

# Modelagem e Implementação de um Ambiente Computacional para Pesquisa, Aplicações e Aprendizagem de Métodos Numéricos

Cleber Paiva de Souza  
Universidade de São Paulo (USP)  
cleber@labrisco.usp.br

Mardel de Conti  
Universidade de São Paulo (USP)  
mbdconti@usp.br

## Abstract

This paper presents the concept design and describes the implementation and use of a computational environment for research, application and learning of numerical methods in Engineering, specifically in Naval Architecture and Ocean Engineering. Some characteristics of this environment are:

1. User-friendly interaction through a web root-interface and various branches, which enable functionalities such intercommunication among users via forum, chat, e-mail; collaborative knowledge construction via *wiki*; remote processing of technical and administrative tools; shared databases' construction with incremental information based on continuous following of processed tools' input and output; e-learning and knowledge management; project management; access control; safety and security management; geometry flow visualization.
2. Remote connection through X-interface secure shell (ssh), enabling access to all the tools installed at the laboratory, including site construction and management, structured languages compilation, file management and others.
3. Adoption, as extensively as possible, of FLOSS (free, libre, *open source software*) concepts, products and practice.
4. Multi-institutional participation.
5. Hide of complexities of the tools and functions accessed by the users; stratified user's ability to interact with the tools; stratified restrictions on processing.

## 1. Introdução

O trabalho apresenta aspectos da concepção, da implantação e do uso de ambiente computacional para pesquisa, aplicações e aprendizagem de métodos numéricos em engenharia.

Nas seções adiante, descrevem-se os requisitos e restrições para o ambiente computacional almejado, a configuração resultante para o ambiente, com base em tecnologias disponíveis na atualidade, a implementação do ambiente e considerações finais.

Algumas das características que resultaram para o ambiente, na concepção de interação amigável por meio de uma interface web, são: a) intercomunicação entre usuários por fóruns, *chats*, e-mail; b) construção de conhecimento colaborativo por *wikis*; c) processamento remoto de ferramentas computacionais técnicas e administrativas; d) construção de bancos de dados compartilhados com informação incremental baseada em processamento das

ferramentas aplicativas; e) aprendizagem e gestão de conhecimento por meio de ferramentas computacionais; f) gestão de projeto; g) controle de acesso; h) gestão de segurança.

## **2. Requisitos e restrições para o sistema computacional do laboratório de pesquisa**

O ambiente dá suporte ao grupo de pesquisa denominado LabNUMERAL - *Laboratory for Numerical Methods in Engineering: Research, Application and Learning*, sediado na Escola Politécnica da USP, cuja atuação concerne métodos numéricos em engenharia, especialmente Dinâmica de Fluidos Computacional (CFD - *Computer Fluid Dynamics*).

Um exemplo da produção do grupo é um conjunto de procedimentos e *software* para determinação da resistência de ondas de navios de deslocamento ao avanço (ver de Conti & Domiciano, 2000 [\[1\]](#); Domiciano, 2002 [\[2\]](#); Domiciano e de Conti, 2006 [\[3\]](#)). Utilizou-se o método de fontes de Rankine, com distribuição contínua parabólica de sua intensidade segundo bi-B-Splines.

Outro exemplo é um método e "*software*" para representação da superfície do casco, por meio de produto tensorial de curvas B-Spline, em que se faz uma aproximação da superfície com base em pontos de tabela de cotas por método dos mínimos quadrados (de Conti et al, 2003 [\[4\]](#)).

Presentemente, o grupo trabalha com método RANS (*Reynolds Averaged Navier Stokes Method*), em que efeitos turbulentos são levados em consideração em termos médios. Uma etapa importante deste método é a geração de malhas no domínio fluido. O grupo optou por geração de malhas estruturadas por transformações transfinitas e elípticas.

Paralelamente, o grupo conduz experimentos por meio da técnica PIV (*Particles Imaging Velocimetry*), em parceria com o IPT (Instituto de Pesquisas Tecnológicas de São Paulo). Parte do *software* (de Conti et al, 2008 [\[5\]](#) e Domiciano et al, 2008 [\[6\]](#)) desenvolvido pelo grupo é registrada no INPI (Instituto Nacional de Propriedade Intelectual).

No entanto, o grupo pretende abrir parte de sua produção computacional sob licença de *software* livre, permitindo a participação de parceiros externos na sua validação, na sua expansão e no seu aprimoramento. O grupo também tem a intenção colaborar com grupos de *software* CFD livre. Por meio dos professores vinculados, o grupo atua em formação acadêmica, seja na ministração de disciplinas em níveis de graduação, pós-graduação e especialização, seja pela orientação de iniciação científica, mestrado ou doutorado. Em disciplinas, tem-se dado ênfase à construção colaborativa de conhecimento. Uma característica dessa última forma de atuação é a possibilidade de individualização da

aprendizagem, tanto no sentido da "educação centrada no aprendiz" (Rogers) quanto no sentido do "construtivismo" (Piaget).

