

Tema
Inginerie Software

UML - OMG
Systems Modeling Language – SysML

Selaru Catalin
Calota Beniamin
Grupa:443A

1.SysML scop și caracteristici cheie

SysML este un limbaj de uz general de modelare grafica care suporta analiza, specificarea, proiectarea, verificarea si validarea sistemelor complexe. Aceste sisteme pot include hardware-ul, software-ul, datele, personalul, procedurile, facilitatile, precum si alte elemente ale sistemelor artificiale si naturale. Limbajul este destinat sa ajute la specificarea arhitecturii sistemelor si sa precizeze componentele acestora, care pot fi apoi proiectate folosind alte sisteme specifice domeniului, cum ar fi UML pentru software-ul de proiectare si VHDL si tri-dimensională pentru modelarea geometrica si pentru proiectare hardware. SysML este destinat aplicatiilor menite sa faciliteze abordari MBSE pentru a crea modele coerente și consecvente.

SysML prezinta urmatoarele aspecte ale sistemelor, componentelor si a altor entitati:

- compozitie structurala, interconectare si clasificare
- functii si mesaje de baza
- constrangeri privind proprietățile fizice si performanta
- alocari intre comportament, structura si constrangeri
- cerintele si relatia lor cu alte cerinte, elementele de proiectare si cazuri de testare

2.SysML schema generala

SysML include noua diagrame. Fiecare tip de diagrama este prezentat mai jos, impreuna cu relatia sa cu diagrama UML:

- *Diagrama pachet* reprezinta modul de organizare a unui pachet care contine elemente din model (la fel ca diagrama pachet UML)
- *Diagrama de cerinte* reprezinta diagrama bazata pe cerintele de text si relatia lor cu alte cerinte, elementele de proiectare si cazuri de testare pentru a sprijini cerintele de trasabilitate (nu in UML)
- *Diagrama de activitate* reprezinta comportamentul in ceea ce priveste ordinea in care sunt efectuate actiunile bazate pe disponibilitatea intrarilor, iesirilor precum si modul in care actiunile transforma intrarile in iesiri (modifica diagrama UML de activitate)
- *Diagrama de secventa* reprezinta comportamentul in ceea ce priveste mesajele schimbate intre sisteme sau intre parti de sisteme (la fel ca diagrama de secvente UML)

- *Diagrama aparatului de functionare* reprezinta comportamentul unei entitati in ceea ce priveste tranzitiile intre functiile sale declansate de evenimente (la fel ca diagrama aparatului de functionare UML)
- *Diagramei cazurilor de utilizare* reprezinta functionalitatea in ceea ce priveste modul in care un sistem este utilizat de catre entitati externe pentru a realiza un set de obiective (la fel ca si utilizarea diagramei UML a cazurilor)
- *Schema bloc de definitie* reprezinta elementele structurale numite blocuri si specifica compozitia lor (modificata fata de diagrama UML de clasa)
- *Schema bloc interna* reprezinta interconectarea si interfetele intre partile componente ale unui bloc (modificare a structurii componente a diagramei UML)
- *Diagrama parametrica* reprezinta constrangeri asupra valorilor de proprietate utilizate pentru a sprijini analiza inginereasca (nu in UML)

Tipurile de elemente de model si simbolurile asociate (de exemplu: elemente de diagrama) care pot aparea pe o diagrama sunt constranse de tipul diagramei. De exemplu, o diagrama de activitate poate include elemente de diagrama care reprezinta actiuni, controlul fluxului si de intrare/iesire a debitului (de exemplu: debit obiect), dar nu elemente de diagrama pentru conectori si porturi. Ca rezultat, o diagrama reprezinta un subset al depozitului modelului de baza. Reprezentările tabelare, cum ar fi tabelele de alocare sunt, de asemenea, sustinute in SysML ca o completare a reprezentarilor diagramei pentru a afisa informatiile model.

3. Introducere in SysML-Lite

SysML-Lite este prezentat aici ca o versiune simplificata a limbajului pentru a ajuta oamenii sa ajunga la modelul de baza a SysML, care nu face parte din standardul SysML. Acesta include sase dintre cele noua diagrame SysML, precum si un mic subset de caracteristici a limbajului disponibile pentru fiecare tip de diagrama. SysML-Lite ofera o capacitate de modelare semnificativa. Aceasta sectiune ofera o scurta introducere la SysML-Lite, impreuna cu un exemplu simplu pentru a evidentia caracteristicile SysML-Lite. Aceasta sectiune include, de asemenea sfaturi initiale pentru a ajuta un modelator nou in utilizarea unui instrument tipic de modelare.

