

PASoS: Processo para definição da arquitetura de Sistemas-de-Sistemas

Thiago J. Inocêncio
Federal University of ABC
Santo André, Brazil
thiago.inocencio@ufabc.edu.br

Gustavo R. Gonzales
Federal University of ABC
Santo André, Brazil
gustavo.romao@ufabc.edu.br

Flávio E. A. Horita
Federal University of ABC
Santo André, Brazil
flavio.horita@ufabc.edu.br

ABSTRACT

A System-of-Systems (SoS) is a class of system characterized by the union of constituent systems that are independent in both operational and managerial and together they can play new functionalities. Because of their evolutionary and interoperability features, SoS are widely used in critical contexts such as disaster management, health care systems, military systems, and smart cities. An architectural process aims to define the stakeholders, constituent systems and their interactions and the emergent behaviors produced by these interactions. All of these elements are defined from an understanding of the context through requirements engineering, architectural descriptions, and modeling of architectural views of the system. Recent research has shown little progress in the study of architectural processes in this class of system and in this sense more researches are needed to meet the lacks found in the literature for the design, analysis, evaluation, and evolution of architectures in SoS. Therefore, this research introduces PASoS, an instance of SOAR (Meta-process for SoS Software Architectures) that aims to define what elements an architectural process in SoS should cope with. Thus, in future research the essential elements of SOAR (Alphas and activity spaces) will be generalized in the three phases of PASoS, in each one of them the SOAR activity spaces that in PASoS are responsible for producing the architectural artifacts (Alphas of SOAR) will be sequentially listed. This process is expected to guide SoS architects through the steps necessary to build the SoS architecture for critical contexts.

RESUMO

Um Sistema-de-Sistemas (SoS) é uma classe de sistema caracterizada pela união de sistemas constituintes que são independentes no modo operacional e gerencial e que juntos desempenham novas funcionalidades. Devido as suas características de evolução e interoperabilidade, os SoS são amplamente usados em diversos contextos críticos como gestão de desastres, sistemas de cuidados com a saúde, sistemas militares e cidades inteligentes. Um processo arquitetural de SoS visa definir os stakeholders, sistemas constituintes e como eles interagem entre si e os comportamentos emergentes produzidos por essas interações. Todos esses elementos são definidos a partir de um entendimento do contexto por

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2019 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM... \$15.00
<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

meio da engenharia de requisitos, descrição da arquitetura e da modelagem das visões arquiteturais do sistema. Pesquisas recentes mostraram poucos avanços no estudo de processos arquiteturais nessa classe de sistema e nesse sentido pesquisas são necessárias para atender às demandas encontradas na literatura para o projeto, análise, avaliação e evolução de arquiteturas em SoS. Com base nisso, esta pesquisa apresenta o PASoS (Processo para definição da Arquitetura de **Sistemas-de-Sistemas**), um processo criado para ser uma instância do meta-processo SOAR ("Meta-process for SoS Software Architectures") que visa definir por quais elementos um processo arquitetural em SoS deve ser composto. Para isso, em trabalhos futuros os elementos essenciais do SOAR (*Alpha* e espaços de atividades) serão generalizados em três fases do PASoS, onde em cada uma delas serão elencados sequencialmente os espaços de atividades definidos pelo SOAR que no PASoS são responsáveis por produzir os artefatos arquiteturais (*Alphas* do SOAR). Espera-se que essa instância de processo oriente os arquitetos de SoS na sequência de etapas necessárias para construção da arquitetura do SoS para contextos críticos.

CCS CONCEPTS

• **Information systems** → *Information systems applications*; • **Software and its engineering** → *Software architectures*; *Ultra-large-scale systems*.

KEYWORDS

Architectural Process, Software Architecture, System-of-systems, critical system

1 INTRODUÇÃO

De acordo com Pressman [18], o projeto arquitetural deve definir a natureza da interação entre as entidades que compõem um sistema (outros sistemas, dispositivos, pessoas) e são obtidas a partir da engenharia de requisitos. A partir disso é especificada a estrutura do sistema, definindo e refinando as entidades pertencentes e suas relações com base nos requisitos levantados. De acordo com o mesmo autor, a arquitetura de software permite avaliar se o projeto arquitetural atende aos requisitos, considerar alternativas arquitetônicas durante as fases iniciais da concepção do software e reduzir os riscos inerentes ao processo de construção de software.

