

Tutorial:

System structure modeling using MagicDraw

Vlad Acretoaie
Technical University of Denmark
Courses 02341, 02264

Abstract

This tutorial provides instructions for system structure modeling using MagicDraw in the context of DTU courses 02341 and 02264.

The required UML diagrams and model elements are introduced based on the Library Management System example used in the lectures.

The following diagrams are featured:

| Diagram | UML Diagram Type |
|-----------------------------|-----------------------------|
| System Context Diagram | Use Case Diagram |
| Domain Architecture Diagram | Use Case Diagram |
| System Structure Diagram | Composite Structure Diagram |

Note: Existing model elements must be re-used whenever possible (i.e. it is redundant and usually wrong to create two actors, components, etc., with the same name).

Project structure

Before starting work on a model, it is good practice to create a clear package structure that indicates where a particular diagram belongs.

This helps separate the various abstraction levels of the system design such as its domain analysis, its static structure, its behavior, and its deployment information.

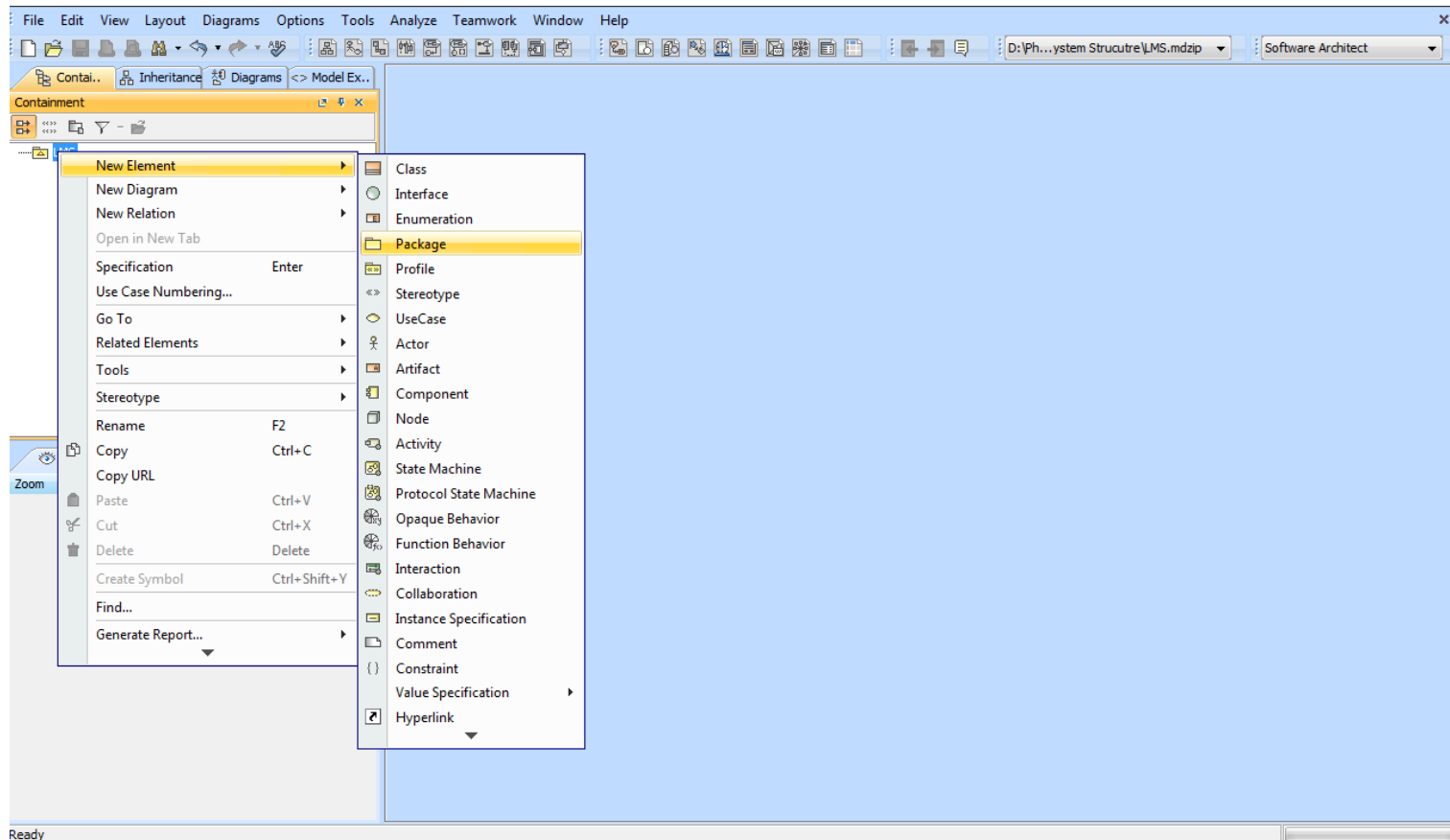
Your project should have the same package structure as the Library Management System (LMS) case study used throughout the lectures.