3.1 SysML-Lite - Schema si caracteristici ale limbajului

Cele sase tipuri de diagrame care fac parte din SysML-Lite sunt evidentiata in figura 3.2. Fiecare diagrama contine un antet diagrama care identifica tipul diagramei. Diagrama SysML-Lite include:

- *diagrama pachet* pentru a capta organizarea modelului
- *diagrama de obligatii* pentru capturarea cerintelor bazate pe text
- *diagrama de activitate* pentru a reprezenta comportamentul sistemului si a componentelor sale
- *schema bloc de definitie* reprezinta ierarhia sistemului
- *schema bloc interna* pentru a reprezenta interconectarea sistemului
- *diagrama parametrica* pentru a capta relatia dintre proprietatile de sistem ce sprijina analiza ingineriasca\

Acest set de diagrame ofera un model standard, cu o capacitate substantiala pentru sistemele de modelare, care acopera multe dintre schemele clasice ale sistemelor.

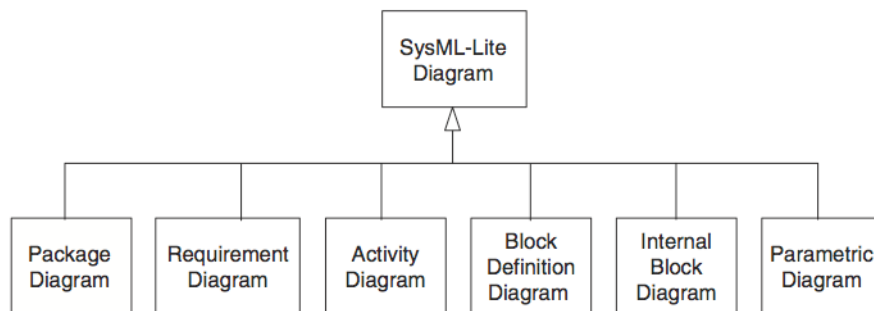


FIGURE 3.2

SysML-Lite includes six of the nine SysML diagrams and a subset of the language features. It is intended to introduce a new modeler to SysML, while providing a substantial modeling capability.

SysML-Lite include un subset de caracteristici lingvistice pentru fiecare din cele sase diagrame SysML. Cateva dintre caracteristicile SysML-Lite sunt reprezentate in diagramele din figura 3.3. Caracteristicile lingvistice ale SysML pot fi adaptate la nevoie. In figura este precizat cu sageti relatiile importante din diagrama. Aceste relatii sunt, in general, in concordanta cu metodele clasice de inginerie a sistemelor, cum ar fi descompunerea functionala si alocarea.

Diagrama pachet este folosita pentru a organiza elementele modelului continute in model. In aceasta diagrama, modelul sistemului apare in antetul diagramei si contine

pachetele pentru cerinte, comportament, structura si parametrii. Fiecare dintre aceste pachete, la radul sau, contine elemente de model care sunt reprezentate pe diagrama de cerinte, diagrama de activitate, diagrama bloc de definitie, diagrama bloc interna si schema parametrica. Elementele modelului, atat pentru definirea diagramei bloc cat si pentru diagrama bloc interna sunt incluse in pachetul de structura.

Diagrama de cerinte reprezinta o ierarhie simpla de text bazata pe cerinte care sunt de obicei parte a unui document din caietul de sarcini. Cerinta de nivel superior numita R1 contine doua cerinte R1.1 si R1.2. Declartia de cerite pentru R1.1 este capturata ca o proprietate a textului cerintei si corespunde textului care va fi gasit pentru aceasta cerinta in documentul caietului de sarcini.

Diagramele de activitate sunt etichetate „act”. Diagrama de activitate numita A0 reprezinta interactiunea dintre Sistemul 1 si Sistemul 2. Nodul initial reprezentat de cercul inchis de umplutura și nodul final reprezentat de bulls-eye indica inceputul si respectiv sfarsitul activitatii. Activitatea specifica o secventa simpla de actiuni incepand cu executarea actiunii A1 si urmata de executarea actiunii A2. Iesirele A1 si intrarile A2 sunt reprezentate de dreptunghiuri pe granita de actiune numite pini. In plus, partiile de activitate etichetate „Sistemul 1” si „Sistemul 2” sunt responsabile pentru efectuarea actiunilor care sunt cuprinse intre partiile. Actiunea numita mai sus indeplineste cerinta A1 R1.2, care este reprezentat de relatia de satisfacere.

