



Ricerca di Sistema elettrico

Realizzazione del Laboratorio Virtuale ONEBus “Optimal **NET**work of electric **BUS**es”

S. Migliori, G. Bracco, M. Mongelli, B. Calosso,
S. Pierattini, G. Ponti, M. Celino

REALIZZAZIONE DEL LABORATORIO VIRTUALE ONEBUS

S. Migliori, G. Bracco, M. Mongelli, B. Calosso, S. Pierattini, G. Ponti, M. Celino (ENEA)

Settembre 2018

Report Ricerca di Sistema Elettrico

Accordo di Programma Ministero dello Sviluppo Economico - ENEA

Piano Annuale di Realizzazione 2016

Area: "Efficienza energetica negli usi finali elettrici e risparmio energia negli usi finali elettrici ed interazione con altri vettori elettrici"

Progetto: "Mobilità elettrica sostenibile"

Obiettivo: "Scenari di mobilità elettrica", sub-obiettivo "Strumenti di supporto TPL"

Responsabile del Progetto: Maria Pia Valentini ENEA

Indice

SOMMARIO.....	4
INTRODUZIONE.....	5
1 IL LABORATORIO VIRTUALE ONEBUS.....	5
1.1 ACCESSO A ONEBUS E ALLE RELATIVE FACILITIES	6
1.2 AMBIENTE DI LAVORO COLLABORATIVO.....	7
1.3 PHONE CONFERENCE	9
1.4 ACCESSO ALL'INFRASTRUTTURA DI CALCOLO E ALL'AMBIENTE DI SVILUPPO	9
1.4.1 <i>ThinLinc</i>	10
1.4.2 <i>Accesso FARO</i>	11
1.4.3 <i>Accesso a server ONEBus windows</i>	12
1.4.4 <i>Archiviazione dati e software in AFS</i>	13
2 SVILUPPO DI UNA STRATEGIA PER UTILIZZO EFFICIENTE DI CRESCO	14
2.1 DESCRIZIONE DEL PROBLEMA	14
2.2 STRUTTURA DEL PROGRAMMA	14
2.3 METODOLOGIE: JOB ARRAY E C++	15
2.4 STRUTTURA DEL PROGRAMMA	16
3 ABBREVIAZIONI ED ACRONIMI.....	17
4 LISTA DELLE FIGURE.....	17

Sommario

L'obiettivo delle attività della Divisione DTE-ICT è di rendere possibile l'utilizzo dell'infrastruttura di supercalcolo ENEA e dei suoi servizi ICT per la realizzazione delle attività del progetto "Mobilità elettrica sostenibile". Il progetto prevede una specifica attività destinata alla realizzazione di strumenti di supporto per la valutazione della fattibilità della mobilità elettrica in un contesto di trasporto urbano. A tal fine è stato previsto lo sviluppo e l'implementazione di modelli numerici e codici di calcolo per l'analisi dei costi di elettrificazione di sottoreti di autobus.

L'utilizzo di strumenti informatici avanzati è fondamentale lungo tutto il percorso destinato a costruire una infrastruttura di calcolo di supporto alle decisioni. Lo sviluppo dei software, realizzati dai partner del progetto, sono di varia natura e hanno requisiti spesso molto diversi, sia in termini di memoria, di potenza di calcolo che di spazio per la conservazione dei dati. Ulteriore richiesta è quella che tutti i software sviluppati, componenti lo strumento di supporto alle decisioni, possano scambiarsi i dati, confrontarli ed eventualmente metterli a disposizione per procedure di visualizzazione avanzata. L'infrastruttura ICT dell'ENEA ha la possibilità di ospitare tutti i codici del progetto mettendo a disposizione accessi dedicati a server ritagliati sulle esigenze del singolo utente.

A tal fine è stato necessario realizzare una piattaforma web, denominata Laboratorio Virtuale ONEBus, per permettere ai partner del progetto di avere un unico punto di accesso all'infrastruttura di supercalcolo dell'ENEA e ai suoi servizi, disponibili all'interno della rete ENEAGRID dell'ENEA. Nel Laboratorio Virtuale, non solo si trovano informazioni su come utilizzare l'infrastruttura di calcolo ENEA ma permette anche attività di lavoro collaborativo, conservare e gestire grandi mole di informazioni e dati. I vantaggi derivanti dalla disponibilità di un Laboratorio Virtuale sono principalmente tre: 1) l'ambiente di lavoro è sempre aggiornato per tutti i codici, 2) tutti i documenti, i software e i dati sono disponibili sempre dallo stesso sito, 3) si ha accesso a servizi e potenze di calcolo che sulle risorse ICT locali non è possibile avere.

In particolare si sottolinea che i codici sviluppati durante il progetto da due partner universitari avevano bisogno di due ambienti di sviluppo diversi e distinti. Il software dell'Università Roma3 è stato sviluppato in ambiente windows mentre il codice dell'Università di Tor Vergata è stato sviluppato in ambiente Linux con la richiesta di utilizzare la grande potenza di calcolo messa a disposizione dal supercalcolatore CRESCO dell'ENEA.

Le attività dell'ultimo anno si sono svolte con la finalità di:

- 1) consolidare le funzionalità e l'accessibilità del Laboratorio Virtuale ONEBus,
- 2) di ottimizzare un ambiente di sviluppo windows con le caratteristiche che permettano l'implementazione e la realizzazione di simulazioni anche multi-tasking,
- 3) di ripensare la strategia di parallelizzazione e di svilupparne una nuova per il codice finalizzato alla ricerca esaustiva di sottoreti di reti complesse sfruttando l'architettura distribuita in ambiente Linux del supercomputer CRESCO,
- 4) assistenza e messa a disposizione di ore calcolo dedicate e storage per la realizzazione delle simulazioni su supercomputer e strumenti per lo scambio di dati tra le diverse risorse di calcolo.

Quindi le attività si sono concluse rendendo disponibile un ambiente di sviluppo e di lavoro integrato per i diversi partner di progetto che è inoltre in grado di ospitare i risultati di progetto anche dopo la conclusione dello stesso. L'intera infrastruttura è stata progettata in modo da poter seguire gli aggiornamenti tecnologici nel settore ICT e di poter crescere in parallelo ad eventuali nuove richieste provenienti da sviluppi progettuali futuri.

