

Um Meta-Modelo Orientado a Objetos como Ontologia para a Descrição de Domínios e Problemas de Planejamento

Rodrigo Rocha Silva¹, Mauricio Gonçalves Vieira Ferreira², Nandamudi Lankalapalli Vijaykumar¹

¹Laboratório Associado de Computação e Matemática Aplicada / ²Centro de Controle de Satélites– Instituto Nacional de Pesquisas Espaciais (INPE) – São José dos Campos – SP – Brasil

rrochas@gmail.com, mauricio@ccs.inpe.br, vijay@lac.inpe.br

Abstract. *One of the major difficulties in development of planners using Artificial Intelligence Planning and Scheduling's technique (AIPS) is the correct description of domains and problems. Use of AIPS has been increasing for planning the activities of satellites. The objective of this paper is to define a Meta-Model for Object Oriented like Ontology for Description of Domains and Planning Problems. This paper discusses the use of ontologies and knowledge management to support the modeling domain area to control satellites. A tool will be developed to support the description of domain and problems within space applications. Axiom will be the basis to both support and define concepts and relationships using graphs as well as allowing generation of framework objects from ontologies.*

Resumo. *Uma das grandes dificuldades no desenvolvimento de planejadores utilizando a técnica de Planejamento e Escalonamento, da área de Inteligência Artificial (AIPS), é a correta descrição de domínios e problemas. A técnica de AIPS vem sendo cada vez mais utilizada para o planejamento das atividades de satélites. Este trabalho vem com o proposito de definir um Meta-Modelo Orientado a Objetos como Ontologia para a Descrição de Domínios e Problemas de Planejamento. Este artigo discute o uso de ontologias e gerência de conhecimento para apoiar a modelagem de domínio da área de controle de satélites apresentando uma ferramenta de suporte à descrição de domínios e problemas da área espacial. Apoiando a definição de conceitos e relações utilizando representações gráficas, além de possibilitar a geração de frameworks de objetos a partir das ontologias, e também suas definições, propriedades e restrições, descritas na forma de axioma.*

1. Introdução

Com a grande utilização da AIPS a Engenharia de Conhecimento passou a ter uma grande importância na concepção dos problemas da área espacial. Com isso para o melhor entendimento e classificação dos domínios e problemas de planejamento; a especificação, modelagem e análise dos domínios se tornam essenciais.

Os três maiores obstáculos para a área de planejamento e escalonamento da área de inteligência artificial (AIPS) em problemas reais são: (i) especificação dos requisitos, (ii) modelagem (geralmente na modelagem de ações) e (iii) análise de domínios, já que estes processos são considerado um grande “gargalo” no desenvolvimento de aplicações reais. [5]

Técnicas como: léxicos, *thesaurus*, dicionários, taxonomias e ontologias tem o proposito de facilitar a representação e comunicação conhecimento entre especialistas de um domínio. O léxico representa os termos (ou descritores) que compõem uma linguagem, que uma equipe de especialistas na área desenvolve, dicionário lista o significado dos termos, organizado em ordem alfabética. Um *thesaurus* (tesauro) é composto por uma lista de termos dentro de um domínio de conhecimento e o relacionamento entre eles, organizado conforme o significado dos termos. Diferentemente dos dicionários, os *thesaurus* não inclui definições detalhadas, já que seu objetivo é auxiliar o usuário na seleção de termos exatos apresentando diferenças mínimas entre estes.

Taxonomias, baseia-se em uma lista estruturada de categorias de assuntos e as relações hierárquicas entre os tópicos, sem incluir a definição dos mesmos. A Ontologia é uma explícita especificação de uma "conceitualização" Thomas R. Gruber[1], isto é, consiste de uma especificação de objetos, conceitos e outras entidades que são assumidas como existentes, além de relações entre conceitos e restrições expressas através de axiomas[2]. O uso de ontologia para representar os conceitos do domínio permite que estes possam ser reusados em várias aplicações.

