

# O Processo de Desenvolvimento de uma Célula Operacional do SIVAM

Ivo Márcio Michalick Vasconcelos/Construtel Tecnologia e Serviços S/A  
Jorgito Matiuzzi Stochero/Cap QEM Exército Brasileiro

## 1. Introdução

A proposta deste trabalho é apresentar um relato da experiência dos autores no desenvolvimento da célula de Vigilância do Espectro Eletromagnético do Subcentro de Coordenação (SCC) dos Centros Regionais de Vigilância (CRV) do Sistema de Vigilância da Amazônia (SIVAM). A ênfase maior é na descrição dos padrões e metodologia de desenvolvimento utilizados num período de aproximadamente três anos, entre 1998 e 2001.

Inicialmente apresentamos uma descrição do SCC/CRV, seguida de uma descrição da célula de Vigilância do Espectro Eletromagnético. Na sequência descrevemos o padrão e metodologia de desenvolvimento utilizados, enfocando a seguir aspectos específicos que achamos importante destacar, como procedimentos de testes e garantia da qualidade, gestão de configuração e uso de inspeções formais.

Concluimos o trabalho buscando destacar a importância da difusão desta experiência entre a comunidade de desenvolvimento de sistemas do país (principal motivação para este trabalho), e esboçando uma proposta para um processo de manutenção e evolução do sistema.

## 2. O SCC/CRV dentro do SIVAM

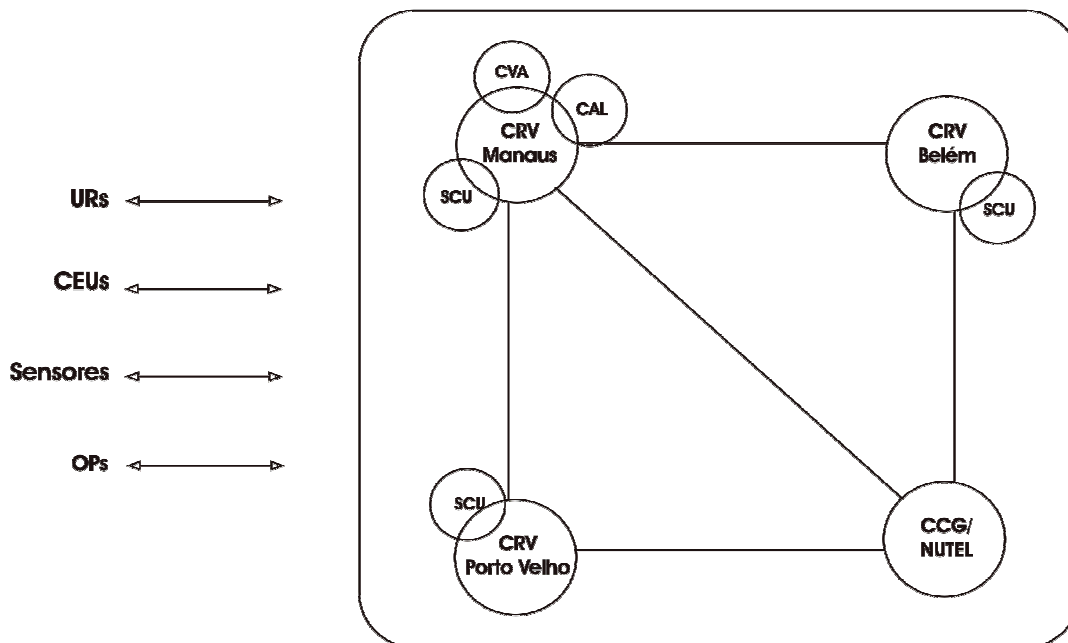
O SCC é o principal sistema do conjunto de sistemas que compõem o SIVAM. O SIVAM é uma extensa rede de coleta e processamento de informações, que recebe dados de Órgãos Parceiros e de sensores espalhados pela Amazônia Legal.

