



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA  
**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

*Modelagem de sistemas complexos usando  
ferramentas de Análise Estruturada e OPN (Object  
Process Network)*

RELATÓRIO FINAL DE PROJETO DE INICIAÇÃO CIENTÍFICA  
(PIBIC/CNPq/INPE)

Gustavo Pinheiro Melo (ITA, Bolsista PIBIC/CNPq)  
E-mail: [gusta.pinheiro@gmail.com](mailto:gusta.pinheiro@gmail.com)

Dr. Geilson Loureiro (LIT/INPE, Orientador)  
E-mail: [geilson@lit.inpe.br](mailto:geilson@lit.inpe.br)

COLABORADORES

Felipe Simon (ITA, Bolsista PIBIC/CNPq)

Julho de 2007

# SUMÁRIO

Capítulo 1 - Introdução e Objetivos do Trabalho .....	1
Capítulo 2 - Plano De Trabalho .....	4
Capítulo 3 - Análises Iniciais .....	5
Capítulo 4 - Progresso .....	5
Capítulo 5 - Trabalhos Futuros .....	16
Capítulo 6 - Conclusões .....	16
Capítulo 7 - Bibliografia .....	16

## 1. Objetivos

---

Este projeto tem por objetivo a utilização de uma meta-linguagem de arquitetura de sistemas, chamada OPN (Object Process Network), de cujo desenvolvimento o orientador participou quando de seu estágio pós-doutoral no MIT (Massachusetts Institute of Technology), para a transformação da notação de modelagem de Análise Estruturada sistemas baseada na metodologia desenvolvida por Yourdon & Mellon e Hatley & Pirbhai, de um simples descritor de sistemas em uma linguagem que suporte computação dos atributos do sistema e simulação de diversas opções de arquitetura, permitindo modificação, combinação e escolha de alternativas de arquiteturas de sistemas de acordo com parâmetros tais como valor, custo, risco, prazo de desenvolvimento.

Objetivos Específicos:

1) Estudar OPM (Object Process Methodology) e OPN (Object Process Network). OPN é uma meta-linguagem de arquitetura de sistemas que permite não somente a descrição de modelos de arquitetura, mas também a computação de parâmetros desses modelos e a simulação desses modelos. No entanto, OPN não possui uma notação especificamente elaborada para a modelagem de comportamento e estrutura de sistemas tais como as notações de Análise Estruturada.

2) Estudar as notações de projeto e Análise Estruturada de sistemas propostas por Yourdon & Mellon e Hatley & Pirbhai. A Análise Estruturada é especialmente interessante para modelagem hierárquica de sistemas complexos a partir de um contexto do sistema. Ela se caracteriza por separar as modelagens essenciais (ou conceituais) de um sistema das modelagens de sua implementação.

3) Identificar como cada notação de Análise Estruturada pode ser traduzida para a notação de modelagem de OPN.

4) Identificar as limitações para essa tradução.

5) Sugerir mudanças em OPN para minorar essas limitações.

6) Analisar a viabilidade da implementação prática de uma ferramenta computacional de modelagem que permita a descrição dos modelos computáveis e simuláveis em Análise Estruturada.

7) Propor plano de implementação, caso viável.

## **2. Plano de Trabalho**

---

- 1) Revisão bibliográfica dos assuntos:
  - 1.1) OPM – Metodologia de modelagem de sistemas que permite modelagem de estrutura e comportamento de sistemas usando uma mesma notação;
  - 1.2) OPN – Análise de tese de doutorado do MIT sobre a ferramenta e seu relacionamento com OPM;
  - 1.3) Análise Estruturada – Através dos métodos de Yourdon & Mellon e Hatley & Pirbhai;
  
- 2) Aprendizado e utilização de ferramentas de modelagem computacionais disponíveis:
  - 2.1) OPN – disponível no LIT, com acessos às últimas versões em desenvolvimento no MIT.
  - 2.2) Cradle – ferramenta computacional que permite a elaboração de modelos em Análise Estruturada.
  
- 3) Desenvolvimento de modelos de um produto em Análise Estruturada.
  - 3.1) Modelo de requisitos – análise de stakeholders, das medidas de efetividade e documentação de seus requisitos.
  - 3.2) Modelos funcionais – contexto, interfaces externas, funções, análise de risco, estrutura funcional (funções e fluxos de energia, material e/ou informação) e comportamento funcional (estados, transições, ações, habilitação e desabilitação de funções).
  - 3.3) Modelagem de Arquitetura Física – contexto, interfaces externas, análise de risco, arquitetura de fluxos, arquitetura de interconexões.
  - 3.4) Modelo de Atributos – parâmetros físicos e seus relacionamentos com desempenho, custo, risco e prazo de desenvolvimento.
  
- 4) Desenvolvimento de modelos de um produto em OPN: Os mesmos modelos desenvolvidos em Análise Estruturada serão desenvolvidos em OPN.
  
- 5) Tradução de Análise Estruturada em OPN.
  - 5.1) Identificação de correspondências entre modelos.
  - 5.2) Generalização dessas correspondências.
  - 5.3) Limitações para tradução.
  - 5.4) Necessidades de modificação em OPN.
  - 5.5) Elaboração de plano de implementação computacional dessa tradução.
  
- 6) Documentação do projeto.