Ontologias possuem fortes relações entre os termos diferenciando-se assim dos *thesaurus* também estando em linguagem que pode ser interpretada por máquinas. Assim uma ontologia é formada pelo menos por: conceitos, relações, atributos, instâncias e axiomas.

Segundo Almeida e Bax[3], uma ontologia pode ser definida como as regras que balanceia a fusão entre conceitos (organizados em uma taxonomia) e relações (i. é, o tipo de interação entre os conceitos) em um domínio do conhecimento e permite aos usuários realizarem consultas a instâncias (elementos específicos, ou seja, os próprios dados) usando conceitos definidos pela ontologia. Representações como as supracitadas são freqüentemente utilizadas com vários objetivos, entre eles, a troca de informações e conhecimentos entre os profissionais de forma clara e sem ambigüidades.

Devido aos fatos colocados, e a grande necessidade de ferramentas que dêem auxilio durante o ciclo de vida de projeto de domínios de planejamento, o presente trabalho pretende investigar os problemas envolvidos na interação de informação pertinente à gestão do conhecimento do planejamento e escalonamento de atividades principalmente no âmbito da área espacial, oferecendo subsídios para a modelagem de domínios e problemas de planejamento através de um meta-modelo orientado a objetos para a descrição de domínios e problemas de AIPS dispondo também de um vocabulário base para a descrição de problemas e domínios desta área, visando auxiliar o mapeamento do conhecimento necessário.

Isso é proposto, considerando que a falta de uma padronização na representação do conhecimento sobre planejamento de atividades para o controle de satélites, dificulta a compreensão por parte de todos os atores envolvidos no processo.

2. Conceituação sobre Meta-Modelo

Em "Os investigadores", Boorstin cita que "Einstein via o pequeno e o grande, o atômico e o cósmico, como um enigma, a fim de descobrir o todo, explicado por leis e pela razão, inspirado pelo que ele chamava de seu sentimento religioso cósmico. O indivíduo percebe a futilidade dos desejos e anseios humanos e a maravilha e perfeita ordem que se revelam tanto na natureza como no mundo do pensamento", (...)" [4].

Um meta-modelo, busca esta idéia, da descoberta do todo, e segundo [5] também pode ser visto como uma representação dos tipos de entidades que podem existir em um modelo, suas relações e restrições de aplicações.

Então baseado-se nas definições de Guarino [6] sobre ontologia na ciência da computação, onde é estabelecido que: *“uma ontologia é uma teoria lógica para relacionar o significado pretendido de um vocabulário formal, isto é, seu comprometimento com uma conceitualização particular do mundo”*; na definição de Fonseca [7] para componentes em que *“ontologia é uma teoria de especificação de vocabulário relativo a um determinado domínio definindo entidades, classes, funções e relacionamentos entre componentes”*; propomos um meta-modelo orientado a objetos como modelo do domínio da AIPS, pois com esta técnica permite alcançar o frisado por Gómez-Pérez [8], que para a construção de uma ontologia, cinco tipos de componentes têm que ser levados em conta: conceitos (termos ou classes, e seus domínios de valores), relacionamentos, funções (relações especiais onde o n-ésimo elemento da relação é único para os n-1 elementos precedentes), axiomas (modelam sentenças que são sempre verdadeiras) e instâncias.

O uso combinado de ontologias e modelos de domínios já foi abordado em outros trabalhos como [9], [10] e [11] dentre outros, porém não foi até o momento proposto como modelagem do domínio da AIPS um meta-modelo orientado a objetos como ontologia para esta área, que seja genérico o suficiente para a descrição de domínio relacionado a missões espaciais e ser uma alternativa às atuais linguagens utilizadas em representação de problemas de planejamento, isso é proposto baseando-se em características como: flexibilidade, baixo acoplamento, alta coesão e reutilização de componentes.