Considering the package structure of the LMS case study, the system structure models presented in this tutorial should be placed in a package called “System Parts”.

Project structure:

Creating a package

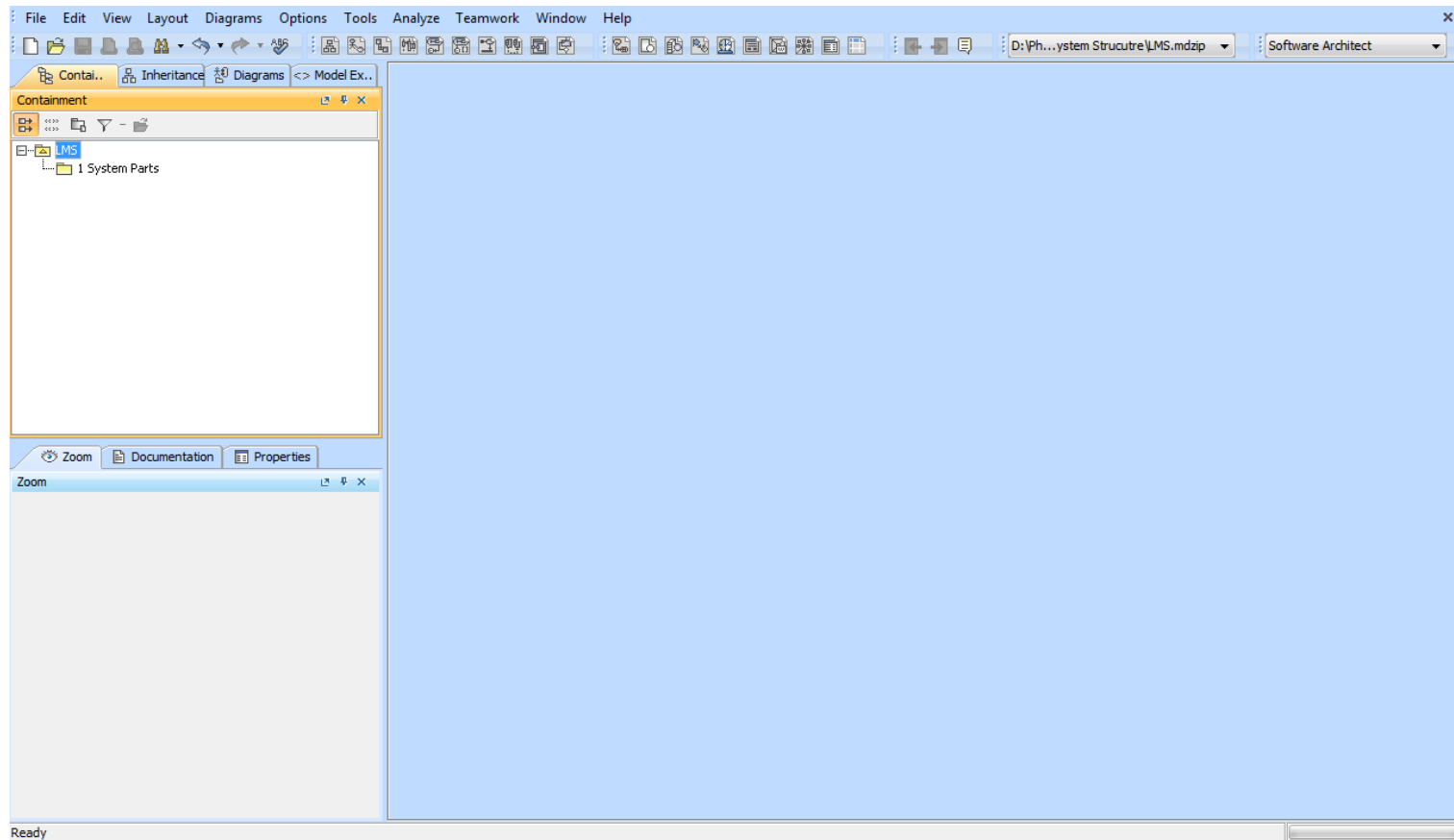
A new **Package** is added from the context menu of the containment hierarchy. Remember to also re-name the containment hierarchy root to your project name.



