Introduction to Unified Architecture Framework® (UAF)

统一架构框架®(UAF)简介

# 作者

Aurelijus Morkevicius, PhD, Head of Solutions, DS No Magic Europe.

# 翻译

默识

# 摘要

# 架构框架在不断发展。考虑到行业需求并解决现有领域特定架构框架的不断变化的格局，2013年9月，创建了针对DoDAF和MODAF（UPDM3.0）统一配置文件的提案请求（后来更名为统一架构框架（UAF））。自提出建议邀请书（RFP）问题以来，UPDM3.0工作组确定了要求列表。仅举几例：国防，工业和政府组织的架构建模支持，安全领域支持，人机系统集成，系统系（SoS）生命周期过程和分析支持。从那以后四年过去，一个全新的UAF1.0成为官方的对象管理组（OMG）标准，即将成为ISO/IEC19517标准。这个新的架构框架（AF）到底是什么？它与其他架构框架有什么不同？它只是已有框架的综合体？或者它是一种现代技术，结合了旧的和现代的最佳实践？

# 本文探讨了UAF，回答了上述基本问题，并特别关注以下问题：

# 1. 适用于解决UAF的哪些问题？

# 2. UAF是北约(NATO)架构框架（NAF）的替代品吗？

# 3. UAF如何与基于模型的系统工程（MBSE）关联？

# 通过将现实世界的UAF应用案例提供给各个行业领域的系统系架构，本文还回答了更多的问题。

# 介绍

应用统一架构框架，使用MBSE方法将架构建模工作转移到SE的一个组成部分（Hause等，2016）。

这有助于系统集成商开发可互操作的系统，可追溯到需求并跨视图，使用一个集成的架构模型，可以进行影响分析，差距分析，权衡分析，模拟和工程分析（Morkevicius等，2017b）。此外，UAF的范围扩展到防御架构之外。它被通用化以适用于任何领域的系统的架构系统。 对UAF的强制性要求之一是“支持国防，工业和政府组织的架构建模”。作为对此要求的回应，UAF 1.0版是行业领域不可知的（OMG，2013）。它的目标是系统系的建模，包括企业和物联网系统。

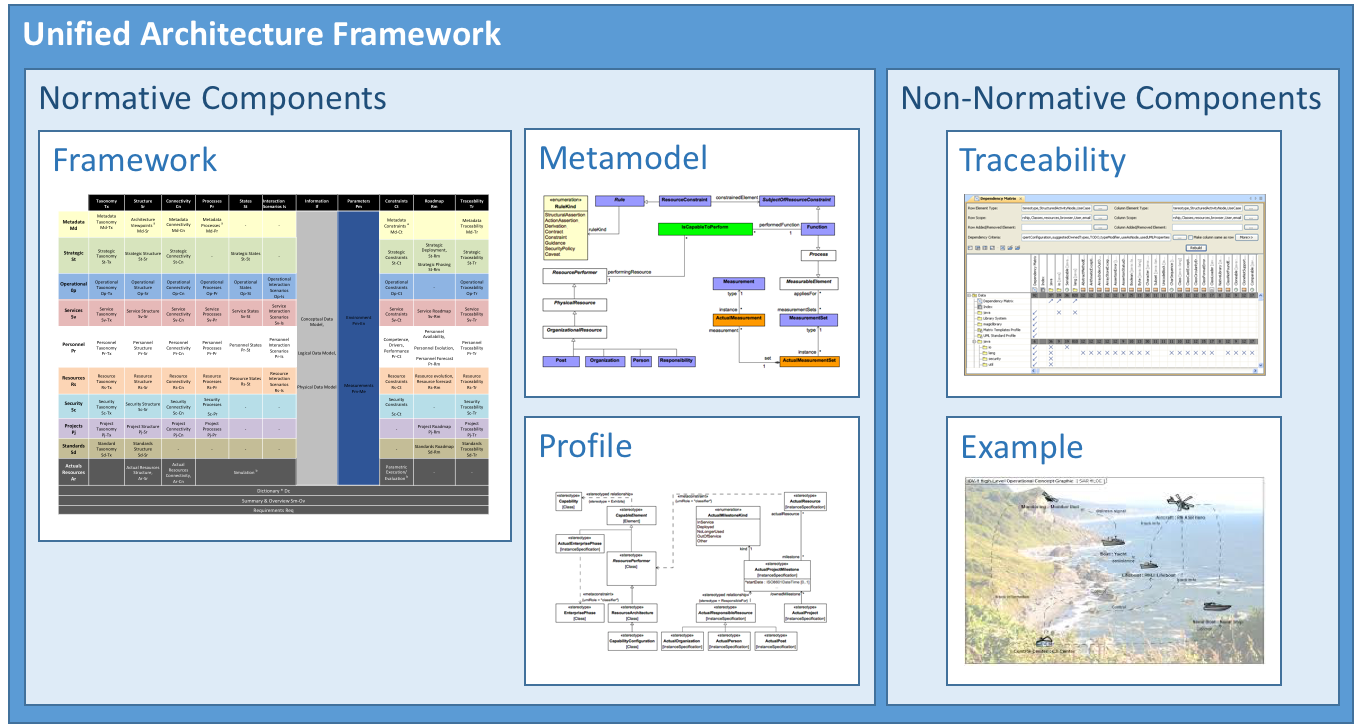
为何选择UAF？从以文档为中心的系统工程方法到基于模型的系统工程（MBSE）的范式转变揭示了MBSE方法演进的差距，其中之一就是针对系统系(SoS)建模没有标准化方法可用。系统建模语言（OMG，2015）是最终解决方案的信念被证明是错误的。根据定义建模语言提供语法和语义，但不提供语用。要成功应用SysML等语言，必须回答各种问题，包括如何构建模型，构建哪些视图，传递哪些工件以及按什么顺序传递。每家公司以不同方式处理这些问题。不遵守标准化方法的组织最终会使用具有不同视图集的不同结构化模型，从而导致模型之间交换数据的能力丧失，与其他团队沟通的能力丧失，工具定制的开销以及对具体的培训。此外，模型变得无法整合和重用（Morkevicius等，2017a）。

