

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL
FACULDADE DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

**UTILIZANDO GRADES COMPUTACIONAIS
NO ATENDIMENTO DE REQUISITOS DE
E-GOV**

EDSON SHIN-ITI KOMATSU

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Ciência da Computação na Pontifícia Universidade Católica do Rio Grande do Sul.

Orientador: Prof. Dr. César Augusto Fonticelha De Rose

Porto Alegre
2009

Dados Internacionais de Catalogação na Publicação (CIP)

K81u Komatsu, Edson Shin-Iti

Utilizando grades computacionais no atendimento de requisitos de e-gov / Edson Shin-Iti Komatsu. – Porto Alegre, 2009.

110 p.

Diss. (Mestrado) – Fac. de Informática, PUCRS.

Orientador: Prof. Dr. César Augusto FonticIELha De Rose.

1. Tecnologia da Informação. 2. Grades Computacionais.
3. Administração Pública – Processamento de Dados.
4. Informação Tecnológica no Governo. 5. Processamento Distribuído. I. De Rose, César Augusto FonticIELha.
II. Título.

CDD 004.36

**Ficha Catalográfica elaborada pelo
Setor de Tratamento da Informação da BC-PUCRS**



TERMO DE APRESENTAÇÃO DE DISSERTAÇÃO DE MESTRADO

Dissertação intitulada **"Utilizando Grades Computacionais no Atendimento de Requisitos de e-Gov"**, apresentada por Edson Shin-Iti Komatsu, como parte dos requisitos para obtenção do grau de Mestre em Ciência da Computação, Processamento Paralelo e Distribuído, aprovada em 06/07/09 pela Comissão Examinadora:

Prof. Dr. César Augusto FonticIELha De Rose -
Orientador

PPGCC/PUCRS

Prof. Dr. Luiz Gustavo Leão Fernandes -

PPGCC/PUCRS

Prof. Dr. Francisco Vilar Brasileiro -

UFCG

Homologada em 15/09/09, conforme Ata No. 16/09 pela Comissão Coordenadora.

Prof. Dr. Fernando Gehm Moraes
Coordenador.

PUCRS

Campus Central

Av. Ipiranga, 6681 - P32 - sala 507 - CEP: 90619-900
Fone: (51) 3320-3611 - Fax (51) 3320-3621
E-mail: ppgcc@pucrs.br
www.pucrs.br/facin/pos

Dedico essa conquista a você,
minha amiga, mãe dedicada,
lutadora, companheira dos bons e
maus momentos, minha esposa
amada Regina, a quem amo e
admiro muito. À minha Filha Katie.
Aos meus pais, Tsutomu e Miyabi
Komatsu

AGRADECIMENTOS

Ao meu orientador, Professor De Rose que me acolheu e dedicou parte de seu tempo nesta minha caminhada, me orientando e corrigindo meus passos.

Aos meus amigos Cirano e Jeronimo, pelas noites e madrugadas mal dormidas, viagens e mais viagens, grato pela amizade, companheirismo e apoio nesta jornada.

Ao Rocine obrigado pela presença, pelos conselhos e pela amizade. Aos colegas da Dataprev, agradeço pela força e à Dataprev pela oportunidade proporcionada pelo Programa de Incentivo à Pós-Graduação.

Aos colegas do Minter que me ajudaram a chegar até aqui. Aos colegas de Campina Grande pelo suporte na utilização do Ourgrid. Aos amigos e amigas pelo apoio nas horas difíceis.

Por fim, Obrigado Regina, por sua paciência, força, crença e confiança.

UTILIZANDO GRADES COMPUTACIONAIS NO ATENDIMENTO DE REQUISITOS DE E-GOV

RESUMO

O Governo Eletrônico (e-Gov) deve servir de base para a gestão de Tecnologia da Informação e Comunicação do Governo Federal e tem como requisitos fundamentais a promoção da cidadania, a priorização da inclusão digital, o uso da gestão do conhecimento, a racionalização de recursos, a padronização de normas, políticas e padrões e a integração com os demais entes federativos. As grades computacionais, por sua vez, fornecem um ambiente para execução de aplicações paralelas no qual recursos distribuídos podem ser utilizados de forma transparente. Este ambiente possibilita o processamento de grandes volumes de dados, compartilhamento de recursos e a redução de custos. Dessa forma, as grades computacionais podem ajudar os órgãos públicos a alcançar seus objetivos. Neste estudo os requisitos do Governo Eletrônico são analisados sob a perspectiva de utilização de grades computacionais para execução aplicações de e-Gov, verificando como as grades podem ser úteis aos órgãos governamentais. Uma aplicação de e-Gov é modelada e testada em ambiente de grades para mostrar o potencial desta plataforma de execução em tais condições.

Palavras Chave: Grades Computacionais, Governo Eletrônico, Aplicações de E-Gov

USING COMPUTATIONAL GRIDS TO FULFILL E-GOV REQUIREMENTS

ABSTRACT

Electronic Government (e-Gov) should serve as the basis for the management of Information Technology and Communication of the Federal Government. It has as basic requirements the promotion of citizenship and digital inclusion, the use of knowledge management, the rationalization of resources, standardization, the use of policies and standards as well as the integration with federal organisms. Computational grids provide an environment for execution of parallel applications in which distributed resources can be used in a transparent way. They enable the processing of large amounts of data, resource sharing and cost reduction. This study analyzes how computational grids could be used to assist public institutions to achieve their e-Gov goals. An e-Gov prototype application is modeled and tested in a grid environment to show the potential of this execution platform in such conditions.

Keywords: Computational Grids, Electronic Government, e-Gov Applications

LISTA DE FIGURAS

Figura 1 - Arquitetura do Sistema Prisma	38
Figura 2 - Classificação de Problemas [GOR08].....	49
Figura 3 - Proposta de Arquitetura para a Qualificação de Dados do CNIS.....	63
Figura 4 - Proposta de Arquitetura para o Sistema Único de Benefícios.....	64
Figura 5 - Proposta de Arquitetura para o Sistema Prisma	67
Figura 6 - Proposta de Arquitetura para o Batimento do Cadastro Eleitoral.....	69
Figura 7 - Fluxograma do processo de ETL do MAIPREV	75
Figura 8 - Proposta para processo de ETL do MAIPREV executando em grades	77
Figura 9 - Uso dos recursos da grade na primeira solução	86
Figura 10 - Utilização da grade na Terceira Solução	90
Figura 11 - Utilização da grade na quarta solução.....	92

LISTA DE TABELAS

Tabela 1 – Dados úteis para a agregação escolhida	79
Tabela 2 – Taxa de compactação do arquivo texto.....	79
Tabela 3 - Tempo médio de execução na primeira solução	84
Tabela 4 - Distribuição de tempo no processamento da aplicação sequencial.....	85
Tabela 5 - Tempo de processamento da aplicação distribuída na primeira solução.....	85
Tabela 6 - Tempo médio de utilização da grade por fase na primeira solução.....	85
Tabela 7 - Tempo médio de execução na segunda solução.....	88
Tabela 8 - Tempo médio de execução na segunda solução passo a passo	88
Tabela 9 - Tempo médio de utilização da grade por fase na Segunda Solução	89
Tabela 10 – Tempo médio de execução na terceira solução.....	89
Tabela 11 - Tempo médio de utilização da grade por fase na terceira solução	90
Tabela 12 - Detalhamento do tempo de execução da aplicação na terceira solução	90
Tabela 13 - Tempo médio de execução na quarta solução	91
Tabela 14 - Tempo médio de utilização da grade por fase na quarta solução	92
Tabela 15 - Comparativo dos tempos de execução das soluções	93
Tabela 16 - Relação entre tempo total de execução e tempo para separação, compactação de dados e submissão de trabalhos	94
Tabela 17 - Comparativo de execução da aplicação em máquinas diferentes	95
Tabela 18 - Comparativo entre Quarta Solução e Solução com Base Distribuída	97
Tabela 19 - Estimando tempo para consolidação da Maciça	98

LISTA DE SIGLAS

APE - Assessoria de Pesquisas Estratégicas
APS - Agência da Previdência Social
BoT - Bag-of-Tasks
CACIC - Configurador Automático e Coletor de Informações Computacionais
CEF - Caixa Econômica Federal
CNIS - Cadastro Nacional de Informações Sociais
CNT- Cadastro Nacional do Trabalhador
COCAR - Controlador Centralizado de Ambiente de Rede
COMPREV - Compensação Previdenciária
CPDF - Centro de Processamento do Distrito Federal
CPF - Cadastro de Pessoa Física
CPRJ - Centro de Processamento do Rio de Janeiro
CPSP - Centro de Processamento de São Paulo
DOE - Department of Energy
EGEE - Enabling Grids for E-science
e-Gov - Governo Eletrônico
e-MAG - Modelo de Acessibilidade de Governo Eletrônico
e-PING - Padrões de Interoperabilidade de Governo Eletrônico
e-PMG - Padrão de Metadados do Governo Eletrônico
ETL - Extração Transformação Carga
HipNet - Homologação das Informações da Previdência
HISCRE - Histórico de Créditos dos Beneficiários
HPC - High-performance computing
IANA - Internet Assigned Numbers Authority
IBGE - Instituto Brasileiro de Geografia e Estatística
INSS - Instituto Nacional do Seguro Social
ITI/PR - Instituto Nacional de Tecnologia da Informação / Presidência da República
LHC – Large Hadron Collider
LSD - Laboratório de Sistemas Distribuídos
MPAS - Ministério da Previdência e Assistência Social
MPS - Ministério da Previdência Social
MTb - Ministério do Trabalho

NSF - National Science Foundation
OGF - Open Grid Forum
OJAL - OurGrid Job Abstraction Layer
ONID - Observatório Nacional de Inclusão Digital
OSG - Open Science Grid
PAB - Pagamento Alternativo de Benefícios
QoS - Qualidade de Serviço
RFB - Receita Federal do Brasil
SABI – Sistema de Administração de Benefícios por Incapacidade
SAGUI - Sistema de Apoio à Gerência Unificada de Informações
SASL - Simple Authentication and Security Layer
SGBD - Sistema Gerenciador de Banco de Dados
SGD - Sistema de Gestão de Demandas
SIAFI - Sistema Integrado de Administração Financeira
SLTI - Secretaria de Logística e Tecnologia da Informação
SOA - Arquitetura Orientada à Serviços
STN - Secretaria do Tesouro Nacional
TI - Tecnologia da Informação
TIC - Tecnologia da Informação e Comunicação
TLS - Transport Layer Security
TSE - Tribunal Superior Eleitoral
UFCG - Universidade Federal de Campina Grande
VDT - Virtual Data Toolkit
VPN – Virtual Private Network
XML - eXtensible Markup Language
XMPP - Extensible Messaging and Presence Protocol

SUMÁRIO

1.	INTRODUÇÃO	23
1.1.	Objetivos	24
1.2.	Organização do Texto	24
2.	CARACTERIZANDO DEMANDAS DE GOVERNO ELETRÔNICO	25
2.1.	O Governo Eletrônico	26
2.2.	Requisitos Básicos do Governo Eletrônico	27
2.2.1.	A Prioridade é a Promoção da Cidadania	27
2.2.2.	Indissociabilidade entre inclusão digital e o Governo Eletrônico	28
2.2.2.1.	O Portal.....	28
2.2.2.2.	e-MAG: Modelo de Acessibilidade de Governo Eletrônico	29
2.2.3.	Gestão do Conhecimento como instrumento estratégico de articulação e gestão das políticas públicas	29
2.2.4.	Adoção de políticas, normas e padrões comuns	29
2.2.4.1.	e-PING: Padrões de Interoperabilidade de Governo Eletrônico	30
2.2.5.	Utilização do <i>software</i> livre como recurso estratégico	31
2.2.5.1.	O Guia Livre - Referência de Migração para <i>Software</i> Livre	32
2.2.5.2.	O Guia Cluster	32
2.2.5.3.	Vantagens na adoção da tecnologia de grades.....	33
2.2.6.	Racionalização de Recursos	34
2.2.7.	Integração com outros níveis de governo e com os demais poderes	34
2.3.	Exemplos de Aplicações de Governo Eletrônico	35
2.3.1.	O Sistema de Cadastro Nacional de Informações Sociais	35
2.3.2.	O Sistema Único de Benefícios.....	36
2.3.3.	O Sistema PRISMA	37
2.3.4.	A Base de Dados da Justiça Eleitoral	38
2.3.5.	O Sistema da Assessoria de Pesquisas Estratégicas	39
2.3.6.	O Sistema Integrado de Administração Financeira do Governo Federal	41
2.4.	Considerações Finais sobre Sistemas de e-Gov	41
3.	GRADES COMPUTACIONAIS.....	43
3.1.	Conceituando Grades	44
3.2.	Tipos de Grades	45
3.3.	Aplicações.....	45
3.4.	Classificação de Aplicações	48
3.5.	Middleware para Grades Computacionais.....	49
3.5.1.	Globus Toolkit.....	49
3.5.2.	Condor	50
3.5.3.	OurGrid	51
3.5.4.	Legion	52
3.6.	Exemplos de Aplicações para Grades Computacionais.....	53
3.6.1.	A Comunidade Ourgrid.....	53
3.6.1.1.	GerpavGrid	53

3.6.1.2.	SegHidro.....	54
3.6.2.	Open Science Grid (OSG).....	54
3.6.2.1.	Reprocessamento de dados para DZero.....	55
3.6.3.	Enabling Grids for E-science (EGEE).....	56
3.7.	Considerações Finais sobre Grades Computacionais.....	56
4.	GRADES ATENDENDO E-GOV	59
4.1.	Grades Atendendo os Requisitos de e-Gov	59
4.1.1.	Adoção de Políticas, Normas e Padrões Comuns.....	59
4.1.2.	Utilização do <i>Software</i> Livre como Recurso Estratégico	60
4.1.3.	Racionalização dos Recursos	61
4.2.	Utilizando grades em Aplicações de E-Gov	61
4.2.1.	Qualificação de Informações do CNIS	62
4.2.2.	Processamento dos dados do Sistema Único de Benefícios (SUB)	64
4.2.3.	PRISMA	65
4.2.4.	Justiça Eleitoral - Análise da Base de Eleitores.....	68
4.3.	Considerações finais sobre Atendimento aos Requisitos de e-Gov	70
5.	ESTUDO DE CASO.....	73
5.1.	O Processo de ETL do Sistema MAIPREV.....	73
5.2.	Descrevendo o Problema	74
5.3.	Proposta de Modelagem para o Sistema MAIPREV	76
5.4.	Implementação de uma Rotina do Processo de ETL	77
5.4.1.	Infra-estrutura da Grade Utilizada	78
5.4.2.	A Primeira Solução.....	79
5.4.2.1.	Programa Principal	80
5.4.2.2.	Programa SEPARAOL	81
5.4.2.3.	Escrevendo um <i>JOB</i>	81
5.4.2.4.	Programa LOCAL	82
5.4.2.5.	Programa CONSOLIDA.....	83
5.4.2.6.	Validando a Primeira Solução	84
5.4.3.	A Segunda Solução.....	86
5.4.4.	A Terceira Solução	89
5.4.5.	A Quarta Solução	91
5.4.6.	Analisando os Resultados Obtidos	93
5.4.6.1.	Comparando o Tempo da Solução Distribuída com o Tempo Real da Aplicação	97
5.4.7.	Aspectos de Segurança	98
6.	CONCLUSÃO E TRABALHOS FUTUROS.....	101
	REFERÊNCIAS	105

1. INTRODUÇÃO

As políticas e a atuação do Governo Federal impactam na vida de toda a população brasileira, seja na economia, na educação, na segurança, no transporte, etc.

Segundo o sítio do Governo Eletrônico [MPO07a] no ano 2000 o Governo Brasileiro lançou as bases para o Governo Eletrônico (e-Gov) com a criação do Comitê Executivo do Governo Eletrônico. Este comitê tinha o objetivo de formular políticas, estabelecer diretrizes, coordenar e articular as ações de implantação do Governo Eletrônico, voltado à prestação de serviços e informações aos cidadãos.

As políticas de e-Gov no Brasil se baseiam em diretrizes que atuam junto ao cidadão, na melhoria da gestão interna e na integração com parceiros e fornecedores. De acordo com o sítio do Governo Eletrônico [MPO07d] as diretrizes do Governo Eletrônico devem servir como referência geral para estruturar as estratégias das ações do Governo Eletrônico. As diretrizes são: a promoção da cidadania como prioridade, a indissociabilidade da inclusão digital, o uso de *software* livre¹, a gestão do conhecimento, a racionalização de recursos, a padronização de normas, políticas e padrões e a integração com os demais entes federativos.

O objetivo das grades computacionais é fornecer uma infra-estrutura que garanta o acesso a recursos distribuídos geograficamente de forma confiável, consistente e persistente. A lógica é que agrupando máquinas de diferentes sítios forneceria poder computacional semelhante àqueles oferecidos pelos supercomputadores [DER06].

As redes de comunicação do Governo Federal são compostas de milhares de computadores cujos recursos podem ser melhor aproveitados. As grades computacionais fornecem às aplicações um ambiente de execução distribuída possibilitando a utilização de recursos computacionais ociosos nos ambientes de Tecnologia da Informação e Comunicação (TIC) do Governo Federal. A grade sendo uma infra-estrutura distribuída para a execução de aplicações paralelas sob demanda possibilita que o Governo Federal empregue esta tecnologia no processamento de várias aplicações de e-Gov otimizando seus recursos. Sendo assim, empregando tecnologias de grades para execução de aplicações de e-Gov é possível atender aos requisitos de e-Gov.

¹ *Software* disponível com a permissão para qualquer um usá-lo, copiá-lo, e distribuí-lo, seja na sua forma original ou com modificações, seja gratuitamente ou com custo. Em especial, a possibilidade de modificações implica em que o código fonte esteja disponível.[GFB07b]

1.1. Objetivos

O objetivo deste trabalho é realizar um estudo sobre as grades computacionais e os requisitos de Governo Eletrônico analisando se estes requisitos podem ser atendidos pelo uso de grades computacionais para executar aplicações paralelas.

Neste sentido, serão verificadas como as aplicações de e-Gov podem se aproveitar dos recursos das grades computacionais visando atender aos requisitos do Governo Eletrônico e serão apresentadas propostas de modelagens para aplicações de e-Gov em grades computacionais.

Por fim, serão desenvolvidas as funcionalidades necessárias de um protótipo para uma aplicação de e-Gov, avaliando o desempenho da aplicação distribuída e a executada de forma seqüencial.

1.2. Organização do Texto

No capítulo 2 o Governo Eletrônico é apresentado descrevendo seu histórico, a situação atual, os documentos associados e seus requisitos.

No capítulo 3 são apresentados conceitos relacionados a grades computacionais e exemplos de aplicações que fazem uso de grades computacionais para o processamento dos dados.

No capítulo 4 os requisitos básicos do Governo Eletrônico são analisados sob a perspectiva de utilização de grades computacionais para processamento de aplicações de Governo Eletrônico e mostradas modelagens teóricas de aplicações de e-Gov em grades.

No Capítulo 5 a viabilidade de se atingir os objetivos do Governo Eletrônico é verificada através de um estudo de caso onde uma parte de uma aplicação de Governo Eletrônico é submetida à execução em uma infra-estrutura de grade de computadores e os resultados obtidos são analisados.

No capítulo 6 é apresentada a conclusão do trabalho e sugestões para futuros trabalhos.

2. CARACTERIZANDO DEMANDAS DE GOVERNO ELETRÔNICO

O Portal do Governo Federal [GFB09a] diz que “Pela Constituição do País, o Governo Federal atua decisivamente na vida dos brasileiros, seja criando normas, implantando programas ou prestando serviços à população. A Presidência da República e Vice-presidência estão à frente da estrutura da administração pública federal, auxiliados por diversos órgãos e entidades controladas, de forma direta ou indireta”.

As ações desenvolvidas pelo Governo Federal impactam na vida de toda a população nas mais diversas áreas como educação, saúde, transporte, segurança, etc. Para que o Governo Federal alcance seus objetivos com sucesso suas ações devem estar alicerçadas nas melhores práticas para a utilização da tecnologia da informação e comunicação na prestação de serviços aos cidadãos brasileiros, na criação de normas ou políticas e na implantação de programas que beneficiem a população.

Para sustentar a utilização da tecnologia da informação e comunicação pelo Governo Federal, foram criadas empresas de tecnologia como a Empresa de Tecnologia e Informações da Previdência Social (Dataprev) e o Serviço Federal de Processamento de Dados (Serpro). Além disso, vários órgãos e instituições possuem uma área de TIC dentro de sua estrutura como nos casos dos bancos federais, de algumas secretarias especiais e ministérios.

Quando se trata de escolher uma tecnologia para resolver um problema, são inúmeras as opções disponíveis no mercado. Estas tecnologias estão em constante desenvolvimento gerando novas versões e novos produtos. A TIC se desenvolve, mas vários sistemas legados ainda continuam em funcionamento no Governo Federal. O que encontramos atualmente é uma diversidade de plataformas de *hardware* e *software* instaladas, utilização de sistemas legados, adoção de sistemas ou soluções proprietárias, multiplicidade de produtos e componentes para uma mesma funcionalidade utilizada nos diversos órgãos do Governo Federal. Portanto, não basta empregar a tecnologia da informação e comunicação para prestar um serviço ou fornecer acesso às informações com qualidade para os cidadãos, é preciso muito mais.

Neste sentido, a aplicação das tecnologias da informação e comunicação por todos os entes federativos nas esferas municipal, estadual e federal na prestação de serviços e no acesso às informações com qualidade para os cidadãos caracterizam o Governo Eletrônico ou e-Gov. Podemos considerar que o Governo Eletrônico é uma tendência global, onde os governos focam no desenvolvimento de políticas e definições de padrões

em TIC interoperáveis para prestar serviços e prover acesso às informações com qualidade.

Segundo o sítio do Governo Eletrônico [MPO07a], “O desenvolvimento de programas de Governo Eletrônico tem como princípio a utilização das modernas tecnologias de informação e comunicação para democratizar o acesso à informação, ampliar discussões e dinamizar a prestação de serviços públicos com foco na eficiência e efetividade das funções governamentais”.

Para promover a universalização do acesso aos serviços público, à transparência de ações, à integração de redes e ao alto desempenho dos sistemas, o Governo Federal estabelece como fundamentos básicos, o estímulo ao acesso à Internet, seja individual, público, coletivo ou comunitário. Para suportar os processos de e-Gov será necessário uma infra-estrutura que integre todas as redes de comunicação do Governo Federal, o estabelecimento de normas de segurança e privacidade, a garantia de alto desempenho e disponibilidade dos sistemas e a integração entre eles.

Para que o governo se apresente como uma instituição única integrada, transparente e eficiente, inúmeros são os desafios que se apresentam. São eles: a falta de integração entre os atuais sistemas de e-Gov, a falta de interligação das redes governamentais, a falta de padrões definidos e a dificuldade de acessibilidade aos sistemas.

Para melhor entendimento da proposta deste trabalho, serão apresentados neste capítulo os requisitos nos quais o Governo Eletrônico está fundamentado. Em seguida são apresentados alguns exemplos de sistemas de e-Gov e a infra-estrutura que suporta estas soluções.

2.1. O Governo Eletrônico

No ano 2000 o Governo Brasileiro lançou a base para o Governo Eletrônico a partir da criação de um Grupo de Trabalho Interministerial instituído pela Portaria da Casa Civil nº 23 de 12 de maio de 2000 [MPO07c] e com a promulgação do Decreto Federal (sem número) de 18 de outubro de 2000, publicado no Diário Oficial da União de 19 de outubro, que criou o Comitê Executivo do Governo Eletrônico.

O Grupo de Trabalho instituído pelo Governo Brasileiro definiu sua visão, políticas e objetivos em relação às práticas de Governo Eletrônico e a interação com as novas tecnologias da informação e comunicação.

Segundo o sítio do Governo Eletrônico [MPO07a], o Governo Eletrônico é um instrumento de transformação da sociedade brasileira, onde os papéis do Governo brasileiro perante a sociedade devem ser repensados. Assim sendo, o Governo Eletrônico deve promover a cidadania e o desenvolvimento dos cidadãos procurando atender às suas demandas enquanto indivíduos e possibilitando o acesso e a consolidação dos direitos da cidadania. O Governo Eletrônico deve funcionar como instrumento de mudança das organizações públicas, de melhoria no atendimento ao cidadão e de racionalização do uso de recursos públicos. O Governo Eletrônico deve contribuir no desenvolvimento do país promovendo o processo de disseminação da tecnologia de informação e comunicação. A promoção, o uso e a disseminação de práticas de Gestão do Conhecimento na administração pública devem ser priorizados pelo Governo Eletrônico.

Os princípios que orientam as ações de Governo Eletrônico, a gestão do conhecimento e a gestão da tecnologia da informação no âmbito do Governo Federal estão descritas na próxima seção.

2.2. Requisitos Básicos do Governo Eletrônico

De acordo com o sítio do Governo Eletrônico [MPO07d], o Governo Eletrônico será implementado segundo sete princípios, que serão adotados como referência geral para estruturar as estratégias de intervenção das ações de Governo Eletrônico, gestão do conhecimento e gestão da Tecnologia da Informação (TI) no Governo Federal.

2.2.1. A Prioridade é a Promoção da Cidadania

Segundo o sítio do Governo Eletrônico [MPO07a], a prioridade do Governo Eletrônico é a promoção da cidadania tendo como referência os direitos coletivos e uma visão de cidadania que não se restringe à somatória dos direitos dos indivíduos (abandonando a visão de que o cidadão é apenas “cliente” dos serviços públicos). Ainda assim, as necessidades e demandas individuais dos cidadãos devem ser atendidas com vínculos aos princípios da universalidade, da igualdade perante a lei e da equidade na oferta de serviços e informações de qualidade.

Os sítios e serviços on-line do Governo Federal fornecem a base para a promoção da cidadania, fornecendo acesso universal aos serviços e informações públicas, devendo

ser estruturados para atender às necessidades dos cidadãos com facilidade, qualidade e segurança.

2.2.2. Indissociabilidade entre inclusão digital e o Governo Eletrônico

Inclusão Digital é possibilitar o acesso às Tecnologias de Informação e Comunicação para todos os cidadãos, permitindo-lhes utilizar desse suporte para melhorar a sua condição de vida. A Inclusão Digital objetiva inserir o cidadão na sociedade da informação, ou seja, fornecer instrumentos e recursos para que o cidadão possa melhorar a sua condição de vida e da região em que vive.

