

GESTÃO DE PERFORMANCE DA APLICAÇÃO MES DA CSN¹

Carlos Henrique Novaga Alves²
Fábio de Carvalho Seabra³
João Batista Nogueira de Oliveira⁴
Leandro de Oliveira Minotti⁵
Leandro Oliveira Barcelos⁶
Lucio Gomes Peixoto Junior⁷
Márcio José Ramalho Lima⁸

Resumo

Este trabalho apresenta a metodologia aplicada ao trabalho de gestão de performance realizado no sistema MES da Companhia Siderúrgica Nacional (CSN), um sistema crítico em operação contínua em regime 24 x 7, responsável pelo controle de produção da unidade Presidente Vargas, situada em Volta Redonda-RJ. A utilização contínua do sistema gerou um grande aumento na quantidade de dados armazenados e também o surgimento de uma série de novas funcionalidades com inúmeras regras complexas de negócio que conseqüentemente, passaram a exigir uma carga maior de processamento das máquinas envolvidas. O impacto de uma possível degradação do sistema gerou lentidão no processo produtivo e também nas tomadas de decisões. A comunicação com os sistemas de Chão de Fábrica (passou a não atender a tempo aos requisitos de *Setup* das várias Linhas Operacionais, acarretando muitas vezes um acréscimo de mensagens de solicitação de Dados Primários, prejudicando ainda mais o processamentos das mesmas. Identificou-se a necessidade de se otimizar o sistema e levá-lo a uma qualidade e desempenho o mais próximo possível do momento de sua partida a 6 anos atrás. Além da redução de uso dos recursos do sistema, o resultado deste trabalho aumentou a estabilização do sistema e conseqüente aumento nos índices de aceitação pelos usuários.

Palavras-chave: Performance; MÊS; Sistema crítico.

MANAGING PERFORMANCE OF THE CSN MES SYSTEM⁹

Abstracts

This paper presents the methodology applied to the performance management of carried through in the MES of the Companhia Siderúrgica Nacional (CSN), a critical system in continuous operation in 24 x 7 regimen, responsible for the production control of President Vargas plant, located in Volta Redonda-RJ. The continuous use of the system also generated a great increase in stored data amount and the sprouting of a series of new functionalities with innumerable complex business rules. These rules had started to demand a bigger load of processing of the involved machines. The impact of a possible degradation of the system also generated slowness in the productive process and in the decisions taking. The communication with shop floor systems was not fulfilling in time the setup requirements in some production lines, causing an addition of request messages of Primary Data, damaging further their processing. It has become substantial optimizing the system in order to get its quality and the performance to the moment of its going 6 years ago. Beyond the reduction of resources use, the result of this work increased the stabilization of the system and consequence increase the indices of acceptance for the users.

Key words: MES; Critical application; Performance.

¹ Trabalho técnico apresentado ao X Seminário de Automação de Processos, 4 a 6 de outubro de 2006, Belo Horizonte – MG.

² Gerente de Projetos, Chemtech

³ Analista de Sistemas, Chemtech

⁴ Analista de Negócios, Companhia Siderúrgica Nacional

⁵ Analista de Sistemas, Chemtech

⁶ Analista de Suporte a Sistemas Operacionais, Companhia Siderúrgica Nacional

⁷ Analista de Suporte a Banco de Dados, Companhia Siderúrgica Nacional

⁸ Coordenador de TI, Companhia Siderúrgica Nacional

INTRODUÇÃO

Quando a CSN decidiu implementar seu modelo integrado de negócios (MIN), com uma camada MES que armazena todos os dados de produção e qualidade da Usina Presidente Vargas, surgiu o desafio: garantir a operação de um sistema crítico em regime contínuo (24 x 7), respondendo rapidamente a situações como erros de procedimento, perda de comunicação, estornos de produção e erros de processamento, de forma a não comprometer o processo produtivo.