# UAF

UAF由三个主要部分组成（所有UAF组件如图1所示）：

* 框架(framework) - 域、模型类型和视点的集合，
* 元模型(metamodel) - 用于根据特定视点构造视图的类型、元组和个体的集合，
* 概要文件(profile) - 基于SysML实现元模型，以在构建视图时应用基于模型的系统工程原理和最佳实践。

两个补充组成部分是：（i）针对其他已有的EAF和UPDM的可追溯性指南; （ii）基于搜救案例研究的示例模型。



*图 1 UAF 框架模组*

以下段落详细介绍了UAF的主要组成部分。

*框架(Framework)*

Grid格式最适合描述UAF的框架组件。它被组织成行和列，其中行是域(Domain)，列是模型种类(Model Kinds)。行和列的交集称为视点(Viewpoint)。 UAF网格总结了现有AF中可用的所有视点。它通过仅选择特定上下文所需的视点，作为构建特定于域的框架的基础。

UAF由12个域组成（OMG，2017）：

* 元数据 - 捕获与整个架构相关的元数据，例如：原则，元模型扩展，要构建的视图，开发体系结构的过程，对外部体系结构的引用，库等。
* 战略 - 描述能力分类，组成，依赖和演变。
* 操作 - 描述支持（展示）能力所需的需求，操作行为，结构和交换。以独立于实现/解决方案的方式定义所有操作元素。
* 服务 - 显示服务规范以及展示能力或支持运营活动所需的这些规范所需和提供的服务级别。
* 人员 - 使您能够了解系统/企业架构中的人类角色。它通过提供从工程界到人力，人员，培训和人为因素社区的结构化联系，为利益相关者的决策提供基础。
* 资源 - 捕获由资源组成的解决方案体系结构，例如：组织，软件，工件，功能配置，实现操作要求的自然资源。通常在SysML或UML中详细描述资源的进一步设计。
* 安全性 - 说明解决特定安全问题所需的安全资产，安全限制，安全控制，系列和措施。
* 项目 - 描述项目和项目里程碑，这些项目如何提供能力，为项目做出贡献的组织以及项目之间的依赖关系。
* 标准 - 显示适用于该体系结构的技术，操作和业务标准。
* 实际资源 - 说明预期或实现的单个资源配置以及它们之间的实际关系。
* 词典 - 提供体系结构中所有元素的定义。
* 摘要和概述 - 以一致的形式提供执行级­­­别的摘要信息，以便快速参考和比较架构描述。



**Metadata**

**Md**

**Taxonomy**

**Tx**

Metadata Taxonomy Md-Tx

**Structure Connectivity Processes States Interaction Information Parameters Constraints Roadmap Traceability**

**Sr Cn Pr St Scenarios Is If Pm Ct Rm Tr**

Architecture

a

Metadata

Connectivity Md-Cn

Metadata

a

Viewpoints

Md-Sr

Processes

Md-Pr

-

-

Metadata Constraints a

Md-Ct

Metadata

Traceability Md-Tr

**Strategic**

**St**

Strategic

Taxonomy St-Tx

Strategic Structure

St-Sr

Strategic

Connectivity St-Cn

-

Strategic States

St-St

-

Strategic

Constraints St-Ct

Strategic

Deployment, St-Rm

Strategic Phasing St-Rm

Strategic

Traceability St-Tr

**Operational** Operational Operational Operational Operational Operational

**0p** Taxonomy Structure Connectivity Processes States Op-Tx Op-Sr Op-Cn Op-Pr Op-St

**Services**

**Sv**

Service Structure

Sv-Sr

Service States

Sv-St

Operational

Interaction Scenarios Op-Is

Service Interaction Scenarios Sv-Is

Operational

Constraints Op-Ct

-

Operational

Traceability Op-Tr

Service

Taxonomy Sv-Tx

Service Service

Connectivity Processes

Sv-Cn Sv-Pr

Service

Constraints Sv-Ct

Service Roadmap

Sv-Rm

Conceptual Data

Model,

Environment

Pm-En

Service

Traceability Sv-Tr

Personnel

Availability,

Personnel **Personnel** Personnel Personnel Personnel Personnel Personnel States Interaction **Pr** Taxonomy Structure Connectivity Processes Pr-St Scenarios