Actiune numita A1 in diagrama de activitate A0 este descompusa in diagrama de activitate numita A1 in actiunile A1.1 si A1.2. Aceste actiuni sunt efectuate de catre Componenta 1 si, respectiv, Componenta 2. Iesirea activitatii A1 este reprezentata de un dreptunghi pe marginea caruia ii corespunde pinul de iesire din actiunea A1 in activitatea A0. Dupa cum se indica in diagramele de activitate pentru A0 si A1, iesirile si intrarile sunt conforme de la un nivel de descompunere la altul.

Schema bloc de definitie este etichetata BDD si este adesea folosita pentru a descrie ierarhia unui sistem similar cu un copac de piese.. Un bloc este utilizat pentru a defini un sistem sau o componenta la orice nivel al ierarhiei sistemului. Diagrama de definitie a blocului in figura prezinta contextul sistemului bloc compus din Sistemul 1 si Sistemul 2. Sistemul 1 este in continuare descompus in Componenta 1 si Componenta 2. Sistemul 1 si Componenta 1 a blocului contine o valoare care poate corespunde unei caracteristici fizice sau de performanta, cum ar fi greutatea sau timpul de raspuns.

Schema bloc interna este etichetata IBD si arata modul in care parti ale sistemului 1 sunt interconectate. Schema cadru reprezinta Sistemul 1. Patrutele mici pe Sistemul 1 si partile sale componente sunt numite porturi si reprezinta interfetele lor. Sistemul 1 este, de asemenea, reprezentat de partiile de activitate in activitatea A0, iar componentele sunt reprezentate de partiile similare de activitate din activitatea A1.

Diagrama parametrica este etichetata „par” și este utilizata pentru a descrie relatiile dintre proprietatile care corespund unei analize de inginerie, cum ar fi

performanta, fiabilitatea sau analiza proprietatilor de masa. In acest exemplu, diagrama parametrica include o constrangere singulara numita Constrangere 1 care corespunde unei ecuatii sau unui set de ecuatii.

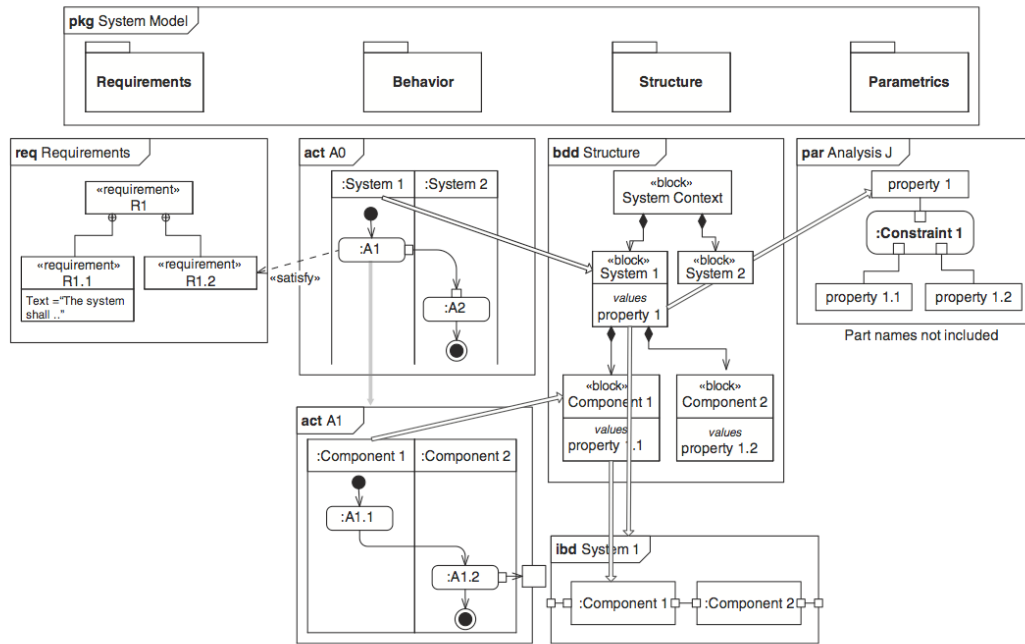


FIGURE 3.3

Simplified diagrams highlighting some of the language features for each kind of diagram in SysML-Lite.

In diagramele de mai sus, doar o mica parte din caracteristicile lingvistice SysML sunt ilustrate pentru a indica unele dintre cheile de constructie utilizate la sistemele de model. Urmatorul model simplificat este cel al unui compresor de aer care ilustreaza modul in care pot fi aplicate diagramele SysML-Lite si caracteristicile lingvistice ale acestora.