No caso da CSN, todos os sistemas produtivos, desde a área de redução até a embalagem de produtos acabados, têm seus dados de produção e qualidade gerenciados pelo MES. Atualmente o MES é um sistema fundamental na manutenção do ritmo de produção da companhia.⁽¹⁾

No sistema MES da CSN, os dados com perfil de séries temporais são armazenados em um historiador de dados de processo PI, enquanto que os dados transacionais são enviados para um banco de dados relacional (Heimdall). Tais informações são armazenadas de forma organizada, possibilitando visualização e pesquisa por parte dos usuários.

Além disso, o MES recebe e interpreta as ordens de processo oriundas do SAP R/3, enviando-as para os sistemas de chão-de-fábrica através de mensagens. Da mesma forma, os dados recebidos do chão-de-fábrica são tratados e enviados ao SAP R/3.

O sistema foi implementado utilizando banco de dados Microsoft SQL Server. Toda a regra de negócios está implementada em procedimentos SQL (*stored procedures* e *triggers*). A interface gráfica foi desenvolvida em VB 6.0. Além do SAP R/3, há também interface com um sistema de armazenamento de dados histórico, o Impromptu. A Figura 1 apresenta a arquitetura lógica do MIN da CSN.

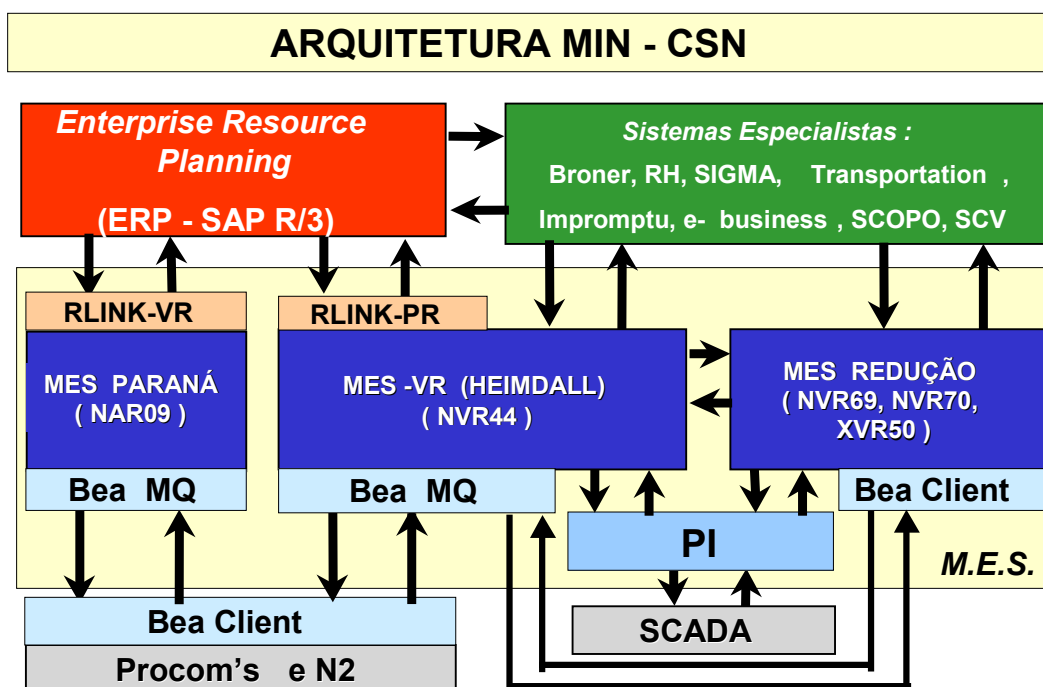


Figura 1. Arquitetura do Modelo Integrado de Negócios da CSN

Os números deste sistema impressionam. A

Tabela 1 apresenta alguns parâmetros que permitem dimensionar o sistema MES CSN.