O SCC está presente em cada um dos três CRVs implantados na Amazônia Legal. A localização geográfica dos CRVs nas cidades de Manaus, Belém e Porto Velho forma um triângulo ao qual se conectam todos os demais nós e cujos vértices constituem a principal fonte de irradiação de informação do sistema.

A figura 1 mostra a estrutura dos nós operacionais do SIVAM. Além dos CRVs, temos ainda:

- **CCG/NUTEL:** o NUTEL, ou Núcleo de Telecomunicações, é único nó operacional do SIVAM localizado fora da Amazônia Legal. Ele é composto por um conjunto de equipamentos que permite disponibilizar em Brasília uma porção estratégica do Banco de Dados do sistema, e encontra-se conectado à rede operacional do sistema, estando portanto apto a comunicar-se diretamente com qualquer um dos outros nós operacionais do sistema.

- **CAL:** o CAL (Centro de Apoio Logístico) localiza-se junto ao CRV de Manaus e dá suporte a necessidades logísticas das diversas unidades do SIVAM.
- **CVA:** o CVA (Centro de Vigilância Aérea) concentra o controle de tráfego aéreo da Região Amazônica. Localizado em Manaus, fica sob a responsabilidade da Força Aérea, que é por lei responsável por orientar, coordenar e controlar as atividades de Aviação Civil e prover a segurança da navegação aérea;
- **SCU:** o SCU (Subcentro de Usuários), presente em cada um dos CRVs, permite que usuários de diferentes Órgãos Parceiros – OPs, ou da comunidade em geral, possam usufruir da tecnologia operacional do sistema. Os SCUs proporcionam acesso seletivo a parcelas da Base de Dados do sistema (de acordo com o perfil de usuário definido pela administração do sistema) e aos aplicativos de visualização de produtos que atendam a necessidades específicas dos usuários que forem utilizar estes sub-centros;
- **URs:** Para estender uma parcela das facilidades do SIVAM a postos remotos associados aos Órgãos Parceiros, foram implantados nestes postos os *kits* usuários, compostos de telefone, fax, e em alguns casos, microcomputador, além de uma antena para transmissão de dados via satélite.
- **CEUs:** Os Centros Estaduais de Usuários (CEUs) são centros implantados junto aos governos dos nove Estados que compõem a Amazônia Legal para disponibilizar às secretarias de Estado acesso a uma parcela das informações, ferramentas e funções aplicativos do SIVAM por meio dos Centros Regionais de Vigilância (CRV) associados.
- **Sensores:** constituem a rede de equipamentos de coleta de dados do SIVAM, espalhada por toda a Região Amazônica ou aerotransportados nas aeronaves do sistema (Vigilância e Sensoriamento Remoto).
- **OPs:** OPs são os Órgãos Parceiros, entidades governamentais com algum tipo de ação na Região Amazônica. Podem enviar e receber informações do sistema mediante solicitação ou segundo interfaces definidas no SICD (ver seção 7).



**Figura 1: Nós Operacionais do SIVAM.**

Como centros de vigilância, os CRVs são divididos em células operacionais responsáveis pela vigilância ambiental, vigilância territorial, vigilância meteorológica, vigilância do espectro eletromagnético, pelo planejamento e controle de operações de campo em apoio a estas áreas de vigilância, pelo gerenciamento de informações gerais provenientes de diversas fontes e pelo processamento de imagens. Cada uma destas células corresponde a um subsistema do SCC. Além disto, em apoio ao desenvolvimento destas células e considerando a arquitetura multicamadas adotada para o sistema, subsistemas de suporte também foram desenvolvidos, a saber:

- **Banco de Dados:** cuida de toda infra-estrutura de Banco de Dados, incluindo o *middleware* representado pela camada de persistência que faz a interface entre o Banco de Dados relacional e as aplicações das células desenvolvidas segundo metodologia Orientada a Objetos.
- **Infra-estrutura de Software:** responsável pelo controle de acesso às aplicações e pelo controle de arquivos de log, dentre outras funções.
- **Simulação e Teste:** permite exercitar nas várias células as interfaces com os sensores do sistema. Muito útil para treinamento de novos usuários e atividades de manutenção do sistema.
- **AGDLIC (Air Ground Data Link Interface Controller):** gerencia todas as interfaces de comunicação de células do SCC com as aeronaves de vigilância e sensoriamento remoto do SIVAM.