3.2 SysML-Lite-exemplu: compresor de aer

În continuare vom prezenta un exemplu de utilizare a SysML-Lite pentru a modela un compresor de aer, care este folosit pentru a alimenta un instrument de aer. Acest model este extrem de simplificat și are scopul de a ilustra și include același tip de diagrame care au fost prezentate în Figura 3.3.

Figura 3.4 prezintă diagrama pachet pentru modelul compresorului de aer și include pachete de cerințe, comportament, structură și parametri. Organizarea modelului urmează un tipar similar cu cel descris în secțiunea referitoare la SysML-Lite de mai sus și este ilustrat în Figura 3.3.

Pachetul de cerințe conține un set de cerințe care ar fi, în general, găsit într-un sistem al caietului de sarcini pentru compresorul de aer. Cerințele sunt capturate în diagrama de cerințe din Figura 3.5. Caietul de sarcini conține o cerință funcțională pentru a comprima aerul, cerințele de performanță care specifică presiunea maximă și debitul maxim, o cerință pentru a specifica cerințele de capacitate de stocare, puterea specificată de sursă de putere necesară pentru a comprima aerul și fiabilitatea. Textul cerinței capacității de stocare apare în diagrama, în timp ce textul pentru celelalte cerințe nu sunt afișate pentru a reduce dezordinea.

Pachetul conține o diagramă a comportamentului activității, reprezentat în Figura 3.6 și specifică modul în care interacționează compresorul de aer cu sistemele externe, inclusiv cu instrumentele de aer, cu cele de reglare a atmosferelor și, indirect, cu operatorul care reprezintă ca partii ale activității.

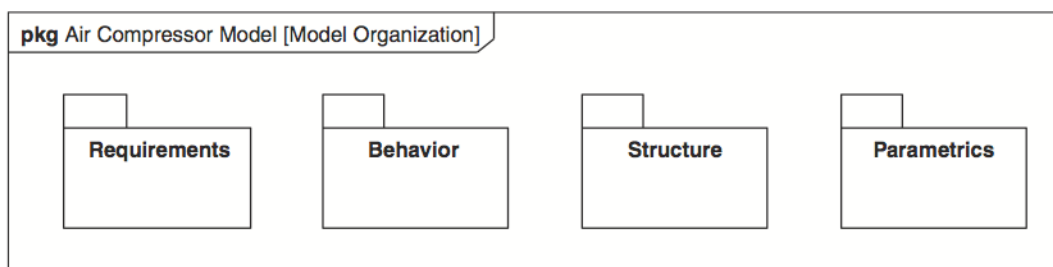


FIGURE 3.4

This package diagram is used to organize the *Air Compressor Model* into packages for *Requirements*, *Structure*, *Behavior*, and *Parametrics*. Each package contains model elements that can be related to model elements in other packages.

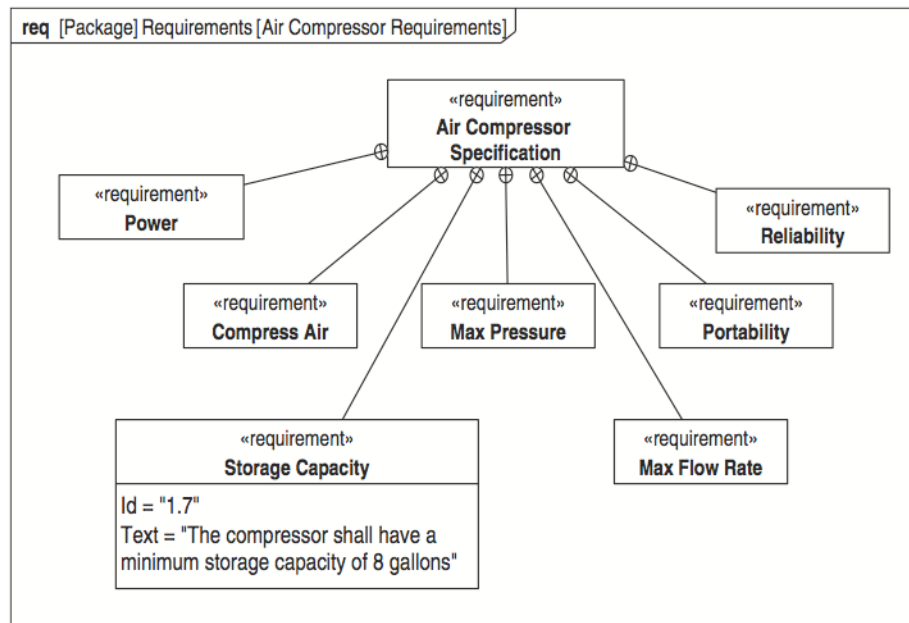


FIGURE 3.5

This requirement diagram represents the requirements contained in the *Requirements* package to specify the *Air Compressor*. Each requirement can include the requirements text that is typically found in a specification document.