Package Structure

Creating a package

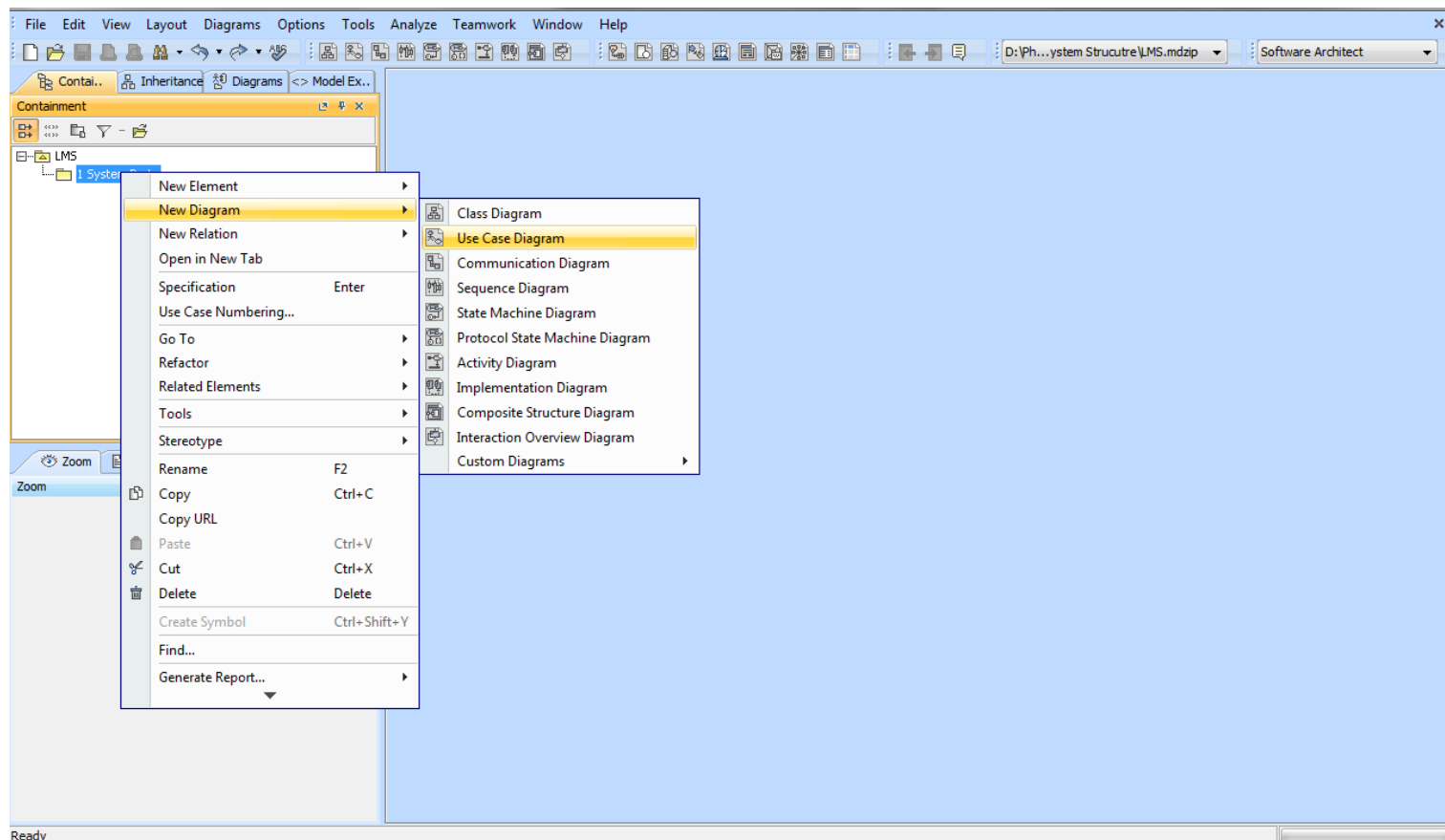
After adding the “System Parts” package, the containment hierarchy should look like the one below (but have your specific top-level package name instead of LMS).