Assim este trabalho supri também o primeiro nível de divergência em ontologia descrito por Klein [12], que é o da linguagem de descrição ou meta-modelo. Onde as divergências inclusas neste nível são as diferenças sintáticas, diferenças no significado das primitivas nas diferentes linguagens e diferenças na expressividade das linguagens. Pois o modelo proposto é baseado em objetos inerentes ao escopo da AIPS não sendo acoplado a nenhuma primitiva pré-estabelecida.

O presente trabalho foi apresentado no IV Fórum de Inteligência Artificial da Região Sul [13], onde foi descrito um meta-modelo de domínio de AIPS que consiste em classes que abstraem a base do conhecimento para descrição de domínios e problemas de planejamento denominado KPlanOO, na Figura 1 observamos as abstrações identificadas e suportadas pelo modelo.

Este trabalho tem forte relação com as áreas de reuso de software e inteligência artificial, já que modelos de domínio são usados, principalmente, pela comunidade de reuso de software [14], e ontologias são aplicadas, principalmente, pela comunidade de inteligência artificial na perspectiva de modelagem de conhecimento.

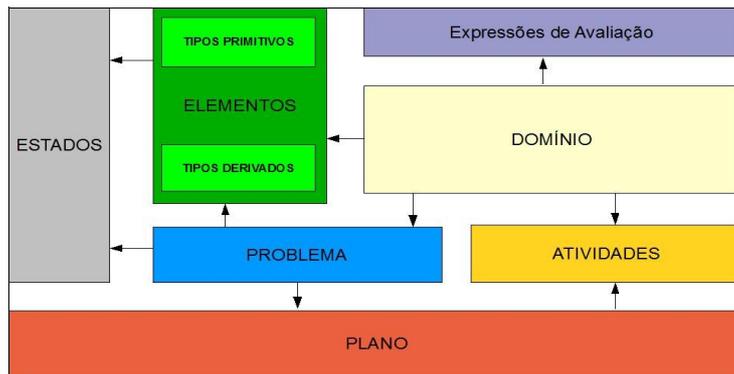


Figura 1. Abstrações identificadas para a Descrição de Domínios e Problemas de Planejamento suportadas pelo KPlanOO.

Porém, com o amadurecimento, passarão a contribuir para outras áreas do conhecimento, a gestão do conhecimento é uma dessas áreas [9], assim este trabalho utiliza ontologias e modelos de domínios como ferramentas de gestão do conhecimento para a AIPS.

3. Utilizando Ontologias e Modelos de Domínio como Ferramentas de Gestão do Conhecimento para AIPS

Fouro[15] compara Modelagem de Domínio e Ontologias e chega às seguintes considerações: Modelos de Domínio e Ontologias são formas de estruturação e organização de conhecimento visando possibilitar a sua reutilização. São o resultado da busca por uma prática de reuso de software sistemática, formal e efetiva.

As duas técnicas visam possibilitar o reuso de informações sobre um domínio armazenando de informações sobre este, podendo estas também contribuírem na tarefa de reutilização durante o desenvolvimento de uma nova aplicação daquele domínio. A principal diferença entre eles, é que a ontologia não assume a pré-existência de nenhum sistema, no domínio a ser modelado. Desta forma, o nível de abstração das ontologias é mais alto que o dos modelos de domínio.

Desde sua origem, o Planejamento é, baseado em conhecimento, isto é, o processo de planejamento envolve a manipulação de conhecimento de fenômenos complexos, tais como as ações [16], no processo de planejamento, o conhecimento (Objetivos, Estado Inicial e Ações) é o ponto de partida de todo o processo.

A construção do KPlanOO apesar de ter sua motivação maior no domínio da área espacial, mais precisamente no que se refere ao controle de atividades de satélites, baseou-se nos tipos de ontologia, apresentado em [17], chegando a um meta-modelo genérico capaz de modelar todo o conhecimento para o processo de planejamento, de diversos domínios e problemas.