### 3. Análises Iniciais

---

As etapas iniciais deste projeto consistiram, como previsto no plano apresentado, na revisão do material bibliográfico proposto. Apresentações foram confeccionadas e apresentadas durante as três primeiras etapas do Plano de Trabalho. O estudo da bibliografia referente à Análise Estruturada revelou ser interessante propor uma metodologia para desenvolver modelos que combine os métodos de Yourdon & Mellon e Hatley & Pirbhai. Este “método combinado” está disponível em anexo (na apresentação *Desenvolvimento do Modelo de Requisitos*). Outras apresentações confeccionadas apresentam de forma sintética noções acerca do assunto tratado neste trabalho.

A conclusão destas etapas iniciais permitiu adquirir conhecimento suficiente para de fato analisar-se a viabilidade de implementação de uma ferramenta capaz de associar linguagens de modelagem tradicionais (como a referida Análise Estruturada) a uma ferramenta de suporte ao processo de decisão (OPN – Object-Process Network). Um maior detalhamento acerca destas linguagens e ferramentas será dado no próximo tópico. Posteriormente, será sumarizado o progresso alcançado por meio deste trabalho.

As etapas seguintes previstas no plano de trabalho envolveram o desenvolvimento de modelos em Análise Estruturada e OPN, possibilitando a identificação de correspondência entre os dois. As conclusões obtidas por meio destas comparações também são apresentadas no tópico a seguir.

### 4. Progresso

---

O processo de desenvolvimento de sistemas sócio-técnicos complexos frequentemente envolve tarefas como a transferência de conhecimento entre as mais diferentes áreas do conhecimento e a computação de parâmetros de interesse. Além disso, o universo de opções a ser considerado aumenta à medida que o sistema estudado se torna mais complexo. A experiência mostrou ser essencial a utilização de técnicas de análises arquiteturas quando do desenvolvimento de tais sistemas. O conceito por trás de tais técnicas é que decisões razoáveis podem ser tomadas avaliando-se parâmetros relevantes para um sistema em específico, comumente relacionadas a aspectos de custo e prazo. Apesar disso, até o momento estão disponíveis apenas ferramentas e técnicas capazes de trabalhar com necessidades específicas. Nenhuma delas se propõe a atacar todas estas principais necessidades em uma abordagem integrada. Este é o motivo pelo qual *Object-Process Network* (OPN) surge como uma ferramenta única na arquitetura de sistemas. Ela permite unificar os processos de decomposição do problema (permitindo administrar sua complexidade), composição (analisando o sistema como um todo) e finalmente avaliando as possíveis arquiteturas.

Essencialmente, esta meta-linguagem executável é uma ferramenta de suporte ao processo de decisão que fornece uma maneira sistemática de usar ferramentas computacionais para avaliar opções durante estágios iniciais do desenvolvimento de sistemas.

Por outro lado, linguagens de modelagem tradicionais como Análise Estruturada, *Unified Modeling Language* (UML) e, mais recentemente, SysML se posicionam como métodos descritivos que nos dão uma visão estática do sistema. Isto significa que elas não nos permitem simular ou computar parâmetros. Entretanto, a maioria dos clientes, por exemplo, compreendem métodos estruturados mais facilmente que outras notações. Como a comunicação com *stakeholders* e usuários é a principal razão para se modelar um sistema, ter um modelo final em Análise Estruturada é uma excelente maneira de apresentar resultados obtidos por OPN. Assim, o grande benefício obtido ao se combinar linguagens descritivas com OPN é que, dado um conjunto de possíveis arquiteturas, modelos comunicativos podem ser automaticamente compostos.

Não apenas ter um modelo final em uma linguagem descritiva, mas também definir opções e restrições que limitam o universo de opções do problema através de um destes modelos certamente facilitaria o trabalho dos chamados *decision-makers*, sem grandes dificuldades para entender o que está sendo modelado.

O objetivo deste projeto se torna então propor uma abordagem para conceber sistemas complexos. Como linguagem descritiva e comunicativa, foi escolhida *Object-Process Methodology* (OPM), devido à possibilidade de especificar tanto aspectos estruturais como comportamentais de um sistema utilizando para tanto um pequeno conjunto de símbolos. Este projeto seria então um primeiro passo rumo à ideal “concepção automática de um sistema”. Este objetivo final permitiria selecionar uma melhor arquitetura uma vez que as funções a serem desempenhadas e restrições tenham sido especificadas. A seguir são apresentadas brevemente as linguagens consideradas neste trabalho.

#### **4.1. A abordagem atual**

Quando do desenvolvimento de um novo sistema ou do aprimoramento de um já existente, a abordagem comum e atual adotada é modelar o problema e sua solução utilizando as mais variadas linguagens descritivas, entre elas SysML, Análise Estruturada e OPM. Mesmo que Análise Estruturada, por exemplo, tente modelar o problema (seus requisitos) desconsiderando aspectos de implementação, é inevitável a tendência a uma arquitetura em particular para o sistema, em vez de considerar as diversas possíveis soluções. Estas linguagens “estáticas” não oferecem recursos simulacionais ou computacionais. São puramente visuais.

Atualmente, ferramentas de suporte à decisão estão complementadas desvinculadas de linguagens de modelagem de sistema. Assim, quando da tomada de decisões, instrumentos para uma compreensão visual da arquitetura do sistema em questão não está disponível e, em contrapartida, quando da modelagem, instrumentos que auxiliem a decidir em que direção seguir não estão disponíveis.