Tabela 1. Números do sistema MES da CSN

Parâmetro	Valor
Instruções SAP R/3 → MES	14.000 por dia
Mensagens MES → SAP R/3	34.000 por dia
Mensagens trocadas com PROCOMS	13.000 por dia
Tamanho do banco de dados	20 Gb
Número de tabelas no banco de dados	400
Nº de procedures	1300
Nº de Views	120
Nº de Triggers	180
Nº de Jobs	50
Nº de usuários com permissão de acesso	3000
Nº de usuários simultâneos	150
Nº de telas	500
Nº de linhas de código	350.000

Na concepção do sistema MES optou-se por utilizar a arquitetura cliente-servidor, que se apresentava robusta e eficaz na época, conforme apresentado na Figura 2.

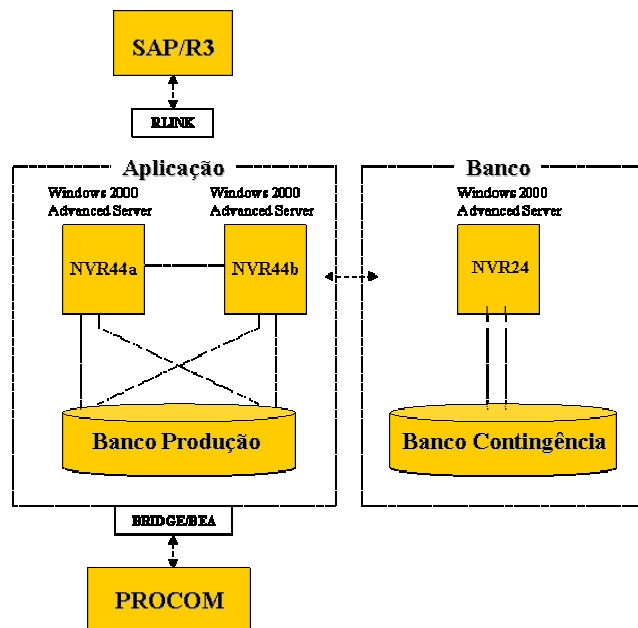


Figura 2. Arquitetura do sistema MES

A utilização contínua do sistema gerou um grande aumento na quantidade de dados armazenados e também o surgimento de uma série de novas funcionalidades com inúmeras regras complexas de negócio que conseqüentemente, passaram a exigir uma carga maior de processamento das máquinas envolvidas. O impacto de uma possível degradação do sistema gerou lentidão no processo produtivo e também nas tomadas de decisões. A comunicação com os sistemas de Chão de Fábrica (PROCOM's - Process Computers) passou a não atender a tempo aos requisitos de

Setup das várias Linhas Operacionais, acarretando muitas vezes um acréscimo de mensagens de solicitação de Dados Primários, prejudicando ainda mais o processamentos das mesmas. Identificou-se a necessidade de se otimizar o sistema e levá-lo a uma qualidade e desempenho o mais próximo possível do momento de sua partida a 6 anos atrás.

A performance de um sistema é em muitas das vezes medida pelo tempo de resposta para uma aplicação. Garantir ou manter uma boa performance, é resultado de estressantes análises e acompanhamento de alguns recursos:

- Recursos de Sistema (hardware) – disco, memória e CPU.
- Sistema Operacional – parametrização do sistema.
- Banco de Dados – parametrização do banco de dados, análise de tabelas, tipos de dados utilizados, índices clusters e nonclusters.
- Aplicação – utilização de códigos mais performáticos.
- Rede – análise e parametrização dos meios físicos de acesso ao servidor.

A Figura 3, demonstra os recursos que foram aplicados no trabalho de performance, seus respectivos ganhos e custos.