### 3. Descrição da célula VEsp

A célula de Vigilância do Espectro Eletromagnético (VEsp) tem como Missão rastrear, localizar e identificar emissores clandestinos no espectro eletromagnético de 3 MHz a 2

GHz (Comunicações) e 0,5 a 18 GHz (Não-Comunicações) na área de cobertura do SIVAM.

Para realizar esta Missão, a célula dispõe de toda uma funcionalidade de captura de dados dos sensores disponíveis (estações terrestres e sistemas aeroembarcados nas aeronaves do SIVAM) necessária para receber e processar de forma assíncrona dados de comunicações e não-comunicações transmitidos eletronicamente para a célula, ingerir os dados em um diretório local e notificar os usuários através de uma Lista de Ocorrências.

A VEsp disponibiliza ainda recursos para o usuário armazenar uma seleção de dados recebidos dos sensores no Banco de Dados do SIVAM. Estes dados podem ser relatórios de contato (descrições de emissões detectadas) e arquivos de áudio interceptados. A VEsp também fornece a seus usuários (analistas e supervisores) acesso a funcionalidades que permitem a ingestão de dados de fitas magnéticas e discos óticos e análise (espacial, temporal e estatística) dos relatórios de contato recebidos e armazenados no Banco de Dados do SIVAM. Os relatórios de contato disponibilizam informações sobre emissores específicos cujas transmissões estão sendo monitoradas. Isto abrange transmissões de comunicações na faixa de HF, VHF e UHF e transmissões de fontes de "não-comunicações", tais como radares de navegação operando na faixa de UHF e SHF.

A partir dos dados recebidos dos sensores e fazendo uso das funcionalidades de análise disponíveis, os usuários da célula mantêm um Banco de Dados de Emissores. A partir deste podem ser gerados produtos como Mapas e Relatórios de Inteligência, e pode ser efetuado um segundo nível de análise, visando a identificação de redes clandestinas de comunicação, o que por sua vez permite gerar um novo conjunto de produtos (como Mapas e Diagramas de Redes, ou Relatórios de Inteligência).

Os sensores disponíveis para a VEsp são:

- Estações terrestres fixas REFEC (Rede Fixa de Exploração de Comunicações), localizadas em Manaus/Boa Vista, Belém e Porto Velho;
- Aeronaves de Vigilância e Sensoriamento Remoto do SIVAM.

A VEsp também está apta a receber informações relativas aos Emissores em órgãos governamentais. Estas informações auxiliam os analistas na tarefa de classificar os emissores como *Legais* ou *Ilegais* (considerando o fato de estarem ou não autorizados a emitir), e *Suspeitos* ou *Não-Suspeitos* (considerando o uso que estejam fazendo do meio de comunicação).

A VEsp está apta a disponibilizar parte de seus dados e produtos para outros usuários do SIVAM, bem como para Órgãos Parceiros autorizados, para análises posteriores. Isto é possível através da Biblioteca de Objetos do sistema. Dentro do SIVAM seu maior cliente é a célula de Vigilância Territorial (VTer), que pode usar produtos gerados pela VEsp em seus Casos de Investigação.

## 4. Norma de Desenvolvimento

A maior parte do desenvolvimento do software do SCC/CRV foi realizada nas dependências da empresa norte-americana Raytheon, em Garland, Texas, Estados Unidos. A Raytheon é uma empresa com forte atuação na indústria de Defesa dos Estados Unidos, e é fornecedora frequente de sistemas para o governo daquele país, em especial para as Forças Armadas.