Pr-Tx Pr-Sr Pr-Cn Pr-Pr Pr-Is

Competence,

Drivers, Performance Pr-Ct

Personnel Evolution,

Logical Data Model,

Personnel

Traceability Pr-Tr

Personnel Forecast

Pr-Rm

**Resources** Resource Resource Resource

**Rs** Taxonomy Structure Connectivity Rs-Tx Rs-Sr Rs-Cn

Resource

Processes Rs-Pr

Resource States

Rs-St

Resource

Interaction Scenarios Rs-Is

Physical Data Model

Measurements

Pm-Me

Resource

Constraints Rs-Ct

Resource evolution, Resource

Resource forecast Traceability Rs-Rm Rs-Tr

**Security**

**Sc**

Security

Taxonomy Sc-Tx

Security Structure

Sc-Sr

Security

Connectivity Sc-Cn

Security

Processes Sc-Pr

Project Processes Pj-Pr

-

-

Security

Constraints

Sc-Ct

-

Security

Traceability Sc-Tr

**Projects**

**Pj**

Project

Taxonomy Pj-Tx

Standard Taxonomy Sd-Tx

Project Structure

Pj-Sr

Project

Connectivity Pj-Cn

-

-

-

Project Roadmap

Pj-Rm

**Standards**

**Sd**

Standards

Structure Sd-Sr

-

-

-

-

-

Project

Traceability Pj-Tr

Standards Roadmap Standards Sd-Rm Traceability

Sd-Tr

**Actuals**

**Resources Ar**

Actual Resources

Structure, Ar-Sr

Actual

Resources Connectivity, Ar-Cn

Simulation b

Parametric

Execution/ Evaluation b

-

-

Dictionary \* Dc

Summary & Overview Sm-Ov Requirements Req

*图 2 UAF 宫格*

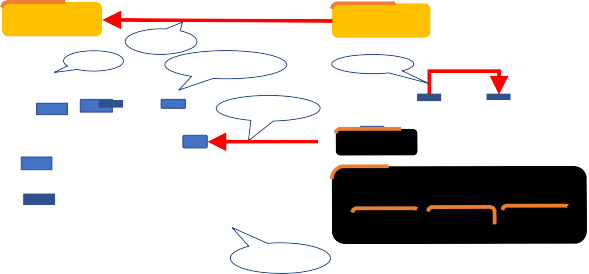
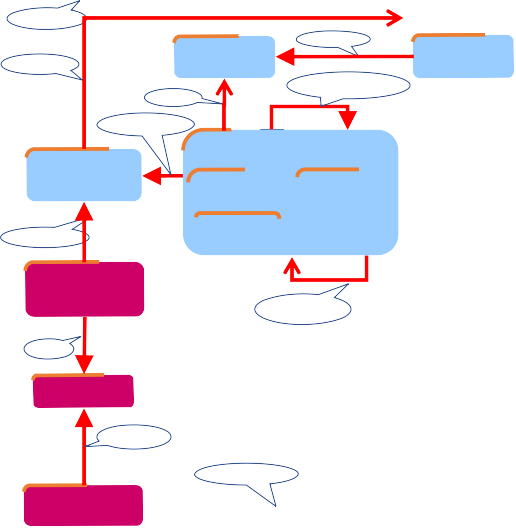
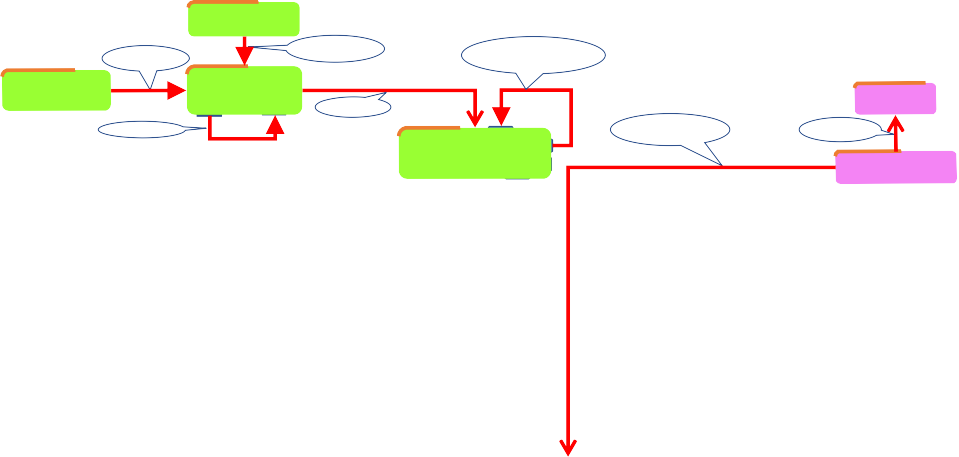
UAF由12种模型组成（OMG，2017）：