O estado da arte é, de um lado, OPN oferecendo um conjunto completo de opções combinatoriais e forte suporte ao processo de decisão, mas necessitando aprimoramentos em detalhamento visual da modelagem e, por outro lado, linguagens como Análise Estruturada, SysML e OPM, oferecem compreensíveis e detalhadas modelos mas sem nenhum recurso de suporte à decisão. Em outras palavras, os processo de decisão e modelagem da arquitetura escolhida são feitos separadamente, o que produz, por exemplo, decisões com menos que a informação disponível a cerca do sistema ou modelos que não serão jamais usados. Esta seção apresenta uma breve revisão de OPN e OPM, uma vez que serão fundamentais para a compreensão das seções seguintes.

### 4.1.1 Object-Process Network

Estudos referentes ao atual processo de modelagem adotado por arquitetos de sistemas provaram ser essencial o desenvolvimento de uma ferramenta capaz de dar suporte a tarefas freqüentes quando do desenvolvimento de um sistema: enumeração do espaço de possibilidades, dadas restrições que limitam o problema; computação de todas as arquiteturas possíveis (considerando o fornecimento de expressões booleanas que limitam tal universo), baseadas nas opções fornecidas na primeira tarefa; e, finalmente, avaliar todas estas soluções, apontando aquelas preferíveis. Prof. Benjamin Koo, em seu PhD no MIT, produziu a chamada Object-Process Network, uma meta-linguagem baseada em um pequeno conjunto de primitivos que unifica as três principais tarefas apontadas em arquitetura de sistemas. Uma ferramenta baseada em Java dá suporte a esta meta-linguagem.

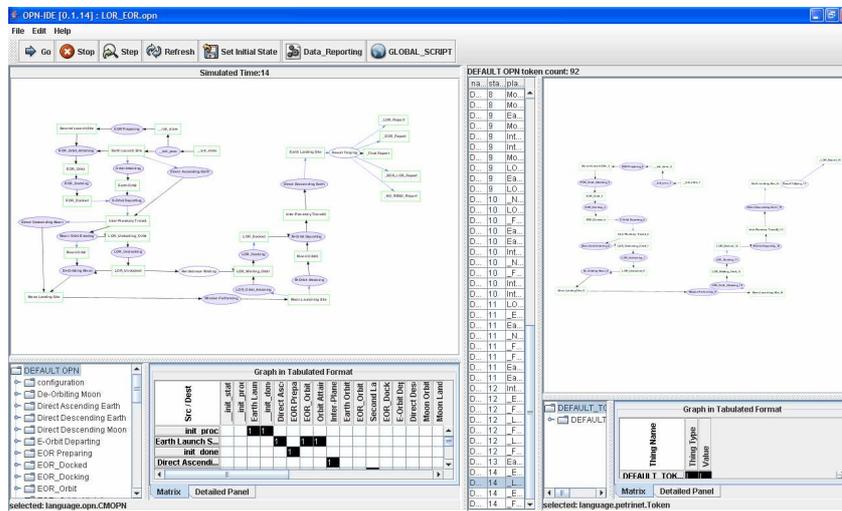


Figura 1. Screen capture da ferramenta OPN.

Um grafo de OPN é composto apenas por processo, objetos e relações conectam duas destas entidades (apenas entre blocos de tipos diferentes). Um processo é definido por funções que modificam parâmetros escolhidos. Assim, as melhores arquiteturas podem ser inferidas através da análise dos valores finais destes parâmetros. Cada *thread* em grafo OPN apontará para um conjunto diferente de valores finais. Figura 1 apresenta na direita um grafo OPN e na esquerda um possível caminho a ser seguido.

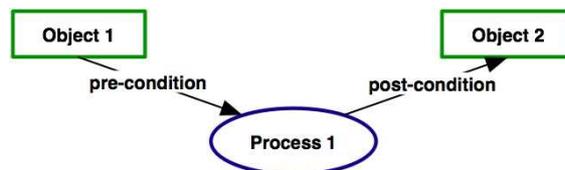


Figura 2. OPN Overview: Objeto é relacionado a processo através de uma pré-condição ou pós-condição

O diferencial em OPN está na sua capacidade de ajudar *stakeholders* a tomar decisões arquiteturais. Atualmente, OPN não oferece a possibilidade de converter uma série de decisões sugeridas em um modelo do sistema de interesse. Além disso, modelar o espaço de opções em um grafo OPN pode ser tornar uma tarefa difícil à medida que o sistema se torna mais complexo.

#### 4.1.2 Object-Process Methodology

Multiplicidade de modelo devido ao excesso de símbolos foi apontada como um sério problema em *Unified Modeling Language* (UML). Uma nova abordagem para desenvolver sistemas em geral foi desenvolvida com o intuito de superar o problema do excesso de notação. Object-Process Methodology utiliza três tipos de entidades: objeto, processo (que modifica o estado de um objeto e estado. Muitos tipos de *links* estão disponíveis para definir o tipo de relações existente entre tais blocos. Estas relações podem ser estruturais ou procedurais. A primeira expressa uma relação persistente, de longa duração; a segunda expressa comportamento do sistema. Um outro recurso em OPM é o suporte tanto para modelos gráficos como textuais: Object-Process Diagrams (OPDs) apresentam o sistema por meio de símbolos para objetos, processos e links; Object-Process Language (OPL) apresenta o sistema por meio de sentenças. Object-Process Case Tool (OPCAT) é o software desenvolvido para demonstrar a aplicabilidade desta metodologia.