O Departamento de Defesa dos Estados Unidos (DoD) sempre buscou exigir de seus fornecedores a adesão completa a normas, padrões e metodologias de desenvolvimento de sistemas. O modelo SW-CMM (*Capability Maturity Model for Software*), por exemplo, foi criado pelo SEI (Software Engineering Institute) a partir de uma solicitação feita por aquele Departamento.

Por ocasião da definição do contrato para fornecimento do SIVAM, o DoD estava em transição dos padrões DOD-STD-2167A (*Defense System Software Development*, lançado em 1988 e voltado mais especificamente para sistemas do tipo *mission-critical*) e MIL-STD-7935A (mais voltado para desenvolvimento de sistemas de informação) para o padrão MIL-STD-498 (*Software Development and Documentation*, lançado em dezembro de 1994).

Posteriormente (1998), o DoD substituiu a MIL-STD-498 pelo padrão IEEE/EIA 12207. No início dos trabalhos de desenvolvimento do SCC/CRV foi definido que o projeto seria desenvolvido a partir de uma customização de *DIDs* (*Data Item Descriptions* – definição de formato e instruções de preenchimento de um item de projeto a ser gerado) dos padrões DOD-STD-2167A e MIL-STD-498. A tabela 1 enumera as 22 *DIDs* do MIL-STD-498, e no desenvolvimento do SCC/CRV foram usadas e customizadas cerca de 15 *DIDs*.

<b>DID</b>	<b>Produto de Software Resultante</b>
Software Development Plan - SDP	Plano para execução do desenvolvimento do software.
Software Test Plan – STP	Plano para condução dos testes de qualificação.
Software Installation Plan – SIP	Plano para instalação do software nos sítios do usuário.
Software Transition Plan – STrP	Plano para transição do software do ambiente de desenvolvimento para o de produção.
Operational Concept Description – OCD	Conceito operacional do sistema.
System/Subsystem Specification – SSS	Requisitos a serem atendidos pelo sistema.
Software Requirements Specification – SRS	Requisitos a serem atendidos por um subsistema.
Interface Requirements Specification – IRS	Requisitos de interface do sistema.
System/Subsystem Design Description – SSDD	Projeto geral do sistema.
Software Design Description - SDD	Projeto de um subsistema.

Interface Design Description - IDD	Projeto das interfaces do sistema.
Database Design Description – DBDD	Projeto do Banco de Dados do sistema.
Software Test Description – STD	Procedimentos e casos de teste para avaliação do sistema.
Software Test Report – STR	Resultados dos testes de avaliação do sistema.
Software Product Specification – SPS	Software executável, arquivos-fonte e informação a ser usada como suporte.
Software Version Description – SVD	Lista de arquivos entregues e informação associada.
Software User Manual – SUM	Instruções para usuários finais.
Software Input/Output Manual – SIOM	Instruções para usuários dependentes de operadores do software.
Software Center Operator Manual – SCOM	Instruções para operadores de software que apóiam usuários.
Computer Operation Manual – COM	Instruções de operação de um computador.
Computer Programming Manual – CPM	Instruções para programação de um computador.
Firmware Support Manual – FSM	Instruções para programação de <i>firmware</i> de dispositivos.

**Tabela 1: DIDs do padrão MIL-STD-498 (fonte: [STD498], página 18)**

Das DIDs listadas na tabela acima, as seguintes NÃO foram aplicadas ao desenvolvimento do SCC: SIOM, SCOM, COM, CPM e FSM. Além disto, foi gerado para o sistema um documento denominado ICD (*Interface Control Document*), que descreve as interfaces externas do sistema (o IDD descreve as interfaces internas, entre subsistemas). Sobre o ICD, veremos maiores detalhes na seção 7.

A figura 2 mostra uma divisão em grupos das DIDs do MIL-STD-498.

