

# Template em LATEX para os artigos da revista LUPS

Nome Autor 1, UFPel; Nome Autor 2, UFPel

**Resumo**—Resumo não é uma introdução ao artigo, mas sim uma descrição sumária da sua totalidade, na qual se procura realçar os aspectos mencionados. Deve ser discursivo, e não apenas uma lista dos tópicos que o artigo cobre. Deve-se entrar na essência do resumo logo na primeira frase, sem rodeios introdutórios nem recorrendo à fórmula estafada "Neste artigo...". Este deve ser escrito em um parágrafo único e conter entre 120 e 180 palavras. Devem-se evitar em um resumo: (i) Símbolos e contrações que não sejam de uso corrente; (ii) Fórmulas, equações, diagramas etc., que não sejam absolutamente necessários. Não citar referências bibliográficas no resumo. Logo após o resumo devem vir as palavras-chave, que são palavras representativas do conteúdo do documento. As palavras-chave devem ser separadas entre si por vírgula e finalizadas por ponto. No mínimo 3 e no máximo 5 palavras.

**Palavras-Chave**—Palavra-chave 1, Palavra-chave 2, Palavra-chave 3, Palavra-chave 4.



## 1 Introdução

Este arquivo é apenas uma demonstração de um artigo da Revista LUPS. Seu objetivo é servir como um "arquivo de partida" para os autores, pois traz uma série de itens comuns, tais como seções, subseções, tabelas, imagens, entre outros que certamente serão utilizados no decorrer da redação de cada capítulo. Sucesso a todos na escrita.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco

laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

O último parágrafo da Introdução costuma ser dedicado a apresentar a estrutura do artigo. Neste parágrafo é feita uma breve descrição da intenção de cada seção.

## 2 Infra-estruturas para computação ubíqua

Esta seção é responsável por apresentar os mais importantes conceitos e definições de Computação Ubíqua e *Grids* Computacionais, bem como os princípios da área e uma série de conceitos relacionados.

- 
- **Nome Autor 1:** Programa de Pós-Graduação em Computação, Universidade Federal de Pelotas - UFPel, Centro de Desenvolvimento Tecnológico - CDTec.  
E-mail: e-mailautor1@inf.ufpel.edu.br
  - **Nome Autor 2:** Programa de Pós-Graduação em Computação, Universidade Federal de Pelotas - UFPel, Centro de Desenvolvimento Tecnológico - CDTec.  
E-mail: e-mailautor2@inf.ufpel.edu.br



Figura 1. Evolução da Computação

## 2.1 Computação Ubíqua

Na primeira Era da Informática, através dos *mainframes*, compartilhados por vários usuários, era disponibilizada a computação as pessoas. Atualmente, na segunda Era da Informática, tem-se a era do computador pessoal [?]. O cenário previsto por Mark Weiser no início da década de 1990 é considerado como a Terceira Era da Informática [?]. A questão abordada por Weiser é que no futuro a computação estaria presente até mesmo nos objetos mais banais de nosso dia-a-dia, como etiquetas de roupas, canetas, livros, entre outros. Esta computação auxiliaria nas tarefas humanas diárias com mínimas mudanças na forma de exercê-las. As tecnologias estariam entrelaçadas com o cotidiano de tal forma que se tornariam indistinguíveis dele.

A Computação Ubíqua (*Ubiquitous Computing*), ou Ubicomp, muitas vezes chamada de Computação Pervasiva (*Pervasive Computing*), é um novo paradigma computacional onde o usuário dispõe de acesso a computação de forma invisível e onipresente. Invisível no sentido de que o usuário não necessita de conhecimentos específicos da tecnologia para beneficiar-se dela. A computação passa a fazer parte da vida do

usuário de forma natural. A onipresença se dá ao fato da possibilidade de uso da tecnologia em qualquer lugar. A figura ?? ilustra a evolução da computação.

Em [?], os autores defendem que a Computação Ubíqua é criada a partir da união de três outras áreas: Computação Móvel, Computação em Grade e Computação com Consciência de Contexto. A seguir são caracterizadas cada uma das áreas.

- **Computação Móvel:** Consiste na mobilidade tanto de hardware quanto de software. No caso de hardware pode-se considerar dispositivos móveis como notebooks e PDAs. No caso de software são aplicações que migram de um dispositivo para outro através de uma rede sem interromper a computação sendo realizada.
- **Computação em Grade:** De acordo com [?], é basicamente Computação Distribuída em um ambiente de grande heterogeneidade com recursos dispersos. A próxima geração de Computação em Grade associa novas tecnologias como webservices e peer-to-peer.
- **Computação com Consciência de Contexto:** Estrutura de software inteligente que

se adapta e reage ao ambiente em que está inserido [?].