O KplanOO baseou-se nos seguintes tipos de ontologias:

- Meta-ontologias, também chamadas de Ontologias Genéricas ou Ontologias Fundamentais, que são reutilizáveis (ou aplicáveis) em diferentes domínios.
- Ontologias de domínio são reutilizáveis em um dado domínio provendo vocabulários sobre os conceitos dentro de um domínio e seus relacionamentos, sobre as atividades que envolvem este domínio e sobre as teorias e princípios elementares que governam aquele domínio.
- Ontologias de aplicações que contêm o conhecimento necessário para modelar situações específicas de uma tarefa em um domínio particular.

Um projeto de ontologia tem abordagens baseadas em projetos de orientação a objetos [18] [19]. Porém, o desenvolvimento de ontologias se difere da modelagem de classes e relacionamentos numa modelagem orientada a objetos (representada em UML). Noy & McGuinness [20] esclarecem que a modelagem orientada a objetos se foca principalmente sobre métodos em classes — um analista toma decisões baseado em propriedades operacionais de uma classe, enquanto um engenheiro de ontologias toma decisões baseado em propriedades estruturais de uma classe, explicitando suas relações e formalizando os conceitos. Como resultado, uma estrutura de classe e relações em uma ontologia pode ser diferente de uma estrutura em uma modelagem orientada a objetos para um domínio similar [19].

Seguindo a afirmação acima, o KPlanOO apesar de ter utilizado a OO como metodologia, foi desenvolvido baseando-se na generalização das propriedades estruturais do domínio da AIPS, chegando assim um modelo estruturado de classes componentizadas que se relacionam de forma que estabeleçam conceitos e métricas fortes para a modelagem do domínio em questão.

4. O KPlanOO como Ontologia para a Descrição de Domínios e Problemas de Planejamento

A versão atual do meta-modelo é uma evolução do apresentado em [13], e sofreu algumas alterações, hoje é composto por 18 classes e 2 Interfaces e suas respectivas relações. A estrutura estática do KPlanOO, encontra-se no diagrama de classes apresentado na Figura 2.

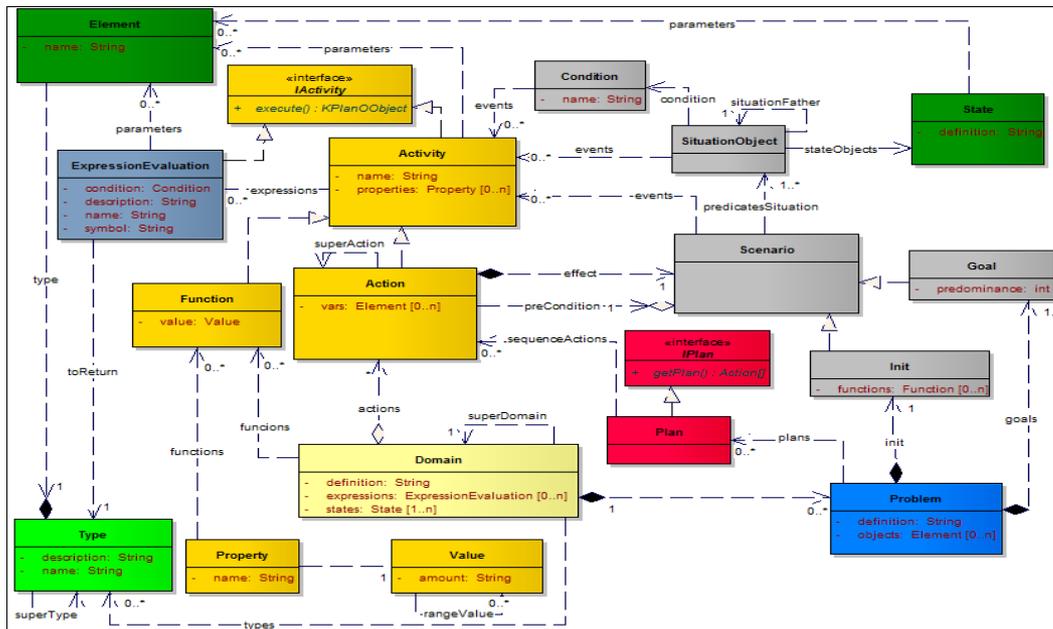


Figura 2: Estrutura estática do KplanOO

Segue a descrição de cada classe:

- A classe **Type** é a abstração dos tipos que podem ser criados em um domínio, ex: Inteiro, Satélite, Região, etc...
- **Value** representa um valor que será configurado no domínio tanto para uma função como para um elemento, a classe **Value** tem um auto-relacionamento que possibilita a configuração de um range de valores.

- **Element** abstrai os objetos que irão fazer parte do problema, também abstrai os parâmetros dos predicados e funções e variáveis necessárias na descrição. **Element** tem um tipo e pode ter valores (Value).
- **ExpressionEvaluation** como o próprio nome já diz, representam expressões de avaliação, estas expressões serão aplicadas a predicados e também poderão ter variáveis aplicadas a avaliação, situação essa muito comum em descrições do domínio de satélite. Exemplo de expressões que poderão ser criadas: > (maior), Not (Negação), And, Or, etc...
- **Activity** abstração de atividades que pode ser utilizadas no processo de planejamento, possui uma lista de elementos que representa os parâmetros, também existe uma lista de propriedades e expressões de avaliação.
- **Action** abstração das ações do domínio, esta contempla a especificação de "SuperAções" através de um auto relacionamento, além da lista de elementos herdada que representa os parâmetros, é definida uma outra lista representando variáveis de cada ação específica. Cada ação possui um efeito e uma pré-condição que são definidos através de relações com a entidade State.
- **Condition** abstração das condições que um objeto em um determinado estado se encontra, cada condição pode ter eventos associados.
- **State** Entidade que representa estados definidos no domínio que serão aplicados aos objetos de problemas. Pode-se fazer uma analogia deste conceito com o conceito de predicados em Linguagens de Modelagem de Domínios como a PDDL.
- **SituationObject** Entidade que representa as situações dos objetos que serão manipulados no planejamento. As situações é o conjunto formado pelo estado dos objetos pelas condições atreladas a este estado. Também possível ter uma super situação e eventos associados a cada situação, estes eventos podem ser executados dada a situação conforme o planejador for elaborado.
- **Scenario** entidade que representa um cenário composto pelos estados dos objetos definidos no problema em questão, sendo estes de tipos declarados no domínio. Possui uma coleção de atividades que representa eventos a serem executados, dado um cenário específico.
- A **Function** abstrai possíveis funções que possam fazer parte de um domínio, possui uma lista de propriedades herdada que possibilitam a configuração de valores iniciais ou uma gama de valores, também possui uma lista de elementos que é herdada representando os parâmetros destas funções. Poderá inserido regras utilizando-se de um conjunto de StateObject.
- **Init** Representação do estado inicial de um problema, possui uma lista de Function's que contempla funções a serem executadas no início do planejamento, o estado representado é a composição de expressões de avaliação sobre um estado.
- **Goal** responsável por representar a meta a ser atingida por um planejador, através de estados para os objetos declarados em problema. Possui um atributo "predominance" que define a prioridade da meta em relação as outras definidas para o mesmo problema.
- **Problem** tem por finalidade representar a descrição dos problemas para um dado domínio, esta é composta por uma instancia da classe **Init**, uma instancia da classe **Goal** e também uma lista de elementos que representará os objetos a serem utilizados pelo planejador.

- **Domain** representa a descrição do domínio de planejamento, composto por uma lista de ações, possui uma lista de estados prováveis para objetos de problemas com tipos definidos. É também possível definir um conjunto de Function's, e o domínio pode ter várias instancias de problema.
- **IPlan** interface para padronizar a aquisição dos planos gerados pelos planejadores, a classe de implementação direta a está interface é a **Plan** que representa os planos gerados.
- **Property** entidade que representa propriedades de uma atividade, essa propriedade pode ter seu valor definido pela execução de uma função.