A construção colaborativa de conhecimento científico tem-se dado basicamente nas formas: a) dissertativa, referente a modelos físico-matemáticos, a métodos numérico-computacionais, à aplicação desses métodos, a procedimentos e análises experimentais, a estudos de casos, a métodos de projeto e sua aplicação; b) cálculos numéricos; c) ensaios experimentais; e d) desenvolvimento, implantação, aplicação e validação de *software*.

Para tanto, ambientes físico e computacional adequados devem ser configurados e implementados.

No ambiente físico local deve haver: a) certo resguardo do grupo de pesquisa com relação ao ambiente externo; b) provimento, em níveis usuais, da segurança, da privacidade e do conforto; c) possibilidade de trabalho individual de cada pesquisador em diferentes máquinas, não ficando determinado *hardware* vinculado a determinado pesquisador; d) favorecimento da interatividade entre os membros do grupo, sem prejuízo da possibilidade de condução individual de certos trabalhos; e) acesso a material bibliográfico de utilização frequente, na forma física ou informatizada; f) possibilidade de realização de cursos com pequenas turmas; g) possibilidade de realização de reuniões de pequenos grupos; eventualmente tal recurso e o anterior podem ser compartilhados com outros grupos de pesquisa; e h) possibilidade de realização de ensaios experimentais, em ambientes laboratoriais compartilhados com outros grupos de pesquisa, na própria instituição ou em instituições parceiras; os equipamentos e modelos físicos de uso específico do grupo devem ter a possibilidade de serem alojados em espaços com segurança e condições adequadas para garantir sua integridade.

Quanto ao ambiente computacional: a) cada pesquisador deve dispor de um ambiente computacional pessoal, não vinculado a máquinas específicas, com acesso restrito, local ou remoto, destinado exclusivamente à sua atuação na pesquisa, com regras de utilização especificadas por termo de uso; b) O ambiente computacional pessoal de cada pesquisador deve prover o acesso a ferramentas computacionais disponibilizadas pelo laboratório aos seus membros, o uso de quotas individuais de armazenamento, e o acesso à rede externa; a capacidade, a segurança e a confiabilidade dos recursos devem ser adequadas à realização das tarefas da pesquisa; e c) Subgrupos de pesquisadores envolvidos em tarefas comuns devem ter a possibilidade de compartilhamento de ferramentas e de espaços de armazenamento, com regras e instrumentos para racionalização, integração e consolidação de suas contribuições, e com manutenção de histórico de versões.

### 3. Configuração e implementação do ambiente computacional

A adoção de uma solução computacional para suprir as necessidades de um grupo de pesquisa requer a análise e avaliação de um conjunto extenso de tecnologias e ferramentas de *software* que as implementam. A composição da solução final deve considerar as limitações e conflitos tecnológicos impostos pelas diversas opções, o que exige o correlacionamento dos itens para a identificação dos pontos conflitantes e diminuir a possibilidade de escolhas com base nas decisões.

No processo de decisão, além da avaliação das funcionalidades da solução são considerados outros aspectos como o modelo de licenciamento, linguagem de programação, interface de usuário, documentação e atividade da comunidade de desenvolvimento. A lista de considerações tidas como importantes para a etapa de decisão das soluções incluem: 1) adoção, sempre que possível, de FLOSS (free, libre, open source software); 2) segurança; 3) utilização colaborativa e multi-institucional; 4) gerenciamento centralizado de usuário, incluindo: a) controle de acesso lógico estratificado e b) controle de processamento estratificado; 5) gerenciamento e monitoramento do ambiente.

Para a utilização e coordenação dos aspectos tecnológicos implementados, faz-se necessário a existência de documentos formais ou semi-formais que definam como cada item pode ser utilizado, suas iterações e as regras que regem aspectos tecnológicos e de propriedade intelectual. Neste sentido, os seguintes requisitos administrativos foram considerados: a) controle de acesso físico; b) políticas de utilização e termos de uso; c) padronização; d) controle de versão do *software*; e) documentação.

Alguns itens acima são descritos com maior detalhamento nas seções adiante.

#### 3.1. A adoção do conceito FLOSS

Na sigla FLOSS, o termo *free* normalmente se refere à condição do *software* não ser comercializado, estando disponível livre de pagamentos, o termo *libre* se refere à condição do acesso não ser restrito, e a expressão *open source* se refere à condição de o código fonte ser explícito, aberto. Os tipos de licença variam, existindo diferentes graus de permissão de uso e de possibilidades de modificações, incluindo modelos duais onde são utilizados dois modelos de licenciamento para um mesmo código com base no público utilizador <sup>[7]</sup>.