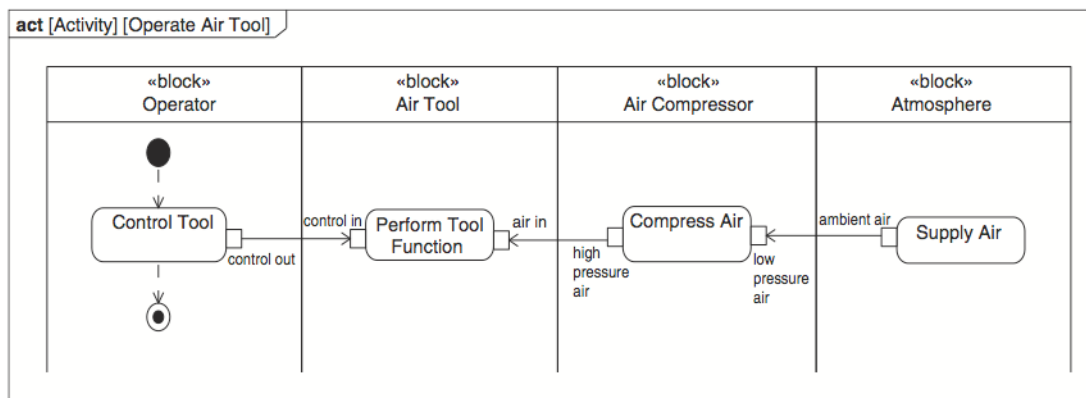


FIGURE 3.6

This activity diagram specifies the interaction between the *Air Compressor*, *Operator*, *Air Tool*, and *Atmosphere* to execute the *Operate Air Tool* activity.

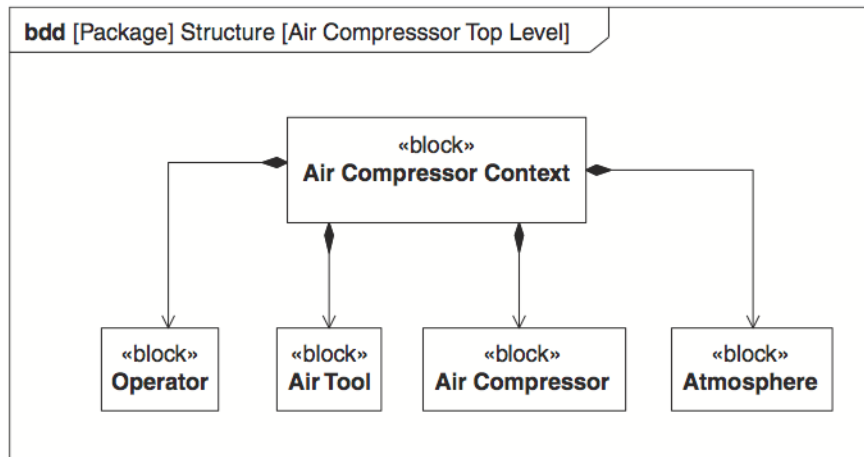


FIGURE 3.7

This block definition diagram represents the *Air Compressor*, *Operator*, *Air Tool*, and *Atmosphere* as blocks. The *Air Compressor Context* block sets the context for the *Air Compressor* and its external environment.

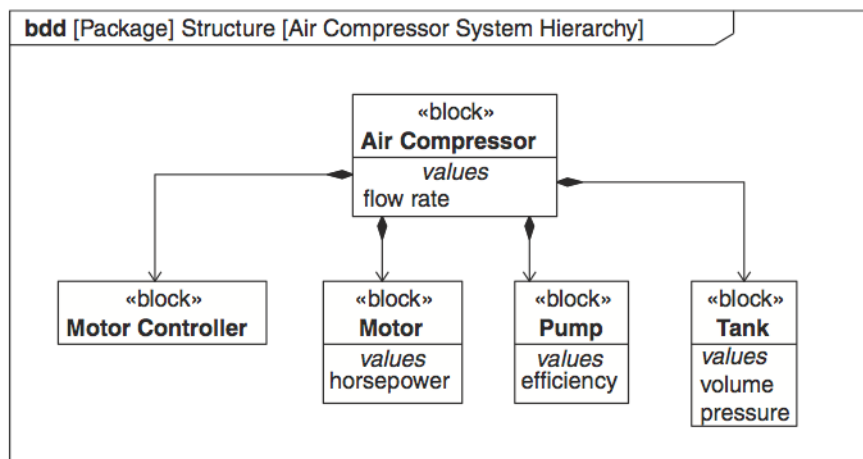


FIGURE 3.8

This block definition diagram represents the *Air Compressor* and its components. The *Air Compressor* block is the same block that is in Figure 3.7.

