

## **PROPOSTA DE MODELO DE BANCO DE DADOS DISTRIBUÍDO PARA O PRONTUÁRIO ELETRÔNICO ÚNICO DO PACIENTE**

### **PROPOSED DATABASE MODEL DISTRIBUTED FOR THE UNIQUE PATIENT ELECTRONIC PRONOUNER**

**Prof. Ms. José da Silva Azanha Neto – Universidade Nove de Julho**

jose.azanha@uninove.br

**Prof. Ms. Débora Vigilia Canne – Universidade Nove de Julho**

debora.canne@uni9.pro.br

**Prof. Dr. Luis Antonio Coppa Ybarra – Universidade Nove de Julho**

lybarra@uni9.pro.br

**Prof. Ms. Antonio Andrade dos Santos – Universidade Nove de Julho**

uni9aas@gmail.com

**Prof. Dr. Jorge Rodolfo Beingolea Garay – Universidade de Mogi das Cruzes/Universidade de São Paulo**

jorgegaray@umc.br

**Prof. Dr. Sergio Takeo Kofuji – Universidade de São Paulo**

kofuji@lsi.usp.br

## **Resumo**

Os modelos atuais de prontuários são isolados e proprietários, tornando as informações do paciente dispersas, redundantes e sem integração, impactando na qualidade dos tratamentos e análises epidemiológicas. Foi proposto um modelo de dados composto por três níveis de informação: Síntese, Síntese Estendida e Prontuário Completo. A síntese é acessada com frequência e distribuída em bases de dados e os outros dois níveis são acessados quando a síntese for insuficiente. A grade computacional oferece objetivos de descentralização, escalabilidade e tolerância a falhas. O modelo foi suportado

pelos servidores JAVA Parallel Processing Framework, GlassFish e MySQL. Nos resultados, foram elaborados 19 casos de uso e o modelo de entidade-relacionamento. Avaliou-se a latência da rede em 4 configurações na transferência da síntese pelo evento no MySQL. De 5 para 6 servidores, reduziu-se 29%. De 5 para 8 servidores, reduziu-se entre 49-67%. Concluiu-se que, a fragmentação do modelo diminuiu a latência, viabilizando sua operacionalização.

**Palavras-chave:** Prontuário Eletrônico do Paciente; Banco de Dados Distribuídos; Computação em Grade; Tolerância a Falhas; Desempenho.

### **Abstract**

Current models of patient records are isolated and proprietary, making patient information dispersed, redundant and unintegrated, impacting treatment quality and epidemiological analysis. It was proposed a data model composed by three levels of information: Synthesis, Extended Synthesis and Complete List. The synthesis is frequently accessed and distributed in databases and the other two levels are accessed when the synthesis is insufficient. The computational grid offers decentralization, scalability, and fault tolerance objectives. The model was supported by the JAVA Parallel Processing Framework, GlassFish and MySQL servers. In the results, 19 use cases and the entity-relationship model were elaborated. The latency was evaluated in 4 configurations in the transfer of the synthesis by the event in MySQL. From 5 to 6 servers, it was reduced by 29%. From 5 to 8 servers, it fell by 49-67%. It was concluded that, the fragmentation of the model decreased the latency, making possible its operationalization.

**Keywords:** Electronic Health Record; Distributed Databases; Grid Computing; Fault Tolerance; Performance.

## Introdução

Não há dúvidas que o desenvolvimento tecnológico e a informatização na área da saúde beneficiaram de forma significativa a qualidade da assistência médica (KING et al., 2014; LUNARDELLI et al., 2015). O alto grau de heterogeneidade das informações e dos serviços de saúde no ambiente hospitalar propiciou o desenvolvimento do Prontuário Eletrônico do Paciente (PEP). O PEP é o conjunto de dados relacionados ao passado, ao presente e ao futuro do estado de saúde física ou mental de um indivíduo. Estes dados devem estar num banco de dados (BD) de onde seja possível recuperar, enviar, armazenar, disponibilizar e analisar dados em texto ou multimídia a partir de um sistema eletrônico (HERSH et al., 2013).