Há prós e contras de se adotar o conceito de *software* aberto; não há como apontar uma tendência universal mais favorável; trata-se mais de uma opção em cada caso. No laboratório em pauta, adota-se, quando possível, o conceito FLOSS.

Veelo (2005 [\[8\]](#); 2009 [\[9\]](#)) faz uma compilação e avaliação das possibilidades de uso de *software* aberto no campo da engenharia naval. Classes de ferramentas abertas citadas por Veelo incluem: modelagem geométrica, visualização, cálculo numérico fundamental (álgebra), método dos elementos finitos, CFD (*Computer Fluid Dynamics*), GUI (*Graphics User Interface*). São usualmente vistas como vantagens, as seguintes características de *software* aberto:

- Possibilidade de revisões e expansões por usuários que de fato têm interesse no tema (*peer review*).
- Criação de oportunidade de interação com pesquisadores de outras instituições que desenvolvem o tema, para além da construção do próprio *software*, pois essa construção exige um entendimento de base do problema e da forma de encaminhamento da resolução.
- Possibilidade de uso de bibliotecas de blocos de programas de fonte aberta já desenvolvidos e testados, sejam de caráter mais fundamental, como por exemplo, relativos a funções matemáticas, interações gráficas, cálculo de escoamento (*Computer Fluid Dynamics*), etc, sejam de caráter mais específico, relativos à geometria, à dinâmica, etc.
- Oportunidade de se trabalhar nas aplicações de fato, em sua modelagem e análise, onde as intervenções mais individualizadas do grupo de pesquisa se fazem mais marcantes, em contraposição - quando se utiliza *software* fechado - à busca de enquadramento do problema em modelos previamente implantados, sem clareza dos detalhes desta implantação.

Uma reconhecida desvantagem é a perda relativa de domínio sobre a ferramenta desenvolvida, com implicações no auferimento de recursos decorrentes de seu desenvolvimento, se bem que a expectativa de ganhos em nichos de mercado relativamente limitados, como é o caso da engenharia naval e oceânica, pode ser facilmente frustrada.

### **3.2. Segurança**

Segurança (*security*) é um atributo que está associado a perigos, como: a) perda de informação; b) acesso indevido à informação; e c) alteração indesejada da informação, seja por manipulações equivocadas por usuários autorizados, ou por manipulações fraudulentas.

O conceito de risco engloba a probabilidade de ocorrência de algum perigo, com uma medida de importância de suas consequências. Quanto à ocorrência, pode-se considerar os seguintes cenários: a) furto e danos intencionais ao *hardware*; b) falhas de infra-estrutura (rede de potência, rede lógica, *hardware*); c) falhas de *software*, incluindo sistema

operacional e ferramentas (aplicativos); e d) manipulação indevida da informação, seja não intencional ou deliberada.

Quanto às consequências possíveis, pode-se listar: 1) atraso de cronogramas; 2) custo para repor e/ou reparar o *hardware*; 3) custo para recuperar, quando possível, a informação perdida ou alterada; custo de sua eventual perda definitiva; 4) degradação da imagem do grupo; impacto moral e emocional; decorrente prejuízo motivacional; 5) prejuízo na qualidade de conteúdo da pesquisa; 6) prejuízo relativo à divulgação por meio de artigos e participações em eventos; 7) apropriação indevida, por terceiros, de conteúdo intelectual desenvolvido pelo grupo; e 8) acesso indevido a material restrito ou sigiloso, seja pessoal ou institucional, com prejuízos morais e legais.

A avaliação do risco pode ser conduzida de forma qualitativa e quantitativa. O caso qualitativo consiste em levantamento formal de opiniões balizadas em experiência por especialistas. No caso quantitativo, utilizam-se dados acumulados de incidentes, quase-acidentes e acidentes. No estudo de caso objeto do presente trabalho, há as seguintes características:

- Alocação física em prédio público com acesso não controlado, com exceção de fins-de-semana e feriados, quando o acesso é parcialmente controlado.
- Precedentes quanto a acesso indevido e furtos no ambiente físico.
- Rede de potência relativamente estável, em que há conhecimento e divulgação prévios na maior parte das interrupções.
- Rede lógica confiável e estável.
- Informação, consistindo em:
  - Conhecimento colaborativo desenvolvido ou em desenvolvimento pelo grupo de pesquisa, colaboradores associados e alunos: o acesso indevido a esta informação pode ameaçar o direito de autoria, de propriedade intelectual e a imagem; a perda ou corrupção da informação pode implicar em perda do conhecimento em si, para além do mero prejuízo do relato sobre o conhecimento, exigindo retrabalho.
  - Dados pessoais e institucionais: sua perda pode ter prejuízos irrecuperáveis materiais e morais, bem como implicações legais.
  - Implementação de funcionalidades: sua operação indevida, inadvertida ou intencional, pode levar a perdas parciais ou globais de funções.
  - *Software* desenvolvido ou em desenvolvimento pelo grupo: sua perda, acesso indevido ou corrupção englobam as anteriores e leva aos prejuízos já relatados acima; sua

alteração sem coordenação ou controle, no processo de desenvolvimento, pode inviabilizar a própria construção.

