Document No.: **PUB000849**
Revision: **-**
Cage Code: **07703**
Date： **TBD**
Contract No.: NNM07AA75C
DRD/CDRL: N/A
WBS No.: **N/A**

Closed Loop Requirements and Analysis Management

Michael Lamoreaux	PLM Systems Engineer Subject Matter Expert	Preparer
Brett Verhoef	PLM Analysis Subject Matter Expert	Preparer
Paul Nelson	PLM Architect	Reviewer
Jon Jarrett	PLM Manager	Reviewer
Ernie Kyed	Systems Engineering Manager	Reviewer

Abstract.

Effective systems engineering involves the use of analysis in the derivation of requirements and verification of designs against those requirements. The initial development of requirements often depends on analysis for the technical definition of specific aspects of a product. Following the allocation of system-level requirements to a product's components, the closure of those requirements often involves analytical approaches to verify that the requirement criteria have been satisfied. Meanwhile, changes that occur in between these two processes need to be managed in order to achieve a closed-loop requirement derivation/verification process.

Herein are presented concepts for employing emerging Teamcenter capabilities to jointly manage requirements and analysis data such that analytical techniques are utilized to effectively derive and allocate requirements, analyses are consulted and updated during the change evaluation processes, and analyses are leveraged during the design verification process. Recommendations on concept validation case studies are also discussed.

Table of Contents

1. Introduction

Orbital ATK employs continuous improvement processes to evolve its engineering practices and tools. Propulsion Systems is leading the evolution for consolidating several engineering processes into Teamcenter (Teamcenter Unified Architecture or TcUA). Two aspects of that overall consolidation effort are requirements and verification data management processes currently performed in Teamcenter Systems Engineering (TcSE) and analysis data management. Solution architectures are being developed and tested for each of these two aspects along with architecture for integrating requirements and analysis data management with each other and with the existing computer-aided design (CAD)/design architecture. The integration of requirements and analysis data in TcUA provides many benefits that are not available with the current processes. These include the following:

a. Connections between requirements data and analysis data enable integrated change management processes.
b. Analyst can directly access the requirements driving their analysis.
c. Analyst can directly provide compliance data into compliance reporting objects helping minimize data handoffs.
d. Systems engineers can directly access the analysis data.
e. Design engineers can directly access requirements and analysis data for integration with part design data.

This paper presents information about the solution architecture for the integration of requirements and analysis data management and the stakeholder needs driving that solution. The role this plays in the overall Orbital ATK Produce Lifecycle Management (PLM) vision is discussed. This paper also describes the concept feasibility testing that will be used to validate the basic solution.

2. Orbital ATK PLM Strategic Vision

The overall vision for Orbital ATK PLM is illustrated in **Figure 1**. It centers around the premise that data generated during one phase of a project is related to data generated in other phases and that these data should be levereaged and reused in order to achieve improved quality and reduced cost. Another central aspect of the Orbital ATK PLM strategic vision is the premise that managing all of this data in a single system facilitates full usage of that data and enables better integrated change management processes.

Two important aspects of this overall vision are management of requirements and verification data and management of analysis and simulation data. These two aspects, although they are often treated separately, are interwined and require coordinated processes and tools in order to maximize management effectiveness.

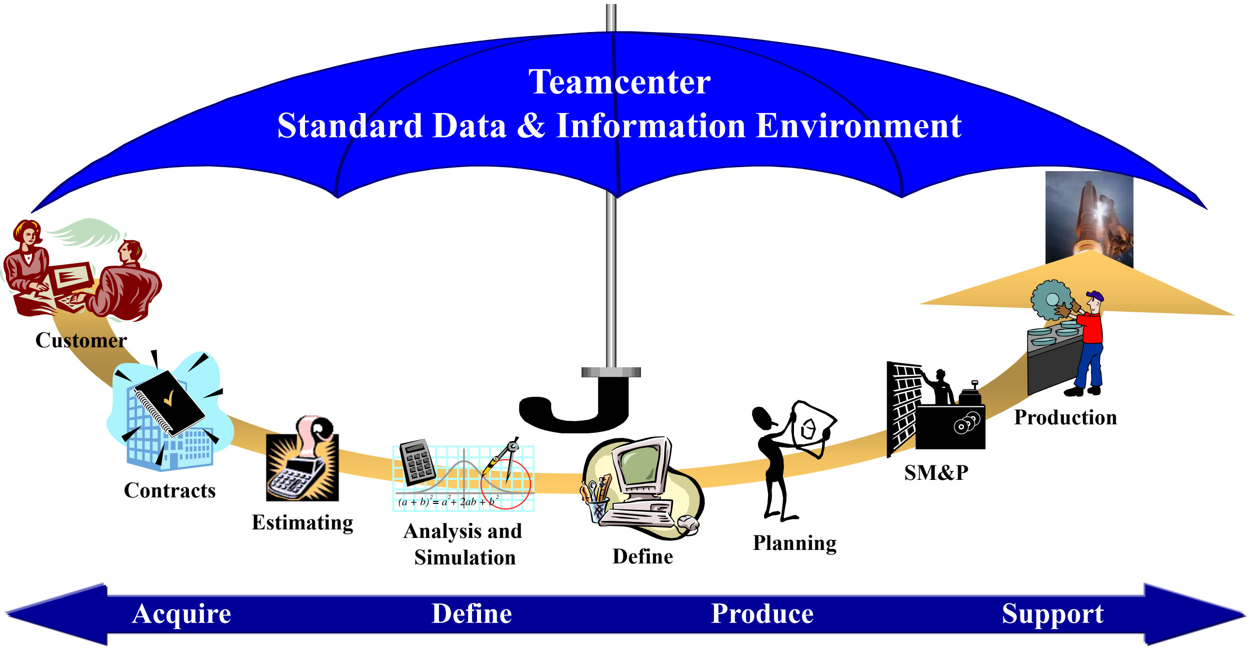


Figure 1. Orbital ATK PLM Vision

3. Interactions Between Requirements and Analysis

There are several areas of interaction between requirements and analysis that are being evaluated in developing an effective data management strategy. These are categorized and discussed below.

3.1 Concept Development

During the concept development and proposal process, there are often high-level sizing and performance analyses to help determine the basic performance requirements for the system, rough estimates for size and mass, and the costs and risks with the project. These analyses are similar to requirement derivation analyses and preliminary design verification analyses. They should be captured in connection with the preliminary derived requirements and later leveraged during the requirement derivation and design verification processes.

3.2 Requirement Derivation

During the process of deriving requirements, it is often necessary to perform analyses to determine a set of properly derived requirements. This can take place at several different levels including analysis to derive stakeholder needs into system requirements, derive system requirements into a coordinated set of subsystem requirements, and so-on down to the component or piece part levels. These derivation efforts can represent a preliminary cut at a verification analysis that should be leveraged during the verification process.

Requirement derivation analyses may include some of the following:

a. Hardware sizing structural analyses
b. Mass predictions (related to the preliminary structural analyses)
c. Dimensional tolerance analyses
d. Timing tolerance analyses
e. Loads and dynamic environment determination analyses
f. Thermal environment determination analyses
g. Ballistics analyses
h. Reliability analyses
i. Hardware separation dynamics analyses
j. Electrical integration analyses
k. Design trade study analyses

3.3 Design Verification

Data from analyses are essential to enable systems engineers to ensure the system design meets its requirements. This comes both in the provision of a pass/fail indication as well as documentation of the analysis process itself. Feeding this information directly into the design verification effort enables the review and oversight processes to take place more efficiently.

3.4 Analysis Scope Definition

Analyses are less expensive and more effective when the scopes of the analyses are understood up front. This allows the analyses to be setup and executed right the first time. The set of requirements, verification requirements, and verification objectives assigned to an analysis play a significant role in defining the scope and determining the type of analysis to be performed. Adding or changing requirements can result in additional iterations of analyses, which may become expensive while adding little value to the program.

3.5 Change Management

3.5.1 Requirement Change Management

While requirement changes after verification has taken place are typically considered inefficiencies, some level of requirement volatility is unavoidable. When requirement changes cannot be avoided, proper management of the requirement/verification data allows for a more effective analysis update process. By easily retrieving the existing analysis, data can be utilized to evaluate whether the analysis needs to be updated at all. This helps ensure that all essential analysis updates are performed without any unnecessary updates. Also, when repeating analyses is necessary, effective analysis data management techniques allow the previous analyses and analysis data to be leveraged for evaluation of the requirement change rather than beginning the analysis from scratch. This improves both the cost effectiveness and quality of the analysis.

3.5.2 Design Change Management

Design changes, like requirement changes, are often unavoidable. They can be triggered by material obsolescence, design improvement opportunities, cost reduction initiatives, changes to interfacing systems or hardware, etc. Sometimes changes to one design can affect analyses for other hardware that use the modified design as an input. For example, changes to a case design may result in the need to update the case structural analysis but may also drive the need to update analyses that determine the temperature of other components where the geometry of the case plays a role in the analysis.