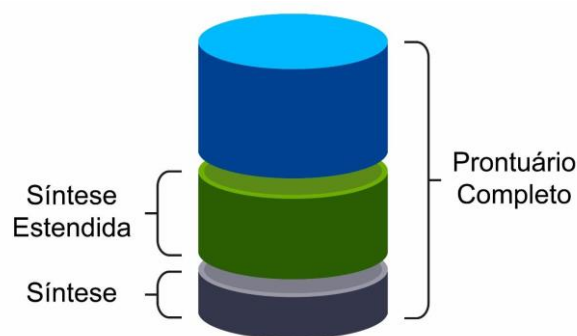
Por outro lado, o conjunto de problemas associados os modelos atuais de PEP é grande. Eles apresentam pouca interoperabilidade e portabilidade, o que piora com a chegada de tecnologias móveis e ubíquas (SILVA et al., 2015). Não oferecem ao paciente acesso ao seu próprio histórico clínico, que em tese é um direito legal que assiste a todos. O PEP de cada pessoa está disperso em diversos sistemas de PEPs, que dependendo de onde o paciente foi atendido há diversos níveis de atenção em saúde (SILVA et al., 2015; COOREVITS et al., 2013). Isso causa a falta de coesão das informações, riscos de erros no tratamento médico, duplicidade de exames e a ausência de uma síntese, que seja dotada de significado clínico, com vistas à análise e tomada de decisão médica eficiente e precisa (GAZZARATA et al., 2017). Nos PEPs atuais, os pacientes não validam os procedimentos e medicamentos que lhes foram ministrados, o que favorece cobranças indevidas, sem o conhecimento dos pacientes ou das instituições pagadoras, tais como planos ou sistemas públicos de saúde.

Não há uma abordagem de PEP que atenda todas estas necessidades nos modelos atuais. Não há perspectivas para que ocorra integração e interoperabilidade entre os PEPs existentes. Migrar dados para um PEP único produz aumento no volume de dados e o crescimento de registros de exames que pode chegar aos 5GB por dia por paciente (FURUIE et al., 2003; HMEIDI et al., 2014). Isso pode tornar qualquer modelo atual de PEP inacessível. Pois, há ambulatórios que atendem mais de 5.000 pacientes por dia, cujo do volume estimado de dados pode chegar a 20TB de informações *in memory* a cada dia. Mesmo que todos os prontuários estejam em servidores locais e com rede de alta velocidade a latência será alta a ponto de inviabilizar os atendimentos (FURUIE et al., 2003; SOARES CRUZ et al., 2011). Deve-

se considerar também que manter PEPs representa alto custo para as instituições, sem que haja retorno tangível. Por isso há pouco esforço em interoperabilidade, integração, segurança e disponibilização online para pacientes.

Diante de tanta diversidade de problemas, com poucas perspectivas de solução em médio prazo, optou-se por iniciar o projeto de um novo modelo de dados que viabilize o desenvolvimento do Prontuário Eletrônico Único do Paciente (PEUP), para que no futuro a interoperabilidade, a integração e a disponibilidade se tornem viáveis. Neste conceito, o PEUP é uma propriedade do paciente. O novo modelo de dados, composto por três níveis de informações e representado pela Figura 1, visa inicialmente manter a disponibilidade de acesso para pacientes e profissionais de saúde e manter um histórico coerente do paciente. Os três níveis de informação são: Síntese, Síntese Estendida (SE) e Prontuário Completo (PC). Muitas perguntas deverão ser respondidas até que este modelo de PEUP se torne viável. O objetivo deste trabalho é avaliar, por simulação, se as tecnologias disponíveis viabilizam a implementação do novo modelo de dados, quanto à integridade de armazenamento, acesso, disponibilidade e desempenho.

Figura 1 - Os três níveis de informação do PEUP  
Do Autor



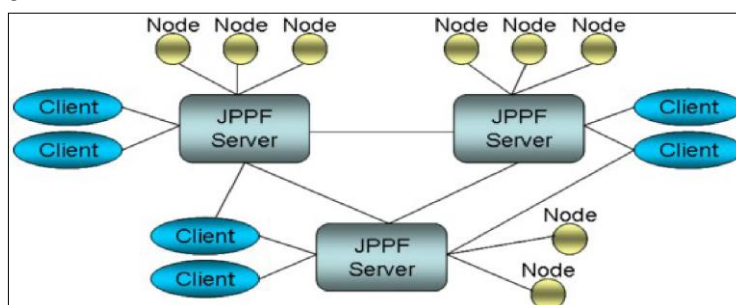
### Relato Circunstanciado

A Síntese foi concebida para ter baixo volume de dados (textos e hiperlinks) e acompanhar o paciente em atendimentos de rotina, emergência ou urgência, independente de sua localização. A SE é uma extensão da Síntese que inclui imagens e arquivos recentes de maior volume. O PC é o que contém o maior volume de dados, pois armazena as informações sobre a saúde do paciente ao longo de sua vida, entretanto, grande parte das informações do PC tem pouca relevância em atendimentos. Por isso ele raramente deve ser acessado, nesta concepção.