1. System Context Diagram

Step 1: Create the diagram

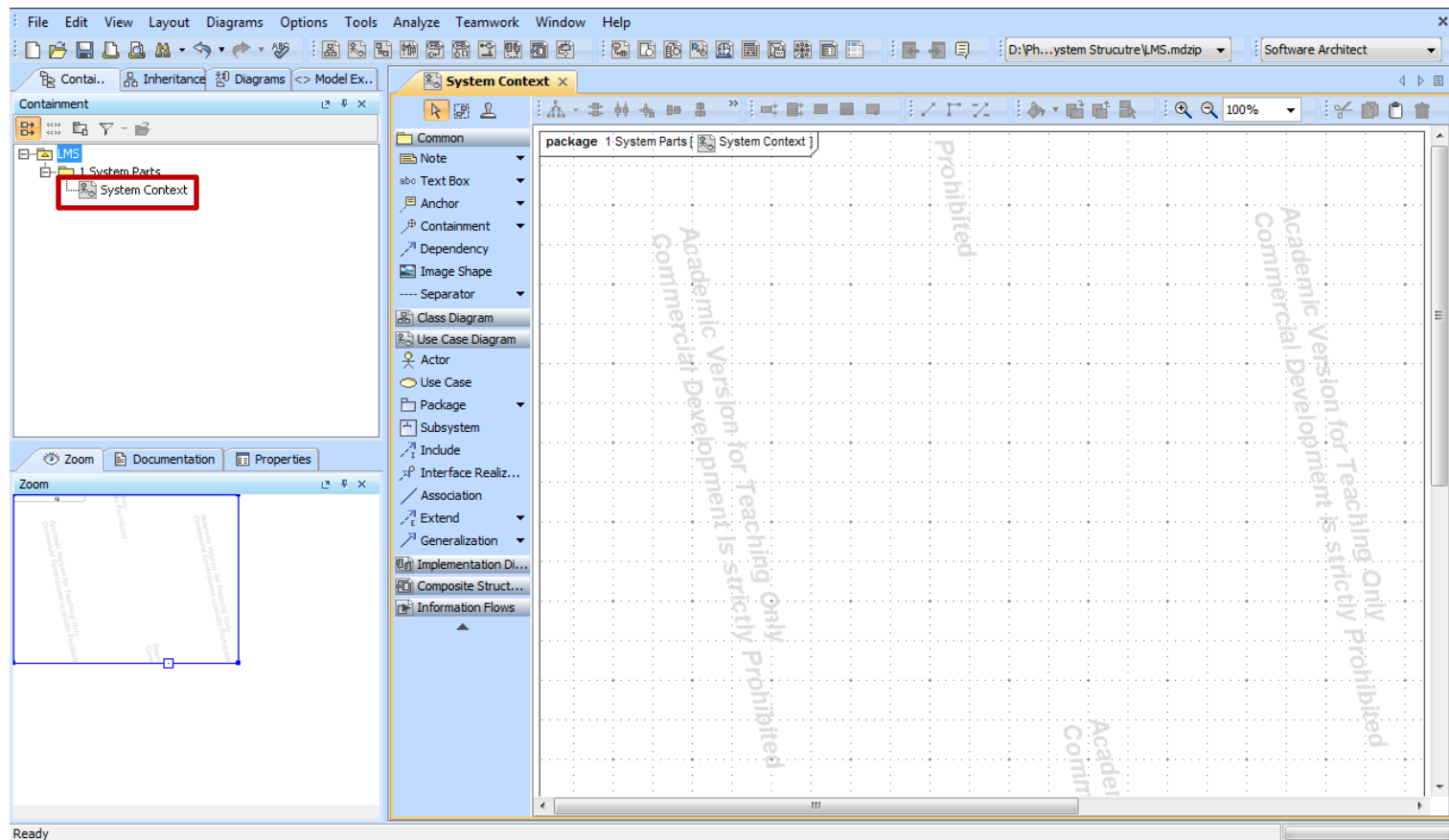
A new **Use Case Diagram** representing the **System Context** is added to the System Parts package from its contextual menu.