Sistemas-de-sistemas (SoS) é um conceito bastante discutido e estudado na área de arquitetura de software, um aspecto importante na construção desse tipo de sistema é a definição de sua arquitetura que é relevante para entender os sistemas constituintes, a relação entre eles e avaliar se juntos são capazes de atender a determinados requisitos de negócio. Hans e Klen [11] realizaram uma revisão sistemática da literatura sobre arquitetura de Sistemas-de-Sistemas e

concluíram que em comparação com outros campos da engenharia de software, a pesquisa em arquiteturas de Sistemas-de-Sistemas cresceu moderadamente e no intuito de equilibrar essa situação, a revisão dispõe temas de estudos passíveis de serem explorados para que futuras pesquisas contribuam com o crescimento de estudos nesse tópico. A revisão levantou que pesquisas são necessárias para desenvolver procedimentos e técnicas para o projeto, análise, avaliação e evolução de arquiteturas de Sistemas-de-Sistemas, visto que a maioria das pesquisas analisadas na revisão apresentam abordagens variadas que solucionam problemas somente para contextos específicos de arquiteturas de SoS. Nesse sentido, esse artigo aborda o projeto arquitetural em Sistemas-de-Sistemas partindo do pressuposto que as entidades integrantes do sistema são outros sistemas constituintes que interagem entre si segundo a natureza do sistema em questão e de acordo com o domínio da aplicação. Portanto o objetivo deste artigo é apresentar um processo arquitetural para SoS, ou seja, um processo bem definido para descrever quais sistemas constituintes serão utilizados com intuito de atender aos requisitos do contexto estudado com base nas características e funcionalidades dos sistemas e a dinâmica pela qual eles trocam informações. Para isso, este artigo propõe o processo chamado PASoS (Processo para definição da Arquitetura de **Sistemas-de-Sistemas**) que visa auxiliar no processo de definição da arquitetura de SoS.

2 REFERENCIAL TEÓRICO

Nos últimos anos, a pesquisa e o desenvolvimento de sistemas convergiu de sistemas isolados (monolíticos) para sistemas complexos compostos de múltiplos outros sistemas heterogêneos e independentes, os Sistemas-de-Sistemas (SoS). Embora ainda não exista uma definição clara de SoS, esses sistemas podem ser entendidos como uma classe de sistemas que é capaz de combinar sistemas constituintes, que são heterogêneos e independentes, e trabalham em consonância para atingir um objetivo [10, 12, 15]. De acordo com Maier [12], esse entendimento pode ser complementado por cinco características que tornam os SoS diferentes dos outros tipos de sistemas:

- (a) *Independência Operacional*: os sistemas são capazes de operar de forma independente em relação ao SoS. Nesse sentido, eles realizam suas operações normalmente ao mesmo tempo que contribuem para o funcionamento do SoS;
- (b) *Independência Gerencial*: cada constituinte de um SoS pode manter sua própria esfera gerencial, ou seja, é gerenciado de forma independente muitas das vezes por entidades que não fazem parte do contexto do SoS;
- (c) *Desenvolvimento evolutivo*: funções e propósitos do SoS evoluem de forma dinâmica uma vez que novos constituintes podem ser agregados para realizar novas tarefas;
- (d) *Comportamento Emergente*: SoS são capazes de fornecer novas funcionalidades que são resultados do trabalho conjunto dos constituintes; essa é a característica central do SoS.
- (e) *Distribuição Geográfica*: os constituintes de um SoS são distribuídos geograficamente em larga escala.

Com base nas características dos SoS apresentadas anteriormente, percebe-se que essa classe de sistemas está também inserida na classe de sistemas críticos de larga escala, pois eles são frequentemente aplicados em diversos contextos críticos tais como militar

[4, 7, 12, 14], saúde [2, 19] e gestão de desastres [1, 6, 8, 9, 15–17], possuem comportamentos emergentes que podem ou não ser desejados e são distribuídos em larga escala (Distribuição Geográfica).