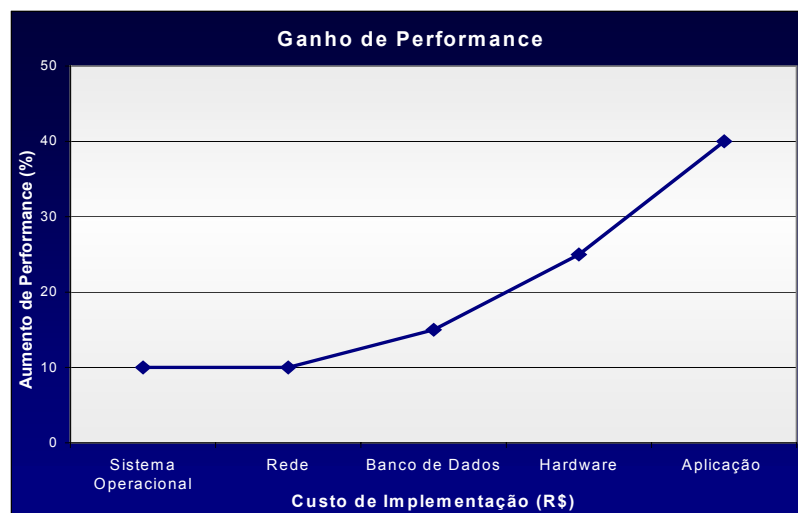


Figura 3. Ganho de performance por área de atuação.⁽²⁾

MATERIAIS E MÉTODOS

As ações realizadas na aplicação propriamente foram divididas nas seguintes etapas:

1. Sistema Operacional:
 - a. Por padrão, o Microsoft Windows 32 bits, quando tem 4GB de memória, reserva 2GB de memória para o próprio sistema operacional, e os outros 2 GB endereça para as aplicações. Assim, o sistema operacional foi parametrizado para endereçar 3GB de memória para as aplicações e somente 1GB ficou reservado para o sistema operacional.
2. Rede:
 - a. O sistema de placas de rede passou a ser configurado, em TEAM (load balance), ao invés de TEAM (Fault Tolerance). Proporcionando que o sistema de rede atendesse uma maior quantidade de solicitações com maior performance.

3. Bando de dados:
 - a. Os backups do banco de dados, foram otimizados e passaram a ser feitos da máquina que está como Standby Server. Diminuindo drasticamente esses acessos de backup na máquina de banco de dados principal.
 - b. Todos os jobs que rodavam no banco de dados, passaram por um racionamento de recursos. Este trabalho identificou vários jobs que poderiam ter seus *schedules* racionados.
4. Hardware:
 - a. O sistema de discos externos, recebeu uma atenção especial e foi totalmente reconfigurado e refeito com a adição de discos mais performáticos. A controladora do array de discos recebeu também mais memória e mais um canal de comunicação. Desta forma, passou-se a trabalhar paralelamente a velocidade de I/O de 320MB/s.
5. Aplicação
 - a. Reestruturação de índices (*Cluster* e *NonCluster*) gerando utilização mais freqüente dos mesmos nas consultas e criação de rotinas de desfragmentação;
 - b. Redefinição dos níveis de isolamento nas transações reduzindo-as ao menor possível diminuindo tempo de travamento (*lock*) de objetos;
 - c. Realização de análises detalhadas das operações, utilizando os planos de execução, diminuindo recompilações e promovendo maior utilização do Buffer Cache;
 - d. Grande parte das tabelas sofreram uma reestruturação no sentido de se definir os melhores tipos de dados a se utilizar e a criação de planos para manutenção (limpeza e *archiving*) através de operações agendadas. Sofreram também uma limpeza de dados manualmente de forma a se obter resultados imediatos além de atualizações de estatísticas;
 - e. Elaboração de um guia de boas práticas de programação que é constantemente seguido e atualizado pelos desenvolvedores e mantenedores do sistema além de relatórios mensais feitos pela equipe de banco de dados apontando objetos específicos a serem reformulados.

RESULTADOS

A Figura 4 apresenta o consumo de CPU e o tamanho da fila de CPU do servidor desde o início dos trabalhos de performance.