Introduzione

L'obiettivo delle attività della Divisione DTE-ICT è di rendere possibile l'utilizzo dell'infrastruttura di supercalcolo ENEA e dei suoi servizi ICT per la realizzazione delle attività del progetto "Mobilità elettrica sostenibile". Il progetto nel complesso è finalizzato a realizzare strumenti di supporto per la valutazione della fattibilità della mobilità elettrica in un contesto di trasporto urbano. A tal fine è stato previsto lo sviluppo e l'implementazione di modelli numerici e codici di calcolo per l'analisi dei costi di elettrificazione di sottoreti di autobus.

L'utilizzo di strumenti informatici avanzati è fondamentale lungo tutto il percorso destinato a costruire una infrastruttura di calcolo di supporto alle decisioni. I vantaggi derivanti dalla disponibilità di un Laboratorio Virtuale sono principalmente tre: 1) l'ambiente di lavoro è sempre aggiornato e identico per tutti i codici, 2) tutti i documenti, il software e i dati sono depositati e funzionanti nello stesso sito, 3) si ha accesso a servizi e potenze di calcolo che sulle risorse ICT locali non è possibile avere.

Per questi motivi è stato necessario realizzare una piattaforma web, denominata Laboratorio Virtuale ONEBus, per permettere ai partner del progetto di avere accesso all'infrastruttura di supercalcolo dell'ENEA e ai suoi servizi, disponibili all'interno della rete ENEAGRID dell'ENEA. Il Laboratorio Virtuale permette anche attività di lavoro collaborativo e di condividere informazioni e dati. Attività di formazione individuale e di gruppo, nonché l'affiancamento nello sviluppo dei codici, hanno permesso un veloce avvio delle attività nella nuova modalità condivisa.

Il presente rapporto descrive il Laboratorio Virtuale nel suo complesso, dalla disponibilità di servizi fino all'accesso via web ai codici di calcolo sviluppati durante il progetto. In particolare si sottolinea che i codici sviluppati durante il progetto da due partner universitari avevano bisogno di due ambienti di sviluppo diversi e distinti ma che potessero dialogare in termini di software e dati. Il software dell'Università Roma3 è stato sviluppato in ambiente Windows mentre il codice dell'Università di Tor Vergata è stato sviluppato in ambiente Linux con la richiesta di utilizzare la grande potenza di calcolo messa a disposizione dal supercalcolatore CRESCO dell'ENEA. Pertanto è stato necessario fornire una tipologia di supporto differente per i due partner di progetto. In particolare il supporto al partner di Tor Vergata ha implicato anche lo sviluppo di una strategia di parallelizzazione per poter accedere in maniera efficiente al supercalcolatore CRESCO.

1 Il Laboratorio Virtuale ONEBus

Uno degli obiettivi del progetto era quello di realizzare una piattaforma virtuale denominata ONEBus "Optimal NEtwork of electric BUSes": procedure di supporto alle decisioni nei processi di elettrificazione del servizio di Trasporto Pubblico Locale" accessibile da remoto, via web sviluppata per facilitare la collaborazione tra i partner che lavorano per la definizione delle procedure di supporto alle decisioni nei processi di elettrificazione del servizio di Trasporto Pubblico Locale nell'ambito degli Accordi di programma tra Ministero dello Sviluppo Economico e soggetti pubblici o a prevalente partecipazione pubblica per il Piano Triennale 2016-2018 della ricerca di sistema elettrico nazionale.

La piattaforma è stata realizzata mediante PLONE poiché consente una gestione completa dei contenuti CMS e di costruire piattaforme web permettendo una stretta collaborazione tra il manager del sistema informatico e chi produce i contenuti informativi. Infatti Plone, una volta che l'amministratore di sistema ha costruito lo scheletro del sito secondo le esigenze del progetto, permette anche agli utenti poco esperti di contribuire alla creazione dei contenuti rendendo così semplice l'aggiornamento continuo delle informazioni. Inoltre è possibile utilizzare l'ambiente di sviluppo Plone senza che sia necessario installare software sul proprio computer locale. In tal modo sia l'attività di sviluppo del sito che quella di inserimento dei contenuti può avvenire in maniera remota, utilizzando gli strumenti via web, da parte di tutti coloro che sono registrati sulla piattaforma che ospita il sito. Un sito web Plone contiene diversi tipi di contenuto, compresi testi, foto e immagini. Questi possono esistere in molte forme: documenti, notizie, eventi, video, file audio e tutti i tipi di

file e dati che possono essere caricati o creati su un sito web. I contenuti possono anche essere caricati dal proprio computer. In un sito Plone è possibile anche creare delle cartelle per raccogliere i contenuti e per definire una struttura di navigazione. La scelta di questa tecnologia è stata guidata dalla necessità di mettere in condizioni i partner di progetto di contribuire attivamente alla creazione di nuovi contenuti. In questo modo il Laboratorio Virtuale diventa un oggetto dinamico, ricco di informazioni e quindi punto di riferimento per un'area applicativa.

Grazie al laboratorio virtuale ONEBus, i partner del progetto e gli utenti interessati e abilitati, una volta registrati, possono accedere direttamente agli strumenti di lavoro, alle risorse hardware e software e ai servizi messi a disposizione dall'infrastruttura di calcolo ICT.

1.1 Accesso a ONEBus e alle relative facilities

La Home Page del laboratorio virtuale ONEBus è raggiungibile da qualsiasi browser digitando il seguente indirizzo:

<http://www.laboratorivirtuali.enea.it/prime-pagine/ONEBus>

La struttura, come è possibile vedere in **Errore. L'origine riferimento non è stata trovata.**, è costituita da un corpo centrale e alcuni portlet a sinistra e a destra che offrono l'accesso a servizi, tecnologie, risorse hardware e software .