In order to evaluate the ability of the proposed updated design to meet the existing requirements, it is often necessary to perform analyses as part of a design change. This can take the form of updates to an existing analysis or introduction of new analyses as part of a delta qualification program. Effective analysis data management can improve this process by eliminating the need to restart analyses from scratch and avoiding duplicate analysis efforts.

3.6 Material Review Board Evaluation

While careful attention is paid to ensuring production hardware stays within the design limits verified during the qualification process, there are inevitably some departures from the established baseline that need to be processed through Material Review Board (MRB) activities. One common approach to addressing these departures is to refer back to the details of the qualification tests and analyses to determine whether adequate margin was present in the design to allow use of the suspect hardware. Sometimes existing verification data are extended via analysis to better evaluate the specific departure. Effective analysis data management facilitates this reevaluation process by making analysis and requirements data traceable and easy to find.

4. Current Condition

The current condition for the system is that requirements and analysis are not as well-coupled as we would like and that analysis data is relatively decentralized with informal methods of data management. This section discusses the current condition of requirement/analysis interrelation at Orbital ATK. For this paper, the focus is primarily on the interactions of requirements and analysis in the requirements derivation and design verification processes.

4.1 Requirement Derivation Current Condition

Systems engineering data is currently managed in TcSE. The relations between the systems engineering data elements are fairly well interrelated and managed. Any analysis effort involved in the requirement derivation effort typically consists of verbal or email requests to the analyst with little or no documented input. Feedback from the analyst to the systems engineer is likewise typically through verbal communication and email with little or no documentation. Analysts are typically not involved in the review of the final specification or subsequent requirement changes.

Analysis models and data are typically maintained on analysts’ hard drives and sometimes on a shared network drive. The only analyses typically documented in ePIC are those marked as customer deliverables and stored as engineering models (EM).

Results that are fed back into the requirement system to drive the requirement derivation are often referenced in notes attached to the requirement elements but there is no real connection maintained between the analysis and the derived requirements that would facilitate retrieval of the analysis data from the requirement objects in TcSE. Some of the challenges experienced because of this current condition are as follows:

a. Inputs into the analysis are not well documented resulting in greater risk of miscommunication
    1. This could cause inconsistency between the results of the analysis and the needs of those who requested the analysis
    2. This could also cause repeat of the analysis if that inconsistency is discovered
b. Inconsistent management of analysis data can increases the risk of data loss
    1. The inability to locate the analysis data when needed can result in compliance audit findings and the need to unnecessarily repeat analyses
c. Analysis and requirements data not linked to each other increases the risk of data inaccessibility
    1. Integrated change management impact evaluations are more difficult and often rely on human memory rather than stored data
    2. Analysts do not have direct access to requirements data
    3. Systems engineers do not have direct access to analysis data

4.2 Design Verification Current Condition

The current approach for design verification typically includes mapping requirements into verification requirements and verification activities. A table containing each of the requirements mapped to a particular activity and its corresponding verification requirements is given to the design engineer to populate for major design reviews. That table is sometimes used as an input into the analysis process. Unfortunately it is also sometimes only used after the fact by pulling data from a previously completed analysis in an attempt to "check the box" next to each requirement.

Sometimes preliminary design verification analyses may be utilized as design development activities. These design maturity analyses are often requested by the program office or project engineer without limited supporting information. Requirement inputs, if any, are typically not a full set of applicable requirements. The means of providing loads and environments data varies from project to project and is sometimes undocumented.

Coordinating requirements with analysis after the analysis is performed encourages the use of substandard compliance data. The requirement and analysis data are managed using similar techniques to those described above for the requirement derivation current condition with similar challenges.

5. Target Condition

The target condition is to bring the requirements and analysis data management processes into a single system under TcUA. This will facilitate the processes of driving design through requirements starting at the concept development phase through final verification. It will also enable the interactions between the requirements and analysis elements to be captured via database relations and managed using integrated change management processes.

6. Stakeholder Needs For Requirements/Analysis Interactions

There are several stakeholders who are concerned with how analysis data is managed in connection with requirements. A user's particular needs depend on their perspective of the system and its design development processes. This section explores some of the high level stakeholder needs associated with requirements and analysis data management and the interactions of those data. This is not intended to cover all stakeholder needs associated with either requirements and verification or analysis; this is a first cut at the stakeholder needs of a few users associated with the management of requirements and analysis data.

6.1 User Perspectives

6.1.1 Requirement Verification Systems Engineer Perspective

From the perspective of a systems engineer working with requirements and verification management, analyses are viewed as methods for deriving requirements, means of verifying that designs meet requirements, and means of evaluating proposed changes to those requirements and designs. From this perspective the analysis is acting as a service provider. Some of the data needs of the systems engineer are described here.

Note: Access to analyses from the perspective of the systems engineer means the ability to access the following types of analysis information:

a. Inputs utilized in the analysis
b. Assumptions and simplifications used in the analysis
c. Type of analysis performed
d. Tools used to perform the analysis
e. Detailed analysis results
f. Summarized analysis results

System engineering needs associated with analyses include access to the following:

a. **Source Requirements Derivation:** With a systems engineer selecting a requirement, including external customer requirements, other source requirements, and internal requirements, Teamcenter shall provide ready access to any analyses utilized to derive that requirement down to lower levels.
b. **Derived Requirements:** With a systems engineer selecting an internal requirement, Teamcenter shall provide ready access to any analyses utilized to derive that requirement from its source requirement(s).
c. **Requirements verification:** With a systems engineer selecting an internal requirement, Teamcenter shall provide ready access to any analyses utilized to verify that requirement.
d. **Verification requirements:** With a systems engineer selecting a verification requirement, Teamcenter shall provide ready access to any analyses that utilize that verification requirement as an input.
e. **Verification objectives:** With a systems engineer selecting a verification objective, Teamcenter shall provide ready access to the analyses used to accomplish that objective.

f. **Compliance statements:** With a systems engineer selecting a compliance statement, Teamcenter shall provide ready access to the analyses used as the verification activity.
g. **Problem report generation:** With a systems engineer selecting a requirement, verification requirement, verification objective, or compliance statement associated with a problem report, Teamcenter shall provide the means for the systems engineer to form a relation between that object and the problem report.

6.1.2 Analyst Perspective

From the perspective of an analyst, requirements related data are viewed as being a combination of inputs and outputs. Requirements are used to define the scope of many analyses. Verification requirements help define the constraints for how analyses should be performed. Verification objectives, contained in analysis plans, perform both roles by further clarifying the scope of the analysis and by refining the constraints applied by the verification requirements. Some of the analysts' needs associated with requirement are described here.

a. **Engineering Analysis Request access:** With an analyst selecting an analysis object, Teamcenter shall provide ready access to the Engineering Analysis Request used to drive the analysis.
b. **Requirements model access:** With an analyst selecting an analysis object, Teamcenter shall provide ready access to the project requirements model starting with any of the linked input or output requirements.
c. **Input stability:** Teamcenter shall provide means for all analysis inputs to be frozen in a way that the frozen status can later be retrieved in order to understand the inputs actually used in the analysis. This includes the following object types:
    1. Requirements
        a. Design development requirements
        b. Loads and environments data book contents
        c. Plan requirements
        d. Interface control document or drawing (ICD) requirements
        e. Etc.
    2. Verification requirements
    3. Verification objectives
    4. CAD design objects
d. **Change visibility:** With an analyst selecting an analysis object, Teamcenter shall visually indicate to the analyst whether any of its inputs have changed.

6.1.3 Design Engineer Perspective

From the perspective of a design engineer, both requirements data and analysis data are viewed as external data. Requirements are inputs into their design process. Design engineers may further derive and refine those requirements to facilitate their design effort. Analyses, along with other verification methods, are used to verify their designs meet those input requirements and derived requirements. Some of the needs of this design engineer relating to requirements and analysis are described here.

a. **Requirements:** With a design engineer selecting a part, Teamcenter shall provide ready access to the requirements associated with that part.
b. **Verification objectives and analysis:** With a design engineer selecting a part, Teamcenter shall provide ready access to the verification objectives associated with that part and the corresponding analyses tied to those objectives.
c. **Design elements and analysis:** With a design engineer selecting a design element, Teamcenter shall provide ready access to the associated analyses.