PLANOS	CONCEITO / REQUISITOS	PROJETOS	QUALIFICAÇÃO E TESTES	MANUAIS DE USO E OPERAÇÃO	MANUAL DE SUPORTE	SOFTWARE
SDP	OCD	SSDD	STP	SUM	CPM	SPS
SIP	SSS	SDD	STD	SCOM	FSM	SVD
STRP	SRS	DBDD	STR	SIOM		
	IRS	IDD		COM		

**Figura 2: Agrupamento das DIDs do padrão MIL-STD-498.**

## 5. Metodologia de Desenvolvimento

A célula VEsp, assim como as demais células operacionais do SCC/CRV, foi desenvolvida segundo uma arquitetura cliente-servidor em três camadas, segundo uma metodologia Orientada a Objetos (OO) baseada na linguagem UML. Os servidores de aplicação da célula foram desenvolvidos em C++, e os clientes foram desenvolvidos em Java. O sistema operacional utilizado foi o HP/UX 11, e a interface entre os módulos foi na sua maioria implementada via CORBA (*Common Object Request Broker Architecture*), um padrão para gerenciamento de objetos, onde os componentes são objetos que podem se comunicar entre si atravessando limites tais como redes de comunicação e sistemas operacionais e linguagens de programação distintos. [Baker97]

Os seguintes artefatos foram gerados durante o processo de desenvolvimento da célula:

- Casos de Uso e seus respectivos Diagramas de Sequência;
- Diagramas e Descrições de Pacotes;
- Diagramas e Descrições de Classes;
- Diagramas de Colaboração;
- Fluxos de Interfaces de Usuário;
- Diagrama de Componentes;
- Diagrama de Implementação (*Deployment Diagram*).

Todos estes artefatos, bem como todas as decisões de projeto tomadas ao longo do desenvolvimento e a matriz de rastreabilidade de requisitos (que mapeia todos os requisitos da célula nos respectivos diagramas de sequência que os implementam) foram documentados em um documento denominado *DSDD (Detailed Software Design Document)*, cujo modelo foi customizado a partir da DID DI-IPSC-81435, do padrão MIL-STD-498 ([**DID-SDD**]). Foi elaborado um DSDD para cada um dos subsistemas do SIVAM.

Para geração dos artefatos citados acima e em apoio às várias atividades de desenvolvimento do projeto, um conjunto de ferramentas foi utilizado. Destas, destacamos as seguintes:

- **RTM** - <http://www.chipware.com> : gerenciamento e rastreabilidade de requisitos;
- **Object Team/Cool:Jex/Telelogic Tau** - <http://www.telelogic.com/products/tau/uml/index.cfm> : geração dos artefatos UML;
- **Meeting Maker** - <http://www.meetingmaker.com/solutions/meetingmaker> : ferramenta de apoio à organização de reuniões de trabalho, que tinham que ser convocadas obrigatoriamente através desta ferramenta;
- **Clearcase** - <http://www.rational.com/products/clearcase/index.jsp> : ferramenta principal usada nas atividades de Gestão de Configuração e apoio a atividades de desenvolvimento distribuído entre várias equipes;

- **ClearDDTS** - [http://www.rational.com/products/clear\\_ddts](http://www.rational.com/products/clear_ddts) : apoio às atividades de Gestão de Configuração, com foco em controle de alterações de software;
- **MS Project**: única ferramenta utilizada fora do ambiente HP/UX, como apoio às atividades de planejamento e acompanhamento de projeto.

## 6. Testes e Garantia da Qualidade

Um ponto extremamente importante definido no início dos trabalhos de desenvolvimento foi de que a garantia e controle da qualidade dos produtos gerados seria feita por uma equipe separada e independente da equipe de desenvolvimento. Esta equipe foi denominada SQA Team (*Software Quality Assurance Team*), e contava com profissionais brasileiros e americanos das empresas desenvolvedoras do sistema. Membros da equipe de SQA tinham presença obrigatória em todas as inspeções e testes formais realizados.