A tabela abaixo traz, para o estudo de caso em foco, uma listagem de ocorrências, consequências, avaliação do risco associado e possíveis medidas que influenciam neste risco. A avaliação do risco foi feita com base em avaliação da probabilidade de ocorrência do evento indesejável (baixa, moderada, alta) e em avaliação da extensão da consequência (baixa, moderada, alta).

Ocorrência: Cenários	Consequências (de acordo com lista acima)	Risco	Medidas Evitadoras	Medidas Mitigadoras
(a) furto	1 a 8	moderado para alto	controle de acesso físico	replicação remota dos dados / criptografia / <i>backup</i>
(b) danos intencionais ao <i>hardware</i>	1 a 6	baixo	controle de acesso físico	redundância dos discos / replicação remota dos dados
(c) falhas de rede de potência	1 a 6	baixo	sistema de ininterrupção de energia	
(d) falhas de rede lógica	1; 4 a 6	baixo	gerenciamento e monitoramento da rede lógica	
(e) falhas de <i>hardware</i>	1 a 6	moderado	política de manutenção e monitoramento do <i>hardware</i>	espelhamento do disco / replicação remota dos dados / sistema de ininterrupção de energia / <i>backup</i>
(f) falhas de <i>software</i>	1; 3 a 8	alto	qualidade de <i>softwares</i>	controle de acesso lógico estratificado / políticas de utilização e termos de uso
(g) manipulação equivocada de informação	1; 3 a 6	moderado	política de conscientização / políticas de utilização e termos de uso / controle de acesso lógico estratificado	<i>backup</i>
(h) manipulação fraudulenta de informação	1; 3 a 8	alto	controle de acesso lógico estratificado, logs de auditoria, políticas de utilização e termos de uso	<i>backup</i>

Como medidas de gestão de risco, foram especificamente consideradas medidas evitadoras (que minoram a probabilidade de ocorrência do perigo) e medidas mitigadoras (que minoram a extensão da consequência, uma vez que o perigo tenha se efetivado).

*Hardware* com discos redundantes: mecanismos de RAID (*redundant array of inexpensive disks*) garantem que os dados nos discos internos do *hardware* sejam organizados para garantir uma maior disponibilidade da informação. No mecanismo de RAID 0; também conhecido como espelhamento; é realizada a réplica de cada bloco de informação para outro disco físico. No mecanismo de RAID 5 é realizado um cálculo de paridade que permite o

acesso aos dados no evento de falha de até 1 disco físico e no mecanismo RAID 6 podem ocorrer falhas em até 2 discos físicos que a informação não será perdida. No cenário proposto utiliza-se a combinação de mecanismos de RAID 1 e RAID 5 para a manutenção das áreas de sistema operacional e área de dados, respectivamente. A implementação do mecanismo de RAID 6 para a área de dados está em fase de avaliação e testes devido ao aumento na quantidade de discos físicos e o aumento da probabilidade de ocorrência de 1 (uma) falha entre eles. Utiliza-se discos internos mais simples e baratos do tipo SATA (*Serial ATA*) com MTBF (*mean time between failures*) menor que discos do tipo SAS (*Serial Attached SCSI*), SCSI (*small computer system interface*) e *Fibre Channel*<sup>[10]</sup>, o que exige um nível de redundância maior para resguardo em caso de falha.

Replicação remota dos dados: o dano decorrente da destruição ou furto do *hardware* estende-se além do valor físico do equipamento e inclui o valor da informação armazenada. Neste cenário, um mecanismo de réplica dos dados faz-se necessário para mitigar os problemas relacionados ao dano ou furto do *hardware*. A disposição destes mecanismos no ambiente não pode limitar-se a área física utilizada pelo *hardware* principal, pois em caso de furto o mecanismo de réplica também seria alvo do ataque. A solução para este problema deve incluir um mecanismo que opere remotamente sobre as informações mantidas no *hardware*. Os processos de réplica remota avaliados incluem sincronismo on-line ou periódico dos dados. Na réplica on-line os blocos de dados são replicados em tempo real para outra área de armazenamento, distante do ambiente do *hardware* principal. Alguns detalhes devem ser considerados no sincronismo em tempo real para garantir que os canais de comunicação fim-a-fim suportem a vazão de dados demandada pelas operações de escrita dos dados, normalmente requisitando a instalação de uma rede de comunicação privada. Na réplica periódica utiliza-se técnicas de diferenciação para que sejam atualizados apenas os dados modificados desde o último processo de sincronismo<sup>[11]</sup>. Optou-se pela implementação do mecanismo de réplica periódica devido a: a) limitações de *link* e infra-estrutura existentes na universidade e b) a criticidade do ambiente não justifica o investimento necessário para a implementação de um mecanismo de réplica on-line. Após sua implementação, o mecanismo deve ser incluído ao processo de gerenciamento e monitoramento do ambiente para que seja realizado o acompanhamento adequado.