1. System Context Diagram

Step 1: Create the diagram

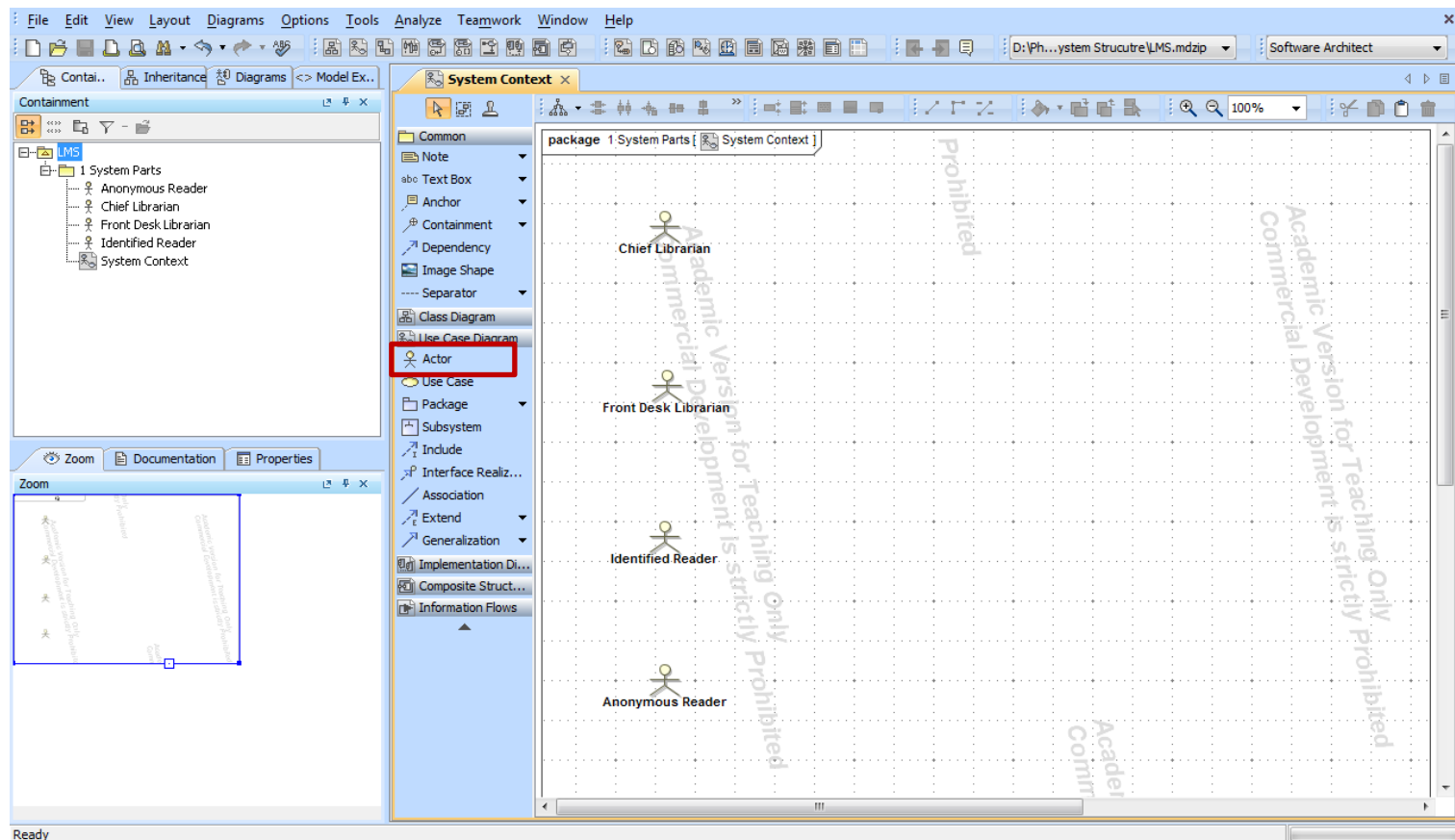
The created **System Context** diagram is highlighted in the project containment hierarchy in red below.