Desta maneira, a síntese deve ser armazenada em BD distribuído, de modo que, as informações de relevância médica estejam sempre disponíveis para acesso. A relevância médica é definida pelos profissionais de saúde e pode ter caráter permanente ou temporário. Permanente são as informações que acompanham o paciente pela vida toda, como tipo sanguíneo, alergias a medicamentos etc. Temporárias são as informações que afetam o estado de saúde do paciente seja ele recente e/ou dos últimos atendimentos.

Os três níveis de informação foram implementados e distribuídos em duas instâncias no servidor de BD MySQL 5.x e possuem o mesmo Modelo de Entidade-relacionamento (MER). A aplicação PEUP foi desenvolvida em JAVA com tecnologia WEB Servlets e foi implantada no servidor de aplicação GlassFish 4. Para a distribuição da aplicação e do BD, utilizou-se a tecnologia JAVA *Parallel Processing Framework* (JPPF) que é uma grade que oferece multiprocessamento e escalabilidade de requisições do PEUP, conforme apresenta a Figura 2. O JPPF suportou todas as requisições HTTP que chegaram ao servidor de aplicação GlassFish 4. Para manter os dados do PEUP, foi implantado o servidor de BD MySQL 5.x com replicação de dados *Master/Master* entre todas as instâncias de BD da grade. Assim, o GlassFish, direcionou as requisições ao servidor JPPF Driver que gerencia os JPPF Nodes, responsáveis pela execução das instruções SQL no MySQL Server. O algoritmo de balanceamento de carga que o JPPF Driver adota por padrão é o “proporcional”, que entrega as tarefas aos JPPF Nodes (SHAMS et al., 2010; XIONG et al., 2010; DEMCHENKO et al., 2012; KURZE et al., 2011).

Figura 2 - Arquitetura de federações da grade JPPF  
XIONG et al., 2010



Conforme a Tabela 1, foram utilizados 35 computadores desktops com *virtual machines* (VM) do sistema operacional Debian 9.5 disponíveis no laboratório de informática da Universidade Nove de Julho. Assim, considera-se que o desempenho

não está próximo do ideal tendo em vista a configuração do hardware de simulação.

As métricas utilizadas para avaliar a arquitetura do PEUP são: (a) desempenho, (b) disponibilidade, (c) confiabilidade e (d) tolerância a falhas. Em (a) o TR foi avaliado em diferentes configurações do ambiente JPPF e para avaliar o desempenho na transferência de dados. Nesse cenário, apesar de contarmos com 31 JPPF Nodes, consideramos um número reduzido de até 4 JPPF Nodes para delimitarmos os testes de (a).

Tabela 1 - Componentes do PEUP no laboratório de informática  
Do Autor

Servidores	Função Do Ambiente	Qtde Recursos
GlassFish Server	Servidor de Aplicação	01
JPPF Driver Server	Nó Mestre (grade)	01
JPPF Node Server	Nó Escravo (grade)	31
JPPF Monitoring Admin	Admin (grade)	01
MySQL Server	Servidor de BD	02

Em (b), (c) e (d) foram aplicados seis cenários com a simulação de falha nos JPPF Nodes, como se pode verificar na Tabela 2. Também consideramos a avaliação do TR para analisar o comportamento do ambiente computacional e o retardo médio diante das situações de falha de recursos.

As requisições enviadas aos seis cenários foram: *Listar Atendimentos*, *Listar Pacientes* e *Listar Usuários*, previamente desenvolvidas para validar o PEUP. Assim, através da simulação da falha de alguns nós de processamento e, conforme a quantidade de recursos foi reduzindo (Tabela 2), analisou-se a capacidade do PEUP em manter a disponibilidade, confiabilidade e tolerância a falhas.