Criptografia: O acesso físico ao *hardware* pode permitir que as informações residentes sejam acessadas por pessoas não autorizadas através da análise das áreas de dados do disco. Para impedir esta ação, deve-se implementar mecanismos de criptografia no sistema de

arquivos que garantam que apenas os detentores da frase secreta tenham acesso aos blocos de dados codificados. Atenção especial deve ser dada ao *software* e mecanismos de criptografia utilizados internamente para assegurar que apenas métodos e opções de criptografia resistentes a ataques sejam escolhidos. Com estes cuidados, a única opção disponível ao atacante é através de mecanismos de força-bruta, pelo qual todas as possíveis combinações são testadas exaustivamente. Uma frase secreta escolhida adequadamente torna o processo de dedução impraticável em um tempo hábil para o uso da informação.

Ininterrupção de energia por *no-break* com *software* de gerenciamento: Cortes e variações no fornecimento de energia podem quebrar a integridade dos dados mantidos no equipamento e fazer com que os dados fiquem inconsistentes, e em alguns casos até irrecuperáveis. Sistemas de bancos de dados, quando não implementados corretamente, podem conter dados incompletos nas tabelas devido a falha na rede elétrica. Como medida evitadora para este problema adotou-se um sistema de *no-break*, que provê a estabilização da rede elétrica e confere uma autonomia básica caso ocorram falhas no fornecimento de energia. Através de um *software* de gerenciamento é possível realizar a operação correta de finalização dos serviços e desligamento do sistema operacional, evitando maiores danos aos dados.

### **3.3. Utilização colaborativa e uso multi-institucional**

Em um grupo de trabalho, há a necessidade de colaboração, iteração e registro do conhecimento ao longo da realização das atividades. Não é difícil encontrar em um grupo, situações onde um conhecimento foi perdido, totalmente ou parcial, por desligamento ou afastamento de seus componentes.

A existência de uma ferramenta centralizada, com uma sintaxe simples, é fundamental para a documentação e arquivamento do conhecimento durante as fases preliminares de trabalho. Em muitos casos, os resultados destes trabalhos são convertidos em publicações ou servem de referência para trabalhos futuros. Dentro de um ambiente acadêmico há requisitos de documentação que são essenciais como a existência de um editor de fórmulas completo e mecanismos de indexação e referência dos conteúdos.

Neste sentido optou-se pelo modelo de *wiki*, onde é utilizado um sistema simples, de fácil edição e com o conjunto de recursos necessários ao projeto. Avaliou-se diversas ferramentas que implementam a tecnologia *wiki*, e entre elas, encontrou-se diversos problemas como a falta de integração com base de dados LDAP, ausência de mecanismos de

controle de acesso estratificado adequados, suporte limitado para sintaxe de fórmulas complexas etc.

No modelo em questão, o acesso a informação é completamente estratificado, o que exige o suporte a recursos avançados nas ferramentas. Dentre as ferramentas, nenhuma delas atendeu completamente a todos os requisitos, por isso, optou-se pela ferramenta que contemplava os itens indispensáveis e adaptou-se os itens restantes.

Para a expressão de fórmulas matemática adotou-se a sintaxe LaTeX por ser um padrão amplamente utilizado e de fácil manuseio. Os mecanismos para o suporte a sintaxe LaTeX foram adicionados a ferramenta *wiki*, incluindo extensões extras para a expansão dos recursos com a utilização de modelos para fórmulas e modificações básicas no código para adequação às necessidades do projeto. Devido as limitações de controle de acesso estratificado encontradas, houve a separação da ferramenta em ramificações direcionadas aos grupos de trabalho. Esta tarefa consistiu na manutenção do núcleo central da ferramenta isolado das ramificações; o que torna o processo de atualização e manutenção do código mais simples; e criou-se ramificações internas, utilizando recursos do sistema operacional e do banco de dados para derivar apenas as necessidades específicas de cada ramificação.

Um aspecto importante do uso colaborativo e multi-institucional é a possibilidade de acesso remoto. Através de ferramentas de acesso remoto disponibilizou-se um ambiente de trabalho completo que permite às pessoas autorizadas; a partir de qualquer ambiente que disponibilize uma conexão remota; a experiência de trabalho semelhante a encontrada por um usuário que trabalha utilizando um equipamento existente no laboratório. Para a escolha deste ambiente avaliou-se as seguintes características:

- Fácil utilização.
- *Software* cliente multi plataforma (Windows/Linux/Mac).
- Possibilidade de acesso à interface através de console de texto ou terminal gráfico.
- Baixo consumo de *link* Internet.
- Qualidade de definição do terminal compatível com a atividade exercida.
- Resumo de sessão em caso de desconexão do cliente.
- Controle de acesso.