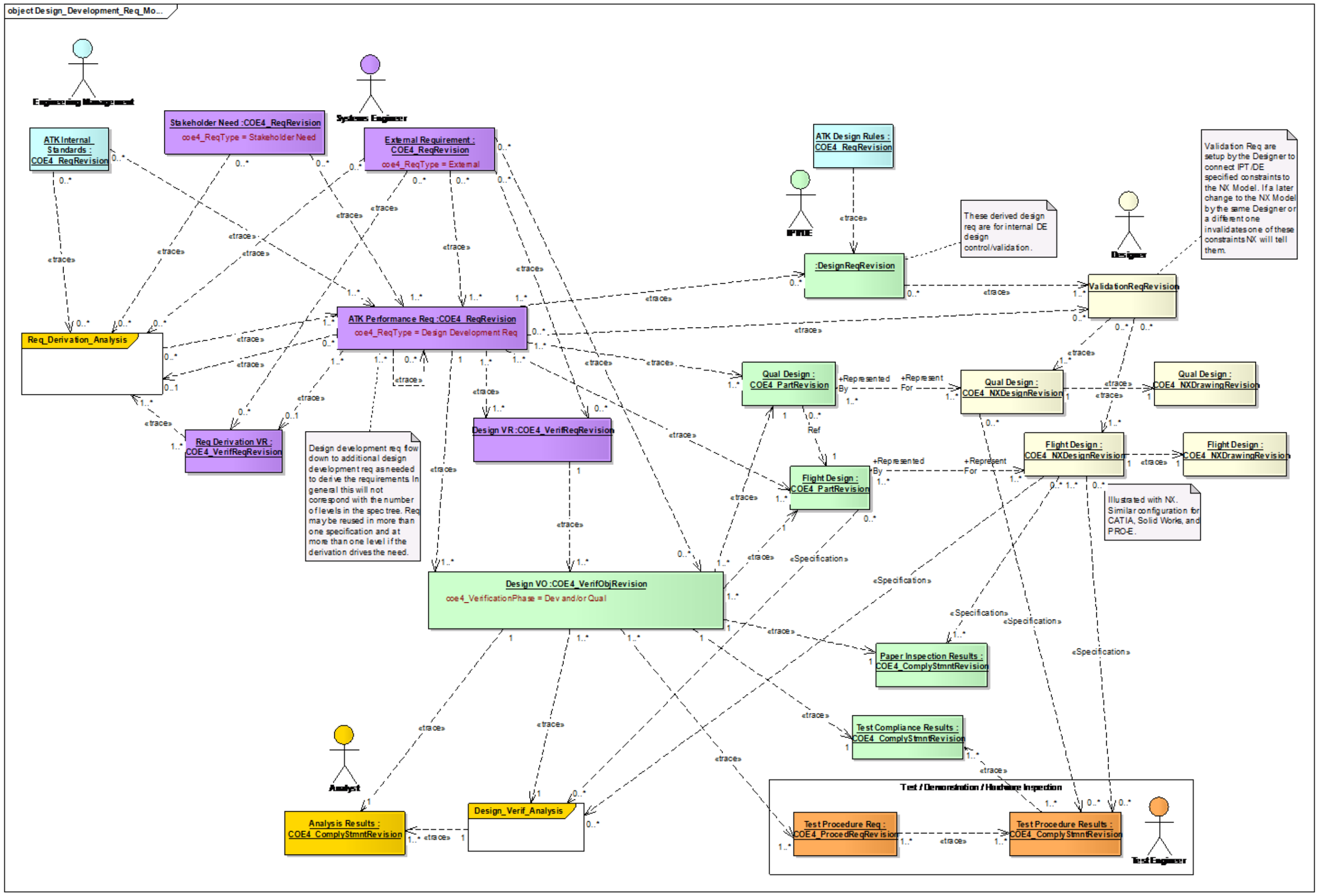
7. Solution Architecture

The architecture of the TcUA systems engineering and analysis interfaces has been designed to address the user needs described above. The TcUA systems engineering architecture has been developed over several years based on general systems engineering practices, TcSE tools and processes, and the need to seamlessly integrate with the rest of the TcUA system. The TcUA analysis architecture has been developed based on out-of-the box analysis management features in TcUA and recommendations from MAYA Simulation Technologies, an external consulting agency familiar with analysis data management in TcUA. The focus of this section is not on the separate systems engineering and analysis architecture solutions but on the integration of these two in an operating system.

7.1 Design Development Architecture

The overall data management architecture for design development, including requirements and related analyses, is shown in the object diagram in Attachment A (full size) and illustrated in **Figure 2**. Enlarged portions of this diagram relating to the interactions between requirements and analysis are captured as separate diagrams below. The object diagram in **Figure 2** shows the major data elements involved in the process of flowing down source requirements into system requirements, verification planning, verification activities, and design/CAD elements. The data model object diagram includes indication of the primary owner of each of the major elements for the purpose of data accessibility evaluation.

As described above there are several types of interactions between requirements and analysis. This paper focusses on two of these major interactions - requirement derivation and requirement verification.


object Design_Development_Req_Mo...
Engineering Management
Systems Engineer
ATK Internal Standards : COE4_ReqRevision
Stakeholder Need :COE4_ReqRevision
coe4_ReqType = Stakeholder Need
External Requirement : COE4_ReqRevision
coe4_ReqType = External
ATK Performance Req :COE4_ReqRevision
coe4_ReqType = Design Development Req
Req_Derivation_Analysis
Req Derivation VR : COE4_VerifReqRevision
Design VR :COE4_VerifReqRevision
Design VO :COE4_VerifObjRevision
coe4_VerificationPhase = Dev and/or Qual
Analyst
Analysis Results : COE4_ComplyStmntRevision
Design_Verif_Analysis
ATK Design Rules : COE4_ReqRevision
IPT/DE
:DesignReqRevision
Designer
ValidationReqRevision
Qual Design : COE4_PartRevision
Qual Design : COE4_NXDesignRevision
Qual Design : COE4_NXDrawingRevision
Flight Design : COE4_PartRevision
Flight Design : COE4_NXDesignRevision
Flight Design : COE4_NXDrawingRevision
+Represented By
+Represent For
Ref
Paper Inspection Results : COE4_ComplyStmntRevision
Test Compliance Results : COE4_ComplyStmntRevision
Test / Demonstration / Hardware Inspection
Test Procedure Req : COE4_ProcedReqRevision
Test Procedure Results : COE4_ComplyStmntRevision
Test Engineer
«trace»
«Specification»
0..*
1..*
0..1
1

Design development req flow down to additional design development req as needed to derive the requirements. In general this will not correspond with the number of levels in the spec tree. Req may be reused in more than one specification and at more than one level if the derivation drives the need.

These derived design req are for internal DE design control/validation.

Validation Req are setup by the Designer to connect IPT/DE specified constraints to the NX Model. If a later change to the NX Model by the same Designer or a different one invalidates one of these constraints NX will tell them.

Illustrated with NX. Similar configuration for CATIA, Solid Works, and PRO-E.

Figure 2. Design Development Data Model Object Diagram.

7.1.1 Requirement Derivation Analysis

While many requirements are simple enough to be derived without a significant analysis effort, when a derivation analysis is needed it is important to properly integrate the analysis and requirement data. The requirement derivation analysis in the overall data management architecture is shown in Attachment A (and Figure 2) as a reference block with the basic inputs and outputs indicated. This portion of Attachment A is enlarged in Figure 3 to highlight the inputs and outputs for the requirement derivation analysis.