Compresorul de aer indeplineste functia care are un nivel scazut de intrare al presiunii aerului si un nivel ridicat al presiunii aerului la iesire. Activitatea incepe de la nodul initial, iar apoi operatorul executa actiunea controlului instrumentelor. Activitatea se completeaza cu executarea acesteia la nodul final de activitate, dupa ce actiunea de control instrumentelor ia sfarsit. Compresorul de aer este in continuare descompus in Figura 3.9.

Pachetul contine structura blocurilor reprezentate in diagramele definite in Figura 3.7 și Figura 3.8. Diagrama de definire a blocului din Figura 3.7 este numita compresor de aer de nivel superior si include un bloc numit compresor de aer contextual care este compus din compresorul de aer si blocurile aferente reprezentand sistemul extern, precum și mediul fizic. In acest exemplu, utilizatorul este operator, sistemul extern este instrumentul de aer, iar mediul fizic este atmosfera. Definitia schemei bloc din figura 3.8 se numeste Sistemul ierarhic al compresorului de aer. Blocul compresorului de aer, in aceasta figura este acelasi cu cel care este prezentată in Figura 3.7, dar aceasta figura se arata ca blocul compresor de aer este compus din componente cum ar fi controler-ul de motor, motorul, pompa si cisterna. Compresorul de aer, motorul, rezervorul si pompa au proprietati, care sunt utilizate pentru a analiza cerintele de debit.

In diagrama de activitate din Figura 3.9 se descompune actiunea numita comprimarea aerului din Figura 3.6 pentru a specifica modul in care componentele de compresorului de aer interactioneaza pentru a comprima aerul. Partitiile de activitate din diagrama de activitate reprezinta componentele compresorului de aer. Controler-ul motor include actiuni de presiune si control a motorului. Motorul genereaza cuplul, pompa pompeaza aerul, iar rezervorul pastreaza aerul. Presiunea scazuta de la intrare si presiunea inalta de la iesire este in concordanta cu intrarea si iesirea actiunii de comprimare a aerului din Figura 3.6. Aceasta activitate este continuta in pachetul de comportament impreuna cu instrumentele care opereaza activitatea de comprimare.

Schema bloc interna, numita de interconectare in Figura 3.10 prezinta modul in care componentele de compresor de aer din figura 3.8 sunt interconectate. Schema cadru a blocului compresorului de aer, precum si porturile de pe rama diagramei reprezinta interfetele externe ale compresorului de aer. Pieseile componente prezentate pe schema bloc interna sunt incluse in pachetul de structura, impreuna cu blocurile reprezentate pe diagrama de definitie a blocului.

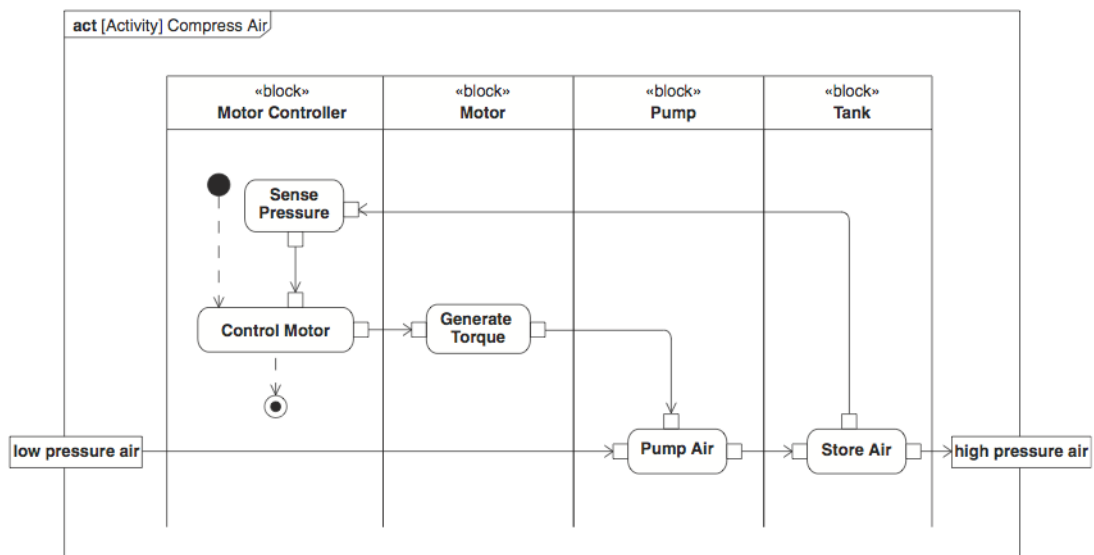


FIGURE 3.9

This activity diagram shows how the components of the *Air Compressor* interact to perform the *:Compress Air* action from Figure 3.6.