1. System Context Diagram

Step 2: Add actors

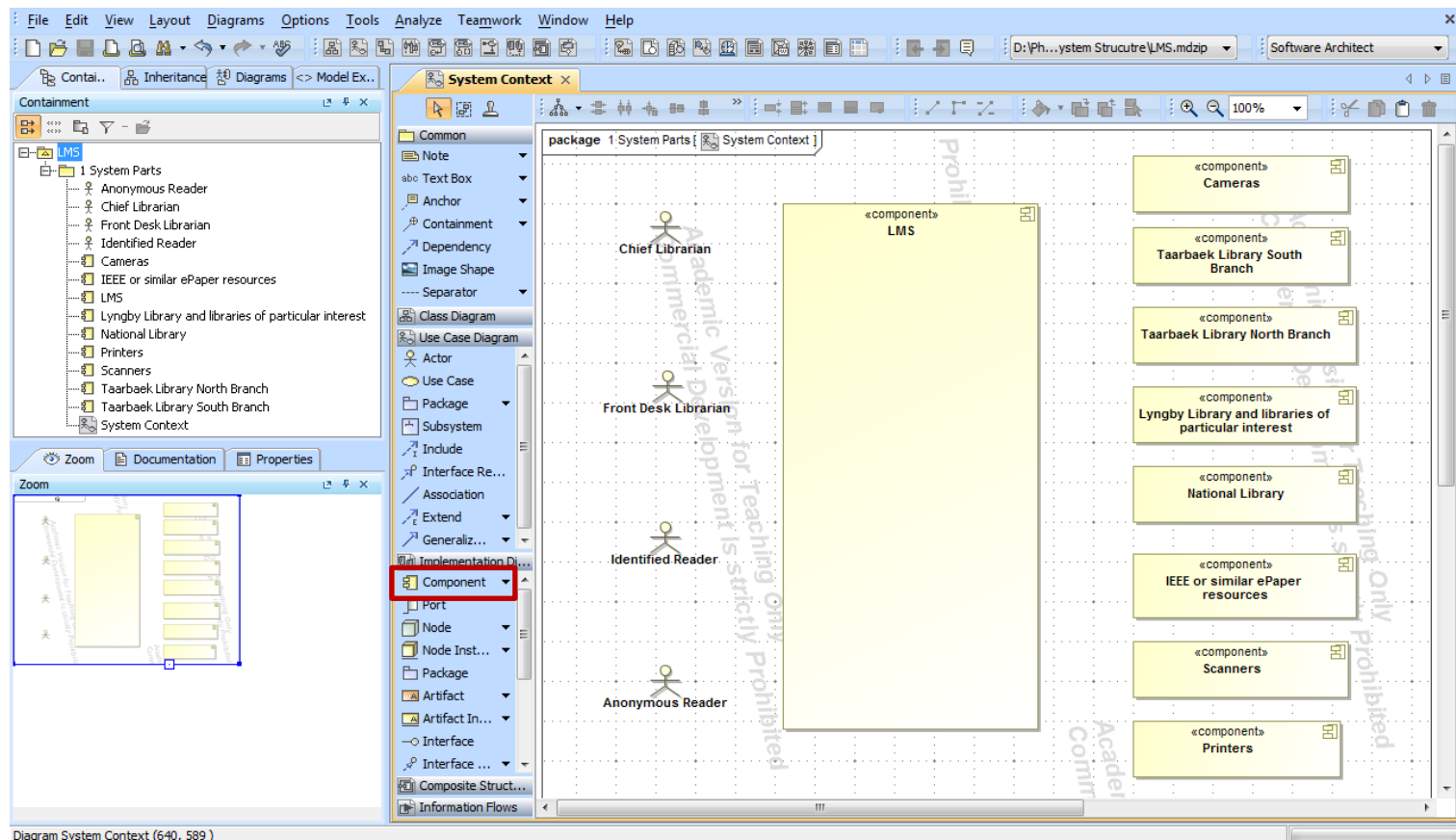
Actors included in the system context are added to the diagram using the toolbar button highlighted in red below.



1. System Context Diagram

Step 3: Add components

Components included in the system context are added to the diagram using the toolbar button highlighted in red below.