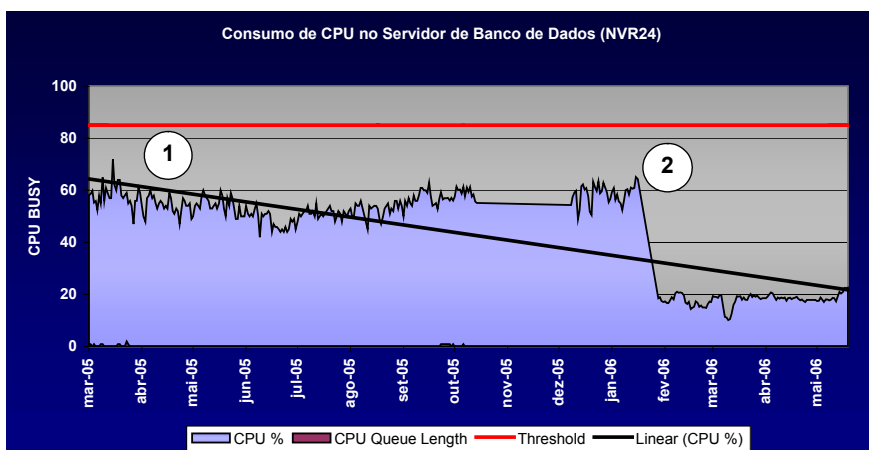


Figura 4. Consumo de CPU no servidor de banco de dados

A partir do evento 1, exibido na figura acima, foi realizado um trabalho de identificação dos objetos cujo número de recompilações de *stored procedures* eram muito altos.

O evento 2 representa os trabalhos de *archiving* das tabelas, racionamento da quantidade de execução diária dos *jobs*, criação de índices de tabelas e otimização de transações.

A Figura 5 apresenta o consumo de I/O do servidor desde o início dos trabalhos de performance.

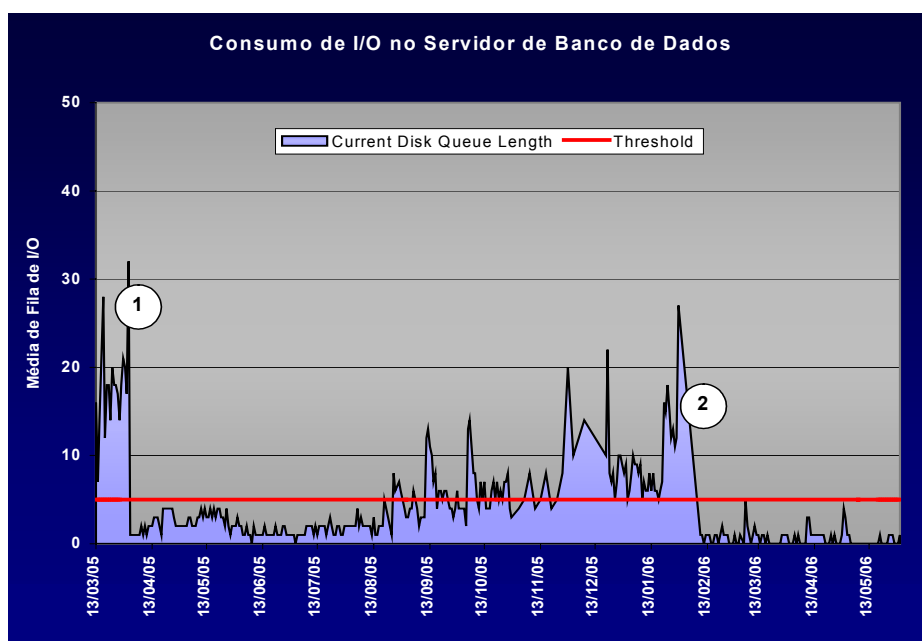


Figura 5. Consumo de I/O no servidor de banco de dados

O evento 1, exibido na figura acima, representa a substituição do servidor principal pelo servidor *standby*, pois este poderia receber uma atualização de *storage* para atender inicialmente a demanda identificada de I/O. Discos mais rápidos foram adicionados ao *storage* complementando a adição de memória *cache* à controladora.