* 分类 - 定义类型的层次结构，例如Projects Taxonomy视图用于定义Project类型的层次结构。
* 结构 - 定义类型或个体的组成（部分与整体的关系），例如资源结构视点用于定义由人力和系统资源组成的能力配置。
* 连通性 - 定义类型或个人之间的接口，例如操作连接用于捕获在执行者之间交换信息的需要。
* 流程 - 捕获基于活动的行为和流程，例如操作流程描述了在实现能力的过程中通常进行的活动。
* 状态 - 捕获基于状态的行为，例如：资源状态捕获资源的状态以及该资源如何响应各种事件和操作。
* 交互方案 - 提供类型之间交互的按时间顺序检查，例如运营互动情景表示由于特定运营情景而对运营交换进行时间有序检查。
* 信息 - 解决有关运营，服务，资源，安全性和其他架构的信息视角。
* 参数 - 显示特定域中某些元素的可测量属性以及相关的条件和环境。
* 约束 - 捕获不同建筑元素的约束，例如服务约束指定传统的文本服务策略，这些策略是对服务规范实现方式的约束。
* 路线图 - 关注架构随时间的变化，例如：项目路线图提供了项目的时间表视角。
* 可追溯性 - 定义不同域中概念之间的可追溯性，例如：标准可追溯性显示了标准对架构中特定元素的适用性。
* 需求 - 用于表示彼此之间以及不同域的UAF体系结构元素的需求，属性和关系（跟踪，验证，满足，细化）。

行和列之间的交集称为视点或视图规范。 这是一种构建视图的方法。 视图是视点的实例（ISO，2011）。 每个视图都可以用不同的演示文稿表示，例如图表，表格，矩阵，图形，图表等.UAF规范为每个视点提供了关于演示文稿的建议。 在本文的建模语言部分中更详细地描述了推荐的演示文稿。

*Metamodel元模型*

根据视点组织UAF域元模型（DMM）。 因此，很容易理解哪些元素（包括类型，个体和元组）可用于构建特定视图。 元素分类为类型，个体和元组是从IDEAS中获取的。 通常，UAF元模型是复杂的4D IDEAS本体的简化版本（IDEAS，2012）。 虽然它被简化，但与大多数现有的企业建模语言和方法相比仍然很强大。



**Enterprise vision**

**Mitigates**

**Is a goal**

**of**

**Is a vision**

**for**

**Depends on/**

**specializes/ contains**

**Security control**

**Risk**

**Affects resource and**

**operational elements**

**Enterprise goal**

**Actual enterprise**

**phase**

**Exhibits**

**Project**

**actual**

**Requirement**

**Contains**

**Provides status**

**for**

**Is part of**

**Links to all**

**elements**

**Capability**

**Maps to**

**Actual Project**

**milestone**

**Standard**

**Defines**

**Operational port**

**Operational**

**interface**

**Owned**

**by**

**Data model**

**Maps to**

**Owned**

**by**

**Links to all**

**elements**

**Operational**

**exchanges**

**Has**

**Data element**

**Information**

**element**

**Measurement**

**Performs**

**Exhibits**

***Operational agent***

**Links to all**

**elements**

**Operational**

**activity**

**Contains**

**Contains**

**Operational**

**performer**

**Known resource**

**Operational**

**architecture**

**Resource port**

**Resource**

**interface**

**Implements**

**Defines**

**Consumes**

**Has**

**Resource**

**exchanges**

**Contains**

**Service**

**specification**

**Classified by**

Actual resource

**Specializes**

**/ contains**

**Implements**

*Resource performer*

**Known Resource architecture**

**resource**

***Organisational***

***resource***

**Fielded**

**capability**

**Has**

**Performs**

***Actual organisational resource***

**Function**

**System**

**Capability configuration**

**Security**

**enclave**

**Responsibility**

**Service port**

**Person**

**Actual**

**person**

**Actual post**

**Actual**

**otganisation**

***Physical resource***

**Organization**

**Defines**

**Resource**

**artefact**

**Software**

**Natural**

**resource**

**Post**

**Specializes/**

**contains**

**implements**

**Service interface**

*图 3 UAF 元模型*

UAF元模型使用颜色编码来区分元组，类型，抽象类型和个体。 阅读元模型时非常有用。 然而，上面的图像描绘了不同的颜色编码。 元素颜色反映网格中的行颜色。 它简化了元模型的外观，显示了UAF元模型中定义的主要概念和关系。

*Profile*

在UML的上下文中，Profile定义了对引用元模型的有限扩展，目的是使元模型适应特定的平台或域。 UAF配置文件定义了UML扩展以支持UAF元模型。它还依赖于SysML配置文件，这是UML的另一个扩展。对SysML的依赖是以继承关系的形式。此继承的目的是继承SysML图形符号和适用于SysML的工程分析技术（例如参数分析）。

UAF简档（UAFP）根据视点组织，与UAF元模型完全相同。通过此组织，可以轻松了解可以使用哪些元素来构建特定视图。除了元模型之外，每个profile元素都扩展了特定的UML元类，并从特定的SysML构造型继承（例如，Operational Performer扩展了UML类并继承自SysML块）。 UAF配置文件还定义了元约束。例如，映射到功能关系只能将活动连接到功能。 UAF元模型实现作为UML配置文件提供了几个主要优点，包括：