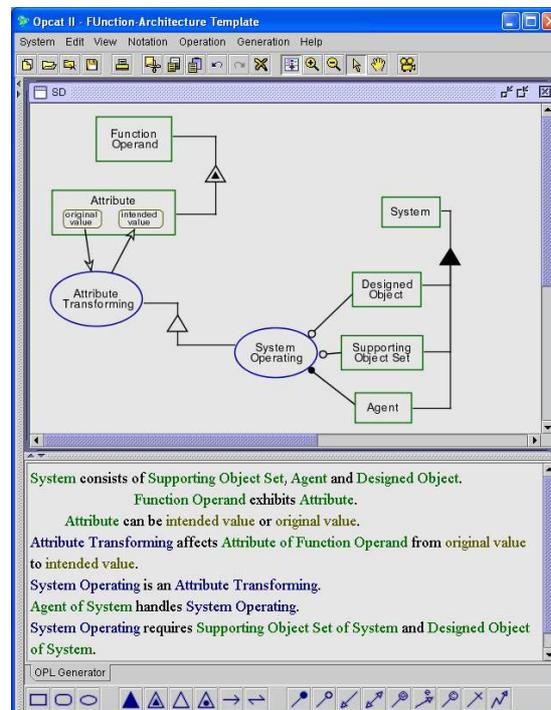


Figura 3. Screen capture de OPCAT.

## 4.2. A Nova Abordagem

O objetivo primário deste estudo era verificar a viabilidade de traduzir um modelo em linguagem descritiva tradicional para um grafo OPN. Estudos mostraram que estas linguagens não são compatíveis. Na realidade, OPN tem uma proposta diferente das outras linguagens. Mas a associação destas duas diferentes abordagens caminhará rumo a uma poderosa ferramenta de modelagem de arquiteturas.

Em vez de considerar um universo de possibilidades, atualmente as linguagens descritivas relacionadas tendem a privilegiar uma arquitetura em especial. Apesar disso, cada uma destas linguagens apresenta um recurso ou aspecto peculiar. Análise Estruturada, por exemplo, permite a criação de uma hierarquia de modelos que não dependem de suas implementações. Em cada nível de abstração, seria possível utilizar OPN para ajudar a escolher melhores arquiteturas a serem desenvolvidas. Este último conceito também poderia ser aplicado utilizando OPM, como será mostrado no exemplo desenvolvido.

Todas estas idéias referentes à união de linguagens e ferramentas existentes dão origem à chamada “Nova Abordagem” em Arquitetura de Sistemas. Essencialmente, ela significa obter o melhor de OPN e linguagens descritivas de uma só vez. A figura a seguir ilustra esta idéia.

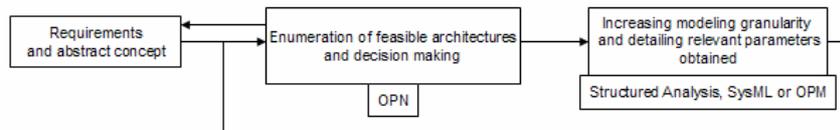
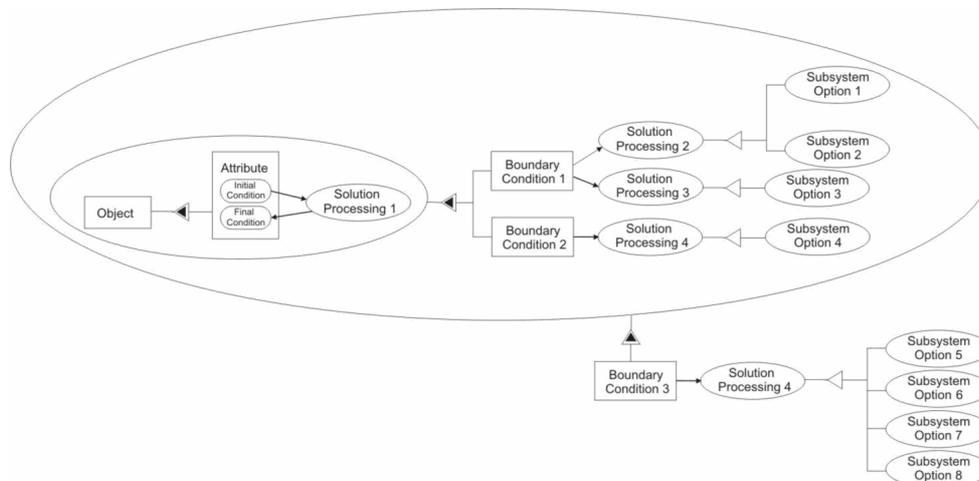


Figure 4. Esquema para a Nova Abordagem

A primeira idéia central é a definição de estados iniciais e finais relacionados ao sistema. Por exemplo, aplicando tal conceito para o Programa Apollo, o estado inicial seria “Homem na Terra” e o final “Homem na Lua”. A fim de evitar o espaço de possibilidades infinito, são definidas as condições de contorno referentes ao sistema. Estas são os principais obstáculos que impedem a mudança do estado inicial para o desejado estado final. Pode-se perguntar: Como transpor tais “obstáculos”? Para cada um deles, um novo subsistema é designado, permitindo os mesmos serem transpostos. Por exemplo, o que impede o homem ir da Terra até a Lua? A primeira destas restrições seria “Distância”. Para esta restrição em específico, um dado número de subsistemas (“opções”) está disponível.