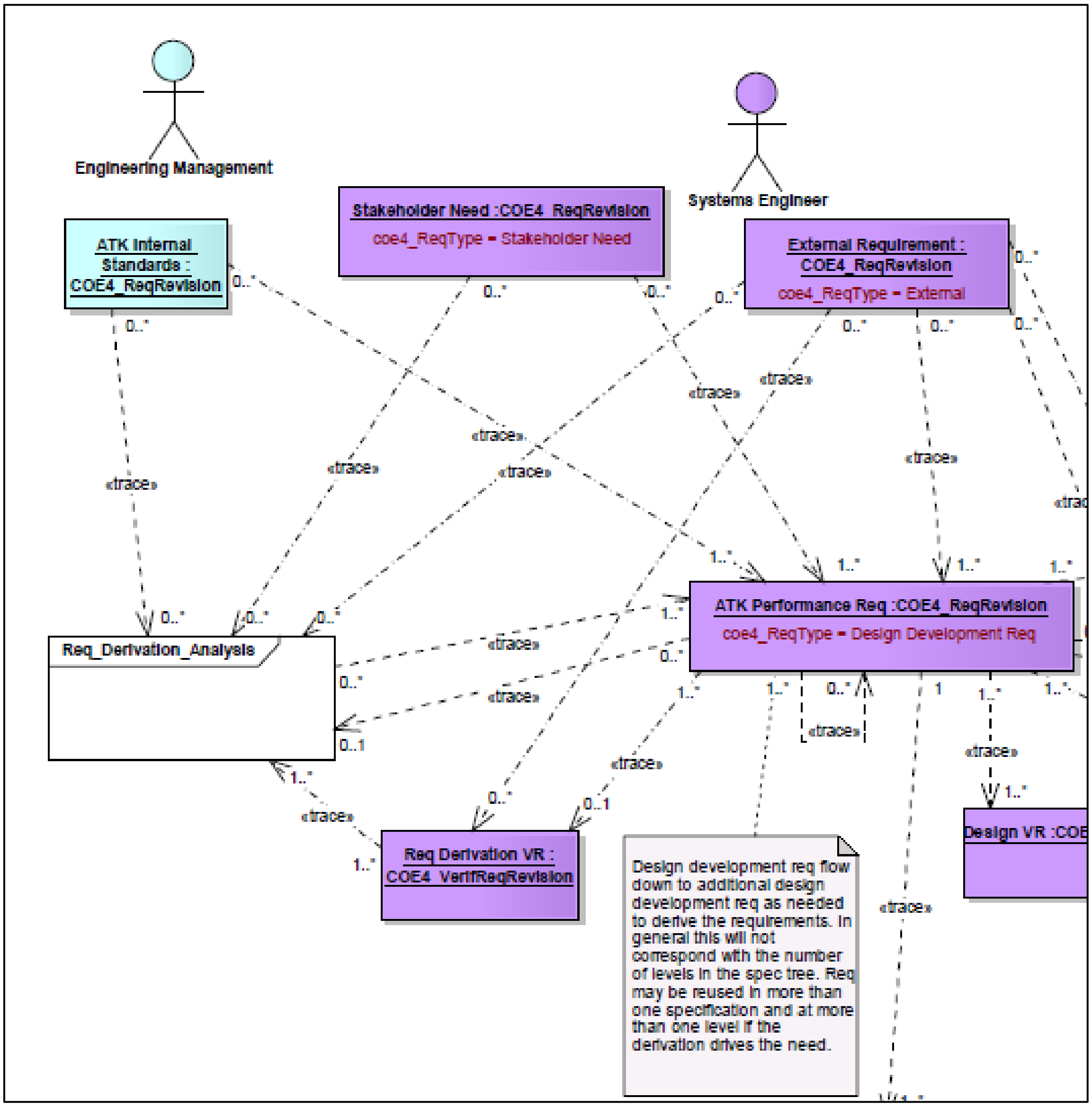

Engineering Management
Systems Engineer
ATK Internal Standards : COE4_ReqRevision
Stakeholder Need :COE4_ReqRevision
coe4_ReqType = Stakeholder Need
External Requirement : COE4_ReqRevision
coe4_ReqType = External
«trace»
0..*
1..*
0..1
1
Req_Derivation_Analysis
ATK Performance Req :COE4_ReqRevision
coe4_ReqType = Design Development Req
Req Derivation VR : COE4_VerifReqRevision
Design VR :COE
Design development req flow down to additional design development req as needed to derive the requirements. In general this will not correspond with the number of levels in the spec tree. Req may be reused in more than one specification and at more than one level if the derivation drives the need.

Figure 3. Requirements Derivation Analysis Inputs/Outputs

Inputs into the requirement derivation analysis and outputs from that analysis could include some combination of the data listed below.

a. Inputs
    1. External requirements
        a. Customer's specifications
        b. Applicable standards, specifications, handbooks, etc.
    2. Stakeholder needs
    3. Internal standards
        a. AHOPS
        b. Policies, procedures, etc.
    4. Internal requirements
        a. Requirements being derived down into lower level requirements
    5. Verification requirements
        a. Typically written by the system engineer
        b. Written expectations on how the analysis will be performed
        c. Can specify the application of additional design maturity margin
    6. Preliminary parts and designs
b. Outputs:
    1. Analysis results memo

While not illustrated in the data model object diagram, the analysis used for requirement derivation could potentially be leveraged for design verification later in the design development process.

7.1.1.1 Detailed Interactions - Requirement Derivation Analysis

A more detailed view of the interactions that occur with a requirement derivation analysis is shown in the object diagram in Attachment B (full size) and illustrated in Figure 4. In this view, the internal elements of the analysis are visible along with additional details about element interactions. The various elements of this detailed interaction model are discussed further in the following sections.

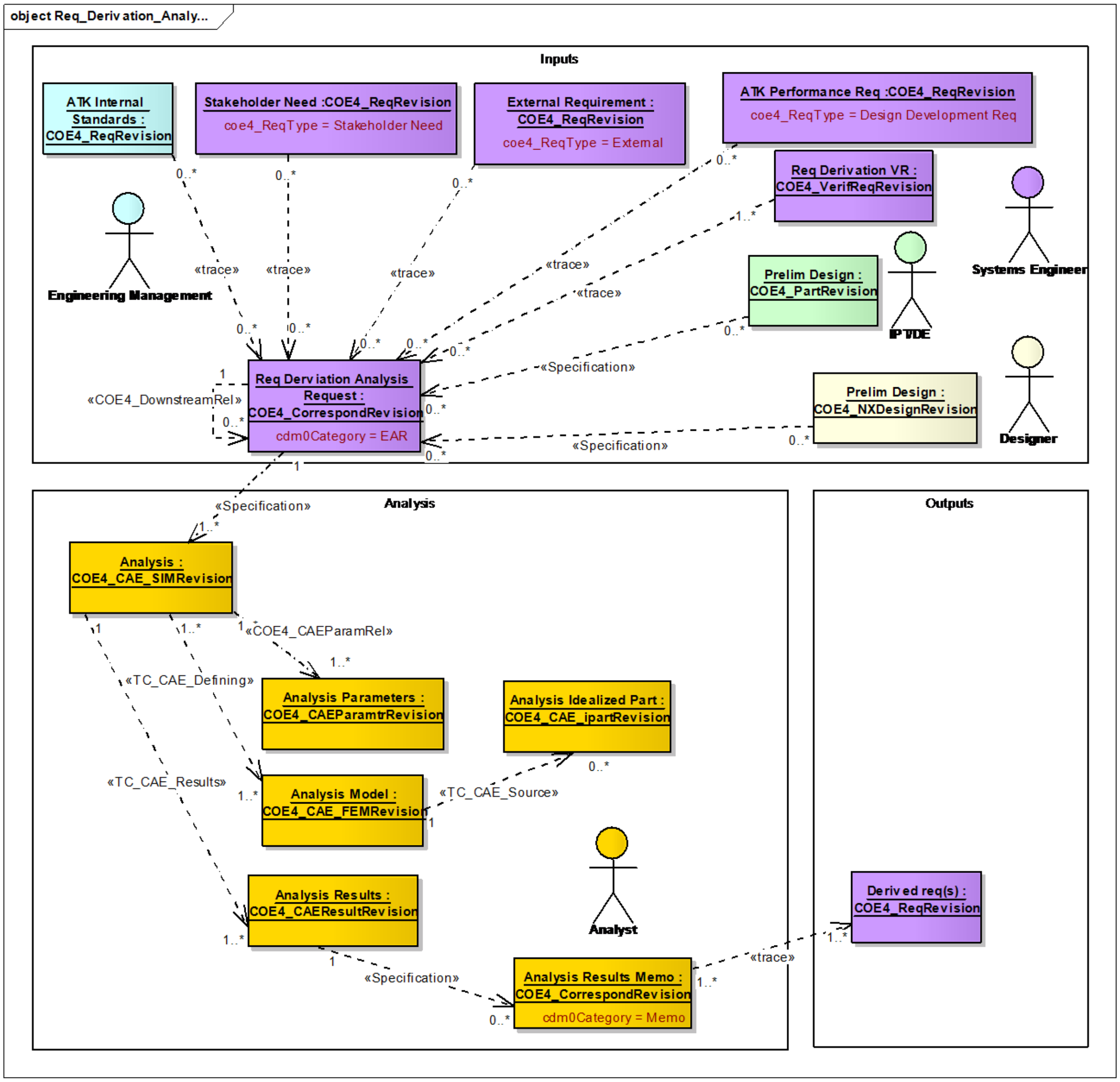

object Req_Derivation_Analy...
Inputs
ATK Internal Standards : COE4_ReqRevision
Stakeholder Need :COE4_ReqRevision
coe4_ReqType = Stakeholder Need
External Requirement : COE4_ReqRevision
coe4_ReqType = External
ATK Performance Req :COE4_ReqRevision
coe4_ReqType = Design Development Req
Req Derivation VR : COE4_VerifReqRevision
Engineering Management
Systems Engineer
Prelim Design : COE4_PartRevision
IPTDE
Prelim Design : COE4_NXDesignRevision
Designer
«trace»
«trace»
«trace»
«trace»
«trace»
«Specification»
«Specification»
«COE4_DownstreamRel»
Req Derviation Analysis Request : COE4_CorrespondRevision
cdm0Category = EAR
0..*
1..*
1
Analysis
«Specification»
Analysis : COE4_CAE_SIMRevision
«COE4_CAEParamRel»
«TC_CAE_Defining»
«TC_CAE_Results»
Analysis Parameters : COE4_CAEParamtrRevision
Analysis Idealized Part : COE4_CAE_ipartRevision
Analysis Model : COE4_CAE_FEMRevision
«TC_CAE_Source»
Analysis Results : COE4_CAEResultRevision
Analyst
«Specification»
Analysis Results Memo : COE4_CorrespondRevision
cdm0Category = Memo
«trace»
Outputs
Derived req(s) : COE4_ReqRevision

Figure 4. Requirement Derivation Analysis - Detailed View

7.1.1.2 Requirement Derivation Analysis Inputs

On the input side of the requirement derivation analysis the systems engineer is responsible for coordinating the inputs through an engineering analysis request. This includes coordination of any ATK internal standards applicable to the requirements or the analysis. The process of coordinating and documenting the analysis inputs drives the overall analysis process to produce a more efficient, higher quality analysis. This is consistent with the continuous improvement cycle where planning work precedes execution as illustrated in Figure 5. While on the surface it may seem faster to start the analysis without coordinating the inptus, the overall efficiency and quality of the analysis diminishes when the planning step is poorly planned, poorly executed, or skipped altogether.