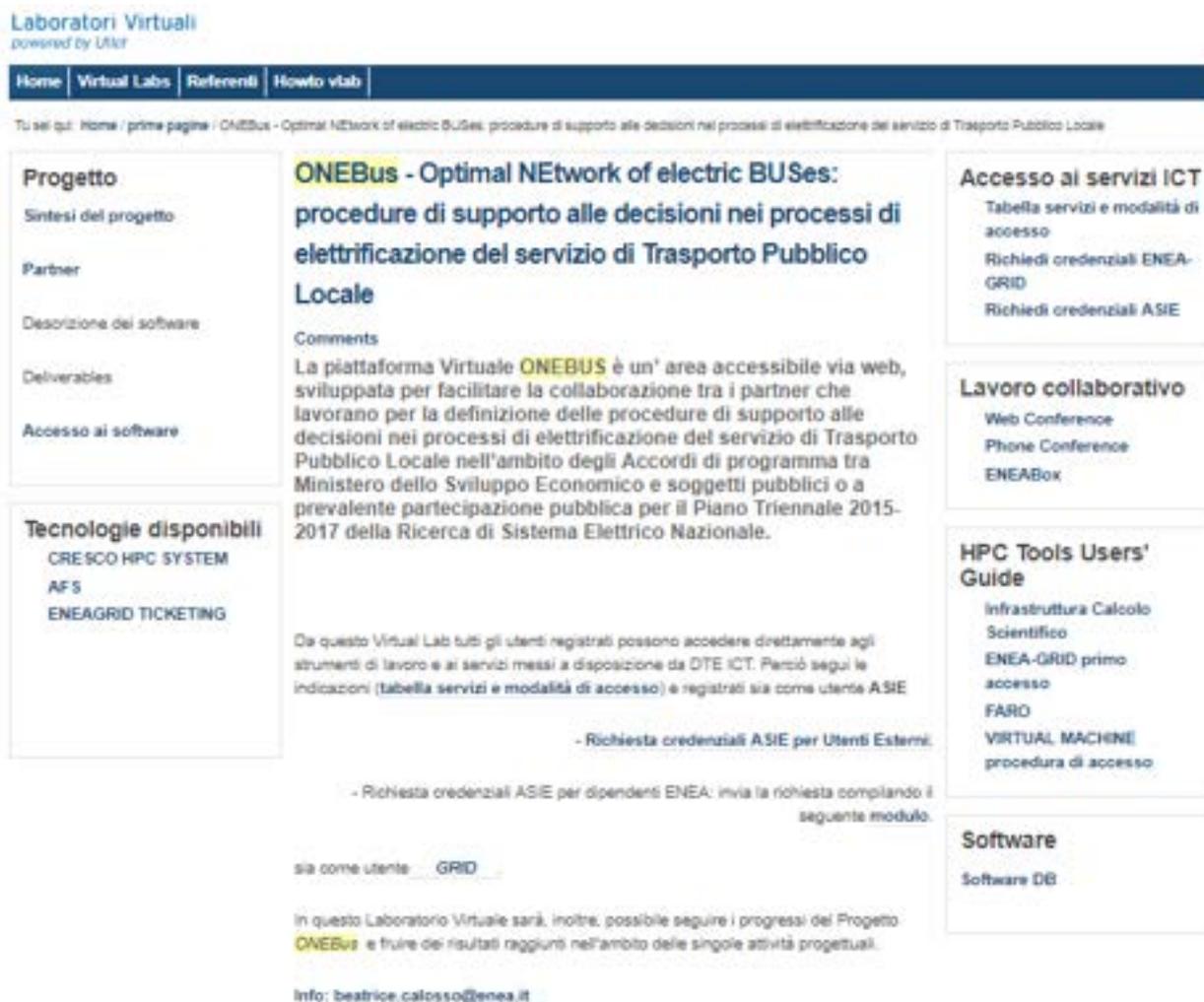


Figura 1: Homepage ONEBus costituito da un corpo centrale informativo e da portlet sui lati per l'accesso ai servizi e alle risorse ENEAGRID dedicate al progetto

Nel corpo centrale sono fornite informazioni generali su come utilizzare i servizi di un Laboratorio Virtuale e vengono forniti link ai siti di documentazione generale.

Sulla sinistra sono visibili due portlet. Il primo “Progetto” contiene i link alla pagina web di riferimento del progetto che viene utilizzata non per il lavoro collaborativo ma per la diffusione dei risultati e per conservare l’evoluzione del progetto in tutte le sue fasi. In questa area mano a mano che saranno disponibili i documenti finali del progetto e i deliverable delle attività potranno essere depositati

Sotto il portlet “Progetto”, il portlet “Tecnologie disponibili” permette di accedere alle sezioni di calcolo e ai software della catena modellistica disponibili in ENEAGRID .

In particolare gli utenti possono accedere a tali tecnologie e nello specifico ai software e alle sezioni AFS, dei server windows o dell’interfaccia FARO mediante accreditamento con autenticazione ASIE e/o ENEAGRID.

Sono state generate 9 utenze in tutto, 3 per ogni partner di progetto:

- ❖ 3 utenze per l’Università Tor Vergata
- ❖ 3 utenze per l’Università di Roma
- ❖ 3 utenze per i colleghi ENEA non DTE-ICT coinvolti

Sulla destra sono disponibili quattro portlet: “Accesso ai servizi ICT”, “Lavoro collaborativo”, “HPC tools users’ guides” e “Software”. “Accesso ai servizi ICT” è utilizzato per fornire una panoramica dei servizi disponibili e di quale tipo di autenticazione serve. Per questo sono subito disponibili i link alle procedure per la richiesta di nuove credenziali. Subito sotto, nel portlet “Lavoro collaborativo” sono inseriti i link alla web conference, phone conference and ENEABox modificati per le attività ONEBus. Nell’ultimo portlet “HPC tools users’ guides” si trovano le user’s guide di tutti gli strumenti messi a disposizione per il Laboratorio Virtuale.

1.2 Ambiente di Lavoro Collaborativo

Previa autenticazione con login e password ASIE, gli utenti accedono, dall’area intitolata “Lavoro collaborativo” del “portlet” a destra della home page del Laboratorio Virtuale, alle diverse facilities messe a punto per il progetto. La procedura per accreditarsi è visualizzabile sul “portlet” denominato “Accesso ai servizi ICT”.

Cliccando da “Lavoro collaborativo” (Figura 2) su “**web conference**” è disponibile una virtual room “<https://connect.portici.enea.it/onebus/>” dedicata mediante la quale gli utenti autenticati hanno a disposizione via web un’area di lavoro con tutti gli strumenti di una video conferenza (chat, microfoni, audio, registrazione, ecc.) e di una web-conference (condivisione di documenti, condivisione di una lavagna virtuale, controllo da remoto dello schermo dell’utente e diversificazione dei ruoli per la gestione dei permessi di utilizzo di tutti i suddetti strumenti) (Figura 3).