Este processo pode continuar se, para o subsistema funcionar apropriadamente, novas condições de contorno tenham que ser definidas. O processo de iteração pode parar quando os subsistemas gerados podem ser considerados “blocos mínimos”, no sentido que não existem novas condições de contorno e não há, assim, necessidade de “descer” em níveis de abstração. Esta idéia está esquematizada na figura 5. Esta é o chamado “Conceito Geral” para a Nova Abordagem.



**Figura 5.** Modelo OPM para o conceito proposto.

Pode-se argumentar que o conceito aqui apresentado não permite tomar decisões de alto-nível, existiria uma “obrigação” de sempre descer para níveis inferiores (geração de subsistemas). Isto não é verdade. O problema é que tomar decisões em altos níveis significa ter menos parâmetros para avaliar as arquiteturas. Imagine que um novo sistema complexo deve ser desenvolvido e existem três “correntes” completamente diferentes a serem seguidas. A menos que este não seja um sistema totalmente novo, não se teria parâmetros para inferir melhores arquiteturas sem a descida para níveis mais baixos de cada uma destas “correntes”. Simular a partir em altos níveis utilizando apenas macro-parâmetros poderia trazer resultados inconsistentes. Por isso, é sugerido continuar o processo de iteração até que “subsistemas mínimos” sejam modelados. Definir um subsistema como “mínimo” é puramente uma questão de conveniência ou disponibilidade de informações. Por exemplo, se for possível inferir expressões que descrevam um dado subsistema (a partir de uma base de dados, por exemplo), então ele pode ser considerado mínimo e não há necessidade de continuar o processo de iteração para ele. Em outras situações, pode ser mais conveniente desenvolver um modelo inteiro para o subsistema e apenas depois será possível descrevê-lo em termos de seus principais parâmetros. Note que *Subsystem Option 1* e *Subsystem Option 2* são considerados no modelo da figura 5 “subsistemas mínimos”.

O primeiro passo irá governar o desenvolvimento do resto do modelo. Trata-se da definição de estados iniciais e finais e condições relacionadas ao problema central. Os exemplos apresentados a seguir mostram sistemas familiares e a definição de mudança de estados relacionada ao problema central (modelos desenvolvidos com OPCAT II – Object Process Case Tool).

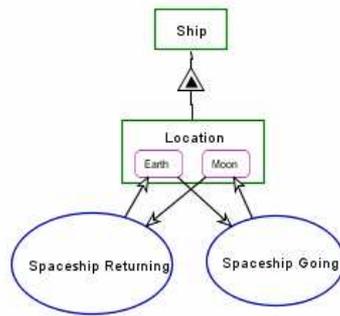


Figura 6. Denifição de mudança de estados para o sistema “Spaceship from Earth to the Moon”.

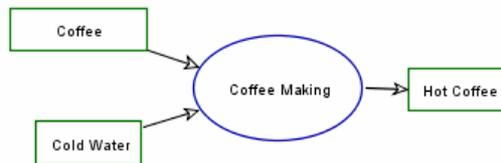
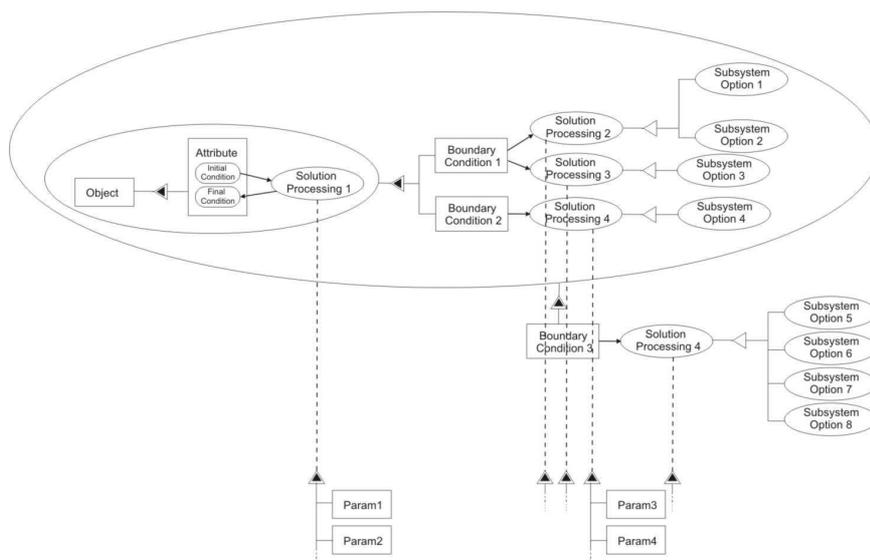


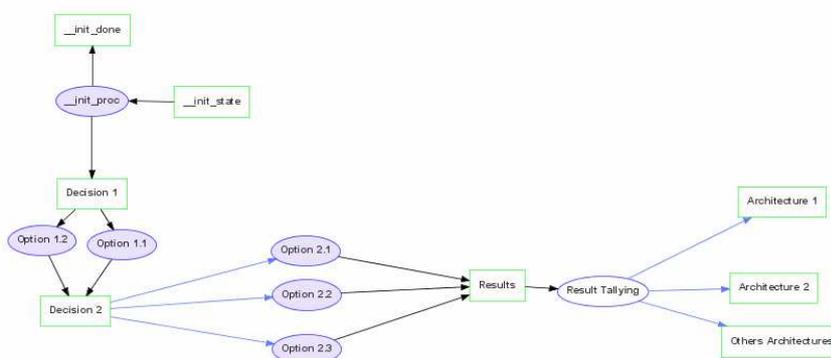
Figure 7. Definição de mudança de estados para o sistema “Coffee Machine”.