* fUML标准可用于执行UAF体系结构
* SysML参数可用于对UAF模型进行分析工程研究
* UAF与SysML，UML和其他OMG标准配置文件完美集成，以确保从系统模型的高级系统到系统和软件工程模型的可追溯性
* UAF易于扩展

上面列出的要点强调UAF概要文件是UAF元模型推荐的基于标准的实现。

*Modeling Language*

UAF元模型不定义图形表示法。系统建模语言（SysML）是以图形方式表示UAF元模型元素的标准方法。 SysML定义了9种图表类型，以及所有SysML元素的表示法（OMG，2015）。因为UAF配置文件是SysML的扩展，所以它继承了相同的表示法。例如，UAF Capability继承自SysML块，这意味着UAF Capability继承了SysML块表示法。

UAF规范提供了应该使用SysML图类型来创建特定UAF视图的指南。重要的是要注意它只是一个指南。也可以使用其他建模语言（例如UML，BPMN等）。除了SysML图表，建议以表格，矩阵或图表格式表示一些UAF视图。

UAFP作为SysML和UML的扩展，支持标准的OMG交换格式（用于模型数据交换的XMI标准和用于图交换的DDI）。此外，UAF支持视图和视点的概念，其中视点指定用于动态地从模型数据构建特定视图的方法。

## Transition path from UPDM to UAFP

UAFP向后兼容UPDM。 UAF的可追溯性组件中提供了兼容性矩阵。 总之，UAFP和UPDM非常相似。 主要区别包括：服务领域; 更好地与SysML集成; 测量; 时间范围; 和期望的效果。 由于UAF支持的新域名，UAFP还有更多的建模结构。