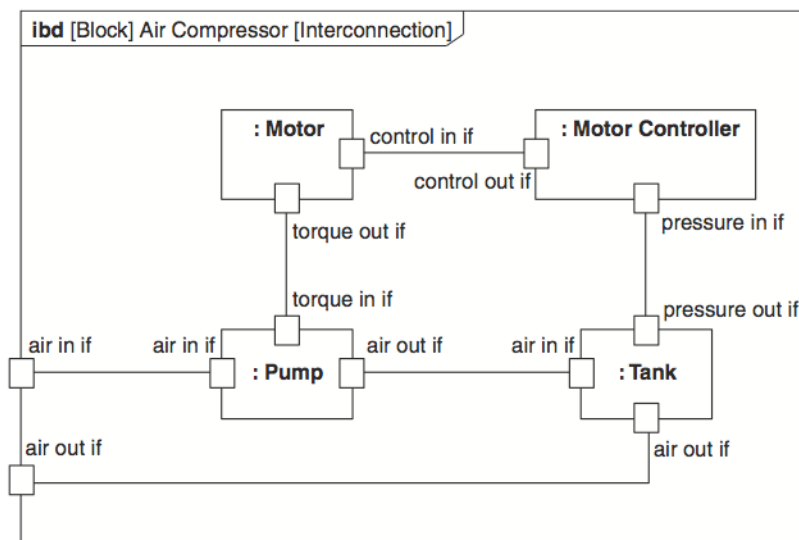


FIGURE 3.10

This internal block diagram shows how the components of the *Air Compressor* are interconnected via their ports, which are used to specify the component interfaces.

Schema de definire bloc numita, contextul de analiza reprezentata in figura 3.11, este utilizata pentru a defini contextul pentru efectuarea analizei debitului. In special, aceasta include un bloc numit rata de analiza a fluxului, care este compus dintr-un bloc de constrangere numit ecuatiile debitului. Acest bloc de constrangere defineste

parametrii ecuatiei si ecuatiile, dar ecuatiile nu sunt specificate in acest moment. Blocul de analiza al debitului se refera la blocul compresor de aer din Figura 3.7, care este obiectul analizei. Definirea contextului de analiza permite o diagrama parametrica care urmeaza sa fie creata pentru a afisa blocul de analiza al debitului asa cum se arata in figura 3.12. Diagrama arata proprietatile compresorului de aer si a partilor sale, care includ debitul, volumul rezervorului si presiunea atmosferica, puterea motorului si eficientei pompei si modul in care acestea sunt executate parametrii ecuatiilor de debit. Rata ecuatiilor de analiza a debitului pot fi rezolvate printr-un instrument de analiza, pentru a stabili valorile proprietatilor pentru compresorul de aer si a partilor componente ale acestuia.

Acest exemplu ilustreaza modul in care compresorul de aer poate fi modelat cu un subset de diagrame SysML si caracteristici lingvistice numite SysML-Lite. Chiar si un model simplu, cum ar fi acest lucru, poate contine multe elemente ale modelului si devin dificil de gestionat. Un instrument de modelare este necesar pentru a construi un model eficient care sa fie coerent si sa gestioneze cu usurinta complexitatea sistemelor.

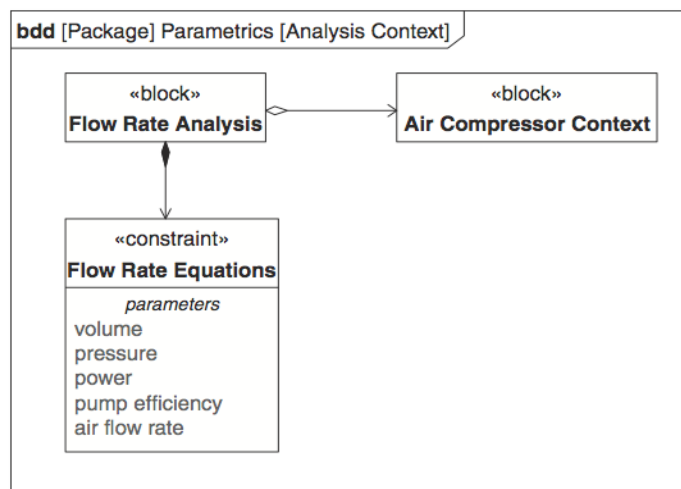


FIGURE 3.11

This block definition diagram is used to specify the *Flow Rate Analysis* in terms of a constraint block that defines the equations and parameters for the analysis (equations not shown), and the *Air Compressor Context* which is the subject of the analysis.