A Inclusão Digital é indissociável do Governo Eletrônico, ou seja, a Inclusão digital é uma política do Governo Eletrônico que deve ser promovida, pois ela é um direito de cidadania.

Vários instrumentos auxiliam na divulgação e promoção da inclusão digital: o Portal de Inclusão Digital do Governo Federal [GFB08b], o Observatório Nacional de Inclusão Digital (ONID) [ONI08], o Modelo de Acessibilidade de Governo Eletrônico (e-MAG) [MPO08a].

2.2.2.1. O Portal

O Portal de Inclusão Digital do Governo Federal [GFB08b] divulga notícias, eventos e materiais de referência sobre o assunto. No portal encontram-se links para os serviços de e-Gov disponibilizados pelo Governo Federal e informações sobre os programas de inclusão digital desenvolvidos pelo Governo Federal e seus parceiros.

O Observatório Nacional de Inclusão Digital é uma iniciativa do Governo Federal em conjunto com a sociedade civil organizada que atua na coleta, sistematização e disponibilização de informações para o acompanhamento e avaliação das ações de inclusão digital no Brasil. Além de ser uma importante ferramenta para os gestores de políticas públicas e iniciativas nessa temática, o ONID disponibiliza informações detalhadas sobre os telecentros existentes em todo o país à sociedade.

O Observatório Nacional de Inclusão Digital [ONI08] mantém o cadastro de telecentros, centros de inclusão digital, infocentros ou outros espaços coletivos sem fins comerciais de uso de tecnologias digitais conectados à Internet. O preenchimento dos dados é feito pelas próprias entidades responsáveis.

2.2.2.2. e-MAG: Modelo de Acessibilidade de Governo Eletrônico

O Modelo de Acessibilidade de Governo Eletrônico tem como objetivo padronizar e facilitar a implementação de sítios e portais do Governo brasileiro focando a questão da acessibilidade. Maiores informações podem ser obtidas na página <http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG>.

2.2.3. Gestão do Conhecimento como instrumento estratégico de articulação e gestão das políticas públicas

A Gestão do Conhecimento e o Capital Social são os instrumentos estratégicos de gestão das informações e de integração de redes organizacionais do Governo Eletrônico à sociedade brasileira.

“A Gestão do Conhecimento é compreendida, no âmbito das políticas de Governo Eletrônico, como um conjunto de processos sistematizados, articulados e intencionais, capazes de assegurar a habilidade de criar, coletar, organizar, transferir e compartilhar conhecimentos estratégicos que podem servir para a tomada de decisões, para a gestão de políticas públicas e para inclusão do cidadão como produtor de conhecimento coletivo”. [MPO07a]

“O Capital Social é visto como fator fundamental para melhoria da capacidade de um país obter maiores vantagens competitivas, por meio do estímulo ao capital intelectual e a crescente percepção da sociedade com relação à sua inserção em uma Nova Economia baseada em redes de informações e os seus possíveis impactos no desenvolvimento do país”. [MPO07a]

Segundo Gartner Group citado por Alvares [ALV09], a gestão do conhecimento “É uma disciplina que promove, com visão integrada, o gerenciamento e o compartilhamento de todo o ativo de informação possuído pela empresa. Esta informação pode estar em um banco de dados, documentos, procedimentos, bem como em pessoas, através de suas experiências e habilidades”.

2.2.4. Adoção de políticas, normas e padrões comuns

A definição e publicação de políticas, padrões, normas e métodos para sustentar as ações de implantação e operação do Governo Eletrônico são imprescindíveis para o seu sucesso. Neste sentido, a arquitetura e-PING [MPO07e] (Padrões de Interoperabilidade

de Governo Eletrônico) define um conjunto mínimo de premissas, políticas e especificações técnicas que regulamentam a utilização da Tecnologia da Informação e Comunicação no Governo Federal, estabelecendo as condições de interação com os demais poderes e esferas de governo e com a sociedade em geral.

A interação entre diferentes sistemas de informação de vários órgãos do Governo Federal é possibilitada pela arquitetura e-PING, proporcionando benefícios como unificação dos cadastros sociais, a unificação dos sistemas de segurança, entre outros.

2.2.4.1. e-PING: Padrões de Interoperabilidade de Governo Eletrônico

O projeto e-PING está sob a responsabilidade dos seguintes órgãos: Secretaria de Logística e Tecnologia da Informação (SLTI), ligada ao Ministério do Planejamento, o Serpro e o Instituto Nacional de Tecnologia da Informação (ITI/PR).

A arquitetura e-PING permite o diálogo entre os diferentes sistemas de informação existentes nas diferentes esferas de governo proporcionando benefícios como a unificação dos cadastros sociais, a unificação dos sistemas de segurança, entre outros.

De acordo com o sítio do Governo Eletrônico [MPO07e], conceitua-se interoperabilidade como:

- “Habilidade de transferir e utilizar informações de maneira uniforme e eficiente entre várias organizações e sistemas de informações. (Governo da Austrália)
- Intercâmbio coerente de informações entre serviços e sistemas. Deve possibilitar a substituição de qualquer componente ou produto usado nos pontos de interligação por outro de especificação similar, sem comprometer as funcionalidades do sistema. (Governo do Reino Unido)
- Habilidade de dois ou mais sistemas (computadores, meios de comunicação, redes e outros componentes de TI) de interagir e intercambiar dados de acordo com um método definido de forma a obter os resultados esperados. (ISO).”

Pode-se dizer que interoperabilidade é o conjunto destes fatores, sem nos esquecer de considerar os sistemas legados, plataformas de *hardware* e *software* antigos em uso.

Para enfrentar esses desafios o Governo Eletrônico através da e-PING [MPO07e] define vários objetivos. Um deles é a definição de normas e padrões para a contratação, o uso e a gestão de infra-estrutura de redes de comunicação (dados, voz e imagem) para o Governo Federal. Além disso, a interoperabilidade entre os diversos níveis de governo deve estar baseada na utilização de padrões abertos e públicos. O mesmo vale para o desenvolvimento de novos sistemas onde devem-se empregar tecnologias abertas. Os

sítios e serviços on-line devem estar integrados para que serviços de qualidade (com foco na usabilidade, disponibilidade e acessibilidade) sejam prestados. Por último, um conjunto de políticas de gestão do conhecimento, inclusão digital, *software* livre e outras correlacionadas ao Governo Eletrônico devem ser elaborados e aplicados.

2.2.5. Utilização do *software* livre como recurso estratégico

A questão do *software* livre está inserida em um amplo cenário de racionalização da utilização de recursos públicos, de interoperabilidade, e de conformidade com as diretrizes de Governo Eletrônico.

Segundo o sítio do Governo Eletrônico [MPO07d] “o *software* livre é uma opção tecnológica do Governo Federal e seu uso será promovido pelos gestores públicos. Ou seja, os gestores públicos devem priorizar soluções, programas e serviços baseados em *software* livre que promovam a otimização de recursos e investimentos em tecnologia da informação”.

De acordo com o sítio do *software* livre [GFB07a], as ações do Governo Eletrônico para utilização do *software* livre são orientadas por diretrizes que promovam a otimização de recursos e investimentos em tecnologia da informação priorizando a adoção de soluções, programas e serviços baseados em *software* livre ampliando a gama de serviços prestados aos cidadãos com qualidade proporcionado pela adoção de ferramentas de desenvolvimento de sistemas e interfaces de usuários baseados na internet, sempre respeitando-se a legislação de sigilo e segurança vigente. Estas diretrizes privilegiam a adoção de plataformas abertas, migrando gradativamente os sistemas legados para a plataforma aberta sem perder de vista a interoperabilidade com estes sistemas. Garantem também, o fortalecimento da livre distribuição de produtos e trabalhos de desenvolvimento colaborativos, incentivando e fomentando o mercado nacional a adotar novos modelos de negócios em tecnologia da informação e comunicação baseados em *software* livre. Havendo assim, a necessidade de formular uma política nacional para o *software* livre que garanta as condições para a mudança da cultura organizacional e promova a capacitação de servidores públicos para utilização de *software* livre.

A questão do *software* livre está inserida em um amplo cenário de racionalização da utilização de recursos públicos, de interoperabilidade, de inclusão digital e de conformidade com as diretrizes de Governo Eletrônico.

2.2.5.1. O Guia Livre - Referência de Migração para *Software* Livre

O Guia Livre [MPO07f] é um documento que serve como referência na migração para *software* livre para qualquer entidade. O guia reúne num único documento uma coletânea de experiências vivenciadas por profissionais de diferentes órgãos públicos e da sociedade, compiladas pelo grupo de trabalho de Migração para *software* livre do Governo Federal e com participação da comunidade de *software* livre brasileira.

Segundo o documento [MPO07f], o Guia Livre tem como objetivo ajudar os administradores a definir uma estratégia para migração planejada e gerenciada através de descrições técnicas de como realizar estas migrações, alinhando as diretrizes e definições deste Guia aos Padrões de Interoperabilidade do Governo Brasileiro.

O guia livre apresenta um detalhamento técnico sobre a utilização de várias ferramentas de *software* em vários ambientes de TI e objetiva criar as condições necessárias para a migração para um ambiente aberto.

2.2.5.2. O Guia Cluster

A partir do Guia Livre foi proposta uma nova publicação contendo experiências e procedimentos para a administração, gestão e o desenvolvimento do ambiente cluster com o emprego de tecnologias livres e abertas. O documento denominado “Guia de Inovação Tecnológica em Cluster e Grid” norteará a internalização na Administração Pública Federal dos benefícios do ambiente cluster e do *software* livre e também permitirá a troca de experiências entre os diversos órgãos governamentais e entidades que fazem uso de ferramentas deste tipo em seus sistemas, agregando assim o conhecimento em torno destas tecnologias. [MPO07a]

Segundo o documento Guia Cluster [MPO07a], o domínio do Governo Brasileiro na utilização da tecnologia de *Cluster* é um recurso estratégico para alcançar os padrões de eficiência, eficácia e efetividade exigida por seus gestores e pela sociedade, além de criar um ciclo positivo de aplicação de novas tecnologias no interior do governo, modernizando a máquina pública.

A versão on-line do Guia Cluster encontra-se disponível no sítio <http://guialivre.governoeletronico.gov.br/guiaonline/guiacluster/> [MPO07a].

2.2.5.3. Vantagens na adoção da tecnologia de grades

O Guia Cluster [MPO07a] apresenta as demandas e os desafios computacionais do Governo Brasileiro e a possibilidade de utilização de tecnologias baseadas em cluster e grade para auxiliar no atendimento destas demandas.

Segundo o Guia Cluster [MPO07a], as vantagens técnicas obtidas com a adoção da tecnologia de grades são:

- “Utilização de *hardware* padrão de mercado em sistemas críticos através da transferência do gerenciamento das funções de alta disponibilidade, tolerância a falhas e balanceamento de carga do *hardware* para o *software*, diminuindo a necessidade de *hardware* especializado, aumentando a concorrência entre as empresas fornecedoras e propiciando ao governo a independência tecnológica de fornecedores de *hardware*.
- Utilização de padrões abertos e interoperáveis, facilitando a integração de sistemas baseados em grades em oposição a sistemas em computação de grande porte que utilizam, em sua grande parte, tecnologias proprietárias e padrões fechados.
- Disponibilidade de soluções baseadas em *software* livre que permitem a implementação de sistemas de cluster e grade sem a necessidade de ônus de licenças de *software*, além de permitir a melhoria, alteração, distribuição e compartilhamento de soluções, segurança, transparência e possibilidade de auditoria plena do sistema.
- Maior facilidade para aumentar ou diminuir a capacidade computacional de acordo com a demanda existente, utilizando grades e clusters computacionais.
- Possibilidade do desenvolvimento de sistemas e serviços que utilizem os conceitos de computação sob demanda, com o objetivo de aproveitar da melhor maneira possível os sistemas e recursos computacionais existentes no governo.
- Possibilidade de realizar o aproveitamento de "ciclos ociosos" de computadores já existentes na atual infra-estrutura de TIC.”

A implantação do e-Gov por parte do Governo brasileiro abre o caminho para a utilização de soluções baseadas em *software* livre e totalmente interoperáveis, como por exemplo, o desenvolvimento de soluções que agreguem maior poder de processamento a baixo custo com a utilização das grades computacionais.

2.2.6. Racionalização de Recursos

De acordo com o sítio do Governo Eletrônico [MPO07d], "O Governo Eletrônico não deve significar aumento dos dispêndios do Governo Federal na prestação de serviços e em tecnologia da informação. Ainda que seus benefícios não possam ficar restritos a este aspecto, é inegável que deve produzir redução de custos unitários e racionalização do uso de recursos."

O compartilhamento de recursos entre os órgãos públicos possibilita atingir alguns objetivos do Governo Eletrônico, como a racionalização de recursos. Um dos exemplos é o Projeto Infovia Brasil, cujo objetivo é "prover uma infra-estrutura de comunicação de voz, dados e imagem com capilaridade superior à existente, com a qualidade necessária, o menor custo possível e com grau de segurança adequado para toda a Administração Pública Federal, de forma a suportar as demandas de serviços dos projetos de Governo Eletrônico." [SAN07]

De acordo com o sítio do Governo Eletrônico [MPO07d], a racionalização de recursos se orienta pela aplicação de novos métodos computacionais com requisitos menores de infra-estrutura tecnológica e pelo compartilhamento de recursos de infra-estrutura, de sítios e de serviços on-line pelas entidades do Governo Federal. Orienta-se também pela otimização dos serviços de comunicação (Projeto Infovia Brasil), pela adoção de padrões abertos de *software*, *hardware* e serviços e pela ampliação da capacidade de negociação da Administração Pública Federal na aquisição de *software* e *hardware* e na contratação de serviços possibilitada pela utilização de padrões, normas e estruturas referenciais de custos.

2.2.7. Integração com outros níveis de governo e com os demais poderes

A Divisão de Poderes dos entes federativos (a União, os Estados, o Distrito Federal e os Municípios) do Estado Brasileiro não podem servir de barreira para a falta de integração das ações governamentais.

"A implantação do Governo Eletrônico não pode ser vista como um conjunto de iniciativas de diferentes atores governamentais que podem manter-se isoladas entre si. Pela própria natureza do Governo Eletrônico, este não pode prescindir da integração de ações e de informações. A natureza federativa do Estado brasileiro e a divisão dos Poderes não podem significar obstáculo para a integração das ações de Governo Eletrônico. Cabe ao Governo Federal um papel de destaque nesse

processo, garantindo um conjunto de políticas, padrões e iniciativas que garantam a integração das ações dos vários níveis de governo e dos três Poderes" [MPO07a]

Para viabilizar a integração entre os diferentes níveis de Governo e os Poderes constituintes, é necessário criar os meios necessários para a integração do Governo Eletrônico com os demais entes federativos. Além disso, é necessário estabelecer estratégias de parceria com estados e municípios através da definição de padrões de interoperabilidade e mecanismos de integração horizontal e vertical de dados e de sistemas nos vários níveis de governo, da promoção do compartilhamento de recursos tecnológicos, de informações, humanos e financeiros, e com o estabelecimento de estratégias de parceria com estados e municípios.

2.3. Exemplos de Aplicações de Governo Eletrônico

As aplicações de Governo Eletrônico se caracterizam pelos benefícios que proporcionam para os cidadãos através da prestação de serviços e disponibilização de informações. A seguir são apresentados exemplos de aplicações de e-Gov e a infraestrutura tecnológica que sustentam estas soluções.

2.3.1. O Sistema de Cadastro Nacional de Informações Sociais

O Cadastro Nacional de Informações Sociais (CNIS) é um projeto de Governo criado com o objetivo de ser uma base de dados que registre as informações necessárias para a garantia dos direitos trabalhistas e previdenciários dos trabalhadores brasileiros. Criado originalmente com a denominação de Cadastro Nacional do Trabalhador (CNT) através do decreto 97.936 de 1989, na forma de consórcio entre Ministério da Previdência e Assistência Social (MPAS), Ministério do Trabalho (MTb) e Caixa Econômica Federal (CEF), posteriormente assumiu, conforme lei 8.212 de 1991, a denominação de CNIS. [DAT07]

As fontes de dados do CNIS têm origem em entidades externas à Previdência Social e internamente também. A diversidade de fontes de dados torna disponível um grande volume de dados e informações que cresce constantemente. Atualmente esta base de dados conta com mais de 190 milhões de registros.

O processamento e armazenamento dos dados do CNIS é realizado atualmente utilizando computador de grande porte pela Empresa de Tecnologia e Informações da

Previdência Social sendo implementado em linguagem Cobol e banco de dados não relacional DMS II.

O sistema apresenta como pontos fracos, a lentidão na consulta à base de dados, a falta de atualização de alguns dados cadastrais como endereço da pessoa física, utilização de sistema gerenciador de banco de dados não relacional, entre outros. A lentidão no acesso à base de dados ocorre em função da tecnologia adotada não suportar com desempenho adequado o grande número de requisições às quais são submetidos os equipamentos utilizados.

A base de dados do CNIS é usada para comprovar os vínculos empregatícios e contribuições do trabalhador no momento que o mesmo faz uso de seus benefícios (solicitação de aposentadoria, por exemplo) junto à Previdência Social. Para que não gere impactos negativos para o Governo e a sociedade, é necessário garantir a qualidade das informações armazenadas na base de dados do CNIS e o desempenho e a disponibilidade da aplicação.

2.3.2. O Sistema Único de Benefícios

Segundo dados do Ministério da Previdência Social [MPS07][MPS09], a Dataprev processa uma folha de pagamento para mais de 26 milhões de segurados e beneficiários correspondendo a um valor superior a 15 bilhões de reais. O Sistema Único de benefícios possui dados de mais de 51 milhões de contribuintes, sendo 40 milhões de trabalhadores com carteira assinada e 11 milhões de contribuintes individuais.

Para que os segurados e beneficiários da Previdência Social recebam os valores a que têm direito, as rotinas operacionais de Cálculo e da Geração de Créditos são executados periodicamente. A maior delas denominada Maciça é executada mensalmente sendo responsável pela geração dos créditos de concessão e revisão dos benefícios previdenciários. Da Maciça derivam outros processos complementares como a geração do Histórico de Créditos dos Beneficiários (HISCRE) e da Declaração Anual de Rendimentos (Imposto de Renda). Após o processamento da Maciça, os créditos gerados são encaminhados para os operadores bancários, empresas conveniadas ou acordos internacionais dependendo do tipo e origem do benefício.

Para uma pequena porção de benefícios que não foram processados pela Maciça, por diversas razões, e devido à sua urgência não podem ser processados na próxima competência, é realizado o processamento denominado Pagamento Alternativo de

Benefícios (PAB). Esta rotina é executada sob demanda de acordo com a necessidade do cliente.

O processamento da Maciça, do PAB e de outras funções do Sistema Único de Benefícios, é realizado nos computadores de grande porte na Dataprev e consomem recursos valiosos destas máquinas. Durante o processamento da maciça, por exemplo, os sistemas ficam indisponíveis por questões de segurança, performance e integridade dos dados do sistema. Utilizando recursos do computador de grande porte é o sistema implementado em linguagem Cobol e usa banco de dados DMS II.

Trata-se de utilização de tecnologia proprietária, cara e que necessita se adequar às diretrizes de e-Gov com finalidade de proporcionar maior disponibilidade e desempenho ao sistema.

2.3.3. O Sistema PRISMA

O sistema Prisma é responsável pela manutenção de benefícios, ou seja, sua função é dotar as agências da Previdência Social de serviços como: habilitação, concessão, atualização e revisão de benefícios, controle de perícia médica, pagamento alternativo de benefícios, cadastramento de contribuinte individual, comunicação de acidente de trabalho, controle de procuradores e entidades filantrópicas, entre outros.

Este sistema foi desenvolvido originalmente em linguagem Pick em plataforma Pick/AP sendo migrado gradativamente para um ambiente com servidores com sistema operacional SCO OpenServer (Unix) e banco de dados D3 (da Raining Data).

A Figura 1 mostra a arquitetura do sistema PRISMA e o fluxo de dados entre os diversos componentes. O Sistema Prisma é acessado pelos servidores do INSS nas Agências da Previdência Social distribuídas pelo país. Os servidores PRISMA ficam hospedados nos Centros de Tratamento de dados do Rio de Janeiro, São Paulo e Brasília sendo acessados através de um emulador de terminal. O servidor APL06 é um servidor de comunicação utilizado para transações on-line do sistema prisma com o Sistema Único de Benefícios (SUB) hospedado no Mainframe. O servidor de comunicação APL09 faz a interligação do sistema CNIS hospedado no Mainframe com os servidores PRISMA. O servidor de comunicação APL10 faz a interligação do Sistema de Administração de Benefícios por Incapacidade (SABI) com os servidores PRISMA. A comunicação do Prisma com as bases de dados de outros sistemas é necessária pela maioria dos processos do Prisma para o atendimento aos serviços dos clientes previdenciários.

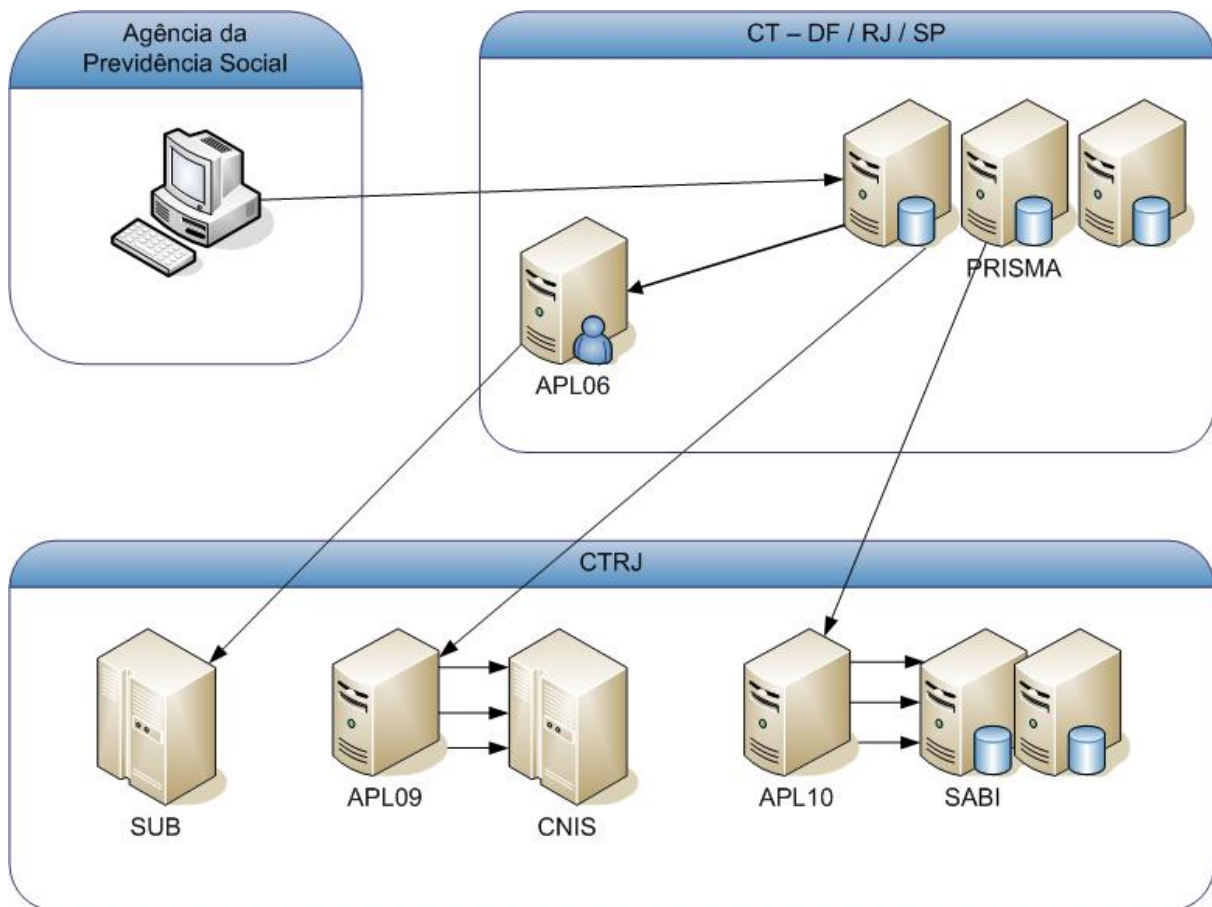


Figura 1 - Arquitetura do Sistema Prisma

Devido a problemas de limitação de licenças do banco de dados D3 (na versão para o sistema SCO OpenServer 5) por máquina adquirida pela Dataprev, onde somente 256 usuários podem estar ativos simultaneamente, incluindo usuários de gerência e monitoração do próprio banco de dados D3, são necessárias centenas de máquinas para armazenar e processar os dados deste sistema.

Estes equipamentos ocupam grandes espaços nos Centros de Tecnologias da Dataprev e possuem vários anos de uso gerando alto custo de manutenção de *hardware* e de controle operacional.

2.3.4. A Base de Dados da Justiça Eleitoral

Segundo dados do Tribunal Superior Eleitoral (TSE) [TSE08], o Brasil possui mais de 130 milhões de eleitores distribuídos pelo país. As informações do eleitorado brasileiro, inscrito no país e no exterior, estão armazenadas no banco de dados do sistema de alistamento eleitoral. (determinado pela Lei nº 7.444, de 20.12.85 [GFB09b]), sendo

unificado em nível nacional. Atualmente este cadastro mantém registros de dados pessoais de todo o eleitorado e de ocorrências pertinentes ao histórico de cada título eleitoral.

Segundo o portal do TSE [TSE08], “A supervisão, orientação e fiscalização voltadas à preservação da integridade de suas informações estão confiadas à Corregedoria-Geral da Justiça Eleitoral, em âmbito nacional, e às corregedorias regionais eleitorais, nas respectivas circunscrições”.

O processo de informatização da justiça eleitoral teve início em 1983 com a instalação de computadores nas zonas eleitorais e no TRE. Com a implantação das urnas eletrônicas a partir de 1996, o processo de apuração e totalização dos votos tornou-se mais seguro. Além disso, o processo de totalização dos votos e a divulgação dos resultados foram agilizados tornando o país uma referência mundial. O TSE planeja a utilização de urna com sistema de identificação por biometria para melhorar ainda mais o nível de segurança destes equipamentos.