Visando o facilitar o uso de padrões de projetos na construção de planejadores OO que utilizem o KplanOO como ferramenta para descrição de domínios e problemas de planejamento, todas as classes conceituais do modelo são sub-classes direta ou indiretamente de uma classe abstrata do modelo nomeada como **KPlanOObject**, na Figura 3 é possível ver a estrutura de herança existente no modelo.

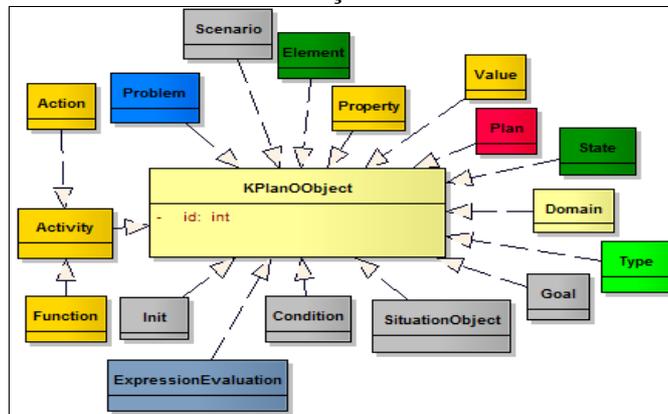


Figura 3: Estrutura de herança presente no KplanOO

A interface **Activity** se caracteriza por seguir a especificação do padrão de projeto “command” que tem como característica encapsular uma requisição como objeto, para clientes parametrizarem diferentes requisições, filas, e suportar operações reversíveis; assim pode haver diferentes implementações de atividades que serão manipuladas da maneira mais conveniente, características como essa tornam o KPlannOO altamente extensível e possibilita seu reuso.

4.1. Validando a estrutura do KPlanOO como Ontologia através da ferramenta Protégé

Foi utilizado o editor de ontologias Protégé, para modelar o KPlanOO para assim validar sua estrutura quanto às métricas para a criação de uma ontologia, na Figura 4, observa-se o KPlannOO em uma visão de árvores disponibilizada pelo plugin “Jambalaya” da plataforma Protégé.

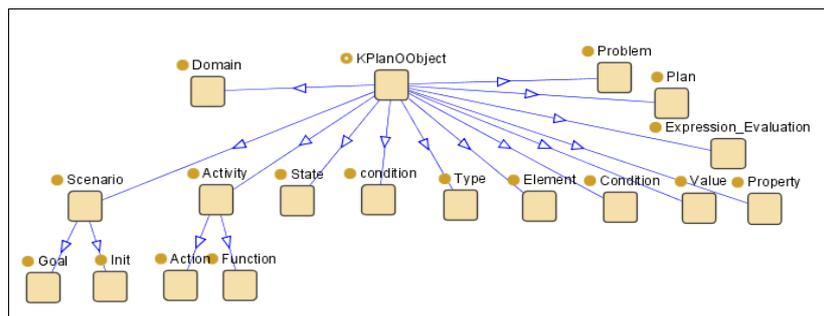


Figura 4: Estrutura do KPanOO modelada na ferramenta Protégé

O KPlanOO foi modelado utilizando a o recurso Protege-Frames da plataforma Protégé. Assim o desenvolvimento se seguiu por interface de usuários e servidor de conhecimento que suportando a construção e desenvolvendo de quadros baseados no domínio da AIPS, adaptando a forma de entrada dos dados e entrando com dados de instanciação, estes dados utilizados é a descrição de um domínio de controle de satélites do INPE descrito em PDDL por Cardoso[21], a Figura 5 apresenta a interface de definição da instancia do domínio.