O evento 2 representa os trabalhos de substituição do backup, que era realizado diretamente na servidor principal e passou a ser realizado no servidor de *standby* diminuindo o acesso a disco. Além disso, o *archiving* das tabelas, racionamento da

quantidade de execução diária dos *jobs*, criação de índices de tabelas e otimização de transações também contribuíram para a melhora no consumo de I/O. A Figura 6 apresenta a projeção do consumo da área em disco antes do início dos trabalhos de otimização e performance.

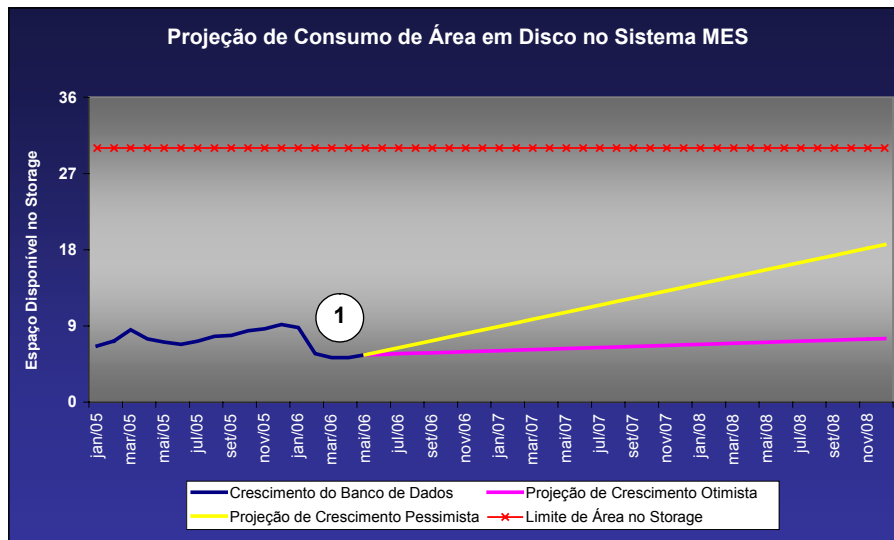


Figura 6. Consumo de área em disco no sistema

O evento 1, exibido na figura acima, mostra a redução e manutenção no consumo de espaço em disco, indicando que a tendência otimista (linha rosa) está sendo mantida em função da constante preocupação com o *archiving* do sistema.

A Figura 7, apresenta um comparativo entre os valores para o Cache Hit Ratio, que representam a 'saúde' do banco de dados em relação a reutilização dos planos de execução armazenados em memórias para cursores, *stored procedures*, *triggers* e outros objetos utilizados pelo banco de dados.

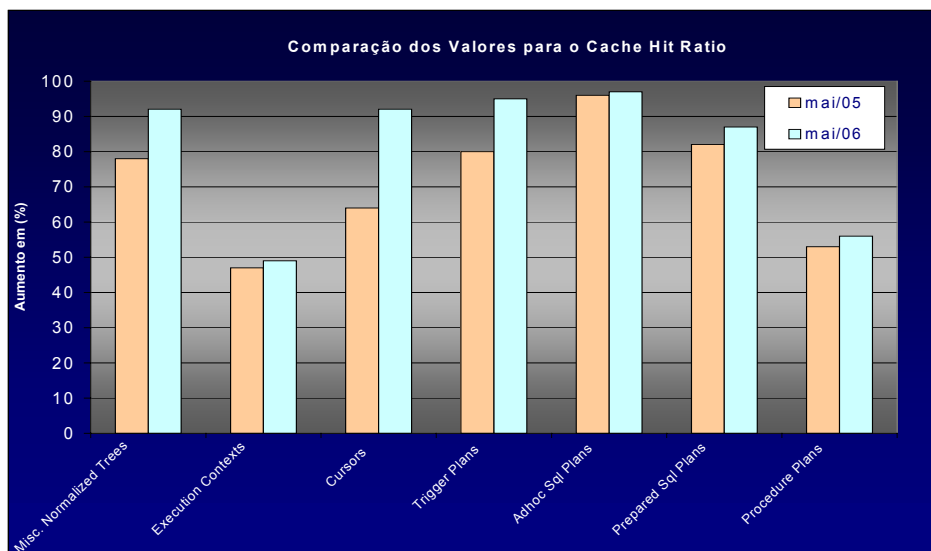


Figura 7. Comparação dos valores de 'saúde' do banco de dados

A Figura 8 apresenta o número de *deadlock* desde o início dos trabalhos de performance.