Mas o processo de identificação do eleitorado apresenta fragilidades, como por exemplo, o título de eleitor sem foto e qualquer informação adicional ou a possibilidade de duplicação de títulos valendo-se de documentos falsos. Com a finalidade de validar o Cadastro Nacional de Eleitores e facilitar o processo de recadastramento dos eleitores sugere-se o batimento destas informações com as diversas bases de dados do Governo Federal. Por exemplo: a base de dados do CNIS, o cadastro de pessoa física (CPF) da Receita Federal entre outros.

2.3.5. O Sistema da Assessoria de Pesquisas Estratégicas

O planejamento estratégico da atual gestão do Ministério da Previdência Social contempla três eixos principais: redução de filas, do desperdício e combate à fraude e à sonegação. No contexto do combate às fraudes contra a Previdência Social, a Assessoria de Pesquisas Estratégicas (APE) [MPS08a] é responsável pelas ações e procedimentos técnicos de inteligência, das metodologias, controles e normas de segurança, realização de programas e atividades de combate à fraude ou quaisquer atos lesivos à Previdência Social. A portaria MPAS nº 36, de 6 de Fevereiro de 2006 [MPS06], publicada no Diário Oficial da União de 07/02/2006 define a estrutura, atribuições, competências e prerrogativas da Assessoria de Pesquisa Estratégica fortalecendo sua atuação.

A Assessoria de Pesquisa Estratégica trabalha em conjunto com a Força-Tarefa Previdenciária. Segundo a Portaria nº 36 [MPS06] algumas competências da Assessoria de Pesquisas Estratégicas são:

- “Subsidiar o MPS e as entidades vinculadas com informações estratégicas decorrentes do exercício da atividade de inteligência;
- Produzir conhecimentos estratégicos visando a identificação de fatos ou situações que possam comprometer o cumprimento da legislação previdenciária;
- Planejar e coordenar o exercício sistemático e permanente de suas ações especializadas e dos Grupos de Trabalho das Forças-Tarefas.”

De acordo com a Agência de Notícias da Previdência [MPS08a], “a Força-Tarefa Previdenciária foi criada há seis anos e é formada por equipes do Ministério da Previdência Social, do Departamento da Polícia Federal e do Ministério Público Federal. Elas atuam em todo o país e são responsáveis por combater fraudes e sonegação fiscal contra o INSS. As suspeitas de irregularidades apuradas pelos técnicos da Previdência Social são encaminhadas à APE da região investigada, que toma as medidas necessárias para solucionar o problema.”

Varrendo a base dos Sistemas de Benefícios da Previdência Social, podem-se identificar problemas na base de dados à procura de padrões (indícios) de irregularidades:

- Beneficiários mortos recebendo pagamentos;
- Indivíduos com mais de um benefício;
- Utilização de um mesmo CPF por pessoas diferentes;
- Utilização de um mesmo endereço por um conjunto de beneficiários;
- Procurações de um mesmo advogado com o mesmo endereço para beneficiários diferentes (o endereço do beneficiário deve ser o endereço residencial do segurado).

As informações manipuladas pela APE são oriundas de diversas bases de dados armazenadas nos computadores de grande porte e de plataforma baixa nos ambientes de TIC da Dataprev. Sendo assim, observa-se uma heterogeneidade de plataformas e sistemas que sustentam e armazenam essas informações. O desafio aqui é manipular uma grande quantidade de informação oriunda de fontes heterogêneas de forma eficiente e rápida para a tomada de decisões pela equipe da APE.

2.3.6. O Sistema Integrado de Administração Financeira do Governo Federal

O Sistema Integrado de Administração Financeira (SIAFI) é o principal instrumento utilizado para registro, acompanhamento e controle da execução orçamentária, financeira e patrimonial do Governo Federal. Este sistema foi definido pela Secretaria do Tesouro Nacional (STN) e desenvolvido pelo SERPRO e implantado em janeiro de 1987 fornecendo ao Governo Federal um instrumento moderno e eficaz de controle e acompanhamento dos gastos públicos.

Segundo o portal SIAFI [MFZ09a], “Hoje o Governo Federal tem uma Conta Única para gerir, de onde todas as saídas de dinheiro ocorrem com o registro de sua aplicação e do servidor público que a efetuou. Trata-se de uma ferramenta poderosa para executar, acompanhar e controlar com eficiência e eficácia a correta utilização dos recursos da União.”

O SIAFI pode ser acessado através da intranet do Serpro, por uma Rede Privada Virtual (VPN), acesso extranet, acesso seguro pela internet e acesso discado seguro. O sistema é hospedado em computador de grande porte no SERPRO.

Segundo o Manual SIAFI WEB [MFZ09b], “Os usuários são responsáveis pela qualidade e veracidade dos dados introduzidos no SIAFI; a STN é responsável pela definição lógica e de normas de utilização; o SERPRO, é responsável pelo desenvolvimento e funcionamento do Sistema, armazenamento e segurança dos dados.”

O sistema é hospedado em computador de grande porte IBM e o usuário interage com o sistema SIAFI usando como interface um emulador de terminal 3270.

Apesar de ser uma ferramenta eficaz no controle da Execução Orçamentária e Financeira o SIAFI deve se adaptar às diretrizes do Governo Eletrônico e às novas tecnologias possibilitando o intercâmbio de dados com outros sistemas.

2.4. Considerações Finais sobre Sistemas de e-Gov

Observa-se que questões como emprego de padrões de interoperabilidade, adoção de padrões abertos e racionalização de recursos são peças chaves para o alcance dos objetivos propostos.

Os sistemas de e-Gov apresentados mostram as dificuldades impostas pela tecnologia utilizada. Em sua maioria utilizam computadores de grande porte como plataforma tecnológica para suportar as soluções desenvolvidas. São soluções caras, proprietárias e muitas vezes trabalhando no limite da infra-estrutura hospedada em seus

sítios podendo levar à paralisação de serviços essenciais à população brasileira. Também pecam pela falta de relacionamento com outros sistemas do Governo Federal. Fica evidente a necessidade de que estes sistemas estejam alinhados com as diretrizes do Governo Eletrônico buscando a racionalização dos recursos, diminuição de custos de *hardware* e *software*, adoção de soluções abertas e interoperáveis, etc.

Para entender melhor o que são grades e como estas podem ser empregadas para e-Gov, no próximo capítulo são apresentados os conceitos sobre grades de computadores e exemplos de aplicações que se beneficiam da utilização desta tecnologia.

3. GRADES COMPUTACIONAIS

Conforme apresentado no capítulo anterior, fica claro que as aplicações de e-Gov precisam se alinhar às diretrizes do Governo Eletrônico para que o Governo Federal possa prestar serviços de qualidade, eficientes e de baixo custo, disseminando a informação e gerando novos conhecimentos. Várias aplicações de e-Gov requerem um grande poder de processamento para que seus processos sejam executados de maneira eficiente. Além disso, necessitam de grande quantidade de memória e espaço de armazenamento crescente, tornando onerosas as soluções em uso atualmente.

Para atingir seus objetivos o Governo Federal define através do Governo Eletrônico o uso que fará da tecnologia de informação e comunicação. O Governo Eletrônico deve prover uma infra-estrutura disponível, de alto desempenho e baixo custo possibilitando o acesso público aos serviços e informações. Para o uso da TIC no Governo Eletrônico foram definidas diretrizes tecnológicas para *hardware*, sistemas operacionais, arquitetura das aplicações, redes de comunicação, banco de dados, ferramentas de produtividade, segurança entre outros aspectos. Por exemplo, uma moderna rede de comunicação de dados, voz e imagem é pré-requisito para interligar os diversos órgãos, instituições e empresas públicas do Governo Federal sedimentando o acesso aos sistemas de Governo Eletrônico. Isto é possível pela utilização de protocolos de comunicação padrão que disponibilizem serviços de comunicação de dados, voz e imagem de forma segura e eficiente.

As tecnologias atualmente empregadas em várias aplicações de e-Gov são proprietárias, obsoletas e caras criando uma dependência tecnológica de fornecedores e fabricantes de *hardware* e de *software*. Assim, para rodar estas aplicações é necessária uma infra-estrutura tecnológica que as suportem com disponibilidade, desempenho, escalabilidade, segurança e custos menores. No capítulo anterior foi apresentado que a utilização de padrões abertos, a racionalização de recursos e a adoção de *software* livre são requisitos do Governo Eletrônico. No caso de sistemas operacionais, devem-se empregar aqueles que não gerem dependências para as aplicações que devem rodar sobre eles, ampliando o número de fornecedores a serem escolhidos, garantindo a portabilidade das aplicações e provendo ambientes de alta disponibilidade, escaláveis e robustos.

Neste sentido, as grades computacionais podem prover o ambiente necessário para a execução de aplicações de e-Gov de forma eficiente e a custos menores. As grades

computacionais possibilitam o processamento de grandes volumes de dados e são uma opção de menor custo quando comparados a computadores de grande porte. Elas possibilitam a utilização de recursos distribuídos geograficamente, permitem a disponibilização de recursos e serviços de forma transparente, permitem o compartilhamento de recursos e serviços e otimizam a utilização de ciclos ociosos de recursos computacionais.

A seguir são apresentados os conceitos relacionados a grades computacionais mostrando os benefícios que a utilização desta tecnologia proporciona.

3.1. Conceituando Grades

De acordo com Foster e Kesselman [FOS99] “o conceito de grade computacional estabelece uma metáfora com a Rede Elétrica (*The Electric Grid*)”. Assim, uma grade computacional é uma rede interligando diversos recursos computacionais distribuídos geograficamente fornecendo recursos sob demanda para execução de aplicações paralelas.

Segundo Foster e Kesselman [FOS99], “Uma Grade Computacional é uma infraestrutura de *hardware* e *software* que provê acesso seguro, consistente, de forma dispersa e a custo baixo à potenciabilidade computacional máxima”. Ainda, uma grade é um sistema que coordena recursos de diferentes tipos, usa interfaces simples e de alto nível, protocolos abertos e de propósitos gerais, e oferece Qualidade de Serviço (QoS) não triviais: segurança, autenticação, escalonamento de tarefas, disponibilidade e tempo de resposta.

No artigo “*The Anatomy of the Grid*” [FOS01] Foster e Kesselman dizem que o problema real e específico das grades é “a coordenação de recursos compartilhados e solução de problemas em organizações virtuais dinâmicas e multi-institucionais”.

Berman [BER03] diz que as grades irão transformar a ciência, a economia, a saúde e a sociedade e define:

Infra-estruturas de grades irão fornecer a capacidade para associar dinamicamente recursos como um conjunto para suportar a execução de aplicações distribuídas em grande escala, com uso intensivo de recursos.

As Grades Computacionais nasceram da comunidade de Processamento de Alto Desempenho, motivada pela idéia de utilizar computadores independentes e amplamente dispersos como plataforma para a execução de aplicações paralelas [FOS99]. Hoje temos a aplicação de grades computacionais não somente na pesquisa acadêmica, mas

também em grandes empresas que buscam soluções para os problemas de computação distribuída.

3.2. Tipos de Grades

Na resenha sobre os principais sistemas de grades existentes elaborado por Krauter, Buyya e Maheswaran [KRA02] em 2002 e citados por Kon e Goldman [KON08] foi proposto uma taxonomia que categoriza as grades de acordo com a sua utilização.

Segundo os autores, uma Grade de Computação (Computing Grid) é um sistema com capacidade de agregar a capacidade computacional de máquinas espalhadas e utilizadas para execução de aplicações paralelas minimizando o tempo de processamento. São voltadas a aplicações de previsão do tempo e simulações, por exemplo. Um subtipo denominado de Grades Computacionais Oportunistas (*Opportunistic Grids* ou *Scavenging Grids*) [GOL04, LIV97] são utilizadas para processar aplicações aproveitando os recursos ociosos de máquinas não dedicadas à grade, como por exemplo, estações de trabalho de um laboratório acadêmico ou de funcionários de uma organização.

As Grades de Dados (*Data Grids*) fornecem tecnologia para obter novas informações a partir do processamento de grande volume de dados armazenados em repositórios distribuídos geograficamente e conectados por redes de longa de distância. Estas grades focam no acesso, na pesquisa e no gerenciamento dos repositórios de dados com habilidade para suportar bases de dados complexas, heterogêneas e distribuídas.

As Grades de Serviços (*Service Grids*) "fornecem serviços viabilizados pela integração de diversos recursos computacionais, como por exemplo, um ambiente para colaboração à distância". [GOL04]

3.3. Aplicações

Segundo Berman [BER03], as grades servem como uma tecnologia que permite um amplo conjunto de aplicações em ciência, negócios, entretenimento, saúde e outras áreas. Na comunidade de grades, esforços de infra-estrutura de grades, esforços de desenvolvimento de aplicações e esforços de *middleware* têm progredido em conjunto, muitas vezes através da colaboração de equipes multidisciplinares.

Foster [FOS99], baseado em suas experiências, identificou cinco classes de aplicações para grades computacionais: aplicações para supercomputação distribuída, aplicações de computação de alto desempenho, aplicações sob demanda, aplicações intensivas de dados e aplicações colaborativas.

As aplicações para supercomputação distribuída usam grades para agregar recursos computacionais substanciais a fim de tentar resolver problemas que não podem ser resolvidos em sistemas simples, por exemplo, agregando os recursos dos supercomputadores de um país ou as estações de trabalho de dentro de uma empresa.

A grade é usada em computação de alta vazão para processar um grande número de tarefas fracamente acopladas ou independentes com a utilização de ciclos ociosos de processadores (frequentemente das estações de trabalho inativas).

As aplicações sob demanda usam as potencialidades da grade para encontrar recursos com requisitos de curto prazo que não estão disponíveis localmente. Estes recursos podem ser computação, *software*, repositórios de dados, sensores especializados, entre outros. Estas aplicações são dirigidas frequentemente por interesses do custo e desempenho.

O foco das aplicações intensivas de dados está na síntese de novas informações a partir dos dados mantidos em repositórios geograficamente distribuídos, em bibliotecas digitais e em bancos de dados.

O primeiro interesse das aplicações colaborativas é permitir e promover as interações humano-a-humanos. Tais aplicações são freqüentemente estruturadas através do uso de um espaço virtual compartilhado. Os recursos computacionais compartilhados podem ser arquivos de dados, resultado de simulações, etc.

Berman [BER03], por sua vez, apresenta uma relação de aplicações que se beneficiam dos recursos disponibilizados pelas grades computacionais, são elas: aplicações das ciências da vida, aplicações orientadas à engenharia, aplicações orientadas a dados, aplicações em ciências físicas, aplicações para colaboração em pesquisas e aplicações comerciais.

As aplicações das ciências da vida (biologia computacional, bioinformática, estudo do genoma, neurociência computacional) estão usando a tecnologia de grades como forma de acesso, coleta e mineração de dados, execução de simulação e análise em grande escala, e para se conectar a instrumentos remotos. Por exemplo: a rede de pesquisa em informática biomédica (BIRN) é um projeto pioneiro que utiliza a infraestrutura de grades para suportar estudos de correlação cruzada de imagens e outros dados essenciais para a neurociência e avanços biomédicos.

A grade tem proporcionado uma importante plataforma para fazer uso intensivo de recursos de aplicações de engenharia mais rentáveis nas aplicações orientadas à engenharia.

Na próxima década os dados virão de instrumentos científicos, experiências, sensores, rede de sensores e de um grande número de novos dispositivos caracterizando as aplicações orientadas a dados. A grade será utilizada para coletar, armazenar e analisar dados e informações, bem como para sintetizar conhecimentos a partir destes dados. Um exemplo é o ambiente distribuído de manutenção de aeronaves (DAME). Nesta aplicação industrial a grade é usada para tratar os gigabytes de dados recolhidos em voo por motores de aeronaves operacionais e integrar a manutenção, o fabricante e os centros de análise.

As aplicações em ciências físicas têm tido um rápido crescimento no uso de grades devido à natureza de suas aplicações, ou seja, necessidade de processar um grande volume de dados. A comunidade de astronomia também tem-se voltado para as grades como um meio para captação, compartilhamento e mineração de dados críticos sobre o universo.

As aplicações de colaboração em pesquisa ilustram as mudanças que a tecnologia da informação está trazendo para a metodologia de pesquisa científica [FOX02]. A ciência capta a nova abordagem para ciência envolvendo a colaboração global distribuída proporcionada pela internet e utilizando conjuntos muito grandes de dados, recursos computacionais altamente escaláveis e visualizações de alto desempenho. Na última década a pesquisa estava centrada na simulação e na integração com a ciência e engenharia (ciência computacional). As grades podem fornecer a base para apoiar a interação à distância, em tempo real, nos trabalhos colaborativos.

Nas aplicações comerciais os conceitos de grades, de Web, de computação distribuída e de informação estão sendo usados de uma forma inovadora, numa ampla variedade de áreas, incluindo controle de inventário, computação corporativa, jogos e assim por diante.

As aplicações são essenciais para as grades e um dos principais efeitos da ampla utilização de grades é incentivar o desenvolvimento de novas aplicações

Por fim, observa-se no âmbito das entidades que prestam serviços públicos uma tendência na adoção de soluções de grades computacionais para o processamento de suas aplicações visando atender às demandas do Governo Eletrônico.

3.4. Classificação de Aplicações

Atualmente é grande a quantidade de dados que são gerados pelos mais diversos dispositivos de coleta de dados (sensores, satélites, etc.). Estes dados precisam ser tratados de forma adequada e no tempo ideal para que as informações geradas sejam úteis para aqueles que fazem uso dela. O processamento desta grande quantidade de dados exige também grande capacidade computacional e pode ser tratada de forma adequada com a utilização de grades computacionais.

As aplicações podem ser classificadas de diversas formas dependendo do critério utilizado, seja a forma de acesso aos dados, a localização da aplicação, o uso recursos computacionais entre outros.

Por exemplo, a aplicação pode ser classificada pela forma com que os dados e a aplicação são disponibilizados para os usuários, neste caso, denominada como aplicação de n camadas. Numa aplicação de uma camada os dados e programas não estão organizados, uma aplicação de duas camadas é denominada cliente-servidor, uma aplicação de três camadas faz uso de um serviço de apresentação e uma aplicação de quatro camadas faz uso de *web services*.

Para o estudo de grades computacionais o modelo apresentado por Gorton et al. [GOR08] (Figura 2) é utilizado para classificar as aplicações conforme o tipo de problema apresentado. Neste modelo as aplicações são classificadas de acordo com o tipo de problema a ser resolvido, considerando a quantidade de dados, o tipo de dados, a localização dos dados e o algoritmo utilizado (complexidade computacional).

Neste modelo, as aplicações para resolver os problemas correntes estão no quadrante 1.

No quadrante 2 ficam as aplicações para resolver os problemas que necessitam de computação intensiva. As aplicações intensivas de computação são caracterizadas por uma crescente complexidade computacional. Uma das formas de resolver os problemas de aplicações intensivas de computação é a utilização de *High-Performance Computing* (HPC)

O foco no quadrante 3 são as aplicações para resolver os problemas intensivos de dados. As aplicações puramente intensivas de dados processam grandes volumes de dados, com formatos heterogêneos e distribuídos em vários locais diferentes. Estas aplicações requerem um gerenciamento dos dados, utilização de técnicas de filtragem e fusão dos dados, e distribuição e localização dos dados eficientemente.

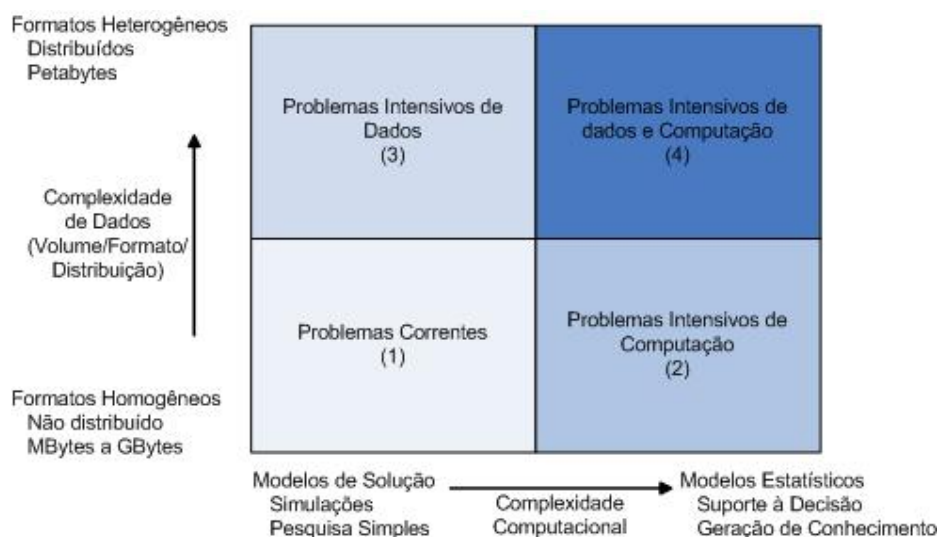


Figura 2 - Classificação de Problemas [GOR08]

Observa-se uma clara sobreposição entre aplicações intensivas de dados e aplicações intensivas de computação no quadrante 4. As aplicações intensivas de computação e de dados combinam a necessidade de processar grandes volumes de dados com crescente complexidade computacional. Estas aplicações requerem novos algoritmos, geração de assinaturas e plataformas especializadas de processamento.

Segundo Gorton et al. [GOR08] são dois os maiores desafios para as aplicações intensivas de dados. O primeiro é o gerenciamento e processamento de um volume de dados (normalmente são sensíveis ao tempo) que cresce de forma exponencial, seja por coleta de sensores e instrumentos, seja da saída de simuladores. O segundo é o tempo e a redução do ciclo de análise dos dados para que sejam utilizados para uma tomada de decisão mais rápida.

3.5. *Middleware* para Grades Computacionais

Nesta seção estão descritos alguns exemplos de *middleware* para grades, mais citados nas referências estudadas. Um *middleware* para grade computacional fornece o meio necessário para conectar as aplicações e usuários à infra-estrutura da grade.

3.5.1. Globus Toolkit

De acordo com a equipe do projeto [GLO08], o Globus ToolKit é um conjunto de ferramentas que inclui serviços e bibliotecas de *software* para descoberta, gerenciamento

e monitoração de recursos bem como gerenciamento de arquivos e segurança que podem ser utilizados para desenvolver aplicações de forma individual ou colaborativa. O Globus ToolKit foi concebido para eliminar os obstáculos que impedem a perfeita colaboração. Os serviços, interfaces e protocolos do Globus permitem aos usuários o acesso a recursos remotos como se estivessem acessando-os localmente e o administrador local determina quem e quando podem utilizar os recursos da sua grade.

Segundo Foster [FOS06], o Globus ToolKit 4 faz uso extensivo de serviços WEB para definir sua interface e estruturar seus componentes. O XML (*Extensible Markup Language*) é utilizado para a descrição, descoberta e invocação de serviços de rede. A arquitetura do Globus ToolKit 4 é formada por 3 componentes: *service implementations*, *containers* e bibliotecas de clientes.

O conjunto de "*service implementations*" implementa uma infra-estrutura de serviços usável. Os serviços fornecidos são: gerenciamento da execução (GRAM), movimentação e acesso aos dados (GridFTP, RFT, OGSA-DAI), gerenciamento de réplicas (RLS, DRS), monitoração e descoberta (Index, Trigger, WebMDS), etc. Três "*containers*" podem usar serviços desenvolvidos para usuários escritos em linguagem Java, Phyton e C. Estes *containers* fornecem implementações de segurança, gerenciamento, descoberta, gerenciamento de estado e outros mecanismos, estendendo o ambiente de serviços WEB. O conjunto de "bibliotecas de clientes" permite que programas clientes escritos em Java, Phyton e C chamem operações no Globus ToolKit 4 e nos serviços fornecidos pelos usuários.

Fazendo uso de mecanismos e abstrações uniformes o Globus Toolkit 4 permite aos clientes interagirem com serviços diferentes de formas similares. O Globus ToolKit e sua documentação podem ser acessados através do endereço <http://www.globus.org/toolkit/>.

3.5.2. Condor

Para a equipe de desenvolvimento do Condor [CON08], ele é um sistema de gerenciamento de trabalho especializado para tarefas de computações intensivas. O Condor fornece mecanismos de enfileiramento de tarefas, políticas de escalonamento, esquemas de priorização, monitoramento e gerenciamento de recursos. As tarefas são executadas de forma serial ou paralela no Condor. O Condor coloca as tarefas em uma fila, define quando e onde as tarefas serão executadas de acordo com uma política, acompanha atentamente os seus progressos e, por fim, informa o usuário após a conclusão das tarefas.

Segundo a equipe do Condor [CON08], ele pode ser usado para gerenciar um cluster com nós dedicados de computação e aproveitar mais eficazmente o poder das CPU desperdiçadas pelas estações ativas. Além disso, o Condor não exige nenhum sistema de arquivos compartilhado entre máquinas diferentes.

De acordo com a equipe do Condor [CON08], a ferramenta ClassAd fornece um framework extremamente flexível e expressivo para associar pedidos de recursos (tarefas) com recursos oferecidos (máquinas). As tarefas podem facilmente condicionar seus requisitos e preferências. Do mesmo modo, as máquinas podem especificar requisitos e preferências sobre as tarefas que eles irão executar. Estes requisitos e preferências podem ser descritos com expressões poderosas, resultando na adaptação do Condor para praticamente qualquer política desejada.

O *software* e a documentação completa do Condor estão disponíveis gratuitamente a partir do sítio do projeto no endereço <http://www.cs.wisc.edu/condor>. O *software* suporta a maioria das versões de Unix e o Windows NT/2000.

3.5.3. OurGrid

O OurGrid [Labs of the World, Unite!!!] tem por objetivo desenvolver uma solução simples, efetiva e aberta para aplicações *Bag-of-Tasks*. Segundo Cirne et al. [CIR03], aplicações "Saco de Trabalhos" ou *Bag-of-Tasks* (BoT) são aquelas aplicações paralelas cujas tarefas são independentes uma das outras. Aplicações "Saco de Trabalhos" são simples e podem ser executadas em grades computacionais distribuídas geograficamente, devido à independência de suas tarefas.

A solução OurGrid em sua versão 4 é composta por quatro componentes: o Broker, o Peer, o Worker e o servidor de comunicação XMPP. O Broker escalona aplicações "Saco de Trabalhos" e permite a seu usuário combinar todas as máquinas (Worker) a que ele tem acesso em sua grade. A comunidade OurGrid forma uma rede *peer-to-peer* que promove o compartilhamento de recursos em escala mundial.

Todos os componentes do OurGrid são *software* livre. Ainda mais importante, os usuários interessados podem se juntar à grade aberta OurGrid, a primeira grade onde a adesão de novos membros é totalmente livre. O download do *software* e a adesão à grade aberta são feitas na página <http://www.ourgrid.org> [OUR07].