Figura 2: Home page della virtual room ONEBus in cui è possibile autenticarsi tramite accesso ASIE. E' possibile partecipare agli eventi nella virtual room anche come ospiti. In questo modo anche personale esterno al progetto ha la possibilità di partecipare alle riunioni e ai seminari formativi.



Figura 3: Interfaccia della virtual room ONEBus. La virtual room permette anche di registrare gli incontri e di renderli disponibili.

1.3 Phone Conference

Cliccando da “Lavoro collaborativo” su “Phone Conference” è possibile prenotare una “Conference Call” via telefono ed invitare gli utenti interessati a parteciparvi (Figura 4). Lo strumento è molto agile ed è possibile adattarlo alle esigenze del Laboratorio Virtuale. Spesso è il primo punto di contatto per successive riunioni nella virtual room.

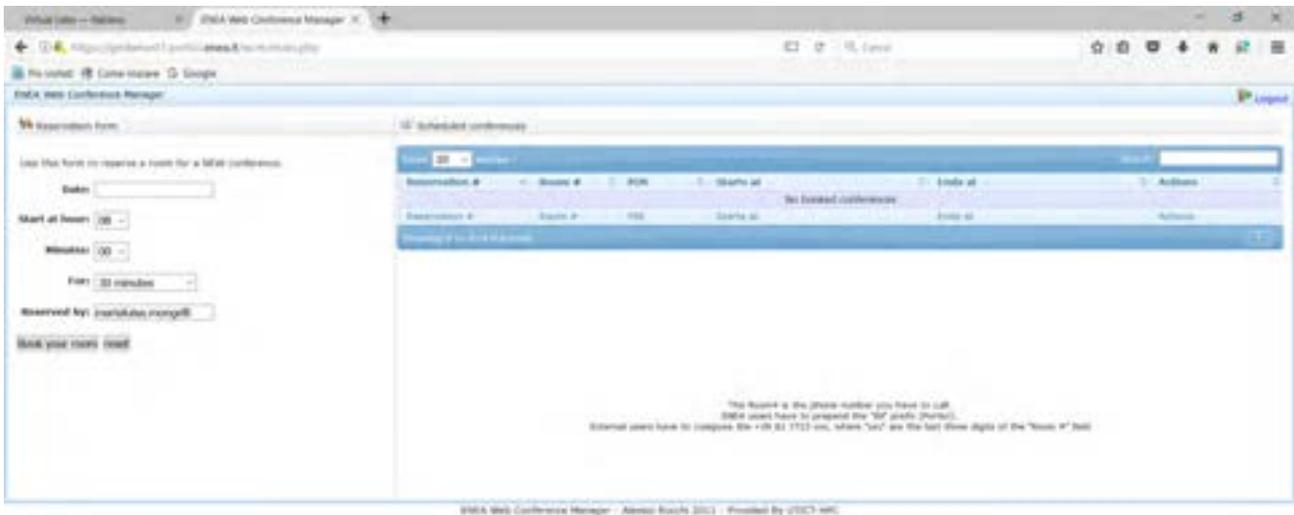


Figura 4: Interfaccia della “Phone Conference”

Cliccando da “Lavoro collaborativo” su “ENEA Box” è possibile accedere all’area di condivisione su owncloud per condividere file, documenti e reportistica relativa agli stati di avanzamento del progetto ecc. (Figura 5). La disponibilità di un’area di condivisione di progetto assicura la massima riservatezza e sicurezza della documentazione di progetto. I documenti sono conservati da ENEA e non da società terze esterne e di origine straniera. Inoltre, ogni utente ha accesso non solo all’area condivisa di progetto ma anche ad una propria.

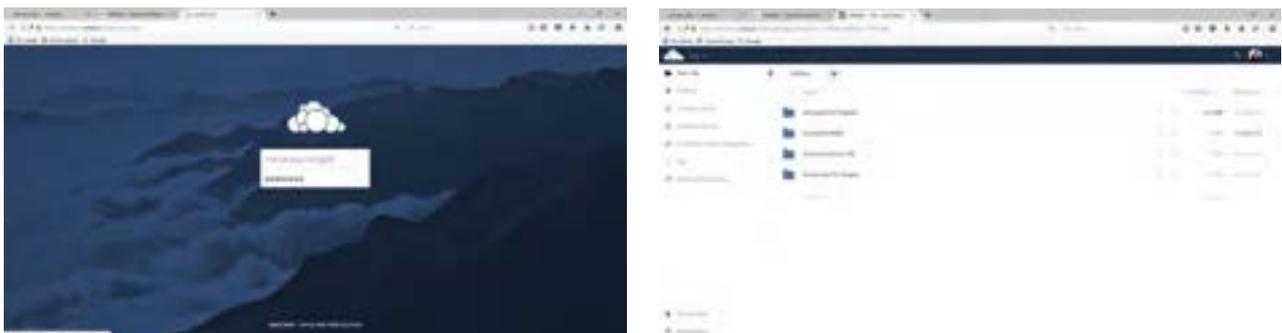


Figura 5: Condivisione dati su ENEABox – accesso (sinistra); interfaccia area di sharing (destra). Tutte le informazioni riguardanti il progetto si trovano su server sicuri e condivisi tra i partner. L’area progettuale è indipendente dalle aree personali dei singoli utenti ONEBus.

1.4 Accesso all’infrastruttura di calcolo e all’ambiente di sviluppo

Dal portlet “Tecnologie disponibili” si possono raggiungere le aree di calcolo dell’infrastruttura CRESCO. Il primo link conduce alla home page del sistema CRESCO dove sono riportati esempi e attività in corso sul sistema nel suo complesso. Per entrare nel sistema CRESCO sono necessarie le autenticazioni di tipo AFS. Per

questo motivo sono state create 9 accessi AFS per tutte le persone che avevano avuto l'autenticazione ASIE. A questo punto ogni utente ONEBus con le credenziali AFS può accedere al sistema CRESCO cliccando sul link "ONEBus Virtual PC". Nella pagine ThinLinc, come mostrato in Figura 6, viene richiesta l'autenticazione AFS. Una volta autenticati in ENEAGRID, si accede ad un menù, Vedi Figura 7, dove vengono offerte varie opzioni di accesso direttamente agli applicativi dei diversi Laboratori Virtuali.

Le opzioni predisposte per ONEBus sono di due tipologie: la prima è di utilizzare FARO2, descritto nel precedente rapporto, per accedere direttamente agli applicativi installati in ambiente Linux e che fanno riferimento ai dati in AFS e la seconda agli applicativi installati nel sistema windows.

