running)
- 12: {
- 13: std::cout << "Job running with ID: "
- 14: << job.get_attribute("JobID") << std::endl;
- 15: sleep(1);
- 16: }



SAGAの例2

LSFの例

- 01: // Submitting a simple job and wait for completion
- 02: //
- 03: saga::job_description jobdef;
- 04: jobdef.set_attribute ("Executable", "job.sh");
- 05:
- 06: saga::job_service js;
- 07: saga::job job = js.create_job ("lsf:lsf.host.net", jobdef);
- 08:
- 09: job.run();
- 10:
- 11: while(job.get_state() == saga::job::Running)
- 12: {
- 13: std::cout << "Job running with ID: "
- 14: << job.get_attribute("JobID") << std::endl;
- 15: sleep(1);
- 16: }

PBSの例

- 01: // Submitting a simple job and wait for completion
- 02: //
- 03: saga::job_description jobdef;
- 04: jobdef.set_attribute ("Executable", "job.sh");
- 05:
- 06: saga::job_service js;
- 07: saga::job job = js.create_job ("pbs:pbs.host.net", jobdef);
- 08:
- 09: job.run();
- 10:
- 11: while(job.get_state() == saga::job::Running)
- 12: {
- 13: std::cout << "Job running with ID: "
- 14: << job.get_attribute("JobID") << std::endl;
- 15: sleep(1);
- 16: }



開発の現状

- Use-caseのサンプリング、要求要件分析
- SAGA-NAREGIアダプタ
 - 開発中
 - プロトタイプが3月中旬に完成
- SAGA-PBSアダプタ
 - 開発に着手



今後の計画

- Use-caseのサンプリング、要求要件分析
 - できるだけ多くの分野から
- NAREGI adaptor プロトタイプの評価
 - SAGAの仕様の精査
- gLite対応
- Pythonライブラリ