Para executar uma aplicação é necessário que recursos computacionais sejam disponibilizados. No contexto de grades a concessão automática de acessos é uma característica importante para aplicações "Saco de Trabalhos" devido ao baixo

acoplamento das tarefas. A solução adotada pelo OurGrid que garante o acesso automático aos recursos das grades é a rede de favores [OUR07]. Os sítios (Peer) que fazem parte da comunidade OurGrid disponibilizam seus recursos ociosos para os membros da comunidade. O estado dos sítios conectados à comunidade OurGrid pode ser acessado no endereço <http://status.ourgrid.org/>.

Segundo Cirne et al. [CIR03], o objetivo do OurGrid é disponibilizar aos usuários das aplicações "Saco de Trabalhos" fácil acesso e uso dos recursos computacionais de forma dinâmica formando uma grande grade computacional disponível. A rede de favores do OurGrid foi projetada com os seguintes objetivos: promover igualdade na alocação e disponibilização de recursos, e priorizar os sítios que mais contribuem com a comunidade.

Além disso, o Ourgrid implementa soluções de segurança na comunicação entre os componentes da grade, no armazenamento dos dados e no acesso aos recursos dos nós remotos.

As características apresentadas pelo Ourgrid como a execução de aplicações *Bag-of-Tasks*, a "Rede de Favores", ser baseada em Software Livre e disponibilizar mecanismos de Segurança viabilizam o uso deste *middleware* de grade nos estudos.

3.5.4. Legion

Segundo Natrajan et al. [NAT08], o portal de grades Legion é uma interface para um sistema de grades. Os usuários interagem com o portal através de uma interface intuitiva a partir da qual podem visualizar seus arquivos, executar e monitorar seus processos e visualizar suas informações. A arquitetura do portal é projetada para acomodar múltiplas infra-estruturas de grades diferentes, sistemas legados e interfaces específicas de aplicações.

De acordo com a equipe do projeto [NAT08], o portal opera inteiramente num servidor Web e não é necessário baixar ou instalar qualquer *software* do Legion na máquina do usuário. Na máquina do cliente é necessário somente de um browser instalado. Os Portais Especiais são interfaces específicas de aplicações suportadas pelo Legion. Os principais serviços de grade suportados pelo Portal de Grades Legion são: segurança, escalonamento, transferência de dados e serviços adicionais.

No serviço de Segurança o usuário faz uso de um mecanismo de acesso e senha para acesso ao portal combinado com a geração de credenciais de sessão baseadas em pares de chaves pública/privada. O serviço de Escalonamento fornece suporte a escalonamento implícito e explícito. No escalonamento explícito o usuário determina em

qual host ele deseja executar suas aplicações. No escalonamento implícito, esta tarefa cabe à infra-estrutura da grade. O serviço de Transferência de dados fornece suporte à transferência de dados para as aplicações. Os Serviços adicionais fornecem facilidades de monitoração de tarefas, visualização do progresso das tarefas, transferência imediata de arquivos e visualização de informações de log.

O Portal de Grades Legion pode ser acessado através do sítio <http://legion.virginia.edu/index.html>.

3.6. Exemplos de Aplicações para Grades Computacionais

Esta seção descreve rapidamente exemplos de alguns projetos e soluções desenvolvidas para o emprego da tecnologia de grades computacionais.

3.6.1. A Comunidade Ourgrid

A comunidade Ourgrid reúne todos os usuários e desenvolvedores do *middleware* OurGrid. O Ourgrid é uma grade computacional lançada em dezembro de 2004 utilizada para execução de aplicações *bag-of-tasks*. O *software* é livre e possui código aberto (distribuído sob a GPL). O *software* é desenvolvido pelo Laboratório de Sistemas Distribuídos (LSD) da Universidade Federal de Campina Grande (UFCG) e conta com o apoio da HP Brazil R&D desde 2003.

3.6.1.1. GerpavGrid

Segundo De Rose et al. [DER06], O GerpavGrid é uma solução desenvolvida pela PUCRS em parceria com outras instituições com objetivo de desenvolver uma solução de grade para a gerência de pavimentos da cidade de Porto Alegre.

O sistema original denominado Gerpav era altamente acoplado com o sistema gerenciador de banco de dados. O GerpavGrid foi implementado como uma extensão ao sistema original e devido ao alto acoplamento da solução foi necessário o desenvolvimento de novas funcionalidades e reescrita de outras partes do sistema para que os objetivos fossem atingidos.

O projeto foi desenvolvido tendo o OurGrid como a infra-estrutura de grade. A utilização de grades de computadores para processar os dados relativos ao piso asfáltico da cidade de Porto Alegre proporcionou ganhos de desempenho de até 80 vezes na

execução da aplicação viabilizando a análise da malha viária da cidade. Como benefício possibilitou a otimização da aplicação dos recursos públicos nas obras de manutenção da malha viária melhorando a qualidade do serviço público prestado. O GerpavGrid mostrou que é viável se utilizar grades computacionais em sistemas de gestão pública com ganhos significativos de desempenho.

O projeto resultou na criação do OJAL (*OurGrid Job Abstraction Layer*) utilizado para escalonar as tarefas para os recursos da grade. Resultou também no desenvolvimento de estratégias para acesso ao banco de dados e escalonamento de tarefas como: *Database Filter*, *Memory Filter*, *Simple Slicing*, *Pipeline Dispatching* e *Distributed Database* [DER06].

3.6.1.2. SegHidro

O SegHidro é uma rede de compartilhamento de dados, conhecimento e processamento em Hidrometeorologia. Segundo o sítio do projeto [SEG09], o SegHidro “propõe uma infra-estrutura de tecnologia para viabilizar o compartilhamento de dados, conhecimento e poder computacional a serviço de uma melhor gestão dos recursos hídricos do país” e pode ser acessado no endereço <http://seghidro.lsd.ufcg.edu.br/>.

O SegHidro provê duas formas de acesso à aplicação. Uma através do Portal SegHidro usando acesso Web para execução de tarefas rotineiras e a outra através do Toolkit SegHidro que oferece um acesso programável para pesquisadores.

O SegHidro utiliza a grade computacional para a gestão de recursos hídricos. Esta área requer vários especialistas para o desenvolvimento de soluções de seus problemas e a computação de alto desempenho é necessária devido à natureza da aplicação.

O OurGrid viabiliza a infra-estrutura necessária para a computação de alto desempenho que o SegHidro necessita. O framework OPeNDAP foi adotado para o compartilhamento de dados por fornecer independência de formatos e facilidade na separação de dados com base nos parâmetros analisados.

O projeto encontra-se numa segunda fase, designado de SegHidro2, e tem por objetivo o desenvolvimento de uma plataforma para executar aplicações de apoio à gestão sustentável de recursos hídricos utilizando grades computacionais.

3.6.2. Open Science Grid (OSG)

Segundo Pordes [POR08b] a Open Science Grid (OSG) [TER09] foi fundada pela National Science Foundation (NSF) [NSF09] e o U.S. Department of Energy (DOE)

[DOE09] com objetivo de fornecer uma infra-estrutura (recursos de rede, processamento e armazenamento) computacional distribuída nacionalmente para pesquisa científica nos Estados Unidos. O OSG também serve como gateway com outras infra-estruturas de grades regionais e internacionais.

As aplicações (do tipo simulação, processamento de grande volume de dados, workflow complexo, tempo real, paralelismo [POR08a]) são a base para o crescimento da OSG. As principais aplicações são nas experiências com *Large Hadron Collider* (LHC), ATLAS, CMS, LIGO, STAR e Tevatron [POR07, POR08a].

O escopo da OSG para Pordes [POR07] se resume a disponibilizar o maior percentual de recursos computacionais e de armazenamento aos pesquisadores, facilitar o uso de sistemas distribuídos pelos pesquisadores, incentivar o compartilhamento e o reuso de programas, estabelecer uma comunidade aberta para troca de experiências e conhecimento facilitando a integração dos novos participantes.

A OSG usa como *middleware* o Virtual Data Toolkit (VDT) [VDT09] que é um conjunto de programas integrados e distribuídos. O VDT se integra a outros *middleware* de grade como o Condor e o Globus e suporta diversos programas abertos como Apache, Tomcat e MySQL.

A OSG agrega mais de 43.000 processadores (usando Linux em sua maioria). São membros do consórcio da OSG 72 instituições de diversos países. No Brasil o consórcio da OSG possui dois membros: a Universidade de São Paulo e a Universidade do Estado do Rio de Janeiro.

3.6.2.1. Reprocessamento de dados para DZero

O experimento DZero é um projeto desenvolvido pelo Fermi National Accelerator Laboratory (Fermilab) no Tevatron Collider com objetivo de estudar as interações de prótons e anti-prótons sob alta energia procurando descobrir a constituição dos elementos que formam o Universo. [66]

Segundo Abbott [ABO07], em 2007 houve a necessidade de reprocessamento de dados do DZero com a utilização de parte dos recursos do projeto e demais advindos da OSG. Um total de 90 TB de dados precisou ser reprocessado. A aplicação ocupava 1 GB e o sistema como um todo 250 TB e os nós da OSG não eram dedicados à aplicação, sendo assim não era possível replicar a aplicação toda nos nós da OSG. Considerada como um problema de otimização foi necessário determinar quais recursos da OSG seriam necessários para executar a aplicação. O objetivo foi encontrar uma solução que

maximizasse o número de recursos disponíveis minimizando o número de sítios no sistema e maximizando a banda de interconexão entre os sítios.

De acordo com Abbott [ABO07] para o reprocessamento foram consumidos 500 GHz de CPUs ao ano requisitando da OSG uma média de 1.500 CPUs por dia por 4 meses. Foram registradas falhas nas tarefas (50%) por problemas de configuração (acesso, direitos bibliotecas incompatíveis, falha de *middleware*, grade fora do ar) ou problemas de entrega dos dados (conectividade).

O projeto foi finalizado com sucesso e seus resultados apresentados no *International Conference on Computing in High Energy and Nuclear Physics* (CHEP07).

3.6.3. Enabling Grids for E-science (EGEE)

O EGEE [EGE09] é um projeto de Grade que fornece infra-estrutura para mais de 10.000 pesquisadores no mundo todo das mais diversas áreas como física de alta energia, ciências da saúde e da terra. A primeira fase do projeto reuniu especialistas de mais de 27 países com objetivo de construir uma infra-estrutura de grade de serviços disponível 24 horas por dia disponibilizando acesso a recursos computacionais para pesquisadores independente de sua localização geográfica com isso atraindo uma grande quantidade de novos usuários à grade.

A primeira fase do projeto EGEE foi finalizada em 31 de março de 2006. A segunda fase do projeto EGEE foi iniciada em 01 de abril 2006 e finalizada em 30 de Abril de 2008. A terceira fase do projeto EGEE foi iniciada em 01 de maio de 2008.

O EGEE usa o *middleware* gLite desenvolvido, certificado e distribuído pelo próprio projeto [GLI09] e segue os padrões definidos pelo Open Grid Forum (OGF) [OGF09] para manter a interoperabilidade com outras infra-estruturas de grades.

Atualmente o projeto conta com a participação de mais de 50 países totalizando 40.000 processadores e vários petabytes de armazenamento. Informações sobre o projeto podem ser localizadas no sítio <http://www.eu-egee.org/>.

3.7. Considerações Finais sobre Grades Computacionais

Existem outros projetos de infra-estruturas de grades no mundo como: E-science grid facility for Europe and Latin America - EELA-2 (Europa e América Latina), D-Grid (Alemanha), National Grid Service (Inglaterra), etc. [WIK09] possibilitando a execução das

mais diversas aplicações com aproveitamento dos recursos computacionais distribuídos geograficamente.

As aplicações para e-Gov usando grades estão sendo desenvolvidas em várias partes do mundo para resolver problemas específicos e podem ser vistas em [FUX05, ARC01, MAU03, SUM08]. Os portais de e-Gov podem usar as mesmas características existentes nos portais de grades [MAA08]. Existe ainda uma grande lacuna entre a pesquisa e a prática nesta área, ou seja, existe um enorme potencial de pesquisa no desenvolvimento de soluções de grades para e-Gov.

As grades computacionais tornam-se uma nova opção tecnológica para a execução de aplicações de e-Gov. Elas possibilitam a utilização de recursos distribuídos geograficamente de forma transparente e segura, garantindo um ambiente para a execução de aplicações paralelas a um custo mais baixo que as soluções empregadas atualmente pelo Governo Federal.

Existem vários *middleware* de grade desenvolvidos como sistemas abertos que executam sobre sistemas operacionais livres e utilizam linguagem de programação portátil e banco de dados aberto.

O emprego da tecnologia de grades no atendimento dos requisitos de e-Gov são apresentados no próximo capítulo. São apresentadas também, propostas de emprego da tecnologia de grades em algumas aplicações de e-Gov.

4. GRADES ATENDENDO E-GOV

Para promover a universalização do acesso aos serviços governamentais, a integração de redes e o alto desempenho dos seus sistemas o Governo Federal definiu requisitos básicos para a implantação do Governo Eletrônico.

As grades computacionais disponibilizam recursos distribuídos de forma transparente, proporcionando um grande poder computacional, para a execução de aplicações. Os exemplos, apresentados no capítulo anterior, mostram que é possível empregar tecnologias não proprietárias e recursos computacionalmente ociosos para o processamento de vários tipos de aplicação.

Sendo as grades uma opção tecnológica para execução de aplicações de e-Gov de maneira eficiente e segura, este capítulo apresenta uma análise teórica de como elas podem contribuir para o atendimento de alguns requisitos de Governo Eletrônico.

Ao final do capítulo são apresentadas alternativas para processamento de aplicações de e-Gov utilizando grades computacionais procurando mostrar as vantagens proporcionadas pela utilização desta tecnologia.

4.1. Grades Atendendo os Requisitos de e-Gov

Nesta seção são apresentados os requisitos básicos do Governo Eletrônico que se relacionam com a utilização de grades computacionais para processamento de aplicações de Governo Eletrônico. Os demais requisitos não possuem uma clara relação com grades computacionais e não serão abordados.

4.1.1. Adoção de Políticas, Normas e Padrões Comuns

No documento de referência da e-PING [MPO07e] estão definidas um conjunto de políticas norteadoras para estabelecimento das especificações dos seus componentes, adotando preferencialmente a utilização de padrões abertos para as especificações técnicas. As áreas tratadas no documento são: Interconexão, Segurança, Meios de Acesso, Organização e Intercâmbio de Informações, e Áreas de Integração para Governo Eletrônico.

Dentre as políticas gerais da e-PING [MPO07e], algumas norteiam o desenvolvimento de sistemas de grades, são elas: os sistemas de e-Gov devem estar

alinhados com as especificações usadas na Internet; o padrão primário para o intercâmbio de dados adotado deve ser o XML (*eXtensible Markup Language*); o Padrão de Metadados do Governo Eletrônico (e-PMG) a ser adotado deve ser desenvolvido com base em padrões internacionalmente aceitos; os padrões adotados devem garantir a escalabilidade em termos de quantidade de usuários, volume de dados processado e quantidade de transações executadas; e a preferência é pela adoção de padrões abertos nas especificações técnicas, exceto nos casos em que requisitos de segurança e integridade de informações sejam críticos.

O documento também faz referência às tecnologias que podem ser utilizadas para computação distribuída, como a utilização de *Web Services* e de Arquitetura Orientada a Serviços (SOA).

As tecnologias de grades existentes hoje estão maduras, são usadas largamente por diversos segmentos da sociedade e fornecem a base para estas padronizações. Assim, através das políticas de padronização, o Governo Eletrônico cria mecanismos para a estruturação dos sistemas em computação distribuída entre os Órgãos do Governo obtendo maiores vantagens das arquiteturas de grades.

4.1.2. Utilização do *Software* Livre como Recurso Estratégico

Como opção tecnológica do Governo Federal, deve-se priorizar soluções, programas e serviços baseados em *software* livre que promovam a otimização de recursos e investimentos em tecnologia da informação. A motivação para a utilização do *software* livre está baseada em aspectos econômicos, na produção e circulação de conhecimento, no acesso às novas tecnologias, no estímulo ao desenvolvimento de *software* em ambientes colaborativos e no estímulo ao desenvolvimento de *software* nacional.

As grandes empresas de processamento de dados do Governo Federal, como Dataprev e SERPRO, e outras entidades do Governo Federal, têm investido no desenvolvimento de aplicações livres, como por exemplo: o CACIC (Configurador Automático e Coletor de Informações Computacionais), o COCAR (Controlador Centralizado de Ambiente de Rede), SAGUI (Sistema de Apoio à Gerência Unificada de Informações), SGD (Sistema de Gestão de Demandas), etc. Estas aplicações estão disponíveis para utilização no Portal de *software* Público Brasileiro [GFB08b].

A chave para atingir este requisito é focar na utilização de *middleware* de grades abertos, sistemas operacionais abertos (distribuições Linux, FreeBSD), banco de dados abertos (PostgreSQL e MySQL), linguagens de programação portáteis (Java), em

detrimento de soluções proprietárias e plataformas fechadas garantindo ao cidadão o direito de acesso aos serviços públicos sem obrigá-lo a usar plataformas específicas.

4.1.3. Racionalização dos Recursos

A redução de custos unitários e racionalização do uso de recursos devem ser proporcionadas pelo Governo Eletrônico através do compartilhamento de recursos entre órgãos públicos no desenvolvimento de soluções (desenvolvimento compartilhado em ambiente colaborativo, envolvendo múltiplas organizações), na operação de soluções e no compartilhamento de equipamentos e recursos humanos.

Este requisito pode ser atingido pelo uso de grades computacionais, mas devem-se considerar algumas restrições como a maturidade das organizações (e de seus dirigentes) em compartilhar recursos (equipamentos, redes, pessoas), o custo adicional introduzido pela implementação de algumas tecnologias de grades e de redes de alta velocidade.

Facilmente constata-se que os recursos computacionais do Governo Federal são imensos e estes são utilizados em sua maioria no período de expediente normal das 08:00 às 18:00 h. Mesmo neste período ocorrem intervalos de ociosidade. Estes recursos podem ser utilizados por grades para processar aplicações de e-Gov, promovendo a utilização de métodos computacionais inovadores que reduzam custos, utilizando padrões abertos e ampliando a capacidade de negociação do Governo Federal de custos de licenças de *software* e aquisição de equipamentos.

4.2. Utilizando grades em Aplicações de E-Gov

Em 2006, estimava-se que a administração pública direta do Governo Federal possuía mais de 300 mil estações de trabalho [MPO07a] em seu parque computacional distribuídas pelo país e interligadas por redes de comunicação de dados (Infovia, rede do SERPRO, rede da Previdência, etc.). A tecnologia de grades poderia ser aplicada em sistemas de e-Gov com objetivo de racionalizar a utilização destes recursos computacionais.

As aplicações de Governo Eletrônico se caracterizam pelos benefícios que propiciam para os cidadãos. Nesta seção propõem-se alternativas para a execução de aplicações de e-Gov, otimizando-as para rodar em paralelo em grades computacionais,

atendendo às demandas computacionais existentes das instituições, com o melhor custo-benefício e o menor risco possível à continuidade do negócio.

4.2.1. Qualificação de Informações do CNIS

A base de dados do CNIS é usada para comprovar os vínculos empregatícios e contribuições do trabalhador. Quando um trabalhador solicita uma aposentadoria, a base do CNIS é consultada para comprovar o direito a este benefício perante a Previdência Social.

Os recursos de uma grade computacional podem ser utilizados para realizar a consolidação dos dados deste imenso banco de dados caracterizado pela diversidade das fontes de dados de origem. De acordo com Favaretto [FAV05],

Segundo Burgess et al. (2004), devido ao grande volume de informação disponível atualmente, a qualidade se tornou um importante fator de escolha entre qual informação utilizar e qual descartar. Lee e Siau (2001) também afirmam que diante da enorme quantidade de dados disponíveis são necessárias ferramentas que gerem informações úteis.

Nossa proposta é processar os dados recebidos de fontes externas para qualificar os campos de endereços usando a padronização dos Correios num ambiente de grades computacionais. Para este processamento será necessário transmitir a tabela de CEP dos correios para todos os nós da grade. Ao final do processamento espera-se obter um ou mais campos de endereços (nome logradouro, número, CEP, bairro, cidade, estado) atualizados com numeração e nomenclatura em conformidade com o padrão dos Correios.

A Figura 3 ilustra a proposta para a qualificação de informações do CNIS em ambiente de grades. A base de dados do CNIS fica armazenada no servidor CV2. A aplicação CNIS é acessada pelos usuários através de um programa cliente nas Agências da Previdência Social e nas Gerências Executivas do INSS. O programa cliente se comunica com o banco de dados do CNIS através do servidor MPACT. As informações do CNIS também estão disponíveis na página do Ministério da Previdência Social para todos os cidadãos. O servidor de comunicação APL09 faz a interface com os principais sistemas previdenciários (Prisma, SUB e SABI).

Os dados enviados, usando padrão XML, pelas entidades que participam do projeto CNIS são recebidas na Dataprev por meio eletrônico. O servidor de Qualificação separa os dados e envia para a grade. Na grade os dados Unitários são processados e

qualificados pelos nós (estações do INSS). Os dados qualificados são devolvidos pela grade à aplicação de qualificação. Os dados são consolidados e armazenados no servidor CNIS.

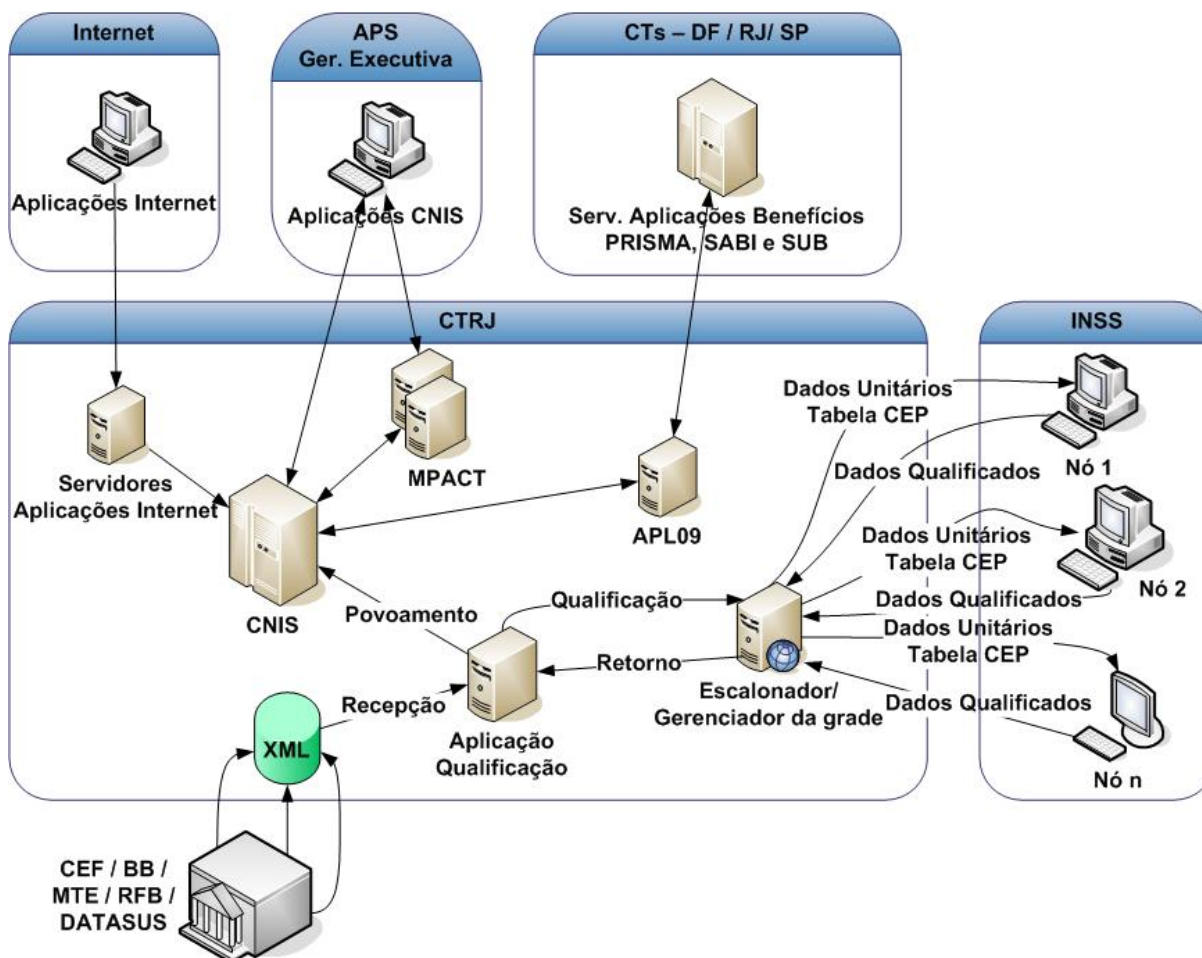


Figura 3 - Proposta de Arquitetura para a Qualificação de Dados do CNIS

Com esta proposta objetiva-se:

- Reduzir a dependência em relação ao computador de grande porte utilizado no processamento das requisições do sistema CNIS;
- Diminuir a sobrecarga à qual o equipamento que abriga o sistema CNIS é submetido;
- Otimizar a utilização dos recursos computacionais existentes na rede da Previdência Social;
- Melhorar a qualidade da informação armazenada no banco de dados do CNIS;
- Diminuir o tempo de indisponibilidade dos sistemas previdenciários;
- Diminuir o tempo de atendimento nas Agências da Previdência Social;
- Atender aos requisitos de e-Gov.

4.2.2. Processamento dos dados do Sistema Único de Benefícios (SUB)

O processamento de dados do Sistema Único de Benefícios consome recursos substanciais dos computadores de grande porte da Dataprev. Entre os processos que utilizam estes recursos destacam-se o processamento da Maciça, do Histórico de Créditos dos Beneficiários (HISCRE), da Declaração Anual de Rendimentos (Imposto de Renda) e do Pagamento Alternativo de Benefícios (PAB).

Conforme apresentados acima inúmeros são os serviços agregados ao Sistema Único de Benefícios. Para processar as informações do SUB propõe-se a utilização de uma grade computacional para a execução de tarefas que demandem alto grau de processamento e recursos do computador de grande porte. Por exemplo, o processamento da maciça seria executado na grade computacional liberando os recursos escassos do computador de grande porte para atendimento às requisições dos usuários.