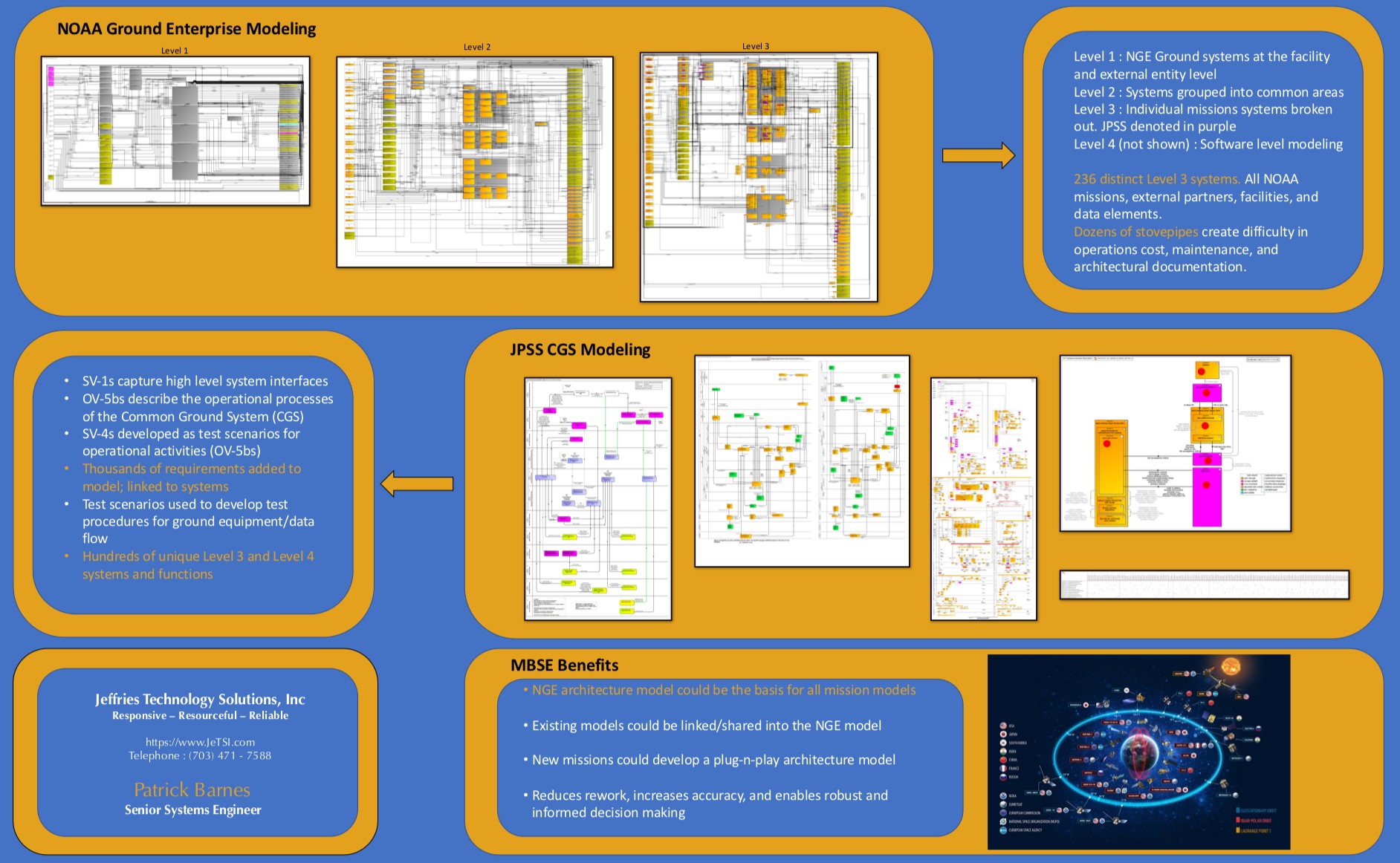
# Application Use Cases

UAF已经有很多应用。国防工业最常被认为是;然而，其他行业的UAF应用非常值得关注。

国家环境卫星，数据和信息服务（NESDIS）地面企业和NASA的联合极地卫星系统（JPSS）地面项目都是Jeffries Technology Solutions，Inc。的系统架构示例的成功系统。两者都是在DoDAF中使用UPDM元模型构建的（UAFP的前身）。图4展示了三个级别的NOAA建模。 1级是设施和外部实体层面的NGE地面系统，2级包括分为公共区域的系统，3级显示分解的各个任务系统。未显示级别4并将其用作软件建模。就UAF而言，图4中的所有图表都描述了资源连接视图。经验教训是，MBSE方法减少了返工，提高了准确性，并实现了强有力的知情决策（Barnes，2018）。

沃尔沃建筑设备的采石场。沃尔沃建筑设备的采石场。如果目标是设计完整的现场解决方案，那么大型，复杂，重型建筑设备的开发可能是困难，耗时且昂贵的，甚至更大。这个例子取自沃尔沃建筑设备公司正在进行的一个名为Electric Site的实际项目。目的是使采石场的运输阶段通电，从挖掘到初级破碎和运输再到二次破碎。这将使二氧化碳减少95％，总运营成本减少25％。 （Kihlström，2018），（Sjöberg，2017）描述了UAF模型如何用于显着影响持续的系统工程工作，以及应用程序开发的软件架构。企业架构模型专门用于确保网站的整体管理。

其他公开案例研究是Tactical Lockheed Martin空中客车直升机（Mitchell，2010），空中客车直升机（Wirtz等，2017），物联网架构的UAF（Morkevicius等，2017）



*Figure 4 NOAA Ground Enterprise Modeling*

# UAF vs. NAF vs. DoDAF vs. TOGAF vs. Archimate

UAF框架是NAF和DoDAF的替代方案。但是，该声明仅涉及UAF框架组件。其他组件补充DoDAF和NAF。例如，UAF元模型和配置文件与NAF和DoDAF兼容。根据用于创建DoDAF或NAF视图的工具，可以使用UAF元模型或配置文件。使用相同的元模型创建UAF，NAF和DoDAF视图可实现框架之间的互操作性。由工具供应商提供转换机制以将DoDAF OV-2转换为NAF L2或UAF操作结构。 UAF命名方案比其他框架提供的方案更直观。因此，有一些例子，其中UAF用于开发架构，架构在交付之前就转换为DoDAF。

UAF与TOGAF的兼容性不同，值得单独讨论。 UAF元模型是TOGAF内容元模型的替代方案。这意味着可以使用遵循TOGAF体系结构开发方法（ADM）的UAF元模型轻松创建TOGAF体系结构。 TOGAF ADM还可以与UAF元模型结合使用来构建UAF视图。这种组合弥补了UAF架构开发方法的不足。

UAF元模型和Archimate都被北约3C委员会命名为创建NAF视图的官方元模型（Ristani，2018）。一方面，它们是竞争的替代品。另一方面，北约要求OMG和Open Group找到一种方法来支持两者之间的互操作性。最初的想法是在OMG UAF小组中开发的，Archimate可以如何与UAF合作，即使用Archimate开发高级能力架构和UAF来创建丰富的运营和资源架构。

轻量级

模型工程

为我所用的架构模型

高级

模型工程

**ArchiMate 元模型 – Flat Models**

**Shared**

**Building Blocks**

**UAF Meta-Model – Modular Models**

通过轻松捕获企业资产来

分析和沟通随着时间的推移企业架构出现的变化及问题

* Capabilities Structured assembly of building blocks for
* Applications advanced enterprise transformation and
* Infrastructures system engineering over multiple time
* Projects scales.

**UAF Foundations**

# UAF to NAF v4 mapping

*Figure 5 UAF vs. Archimate*

一个自然要考虑的问题是如何用UAF模拟NAF。 UAFP是对NAF架构进行建模的完美工具。 需要了解的最重要的事情之一是需要使用哪些UAFP概念来构建特定的NAF视图。 该信息可在UAF规范中获得; 但是，这不是直截了当的。 每个UAF观点都有一个定义的元模型。 NAF建模者需要做的就是查看UAF到NAF的映射表（图