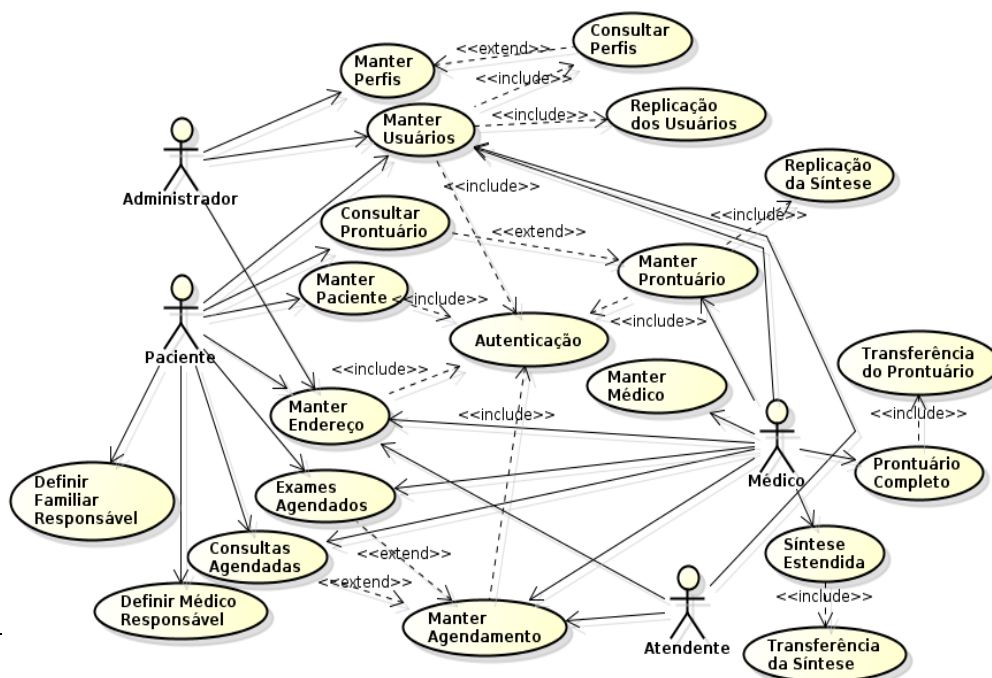
Tabela 2 - Quantidade de JPPF Nodes do PEUP e os testes realizados com a simulação de falha de nós  
Do Autor

Teste Nº	Descrição	Nós Inativos	Nós Ativos
0	Sem falha	0	31
1	Falha de nós	15	16
2	Falha de nós	12	4
3	Falha de nós	1	3
4	Falha de nós	1	2
5	Sem falha	0	1

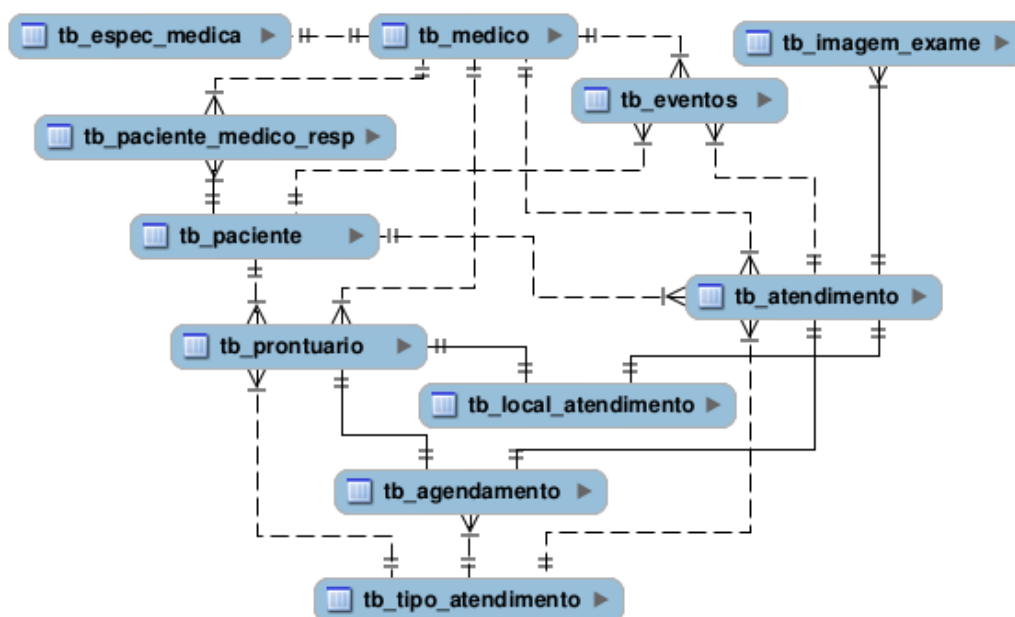
O ambiente de teste montado para esta pesquisa foi implementado em laboratório e com a utilização de VMs e, portanto, não corresponde a uma escala real, com servidores dotados de grande poder de processamento e hardware dedicado. Dessa forma, o volume de dados utilizado nos testes de desempenho foi estimado como proporcional a um ambiente real de bom desempenho e compatível com a tecnologia atual.