Um *deadlock* ocorre quando duas transações tem travamento (*lock*) em objetos separados e cada transação solicita *lock* em um outro objeto já em transação. Cada transação aguarda muito pela liberação do outro objeto em *lock*. Um *deadlock* pode

ocorrer quando várias transações longas são executadas simultaneamente no banco de dados.

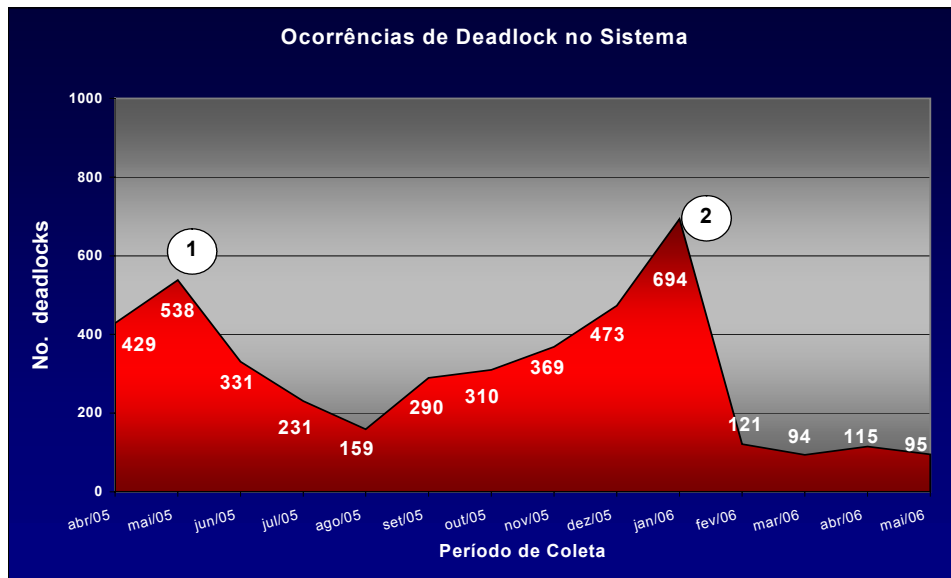


Figura 8. Ocorrência de *deadlocks*

O evento 1, exibido na figura acima, representa a substituição do servidor principal pelo servidor *standby*, pois este poderia receber uma atualização de *storage* para atender inicialmente a demanda identificada de I/O. Discos mais rápidos foram adicionados ao *storage* complementando a adição de memória *cache* à controladora.

O evento 2 representa os trabalhos de *archiving* das tabelas, racionamento da quantidade de execução diária dos *jobs*, criação de índices de tabelas e otimização de transações.

A Figura 9 apresenta um comparativo dos tempos de execução das 10 principais transações do sistema antes e depois dos trabalhos de performance. Observa-se que houve redução sensível após a otimização dos objetos.