6）并找到相应的UAF视图。 例如，NAF L2对应于UAF运营结构。

Th

e U

AF/

P Gri

d

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Taxonomy Tx** | **Structure Sr** | **Connectivity Cn** | **Processes Pr** | **States St** | **Interaction Scenarios Is** | **Information If** | **Parameters Pm** | **Constraints Ct** | **Roadmap Rm** | **Traceability Tr** |
| **Metadata Md** | Metadata Definitions **A1** | Architecture Products **A2** | Architecture Correspondence **A3** | Methodology Used **A4** | **-** | - |  |  | - | - | - |
| **Strategic St** | Capability Taxonomy **C1** | Enterprise Vision  **C2** | Capability Dependencies **C3** | Standard Processes **C4-** | Effects  **C5** | - | Logical Data Model  **L7**  Physical Data Model  **P7** | Performance Parameters **C7** |  |  |  |
| Planning Assumptions **C8** | Capability Roadmap  **Cr** | - |
| **Operational 0p** | Node Types  **L1** | Logical Scenario  **L2, L2-L3** | Node Interactions **L3, L2-L3** | Logical Activities  **L4** | Logical States  **L5** | Logical Sequence **L6** | Logical Constraints **L8** | - | - |
| **Services Sv** | Service Taxonomy **S1** | - | Service Interfaces **S3, S7** | Service Functions **S4** | Service States  **S5** | Service Interactions **S6** | Service Policy  **S8** | Service Roadmap  **Sr** | Service Traceability **C1-S1** |
| **Personnel Pr** | Resource Types  **P1** | Resource Structure **P2** | Resource Connectivity **P3** | Resource Functions **P4** | Resource States  **P5** | Resource Sequence **P6** | - | - | - |
| **Resources Rs** | Resource Constraints **P8** | Configuration Management **Pr** | Resource Traceability **L4-P4** |
| **Security Sc** | - | - | - | - | - | - | - | - | - |
| **Projects Pj** | - | - | - | - | - | - | - | Lines of Development **Lr** | - |
| **Standards Sd** | - | - | - | - | - | - | - | Standards  **A8** | |
| **Actuals Resources Ar** |  | Resource Structure P2 | - | - | | | - | - - | |

*Figure 6 NAF v4 to UAF mapping*

Dictionary -

Summary & Overview **A5, A6** Requirements -

1

# Summary and Conclusions总结和结论

UAF是系统，系统系统和企业工程中使用的最佳实践的集合，已有30多年的历史。 此外，UAF适用于任何领域。 它仍然是构建DoDAF，NAF和MODAF架构的完美选择。

UAFP简介是建立UAF模型的正式方式。 这种正式的背景允许用户构建精确的可执行体系结构，执行自动权衡分析，运行假设情景，验证需求，并为系统和软件模型添加可追溯性。

UAF，UAFP及其前身UPDM的应用已在不同领域得到实际验证。 有多个公开可用的案例研究证明UAF是构建成熟和丰富的系统架构系统的首选。

# Bibliography

[1] Barnes, P. (2018), Utilizing MBSE to Modularly Architect the NESDIS Ground Enterprise. 14th Annual Symposium on New Generation Operational Environmental Satellite Systems, available at <http://jetsi.com/wp-content/uploads/2018/01/AMS2018Poster_Final.pdf>

[2] Bleakley, G. and Morkevicius, A. (2016), Transitioning from UPDM to the Unified Architecture Framework (UAF). Integrated Enterprise Architecture 2016.

[3] Hause, M., Bleakley, G. and Morkevicius, A. (2016), Technology Update on the Unified Architecture Framework (UAF). INCOSE International Symposium, 26: 1145–1160.

[4] Hause, M., Bleakley, G. and Morkevicius, A. (2017), Technology Update on the Unified Architecture Framework (UAF). INSIGHT, 20: 71-78.

[5] IDEAS Group. (2007), *The IDEAS Model. URL:* [http://www.ideasgroup.org/foundation/,](http://www.ideasgroup.org/foundation/) Accessed May 2018.

[6] ISO/IEC/IEEE 42010:2011 - Systems and software engineering - Architecture description. Iso.org. 2011-11-

24. Accessed 2018.

[7] Kihlström, L. (2018), How to use Architecture frameworks to speed up systems development. UAF & MBSE tutorials, Reston. URL: https://www.omg.org/cgi-bin/doc?omg/2018-03-11.pdf, Accessed May 2018.

[8] Mitchell, S. M. (2010), Efficient Management of Configurations in the Model-Based System Development of a Common Submarine Combat System, AFCEA-GMU Critical Issues in C4I Symposium.

[9] Morkevicius, A., Aleksandraviciene, A., Mazeika, D., Bisikirskiene, L. and Strolia, Z. (2017a), MBSE Grid: A Simplified SysML-Based Approach for Modeling Complex Systems. INCOSE International Symposium, 27: 136-150.

[10] Morkevicius, A., Bisikirskiene, L., and Bleakley, G. (2017b), Using a systems of systems modeling approach for developing Industrial Internet of Things applications. 12th System of Systems Engineering Conference (SoSE), Waikoloa, HI: pp. 1-6.

[11] Ristani, K. (2018), A NATO ACaT Perspective on the Use of Architecture and Standards. UAF & MBSE Summit, Reston. URL: https://www.omg.org/cgi-bin/doc?omg/2018-03-03.pdf, Accessed May 2018. [12] Object Management Group (OMG). (2015), OMG Systems Modeling Language (OMG SysML™), V1.4,

OMG Document Number: formal/2015-06-03, URL: [http://www.omg.org/spec/SysML/1.4/PDF,](http://www.omg.org/spec/SysML/1.4/PDF) Accessed October, 2016.