1.4.1 ThinLinc

ThinLinc è un prodotto software che consente di implementare un'infrastruttura di calcolo basata su server. Centralizzando le risorse di elaborazione nel ata center anziché distribuirle tra gli utenti finali, i sistemi IT diventano più efficienti, sicuri e accessibili. ThinLinc, tramite un'interfaccia di tipo user-friendly, consente agli utenti finali, da remoto, dai propri PC, direttamente nelle loro aziende, l'accesso alla griglia ENEA, spostandosi in modo trasparente da una macchina di front-end all'altra. ThinLinc, installato su una singola macchina o in una configurazione cluster, fornisce accesso a pplikazioni, dati, risorse hardware e software utilizzando una configurazione client / server.

Le sessioni ThinLinc vengono pubblicate sul client utilizzando TigerVNC, un'implementazione ottimizzata del protocollo VNC. Ogni sessione riceve un server X dedicato sul lato server e sono supportati numerosi ambienti desktop. Tutto il traffico dati tra client e server è inviato via SSH criptato eliminando la necessità di ulteriori strati di sicurezza come la VPN. Inoltre ThinLinc è in grado automaticamente di bilanciare il carico riducendo ridondanze aumentando la scalabilità. ThinLinc può anche integrarsi con gli ambienti Windows, tramite il protocollo RDP. In questo modo, è possibile accedere interi desktop Windows e utilizzare applicazioni Windows su un desktop Linux.



Figura 6: Interfaccia ThinLinc per l'accesso in ENEAGRID tramite autenticazione AFS. L'accesso alle aree dedicate e alle risorse di ONEBus può avvenire in questo modo via web e da qualsiasi postazione remota.



Figura 7: Interfaccia ThinLinc dove sono elencate le risorse disponibili in ENEAGRID. Una connessione diretta è disponibile verso il server windows ONEBus

1.4.2 Accesso FARO

Richiamando l'interfaccia FARO2 dal menu ThinLinc, Figura 7, si apre la finestra FARO2 mostrata in Figura 8. L'interfaccia FARO2 permette di accedere direttamente alle risorse di calcolo tramite finestre xterm. Tramite un sistema a tendine sono elencate tutte le risorse di calcolo disponibili in ENEAGRID. Quindi, scelta la risorsa di calcolo e richiesta la finestra xterm, si è in grado di operare direttamente sul computer di riferimento richiamando con linea di comando le applicazioni.

In questa modalità è possibile lavorare utilizzando un singolo processore, in modalità seriale, oppure in modalità parallela utilizzando le diverse sezioni parallele del computer CRESCO.

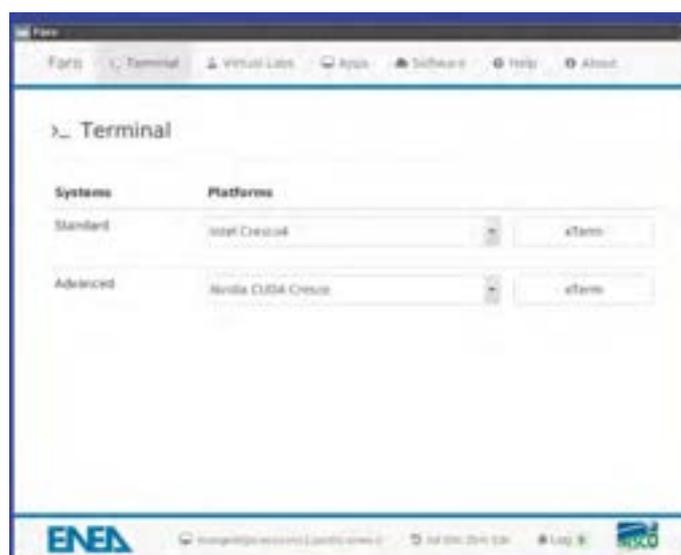


Figura 8: Interfaccia FARO 2.0: finestra per l'accesso a tutte le piattaforme di supercalcolo disponibili in ENEAGRID ed eventualmente anche esterne. Da questa finestra è possibile lanciare interfaccia xterm o entrare nel Virtual Lab dedicato dove si troveranno le applicazioni sviluppate in ONEBus.

1.4.3 Accesso a server ONEBus windows

E' stato predisposto un server windows versione 12, con 500 GB di disco dati e quattro core per le attività ONEBus (Figura 9). A tutto il personale impegnato nel progetto è stato dato accesso a questo server e unicamente alla persona incaricata di sviluppare e installare l'applicativo sono stati dati privilegi di amministratore. Questo server è raggiungibile tramite interfaccia ThinLinc, cliccando sulla voce "ONEBUS Project – Windows desktop". A questo punto viene chiesta la password ASIE per accedere al desktop del server windows dedicato a ONEBUS.

Come mostrato nella Figura 9 per il progetto ONEBUS è stata sviluppata una interfaccia con due finestre. La prima finestra serve per lanciare i codici di calcolo ONEBUS e BEST, sviluppati da ENEA e dall'Università Roma3.

Nella seconda finestra, denominata "Accesso ai dati" (vedi Figura 10) è possibile trovare i comandi necessari per avere accesso diretto ai dati disponibili in AFS e di norma utilizzati dagli utenti interessati a sfruttare il supercomputer CRESCO. In questa finestra sono disponibili i comandi per l'autenticazione AFS in ambiente windows. Quindi l'utente registrato a ONEBUS, sul server windows, ha contemporaneamente accesso ai dati disponibili sul proprio computer locale, sul server windows e in AFS. Questo permette di poter lanciare in entrambi gli ambienti i codici di calcolo per il confronto diretto dei risultati. Inoltre in questo modo tutti i tool windows e Linux sono disponibili nella stessa piattaforma per preparare gli input e analizzare gli output sullo stesso set di dati.