Além dessas características, Maier [13]¹ identifica cinco categorias de SoS:

- (i) *Closed SoS*: o sistema é gerenciado por uma entidade que controla os objetivos, financiamento e autoridade dos constituintes com vistas a realização de um propósito;
- (ii) *Acknowledged*: a entidade que gerencia o sistema utiliza sua autoridade e recursos para fazer o SoS trabalhar melhor. No entanto, essa autoridade é limitada, uma vez que os constituintes têm seus objetivos individuais que são atendidos em paralelo aos objetivos do SoS;
- (iii) *Open SoS*: o sistema tem um propósito global e é gerenciado por uma entidade que não exerce controle sobre os constituintes, sendo assim a entidade deve encontrar meios de convencer os constituintes a cooperar com o SoS de forma voluntária;
- (iv) *Virtual SoS*: nesse caso não existe um objetivo global e nem mesmo uma entidade que gerencia o sistema, os constituintes interagem entre de si de forma deliberada e tal interação gera indiretamente novos comportamentos emergentes;
- (v) *Organic SoS*: podem ser considerados como *Virtual SoS*, mas são voltados para domínios sociais em vez de técnicos.

Já no ponto de vista da *MITRE Corporation* [3], os SoS são classificados em quatro categorias:

- (i) *virtual SoS*: o sistema não possui um objetivo global, os constituintes podem não se conhecer, eles simplesmente descobrem possíveis serviços que podem ser compartilhados;
- (ii) *collaborative SoS*: os constituintes trabalham de forma voluntária com intuito de atingir objetivos comuns ou até mesmo individuais através de mecanismos avançados de comunicação;
- (iii) *acknowledged SoS*: o sistema é construído para desempenhar atividades específicas, mas os constituintes mantêm seu próprio gerenciamento, financiamento e autoridade em paralelo;
- (iv) *directed SoS*: o sistema é construído para exercer atividades bem específicas cuja autoridade é exercida por uma única entidade que é responsável pelo gerenciamento.

Com base nessas categorias, em um primeiro momento o processo proposto nesta pesquisa será concebido para atender à categoria de SoS conhecidos (*acknowledged*) conforme descrição da Tabela 1. Nessa classe de SoS a entidade que gerencia o sistema utiliza sua autoridade e recursos para fazer o SoS trabalhar melhor de tal forma que objetivos, gerencialmente, recursos utilizados e autoridade são bem definidos [5], no entanto, essa autoridade é limitada, uma vez que os constituintes têm seus objetivos individuais que são atendidos em paralelo aos objetivos do SoS.

¹ Este artigo decidiu por manter os termos em Inglês visto que não existe um consenso estabelecido quanto a tradução adequada para o Português.

Tabela 1: Relação entre as categorias de SoS da MITRE Corporation [3] e Maier [13]

MITRE Corp. [3]	Maier [13]	Descrição
Virtual SoS	Virtual SoS & Organic SoS	O sistema não possui um objetivo global e também não possui uma entidade gerenciadora. Dessa forma, os constituintes podem não se conhecer e interagem entre si de forma indireta com intuito de descobrir possíveis serviços que podem ser compartilhados.
Collaborative SoS	Open SoS	O sistema tem um objetivo global e é gerenciado por uma entidade que não exerce controle sobre os constituintes que cooperam com o sistema de forma voluntária.
Acknowledged SoS	Acknowledged SoS	O sistema é construído com objetivos específicos, possui uma entidade gerenciadora que exerce controle limitado sobre os constituintes.
Directed SoS	Closed SoS	O sistema exerce objetivos específicos coordenados por uma entidade gerenciadora que exerce controle sobre os constituintes com vistas à realização dos objetivos comuns.