As informações apresentadas anteriormente (figura 5) ainda não são suficientes para simular em OPN. Esta ferramenta de suporte à decisão requer parâmetros que serão modificados pelas funções e, finalmente, avaliados. O próximo modelo apresenta estes parâmetros necessários. Eles estão conectados aos seus respectivos processos por meio de linhas pontilhadas. Estes parâmetros estão relacionados a um processo por meio de funções que o descreve.



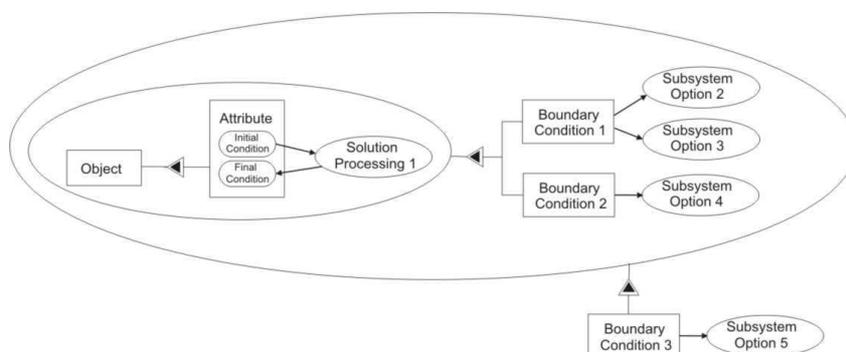
**Figure 8.** Modelo OPM para o conceito proposto com parâmetros a serem simulados.

Estão disponíveis opções e parâmetros (relacionados a estas opções por meio de funções e assim um modelo OPN pode ser criado. Baseando-se em requisitos mínimos, o processo de simulação irá apontar possíveis arquiteturas. Estes requisitos podem ser risco, tempo, custo etc.



**Figura 9.** Modelo OPN a ser simulado.

Os resultados do processo de simulação permitem derivar uma solução (“arquitetura”) para o sistema a ser desenvolvido (considerada entre as melhores possíveis).



**Figure 10.** OPM model derivado a partir do processo de simulação.

### 4.3. Estudo de Caso – Mercado de Sodas

No último tópico, o novo conceito aqui introduzido foi detalhadamente descrito, mas nenhuma aplicação concreta foi apresentada. Poder-se-ia argumentar acerca da inaplicabilidade de tal método para sistemas com alguma complexidade. O exemplo a seguir apresentado é prova de que é possível desenvolver novos sistemas ou aperfeiçoar existentes por meio desta abordagem. O caso estudado foca questões logísticas da distribuição de refrigerantes. O problema pode ser sintetizado por “Entregar uma Lata de Refrigerante a um cliente”.

O modelo OPM na figura 11 mostra a definição de estados e condições para o mercado de sodas. Note como o “núcleo” do problema (mudança de estados) está relacionado às condições de contorno como por exemplo *Distance*. Novos subsistemas devem ser modelados a fim de “solucionar” as referidas condições. Este é a primeira iteração. O processo de iteração continua até que “minimum subsystems” são modelados, como *Car*, *Train*, *Airplane*, etc. O próximo modelo apresentado acrescenta os parâmetros a serem simulados (figura 12). Nova notação foi adicionada ao conjunto original de símbolos fornecido por OPM. As linhas pontilhadas na figura 11, por exemplo, representam relações existentes no sistema, o que permitirá mais adiante derivar um modelo OPM para os resultados fornecidos por OPN (daí a importância destas linhas).

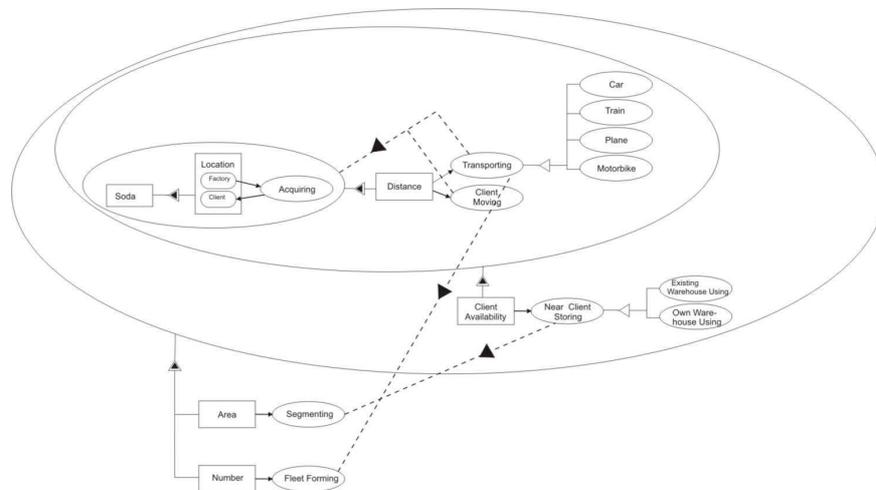


Figure 11. Modelo OPM para a logística da distribuição de refrigerantes.