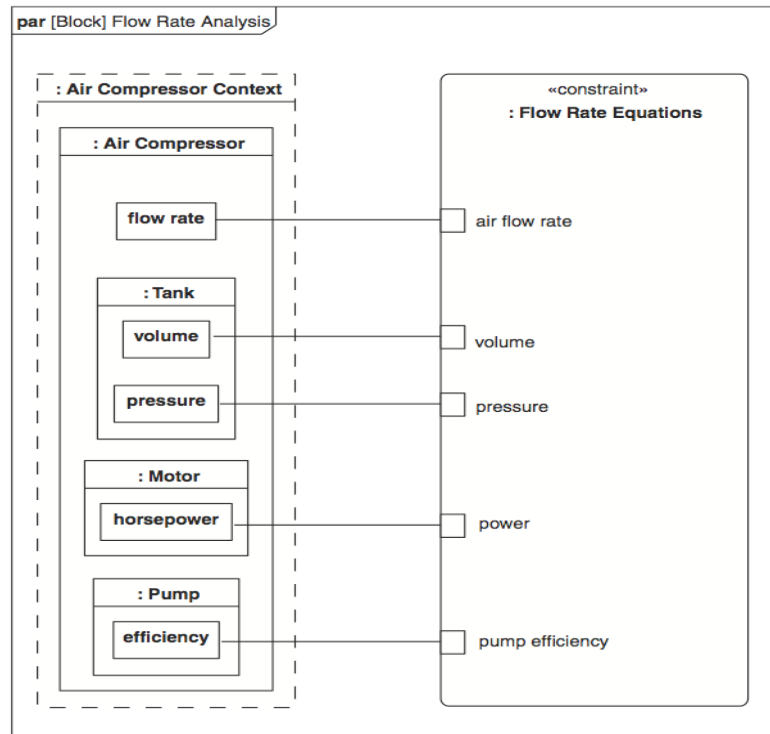


FIGURE 3.12

This parametric diagram represents the *Flow Rate Analysis*, and how the parameters of the equations are bound to the properties of the design. Once captured, this analysis can be provided to an analysis tool to perform the analysis. The equations are not shown in the figure.

4, Concluzii

SysML este un limbaj grafi de uz general pentru sisteme ce pot include dispozitive hardware, aplicatii software, date, oameni, precum si alte elemente dintr-un mediu fizic. Limbajul poate fi folosit in toate stadiile de proiectare incepand de la cerinte, structura, comportament pana la parametrii, oferind astfel o descriere completa a sistemului proiectat, a componentelor acestuia, cat si a mediului in care acesta urmeaza sa fie folosit.

Limbajul include noua tipuri de diagrame fiecare dintre ele avand mai multe caracteristici specifice. Semantica limbajului ofera dezvoltatorului posibilitatea de a modela un sistem in care fiecare diagrama reprezinta un alt mod de a privi sistemul ce urmeaza a fi creat. Este prezenta si posibilitatea de a lega intre ele elemente care se afla in diagrame diferite. Pentru a usura procesul de invatare a fost creat SysML-Lite. Acesta contine doar sase din cele noua diagrame prezente in limbajul SysML, precum si un set redus de elemente pentru fiecare diagrama. Invatarea acestui numar redus de elemente si diagrame si stapanirea lor, duce la formarea unei baze solide pentru dezvoltarea folosind “limbajul mama” SysML.

SysML aduce posibilitatea reprezentarii sistemelor din mai multe perspective. Fiecare perspectiva este una complexa din punctul ei de vedere, dar in acelasi timp ofera consistenta modelului prin suprapunerea cu celelalte perspective. Invatarea limbajului SysML introduce propriile concepte si propria complexitate prin studierea proceselor, metodelor si a uneltelor folosite de acest limbaj. Totusi folosirea acestui limbaj duce la o mai buna intelegere a sistemului ce urmeaza a fi creat. Limbajul SysML, prin complexitatea sa, provoaca la proiectarea nu doar a sistemelor simple ci si a celor complex in care pot aparea multe variabile.

Bibliografie:

1.Sanford Friedenthal, Alan Moore, Rick Steiner - A Practical Guide to SysML 2nd Edition: The Systems Modeling Language;