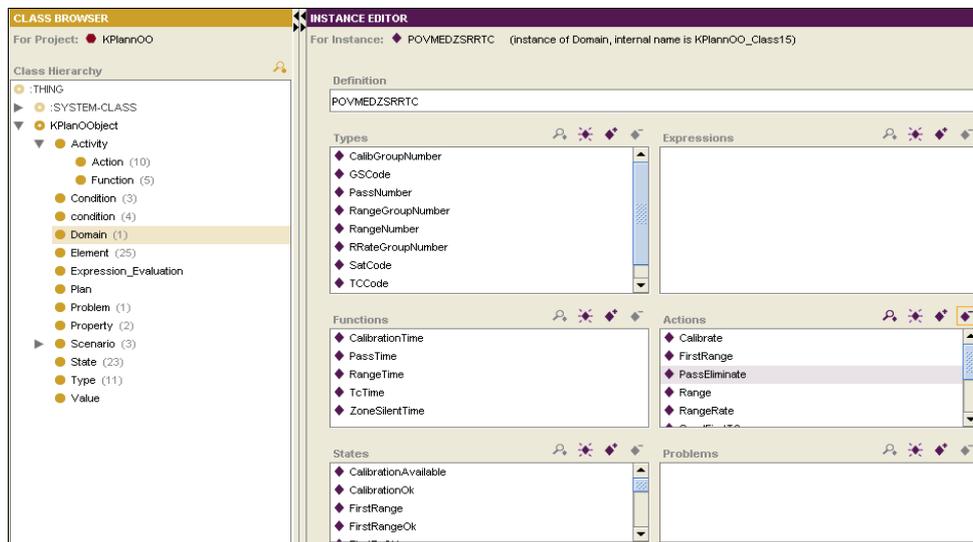


Figura 5: Definição de domínio para de controle de satélites do INPE baseado no KplanOO utilizando a ferramenta Protégé

5. Conclusão

A abordagem aqui descrita junto com outros trabalhos, faz parte de uma iniciativa do centro de controle de satélites do INPE de automatizar as operações de seus satélites.

O modelo proposto neste trabalho traz os conceitos de planejamento e escalonamento da área de IA para os sistemas de operação de satélites, juntamente com conceitos de Ontologia e Orientação a Objetos. Além de auxiliar a geração de planos para operações espaciais, também contribuirá na representação do conhecimento do domínio de planejamento, facilitando assim o correto mapeamento dos dados de entrada necessários para a geração da definição do domínio.

Esta abordagem abre outras possibilidades no desenvolvimento de aplicações de planejadores facilitando também o reuso e uma maior adesão ao AIPS e a Orientação a Objetos como técnica em missões espaciais. Espera-se como contribuição a facilidade para a a geração de conhecimento para o uso da AIPS, novas possibilidades de desenvolvimento de planejadores que usufruam de flexibilidade, extensibilidade, reuso, padronização de projetos e reuso .

6. Referências

- [1] Gruber, T.R. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. International Journal of Human-Computer Studies (IJHCS),43(5/6),907 928,1995.
- [2] Zlot, F., Oliveira, K.M., Rocha., A.R. Modeling Task Knowledge to Support Software Development.SEKE'02, Itália, 2002.