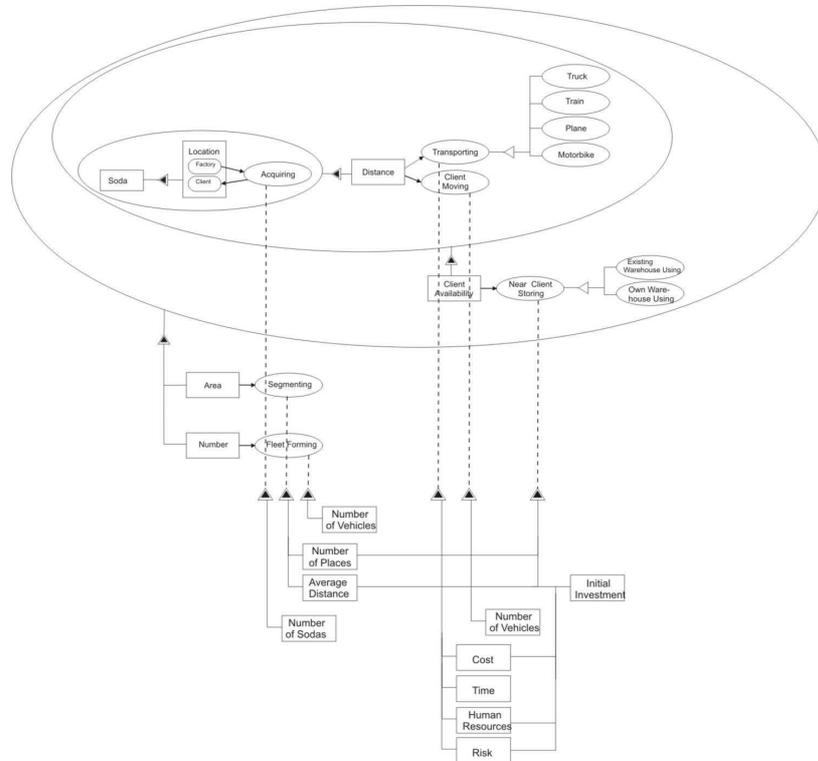
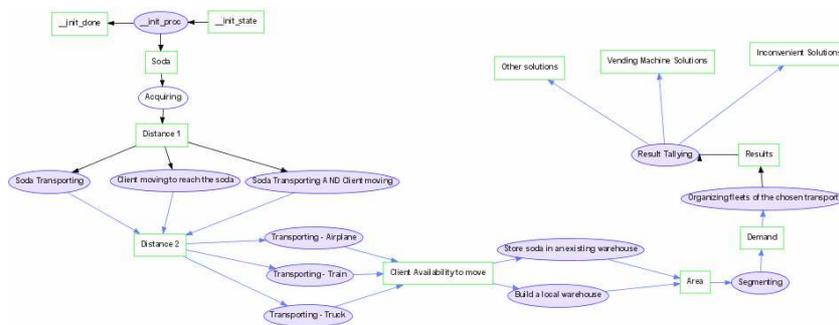


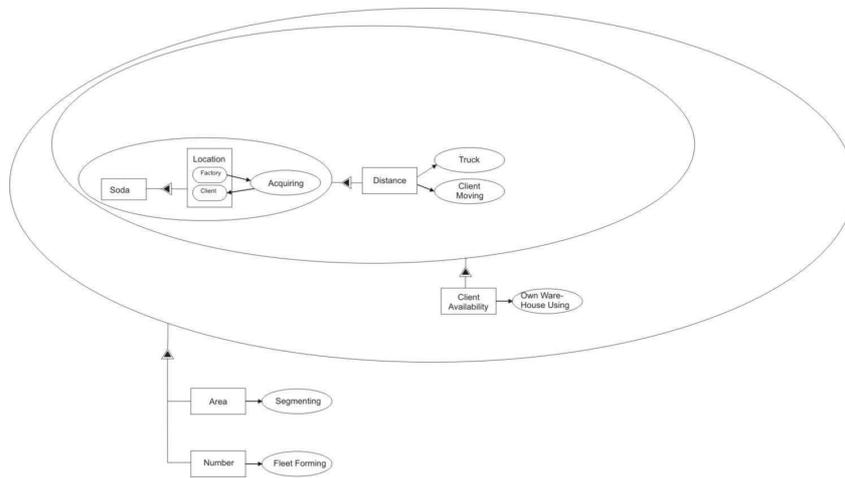
Figura 12. Modelo OPM com parâmetros (relacionados a suas respectivas funções pelas linhas pontilhadas).

Baseando-se na informação obtida no primeiro processo, pode-se agora derivar um modelo OPN simulável, capaz de enumerar soluções para os problemas em questão seguindo as restrições definidas (figura 13).