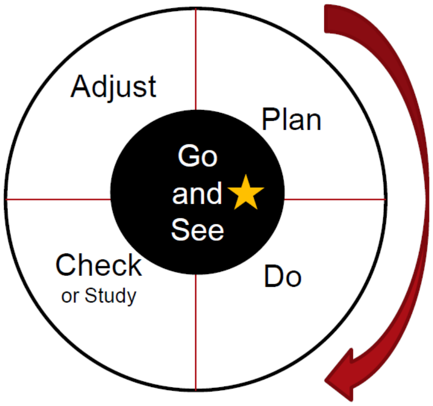


Figure 5. Continuous Improvement Cycle[1].

Traceability of the input data is a critical factor should a situation arise where the analysis needs to be reevaluated. The status of the inputs at the commencement of the analysis should be captured such that engineers can later reference the input data to develop a clear understanding about the significance of the analysis results and its applicability to future requirement iterations.

The various inputs are related to the engineering analysis request through database relations such as trace links. This enables effective change management processes by providing the means to perform impact analyses if any of the inputs or the analysis needs to be updated.

7.1.1.3 Analysis Elements

The object diagram in Figure 4 illustrates the use of several elements to store the details needed by the analyst. Those elements include parameters used as inputs, model data including the mesh definition and simplified (idealized) geometry, and results of the analysis.

Results and conclusions from the analysis are documented in a memo generated by the analyst and used as a communication tool to provide written feedback to the systems engineer about requirement derivation recommendations.

The analyst relates these database elements together using the database relations shown in the object diagram to enable impact assessments during change management processes. TcUA includes an analysis management model that helps with that process by visually marking analysis elements and inputs that are out of date.

7.1.1.4 Requirement Derivation Analysis Outputs

As shown in the object diagram in Figure 4 the systems engineer will update or generate requirement objects based on the results of the analysis. The systems engineer will relate the analysis documentation to the derived requirements via database trace links to enable effective change management.

7.1.2 Design Verification Analysis

Many design development requirements involve some level of analysis during the verification process. Following the continuous improvement cycle shown in Figure 5, these verification analyses need to be planned prior to execution. The design verification analysis is shown in Attachment A (and Figure 2) as a reference block with the basic inputs and outputs indicated. This portion of Attachment A is enlarged in Figure 6 to highlight the inputs and outputs for the design verification analysis.

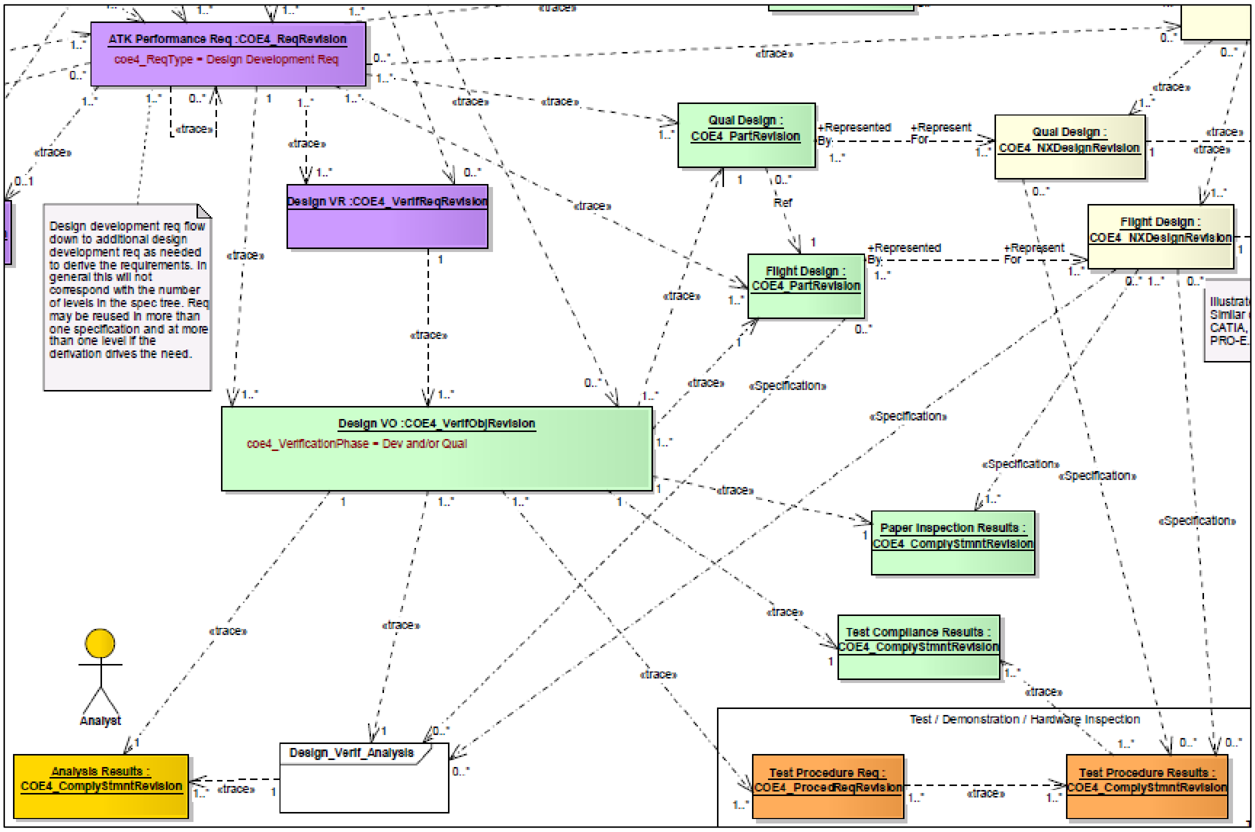

ATK Performance Req :COE4_ReqRevision
coe4_ReqType = Design Development Req
Qual Design : COE4_PartRevision
Qual Design : COE4_NXDesignRevision
Flight Design : COE4_PartRevision
Flight Design : COE4_NXDesignRevision
+Represented By
+Represent For
Ref
Design VR :COE4_VerifReqRevision
Design development req flow down to additional design development req as needed to derive the requirements. In general this will not correspond with the number of levels in the spec tree. Req may be reused in more than one specification and at more than one level if the derivation drives the need.
Design VO :COE4_VerifObjRevision
coe4_VerificationPhase = Dev and/or Qual
«trace»
«Specification»
Paper Inspection Results : COE4_ComplyStmntRevision
Test Compliance Results : COE4_ComplyStmntRevision
Analyst
Analysis Results : COE4_ComplyStmntRevision
Design_Verif_Analysis
Test / Demonstration / Hardware Inspection
Test Procedure Req : COE4_ProcedReqRevision
Test Procedure Results : COE4_ComplyStmntRevision
Illustrat Similar CATIA, PRO-E.
1
1..*
0..*
0..1

Figure 6. Design Verification Analysis Inputs/Outputs

Inputs into the design verification analysis include verification objectives and CAD design elements. Outputs are compliance statements.

7.1.2.1 Detailed Interactions - Design Verification Analysis

A more detailed view of the interactions that occur with a design verification analysis is shown in the object diagram in Attachment C and illustrated in Figure 7 for convenient reference. In this view the internal elements of the analysis are visible along with additional details about element interactions. The various elements of this detailed interaction model are discussed further in the following sections.