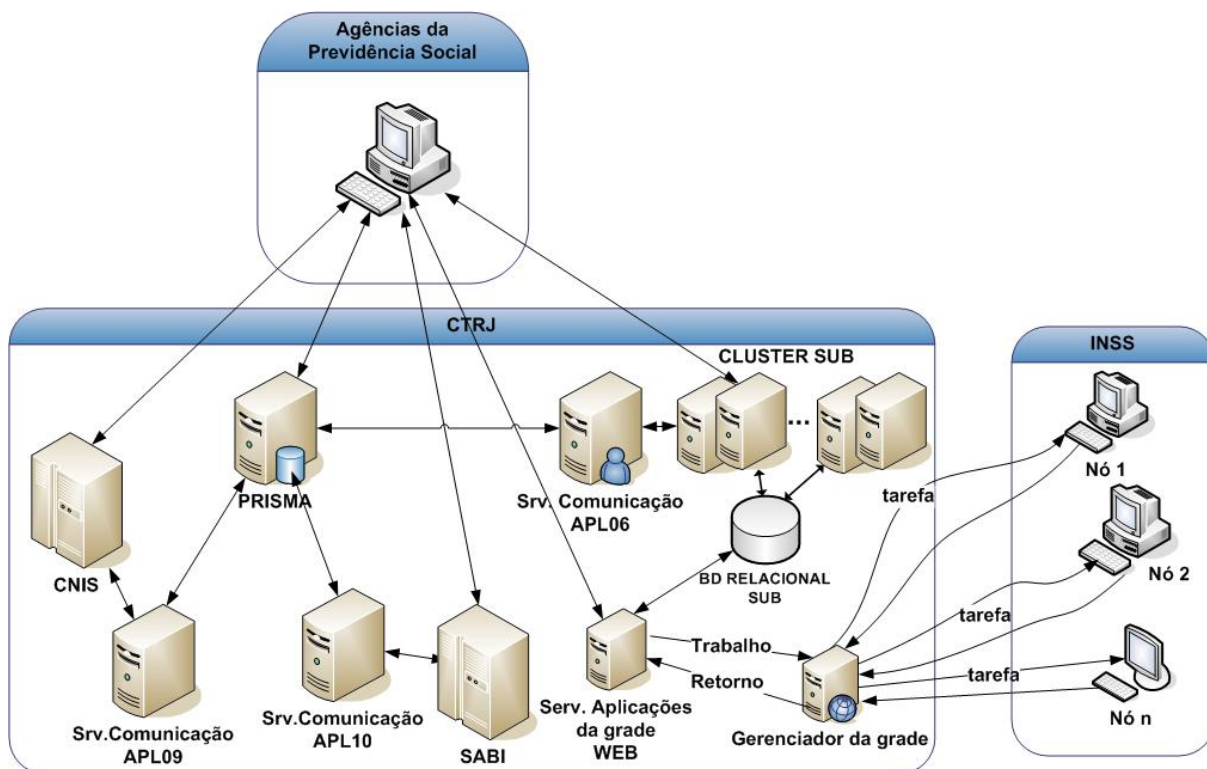


Figura 4 - Proposta de Arquitetura para o Sistema Único de Benefícios

Num segundo momento, o computador de grande porte deve ser substituído por equipamentos de menor porte agregados em forma de cluster conectados a um sistema de armazenamento independente.

A Figura 4 apresenta a solução proposta. Os sistemas previdenciários são acessados pelo usuário do INSS nas Agências da Previdência Social. Os servidores de comunicação APL06, APL09 e APL10 são responsáveis pela interface entre estes sistemas. Nesta arquitetura, o Mainframe que hospeda o sistema SUB será substituído por um cluster de servidores de pequeno porte. Os dados serão armazenados num sistema de armazenamento de discos de alta performance utilizando um banco de dados relacional. Tarefas que necessitem de grande poder de processamento utilizarão os recursos da grade computacional para serem executadas.

Com esta solução objetiva-se:

- Reduzir a dependência em relação ao computador de grande utilizado no processamento das requisições do sistema SUB;
- Diminuir a sobrecarga a qual o equipamento que abriga o sistema SUB é submetido;
- Otimizar a utilização dos recursos computacionais existentes na rede da Previdência Social;
- Diminuir o tempo de indisponibilidade dos sistemas previdenciários;
- Diminuir o tempo de atendimento nas Agências da Previdência Social;
- Atender aos requisitos de e-Gov;
- Independência de fornecedores de *software* e *hardware*.

Para que os objetivos desta proposta sejam atingidos propõe-se a utilização de servidores com sistema operacional Linux para o cluster e para a grade, sistema gerenciador de banco de dados relacional PostgreSQL para armazenamento de dados do SUB, linguagem de programação Java para o desenvolvimento do sistema, servidor de apresentação WEB usando Apache.

A arquitetura proposta facilita o gerenciamento e a diminuição de custos, possibilitando o crescimento da arquitetura de cluster e de grades à medida que mais recursos são adicionados.

4.2.3. PRISMA

O Sistema PRISMA está disponível em todas as agências da Previdência Social, sendo utilizado diariamente como ferramenta básica para a habilitação, concessão, atualização e revisão de benefícios previdenciários.

Inicialmente a transmissão de dados entre os Postos de Benefícios e a Dataprev teve por finalidade permitir que o processamento centralizado na Dataprev fosse

executado a partir dos dados enviados diretamente das Agências de benefícios e que o retorno, resultante deste processamento, fosse refletido também na base de dados local, em cada Agência. O então sistema de comunicação proprietário Transfer/GTX (via RENPAC) permitiu a ligação de cada Agência de Benefícios ao computador central da Dataprev, transmitindo as informações através de linha discada. O sistema operacional PICK/AP foi o escolhido inicialmente para a implantação do Sistema.

A partir de 1997, a Dataprev iniciou um processo de migração do ambiente inicial com adoção do protocolo de comunicação TCP/IP e do ambiente PICK/D3 com sistema operacional SCO OpenServer 5, com o objetivo de fornecer melhor solução e rapidez ao processo de comunicação, além de estabilidade do ambiente.

Este ambiente estava instalado em cada uma das 1.200 unidades de atendimento do INSS até o ano de 2000. A partir desta data iniciou-se um processo de concentração das Bases D3 no ambiente centralizado nas Unidades Regionais de Atendimento, diminuindo o sensivelmente o número de servidores, melhorando o suporte aos sistemas e a capacidade dos equipamentos servidores. Com a concentração de bases foi possível melhorar a segurança dos dados através da utilização de mecanismo de replicação do banco de dados on-line.

A partir de 2005, um processo de centralização culminou com a hospedagem dos servidores nos três grandes Centros de Processamento localizados no Rio de Janeiro (CPRJ), em São Paulo (CPSP) e no distrito Federal (CPDF). Atualmente, o sistema Prisma está distribuído em mais de 100 servidores. O critério para distribuição dos servidores leva em consideração a topologia da rede (influenciado pela concessão dos serviços de telecomunicações controlados pela ANATEL), privilegiando as ligações diretas entre as Agências da Previdência Social (APS) e os Centros de Processamento.

A licença de *software* do banco de dados D3 adquirida pela Dataprev para suportar a aplicação possui limitação na quantidade de usuários ativos simultaneamente. Consequentemente precisam-se de centenas de máquinas para armazenar e processar os dados deste sistema. Estes equipamentos ocupam grandes espaços nos Centros de Processamento e apresentam alto custo de manutenção.

Para solucionar este problema, pode-se:

- Utilizar a virtualização para consolidar os diversos servidores físicos em equipamentos atualizados tecnologicamente com maior capacidade de processamento e menores custos operacionais e de manutenção, visto que a empresa possui equipamentos que atendem aos requisitos exigidos.

- Portar a aplicação para um ambiente de processamento paralelo, com a utilização de clusters ou grades de dados, com utilização de ferramentas de código aberto para o desenvolvimento e processamento das aplicações e também para o armazenamento dos dados.

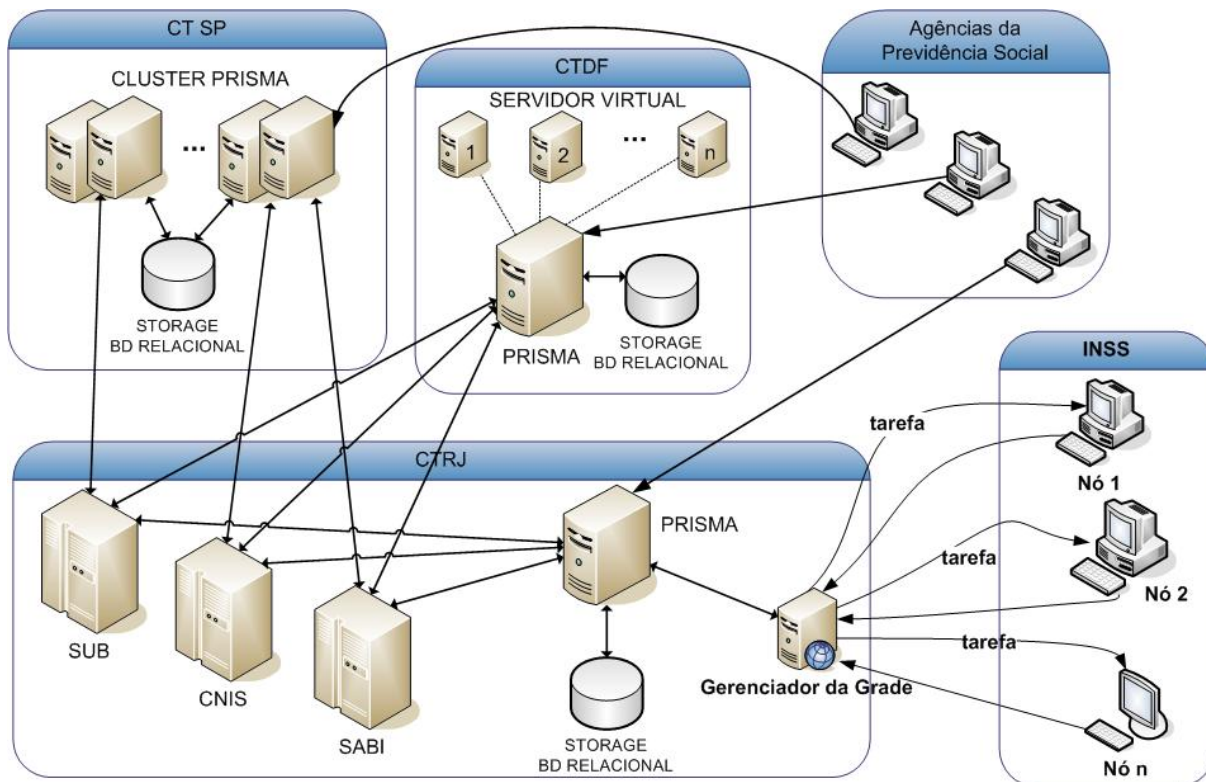


Figura 5 - Proposta de Arquitetura para o Sistema Prisma

A Figura 5 mostra a proposta para a arquitetura do sistema PRISMA. Nesta arquitetura estão representadas as três abordagens. Na primeira denominada “CLUSTER PRISMA”, servidores serão agrupados em forma de cluster para hospedar o sistema Prisma acessando um banco de dados relacional. Na segunda abordagem, denominada “SERVIDOR VIRTUAL” são utilizados servidores com maior capacidade de processamento, memória e disco para agrupar os servidores atuais com uso de técnicas de virtualização. Na terceira abordagem, o sistema Prisma utilizará os recursos de uma grade computacional para processar os dados.

Na primeira e terceira abordagens será necessário adaptar a aplicação à nova infraestrutura e será utilizada interface web para acesso à aplicação PRISMA e o banco de dados será relacional. Na segunda abordagem, a mudança será transparente para usuário, não havendo necessidade de reescrever a aplicação.

Os servidores serão dotados com sistema operacional Linux para o cluster e para a grade, sistema gerenciador de banco de dados relacional PostgreSQL, linguagem de programação Java, servidor de apresentação WEB usando Apache.

Com esta solução objetiva-se a alcançar a independência de fornecedores de *software* e *hardware*, diminuir os custos de manutenção de *hardware* e operação dos servidores, otimizar a utilização dos recursos computacionais existentes na rede da Previdência Social, aumentar a capacidade de processamento do sistema e diminuir o tempo de atendimento nas Agências da Previdência Social procurando atender aos requisitos de e-Gov.

4.2.4. Justiça Eleitoral - Análise da Base de Eleitores

Segundo dados do Tribunal Superior Eleitoral (TSE) [TSE08], o Brasil possui mais de 130 milhões de eleitores distribuídos pelo país. Com a implantação das urnas eletrônicas a partir de 1996, foram eliminadas fraudes cometidas no processo de apuração e totalização dos votos, e acelerou-se o processo de totalização dos votos e divulgação dos resultados.

Ainda que o processo eleitoral brasileiro tenha evoluído, a identificação dos eleitores não seguiu o mesmo caminho. Às vésperas de todas as eleições, notícias são divulgadas pela imprensa acerca de fraudes de títulos eleitorais falsos, de municípios que possuem mais eleitores que sua população, entre outros. O eleitor é identificado por um documento que contém apenas seu nome e data de nascimento que pode propiciar pequenas fraudes, como a votação no lugar de outra pessoa ou até de eleitor já falecido. Para resolver este problema o novo título que será confeccionado pelo TSE terá fotografia, impressão digital eletrônica, registro geral (RG), CPF e o tipo sanguíneo do eleitor, o qual requererá um trabalhoso processo de recadastramento de milhões de eleitores.

Para melhorar a eficácia do processo de recadastramento e da análise da base de dados de eleitores pode-se utilizar cluster ou grade computacional para o batimento dos dados do TSE com diversas bases de dados. As bases de dados que poderiam ser usadas para esta análise seriam:

- Informações estatísticas da população dos municípios do Instituto Brasileiro de Geografia e Estatística (IBGE);
- Cadastro de Contribuintes Individuais ativos da Previdência Social (MPS);
- Cadastro de Trabalhadores registrados em carteira e contribuintes ativos da Previdência Social;

- Cadastro de Pessoa Física (CPF) da Receita Federal do Brasil (RFB);
- Registro de Nascimentos e Óbitos emitidos pelos cartórios.

Na Figura 6 está representada a proposta de arquitetura para o batimento do cadastro de eleitores usando grades computacionais. A validação dos títulos eleitorais seria realizada através da verificação dos cadastros mencionados acima.

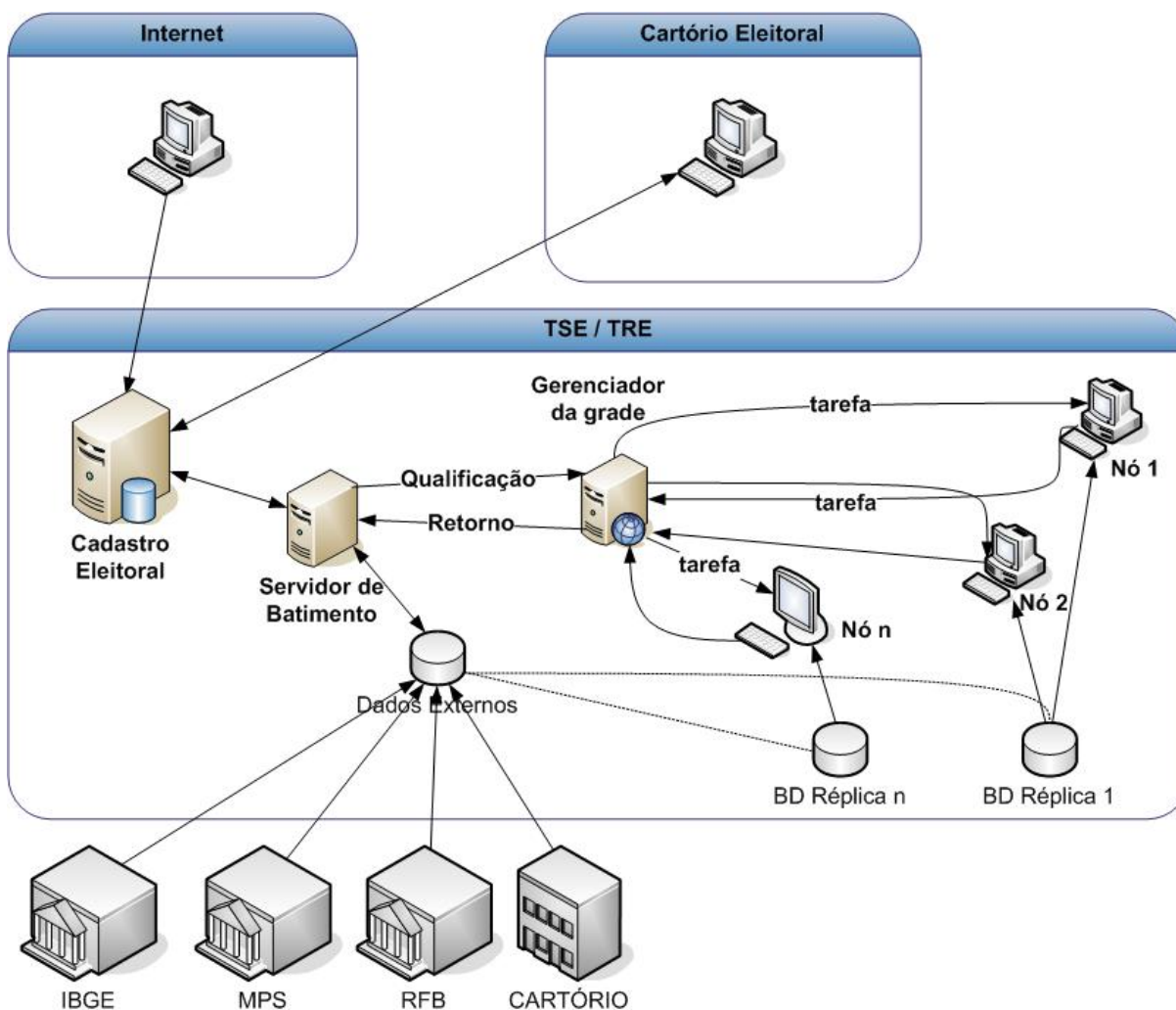


Figura 6 - Proposta de Arquitetura para o Batimento do Cadastro Eleitoral

Nesta proposta os dados externos seriam replicados em bancos de dados distribuídos nos sítios onde se encontram os nós da grade de modo a diminuir o tempo de acesso aos dados necessários para o batimento dos dados.

O resultado final da validação poderia ser consultado pela internet ou pela intranet dos cartórios eleitorais, tribunal regional eleitoral e tribunal superior eleitoral.

O batimento do cadastro de eleitores tem como objetivos:

- Diminuir a possibilidade de uma pessoa votar no lugar de outra;
- Impedir a votação com títulos de pessoas já falecidas;
- Aumentar a segurança no processo de cadastramento dos eleitores;
- Atender aos requisitos de e-Gov.

Este procedimento não quer substituir os processos de identificação do Tribunal Superior Eleitoral, é uma proposta para melhorar a qualidade dos dados armazenados nos bancos de dados dos eleitores.

Este procedimento poderia ser estendido aos programas sociais do Governo Federal. Programas como a Bolsa Família poderia ser alvo de validação de dados armazenados nos centros de tecnologia da informação do Governo Federal.

4.3. Considerações finais sobre Atendimento aos Requisitos de e-Gov

Neste estudo foram escolhidas aplicações que apresentassem dificuldades (falta de recursos computacionais, grande tempo de processamento, utilização de tecnologias legadas, etc.) no processamento dos dados e que indicassem potencial de ganho com a utilização de grades computacionais. Existem inúmeras aplicações de e-Gov, portanto, foi necessário escolher algumas para o desenvolvimento deste trabalho. O conhecimento adquirido sobre a Previdência Social foi um facilitador para a utilização de algumas aplicações previdenciárias. Aplicações de outras áreas de Governo, também, foram estudadas.

As aplicações estudadas, especificamente, apresentam algumas características comuns para a paralelização utilizando grades computacionais. São elas:

- Grossa granularidade – existência de pouca comunicação entre as tarefas;
- Processamento de uma grande quantidade de dados – as bases de dados possuem centenas de milhões de registros armazenados;
- Longo tempo de integração – apresentam um tempo de execução muito alto;
- Grande fluxo de dados - As informações são oriundas de várias fontes de dados.

Assim, as análises teóricas indicam que é viável o emprego de grades computacionais no ambiente de TIC do Governo Federal para a execução de aplicações de e-Gov beneficiando o cidadão brasileiro que necessita dos serviços públicos.

Para comprovar essa viabilidade, no próximo capítulo é apresentado um estudo de caso em que uma aplicação de e-Gov é modelada e um módulo implementado para execução em infra-estrutura de grade computacional procurando atender aos princípios do Governo Eletrônico.

5. ESTUDO DE CASO

A análise dos requisitos de e-Gov em ambiente de grades indicam que é viável atender aos requisitos do Governo Eletrônico empregando tecnologias não proprietárias e recursos computacionalmente ociosos.

Para comprovar esta viabilidade este capítulo apresenta uma descrição do processo de ETL² (Extração Transformação Carga) do sistema MAIPREV, os principais problemas, a modelagem da solução proposta e a implementação de um módulo do sistema MAIPREV utilizando os recursos de uma grade computacional.

A seção de implementação apresenta a infra-estrutura de grade utilizada nos testes, as alternativas de solução desenvolvidas e os resultados obtidos na execução da rotina na grade.

No final do capítulo são descritos alguns aspectos de segurança da solução proposta.

5.1. O Processo de ETL do Sistema MAIPREV

O Sistema MAIPREV tem como objetivo principal o cruzamento e a combinação de dados englobando as áreas de Arrecadação e de Benefício com a função de permitir a descoberta de padrões de comportamento indicativos de fraude e irregularidades. Para atingir seus objetivos, a Assessoria de Pesquisas Estratégicas (APE) utiliza-se do sistema MAIPREV que consolida as informações de diversas bases de dados previdenciários (Prisma, CNIS, SABI e SUB).

A base de dados do sistema MAIPREV conta atualmente com 370 GB de dados armazenados em banco de dados proprietário SQL Server 2000 hospedado em uma máquina rodando o sistema operacional Windows 2003 Server. Os dados são carregados no banco de dados através de processo de Extração, Transformação e Carga (ETL)

O objetivo aqui é propor uma alternativa para o processo de ETL do *datawarehouse* (DW) do MAIPREV. As grades serão o pilar para a alimentação do *datawarehouse*, fornecendo os recursos necessários para a consolidação das bases de dados, sem sobrecarregar os servidores de plataforma baixa e computadores de grande porte.

No processo de ETL os dados são extraídos de diversas fontes externas e manipulados para atender às regras de negócio e depois carregados num

² ETL - *Extract Transform Load*

datawarehouse, ou seja, representa a maneira pela qual os dados são povoados num DW.

A qualidade das informações armazenadas num DW é imprescindível para que a aplicação seja bem sucedida. Em seu estudo Favaretto [FAV05] escreveu sobre a qualidade da informação: "Mahnic e Rozanc (2001) colocam que a qualidade da informação é um pré-requisito para implantação e desenvolvimento de sistemas de informação, como por exemplo, o *datawarehouse*, pois sem este atributo as informações resultantes destes sistemas também serão de baixa qualidade".

Neste processo de ETL, a fase de Extração tem como resultado do processamento de várias extrações das bases de dados dos sistemas CNIS, PRISMA, SUB e SABI arquivos em formato texto contendo dados unitários. Este processamento é realizado de forma automatizada (execução de *jobs* e *scripts*) nos computadores de grande porte e de plataforma baixa na Dataprev e os arquivos resultantes são transmitidos através da rede para o ambiente de produção da APE.

O processo de Transformação aplica um conjunto de regras e funções aos dados extraídos para qualificar as informações e agregar os dados. Na qualificação de informações, por exemplo, os campos de endereçamento são analisados se estão em conformidade com a padronização dos Correios e corrigidos. Na agregação de dados transformam-se os dados unitários em dados derivados que atendam a determinadas condições e dimensões, por exemplo, somar a quantidade e valores de benefícios pagos por Agência da Previdência Social (APS) por espécie de benefício. Esse processo é feito de forma manual utilizando ferramentas de manipulação de dados (Microsoft Access e Visual Basic) por uma equipe de 7 analistas e leva aproximadamente 45 dias para ser concluído.

Na carga do banco de dados do MAIPREV é utilizada o Serviço de Transformação de Dados (DTS) do Microsoft Access para exportar dados para a base de dados do SQL Server.

5.2. Descrevendo o Problema

A partir dos dados oriundos do Sistema de Pagamentos da Previdência Social denominada MACIÇA são geradas 53 agregações diferentes. Os dados da CONCESSÃO (Informação sobre os benefícios concedidos) e do PAB (Pagamento Alternativo de Benefícios) também são agregados. Os dados resultantes do processo de agregação são armazenados em banco de dados para acesso dos usuários da APE. A Figura 7 ilustra o

fluxograma do processo de ETL do Maiprev. A parte destacada em cor amarela será o escopo do estudo de caso.