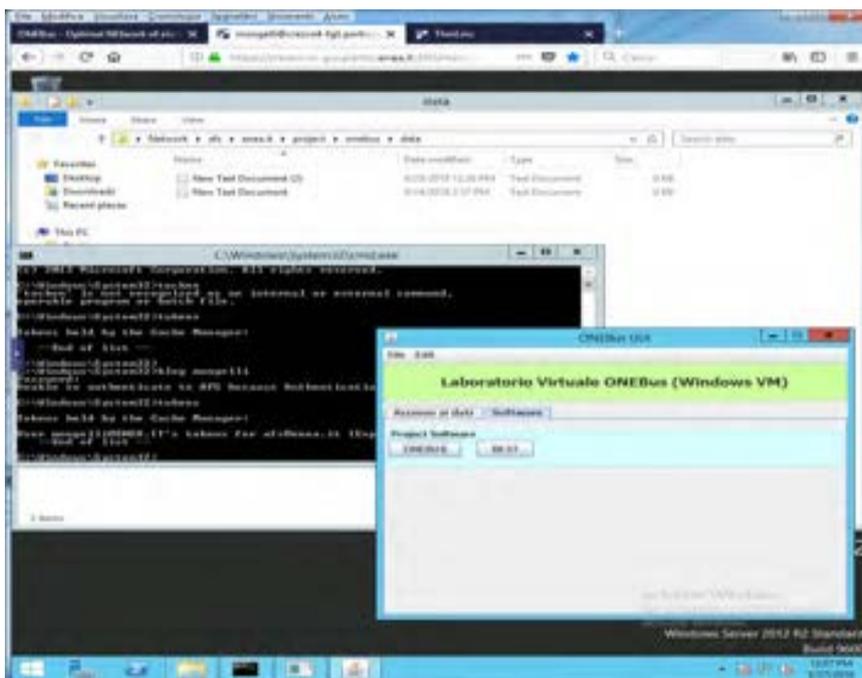


Figura 9: Interfaccia FARO 2.0 ambiente di lavoro per il laboratorio virtuale ONEBus. I codici sviluppati in ONEBus, sono stati installati e resi accessibili sulla virtual machine dedicata.

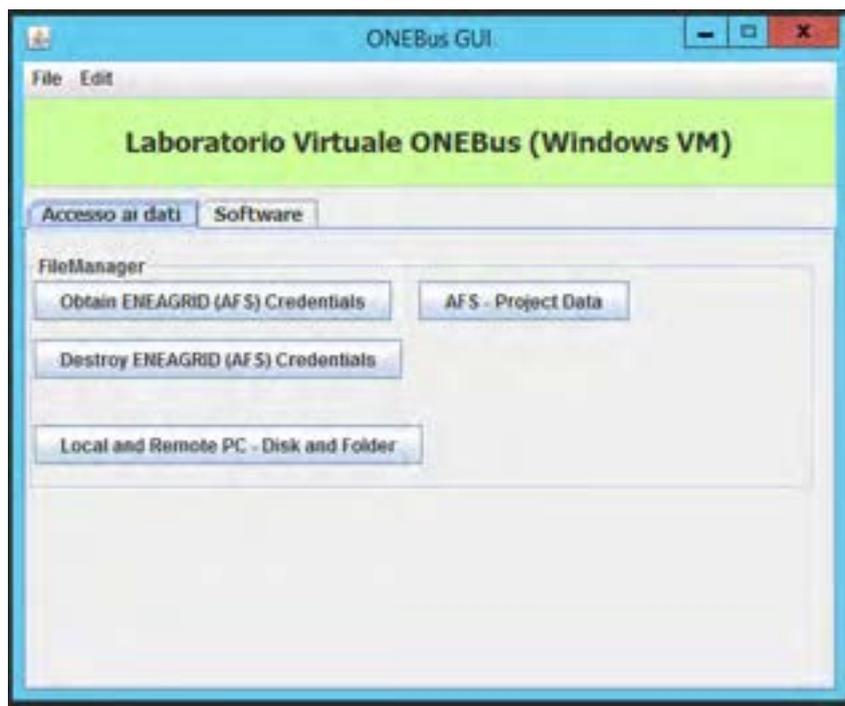


Figura 10: Interfaccia FARO 2.0 per l'autenticazione AFS in ambiente windows e avere l'accesso ai dati disponibili in ENEAGRID.

1.4.4 Archiviazione dati e software in AFS

In ambiente AFS è stato predisposto un area disco progettuale con grande capacità di memoria. Questa area è predisposta prevalentemente per lo storage di grandi quantità di dati che provengono dall'esecuzione di applicazioni in parallelo ma può essere utilizzata anche per la condivisione di informazioni. L'area dati in AFS consta di quattro sottoaree: eseguibili, software, dati, documenti,:

[/afs/.enea.it/project/onebus/bin](#)

[/afs/.enea.it/project/onebus/src](#)

[/afs/.enea.it/project/onebus/data](#)

[/afs/.enea.it/project/onebus/doc](#)

La prima contiene tutti gli eseguibili che sono richiamabili dal generico utente che accede all'infrastruttura tramite FARO2 o in modo diretto via xterm. La sottoarea dati contiene sia i dati di input provenienti da database esterni o da elaborazioni dei partner che i dati di output delle applicazioni parallele. Infine la sottoarea documenti e software contengono i manuali d'uso delle applicazioni e i sorgenti delle applicazioni sviluppate nel progetto.

Le prime due sottoaree nelle directory [/.enea.it/](#) sono utilizzate per lo sviluppo del codice e il testing. Quando il codice è pronto queste aree possono essere rilasciate, diventando [/enea.it/](#) per il suo utilizzo da parte degli utenti registrati al servizio onebus.

In questa area dedicata si possono trovare i codici sorgenti del codice sviluppato e una sua installazione. E' stato inoltre predisposto un caso test sulla città di Roma per la verifica immediata del suo funzionamento. Informazioni dettagliate sulla procedura per lanciare il run di test sul supercomputer CRESCO sono disponibili in un file README disponibile in questa area progettuale.

2 Sviluppo di una strategia per utilizzo efficiente di CRESCO

E' proseguita nel presente anno l'attività di affiancamento all'Università Tor Vergata per lo sviluppo di un codice altamente parallelo in grado di utilizzare in maniera efficiente l'infrastruttura di supercalcolo CRESCO dell'ENEA. L'attività di affiancamento è stata necessaria visto la complessità del problema che si doveva affrontare. Il problema consisteva nel trovare un insieme di soluzioni ottime calcolando tutte le possibili soluzioni in maniera esaustiva. Nell'annualità precedente era stata sviluppata una strategia di parallelizzazione e quindi una prima versione del codice che però ha mostrato alcuni limiti che sono descritti nel seguito. Pertanto in questo ultimo anno si è proceduto ad una riformulazione della strategia di parallelizzazione più adatta alla complessità del problema che si doveva affrontare. Quindi si è proposto e testato l'utilizzo di strumenti avanzati di calcolo parallelo che hanno permesso la suddivisione del problema complessivo in tanti task più piccoli ma più facilmente gestibili dal sistema parallelo per sfruttare in pieno le potenzialità del supercomputer ENEA, come descritto nelle sezioni successive.