Os casos de uso apresentados na Figura 3 são representações dos requisitos funcionais e dos usuários, que permitem o entendimento dos eventos realizados por usuários do PEUP. Assim, para os testes foram especificados 19 casos de uso e 4 atores conforme o diagrama de caso de uso da Figura 3. O MER representado pela Figura 4 contém as entidades necessárias ao BD para o controle dos dados armazenados e configurações do PEUP. No que se refere ao modelo proposto para os três níveis de informação, os testes foram realizados para manter a Síntese na tabela *tb\_atendimento*, a SE e o PC na tabela *tb\_prontuario*. Também foram realizados os testes de replicação de atendimentos gravados na Síntese que são replicados para outra instância do BD da grade. Foram realizados os testes da *trigger* que faz a transferência dos dados de atendimentos da *tb\_atendimento* para a *tb\_prontuario*, mantendo os registros por tempo indeterminado. Os dados contidos nos testes de replicação *master/master* são fictícios, mas foram carregados no BD com base em padrões médicos de atendimento. Este modelo foi extraído do MySQL Workbench 6 e implementado no MySQL 5.

Figura 3 - DCU como resultado dos requisitos do PEUP



Do Autor  
 Figura 4 - Resultado do MER baseado nos requisitos do PEUP  
 Do Autor



Os resultados apresentados na Tabela 3 foram direcionados apenas à métrica do TR para avaliar (a), onde foram utilizados um ambiente computacional com configurações contendo de 5 a 8 servidores da grade JPPF para executar paralelamente as operações de Cadastrar Pacientes, Listar Pacientes, Consultar Paciente e Consultar Usuário.

Tabela 3 - A configuração da grade em 4 cenários na execução simultânea de tarefas e os TRs

Do Autor

Servidores:	GlassFish	Driver	Node	Admin	MySQL	TR
Quantidade	1	1	1	1	1	12,1s
Servidores:	1	1	2	1	1	8,6s
	1	2	3	1	1	6,2s
	1	1	4	1	1	4,0s

Os resultados apresentados na Tabela 4 mostram o TR entre os servidores MySQL distribuídos em duas regiões geográficas distintas, aqui representadas pelos endereços IPs. O primeiro servidor MySQL foi iniciado em *localhost* e o outro servidor em VM. Os testes foram feitos por meio de consultas iniciando com 1.000 atendimentos de pacientes. Quando estas quantidades foram aumentando para

210.960, 1.186.959 e 1.716.816 atendimentos, não houve impacto significativo no TR das consultas, conforme mostra a Tabela 4.

Tabela 4 - Comparativo de consultas de atendimento através da ferramenta MySQL Workbench  
Do Autor

Consulta de Registros do Atendimento				
MySQL (IP)	Número de Registros			
	1.000	210.960	1.186.959	1.716.816
192.168.42.122	0,002s	2.088s*	4,746s	6,456s
192.168.42.150	0,011s	7.101s	9,758s	11,710s

\*segundos

O TR da transferência da SE entre os servidores GlassFish, JPPF e MySQL entre duas regiões distantes (de 192.168.42.150 para 192.168.42.122), foi de 4,168s para 100 registros de atendimento por paciente. Para a transferência do PC considerando o mesmo cenário da transferência da SE, esta ocorreu com um volume maior de registros de atendimentos totalizando 2.217 com TR de 17,562s.

Verificou-se a integridade de armazenamento em todas as condições operacionais do modelo. Não foram constatadas falhas de integridade nos dados. Nos testes de replicação foram suficientes para garantir a integridade das instâncias.

A Tabela 5 apresenta os cenários propostos nos testes realizados para avaliar a TR das métricas (b), (c) e (d).