3 METODOLOGIA

O objetivo deste artigo é propor um processo para definir o projeto arquitetural de SoS. O PASoS foi concebido para ser uma instância do SOAR (*Meta-process for SoS Software Architectures*), um meta-processo proposto por Gonçalves et al. [5] que define por quais elementos um processo arquitetural deve ser composto com intuito de auxiliar adequadamente o desenvolvimento de arquiteturas de SoS. A PASoS é composto por 3 fases: I. engenharia de requisitos, II. Definição da arquitetura e III. Modelagem das visões arquiteturais. Em nossa abordagem adequamos os conceitos de *Alphas* e espaços de atividades (*Activity Spaces*) definidos no SOAR, nesse sentido cada *Alpha* representa um artefato que será gerado pelo PASoS e os espaços de atividades serão representados por atividades dentro de cada uma das fases. Com isso, cada fase será composta por um conjunto de atividades (espaços de atividades do SOAR) onde em cada uma delas serão descritos um conjunto de passos que deverão ser seguidos para produzir os artefatos (*Alphas*), assim os artefatos serão construídos de forma incremental conforme o arquiteto finaliza as atividades.

Com intuito de auxiliar o arquiteto no registro das informações pertinentes a arquitetura do SoS, o PASoS define um conjunto de artefatos que serão produzidos durante as fases do processo, nesse sentido cada um dos *Alphas* do SOAR representa um artefato a ser produzido que será preenchido pelo arquiteto com auxílio dos scripts do processo, um artefato pode ser um documento ou diagrama. Os artefatos produzidos no PASoS são classificados como artefatos de entrada e saída, o primeiro é necessário quando no início da execução de uma nova atividade for necessário um conjunto de informações relevantes para que seja possível discorrer adequadamente os passos daquela atividade e consequentemente o segundo é o resultado da completa execução de uma atividade.

Cabe ressaltar que um artefato pode estar presente em diversas fases e dependendo da dinâmica de uma atividade tal artefato pode ser de entrada e/ou de saída, isso porque durante a execução do processo o que se está objetivando é a confecção e aprimoramento desses artefatos, pois de fato eles representam a definição da arquitetura do SoS.

Para auxiliar na confecção dos artefatos para cada um deles será definido um *script* de preenchimento que contém descrições de

como aquele artefato deverá ser produzido. No caso de um documento com diversos campos, um *script* de preenchimento descreverá o que se espera que o arquiteto preencha nos campos disponíveis, no caso de um diagrama, qual tipo e quais elementos poderão ser utilizados.

4 DEFINIÇÃO DO PROCESSO

O processo da definição da arquitetura de Sistemas-de-Sistemas proposto neste artigo pode ser visualizado na Figura 1 que mostra a sequência de fases necessárias que serão seguidas pelo arquiteto. Em cada uma dessas fases existem *scripts* e modelos de documentos para facilitar o processo como um todo.

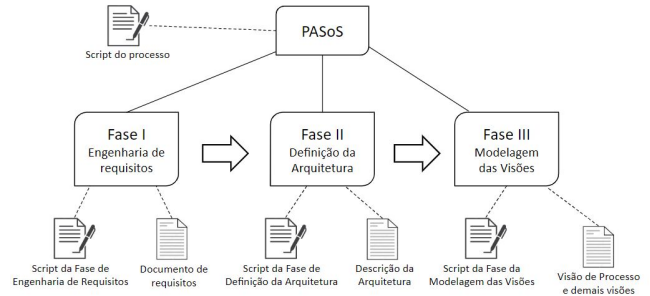


Figura 1: Fases do processo de definição da arquitetura de SoS

Os *scripts* são instruções (passo-a-passo) de como o arquiteto deve proceder na referida fase, eles descrevem quais são os artefatos de entradas e o que será produzido no final da fase que possivelmente servirá de artefato de entrada para a fase subsequente.

O propósito do PASoS é guiar a definição da arquitetura de Sistemas-de-Sistemas, para isso são necessários os requisitos dos sistemas levantados junto as partes interessadas, tais requisitos servem como artefatos de entrada do processo, a partir do qual as fases de engenharia de requisitos, definição arquitetural e modelagem das visões arquiteturais serão baseadas. No final, após a condução do processo, espera-se a produção de uma definição completa da arquitetura com as visões arquiteturais modeladas e com as principais decisões de projeto que serão seguidas no desenvolvimento do SoS.

A fase 1 (Engenharia de requisitos) se refere ao processo de definir, documentar e manter os requisitos do sistema, ela ocorre antes da fase de definição arquitetural do sistema, uma vez que é necessário conhecer as necessidades dos *stakeholders* do projeto para então descobrir quais serão os sistemas constituintes do SoS, os atributos e relações entre eles. Nesse sentido a partir do documento de requisitos e dos cenários de casos de uso o arquiteto tem a possibilidade de iniciar com a definição dos estilos e padrões arquiteturais que serão adequados para atender melhor aos requisitos.