Para prover estes recursos utilizou-se a tecnologia de *shell* seguro com a exportação de interface gráfica. Com este ambiente todo o tráfego de comunicação ocorre através de um canal de comunicação seguro, onde os dados são criptografados. A exportação da interface gráfica provê recursos adicionais aos usuários, pois remove a limitação existente nas conexões

remotas através de um console de texto convencional. Em uma interface gráfica o usuário tem acesso a todas as ferramentas e configurações idênticas a uma conexão realizada em uma estação de trabalho do laboratório.

É implementado um conjunto de sistemas operacionais em ambiente virtualizado que permitem a recuperação rápida em caso de dano ao ambiente. Desta forma, é possível permitir aos usuários a realização de testes e modificações do sistema operacional, sem comprometer o funcionamento do ambiente ou laboratório. Em um ambiente tradicional, estas permissões são cedidas nas estações de trabalho, e problemas que tornem o sistema operacional indisponível, impedem sua utilização por qualquer outro usuário até o que problema seja resolvido, muitas vezes necessitando a reinstalação do sistema operacional da estação. Com a implementação de um ambiente virtualizado todas as máquinas locais são protegidas, o que garante uma disponibilidade maior dos equipamentos para os usuários, e não impede a realização dos testes, pois há ambientes virtualizados configurados para este fim. Em caso de dano ao sistema virtualizado é possível repor o ambiente em até 10 minutos, o que seria inviável em uma operação com um *hardware* completo.

Através de um portal web, tem-se um ponto comum de acesso às ferramentas e outras capacitações do ambiente computacional. Através desta interface é disponibilizada um conjunto de informações públicas como uma apresentação do grupo de pesquisa, notícias relacionadas às equipes de trabalho, artigos, materiais para referência, etc. Durante a avaliação da ferramenta de CMS (*Content management system*) procurou-se por uma ferramenta capaz de estratificar o acesso as informações utilizando o ambiente de usuários centralizado com a tecnologia LDAP. Em um segundo momento a ferramenta será expandida com a utilização de módulos para a disponibilização das ferramentas desenvolvidas pelo grupo.

### **3.4. Gerenciamento centralizado de usuário**

O gerenciamento de usuários torna-se complexo e trabalhoso a medida que cresce a quantidade de aplicações e ferramentas que demandam recursos de autenticação e autorização. Os usuários têm que memorizar múltiplas senhas para obter acesso ao conjunto de ferramentas e aplicações específicas que fazem parte das suas atividades diárias, que torna o gerenciamento de usuários rapidamente confuso. Algumas tentativas de utilização do mesmo usuário e senha para todos os acessos pode aparentemente resolver o problema, mas restrições de negócio e de aplicações farão com que este sincronismo se quebre facilmente.

Devido ao crescente volume de aplicativos e ferramentas necessárias para cobrir as necessidades diárias do grupo de trabalho, pesquisou-se tecnologias que pudessem eliminar o problema das múltiplas bases de usuários, com a centralização do repositório para o gerenciamento de usuários. Esta tecnologia deveria incorporar recursos de controle centralizado para a aplicação de políticas, gerenciamento dos acessos, integração com diversos sistemas operacionais, máquinas virtuais, aplicações web desenvolvidas internamente e externamente, ferramentas de gerenciamento, portal, *wiki*, acesso remoto, etc.

A tecnologia LDAP (*Lightweight Directory Access Protocol*)<sup>[12]</sup> atendeu as necessidades do grupo de trabalho, pois permitiu a integração dos controles de acesso das ferramentas atuais e futuras. Por ser uma tecnologia bem definida, diversas aplicações suportam ou possuem extensões para o suporte e uso desta integração. Devido a flexibilidade adquirida com a tecnologia, o suporte ao protocolo LDAP tornou-se um requisito na adoção de futuras ferramentas que necessitem de autenticação e controle de acesso estratificado.

Para cada aplicação ou ferramenta disponibilizada modela-se a estrutura de informação no repositório LDAP para fornecer o nível de estratificação necessário para a realização do controle de acesso. A partir desta organização os usuários são incluídos apenas aos grupos e ferramentas aos quais possuem envolvimento. Em um processo de criação de usuário, avaliam-se as necessidades específicas de acesso do usuário (*need-to-know*) e habilita-se apenas os acessos estritamente necessários para a realização de suas funções (*least privilege*). Há a possibilidade de adoção de uma hierarquia para a redistribuição de permissões, de forma que a responsabilidade pelo gerenciamento de permissões seja distribuída entre os coordenadores.