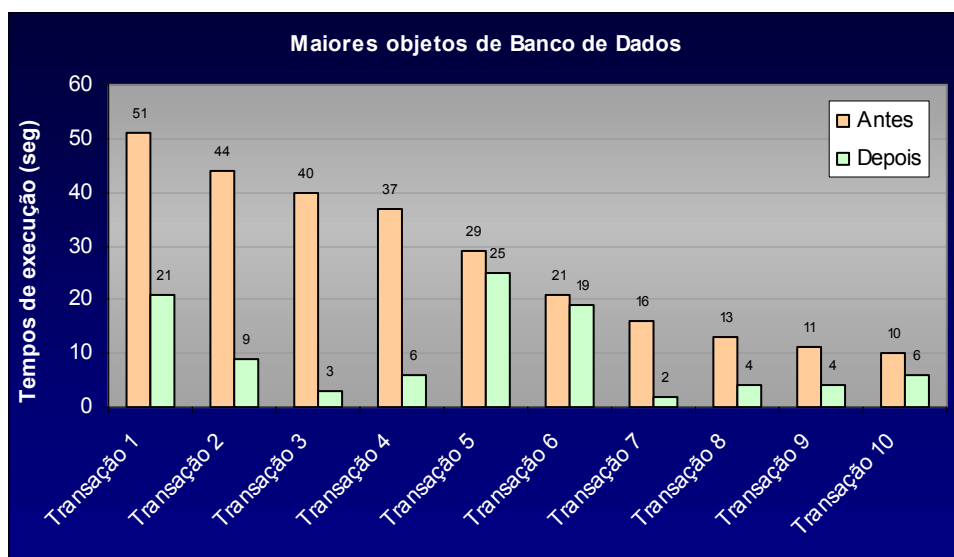


Figura 9. Tempo de execução das principais transações do sistema

DISCUSSÃO

Inicialmente foram realizadas várias alterações na Infra Estrutura de hardware do sistema e em sua Base de Dados substituindo e/ou adicionando recursos mais performáticos aos servidores. Após os investimentos realizados, os sistemas obtiveram melhora significativas, porém, com o passar do tempo, o sistema voltou a apresentar novos sinais de degradação. Uma série de regras de negócio foram redefinidas internamente proporcionando mesmas informações, porém, de forma mais otimizada e veloz, permitindo aos operadores uma maior agilidade e maior confiabilidade aos equipamentos. Em termos de comunicação com sistemas externos, foram redefinidos os padrões de tráfego de mensagens objetivando uma otimização entre os sistemas.

A forma de atuação nas aplicações foi baseada em pesquisas e estudos que permitiram a utilização de ferramentas adequadas e melhores práticas de desenvolvimento proporcionando conseqüentemente, uma atuação direta no foco dos problemas. Surgiu uma maior integração entre equipes de Infra Estrutura, Suporte a Banco de Dados, Manutenção e Melhoria do sistema, permitindo total disseminação de conhecimentos e promovendo um crescimento equilibrado e já enquadrado nos requisitos e práticas de desenvolvimento recomendadas.

Tais otimizações e remodelagens do sistema como um todo, o tornaram mais confiável diminuindo consideravelmente determinadas operações de contingência.

CONCLUSÃO

Inicialmente o trabalho de performance tinha como objetivo principal, diminuir o consumo de recursos dos servidores postergando altos investimentos, que no início pareciam ser inevitáveis. Ao final do trabalho reduziu-se o consumo de recursos computacionais em mais de 50%. Ao final desta primeira fase do trabalho, despertou-se na empresa, a necessidade e conseqüentemente os ganhos obtidos com o controle da qualidade dos sistema, em questões como por exemplo: o código dos programas e não simplesmente a funcionalidade.

REFERÊNCIAS

- 1 VARGAS, F. M., ALMEIDA, A. E. R., CARMO, R.W., MANSUR, R., LIMA, M. J. R., RUBIÃO, L. E. G., Mantendo um sistema MES de missão crítica, **Anais do VII Seminário de Automação de Processos**, Associação Brasileira de Metalurgia e Materiais, Outubro de 2003.
- 2 ENGLAND, P.R. Microsoft SQL Server 2000 - Performance Optimization and Tuning HandBook, Boston: Digital Press, 2001

BIBLIOGRAFIA

- 1 HENDERSON, K. The Gurus Guide to SQL Server Architecture and Internals, London: Addison - Wesley, 2003
- 2 Guia de ajuda on-line para aplicações Microsof (Web Site Microsoft Microsoft Developer Network). Disponível em <http://msdn.microsoft.com>.