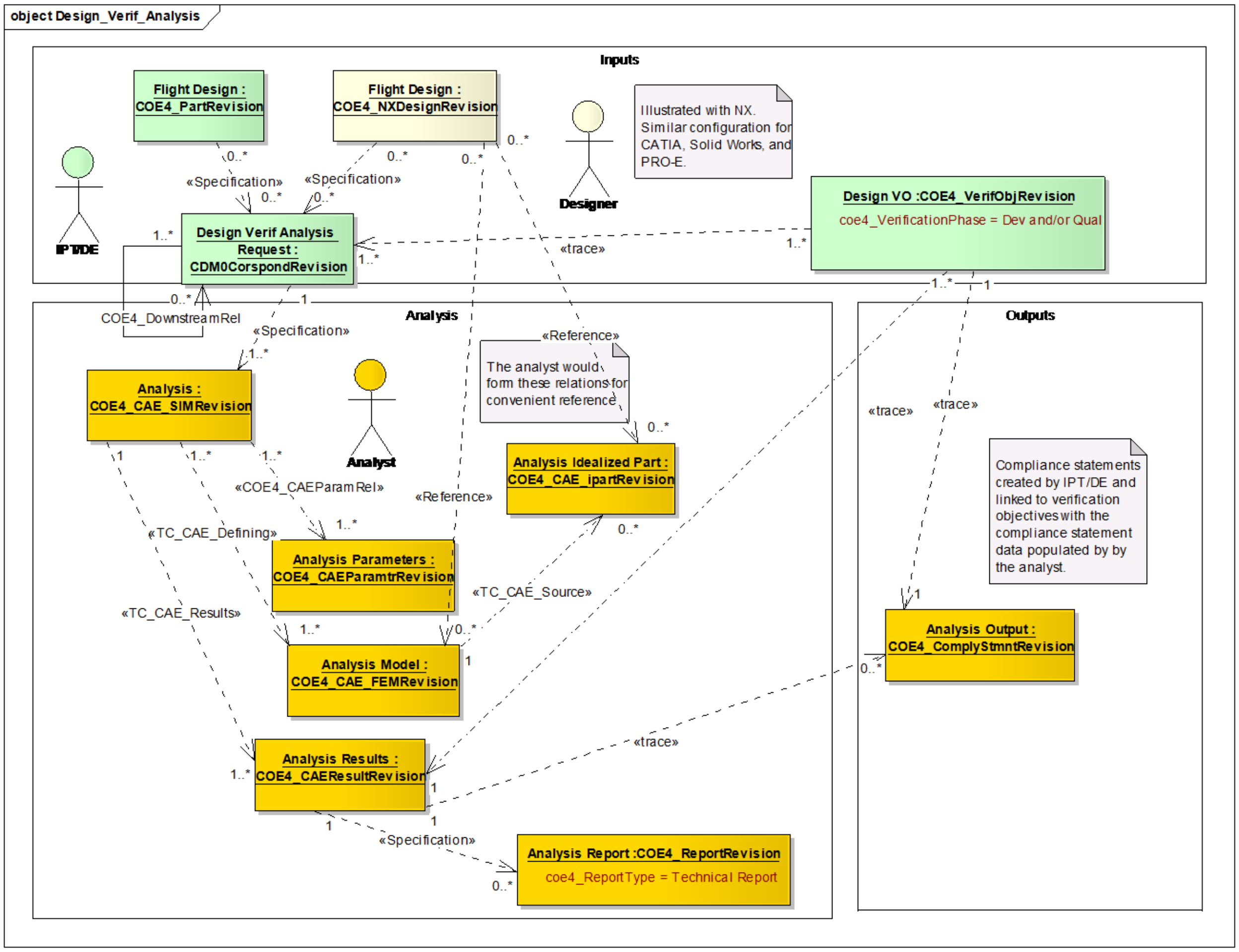

object Design_Verif_Analysis
Inputs
Flight Design : COE4_PartRevision
Flight Design : COE4_NXDesignRevision
Illustrated with NX. Similar configuration for CATIA, Solid Works, and PRO-E.
Designer
IPT/DE
Design VO :COE4_VerifObjRevision
coe4_VerificationPhase = Dev and/or Qual
Design Verif Analysis Request : CDM0CorspondRevision
«Specification»
«Specification»
«trace»
COE4_DownstreamRel
Analysis
Analyst
Analysis : COE4_CAE_SIMRevision
«Reference»
The analyst would form these relations for convenient reference
Analysis Idealized Part : COE4_CAE_ipartRevision
«COE4_CAEParamRel»
«Reference»
«TC_CAE_Defining»
Analysis Parameters : COE4_CAEParamtrRevision
«TC_CAE_Source»
«TC_CAE_Results»
Analysis Model : COE4_CAE_FEMRevision
Analysis Results : COE4_CAEResultRevision
«trace»
«Specification»
Analysis Report :COE4_ReportRevision
coe4_ReportType = Technical Report
Outputs
«trace»
«trace»
Compliance statements created by IPT/DE and linked to verification objectives with the compliance statement data populated by by the analyst.
Analysis Output : COE4_ComplyStmntRevision
0..*
1..*
1

Figure 7. Design Verification Analysis - Detailed View

7.1.2.2 Design Verification Analysis Inputs

On the input side of the design derivation analysis, the design engineer generates verification objectives for each of the applicable requirements based on corresponding verification requirements. These verification objectives are typically grouped into analysis plans. The design engineer also coordinates the part and design elements that act as inputs into the design analysis process. These inputs are coordinated through an engineering analysis request. The status of the inputs at the commencement of the analysis should be captured for later reference. Traceability of the input data provides a clear understanding about the significance of the analysis results and its applicability to future design iterations should the analysis ever need to be reevaluated.

The various inputs are related to the engineering analysis request through database relations such as trace links. This enables effective change management processes by providing the means of perform impact analyses if any of the inputs or the analysis needs to be updated.

7.1.2.3 Analysis Elements

The object diagram in Figure 7 illustrates that the same types of elements may be used in design verification processes as were used in the requirement derivation processes. In some cases the very same analysis started in the requirement derivation process can be extended to achieve the purposes of the design verification analysis. In other cases new analyses will be generated.

The analyst relates these analysis database elements together using the database relations shown in the object diagram to enable impact assessments during change management processes. The analyst will form additional links from the input CAD design elements to the analysis model and idealized parts in order to provide a more convenient direct reference. Finally, the design analysis process also includes the analyst mapping the individual verification objectives to individual analysis results elements. This mapping process helps ensure that all analyses have proper accounting and allows the change management process to work more efficiently.

7.1.2.4 Design Verification Analysis Outputs

As shown in Figure 7, the analyst generates compliance statements for each of the verification objectives. The compliance statements are collected into a technical report and released as the compliance documentation.

8. Solution Validation Testing Case Studies

8.1 Case Study Test Environments

Case study testing will be performed primarily in two environments. First, preliminary testing will be performed in a virtual machine environment (also called an individual development environment or iDev). This testing will be performed primarily by the engineers developing the systems engineering and analysis architectures. The goal of this testing is to provide adequate confidence before deploying the integrated configuration to the development server.

After testing in the iDev environments have been completed, the integrated analysis/requirements configuration will be evaluated by the information technology team responsible for TcUA (Center of Excellence, or COE). Passing the COE evaluation authorizes deploying the configuration to the development server. In this environment multiple users will be invited to help execute the case studies. This will provide an opportunity to really test how well the data is accessible from the perspective of each of the participating users. This will also provide an opportunity for feedback to the engineers developing the TcUA configurations prior to deployment to the qualification or production environments.

8.2 Case Study 1: Allocation - Timing Tolerance Analysis

Case study 1 will focus on derivation of a requirement via a relatively simple timing tolerance analysis followed by design verification of the related design using the same analysis. It will use a fictitious system-level arming time requirement for the flight termination system (FTS) safe and arm (S&A) device. The input requirement for this test case will be a total system timing requirement that will be derived into requirements for an electronics subsystem and the FTS. The analysis will be performed using an Excel spreadsheet.

8.2.1 Case Study 1 Inputs

a. Dummy customer requirement for FTS S&A safe timing

8.2.2 Case Study 1 Objectives

a. Demonstrate a simple analysis by systems engineer using spreadsheet
b. Demonstrate the process of generating derived subsystem requirements from analysis
c. Demonstrate iteration of the derivation process as inputs and design mature
d. Demonstrate reuse of analysis for verification purposes with handoff to a system-level design engineer
e. Demonstrate that the right data is accessible at each step in the process including change management

8.3 Case Study 2: Hardware Structural Analysis

Case study 2 will focus on design verification for a more complex use case that involves finite element models and idealized parts. It will involve using a load requirement and a factor of safety requirement, a simplified part geometry, and a finite element structural analysis used to verify compliance to those requirements.

8.3.1 Case Study 2 Inputs

a. Dummy customer requirements for loads and factors of safety
b. Dummy preliminary design concept

8.3.2 Case Study 2 Objectives

a. Demonstrate a more complex analysis performed by a dedicated analyst
b. Demonstrate design analysis driven through verification objectives
c. Demonstrate the process of generating compliance statements
d. Demonstrate that the right data is accessible at each step in the process including change management

8.4 Case Study 3: Thermal Conditions Determination Analysis

Case study 3 will focus on a combination of elements from case studies 1 and 2. It will involve requirement derivation using a preliminary thermal analysis model to take external environments and predicted data for self-heating and determine the predicted temperatures of the various hardware items. After adding some design maturity margin, the predicted temperatures would then be flowed into derived requirements for those hardware items specifying that they handle the predicted temperatures. This case study will then proceed to the verification step for the system level where the preliminary analysis is transformed into a more complex but higher fidelity model in order to verify that the design of the system is consistent with allocations to the hardware items (the system design includes enough thermal protection features to keep the hardware items within the prescribed temperature range).