### 2.1.1 Princípios da Computação Ubíqua

De acordo com [?], pelos menos três princípios são identificados na Computação Ubíqua, a saber:

- **Diversidade:** Ao contrário do computador pessoal, que é um dispositivo de propósito geral que atende várias necessidades distintas do usuário, os dispositivos ubíquos acenam com uma nova visão da funcionalidade do computador, que é a de propósito específico, que atende necessidades específicas de usuários particulares. Apesar de vários dispositivos poderem oferecer funcionalidades que se sobrepõem, um pode ser mais apropriado para uma função do que outro.
- **Descentralização:** Na computação ubíqua as responsabilidades são distribuídas entre vários dispositivos pequenos que assumem e executam certas tarefas e funções. Estes dispositivos cooperam entre si para a construção de inteligência no ambiente, que é refletida nas aplicações.
- **Conectividade:** Na computação ubíqua, tem-se a visão da conectividade sem fronteiras, em que dispositivos e as aplicações que executam neles movem-se juntamente com o usuário, de forma transparente, entre diversas redes heterogêneas, tais como as redes sem fio de longa distância, entre outras.

## 2.2 Grids Computacionais

Grids Computacionais, também conhecidos por grades, são infraestruturas de computação que conectam múltiplos recursos computacionais de diversos computadores para permitir a execução de aplicações com alta demanda de recursos computacionais, memória e espaço em disco. A figura ?? ilustra essa ideia. Neste capítulo, toda vez que houver referência as palavras *grid* ou grade computacional, deve-se entender como sinônimos. A ideia do *grid* é utilizar todo recurso computacional ocioso para permitir que novos tipos de aplicações sejam criados ou estender aplicações existentes utilizando recursos de sistemas distribuídos altamente conectados [?].

De acordo com [?], *Grids* Computacionais nasceram da comunidade de Processamento de Alto



Figura 2. Ambiente de Grid Computacional

Desempenho, motivada pela utilização de computadores independentes e dispersos como plataforma de execução de aplicações paralelas.

O termo *grid* computacional surgiu da analogia com as redes de interligação do sistema de energia elétrica (do inglês *power grids*), nas quais utiliza-se a eletricidade sem necessariamente saber a localização física da usina que a gerou, sendo sua utilização totalmente transparente aos seus usuários [?], [?]. Ao contrário dos sistemas distribuídos tradicionais, esse ambiente precisa considerar questões como segurança, acesso uniforme aos recursos geograficamente distribuídos (independentemente da localização física dos mesmos), descoberta e agregação dinâmica de recursos e qualidade de serviço. Dessa forma, entre os grandes desafios apresentados por esse ambiente, são mais relevantes [?]:

- **heterogeneidade:** os recursos de hardware e software, que constituem o ambiente podem ser os mais diversos possíveis e agregam uma grande quantidade de tecnologias diferentes.
- **escalabilidade:** o aumento do número de recursos pode acarretar em perda de desempenho do ambiente.
- **dinamismo:** a natureza dinâmica desse tipo de ambiente deve ser cuidadosamente tratada porque a ocorrência de falhas na rede de comunicação e nas máquinas agregadas pode alterar a disponibilidade dos recursos.
- **compartilhamento de recursos:** como os recursos não são dedicados, pode haver uma elevada variação na quantidade de poder computacional disponível no ambiente.

- **multiplicidade de domínios administrativos:** os recursos podem pertencer a diferentes proprietários, cada um com regras distintas para a gerência e utilização dos recursos.

Diante de todos esses desafios, é necessário que a computação em *grid* ofereça meios para que o desenvolvedor da aplicação possa escrever programas em linguagens de alto nível, capazes de acessar o *grid* e utilizar seus recursos de forma eficiente e transparente sem que para isso o usuário tenha que se preocupar em tratar questões do ambiente [?].

Os *Grids* se tornaram possíveis devido à grande melhoria em desempenho e redução de custos, tanto de redes de computadores, quanto de microprocessadores. Através deles é possível explorar as potencialidades das redes de computadores, com o objetivo específico de disponibilizar camadas virtuais que permitam a um usuário ter aplicações altamente exigentes, bem como aderir a comunidades virtuais de grande escala, com uma grande diversidade de recursos de computação e de repositórios de informações [?].