No nível de produtos de software, foram realizados testes ao nível de unidade (subconjunto de funcionalidades do subsistema que implementa uma parte dos requisitos de software definidos e pode ser testado de forma isolada), FQT (*Formal Qualification Test*), onde a célula era testada de forma isolada em um ambiente controlado de testes, FAT (*Factory Acceptance Test*), teste do subsistema (com todos os subsistemas integrados) em fábrica e SAT (*Site Acceptance Test*), teste do subsistema em campo (o do CRV de Manaus foi concluído com sucesso em julho de 2002).

## 7. Gestão de Configuração - o exemplo do ICD/ICWG/CCB

Devido à multiplicidade de sensores interligados ao SCC/CRV, este sistema possui um grande número de interfaces externas a serem definidas, documentadas, mantidas e publicadas. Por esta razão, foi definido que o sistema teria um documento de apoio chamado ICD (*Interface Control Document*), onde todas as interfaces externas seriam definidas com grau de detalhe suficiente para que as partes envolvidas pudessem implementar suas respectivas interfaces.

É importante ressaltarmos que quando falamos de interfaces externas nos referimos a interfaces entre subsistemas do SCC/CRV e entidades fora do sistema, que podem ser sensores ou Órgãos Parceiros. As interfaces internas ao sistema (entre subsistemas) foram capturadas e documentadas nos documentos IRS (*Interface Requirements Specification*) e IDD (*Interface Design Document*).

Pelo grande número de subfornecedores envolvidos e pelo potencial impacto que pode ser causado por erros e alterações no ICD, este documento foi submetido a um nível adicional de gestão de configuração: além de se obrigar suas alterações a serem aprovadas por um CCB (*Configuration Control Board*), foi criado ainda um ICWG (*Interface Control Working Group*). Trata-se de um grupo formado por representantes de todos os subfornecedores envolvidos, responsável em conjunto pela elaboração e manutenção do documento durante o período de desenvolvimento. O ICWG era responsável pela análise técnica de propostas de mudanças, inserções e remoções de interfaces capturadas no ICD. Ele se reunia sob demanda, ou periodicamente, nas etapas



iniciais do desenvolvimento. Todas as recomendações feitas pelo ICWG para o ICD deviam posteriormente ser aprovadas pelo CCB para que pudessem ser efetivadas em uma futura versão do documento.

A linguagem IDL (*Interface Definition Language*) é usada para definir interfaces CORBA de forma independente das linguagens de programação utilizadas na implementação das interfaces. Sua sintaxe é similar à das linguagens Java e C++, mas é bem mais simples, por prescindir de construções como variáveis, estruturas de controle ou comandos. Como a maioria das interfaces do sistema foi implementada em CORBA, o ICD incluiu todos os arquivos IDL gerados, sendo portanto um documento de extrema importância para qualquer processo de manutenção que mostrar-se necessário em alguma das interfaces externas do sistema. Arquivos IDL são arquivos-fonte escritos na linguagem IDL de um servidor CORBA específico (no caso do SCC/CRV foi usado o IONA Orbix - [Baker97]).

## 8. Uso de Inspeções Formais

Inspeções Formais são revisões técnicas de produtos gerados no ciclo de vida do desenvolvimento de software conduzidas com o propósito de descobrir e eliminar defeitos nestes produtos. Podem ser aplicadas a qualquer produto ou parcela de produto gerado no processo de desenvolvimento de software, incluindo requisitos, projeto e código. Incorporadas ao processo de desenvolvimento de produtos de software, são realizadas geralmente nos estágios iniciais do ciclo de vida do desenvolvimento do produto.