Tabela 5 - Resultados dos Tempos de Resposta em seis cenários de testes  
Do Autor

Cenários	Nós Disponíveis	TR(s*)
0	31	4,55
1	16	4,83
2	4	4,02
3	3	6,24
4	2	8,60
5	1	12,10

\*segundos

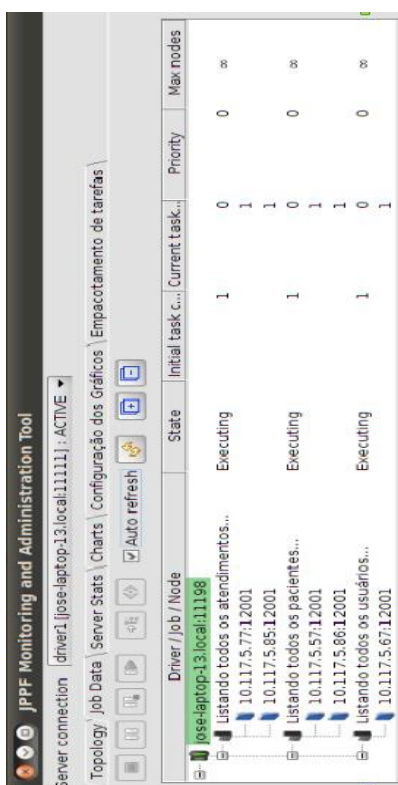
No cenário 0, as requisições “Listar Atendimentos”, “Listar Pacientes” e “Listar Usuários” foram enviadas para o servidor Glass Fish e, conseqüentemente, para o JPPF Driver. Observou-se que o algoritmo proporcional faz o balanceamento de carga de tarefas entre os JPPF Nodes, distribuindo-os de modo adequado e deixando os demais nós da grade (28) ociosos. Estes recursos ociosos poderiam ser acionados, de

acordo com o aumento da demanda e escalabilidade necessária.

Os demais cenários consideraram um aumento na falha de recursos e observou-se a capacidade da grade computacional de sobreviver a um número de falhas de recursos e se manter disponível. Desta forma, observa-se na Figura 5(1) que, mesmo com a redução de 30 JPPF Nodes, o PEUP continuou atendendo as requisições dos usuários, apesar do aumento no TR apresentado na Tabela 5 e a fila de requisições da Figura 5(2). Além disso, verificou-se que alguns JPPF Nodes se mantiveram ociosos, sendo esta, uma característica de escalabilidade, que economiza recursos quando a demanda acessos de usuários é baixa. Inversamente, seriam utilizados nós ociosos aumentando os recursos de processamento.

A proposta de uma arquitetura em grade para o PEUP, visando avaliar a disponibilidade, confiabilidade e a tolerância a falhas é tecnicamente viável de implementar, mesmo considerando um BD distribuído e um MER com três níveis de informação. Assim, a adoção de um número sempre maior de JPPF Nodes que a demanda de acessos de usuários do PEUP requer, é uma estratégia interessante do ponto de vista do desempenho, pois haverá sempre mais nós disponíveis que o número de requisições. Além disso, foi identificado que a grade economiza recursos quando opera em baixa. Com relação à falha de seus componentes, o ambiente do PEUP se mostrou bem tolerante a falha e manteve a disponibilidade do sistema aos usuários simulados.

Figura 5 – Telas do JPPF Admin. (1)Execução de tarefas entre os nós. (2)Execução de um nó para simular a fila e a disponibilidade de recursos  
Do Autor



(1)



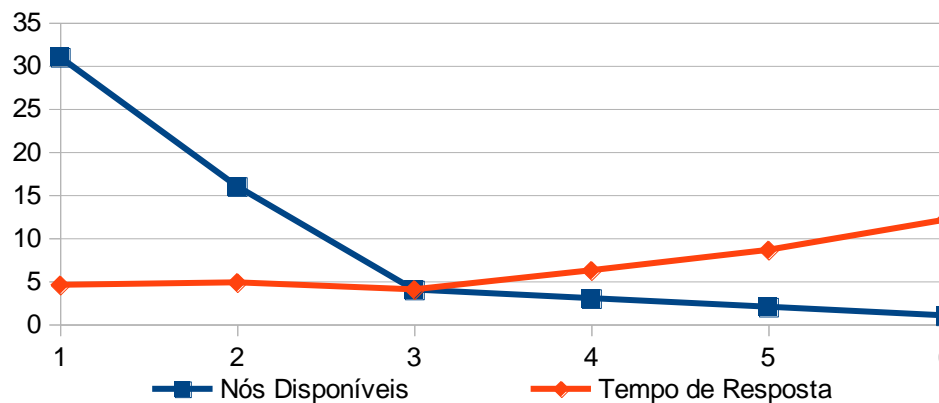
(2)