### 3 Estudo de caso: SimGrid

A ferramenta *SimGrid*, concebida em 1999, foi fruto de um projeto de pós-doutorado desenvolvido por Henri Casanova. A motivação foi a necessidade de se utilizar simulação, ao invés de experimentos reais, no estudo prático de algoritmos de escalonamento para aplicações científicas paralelas em plataformas computacionais distribuídas e heterogêneas [?].

A tabela ?? resume alguns aspectos da ferramenta *SimGrid*.

Tabela 1  
Alguns aspectos da ferramenta *SimGrid*

Aspectos	SimGrid
Linguagem de Implementação	C
Plataformas Suportadas	Linux, Mac, Win
Interface da ferramenta	Modo texto
Suporte a simulação <i>multithreaded</i>	Não

Dentre as características da ferramenta *SimGrid*, destacam-se as seguintes:

- Facilita o estudo de estratégias de escalonamento de tarefas.

- Provê componente que auxilia o desenvolvimento de aplicações reais para *grids*.
- Provê componente que auxilia o desenvolvimento de programas MPI.
- Provê componente para a avaliação de tarefas paralelas em sistemas distribuídos.

#### Arquitetura do *SimGrid*

A arquitetura do *SimGrid*, ilustrada na figura ?? mostra os componentes da arquitetura e suas relações. Cada componente é descrito na sequência.

##### 1) Ambiente MSG

O ambiente MSG visa a realização de simulações em termos de agentes de comunicação. É usualmente utilizado no estudo de heurísticas para um problema, possibilitando uma comparação entre elas. O realismo da simulação não é objetivo principal deste módulo, e muitos detalhes técnicos da plataforma do *grid* são omitidos.

##### 2) Ambiente GRAS

GRAS (*Grid Reality and Simulation*) é um ambiente concebido como um *framework* que facilita o desenvolvimento de aplicações distribuídas orientadas a eventos.

##### 3) Ambiente SMPI

O ambiente SMPI é um *framework* que permite executar aplicações MPI não alteradas, tanto em modo de simulação, quanto em modo de aplicação real (atuando como um GRAS para MPI).

### 4 Considerações Finais

Esta seção tem por objetivo apresentar as considerações finais do capítulo.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad

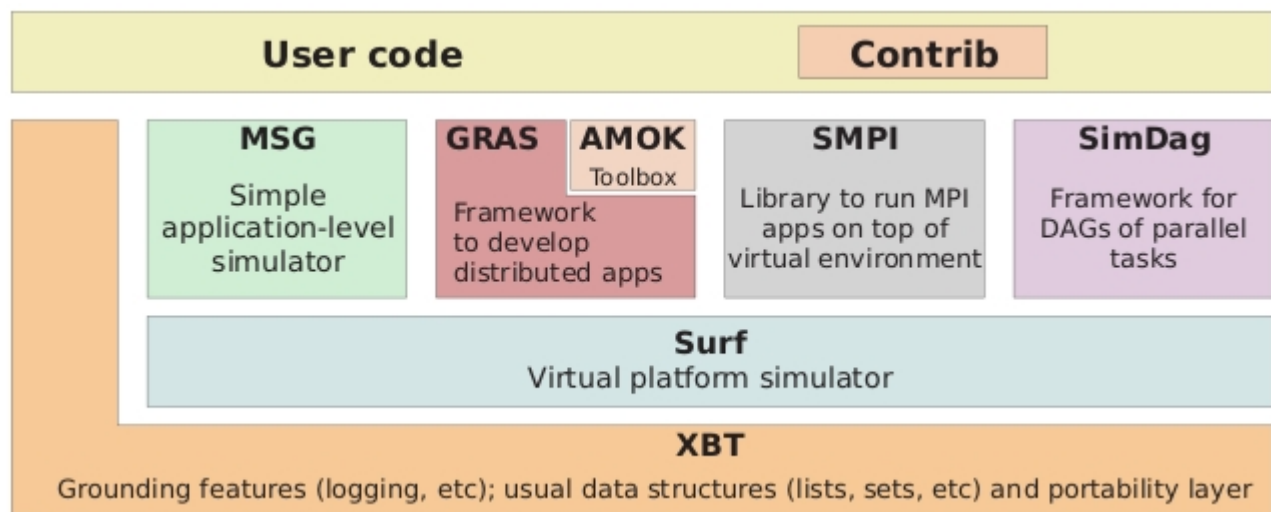


Figura 3. Componentes do SimGrid e suas relações

minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

## Referências

- [1] Mark Weiser. The Computer for the 21st Century. *Scientific American*, 265(3):94–104, September 1991.
- [2] Jan Krikke. T-engine: Japan's ubiquitous computing architecture is ready for prime time. *IEEE Pervasive Computing*, 4:4–9, April 2005.
- [3] Adenauer Yamin, Iara Augustin, Luciano Cavalheiro da Silva, Rodrigo Araujo Real, Alberto E. Schaeffer Filho, and Claudio F. Resin Geyer. Exehda: adaptive middleware for building a pervasive grid environment. In *Proceeding of the 2005 conference on Self-Organization and Autonomic Informatics (I)*, pages 203–219, Amsterdam, The Netherlands, The Netherlands, 2005. IOS Press.
- [4] Adenauer Corrêa Yamin. Arquitetura para um ambiente de grade computacional direcionada às aplicações distribuídas, móveis e conscientes de contexto da computação pervasiva. Tese de doutorado em ciência da computação, Instituto de Informática-UFRGS, Porto Alegre-RS, 2004.
- [5] Bill N. Schilit and Marvin M. Theimer. Disseminating active map information to mobile hosts, 1994.
- [6] R. B. Araujo. Computação ubíqua: princípios, tecnologias e desafios. *XXI Simpósio Brasileiro de Redes de Computadores*, pages 45–115, 2003.
- [7] Ian Foster, Carl Kesselman, and Steven Tuecke. The anatomy of the grid: Enabling scalable virtual organizations. *Int. J. High Perform. Comput. Appl.*, 15:200–222, August 2001.
- [8] Ian Foster and Carl Kesselman, editors. *The grid: blueprint for a new computing infrastructure*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.
- [9] Ian Foster, Carl Kesselman, Jeffrey M. Nick, and Steven Tuecke. Grid services for distributed system integration. *Computer*, 35:37–46, June 2002.
- [10] Ian Foster and Carl Kesselman. *The Grid 2: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2003.
- [11] Parvin Asadzadeh, Rajkumar Buyya, Chun Ling Kei, Deepa Nayar, and Srikumar Venugopal. Global grids and software toolkits: A study of four grid middleware technologies. In *HIGH PERFORMANCE COMPUTING: PARADIGM AND INFRASTRUCTURE*, LAURENCE YANG AND MINYI GUO (EDS), ISBN: 0-471-65471-X. Wiley Press, 2005.
- [12] Aletéia Patrícia Favacho Araújo. Paralelização autônoma de metaheurísticas em ambientes de grid. Tese de doutorado em ciência da computação, Pontifícia Universidade Católica do Rio de Janeiro/PUC-Rio, Rio de Janeiro, RJ, 2008.
- [13] Mark Baker, Rajkumar Buyya, and Domenico Laforenza. Grids and grid technologies for wide-area distributed computing. *Softw. Pract. Exper.*, 32:1437–1466, December 2002.
- [14] Henri Casanova. Simgrid: A toolkit for the simulation of application scheduling. In *Proceedings of the 1st International Symposium on Cluster Computing and the Grid, CCGRID '01*, pages 430–, Washington, DC, USA, 2001. IEEE Computer Society.



**Nome do Autor 1** Colocar uma biografia do Autor 1 aqui, similar a apresentação do Currículo Lattes. Exemplo: Mestrando do Programa de Pós-Graduação em Computação da Universidade Federal de Pelotas, na linha de pesquisa Sistemas Paralelos e Distribuídos. Também é pós-graduando *Lato Sensu* em (...) pela Universidade (...). Possui Bacharelado (2011) em Ciência da Computação pela Universidade (...). Tem experiência na área de Ciência da Computação, com ênfase nas áreas de (...). Quanto a foto, não deve ser de corpo inteiro (deve ser similar ao avatar de exemplo) e convertida em escala de cinza.



**Nome do Autor 2** Colocar uma biografia do Autor 2 aqui, similar a apresentação do Currículo Lattes. Exemplo: Mestrando do Programa de Pós-Graduação em Computação da Universidade Federal de Pelotas, na linha de pesquisa Sistemas Paralelos e Distribuídos. Também é pós-graduando *Lato Sensu* em (...) pela Universidade (...). Possui Bacharelado (2011) em Ciência da Computação pela Universidade (...). Tem experiência na área de Ciência da Computação, com ênfase nas áreas de (...). Quanto a foto, não deve ser de corpo inteiro (deve ser similar ao avatar de exemplo) e convertida em escala de cinza.