Ferramentas para prover recursos de controle de processamento estratificado estão em processo de pesquisa. Para o cenário proposto onde há a presença de grupos de trabalho utilizando ambientes remotos para a execução de aplicações científicas, existe a necessidade de limitação do consumo dos recursos do *hardware*, pois caso contrário, uma operação ou execução pode comprometer a atividade dos demais grupos devido ao uso completo do processador ou memória típicos de determinadas aplicações científicas.

Todas as atividades e acessos realizados são registradas em arquivos nos níveis necessários para prover uma investigação em caso de incidente.

### **3.5. Gerenciamento e monitoramento do ambiente**

Com o crescimento da quantidade de estações de trabalho, servidores, máquinas virtuais, *softwares* e aplicativos no ambiente, tornam-se impraticáveis o gerenciamento e supervisão de cada componente isolado. Não é possível a realização de ações preventivas neste cenário, pois não existem mecanismos para o acompanhamento e histórico dos componentes.

Formas alternativas e centralizadas devem existir para o monitoramento e gerenciamento do ambiente, que permitam visualizar a saúde dos componentes individuais a partir de um ponto de controle e acesso. A adoção da tecnologia SNMP (*Simple Network Management Protocol*) e respectivos agentes de monitoramento e acesso remoto aos recursos permitiram o monitoramento de todos os pontos cruciais para o funcionamento do ambiente, com o registro histórico destas informações. Com base nos dados coletados é possível detectar padrões de uso dos recursos e realizar ações preventivas. Um componente de notificação envia mensagem de alerta sobre eventos ocorridos no ambientes.

## **4. Gerenciamento administrativo**

Nesta seção são detalhados os aspectos pertinentes ao gerenciamento administrativo do ambiente que fornecer aos coordenadores o nível de controle para a gestão do ambiente computacional.

### **4.1. Controle de acesso físico**

O acesso às áreas que hospedam o *hardware* da infra-estrutura computacional deve ser restrito, sempre que possível, para evitar ações não intencionais e fraudulentas. As restrições se vinculam às políticas de controle; por exemplo, usuários autorizados podem ter acesso mediante chaves, senhas, cartões, biometria, etc; alarmes reforçam a segurança. Outra alternativa seria a alocação do *hardware* em armários trancados. No caso em foco, ainda não há uma área específica para a manutenção dos equipamentos; portanto, medidas de controle de acesso físico foram as indicadas.

### **4.2. Políticas de utilização e termos de uso**

A elaboração de uma política de utilização e termo uso do ambiente tem o objetivo de garantir ao gestor que todos os membros participantes do grupo estejam cientes e de acordo com as regras que regem o acesso, manipulação e desenvolvimento de qualquer informação coberta no âmbito do projeto. Para este fim, o documento deve esclarecer os níveis de atuação e

responsabilidades do participante de forma clara e precisa, muitas vezes com documentos específicos direcionados para a atividade exercida pelo membro. Este processo pode requerer a consulta de especialistas da área de pesquisa ou advogados que devem avaliar se as políticas e termos garantem os direitos do projeto e de seus envolvidos.

O intuito de uma política de utilização e termo de uso não é o de limitar o participante, mas o de assegurar que os interesses do projeto com relação a questões de sigilo, acesso à informação, segurança da informação e direitos autorais e de propriedade intelectual sejam resguardados.

### **4.3. Padronização**

A padronização das atividades do grupo simplifica o acesso e a organização da informação no ambiente. Diversos mecanismos de padronização foram adotados afim de facilitar e organizar as atividades do grupo, incluindo:

Padronização de textos e nomes de páginas na *wiki*: A inclusão de informações em uma ferramenta de *wiki* pode, com o aumento do volume de dados armazenados, tornar o processo de recuperação do conhecimento complexo. Uma página de *wiki* não possui vínculos à outras páginas, o que muitas vezes faz com que trabalhos completos sejam considerados de forma parcial. A padronização em uma *wiki* pode incluir a adoção de um prefixo para as páginas relacionadas, agrupamento de páginas, capitalização de caracteres, *links* que remetam à página principal, *links* de acesso rápido, monitoria de páginas, etc.

Padronização dos repositórios de informação: Deve-se manter a menor quantidade possível de repositórios de informação para não tornar difícil o processo de localização de informações. Alguns exemplos incluem a disponibilização dos arquivos de trabalho dos usuários na pasta "usuários", os arquivos dos diversos grupos de pesquisa na pasta "grupos", os arquivos de desenvolvimento do projeto ABC na pasta "projetos/abc", etc.

Padronização da linguagem e estilo de programação utilizada: Deve-se unificar a linguagem de programação utilizada pelo grupo, adotando-se padrões para a nomeação e declaração de funções e estilos de programação. O processo de integração dos trabalhos dos grupos pode vir a exigir o retrabalho da equipe se não houver uma uniformidade no desenvolvimento. Esta decisão deverá apoiar-se no conhecimento existente, incluindo a linguagem de programação de domínio pelo grupo, facilidade de utilização e recursos disponíveis na linguagem.