Nos resultados dos testes de (a), foi avaliado o TR em quatro configurações contendo de 5 a 8 servidores. Os cenários que foram modificados de 5 para 6 servidores, houve uma redução de 29% no TR. Nos cenários que foram modificados de 5 para 8 servidores, houve uma redução de 49% (com dois servidores Driver JPPF) a 67% (com um servidor Driver JPPF) no TR. A transferência da síntese e do *login* do usuário, além de serem pequenos volumes de dados, são transferidos em tempo real a partir do método de replicação do MySQL, mantendo a integridade e disponibilizando o acesso ao paciente ou ao profissional de saúde independentemente da sua localização. Nestes casos, a transferência ocorre na casa de poucos milissegundos. A transferência da SE baseada em 100 registros de atendimentos por paciente se mostrou eficiente sem comprometer o TR dos servidores, mesmo considerando que o ambiente de teste não foi realizado em servidores com *hardwares* dedicados. Mesmo assim, os resultados mostraram um bom desempenho em torno de 4s para a execução completa da transação no BD. A transferência do PC do paciente baseada em 2.217 registros de atendimentos também respondeu com um tempo relativamente baixo (17s) mesmo considerando que este ambiente de teste também não foi realizado em *hardwares* servidores. De qualquer maneira, o aumento de requisições desta natureza pode afetar o desempenho do PEUP e, por este motivo, que tais solicitações devem

ser executadas fora do horário de atendimentos. Caso ocorra um aumento das requisições de transferências de atendimentos tanto da SE quanto do PC, o número de servidores de BD e nós JPPF pode ser ampliado para garantir a escalabilidade .

A Figura 6 apresenta um gráfico que inicializa a discussão dos resultados de (b), (c) e (d) com um comparativo da variação dos tempos de resposta conforme os testes avançam no sentido de simular a indisponibilidade de nós de processamento da grade. Assim, observa-se que o TR não aumentou de forma significativa. Isto é, o TR foi de 6,24s quando o cenário com três requisições chegara aos três nós e 8,60s para dois nós.

Figura 6 - Comparativo da variação dos TRs para os cenários apresentados Do Autor



Pode-se observar através da Tabela 6 que, pelo valor expresso no TR para os com os JPPF Nodes disponíveis, que a curtose nesse caso é menor do que zero, ou seja, é tida como platicúrtica (curtoses < 0), indicando que há pouca concentração dos dados em torno da média, fazendo com que a curva fique mais achatada. Isso nos indica que necessitamos realizar maior quantidade de testes em ambientes diversos, buscando aperfeiçoar o tempo em relação aos nós disponíveis, diminuindo a dispersão entre os dados.

Tabela 6 - Relação da Curtose Do Autor

Variável	N (Testes)	Curtose
TR	6	0,78
Nós Disponíveis	6	1,72

Na Tabela 7, pode-se verificar por intermédio do desvio-padrão em que a

interpretação dada é a seguinte: quanto mais próximo de zero, menos os dados medidos (3,11s) variam em torno da média (6,72s). Entre outras conclusões, pode-se dizer que ao menos 75% dos JPPF Nodes possuem um TR aceitável, dentro do ambiente em que ocorreram os testes. O coeficiente de variação (média/desvio-padrão) é de aproximadamente 45% de dispersão relativa. Com o aumento de testes será possível encontrar outras possíveis falhas e conseqüentemente melhorar o TR.

Tabela 7 - Desvio-Padrão dos Testes

Do Autor

Variável	N (Testes)	Média	Desvio Padrão
TR	6	6,72	3,11

Com base na tabela 7 o teste realizado possui um coeficiente de variação (CV) de 46,28%. Este valor é muito alto e representa grande dispersão dos dados, tornando os resultados não muito confiáveis, sendo necessário fazer outros testes futuros para comparar os dados.

O ambiente de validação utilizado para o modelo do PEUP apresentou um comportamento interessante, referente à economia de recursos de processamento, o que garante o funcionamento do modelo, ou seja, todas as tarefas enviadas foram prontamente atendidas. Esta flexibilidade permite manter o ambiente do modelo do PEUP com alta disponibilidade e confiabilidade de serviço.

## Conclusão

Com o aumento do número de servidores JPPF, foi verificado uma redução de até 67% no TR contribuindo para a redução da latência na rede, fato importante para a disponibilidade e desempenho do PEUP.

A disponibilidade dos registros mantidos na Síntese, garante o acesso às informações relevantes do prontuário do paciente em qualquer BD da grade computacional.