8.4.1 Case Study 3 Inputs

e. Dummy customer requirements for loads and factors of safety

8.4.2 Case Study 3 Objectives

a. Demonstrate a more complex analysis performed by a dedicated analyst
b. Demonstrate the process of generating derived subsystem requirements from analysis
c. Demonstrate iteration of the derivation process as inputs and design mature
d. Demonstrate the evolution of the analysis when design is mature enough to use higher fidelity analysis tools while keeping traces to the preliminary analysis
e. Demonstrate the process of generating compliance statements
f. Demonstrate that the right data is accessible at each step in the process including change management

9. Conclusions

Integrating requirements and verification data management with analysis data management processes in TcUA enables Orbital ATK to more effectively execute their engineering processes. Benefits include the following.

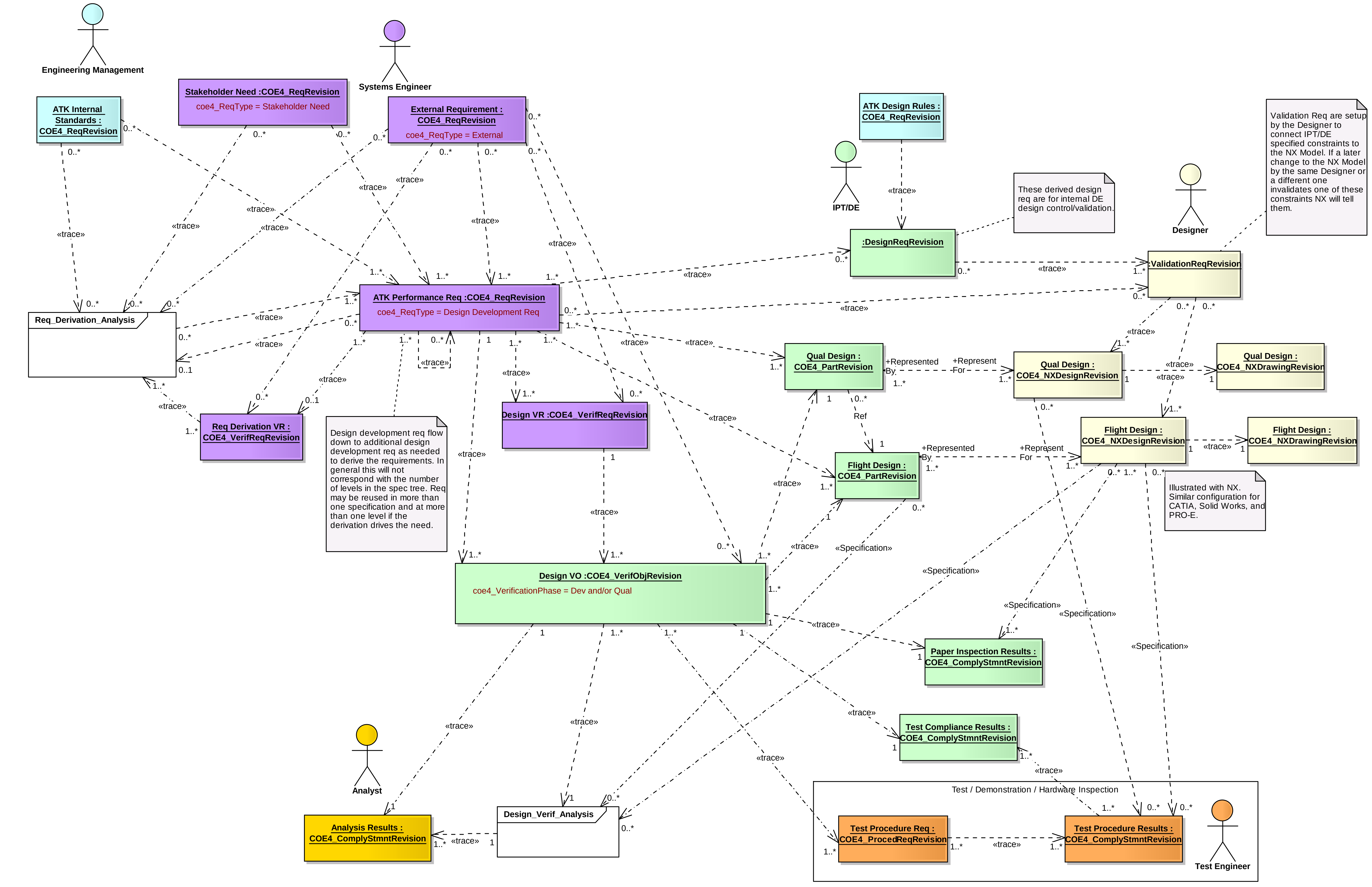
a. Connections between requirements data and analysis data enable integrated change management processes
b. Analyst can directly access the requirements driving their analysis
c. Analyst can directly provide compliance data into compliance reporting objects helping minimize data handoffs
d. Systems engineers can directly access the analysis data
e. Design engineers can directly access requirements and analysis data for integration with part design data

This paper has presented information about the proposed architecture solution to achieve these benefits and the recommended concept feasibility testing to validate that solution.

Continuing efforts to validate these solutions and expand our TcUA capability should be supported by management through budget and resource allocation.

10. References

1. PES Training, http://launch.atk.com/organizations/PES/Training%20Manuals/Day1PMc-Problem%20Solving%20(Kata%20addition).pdf