2.1 Descrizione del problema

Il codice sviluppato nella annualità precedente è stato verificato e benchmarkato utilizzando le risorse di calcolo di due cluster dell'infrastruttura CRESCO: Cresco3 e Cresco4. Il codice era stato sviluppato in linguaggio C++ sfruttando il protocollo di comunicazione tra processi chiamato MPI (Message Passing Interface). Era stato utilizzato il linguaggio C++ piuttosto che altri per avere garanzie sulle prestazioni complessive del codice: il linguaggio C++ offre infatti un mezzo di astrazione che non produce overhead delle prestazioni a runtime rendendo possibile scrivere un codice efficiente sebbene con un alto livello di astrazione.

Il programma pensato per essere eseguito su molteplici nodi del cluster necessita di un protocollo di comunicazione tra i computer. La scelta in questo caso è ricaduta su MPI perché è considerato lo standard per la comunicazione tra nodi appartenenti ad un cluster di computer che eseguono un programma parallelo sviluppato per sistemi a memoria distribuita. Le librerie di Message Passing Interface sono un protocollo condiviso da più di 40 organizzazioni, tra cui vendor, ricercatori, sviluppatori di librerie software e utenti. Tuttavia, MPI non è uno standard IEEE o ISO, ma è diventato di fatto uno standard delle industrie per poter scrivere programmi che hanno bisogno di scambiare messaggi su piattaforme di HPC. L'obiettivo principale di MPI è quello di stabilire uno standard flessibile, efficiente e portabile. Infatti, MPI rispetto alle precedenti librerie utilizzate per il passaggio di parametri tra nodi, ha il vantaggio di essere molto portabile (è stata implementata per moltissime architetture parallele) e veloce (è stata ottimizzata per ogni architettura).

2.2 Struttura del programma

La struttura del programma che era stata definita nella prima annualità è costituita da quattro fasi principali:

- 1) Inizializzazione dell'ambiente MPI, delle variabili e degli oggetti necessari per l'analisi della rete.
- 2) Il bilanciamento del carico, effettuato dal nodo master, per comunicare a tutti i processori il carico di lavoro da svolgere.
- 3) L'esplorazione della porzione di rete che è stato deciso dall'algoritmo di load balancing.
- 4) Il merge dei risultati che sono stati calcolati dai diversi processori.

L'analisi di questo approccio ha mostrato come questa strategia di parallelizzazione ha diversi punti deboli:

- Mancanza di salvataggio di stato: se i diversi processori non riescono a terminare l'analisi delle linee di autobus a loro assegnata prima del termine massimo assegnato loro dalla queue, il calcolo effettuato viene perso.
- Complessità nel bilanciamento del carico: il bilanciamento del carico è uno dei problemi principali in questo tipo di reti. Sono stati implementati ben quattro diverse versioni di bilanciamento che hanno richiesto sforzo mentale e impegno nella scrittura del codice. Nonostante ciò, l'utilizzo

delle risorse non è risultato ottimale, seppur si sia raggiunto un bilanciamento del carico accettabile.

- Errori di comunicazione tra master e slaves: la comunicazione tra master e slaves gestita tramite MPI introduce latenze e alcune volte anche errori che rischiano di bloccare l'esecuzione del programma.
- Scrittura del codice complessa: l'utilizzo di MPI per far comunicare master e slaves e la scrittura di algoritmi di bilanciamento del carico hanno implicato la scrittura di un codice complesso e difficile da leggere.

Questa analisi ha messo in luce problemi sulla correttezza dei dati e sull'efficienza di parallelizzazione. In particolare il problema di analizzare un insieme di linee comporta l'analisi di ogni linea con tutte le altre escludendo le ripetizioni. Questo comporta un forte sbilanciamento intrinseco quando il carico di lavoro viene distribuito tra i diversi processori di calcolo. Inoltre si è potuto constatare che il carico di lavoro per ogni linea è contenuto, è lo stesso per ogni linea ed è indipendente da ogni altro calcolo sulle altre linee. Queste considerazioni hanno suggerito di modificare nell'ultimo anno di attività la strategia di parallelizzazione ed utilizzare l'approccio basato sui Job Array che verranno descritti nella prossima sezione.

2.3 Metodologie: Job Array e C++

La nuova versione del programma è stata realizzata in linguaggio C++ ma è stato eliminato completamente l'utilizzo di MPI per far fronte all'introduzione di una metodologia di parallelizzazione chiamata Job Array. Il sistema di code sul supercomputer CRESCO si basa sul software LSF (Load sharing facilities). Uno dei pregi di LSF è la possibilità di sottomettere programmi multipli in un unico run, usando la tecnologia del Job Array, ossia un insieme di oggetti dello stesso tipo, con lo stesso nome, identificati da un indice, ma che assumono valori diversi in input per produrre diversi output. Questo approccio è particolarmente utile se i job dell'array condividono lo stesso eseguibile (ad es. uno script o un'applicazione) e tipo di risorsa, ma hanno differenti file di input oppure l'eseguibile può gestire una variabile associata ad un indice.

I job array sono gruppi di job con lo stesso eseguibile e gli stessi requisiti di risorse, ma differenti file di input. I job array possono essere sottomessi, controllati e monitorati come una singola unità o come job individuali o gruppi di job. Infatti, ogni job sottomesso da un job array condivide lo stesso job ID in quanto fa parte dello stesso job array, e al tempo stesso è unicamente referenziato grazie all'uso dell'indice dell'array. La dimensione e la struttura di un job array è definita quando il job array viene creato.