Após a fase de engenharia de requisitos inicia-se a fase de definição arquitetural. De acordo com Pressman [18] cada decisão arquitetural pode ser documentada para revisão futura por parte dos *stakeholders* com intuito de entender a descrição da arquitetura proposta. Nessa fase utilizamos o *template* de descrição da decisão da arquitetura, conforme disposto por Pressman [18]. Assim como a fase 2, a fase de definição arquitetural possui um *script* para guiar o arquiteto.

A terceira e última fase é responsável por derivar as definições da fase 2 em visões arquiteturais, isso porque diferentes partes interessadas verão uma arquitetura a partir de diferentes pontos de vista que são impulsionados por diferentes conjuntos de preocupações. Isso implica que uma descrição arquitetônica é, na verdade, um conjunto de produtos de trabalho que refletem visões diferentes do sistema. Nessa fase são modeladas a visão lógica, visão de implantação, visão de processo e visão física.

5 CONCLUSÃO

Um Sistema-de-Sistemas (SoS) é uma classe de sistema caracterizada pela união de sistemas constituintes que são independentes no modo operacional e gerencial e que juntos desempenham novas funcionalidades. Com base nas características inerentes desses sistemas é possível considerá-los como sistemas críticos de larga escala, pois são frequentemente utilizados em contextos críticos como militar, internet, energia, e gestão de desastres naturais, possuem comportamentos emergentes e são distribuídos em larga escala. Um projeto arquitetural de SoS deve definir a natureza da interação entre os constituintes que compõem o SoS. Tais constituintes e suas relações são obtidas a partir da execução de um processo arquitetural. Nesse estudo foi apresentado um processo arquitetural de SoS definido com base no meta-processo SOAR que define quais são os elementos necessários que um processo arquitetural de SoS deve contemplar. Esse trabalho se limitou a definir os elementos do processo. Em trabalhos futuros todas as atividades, artefatos, *scripts* e formulários do PASoS retratados que estão em fase de confecção serão apresentados com detalhes com base no meta-processo SOAR e o processo completo será aplicado e avaliado por especialistas em estudos de casos que trabalhem com contextos críticos (e.g. gestão de desastres).

AGRADECIMENTOS

Os autores agradecem ao apoio financeiro realizado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Processo 437937/2018-6.