- [3] ALMEIDA, M. B.; BAX, M. P. Uma visão geral sobre ontologias: pesquisa sobre definições, tipos, aplicações, métodos de avaliação e de construção. *Ciência da Informação*, Brasília, DF, v. 32, n. 3, p. 7-20, 2003.
- [4] BOORSTIN, D.J. (2003) *Os investigadores – A história da permanente busca do homem para compreender o seu mundo*, Rio de Janeiro, ed: Civilização Brasileira.
- [5] GÓMEZ-SANZ, J.J.. (2001) "Meta-modelos para el desarrollo de sistemas multi-agente," Caepia disponível no site [<http://grasia.fdi.ucm.es/SP/publicaciones/index.html>] acessado em 05/2004.
- [6] Guarino, N. *Formal Ontology and Information Systems*. Proceedings of FOIS'98, Trento, Italy, 6-8 June (1998). Amsterdam, IOS Press, pp. 3—15. (<http://www.ladseb.pd.cnr.it/infor/Ontology/Papers/FOIS98.pdf>)
- [7] Fonseca, F., Egenhofer, M. et al. *Ontologias e Interoperabilidade Semântica entre SIGs*. In: II Workshop Brasileiro em Geoinformática - GeoInfo2000, Proceedings. São Paulo.
- [8] Gómez-Pérez, A. e Benjamins, V.R. *Overview of knowledge sharing and reuse components: Ontologies and problem-solving methods*. In: International Joint Conference on Artificial Intelligence (IJCAI-99), Workshop on Ontologies and Problem-Solving Methods (KRR5). Stockholm, Sweden, (1999).
- [9] MANGAN, M. A. S. ; MURTA, L. G. P. ; SOUZA, J. M. ; WERNER, C. M. L. . Modelos de Domínio e Ontologias: uma comparação através de um estudo de caso prático em hidrologia. In: International Symposium on Knowledge Management/Document Management (ISKM/DM'2001), 2001, Curitiba. IV International Symposium on Knowledge Management/Document Management (ISKM/DM'2001), 2001. p. 149-172.
- [10] Faria, C. G., "Uma técnica para a aquisição e construção de modelos de domínio e usuários baseados em ontologias para a engenharia de domínio multiagente", Dissertação (Mestrado) – Centro Universitário do Maranhão, 2000.
- [11] Guizzard G., "Uma Abordagem Metodológica de Desenvolvimento para e com Reuso, Baseada em Ontologias Formais de Domínio". Dissertação de Mestrado. Universidade Federal do Espírito Santo, 2000.
- [12] KLEIN, M. Combining and Relating Ontologies: An Analysis of Problems and Solutions. In: *Workshop on Ontologies and Information Sharing at the 17th International Joint Conference on Artificial Intelligence*, 2001, Seattle, USA, p. 53-62.
- [13] Silva, R. R.; Ferreira, M. G. V.; Vijaykumar, N. L. *Descrivendo domínios e problemas de planejamento através de um modelo orientado a objetos*. In: IV Fórum de Inteligência Artificial da Região Sul 2008, Canoas, RS, Brasil.
- [14] ARANGO, G., 1994, "Software Reusability - Domain analysis methods", *Ellis HorWood*, pp. 17-49.
- [15] FOURO, A. M. M., WERNER, C. M. L., 2001, "Modelos de domínios ou Ontologias?", *RTInfo – Revista Tecnologia da Informação*.
- [16] T. S. Vaquero. *itSIMPLE: Ambiente integrado de Modelagem e Análise de Domínios de Planejamento Automático*. Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo, São Paulo, 2007.
- [17] SOUZA, J. F. ; SIQUEIRA, S. W. M. ; AZEVEDO, L. G. ; BAIAO, F. ; LOPES, M. ; SANTORO, F. M. ; CAPPELLI, C. ; NUNES, V. ; MAGDALENO, A. *Gestão de Ontologias*. Rio de Janeiro: Departamento de Informática Aplicada da UNIRIO, 2008 (Relatório Técnico).
- [18] RUMBAUGH, J.R., BLAHA, M.R., LORENSEN, W., *et al.* Object-oriented modeling and design. Upper Saddle River: Prentice-Hall, 1991.
- [19] BOOCH, G., RUMBAUGH, J., JACOBSON, I. *The unified modeling language user guide*. Addison-Wesley, 1999.
- [20] Noy, N.F., MCGuinness, D.L. *Ontology development 101: A guide to creating your first ontology*, Relatório Técnico KSL-01-05, 2001, Stanford Knowledge Systems Laboratory, Stanford.
- [21] L. S. Cardoso. *Aplicação da Tecnologia de Agentes de Planejamento em Operações de Satélites*. Dissertação de Mestrado em Ciência da Computação. INPE, 2006.