#### **4.4. Controle de versão do *software***

O processo de desenvolvimento de *software* quando realizado por um grupo de trabalho contém características que demandam a utilização de ferramentas adequadas. Para coordenar todos os requisitos faz-se necessário a implementação de um *software* de controle de versões, que possui algumas características como:

**Trabalho em equipe:** Permite que a equipe de trabalho tenha acesso ao mesmo repositório de manutenção do código fonte. Todas as informações são mantidas em um repositório central, facilitando a implementação de políticas de *backup* e controle de acesso.

**Controle de acesso ao código-fonte:** Permite o controle de acesso estratificado a partes do código fonte com a implementação de listas de controle de acesso e registro em arquivos de log de todas as operações realizadas no repositório.

**Bloqueio de operações simultâneas:** É utilizado um recurso bloqueio que mantém a consistência dos arquivos armazenados no repositório. Um arquivo aberto para modificação por um membro da equipe não pode ser modificado simultaneamente por outro membro.

**Controle do histórico:** O recurso de histórico permite a reversão dos arquivos às suas versões anteriores sem comprometer a integridade da informação. Nenhuma informação é definitivamente excluída do sistema, o que permite a visualização histórica a qualquer momento.

Na etapa atual do projeto estão sendo avaliadas e testadas ferramentas de SCM (*Source Configuration Management*) que proverão os recursos acima aos grupos de trabalho e pesquisa.

#### **4.5. Documentação**

O processo de documentação de qualquer atividade visa registrar o conhecimento adquirido durante as etapas de evolução do conhecimento. Sem um processo de documentação bem definido, pode-se gerar retrabalho para a reprodução do conhecimento prévio por novos participantes do projeto. O processo de documentação implementado cobre os seguintes aspectos, conforme abaixo:

**Documentação das atividades:** Cada participante registra a evolução do seu conhecimento através da ferramenta de *wiki*. Os demais participantes do grupo interagem com a evolução do conhecimento realizando modificações, corrigindo materiais, evoluindo com o conhecimento e realizando discussões através da ferramenta. Em um segundo cenário, a ferramenta é utilizada para a organização pessoal do trabalho como um auxiliador no

desenvolvimento de artigos, documentações de ferramentas, manuais de operação e escrita de dissertações.

Documentação do código fonte: A documentação do código fonte de um *software* requer atenção especial para prover a continuidade do projeto, evolução do conhecimento e iteração entre os desenvolvedores. O processo de continuidade e evolução do desenvolvimento será agilizado se o *software* obedecer padrões para o desenvolvimento e documentação do código. Com isto, novos colaboradores podem entender rapidamente o funcionamento das funções, módulos e suas interligações através da leitura da documentação existente. Esta implementação está em processo de avaliação e deve ser integrada em breve ao processo de desenvolvimento de *software*.

## 5. Considerações finais

O trabalho mostrou um processo de projeto de ambiente computacional para atender demandas de pesquisa e desenvolvimento, bem como de formação acadêmica, o qual tem como principais características o uso, sempre que possível, de sistemas livres (FLOSS) e de colaboração na geração do conhecimento.

## Bibliografia

1. M. B. de Conti and V. Domiciano, *Escoamento Potencial devido a Corpos Imersos Profundamente Avançando com Velocidade Constante*, EPUSP, Ed. 2000.
2. V. Domiciano, *Cálculo da Resistência de Ondas através de um Método de Elementos de Contorno*. Escola Politécnica da USP, 2002.
3. V. Domiciano and M. B. de Conti, *Escoamento Potencial devido a Corpos Avançando com Velocidade Constante na Presença de Superfície Livre*, EPUSP, Ed. 2006.
4. M. B. de Conti and R. Beck, "Considerations about B-Spline hull surface representation," 2003.
5. M. B. de Conti and V. Domiciano, "Software Poseidon," 2008.
6. V. Domiciano and M. B. de Conti, "Software Neptune," 2008.
7. M. Välimäki, "Dual Licensing in Open Source Software Industry", *Systemes d'Information et Management*, vol. 8, iss. 1, pp. 63–75, 2003.
8. B. N. Veelo, "Free Software as an Option for Ship Design", *Schiffstechnik*, vol. 52, pp. 172–188, 2005.
9. B. N. Veelo, "The Potential of Free Software for Ship Design".
10. A. Brinkmann and S. Effert, "Storage Cluster Architectures", *Paderborn Center for Parallel Computing*, pp. 113–121, 2008.
11. Tridgell, A., "Efficient Algorithms for Sorting and Synchronization", 2000.
12. S. Tuttle, A. Ehlenberger, R. Gorthi, J. Leiserson, R. Macbeth, N. O. S. ad Ranahandola, M. Storrs, and C. Yang, *Understanding LDAP - Design and Implementation*. IBM Redbooks, 2004.