O conceito de uso de Inspeções Formais como mecanismo de melhoria do processo de desenvolvimento de software foi introduzido por Michael Fagan na IBM em 1976 ([Fagan76]), buscando ganhos de qualidade e aumento da produtividade dos programadores. O processo de Inspeção Formal envolve cinco elementos, a saber:

- Passos de inspeção bem definidos;
- Papéis bem definidos para os participantes do processo (moderador, secretário, leitor, autor e inspetor);
- Coleta formal de dados do processo de inspeção e do produto inspecionado;
- O produto a ser inspecionado;
- Uma infra-estrutura de suporte.

Inspeções Formais foram extensivamente utilizadas ao longo de todo o processo de desenvolvimento do sistema. Tivemos a oportunidade, dentro do Plano de Treinamento definido para as equipes de desenvolvimento, de realizar treinamentos como Inspetor e Moderador de Inspeções Formais. Como tal, participamos de um grande número de inspeções formais em vários dos papéis previstos (Moderador, Inspetor, Autor e Secretário). Com isto pudemos atestar a extrema utilidade da adoção de tal prática (ou alguma prática similar, porém menos formal, como as *peer reviews*) no desenvolvimento de sistemas de informação, em especial por três razões:

1. A razoável quantidade de defeitos encontrada ANTES da liberação do produto inspecionado, reduzindo bastante o retrabalho.

2. No caso de código, garante-se a adesão às normas de codificação definidas, o que por sua vez facilita a manutenção futura do sistema.
3. A quantidade de defeitos é registrada por tipo de defeito, fase do projeto e tipo de artefato/produto sendo inspecionado. Com isto, avaliações e estimativas futuras são realizadas com base no histórico das medições, garantindo uma melhor precisão.

Sempre que possível, as inspeções contaram com representantes do cliente final (facilitando o acompanhamento do desenvolvimento), e a participação de um representante do SQA (time de garantia da qualidade) era obrigatória. No caso das inspeções de código, os formulários de conclusão bem sucedida dos testes unitários eram pré-condição para a realização da inspeção.

## 9. Conclusão

O principal objetivo deste trabalho foi buscar difundir e documentar a experiência dos autores no desenvolvimento do SIVAM entre a comunidade acadêmica, órgãos governamentais e empresas de tecnologia do país. Além disso, entendemos que o processo de manutenção e evolução do sistema, fundamental para que este se mantenha eficaz e atendendo aos princípios básicos para os quais foi criado, deve ser facilitado pelos padrões e metodologias utilizados e pela entrega ao cliente final, no caso o Governo brasileiro, por meio do CENSIPAM (Centro Gestor e Operacional do Sistema de Proteção da Amazônia, atual responsável pelo sistema), do código-fonte e ambiente de desenvolvimento.

Com este trabalho acreditamos ter mostrado que a evolução do SIVAM pode seguir um mecanismo livre que mantenha o sistema independente de um grupo de fornecedores específicos, podendo se beneficiar do conhecimento e experiência de atores nacionais capacitados trabalhando a partir das fundações criadas no desenvolvimento do sistema. Entendemos ainda que esta evolução pode e deve se beneficiar muito das experiências adquiridas pelo CENSIPAM a partir do início de operação do sistema.

## 10. Bibliografia

- [Baker97] Baker, Seán. **“CORBA Distributed Objects Using Orbix”**. ACM Press/Addison-Wesley, 1997.
- [DID-SDD] V/A. **“Data Item Description – Software Design Description - DI-IPSC-81435”**. DoD, Dezembro de 1994. (disponível em <http://www2.umassd.edu/SWPI/DOD/MIL-STD-498/SDD-DID.PDF> )
- [Fagan76] Fagan, M. **“Design and Code Inspections to Reduce Errors in Program Development.”** IBM Systems Journal 15, 3 (1976): 182-211.
- [STD498] V/A. **“MIL-STD-498 Overview and Tailoring Guidebook”**. JLC/JPCC on Computers Resources Management, Janeiro de 1996. (disponível em <http://www2.umassd.edu/SWPI/DOD/MIL-STD-498/498GBOT.PDF> )