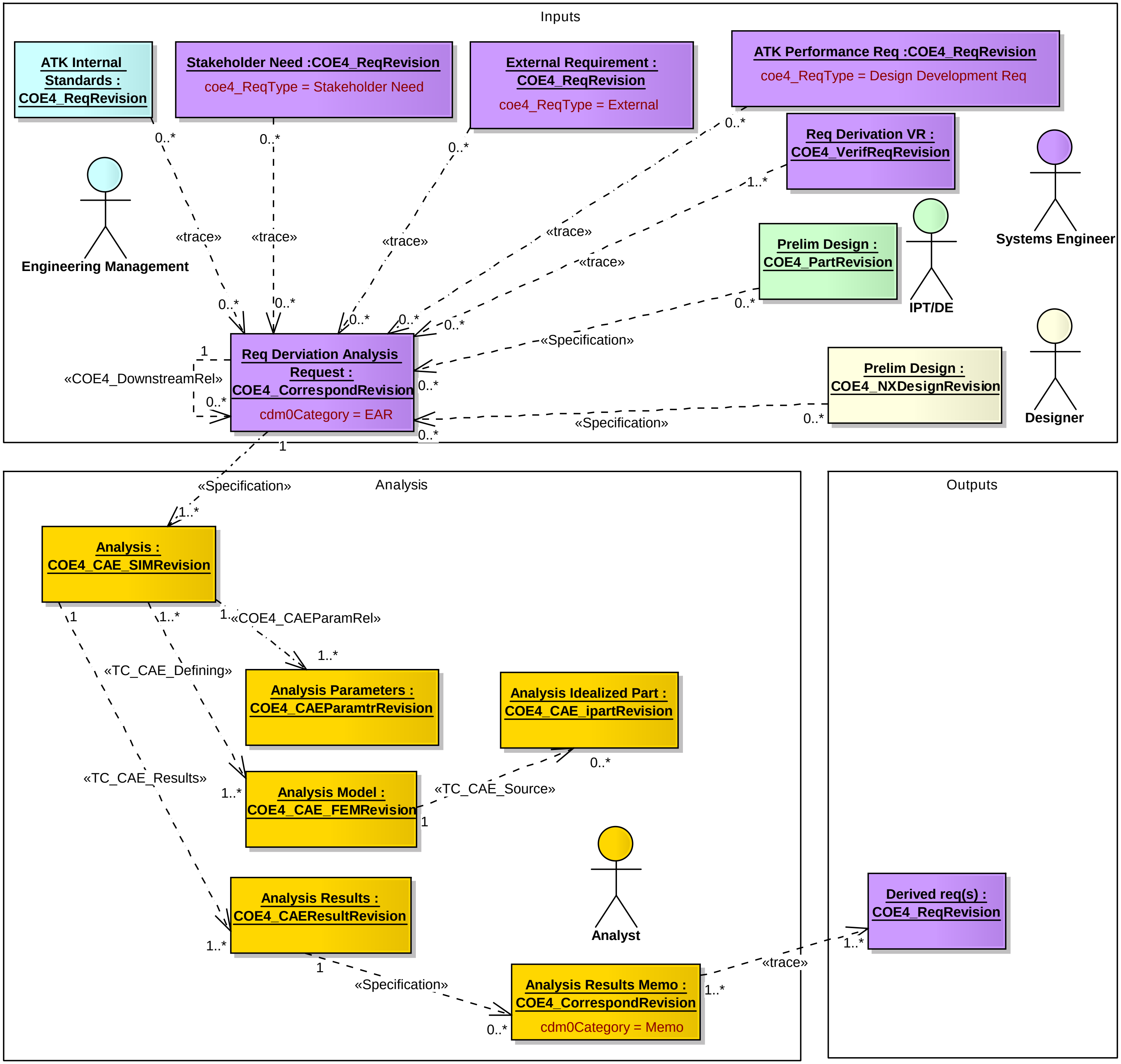

Engineering Management
Systems Engineer
IPT/DE
Designer
Analyst
Test Engineer
ATK Internal Standards : COE4_ReqRevision
Stakeholder Need :COE4_ReqRevision
coe4_ReqType = Stakeholder Need
External Requirement : COE4_ReqRevision
coe4_ReqType = External
ATK Design Rules : COE4_ReqRevision
:DesignReqRevision
:ValidationReqRevision
Req_Derivation_Analysis
ATK Performance Req :COE4_ReqRevision
coe4_ReqType = Design Development Req
Req Derivation VR : COE4_VerifReqRevision
Design VR :COE4_VerifReqRevision
Qual Design : COE4_PartRevision
Qual Design : COE4_NXDesignRevision
Qual Design : COE4_NXDrawingRevision
Flight Design : COE4_PartRevision
Flight Design : COE4_NXDesignRevision
Flight Design : COE4_NXDrawingRevision
Design VO :COE4_VerifObjRevision
coe4_VerificationPhase = Dev and/or Qual
Paper Inspection Results : COE4_ComplyStmntRevision
Test Compliance Results : COE4_ComplyStmntRevision
Analysis Results : COE4_ComplyStmntRevision
Design_Verif_Analysis
Test / Demonstration / Hardware Inspection
Test Procedure Req : COE4_ProcedReqRevision
Test Procedure Results : COE4_ComplyStmntRevision
«trace»
«Specification»
+Represented By
+Represent For
Ref
0..*
1..*
0..1
1

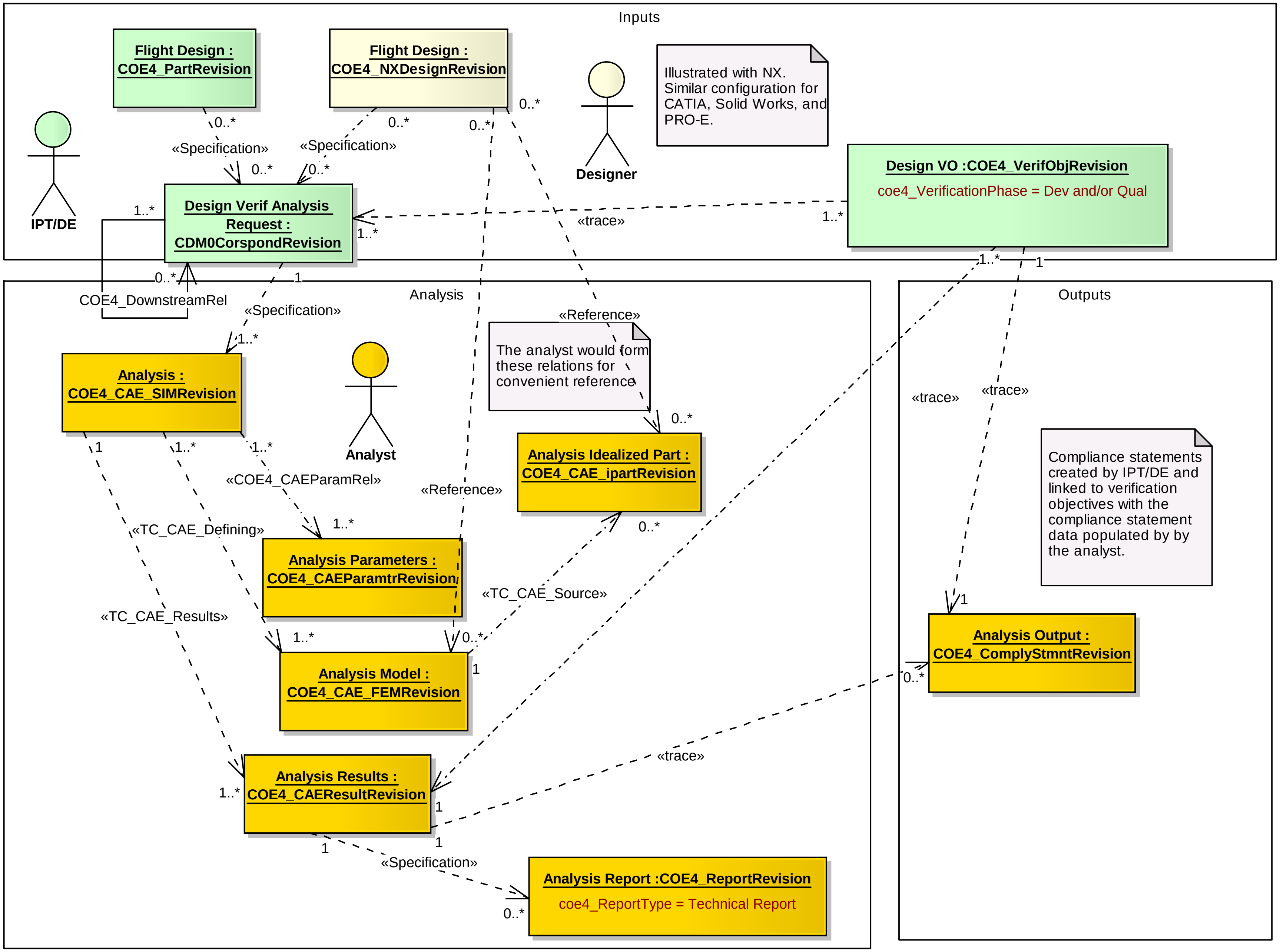
These derived design req are for internal DE design control/validation.

Validation Req are setup by the Designer to connect IPT/DE specified constraints to the NX Model. If a later change to the NX Model by the same Designer or a different one invalidates one of these constraints NX will tell them.

Design development req flow down to additional design development req as needed to derive the requirements. In general this will not correspond with the number of levels in the spec tree. Req may be reused in more than one specification and at more than one level in the spec tree if the derivation drives the need.

Illustrated with NX. Similar configuration for CATIA, Solid Works, and PRO-E.


Inputs
ATK Internal Standards : COE4_ReqRevision
Stakeholder Need :COE4_ReqRevision
coe4_ReqType = Stakeholder Need
External Requirement : COE4_ReqRevision
coe4_ReqType = External
ATK Performance Req :COE4_ReqRevision
coe4_ReqType = Design Development Req
Req Derivation VR : COE4_VerifReqRevision
Engineering Management
Systems Engineer
Prelim Design : COE4_PartRevision
IPT/DE
Prelim Design : COE4_NXDesignRevision
Designer
Req Derviation Analysis Request : COE4_CorrespondRevision
cdm0Category = EAR
«trace»
«trace»
«trace»
«trace»
«trace»
«Specification»
«Specification»
«COE4_DownstreamRel»
0..*
0..*
0..*
0..*
1..*
0..*
0..*
1
0..*
0..*
0..*
0..*
0..*
0..*
0..*
0..*
0..*
0..*
1
Analysis
«Specification»
1..*
Analysis : COE4_CAE_SIMRevision
1
1..*
1
«COE4_CAEParamRel»
1..*
«TC_CAE_Defining»
Analysis Parameters : COE4_CAEParamtrRevision
Analysis Idealized Part : COE4_CAE_ipartRevision
0..*
«TC_CAE_Results»
1..*
Analysis Model : COE4_CAE_FEMRevision
«TC_CAE_Source»
1
Analysis Results : COE4_CAEResultRevision
1..*
Analyst
1
«Specification»
0..*
Analysis Results Memo : COE4_CorrespondRevision
cdm0Category = Memo
1..*
«trace»
1..*
Outputs
Derived req(s) : COE4_ReqRevision


Inputs
Flight Design : COE4_PartRevision
Flight Design : COE4_NXDesignRevision
IPT/DE
Designer
Design Verif Analysis Request : CDM0CorspondRevision
Design VO :COE4_VerifObjRevision
coe4_VerificationPhase = Dev and/or Qual
«Specification»
«Specification»
«trace»
0..*
0..*
0..*
0..*
0..*
0..*
1..*
1..*
1..*
0..*
COE4_DownstreamRel
Analysis
Outputs
1
1..*
1
«Specification»
1..*
Analysis : COE4_CAE_SIMRevision
Analyst
«Reference»
0..*
Analysis Idealized Part : COE4_CAE_ipartRevision
1
1..*
1..*
«COE4_CAEParamRel»
«Reference»
«TC_CAE_Defining»
1..*
Analysis Parameters : COE4_CAEParamtrRevision
0..*
«TC_CAE_Source»
«TC_CAE_Results»
1..*
0..*
1
Analysis Model : COE4_CAE_FEMRevision
Analysis Results : COE4_CAEResultRevision
1..*
1
«trace»
1
1
«Specification»
0..*
Analysis Report :COE4_ReportRevision
coe4_ReportType = Technical Report
«trace»
«trace»
1
Analysis Output : COE4_ComplyStmntRevision
0..*

Illustrated with NX. Similar configuration for CATIA, Solid Works, and PRO-E.

The analyst would form these relations for convenient reference

Compliance statements created by IPT/DE and linked to verification objectives with the compliance statement data populated by by the analyst.