REFERÊNCIAS

- [1] R. Alexander, D. Kazakov, and T. Kelly. 2006. System of Systems Hazard Analysis Using Simulation and Machine Learning. In *Proceedings of the 25th International Conference on Computer Safety, Reliability, and Security*, J. Górski (Ed.). Lecture Notes in Computer Science, Vol. 4166. Springer-Verlag Berlin Heidelberg, Germany, 1–14. https://doi.org/10.1007/11875567_1
- [2] R. Benabidallah, S. Sadou, A. Esnault, and M. A. Nacer. 2018. Simulating systems of systems using situation/reaction paradigm. *Concurrency and Computation: Practice and Experience* 0, 0 (2018). <https://doi.org/10.1002/cpe.4921> to be published.
- [3] MITRE Corporation. 2014. *Systems Engineering Guide*. The MITRE Corporation. <http://www.mitre.org/sites/default/files/publications/se-guide-book-interactive.pdf>
- [4] C. E. de Barros Paes, V. V. Graciano Neto, T. Moreira, and E. Y. Nakagawa. 2018. Conceptualization of a System-of-Systems in the Defense Domain: An Experience Report in the Brazilian Scenario. *IEEE Systems Journal* 13 (2018), 1–10. <https://doi.org/10.1109/JSYST.2018.2876836>
- [5] Marcelo Benites Gonçalves, Flavio Oquendo, and Elisa Yumi Nakagawa. 2015. A Meta-process to Construct Software Architectures for System of Systems. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing (SAC '15)*. ACM, New York, NY, USA, 1411–1416. <https://doi.org/10.1145/2695664.2695737>
- [6] V. V. Graciano Neto. 2017. A Model-Based Approach Towards the Building of Trustworthy Software-Intensive Systems-of-Systems. In *Proceedings of the 39th IEEE/ACM International Conference on Software Engineering Companion*. IEEE, USA, 425–428. <https://doi.org/10.1109/ICSE-C.2017.28>
- [7] C. Guariniello and D. DeLaurentis. 2017. Supporting design via the System Operational Dependency Analysis methodology. *Research in Engineering Design* 28, 1 (Jan. 2017), 53–69. <https://doi.org/10.1007/s00163-016-0229-0>
- [8] M. Guessi, F. Oquendo, and E. Y. Nakagawa. 2016. Checking the architectural feasibility of Systems-of-Systems using formal descriptions. In *Proceedings of the 11th System of Systems Engineering Conference*. <https://doi.org/10.1109/SYSOSE.2016.7542939>
- [9] F. E. A. Horita, D. H. Rhodes, T. J. Inocêncio, and G. R. Gonzales. 2019. Building a conceptual architecture and stakeholder map of a system-of-systems for disaster monitoring and early-warning: A case study in Brazil. In *Proceedings of the 15th Brazilian Symposium on Information Systems*. ACM, New York, NY, USA, 1–8.
- [10] Mo Jamshidi. 2008. *Systems of Systems Engineering: Principles and Applications*. CRC Press.
- [11] John Klein and Hans van Vliet. 2013. A Systematic Review of System-of-systems Architecture Research. In *Proceedings of the 9th International ACM Sigsoft Conference on Quality of Software Architectures (QoSA '13)*. ACM, New York, NY, USA, 13–22. <https://doi.org/10.1145/2465478.2465490>
- [12] Mark W. Maier. 1998. Architecting principles for systems-of-systems. *Systems Engineering* 1, 4 (1998), 267–284. [https://doi.org/10.1002/\(SICI\)1520-6858\(1998\)1:4<267::AID-SYS3>3.0.CO;2-D](https://doi.org/10.1002/(SICI)1520-6858(1998)1:4<267::AID-SYS3>3.0.CO;2-D)
- [13] Mark W. Maier. 2015. *The Role of Modeling and Simulation in System of Systems Development*. John Wiley Sons, Ltd, Chapter 2, 11–41. <https://doi.org/10.1002/9781118501757.ch2> arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781118501757.ch2>
- [14] A. Mour, C. R. Kenley, N. Davendralingam, and D. DeLaurentis. 2014. Agent-Based Modeling for Systems of Systems. *INCOSE International Symposium* 23, 1 (June 2014), 973–987. <https://doi.org/10.1002/j.2334-5837.2013.tb03067.x>
- [15] Elisa Y. Nakagawa, Marcelo Gonçalves, Milena Guessi, Lucas B. R. Oliveira, and Flavio Oquendo. 2013. The State of the Art and Future Perspectives in Systems of Systems Software Architectures. In *Proceedings of the 1st International Workshop on Software Engineering for Systems-of-Systems (SESoS)*. 13–20. <https://doi.org/10.1145/2489850.2489853>
- [16] F. Oquendo. 2018. Exogenously Describing Architectural Emergent Behaviors of Systems-of-Systems with SosADL. In *Proceedings of the 13th Annual Conference on System of Systems Engineering*. IEEE, USA, 268–275. <https://doi.org/10.1109/SYSOSE.2018.8428715>
- [17] F. Oquendo. 2018. On the Emergent Behavior Oxymoron of System-of-Systems Architecture Description. In *Proceedings of the 13th Annual Conference on System of Systems Engineering*. IEEE, USA, 417–424. <https://doi.org/10.1109/SYSOSE.2018.8428727>
- [18] Roger Pressman. 2010. *Software Engineering: A Practitioner's Approach* (7 ed.). McGraw-Hill, Inc., New York, NY, USA.
- [19] N. Wickramasinghe, S. Chalasani, R. V. Boppana, and A. M. Madni. 2007. Healthcare System of Systems. In *Proceedings of the IEEE International Conference on System of Systems Engineering*. IEEE, USA. <https://doi.org/10.1109/SYSOSE.2007.4304283>