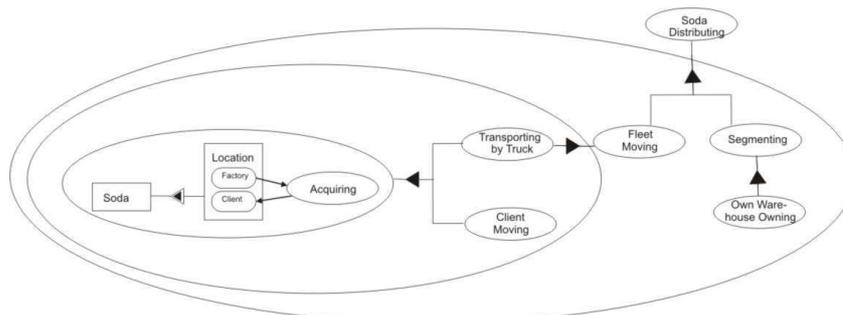
**Figure 13.** Modelo OPN gerado

Como exposto anteriormente, OPN se encaixa como uma ferramenta de suporte à decisão. Assim, seus recursos de análise apontariam uma variedade de modelos desejáveis para o problema em estudo. Por exemplo, após simular as opções de arquitetura, pode-se construir instâncias de sistemas usando OPM. Dadas uma série de decisões tomadas a partir do processo de simulação, é possível enfim montar um modelo OPM (figura 14). Na figura 8, isto equivale a substituir todos *Solution Processing* pelos *Subsystem Option* selecionados.



**Figure 14.** Modelando as decisões tomadas utilizando OPM.

O último passo no processo de composição do sistema é, baseando-se nas decisões tomadas, modelar o sistema propriamente. Isto pode ser sistematicamente feito eliminando-se todas as condições de contorno, seus links e as opções não selecionadas e tornando contínuas as linhas pontilhadas da figura 12.



**Figure 15.** Modelo OPM para a solução/arquitetura escolhida

## **5. Trabalhos Futuros**

---

Como próximo passo em direção do objetivo final de mecanizar o processo de concepção de sistemas, novos estudos devem ser feitos. Algumas questões permanecem em aberto. Como decidir até que ponto se deve modelar a fim de não despendar muito esforço modelando uma solução que nunca será desenvolvida? Quando da composição e arquitetura de modelos utilizando diferentes notações de diferentes porções de modelos, como tornar o “todo” coerente? Como integrar diferentes modelos se estes são escritos com notações de acordo com a conveniência do arquiteto? Solucionadas estas questões, o próximo passo previsto é a mecanização da abordagem proposta através de um software de interface amigável.

## **6. Conclusões**

---

Neste estudo foi apresentada uma inovadora abordagem para o desenvolvimento de sistemas complexos. A aplicabilidade deste método que se propõe a gerar automaticamente arquiteturas foi mostrada por meio do estudo de logística para o mercado de refrigerantes. Além da possibilidade de mecanizar o processo de concepção de sistemas, esta abordagem permitiria identificar novas soluções tradicionalmente descartadas por requisitos baseados em experiências passadas. Por exemplo, o Programa Apollo foi governado nos anos 60 por tempo e risco. A solução apontada foi viajar via órbita lunar. Sistemas de exploração espaciais atuais são preferencialmente governados por custo. A solução apontada seria então um vôo direto ao solo Lunar, por exemplo. Certamente, mudar requisitos influenciará no espaço de opções em questão. Utilizando tal metodologia significa que não será necessário iniciar todo o processo de decisão novamente caso estas restrições mudem. Novos estudos são esperados para responder aos problemas apontados para esta Nova Abordagem.

## **7. Bibliografia**

---

- [1] D. Dori, Object-Process Methodology: A Holistic Systems Paradigm: Springer-Verlag, 2002.
- [2] Yourdon, Edward. Modern Structured Analysis. Prentice Hall, 1989.

- [3] Derek J. Hatley, Imitiaz A. Pirbhai. *Strategies for Real-Time System Specification*. Dorset House, 1988.
- [4] G. Booch, J. Rumbaugh, I. Jacobson. *The Unified Modeling Language User Guide*. Addison-Wesley Professional, 2005.
- [5] B. H. Y. Koo. *A Meta-language for Systems Architecting*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 2005.
- [6] W. L. Simmons, B. H. Y. Koo, and E. F. Crawley. Architecture generation for Moon-Mars exploration using an executable meta-language. In *Proceedings of AIAA Space 2005*, 30 August - 1 September, Long Beach, CA, 2005.
- [7] W. L. Simmons, B. H. Y. Koo, and E. F. Crawley. Space systems architecting using meta-languages. In *56th International Astronautical Congress*, 2005.
- [8] W. L. Simmons, B. H. Y. Koo, and E. F. Crawley. A Computational Method for Mapping The Decision Space of the Lunar Exploration Program. In *57th International Astronautical Congress*, 2006.
- [9] D. Dori, I. Reinhartz-Berger, and A. Sturm, "Developing Complex Systems with Object-Process Methodology with OPCAT", presented at *Conceptual Modeling - ER 2003*, 2003.
- [10] [http://opn.mit.edu/index.php/OPN\\_Quickstart](http://opn.mit.edu/index.php/OPN_Quickstart). Accessed on: Sept. 1st 2006.
- [11] <http://www.opcat.com>. Accessed on: Sept. 1st 2006.
- [12] <http://www.omgsysml.org>