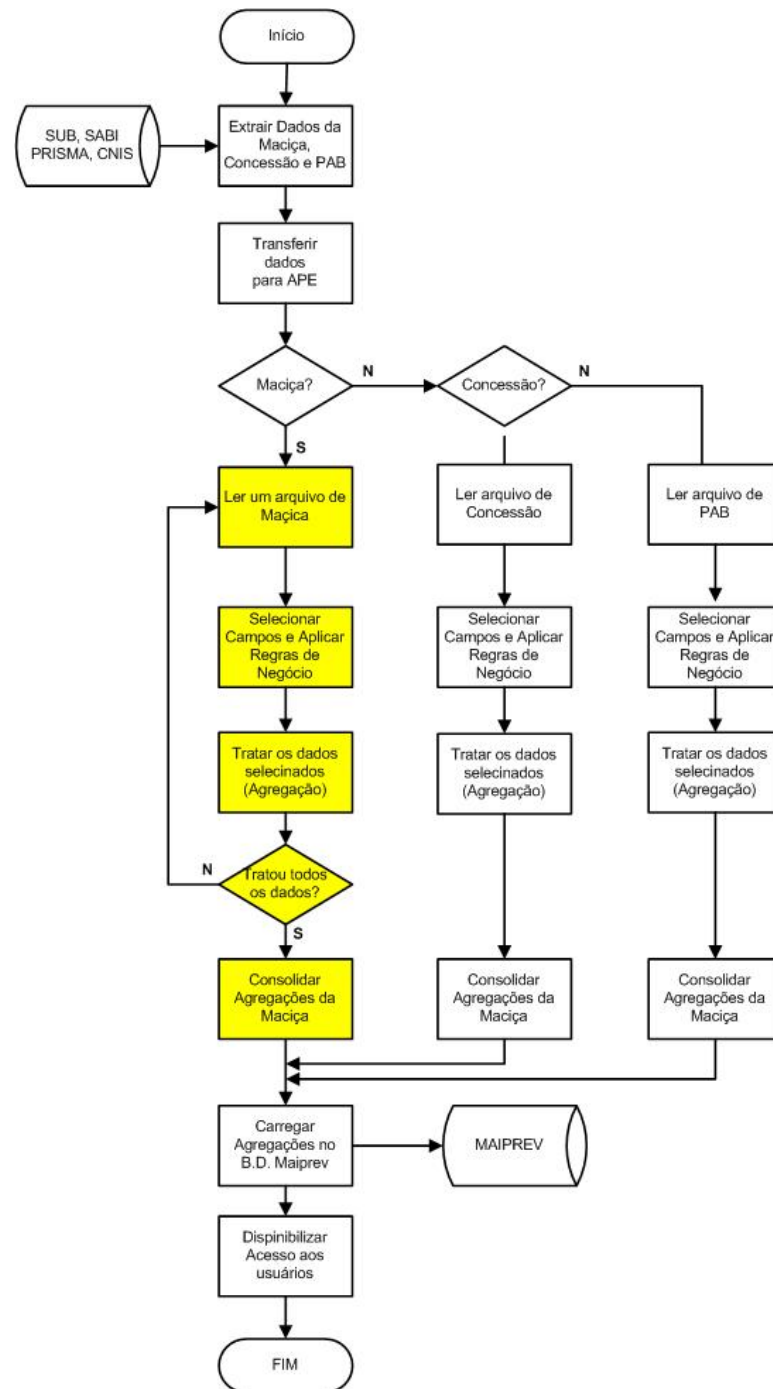


Figura 7 – Fluxograma do processo de ETL do MAIPREV

Os dados são extraídos do mainframe e disponibilizados em formato texto num repositório de FTP para que os analistas os capturem e executem as consolidações necessárias. O volume de dados disponibilizado para processamento totaliza 24 GBytes

divididos em 16 arquivos de 1,5 GBytes cada. A tabela original possui 143 campos totalizando 1.417 caracteres por linha.

O problema é que todo este trabalho consome aproximadamente 45 dias para ser processado por 7 analistas. Isto cria uma defasagem muito grande entre a geração dos dados no computador de grande porte e a disponibilização das informações consolidadas para o usuário da APE. Como os processos não são totalmente automatizados podem ocorrer problemas de inconsistência na consolidação dos dados por erro humano.

Para solucionar estes problemas é proposta uma modelagem das rotinas de agregação do sistema MAIPREV utilizando grades computacionais.

5.3. Proposta de Modelagem para o Sistema MAIPREV

A proposta é executar a fase de Transformação do processo de ETL do MAIPREV utilizando uma infra-estrutura de grades computacionais para executar a qualificação de informações e a agregação dos dados derivados. A proposta é apresentada na Figura 8.

Nesta proposta os processos de agregação das bases de dados da MACIÇA, PAB e CONCESSÃO são todos executados pelos nós da grade computacional, automatizando esta parte da aplicação.

Na máquina denominada servidor de consolidação, os dados originais passam por um tratamento inicial e são criadas as tarefas que serão submetidas aos nós da grade. O gerenciador da grade recebe as requisições de tarefas e as distribui junto com os dados para os nós ociosos da grade, ficando responsável pela alocação dos recursos necessários para a execução das tarefas. Os nós da grade são estações do INSS configuradas para serem utilizadas quando ociosas. As tarefas submetidas às grades devolvem os dados derivados que passam por um processo de consolidação antes de serem disponibilizadas para carga no banco de dados do sistema MAIPREV.

A implementação da solução proposta prevê a utilização de sistema operacional linux no servidor de consolidação e no gerenciador de grade. As estações de trabalho devem ser incorporadas como nós da grade computacional utilizando o sistema operacional instalado no ambiente computacional. O *middleware* de grade escolhido para suportar a aplicação foi o Ourgrid. Desta forma, garante-se o uso de soluções abertas, escaláveis e portáteis atendendo aos requisitos de e-Gov.

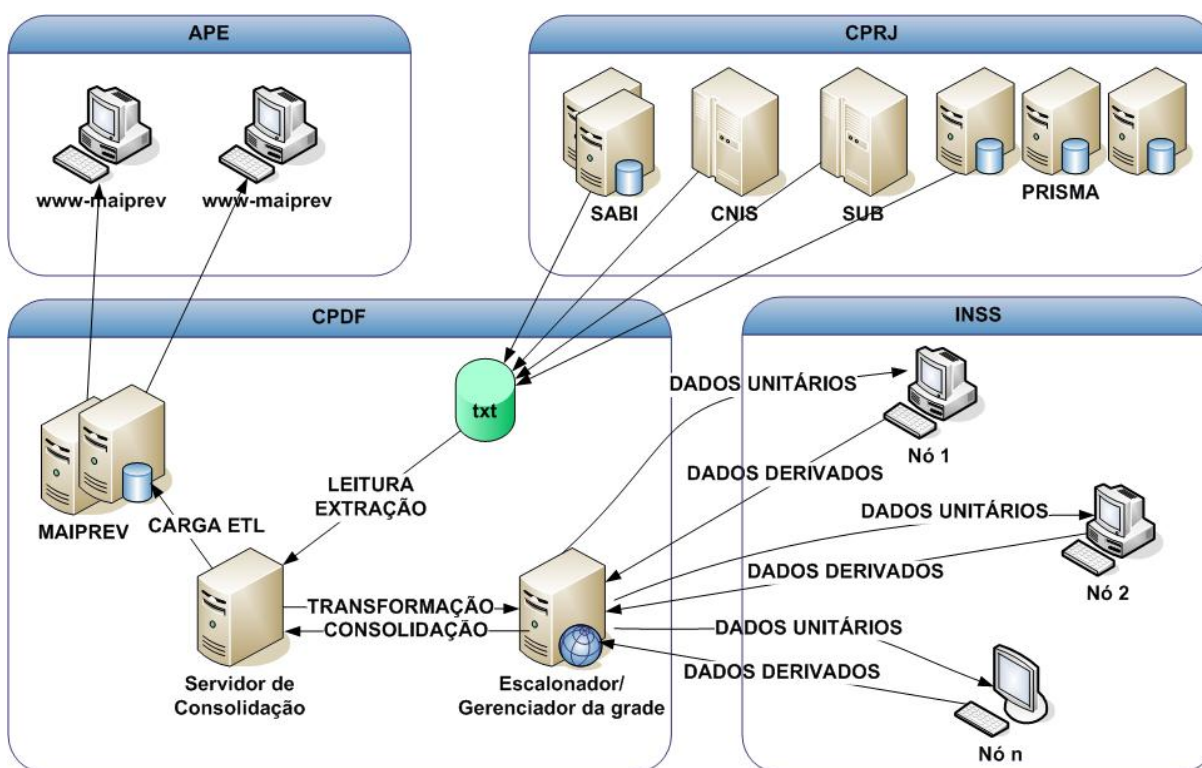


Figura 8 - Proposta para processo de ETL do MAIPREV executando em grades

5.4. Implementação de uma Rotina do Processo de ETL

Para descobrir se é possível obter benefícios com a utilização de grades computacionais para processar estas informações foi implementada uma Rotina do Processo de ETL. A partir de agora serão detalhados a escolha da rotina de agregação, a infra-estrutura de grade utilizada e os resultados obtidos na execução de uma rotina na grade (aplicação distribuída).

Dado o grande volume de dados a ser processado e ao grande conjunto de agregações da fase de Transformação do processo de ETL do MAIPREV optou-se por desenvolver uma das rotinas de cálculo da agregação de uma das visões do sistema MAIPREV.

Os campos das tabelas geradas pelas agregações da base de dados da maíça e dos dados nelas armazenadas foram analisados e observou-se que os campos derivados das agregações eram idênticos alterando-se apenas os critérios pelas quais os dados eram agrupados. Assim, uma das agregações foi selecionada como candidata para este estudo. Esta agregação recebe, no sistema MAIPREV, o nome no formato “MACICAMMAAAAOLCONCESSORAOLMANUTENCAO” onde MM representa o mês e

AAAA o ano do pagamento do benefício. Após escolhida a candidata para estudo, foi realizada uma análise nos dados de origem para otimizar esta rotina da aplicação.

5.4.1. Infra-estrutura da Grade Utilizada

Para executar a rotina desenvolvida na grade, foi utilizado o *middleware* OurGrid na versão 4. Nesta versão, os elementos que compõem a infra-estrutura da grade são: servidor XMPP, Peer, Worker e Broker.

O servidor de comunicação é baseado no protocolo XMPP (*Extensible Messaging and Presence Protocol*) e faz o processamento de toda comunicação entre os componentes do OurGrid 4. Os componentes do OurGrid são certificados para trabalhar com o servidor Openfire (<http://www.igniterealtime.org/projects/openfire/index.jsp>).

O OurGrid Peer é o componente que gerencia os Workers. O OurGrid Worker é um nó da grade responsável pela execução das tarefas remotas. O OurGrid Broker é usado para submeter as tarefas à grade.

O arquivo de configuração .sdf contém informações sobre os Workers que fazem parte da grade e é usado pelo Peer para determinar quais Workers estão associados à grade. O arquivo de configuração .gdf é utilizado pelo Broker para indicar a qual Peer ele deve se associar para submeter os trabalhos à grade. O arquivo de configuração .jdf é utilizado pelo Broker e contém a descrição dos trabalhos que serão submetidos à grade.

Para os testes realizados, uma máquina foi configurada com o Peer e o Broker e o servidor XMPP. Nesta máquina os dados da maciça foram disponibilizados e a aplicação executada. Foram utilizadas 20 máquinas configuradas como Worker.

Especificação do Servidor XMPP/Peer/Broker:

- Processador: Intel Dual Core 1,6 Ghz
- Memória: 2,5 GB
- HD: 120 GB (partição de 30 GB para linux)

Especificação do Worker:

- Processador: AMD Pentium 4 1,8 Ghz
- Memória: 512 MB
- HD: 160 GB (partição de 25 GB para linux)

5.4.2. A Primeira Solução

Em função do grande volume de dados disponibilizado pelo processamento da Maciça, foi necessário reduzir a quantidade de dados trocados na infra-estrutura da grade.

O primeiro passo foi a redução da quantidade de dados úteis a serem transmitidos para os nós da grade de computadores. Cada registro do arquivo original da Maciça é composto por 143 campos totalizando 1.417 caracteres. Analisando os dados foram identificados quais campos eram realmente necessários para o processamento reduzindo a quantidade de dados para 7 campos. Foram utilizados 53 caracteres obtendo uma redução de 96% no volume de dados, conforme apresentados nas tabelas 1 e 2.

Tabela 1 – Dados úteis para a agregação escolhida

Campo	Tamanho
OL-CONCESSAO	8
OL-MANUTENCAO	8
VALOR	14
DATA DER	8
DATA DIB	8
DATA DDB	8
ANO NASCIMENTO	4
Total	58

Tabela 2 – Taxa de compactação do arquivo texto

OL Concessão	Sem Compactação (bytes)	Compactado (bytes)	Taxa de Compactação
2001010	2.385.193	79.891	96,70%
2001020	1.096.869	36.543	96,70%
2001030	3.798.656	127.057	96,70%
2001040	2.283.890	76.546	96,60%
...
20024070	329.869	11.195	96,60%
21001010	2.429.325	80.574	96,70%
21001020	1.088.373	36.159	96,70%
Total	1.043.229.799	34.558.606	96,70%

Ainda assim, o volume de dados a ser processado era grande. Antes de submeter os dados à grade os dados foram compactados para diminuir o tempo de transferência. Como se tratava de arquivo texto obteve-se taxas médias de compressão de 96,7% dependendo do tamanho do arquivo conseqüentemente diminuindo sensivelmente o tempo para a transferência dos arquivos.

No segundo passo foi definido um critério para a separação dos dados para que cada tarefa pudesse ser executada de forma independente.

A rede de atendimento do INSS que presta serviço de atendimento à população brasileira conta com mais de 1.300 unidades distribuídas pelo país, cada qual identificado com um número denominado OL. Foi utilizado o campo OL Concessão do arquivo da Maciça, que identifica a Agência responsável por conceder o benefício do segurado da Previdência Social, para separar os dados a serem executadas pelas tarefas.

Este agrupamento por OL Concessão nos permitiu obter dados que pudessem ser processados por tarefas independentes em nós de uma grade de computadores, caracterizando-se assim a utilização de aplicações do tipo *Bag-of-Tasks*.

5.4.2.1. Programa Principal

O objetivo é obter os dados separados por OL Concessão e submeter cada arquivo gerado para a grade de computadores como uma tarefa.

O programa principal faz chamada ao programa separaol que separa os dados de origem usando como chave o campo OL Concessão obtendo como resultado arquivos separados por OL e em seguida cada um destes arquivos são compactados. Para cada arquivo gerado é associada uma tarefa que será executada no nó remoto. Os dados e as tarefas são submetidos para execução na grade de computadores, neste caso usando OurGrid 4. A submissão na grade é monitorada e ao final do processamento os resultados obtidos em cada uma das tarefas são consolidados pelo programa consolida. Este programa gera um arquivo texto com os dados já agregados a serem carregados na base de dados do sistema MAIPREV.

Seu algoritmo está descrito abaixo:

- Para cada arquivo da Maciça, Separar os Dados por OL
 - Executar programa separaol
- Para cada OL encontrada nos arquivos da Maciça
 - Compactar o arquivo da OL gerada pelo separaol
 - Gerar uma tarefa associada ao arquivo de OL correspondente
- Submeter as tarefas geradas para execução na Grade
 - broker addjob maiprev.jdf
- Monitorar o status das tarefas
- Após o término de todas as tarefas consolidar os dados
 - Executar programa consolida

5.4.2.2. Programa SEPARAOL

Este programa faz leitura do arquivo da Maciça, extraindo linha a linha somente os campos necessários. Os campos extraídos são mostrados na tabela 1. Estes dados são gravados em vários arquivos texto nomeados com a identificação da OL Concessão, contendo somente os registros pertencentes àquela OL. Uma Lista contendo a relação de OLs Concessão é gerada e armazenada em memória.

Ao final do programa a Lista de OLs Concessão é gravada num arquivo para uso posterior pelo processo de consolidação dos resultados obtidos pelo processamento nos nós da grade.

O algoritmo do programa SEPARAOL está descrito abaixo:

```

Inicializar Lista de OL Concessão
Ler o arquivo da Maciça a ser processado
Enquanto não for final de arquivo
    Separar cada linha e recuperar os campos (OL Concessão, OL Manutenção, Valor,
    Data-DER, Data-DIB, Data-DDB, Ano nascimento)
    Armazenar o dado em seu respectivo arquivo identificado pela OL Concessão
    Se OL Concessão não for localizada na Lista então
        Incluir OL Concessão na Lista
Fim-Enquanto
Armazenar a lista de OL Concessão em arquivo para uso posterior na consolidação.

```

5.4.2.3. Escrevendo um *JOB*

Na descrição de um trabalho, a cláusula *job* identifica os atributos comuns para o trabalho como um todo e a cláusula *task* define os atributos e comandos específicos para cada tarefa que compõe a aplicação paralela. Ambas são compostas por sub-cláusulas conforme descritas abaixo.

Na cláusula *job*, a sub-cláusula *label* associa um nome ao trabalho, no exemplo abaixo o trabalho foi identificado com o nome "MAIPREV". A sub-cláusula *remote* define os comandos que serão executados por todas as tarefas, ou seja, todos os nós irão processar um mesmo conjunto de comandos alterando-se somente os dados que serão processados. A variável \$PLAYPEN identifica o local onde a tarefa será executada no nó remoto (denominado worker). A variável \$STORAGE define o caminho para o diretório de armazenamento de arquivos que podem ser compartilhados evitando que os arquivos sejam retransmitidos para worker.

Na cláusula *task*, as tarefas são descritas através de três sub-declarações: *init*, *remote* e *final*. A sub-cláusula *init* identifica os dados e os programas a serem transferidos

para o nó da grade. Estes arquivos são transferidos utilizando-se os comandos *put* e *store*. O comando *store* verifica se o arquivo existe no worker e o transfere somente se não existir no nó remoto no diretório \$STORAGE. A sub-cláusula *remote* descreve os comandos a serem executados no nó remoto. A cláusula *final* retorna os resultados de volta para o usuário solicitante ao final do processamento.

A variável \$JOB é um número que identifica o trabalho que está sendo executada na grade e a variável \$TASK é um número que identifica a tarefa associada a um determinado trabalho. O número de tarefas a serem executadas pelos nós remotos corresponde ao número de cláusulas *task* encontradas na definição do trabalho.

```

job :
  label : Maiprev
task:
  init : put 02001010.zip 02001010.zip
        store local local
  remote: unzip 02001010.zip ; cp $STORAGE/local $PLAYPEN ; chmod +x
         local ; ./local 02001010
  final : get 02001010.res 02001010-$JOB-$TASK.res
task:
  init : put 02001020.zip 02001020.zip
        store local local
  remote: unzip 02001020.zip ; cp $STORAGE/local $PLAYPEN ; chmod +x
         local ; ./local 02001020
  final : get 02001020.res 02001020-$JOB-$TASK.res
task:
  init : put 02001030.zip 02001030.zip
        store local local
  remote: unzip 02001030.zip ; cp $STORAGE/local $PLAYPEN ; chmod +x
         local ; ./local 02001030
  final : get 02001030.res 02001030-$JOB-$TASK.res
task:
  init : put 02001040.zip 02001040.zip
        store local local
  remote: unzip 02001040.zip ; cp $STORAGE/local $PLAYPEN ; chmod +x
         local ; ./local 02001040
  final : get 02001040.res 02001040-$JOB-$TASK.res

```

Após o processamento de todas as tarefas, os arquivos resultantes com extensão .res são consolidadas para encontrar o resultado final da aplicação.

5.4.2.4. Programa LOCAL

Este programa é executado em cada um dos nós da grade de computadores. Ele lê o arquivo contendo os dados de uma OL Concessão e para cada OL Manutenção contido

no arquivo executa os cálculos necessários para obter o resultado parcial da agregação. Os resultados são armazenados numa tabela cuja chave é a OL Manutenção.

Depois esta tabela é lida e os dados são consolidados e gravados em um arquivo texto (com extensão .res) que será retornado para o usuário. Cada linha contém informações da OL Concessão, OL Manutenção e dados derivados. Estes dados parciais serão consolidados no final do processo.

O algoritmo do programa LOCAL está descrito abaixo:

```

Ler o arquivo a ser processado
Enquanto não for final de arquivo faça
  Separa cada linha por campo (OL Concessão, OL Manutenção, Valor, Data-DER,
  Data-DIB, Data-DDB, Ano Nascimento)
  Para cada OL Manutenção, executar os cálculos e armazenar numa tabela
    Qtde := Qtde + 1;
    RendaTotal := RendaTotal + valor;
    Calcular o Tempo de Pagamento Alternativo de Benefício (Data DDB -
    Data DIB)
    Calcular o Tempo de Concessão (Data DDB - Data DER)
    Calcular o Tempo de Manutenção ((Data Atual - Data DIB) DIV 365)
    Calcular a Idade Média do Segurado no início do recebimento do Benefício
    (Ano DIB - Ano Nascimento)
    Calcular a Idade Média do Segurado na Data Atual (Ano Atual - Ano
    Nascimento)
  Fim-enquanto
Abrir o arquivo Destino e posicionar no final
Ler a tabela de OLs Manutenção
Para cada linha na tabela faça
  Consolidar os indicadores INDPAB, RAPIDEZ, DURACAO, MIDADEDIB e
  MIDADEATUAL
  Gerar linha com OL Concessão, OL Manutenção, Qtde, Total, INDPAB,
  RAPIDEZ, DURACAO, MIDADEDIB e MIDADEATUAL
  Gravar a linha no Arquivo Destino
Fim-Para
Fechar Arquivo Destino

```

5.4.2.5. Programa CONSOLIDA

O programa consolida faz a leitura de todos os resultados obtidos pelo processamento nos nós da grade pelo programa local e faz a totalização por OL Concessão, Gerência Executiva, Unidade da Federação e Brasil. Estes dados são guardados em uma tabela temporária para uso na consolidação final na qual os indicadores finais são calculados. Após o cálculo dos indicadores, o resultado final é armazenado em um arquivo para posterior carga no banco de dados do sistema MAIPREV.

Seu algoritmo está descrito abaixo:

```

Ler a lista de OL Concessão
Para cada elemento na Lista faça
    Ler o arquivo de resultante do processamento na grade para a OL Concessão
    Para cada linha do arquivo faça
        Totalizar por OL Concessão
        Totalizar por OL Concessão + OL Manutenção
        Totalizar por Gerência Executiva
        Totalizar por Gerência Executiva + OL Manutenção
        Totalizar por UF
        Totalizar por UF + OL Manutenção
        Totalizar por Brasil
        Totalizar por Brasil + OL Manutenção
        Armazenar os dados numa tabela de forma incremental
    Fim-Para
Fim-Para
Abrir arquivo para armazenar os resultados
Ler a tabela com dados processados
Para cada linha da tabela faça
    Calcular os indicadores
    Armazenar os indicadores obtidos no arquivo linha a linha
Fim-Para
Fechar o arquivo de resultados
  
```

5.4.2.6. Validando a Primeira Solução

Durante os testes foi identificado que a solução adotada não obteve a performance esperada. Nos testes executados observou-se que a solução distribuída ficou um pouco mais rápida que a solução executada de forma sequencial, com fator de aceleração³ [DER08] (*Speed UP*) de 1,3. O fator de aceleração indica quantas vezes a aplicação distribuída é mais rápida que a versão sequencial para executar uma tarefa. Os dados de execução da primeira solução distribuída e da aplicação sequencial são apresentados na tabela 3. A aplicação distribuída foi executada na mesma máquina da aplicação sequencial, ou seja, no servidor XMPP/Peer/Broker.

Tabela 3 - Tempo médio de execução na primeira solução

Tempo de execução (hh:mm:ss)	Média
Aplicação Distribuída	00:45:21
Aplicação Sequencial	00:59:07
Fator de Aceleração	1,3

Na tabela 4 é apresentada a distribuição de tempo no processamento da aplicação sequencial. Foram observados também problemas de travamento do Peer durante a

³ Fator de Aceleração = Tempo Aplicação Seqüencial / Tempo Aplicação Distribuída

execução das tarefas na grade devido ao grande volume de tarefas geradas pela aplicação, no caso foram geradas 1.030 tarefas.

Tabela 4 - Distribuição de tempo no processamento da aplicação sequencial

Tempo de execução (hh:mm:ss)	Média	%
Separação por OL	00:31:23	53,10%
Agregação por OL	00:27:42	46,90%
Consolidação	00:00:02	0,10%
Total	00:59:07	100,00%

Analisando o tempo de execução da aplicação distribuída observou-se que a maior parte do tempo de processamento da aplicação foi gasto na separação dos dados e na execução dos trabalhos na grade. A tabela 5 mostra detalhadamente o tempo gasto na execução de cada um dos passos executados pela aplicação.

Tabela 5 - Tempo de processamento da aplicação distribuída na primeira solução

Tempo de execução (hh:mm:ss)	Média	%
Separar dados por OL Concessão	00:18:23	40,50%
Compactar arquivos e gerar trabalho	00:00:57	2,10%
Submeter o trabalho à grade	00:00:11	0,40%
Monitorar o Fim dos Trabalhos	00:25:49	56,90%
Consolidar os Resultados	00:00:01	0,00%
Total	00:45:21	100,00%

Na tabela 6, observa-se que, em média, somente 13% do tempo de execução de uma tarefa na grade foi utilizado com o processamento da aplicação (*Remote*) e 87% foi utilizada para a transferência de dados de entrada e saída. Observou-se, também, uma grande variação nos tempos de execução dos nós da grade, ou seja, os tempos de execução de uma tarefa na grade variaram de um mínimo de 5.246 ms a um valor máximo de 21.768 ms.

Tabela 6 - Tempo médio de utilização da grade por fase na primeira solução

Fases de uma tarefa	Média	%
<i>Init</i>	6.757	47%
<i>Remote</i>	1.928	13%
<i>Final</i>	5.680	40%
Total	14.365	100%

Nesta solução foi identificado que os nós da grade passavam a maior parte do tempo transferindo dados ao invés de processá-los, sem obter ganhos significativos de performance para a aplicação. A Figura 9 mostra que os recursos da grade ficavam a maior parte do tempo em uso pela aplicação.