As transferências da SE e do PC se mostraram eficientes mesmo com um volume maior de dados, pois ocorreram em 4 e 17s, respectivamente. Assim, em poucos segundos, os registros do paciente estarão disponíveis para acesso, mesmo fora da base local do mesmo. A manutenção da Síntese, da SE e do PC são viáveis a partir dos métodos implementados em todos os BDs da grade.

Assim, o modelo do PEUP se mostrou viável para a implantação e na manutenção de dados no ambiente JPPF baseada na distribuição regional de servidores de BD. Mesmo que a transferência do PC não tenha comprometido o desempenho do PEUP, deve-se considerar que esta demanda seja atendida fora do horário de pico de atendimentos. A segmentação do PEUP em três níveis de informação diminui significativamente o TR e viabiliza sua operacionalização. No modelo proposto, o volume de acessos é inversamente proporcional ao volume de dados.

A falha de JPPF Nodes, que variaram entre 48 a 97% em seis cenários, a arquitetura em grade do PEUP se mostrou confiável e disponível aos usuários, mesmo lidando com um modelo de BD distribuído em três níveis de informação. Além da confiabilidade e disponibilidade, a arquitetura se mostrou sólida em relação à tolerância a falhas, que é fator importante em ambientes que sofrem com a concorrência de acesso.

Pode-se observar que, com a falha de 30 JPPF Nodes, um único nó respondeu as requisições dos usuários simulados, verificando que o desempenho do PEUP tenha diminuído, mesmo operando em rede local. Portanto, considerando as situações de falha e manutenção corretiva e/ou preventiva, a arquitetura de grade do PEUP se mostrou viável para lidar com desafios de confiabilidade, disponibilidade e tolerância a falhas. Tais questões estão cada vez mais presentes em aplicações na área da saúde, sejam eles, disponíveis através da nuvem ou em redes privadas de longa distância.

## **Referências Bibliográficas**

COOREVITS, P. et al. Electronic health records: new opportunities for clinical research. *Journal of internal medicine*, v. 274, n. 6, 2013, pp. 547-560.

DEMCHENKO, Y. et al. (2012) "Intercloud Architecture for interoperability and integration". In: *IEEE International Conference on Cloud Computing Technology and Science*, 2012, pp. 666-674.

FURUIE, S.S. et al. Prontuário eletrônico de pacientes: Integrando informações clínicas e imagens médicas. *Revista Brasileira de Engenharia Biomédica*; 2003 19(3), pp. 125-137.

GAZZARATA, R., GIANNINI, B., GIACOMINI, M.A. SOA-based platform to support clinical data sharing. *Journal of Healthcare Engineering*, v. 2017, 2017.

HERSH, W.R. et al. Caveats for the use of operational electronic health record data in comparative effectiveness research. *Medical care*, v. 51, n. 8 0 3, 2013, pp. S30.

HMEIDI, I. et al. Survey of Cloud Computing Web Services for Healthcare Information Retrieval Systems. In: *The International Conference on Computing Technology and Information Management (ICCTIM)*. Society of Digital Information and Wireless Communication, 2014. pp. 291.

KING, J. et al. Clinical benefits of electronic health record use: national findings. *Health services research*, v. 49, n. 1pt2, 2014, pp. 392-404.

KURZE, T. et al. (2011) Cloud Federation. In: *The Second International Conference on Cloud Computing, Grids and Virtualization*, 2011, pp. 32-38.

LUNARDELLI, R.S.A., TONELLO, I.M.S, MOLINA, L. G. (2015). A Constituição da Memória dos Procedimentos em Saúde no Contexto do Prontuário Eletrônico do Paciente. *Informação & Informação*, 19(3), 2015, pp. 107-124.

SHAMS, R. et al. A survey of medical image registration on multicore and the GPU. *IEEE Signal Processing Magazine*, v. 27, n. 2, 2010, pp. 50-60.

SILVA, B.M.C. et al. Mobile-health: A review of current state in 2015. *Journal of biomedical informatics*, v. 56, 2015, pp. 265-272.

SOARES CRUZ, J.A., FLORES, D., CORREA GARCIA, O.M. O Prontuário Eletrônico de Paciente (PEP) como memória, patrimônio documental e cultural. *Em Questão*, v. 17, n. 2, 2011.

XIONG, J., WANG, J., XU J. (2010) Research of Distributed Parallel Information Retrieval Based on JPPF. In: *International Conference of Information Science and Management Engineering ISME*, 1:109-111; DOI:10.1109/ISME, 2010.