[13] Object Management Group (OMG). (2017), Unified Architecture Framework (UAF) 1.0, URL: https://[www.omg.org/spec/UAF/1.0/,](http://www.omg.org/spec/UAF/1.0/) Accessed May 2018.

[14] Object Management Group (OMG). (2009), Unified Profile for the Department of Defense Architecture Framework (DoDAF) and the Ministry of Defence Architecture Framework (MODAF), available at [http://www.omg.org/spec/UPDM/.](http://www.omg.org/spec/UPDM/)

[15] Object Management Group (OMG). (2013), Unified Profile for DoDAF/MODAF (UPDM) 3.0 Request for Proposal), available at http://www.omg.org/cgi-bin/doc.cgi?c4i/2013-9-11.

[16] Okon, W. (2012), Moving Towards an Unified Architecture and the US Government Information Sharing Environment, 1105 Enterprise Architecture Conference.

[17] Sjöberg, P., Kihlström, L. and Hause, M. (2017), An industrial example of using Enterprise Architecture to

speed up systems development. INCOSE International Symposium, 27: 401-417.

[18] Wirtz, J., Paper, P., Poussard, S. (2017). Introduction of MBSE at Airbus Helicopters. UPDM & MBSE tutorials, Brussels. URL: [https://www.omg.org/cgi-bin/doc?c4i/17-06-08.pdf. Accessed May 2018](https://www.omg.org/cgi-bin/doc?c4i/17-06-08.pdf.%20Accessed%20May%202018).

[1] Barnes，P。（2018），利用MBSE模块化地构建了NESDIS地面企业。第14届新一代运行环境卫星系统年度研讨会，见http://jetsi.com/wp-content/uploads/2018/01/AMS2018Poster\_Final.pdf

[2] Bleakley，G。和Morkevicius，A。（2016），从UPDM过渡到统一架构框架（UAF）。 2016年集成企业架构。

[3] Hause，M.，Bleakley，G。和Morkevicius，A。（2016），统一架构框架（UAF）的技术更新。 INCOSE国际研讨会，26：1145-1160。

[4] Hause，M.，Bleakley，G。和Morkevicius，A。（2017），统一架构框架（UAF）的技术更新。洞察，20：71-78。

[5] IDEAS集团。 （2007），IDEAS模型。网址：http：//www.ideasgroup.org/foundation/,2011年5月访问。

[6] ISO / IEC / IEEE 42010：2011 - 系统和软件工程 - 架构描述。 Iso.org。 2011-11-

24. 2018年访问。

[7]Kihlström，L。（2018），如何使用架构框架来加速系统开发。 UAF和MBSE教程，Reston。网址：https：//www.omg.org/cgi-bin/doc？tom / 201-03-11.pdf，2018年5月访问。

[8] Mitchell，S。M.（2010），基于模型的共同潜艇作战系统系统开发中的配置的有效管理，AFCEA-GMU C4I研讨会中的关键问题。

[9] Morkevicius，A.，Aleksandraviciene，A.，Mazeika，D.，Bisikirskiene，L。和Strolia，Z。（2017a），MBSE Grid：一种简化的基于SysML的复杂系统建模方法。 INCOSE国际研讨会，27：136-150。

[10] Morkevicius，A.，Bisikirskiene，L。和Bleakley，G。（2017b），使用系统建模方法开发工业物联网应用。第12届系统工程系统会议（SoSE），威可洛亚，HI：第1-6页。

[11] Ristani，K。（2018年），北约对建筑和标准使用的看法。 UAF和MBSE峰会，Reston。网址：https：//www.omg.org/cgi-bin/doc？tom / 201-03-03.pdf，2018年5月访问。[12]对象管理组（OMG）。 （2015），OMG系统建模语言（OMG SysML™），V1.4，

OMG文件编号：formal / 2015-06-03，URL：http：//www.omg.org/spec/SysML/1.4/PDF，2016年10月访问。

[13]对象管理组（OMG）。 （2017），统一架构框架（UAF）1.0，URL：https：//www.omg.org/spec/UAF/1.0/,2011年5月访问。

[14]对象管理组（OMG）。 （2009年），国防部体系结构框架统一简介（DoDAF）和国防部体系结构框架（MODAF），可从http://www.omg.org/spec/UPDM/获取。

[15]对象管理组（OMG）。 （2013），DoDAF / MODAF（UPDM）3.0提案申请统一简介），见http://www.omg.org/cgi-bin/doc.cgi?c4i/2013-9-11。

[16] Okon，W。（2012），迈向统一架构和美国政府信息共享环境，1105企业架构会议。

[17]Sjöberg，P.，Kihlström，L。和Hause，M。（2017），使用企业架构的工业实例

加快系统开发。 INCOSE国际研讨会，27：401-417。

[18] Wirtz，J.，Paper，P.，Poussard，S。（2017）。在空客直升机上引入MBSE。 UPDM和MBSE教程，布鲁塞尔。网址：https：//www.omg.org/cgi-bin/doc？c4i / 17-06-08.pdf。 2018年5月访问。