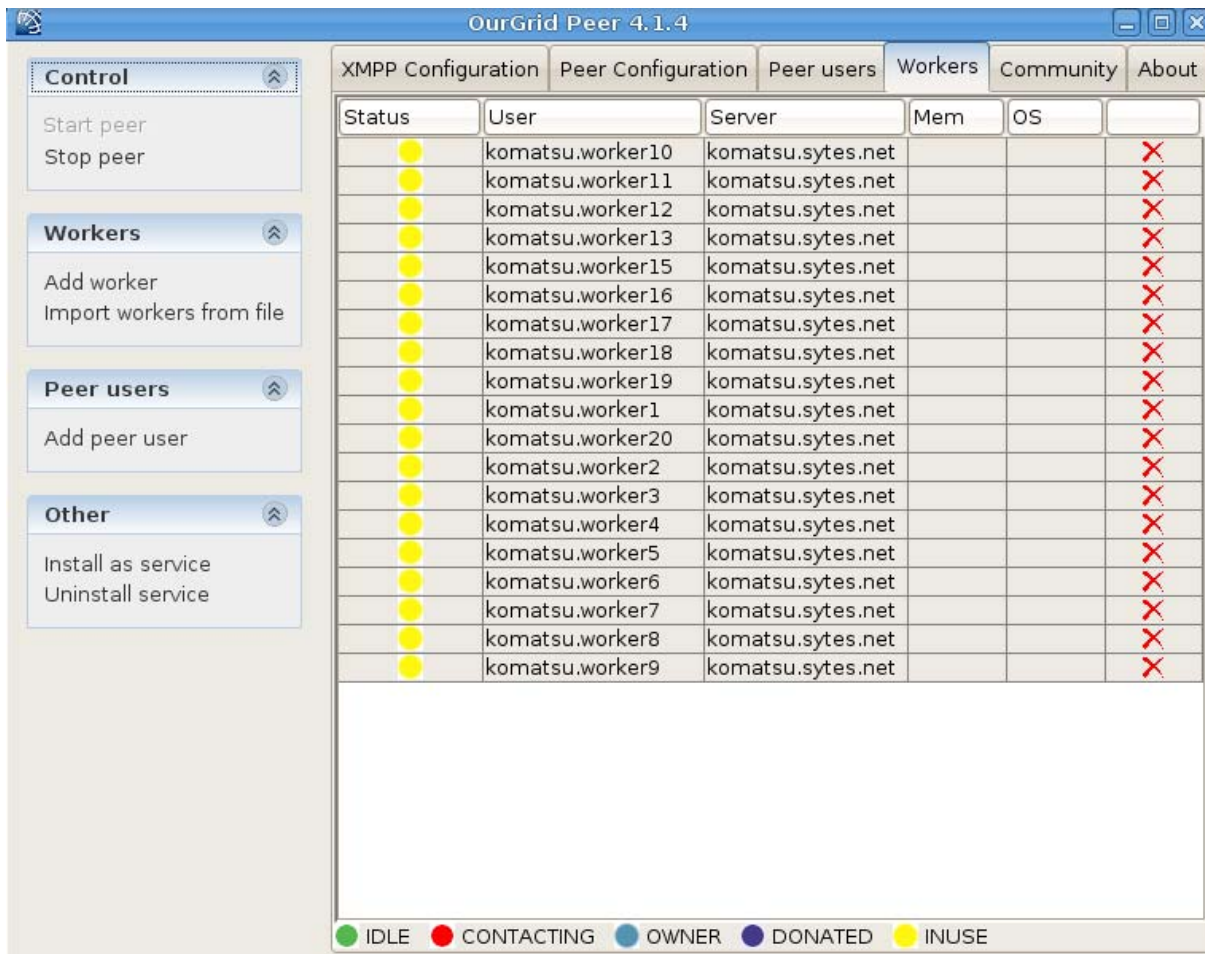


Figura 9 - Uso dos recursos da grade na primeira solução

5.4.3. A Segunda Solução

Foi identificado na primeira solução que os trabalhos demoravam serem submetidas para a grade, pois eram geradas somente após a análise de todos os arquivos de dados. Isso deixava as máquinas que compõem a grade muito tempo ociosas antes de começar o processamento e depois de iniciado o processamento na grade os trabalhos demoravam a serem executados na grade devido ao grande número de tarefas geradas.

Na segunda solução, foi alterada a forma como a aplicação tratava a agregação dos dados. Na primeira solução toda a base de dados foi lida para a geração dos arquivos separados por OL Concessão e depois cada arquivo gerado foi submetida à grade. Nesta nova abordagem, para cada arquivo da maciça lida (total de 16 arquivos) os dados eram separados por OL Concessão e imediatamente submetidos à execução na grade.

O algoritmo desta solução esta descrita abaixo:

- Para cada arquivo da Maciça faça
 - Separar por OL Concessão os dados em arquivos diferentes
 - Compactar o arquivo texto
 - Gerar um trabalho
 - Submeter o trabalho para a grade
- Monitorar a execução dos trabalhos na grade
 - ao terminar um trabalho, descompactar os resultados obtidos
- Executar a consolidação dos resultados obtidos

Além disso, com objetivo de aumentar o tempo de processamento no nó da grade, modificou-se a geração das tarefas e aumentou-se o número de arquivos que eram processados por cada uma das tarefas, ou seja, ao invés de uma tarefa conter uma OL concessão passaram a conter mais OLs a serem processadas. O resultado desta modificação pode ser vista na descrição do trabalho abaixo na qual 5 OLs Concessão foram agrupadas numa mesma tarefa (*Task*).

```

job :
  label : Maiprev
task:
  init : put 02001010.zip 02001010.zip
        put 02001020.zip 02001020.zip
        put 02001030.zip 02001030.zip
        put 02001040.zip 02001040.zip
        put 02001050.zip 02001050.zip
        store local local
  remote: unzip 02001010.zip; unzip 02001020.zip; unzip 02001030.zip; unzip
          02001040.zip; unzip 02001050.zip; cp $STORAGE/local $PLAYPE N; chmod +x
          local; ./local 02001010; ./local 02001020; ./local 02001030; ./local 02001040;
          ./local 02001050
  final: get 02001010.res 02001010.res
        get 02001020.res 02001020.res
        get 02001030.res 02001030.res
        get 02001040.res 02001040.res
        get 02001050.res 02001050.res
task:
  init : put 02001060.zip 02001060.zip
        put 02001070.zip 02001070.zip
        put 02001080.zip 02001080.zip
        put 02001090.zip 02001090.zip
        put 02001100.zip 02001100.zip
        store local local
  remote: unzip 02001060.zip; unzip 02001070.zip; unzip 02001080.zip; unzip
          02001090.zip; unzip 02001100.zip; cp $STORAGE/local $PLAYPEN; chmod +x
          local; ./local 02001060; ./local 02001070; ./local 02001080; ./local 02001090;
          ./local 02001100
  final: get 02001060.res 02001060.res
        get 02001070.res 02001070.res
        get 02001080.res 02001080.res

```

get 02001090.res 02001090.res
get 02001100.res 02001100.res

Os arquivos gerados pelo processo de separação de OLs foram agrupados em grupos de 5, 7 e 10 escolhidos aleatoriamente para verificar se haveria algum ganho. Os tempos médios obtidos na execução desta solução distribuída são apresentados na tabela 7. Agrupamentos maiores poderiam ser testados, mas em função de não obter ganhos significativos com esta técnica, os testes foram encerrados.

Tabela 7 - Tempo médio de execução na segunda solução

Tempo de execução (hh:mm:ss)	Média
Aplicação Distribuída - tarefa com 1 OLs	00:34:12
Aplicação Distribuída - tarefa com 5 OLs	00:32:09
Aplicação Distribuída - tarefa com 7 OLs	00:33:02
Aplicação Distribuída - tarefa com 10 OLs	00:31:49
Aplicação Distribuída - Média	00:32:48
Aplicação Sequencial	00:59:07
Fator de Aceleração	2,25

A tabela 8 apresenta o tempo médio obtido no processamento da aplicação na segunda solução a cada passo. Quase 97% do tempo de processamento estão concentrados na separação, compactação dos dados e submissão de cada trabalho à grade de computadores. O tempo de monitoração da grade é pequeno, pois muitas das tarefas executadas na grade foram finalizadas antes da submissão de novas tarefas à grade, restando no *final* o processamento de poucas tarefas, justificando o tempo residual na monitoração.

Tabela 8 - Tempo médio de execução na segunda solução passo a passo

Tempo de execução (hh:mm:ss)	1 OL	5 OLs	7 OLs	10 OLs	Média	%
Separar/Compactar/ Submeter Trabalho	00:33:08	00:31:45	00:30:55	00:31:19	00:31:47	96,90%
Monitorar Trabalho	00:01:03	00:00:23	00:02:06	00:00:29	00:01:00	3,10%
Consolidar	00:00:01	00:00:01	00:00:01	00:00:01	00:00:01	0,10%
Total	00:34:12	00:32:09	00:33:02	00:31:49	00:32:48	100,00%

Os tempos médios (ms) de utilização da grade em cada uma das fases estão apresentados na tabela 9. Observou-se nesta solução uma grande variação nos tempos de execução em cada um dos nós, pois apesar de cada tarefa ter um número fixo de OLs, cada OL possui uma quantidade diferente de registros.

Tabela 9 - Tempo médio de utilização da grade por fase na Segunda Solução

Fases	1 OL	%	5 OLs	%	7 OLs	%	10 OLs	%
<i>Init</i>	6.666	59%	24.473	81%	25.599	80%	24.378	70%
<i>Remote</i>	424	4%	1.605	5%	2.563	8%	3.850	11%
<i>Final</i>	4.229	37%	4.320	14%	3.867	12%	6.639	19%
Total	11.319	100%	30.398	100%	32.030	100%	34.867	100%

5.4.4. A Terceira Solução

A segunda solução se apresentou mais eficiente que a primeira e também em relação à aplicação sequencial, mas a maior parte do tempo era utilizada no processamento inicial de separação, compactação dos dados e submissão dos trabalhos à grade.

Na terceira solução foi modificado o local onde o processo de separação de OL era executado. Nesta abordagem o processo de separação por OL Concessão foi submetido à grade, ou seja, a aplicação separa as colunas necessárias, compacta-os e envia para a grade executar o processo de separação de OL e agregação dos dados. Assim, para cada arquivo da maciça foi gerado um trabalho contendo uma tarefa.

O algoritmo desta solução está descrita abaixo:

- Para cada arquivo da Maciça faça
 - Separar somente as colunas necessárias em um arquivo texto
 - Compactar o arquivo texto
 - Gerar um trabalho
 - Submeter o trabalho para a grade
- Monitorar a execução dos trabalhos na grade
 - ao terminar um trabalho, descompactar os resultados obtidos
- Executar a consolidação dos resultados obtidos

Os tempos médios obtidos na execução da aplicação utilizando esta abordagem são apresentados na tabela 10.

Tabela 10 – Tempo médio de execução na terceira solução

Tempo de execução (hh:mm:ss)	Média
Usando Grade	00:24:07
Aplicação Sequencial	00:59:07
Fator de Aceleração	2,45

Os tempos médios (ms) de utilização da grade em cada uma das fases estão apresentados na tabela 11.

Tabela 11 - Tempo médio de utilização da grade por fase na terceira solução

Fases de uma tarefa	Média	%
<i>Init</i>	56.583	49,00%
<i>Remote</i>	55.608	48,20%
<i>Final</i>	3.295	2,90%
Total	115.486	100,00%

A tabela 12 apresenta os tempos médios obtidos em cada um dos passos executados pela aplicação na terceira solução. Nesta solução observa-se, também, a concentração do processamento na fase de separação, compactação de dados e submissão do trabalho à grade.

Tabela 12 - Detalhamento do tempo de execução da aplicação na terceira solução

Tempo de execução (hh:mm:ss)	Média	%
Separar Colunas/Compactar/Submeter Trabalho	00:22:24	92,90%
Monitorar Trabalho	00:01:42	7,00%
Consolidar	00:00:01	0,10%
Total	00:24:07	100,00%

The screenshot shows the 'OurGrid Peer 4.1.4' application window. The 'Workers' tab is selected, displaying a table of worker information. The table has columns for Status, User, Server, M..., OS, and a red 'X' icon. The workers listed are komatsu.worker13 through komatsu.worker20, all connected to komatsu.sytes.net. The status of the first two workers (13 and 9) is 'INUSE' (yellow), while the others are 'IDLE' (green). The legend at the bottom indicates: IDLE (green), CONTACTING (red), OWNER (blue), DONATED (purple), and INUSE (yellow).

Status	User	Server	M...	OS	
INUSE	komatsu.worker13	komatsu.sytes.net			X
INUSE	komatsu.worker9	komatsu.sytes.net			X
IDLE	komatsu.worker10	komatsu.sytes.net			X
IDLE	komatsu.worker11	komatsu.sytes.net			X
IDLE	komatsu.worker12	komatsu.sytes.net			X
IDLE	komatsu.worker14	komatsu.sytes.net			X
IDLE	komatsu.worker15	komatsu.sytes.net			X
IDLE	komatsu.worker16	komatsu.sytes.net			X
IDLE	komatsu.worker17	komatsu.sytes.net			X
IDLE	komatsu.worker18	komatsu.sytes.net			X
IDLE	komatsu.worker19	komatsu.sytes.net			X
IDLE	komatsu.worker1	komatsu.sytes.net			X
IDLE	komatsu.worker20	komatsu.sytes.net			X
IDLE	komatsu.worker2	komatsu.sytes.net			X
IDLE	komatsu.worker3	komatsu.sytes.net			X
IDLE	komatsu.worker4	komatsu.sytes.net			X
IDLE	komatsu.worker5	komatsu.sytes.net			X
IDLE	komatsu.worker6	komatsu.sytes.net			X
IDLE	komatsu.worker7	komatsu.sytes.net			X
IDLE	komatsu.worker8	komatsu.sytes.net			X

Figura 10 - Utilização da grade na Terceira Solução

Nesta solução o uso da grade foi no máximo de 10% conforme apresentado na Figura 10. Somente dois nós da grade foram usadas simultaneamente, por que as execuções dos trabalhos terminavam antes de novas tarefas serem submetidas à grade.

5.4.5. A Quarta Solução

Na solução anterior foi observado um tempo médio alto na transferência dos dados. A quarta solução faz uma otimização da solução anterior, com objetivo de diminuir o tempo de processamento das tarefas na grade. Para isto, antes de transferir os dados para a grade, os arquivos são quebrados em 5 partes cada um contendo 230.000 registros procurando obter arquivos menores e geração de mais tarefas.

Assim, a aplicação faz um corte nos dados a cada 230.000 registros, compacta-os e envia para a grade executar o processo de separação de OL e agregação dos dados. Para cada arquivo da maciça foi gerado um trabalho contendo 5 tarefas.

O algoritmo desta solução está descrita abaixo:

- Para cada arquivo da Maciça faça
 - Separar somente as colunas necessárias em um arquivo texto
 - Quebrar o arquivo gerado a cada 230.000 registros
 - Compactar os arquivos
 - Gerar um trabalho para cada arquivo
 - Submeter os trabalhos para a grade
- Monitorar a execução dos trabalhos na grade
 - ao terminar um trabalho, descompactar os resultados obtidos
- Executar a consolidação dos resultados obtidos

A tabela 13 apresenta os tempos médios de execução desta solução.

Tabela 13 - Tempo médio de execução na quarta solução

Tempo de execução (hh:mm:ss)	Média
Aplicação Distribuída	00:23:03
Aplicação Sequencial	00:59:07
Fator de Aceleração	2,56

Na prática, o processamento dos trabalhos na grade foi mais rápido, mas foi detectado um problema no mecanismo de monitoração de final dos trabalhos. Este mecanismo verificava um a um se os trabalhos haviam sido finalizados monitorando o estado das tarefas cuja informação era fornecida pelo Broker e se tornou muito lento à medida que o número de tarefas aumentava. Dessa forma, o mecanismo de monitoração foi alterado obtendo o estado das tarefas através de um arquivo de sinalização de fim de

tarefa enviado pelo Worker após o processamento e retorno dos dados para o usuário. Os tempos médios (ms) obtidos com a utilização da grade com a quarta solução são apresentados na tabela 14.

Tabela 14 - Tempo médio de utilização da grade por fase na quarta solução

Fase de uma tarefa	Média	%
<i>Init</i>	14.611	49,40%
<i>Remote</i>	11.130	37,60%
<i>Final</i>	3.835	13,00%
Total	29.576	100,00%

Conforme se apresenta na Figura 11, a utilização da grade foi de 25%, em média, na quarta solução.

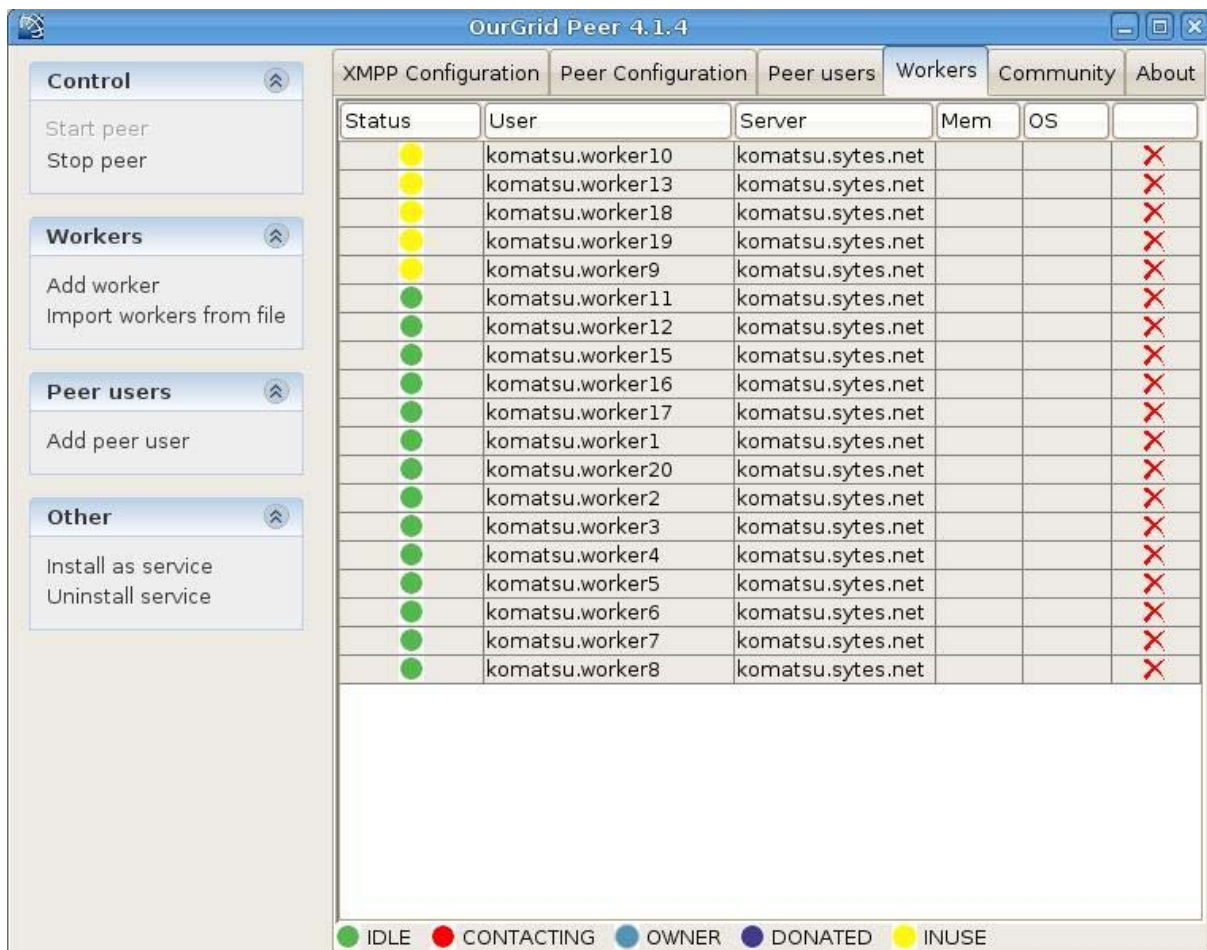


Figura 11 - Utilização da grade na quarta solução

5.4.6. Analisando os Resultados Obtidos

No processamento da porção da aplicação utilizada nos testes foram elaboradas quatro soluções. Nos testes realizados foram utilizados 16 arquivos de maciça totalizando 17.681.821 registros ocupando 24.491.988 bytes de espaço em disco.

Conforme resumido na tabela 15 a terceira e quarta soluções apresentaram os melhores indicadores de tempo se comparados à execução da aplicação sequencial. Apesar disso, o uso dos recursos da grade foi mínimo. Esta limitação no uso da grade se deve ao fato das tarefas em execução terminarem antes da grade receber novos trabalhos.

Tabela 15 - Comparativo dos tempos de execução das soluções

Solução	Tempo Execução	Fator de Aceleração	Trabalhos	Tarefas	Utilização Grade
Aplicação Sequencial	00:59:07	-	-	-	0%
Primeira Solução	00:45:21	1,3	1	1030	100%
Segunda Solução	00:32:48	1,8	16	152	100%
Terceira Solução	00:24:07	2,45	16	16	10%
Quarta Solução	00:23:03	2,56	16	80	25%

Além da falta de performance de algumas soluções, durante a execução dos testes, nos deparamos com vários problemas, os quais se destacaram:

- Quantidade de memória RAM na máquina com Peer e Broker. A máquina utilizada no processamento de uma base de dados maior (com 24 milhões de registros e 33 Gbytes de dados) acusou falta de memória na instância do Broker e do Peer em execução reportada pelo Java.
- Travamento do Peer na execução da primeira solução devido ao grande número de tarefas submetidos à grade. Neste caso, o Peer foi reinicializado e o processamento no Broker foi retomado automaticamente com o processamento de todas as tarefas até o final.
- Na submissão dos trabalhos na comunidade foram detectados problemas de *time-out* no envio dos dados à grade e na recepção da resposta pelo usuário. No teste em particular, foi utilizada uma grade instalada na Universidade do Estado de Mato Grosso (Unemat) em Colíder. Esta grade era utilizada para executar testes de desempenho de algoritmos de escalonamento de tarefas em grades. O link de comunicação de dados utilizado foi com tecnologia 3G com velocidade de 250 Kbps nesta grade e IP dedicado de 512 Kbps na grade da Unemat. Os testes foram refeitos utilizando um link 3G com velocidade de 1,0 Mbps e os resultados

obtidos não foram eficientes. Por causa da instabilidade da conexão à internet neste Peer não foi possível conseguir resultados satisfatórios. Nenhum dos trabalhos submetidos à grade foi concluído, somente algumas tarefas do trabalho submetido foram finalizadas com tempos de execução altos das tarefas quando comparados com a solução executada localmente. Ao tentar usar os recursos disponíveis na grade da comunidade OurGrid, também, foram identificados os mesmos problemas. Na quarta solução executada na grade da comunidade OurGrid, houve um acréscimo de tempo em média de 509%. A maioria das tarefas foi abortada por *time-out* na conexão.

Ainda assim, o que chamou a atenção foi o tempo que a aplicação utilizava para fazer o processamento inicial, como pode ser observado na tabela 16. Nos testes executados a maior parte do tempo (mais de 90%) foi utilizado com a preparação dos dados originais para processamento na grade, ou seja, separando as colunas (campos) necessárias, compactando os dados, gerando os trabalhos e submetendo-os para a grade de computadores.

Tabela 16 - Relação entre tempo total de execução e tempo para separação, compactação de dados e submissão de trabalhos

Solução	Tempo de Execução	Separação/ Compactação	%
Aplicação Sequencial	00:59:07	00:31:23	53,10%
Primeira Solução	00:45:21	00:19:31	43,00%
Segunda Solução	00:32:48	00:31:47	96,90%
Terceira Solução	00:24:07	00:22:24	92,90%
Quarta Solução	00:23:03	00:22:25	97,30%

Depois de submetidas à grade a maioria das tarefas terminava antes de novas submissões à grade. Neste caso, as propostas que podem melhorar o desempenho da aplicação devem estar concentradas nesta parte da aplicação. A melhor solução encontrada até então foi a quarta solução que faz uso do mecanismo de *pipeline* [DER06], ou seja, à medida que as tarefas vão sendo geradas elas vão sendo submetidas à grade.

Dentre os fatores que influenciaram a obtenção destes tempos, destacam-se a velocidade de gravação do disco, pois os arquivos gerados são gravados em disco para que possam ser transferidos até a máquina remota pela infra-estrutura da grade. O ideal seria transferir os dados pré-processados diretamente da memória para o nó da grade, sem necessitar de um armazenamento secundário, mas é um recurso que não está disponível na grade utilizada.

A tabela 17 mostra uma comparação de execução da quarta solução em máquinas diferentes. A configuração das máquinas é:

- Configuração da máquina (1):

- INTEL DUAL CORE 1,6 Ghz
- 2,5 GB RAM
- HD 120 GB (PARTIÇÃO LINUX DE 30 GB) - SATA 150 MB/s - 5400 rpm

- Configuração da máquina (2):

- AMD ATHLON 64 X2 DUAL CORE 1,9 Ghz
- 2,0 GB RAM
- HD 160 GB (PARTIÇÃO LINUX DE 50 GB) - SATA 3,0 Gbps - 7200 rpm

Tabela 17 - Comparativo de execução da aplicação em máquinas diferentes

Soluções	Tempo de Execução	Fator de Aceleração
Aplicação Sequencial	00:59:07	-
Quarta Solução Máquina (1)	00:22:25	2,56
Quarta Solução Máquina (2)	00:11:43	5,05

Pode-se observar na tabela 17 que houve um significativo ganho na quarta solução executada na máquina (2). Quando comparamos a execução da aplicação sequencial na máquina (1) com a quarta solução na máquina (2) obteve-se um fator de aceleração de 5,05. Observa-se que a máquina (2) possui um HD com velocidade superior que a máquina (1). Desse modo recomenda-se a utilização de mecanismos de armazenamento secundário de alta velocidade para a máquina de consolidação.

De Rose et al. [DER06] mostrou que é possível obter ganhos de desempenho de até 80 vezes com a aplicação GerpavGrid utilizando uma infra-estrutura de grades. Em função de não obter-se resultados satisfatórios, quando comparados com a aplicação GerpavGrid, propõe-se a seguir algumas alternativas para melhorar o processamento desta aplicação distribuída.

Com relação ao processamento, uma proposta é que todo o processo de preparação dos dados seja executado nos nós da grade e em seguida a agregação. O arquivo de maciça original será quebrado a cada N registros e utilizando técnicas de processamento paralelo serão disparados threads que executem a compactação, geração de trabalhos e submissão dos trabalhos à grade.

Para determinar o número N seria necessário fazer um estudo para descobrir o melhor indicador. Por exemplo, na quarta solução com arquivos quebrados a cada 230.000 registros o tamanho médio do arquivo compactado obtido foi de 434,4 Kbytes

que levava em torno de 14,6 segundos para ser transferido para o nó da grade, em rede local.

Abaixo apresentamos uma proposta de algoritmo utilizando esta abordagem:

- Para cada arquivo da Maciça faça
 - Quebrar o arquivo original a cada N registros
 - A cada arquivo gerado disparar threads para
 - Compactar o arquivo
 - Gerar um trabalho
 - Submeter o trabalho para a grade
- Monitorar a execução dos trabalhos na grade
- Executar a consolidação dos resultados obtidos

O ideal seria trabalhar com arquivos pequenos que possam ser carregados para a memória rapidamente e gerar as tarefas, mas o processo de quebra do arquivo original em arquivos menores também não compensou para uma única agregação devido à lentidão do dispositivo de armazenamento secundário.

Uma segunda proposta é pensando na aplicação como um todo. O conjunto de dados da maciça necessários para processar as 53 agregações pode ser reduzido a poucos campos, sendo assim, um pré-processamento de dados sobre a base da maciça disponibilizaria somente os dados necessários para a agregação proporcionando ganhos no tempo de execução. Além disso, observa-se que a maior parte do tempo utilizada na grade diz respeito à transferência de dados. Pode-se então obter melhor desempenho colocando mais funcionalidades no programa de agregação, ou seja, colocar mais agregações concentradas num único nó para um determinado conjunto de dados similar.