Nel caso dei job array, LSF schedula e distribuisce i jobs dell'array indipendentemente l'uno dall'altro, in quanto pur condividendo l'ID ognuno di essi è identificato univocamente da un indice. I principali vantaggi dell'utilizzo dei job array sono:

- Non si deve gestire la comunicazione tra i diversi nodi del cluster: infatti, i nodi non hanno bisogno di comunicare tra di loro, ogni job è gestito da un nodo in maniera separata dal resto dei processori del cluster. Solo una volta terminata l'intera esecuzione dei job presenti nel job array si può controllare lo stato dell'esecuzione. Ma anche in questo caso non vi è comunicazione tra diversi nodi del cluster.
- Non si devono scrivere algoritmi per il bilanciamento del carico: il bilanciamento del carico viene fatto sulla base dei file di input che diamo in input ad ogni singolo job. Ma la gestione dei nodi e la comunicazione viene affidata al sistema LSF.
- Scrittura del codice semplificata: con l'utilizzo dei job array si devono scrivere dei semplici programmi che utilizzano C++, senza dover utilizzare MPI o scrivere algoritmi complessi per il bilanciamento del carico.
- Si rende necessario il salvataggio dello stato del lavoro: tra uno step e l'altro del programma si ha la necessità di salvare il lavoro fatto nello step precedente per poterlo recuperare e analizzare nello step successivo.

2.4 Struttura del programma

La struttura del nuovo programma è ripartita in 3 step.

Primo step

Nel primo step vengono eseguiti i seguenti passi:

- a. Controllo dei parametri di input scelti dall'utente: si analizza se tali parametri sono corretti e consistenti, anche per gli step futuri.
- b. Creazione delle cartelle necessarie per l'esecuzione di tutti gli step successivi.
- c. Creazione di molteplici file in cui vengono definiti porzioni dello spazio delle soluzioni da esplorare. In ogni file ci sarà un insieme di combinazioni di linee che devono essere analizzate.
- d. Creazione di un file di configurazione che conterrà le informazioni base, che riguardano l'esecuzione corrente.
- e. Creazione di un file che contiene la lista degli indici dei file di input che si devono ancora esplorare. Alla prima iterazione tale file conterrà tutti gli indici dei file che sono stati creati al punto c.

Secondo step

Nel secondo step vengono eseguiti i seguenti passi:

- a. Parser dei parametri passati tramite la riga di comando ed il file di configurazione.
- b. Controllo se l'indice del job analizzato è all'interno dell'intervallo della grandezza del job array.
- c. Lettura del file dal quale si possono ricavare gli indici dei file che contengono le porzioni di rete ancora da esplorare. Questo file viene preparato dallo step 1, per la prima esecuzione dello step 2. Invece, questo file è preparato dallo step 3, se ci saranno altre esecuzioni dello step 2.
- d. Si calcola l'intervallo dei file che deve analizzare il job index. Per fare questo si ha bisogno di conoscere il numero dei file ancora da analizzare e la grandezza del job array.
- e. Per ogni file con la porzione dello spazio delle soluzioni, si esplorano le soluzioni, e poi si crea il file con le migliori sottoreti.
- f. Salvare in un file (uno per ogni job nell'array) l'indice dei file che sono stati processati interamente con successo

Terzo step

Nel terzo step vengono eseguiti i seguenti passi:

- a. Parser dei parametri passati tramite la riga di comando ed il file di configurazione.
- b. Controllo se ci sono alcuni file dello spazio delle soluzioni che ancora non sono stati analizzati.
- c. Mantenimento di un valore intero per il numero di file ancora da analizzare
 - a. Se tale numero è 0 significa che tutto lo spazio delle soluzioni della rete è stato analizzato
 - i. Dunque, si procede a calcolare la migliore soluzione tra quelle prodotte dai diversi job dell'array.
 - b. Altrimenti, se il numero è maggiore di 0, significa che ci sono ancora dei file da analizzare.
 - i. Si sceglie la grandezza del job array, in base a quanti file sono rimasti da analizzare.
 - ii. Si aggiorna il file indexes.txt con gli indici dei file da analizzare durante la successiva esecuzione dello step 2.

3 Abbreviazioni ed acronimi

DTE: Dipartimento Tecnologie Energetiche

ICT: Information and Communication Technology

HPC: High Performance Computing

TPL: Trasporto Pubblico Locale

AFS: Advanced File System

GPFS: General Parallel File System

ASIE: Accesso Servizi Informatici ENEA

FARO: Fast Access to Remote Objects

CRESCO: Centro computazionale di RicErca sui Sistemi Complessi

GUI: Graphical User Interface

MPI: Message Passing Interface

LSF: Load Sharing Facility

4 Lista delle Figure

Figura 1: Homepage ONEBus costituito da un corpo centrale informativo e da portlet sui lati per l'accesso ai servizi e alle risorse ENEAGRID dedicate al progetto

Figura 2: Home page della virtual room ONEBus in cui è possibile autenticarsi tramite accesso ASIE. E' possibile partecipare agli eventi nella virtual room anche come ospiti. In questo modo anche personale esterno al progetto ha la possibilità di partecipare alle riunioni e ai seminari formativi.

Figura 3: Interfaccia della virtual room ONEBus. La virtual room permette anche di registrare gli incontri e di renderli disponibili.

Figura 4: Interfaccia della "Phone Conference"

Figura 5: Condivisione dati su ENEABox – accesso (sinistra); interfaccia area di sharing (destra). Tutte le informazioni riguardanti il progetto si trovano su server sicuri e condivisi tra i partner. L'area progettuale è indipendente dalle aree personali dei singoli utenti ONEBus.

Figura 6: Interfaccia ThinLinc per l'accesso in ENEAGRID tramite autenticazione AFS. L'accesso alle aree dedicate e alle risorse di ONEBus può avvenire in questo modo via web e da qualsiasi postazione remota.

Figura 7: Interfaccia ThinLinc dove sono elencate le risorse disponibili in ENEAGRID. Una connessione diretta è disponibile verso il server windows ONEBus

Figura 8: Interfaccia FARO 2.0: finestra per l'accesso a tutte le piattaforme di supercalcolo disponibili in ENEAGRID ed eventualmente anche esterne. Da questa finestra è possibile lanciare interfaccia xterm o entrare nel Virtual Lab dedicato dove si troveranno le applicazioni sviluppate in ONEBus.

Figura 9: Interfaccia FARO 2.0 ambiente di lavoro per il laboratorio virtuale ONEBus. I codici sviluppati in ONEBus, sono stati installati e resi accessibili sulla virtual machine dedicata.

Figura 10: Interfaccia FARO 2.0 per l'autenticazione AFS in ambiente windows e avere l'accesso ai dati disponibili in ENEAGRID.