Outro caminho que pode ser mais bem explorado é utilizar uma base de dados distribuída (*Distributed Database* [DER06]) para minimizar o tempo de transferência na grade. Simulamos esta situação colocando os dados em porções de 100.000 registros cada em um local compartilhado na rede e na execução da tarefa na grade os arquivos de dados foram transferidos pela aplicação remota ao invés do arquivo de dados ser transferido através dos recursos da grade. Pode-se observar na tabela 18 que houve ganho no tempo de transferência dos dados na sub-cláusula *Init*, o que compensa a perda sofrida na execução da aplicação remota (sub-cláusula *Remote*). No total houve ganho de 52% na execução de uma tarefa na grade na solução usando uma base de dados distribuída. No computo geral da aplicação usando um base de dados distribuída não houve ganho significativo de performance, em média 3,1%, quando comparado com a quarta solução.

Tabela 18 - Comparativo entre Quarta Solução e Solução com Base Distribuída

Tempo na grade (ms)	<i>Init</i>	<i>Remote</i>	<i>Final</i>	Total
Quarta Solução	13.686,30	4.831,90	4.282,30	22.800,50
% Relativo à Quarta Solução	60%	21%	19%	100%
Solução com Base Distribuída	460,3	6.619,00	3.916,00	10.995,20
% Relativo à Solução com Base Distribuída	4%	60%	36%	100%
Ganho / Perda	13.226,00	-1.787,10	366,3	11.805,30
%	97%	-37%	9%	52%

Terminando as propostas, o ideal é que na geração da exportação dos dados no mainframe, somente os dados necessários para a agregação fossem exportados. No exemplo trabalhado foram tratados 16 arquivos com 1,5 GB de dados em média e foram utilizados somente 4% deste total para processar a agregação. Usando o mesmo raciocínio para as outras agregações foi identificado que são necessários somente 6,6% ou 93 caracteres da base total da maciça para processar as 53 agregações desta base de dados, pois a maioria dos campos são comuns às agregações, alterando-se somente o critério pelas quais são totalizadas. A exportação somente dos dados necessários à agregação como um todo da base de dados da maciça geraria ganhos de performance à medida que reduz a quantidade de dados a ser processada e o algoritmo da aplicação remota, que antes processava somente uma agregação, agora pode ser utilizado para processar *N* agregações diferentes.

5.4.6.1. Comparando o Tempo da Solução Distribuída com o Tempo Real da Aplicação

Atualmente o processo de ETL do banco de dados do MAIPREV leva 45 dias para processar os 24 GBytes de informações da Maciça, 300 MBytes de dados do PAB e 600 MBytes de dados da Concessão. Não foi possível obter o desmembramento de tempo para cada base e para os demais processos como o de carga do banco de dados com os resultados obtidos. Mas numa rápida análise é possível verificar que haverá ganho otimizando a aplicação e utilizando grades de computadores para processar os dados. Veja na tabela 19 que no processamento do arquivo da maciça são geradas 53 agregações e se multiplicarmos pelo tempo médio da melhor solução distribuída (quarta), em menos de um dia teremos os dados consolidados, sem considerar que ainda há margem para otimização da aplicação quando vista como um todo, conforme relatado anteriormente.

Tabela 19 - Estimando tempo para consolidação da Maciça

	1 Agregação	53 Agregações
Aplicação Sequencial	00:59:07	2 dias 04:13:11 hs
Quarta Solução	00:23:03	20:21:39 hs
Solução Maiprev	---	45 dias

5.4.7. Aspectos de Segurança

A diretriz geral do governo para segurança é a implantação da infra-estrutura de chave pública no âmbito do Poder Executivo Federal. Para a Previdência Social, deve-se implementar não apenas a infra-estrutura de chave pública, mas também políticas e padrões de segurança robustos e aprovados que garantam a integridade e o sigilo das informações, possibilite o controle de acesso, a detecção de vírus e assinatura digital.

Além disso, uma política de segurança uniforme deve ser desenvolvida, contemplando todas as unidades da Previdência Social e definindo os padrões e diretrizes de segurança física e lógica a serem implementadas. As políticas deverão incluir aspectos relativos aos procedimentos de backup e recuperação de dados e sistemas, autorização de acesso a dados e os sistemas de monitoramento geral. Deve-se instrumentalizar a infra-estrutura de segurança com modernas ferramentas de *hardware* e *software* que permitam a gerência e monitoração da segurança em toda Previdência Social

As informações tratadas pela Assessoria de Pesquisas Estratégicas e de Gerenciamento de Riscos no combate à evasão fiscal e crimes previdenciários são sensíveis e objeto de políticas de Segurança da Previdência Social. As grades de computadores fornecem uma infra-estrutura transparente para a execução de aplicações, ou seja, os dados e a aplicação são submetidos aos hosts remotos sem a interferência do usuário. Neste contexto, é necessário que mecanismos de segurança garantam que as informações trafegadas na rede e armazenadas nos nós da grade sejam utilizadas somente pela aplicação distribuída.

Esta segurança deve ser fornecida pelo *middleware* de grade e deve garantir a confidencialidade e a integridade dos dados trocados entre as entidades da grade. Um dos motivos pelos quais o OurGrid foi escolhido para os testes foi a segurança.

A comunicação no OurGrid é realizada através de troca de mensagens implementada no servidor XMPP. O protocolo XMPP fornece vários níveis de segurança a nível de protocolo. A identificação no XMPP pode ser obtida através da utilização de certificados válidos que confirmam a identidade do usuário.

Existem dois tipos de encriptação com XMPP usados na comunicação cliente-a-servidor. O SASL (*Simple Authentication and Security Layer*) é utilizado durante o estabelecimento e autenticação da conexão. O TLS (*Transport Layer Security*) é utilizado para criptografar as transmissões entre o cliente e o servidor após o estabelecimento da conexão. Além do SASL e TLS, o mecanismo *server dialback* pode ser usado para proteção contra falsificação de domínio na comunicação servidor-a-servidor.

Além disso, as comunicações XMPP utilizam conexões na porta 5222 (cliente-a-servidor) ou porta 5269 (servidor-a-servidor), conforme registrado com a IANA (*Internet Assigned Numbers Authority*), possibilitando a configuração adequada dos firewalls da entidade.

Por outro lado, a aplicação transfere dados para os nós da grade e estes ficam armazenados. Para evitar o acesso indevido a estes dados, o OurGrid implementa a virtualização para executar as aplicações remotamente, garantindo que os dados não serão acessados indevidamente por terceiros. Assim, as tarefas podem ser executadas em máquinas virtuais sem acesso à rede e mesmo as máquinas sejam afetadas em algum instante, a cada tarefa ela estará em seu estado inicial tornando uma opção segura para execução das tarefas. Atualmente existem duas de máquinas virtuais disponíveis para o OurGrid: o Vserver para Linux e VMWare Server para Windows e Linux.

Utilizando os recursos de segurança oferecidos pelo *middleware* de grade garante-se que os dados estarão seguros, íntegros e confiáveis liberando o desenvolvedor a aplicar seus esforços no desenvolvimento de uma aplicação que consiga extrair o máximo da infra-estrutura da grade.

6. CONCLUSÃO E TRABALHOS FUTUROS

As grades computacionais disponibilizam recursos computacionais para o desenvolvimento de novas aplicações nas mais diversas áreas, como a saúde, a engenharia, a física, a pesquisa, o comércio, entre outros, de forma independente ou colaborativa. Esta colaboração só é possível devido à característica distribuída das grades computacionais.

Foi apresentado neste trabalho que o Governo Brasileiro, através do e-Gov, estabelece diretrizes e políticas voltadas para a promoção da cidadania, definindo padrões e normas a serem adotados por todos os entes federativos. A contribuição deste trabalho é fornecer um referencial para a migração de aplicações de e-Gov em ambiente de grades computacionais levando em consideração os princípios do Governo Eletrônico.

Com este trabalho verificou-se que é possível atender a alguns dos requisitos de e-Gov, sob certas condições, com a utilização de grades computacionais beneficiando os cidadãos brasileiros. O uso desta tecnologia possibilita a prestação de serviços de melhor qualidade e maior rapidez. O estudo de aplicações de e-Gov e a modelagem delas executando sobre grades computacionais procura atender a alguns dos princípios do Governo Eletrônico:

- A utilização do *software* livre como recurso estratégico é proporcionado pela utilização de *middleware* aberto (OurGrid) para grades computacionais sobre ambientes com sistemas operacionais Linux;
- A utilização de infra-estrutura de grades proporciona uma melhor utilização dos recursos computacionais ociosos atendendo à diretriz “racionalização dos recursos”;
- A adoção de políticas, normas e padrões comuns garantem a interoperabilidade. Recomenda-se a utilização de sistemas operacionais abertos (Linux, FreeSBD, etc.), ambiente de desenvolvimento e linguagens de programação abertos (Java), padrão XML para troca de dados, etc., possibilitando a troca de dados e execução de aplicações em independente da plataforma tecnológica utilizada pelo usuário;
- A disponibilização de serviços públicos pode ser realizada através dos Portais do Governo sustentados por uma infra-estrutura de grades para o processamento e armazenamento de dados.

Na aplicação estudada, considera-se que o formato de arquivo texto não é o padrão a ser utilizado para troca de dados do Governo Eletrônico. Por questões internas o que foi

disponibilizado para o sistema MAIPREV estava neste formato. Neste caso, recomenda-se a utilização do XML como padrão para troca de dados. Também, recomenda-se a utilização de equipamentos com melhor performance para o processamento das aplicações, utilizando mecanismos de armazenamento secundário de alta velocidade e com quantidade de memória suficientemente grande para comportar todo o conjunto de dados a ser processado. Como recursos computacionais no Governo Federal são distribuídos por todo o território nacional, a utilização de bases distribuídas para o processamento das aplicações na grade é uma das alternativas que minimizam o tempo de transmissão de dados na grade. A informação é o ativo mais precioso de uma instituição e esta deve estar disponível, íntegra e confiável para os usuários que possuem permissão de acesso a ela. Os mecanismos de segurança devem ser implementados a nível de *middleware* da grade para que o desenvolvedor se preocupe somente no desenvolvimento de uma aplicação que explore eficientemente todos os recursos disponibilizados pela infra-estrutura de grades.

A implementação de uma porção da aplicação MAIPREV em grade obteve um fator de aceleração de 2,56 quando comparado com a aplicação sequencial, mas quando comparada com a versão atualmente em uso o ganho foi mais significativo. Um dos motivos foi a grande massa de dados (24 GB) a ser manipulado com aproveitamento para a aplicação de somente 4%. A aplicação tomou maior parte do tempo separando os dados do que propriamente processando os dados para geração da aplicação. Pode-se dizer que o tempo de processamento da aplicação foi quase igual ao tempo de separação dos dados, pois como as tarefas foram executadas em paralelo, eram rapidamente executadas, restando um pequeno tempo adicional para o retorno dos dados e consolidação do resultado final. Neste caso trabalhou-se com uma pequena parte da aplicação.

Ao analisar com a aplicação como um todo se verificou que serão necessários somente 6,6% da massa total de dados para processar as agregações da maciça. Neste caso, seria necessária somente uma operação de separação dos dados para serem utilizados por todas as agregações da maciça. Tendo somente um processo de separação de dados e processando todas as agregações da maciça acredita-se que para a aplicação como um todo haverá um ganho significativo de tempo de processamento. A solução a ser implementada usaria técnicas como *Memory Filter*, *Pipeline Dispatching* e *Distributed Database* [DER06] para submissão das tarefas à grade. Não foi possível mensurar o tempo devido à falta de tempo disponível para melhorar os estudos.

Como sugestões de trabalhos futuros incluem-se:

- Implementação da aplicação MAIPREV por completo em ambiente de grades;
- Modelagem e implementação de outras aplicações de e-Gov em ambiente de grades.

Este trabalho é o ponto inicial para outros que podem proporcionar à sociedade elementos que auxiliem na promoção da cidadania, na integração com outros entes federativos e no uso racional dos recursos computacionais, atendendo às diretrizes do Governo Eletrônico.

REFERÊNCIAS

- [ABO07] Abbott, B.; Baranovski, A.; Diesburg, M.; Garzoglio, G.; Kurca, T.; Mhashilkar, P. "DZero Data-Intensive Computing on the Open Science Grid". In: International Conference on Computing in High Energy and Nuclear Physics (CHEP07), vol. 1, Setembro 2007, 9p.
- [ALV09] Alvares, L. "Gestão do Conhecimento". Capturado em: <http://sebastiaoilima.googlepages.com/GestaoDoConhecimentoLilianAlvares.ppt>, Março 2009.
- [ARC01] Arcieri, F.; Cappadozzi, E.; Nardelli, E.; Talamo, M. "SIM: a working example of an e-government service infrastructure for mountain communities". In: 12th International Workshop on Database and Expert Systems Applications, vol. 1, 2001, pp. 407-411.
- [BER03] Berman, F.; Fox, G.; Hey, T. "The Grid: Past, Present, Future". In: Grid Computing - Making the Global Infrastructure a Reality. John Wiley & Sons, 2003, capítulo 1, pp. 9-50.
- [CIR03] Cirne, W.; Brasileiro, F.; Sauv e, J.; Andrade, N.; Paranhos, D.; Santos-Neto, E.; Medeiros, R. "Grid Computing for Bag of Tasks Applications". In: Third IFIP Conference on E-Commerce, E-Business and E-Government, Setembro 2003, 27p.
- [CON08] Condor Project. "What is Condor?". Capturado em: <http://www.cs.wisc.edu/condor/description.html>, Janeiro 2008.
- [DAT07] Dataprev. "CNIS - Cadastro Nacional de Informa es Sociais". Capturado em: <http://www.dataprev.gov.br/servicos/cnis/cnis.shtm>, Dezembro 2007.
- [DAT08] Dataprev. "Relat rio de Gest o". Capturado em: <http://www.dataprev.gov.br/relgestao/Relatorio de Gestao 2004.pdf>, Janeiro 2008.
- [DER06] De Rose, C. A. F.; Ferreto, T.; Farias, M. B.; Dias, V. G.; Cirne, W.; Oliveira, M. P. M.; Saikoski, K. B.; Danielecki, M.L. "GerpavGrid: Using the Grid to Maintain the City Road System". In: 18th Symposium on Computer Architecture and High Performance Computing (SBAC-PAD), vol. 1, Outubro 2006, pp. 73-80.
- [DER08] De Rose, C. A. F.; Navaux, P. O. A. "Arquiteturas Paralelas". Editora Bookman, 2008, 152p.

- [DOE09] DOE. "U.S. Department Of Energy". Capturado em: <http://www.energy.gov>, Abril 2009.
- [DTS08] DATASUS. "SIM - Sistema de Informações de Mortalidade". Capturado em: http://w3.datasus.gov.br/datasus/datasus.php?area=361A3B368C7D469E1F368G12HIJd3L1M0N&VInclude=../site/din_sist.php&VSis=1&VAba=0&VCoit=469, Janeiro 2008.
- [EGE09] EGEE. "EGEE Portal: Enabling Grids for E-sciencE". Capturado em: <http://www.eu-egee.org>, Abril 2009.
- [FAV05] Favaretto, F. "Proposta de medição da qualidade da informação". In: XII Simpósio de Engenharia de Produção, vol. 1, 2005, pp. 1-7.
- [FER09] Fermi National Accelerator Laboratory. "DZero Experiment". Capturado em: <http://www-d0.fnal.gov>, Abril 2009.
- [FOS01] Foster, I.; Kesselman, C.; Tuecke, S. "The Anatomy of the Grid. Enabling Scalable Virtual Organizations". International Journal of Supercomputer Applications, vol 15-3, Agosto 2001, pp. 200-222.
- [FOS02] Foster, I. "What is the Grid? A Three Point Checklist". GRIDToday, vol. 1-6, Julho 2002, 4p.
- [FOS06] Foster, I. "Globus ToolKit 4: *Software* for Service-Oriented Systems". In: IFIP International Conference on Network and Parallel Computing, Springer-Verlang LNCS 3779, 2006, pp. 2-13.
- [FOS99] Foster, I.; Kesselman, C. "Computational Grids". In: The Grid: Blueprint for a New Computing Infrastructure. Morgan-Kaufman, 1999, Capítulo 2, 29p.
- [FOX02] Fox, G. "e-Science meets computational science and information technology". Computing in Science and Engineering, vol. 4-4, Julho 2002, pp. 84_85.
- [FUX05] Fu, X.; Peng, D.; Xu, H.; Lu, Y.; Xiao, S. "Research of e-government information portal applications based on grid technology". In: Ninth International Conference on Computer Supported Cooperative Work in Design, vol. 1, Maio 2005, pp 384-389.
- [GFB07a] Governo Federal do Brasil. "SOFTWARE LIVRE – Diretrizes". Capturado em: <http://www.softwarelivre.gov.br/diretrizes>, Dezembro 2007.

- [GFB07b] Governo Federal do Brasil. "O que é *Software* Livre?". Capturado em: <http://www.softwarelivre.gov.br/SwLivre>, Dezembro 2007.
- [GFB08a] Governo Federal do Brasil. "Portal Inclusão Digital". Capturado em: <http://www.inclusaodigital.gov.br/inclusao>, Novembro 2008.
- [GFB08b] Governo Federal do Brasil. "Portal do *Software* Público Brasileiro". Capturado em: <http://www.softwarepublico.gov.br>, Novembro 2008.
- [GFB09a] Governo Federal do Brasil. "Portal do Governo Brasileiro". Capturado em: http://www.brasil.gov.br/governo_federal, Abril 2009.
- [GFB09b] Governo Federal do Brasil. "Lei nº 7.444, de 20 de dezembro de 1985". Capturado em: <http://www.planalto.gov.br/CCIVIL/leis/1980-1988/L7444.htm>, Abril 2009.
- [GLI09] GLite. "EGEE > gLite". Capturado em: <http://glite.web.cern.ch/glite>, Abril 2009.
- [GLO08] The Globus Alliance. "Globus ToolKit". Capturado em: <http://www.globus.org/toolkit/about.html>, Janeiro 2008.
- [GOL04] Goldchleger, A. "InteGrade: Um Sistema de *Middleware* para Computação em Grade Oportunista". Dissertação de Mestrado, Departamento de Ciência da Computação, IME/USP, 2004, 124p.
- [GOR08] Gorton, I.; Green_eld, P.; Szalay, A.; Williams, R. "Data- Intensive Computing in the 21st Century". IEEE Computer Society, vol. 41-4, Abril 2008, pp. 30-32.
- [IBG08] IBGE. "Pesquisa Nacional por Amostra de Domicílios". Capturado em: <http://www.sidra.ibge.gov.br/pnad/default.asp>, Janeiro 2008.
- [KON08] Kon, F.; Goldman, A. "Minicurso: Grades Computacionais: Conceitos Fundamentais e Casos Concretos". In: Jornada de Atualização em Informática (JAI 2008) do Congresso da Sociedade Brasileira de Computação, 2008, 50p.
- [KRA02] Krauter, K.; Buyya, R.; Maheswaran, M. "A Taxonomy and Survey of Grid Resource Management Systems for Distributed Computing". *Software: Practice & Experience*, vol. 32-22, Fevereiro 2002, pp. 135_164.
- [LIV97] Livny, M.; Basney, J.; Raman, R.; Tannenbaum, T. "Mechanisms for High Throughput Computing". *SPEEDUP Journal*, vol. 11-1, 1997, 6p.

- [MAA08] Maad, S. Coghlan, B. "Assessment of the Potential use of Grid Portal Features in e-Government". Journal: Transforming Government: People, Process and Policy, vol. 2-2, 2008, pp. 128-138.
- [MAU03] Mauher, M. "Government network to serve e-government requirements". In: 7th International Conference on Telecommunications (ConTEL 2003), vol 1, 2003, pp. 1-12.
- [MPO07a] Ministério do Planejamento, Orçamento e Gestão. "Governo Eletrônico – Governo Eletrônico". Capturado em: <http://www.governoeletronico.gov.br>, Novembro 2007.
- [MPO07b] Ministério do Planejamento, Orçamento e Gestão. "Guia de Inovação Tecnológica em Cluster e Grid". Capturado em: <http://guialivre.governoeletronico.gov.br/guiaonline/guiacluster>, Novembro 2007.
- [MPO07c] Ministério do Planejamento, Orçamento e Gestão. "Histórico - Governo Eletrônico". Capturado em: <http://www.governoeletronico.gov.br/ogov.br/historico>, Novembro 2007.
- [MPO07d] Ministério do Planejamento, Orçamento e Gestão. "Princípios e Diretrizes - Governo Eletrônico". Capturado em: <http://www.governoeletronico.gov.br/ogov.br/principios>, Novembro 2007.
- [MPO07e] Ministério do Planejamento, Orçamento e Gestão. "e-PING – Governo Eletrônico". Capturado em: <http://www.governoeletronico.gov.br/acoes-projetos/e-ping-padres-de-interoperabilidade>, Dezembro 2007.
- [MPO07f] Ministério do Planejamento, Orçamento e Gestão. "Guia Livre – Governo Eletrônico". Capturado em: <http://www.governoeletronico.gov.br/acoes-projetos/guia-livre>, Dezembro 2007.
- [MPO08a] Ministério do Planejamento, Orçamento e Gestão. "e-MAG – Modelo De Acessibilidade De Governo Eletrônico". Capturado em: <http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG>, Novembro 2008.
- [MPS06] Ministério da Previdência Social. "Portaria MPAS nº 36, de 6 de Fevereiro de 2006". Publicada no D.O.U em 07/02/2006
- [MPS07] Ministério da Previdência Social. "Ministério da Previdência Social". Capturado em: <http://www.mps.gov.br>, Novembro 2007.

- [MPS08a] Ministério da Previdência Social. “AgPREV”. Capturado em: http://www.previdencia.gov.br/agprev/agprev_mostraNoticia.asp?Id=26488&ATVD=1&xBotao=1, Janeiro 2008.
- [MPS08b] Ministério da Previdência Social. “Boletim Estatístico da Previdência Social BEPS”. Capturado em: http://www.previdenciasocial.gov.br/pg_secundarias/previdencia_social_13_05.asp, Janeiro 2008.
- [MPS09] Ministério da Previdência Social. “Ata da 149ª REUNIÃO ORDINÁRIA do CONSELHO NACIONAL DE PREVIDÊNCIA SOCIAL (CNPS)”. Capturado em: http://www.mps.gov.br/arquivos/o_ce/3_090316-160109-326.doc, Abril 2009.
- [NAT08] Natrajan, A. Nguyen-Tuong, A. Humphrey, M. A. Grimshaw A. S. “The Legion Grid Portal”. Capturado em: <http://legion.virginia.edu/papers/GCE01.pdf>, Março 2008.
- [NSF09] NSF. “nsf.gov - National Science Foundation - US National Science Foundation (NSF)”. Capturado em: <http://www.nsf.gov>, Abril 2009.
- [OGF09] OGF. “Open Grid Forum”. Capturado em: <http://www.ogf.org>, Abril 2009.
- [ONI08] Onid. “ONID: Observatório Nacional de Inclusão Digital”. Capturado em: <http://www.onid.org.br>, Novembro 2008.
- [OSG09] OSG. “Open Science Grid Home Page”. Capturado em: <http://www.opensciencegrid.org>, Abril 2009.
- [OUR07] OurGrid Community. “OurGrid. What is OurGrid”. Capturado em: <http://www.ourgrid.org>, Novembro 2007.
- [POR07] Pordes, R.; Petravick, D.; Kramer, B.; Olson, D.; Livny, M.; Roy, A.; Avery, P.; Blackburn, K.; Wenaus, T.; Würthwein, F.; Foster, I.; Gardner, R.; Wilde, M.; Blatecky, A.; McGee, J.; Quick, R. “The Open Science Grid”. Journal of Physics: Conference Series, vol. 78, 2007, 15 p.
- [POR08a] Pordes, R. “Challenges facing production grids”. In: 16th High Performance Computing and Grids in Action. Advances in Parallel Computing, vol. 1, 2008, pp. 506-521.
- [POR08b] Pordes, R.; Altunay, M.; Avery, P.; Bejan, A.; Blackburn, K.; Blatecky, A.; Gardner, R.; Kramer, B.; Livny, M.; McGee, J.; Potekhin, M.; Quick, R.; Olson, D.; Roy, A.; Sehgal, C.; Wenaus, T.; Wilde, M.; Würthwein, F. “New science on

the Open Science Grid". Journal of Physics: Conference Series, vol. 125, 2008, 6p.

- [SAN07] Santos, R. S. "Governo Eletrônico para Todos". Capturado em: <http://www.camara.gov.br/Internet/Eventos/IPAIT/Documentos/S%C3%ADtio%20-%20III%20IPAIT/Portugu%C3%AAs/Painel%201/Painel%20Brasil%20-%20Rogerio%20Santanna.pdf>, Dezembro 2007.
- [SEG09] SegHidro. "Projeto SegHidro". Capturado em: <http://seghidro.lsd.ufcg.edu.br>, Março 2009.
- [SIL06] Silva Filho, A. M. "Os Três Pilares da Gestão do Conhecimento". Revista Espaço Acadêmico, vol. 58, Março 2006, 2p.
- [SUM08] Sumathi, P.; Punithavalli, M. "Constructing a Grid Simulation for E-Governance Applications Using GridSim". Journal of Computer Science, vol. 4-8, 2008, pp. 674-679.
- [TAM09] Tamanduá. "Projeto Tamanduá". Capturado em: <http://tamandua.speed.dcc.ufmg.br>, Março 2009.
- [TER09] TeraGrid. "TeraGrid". Capturado em: <http://www.teragrid.org>, Abril 2009.
- [TSE08] TSE. "Tribunal Superior Eleitoral – TSE". Capturado em: <http://www.tse.gov.br/internet/index.html>, Novembro 2008.
- [VDT09] VDT. "Virtual Data Toolkit". Capturado em: <http://vdt.cs.wisc.edu>, Abril 2009.
- [XUE08] Xue, Y.; Wan, W.; Li, Y.J.; Guang, J.; Bai, L.Y.; Wang, Y.; Ai, J.W. "Quantitative Retrieval of Geophysical Parameters Using Satellite Data". IEEE Computer Society, vol. 41-4, Abril 2008, pp. 33-40.
- [WIK09] Wikipedia. "Grid Computing". Capturado em: http://en.wikipedia.org/wiki/Grid_computing, Abril 2009.