1. System Context Diagram

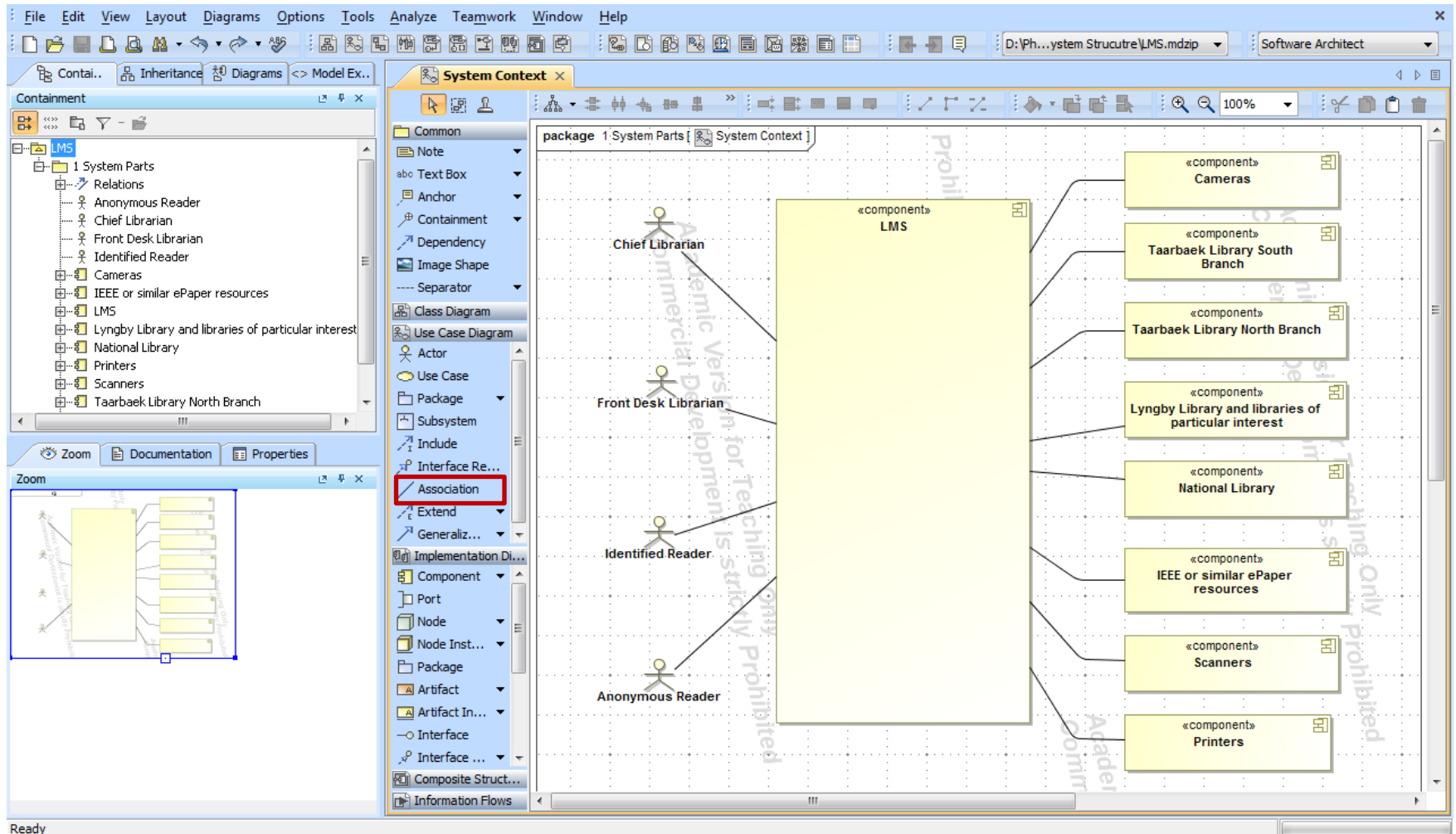
Step 4: Add associations

Associations between actors and components are added to the diagram using the toolbar button highlighted in red on the next slide.

Note: Actors and components are directly connected by associations. Ports are not required on the System Context diagram, since UML does not support connecting actors with ports on a Use Case diagram.

1. System Context Diagram

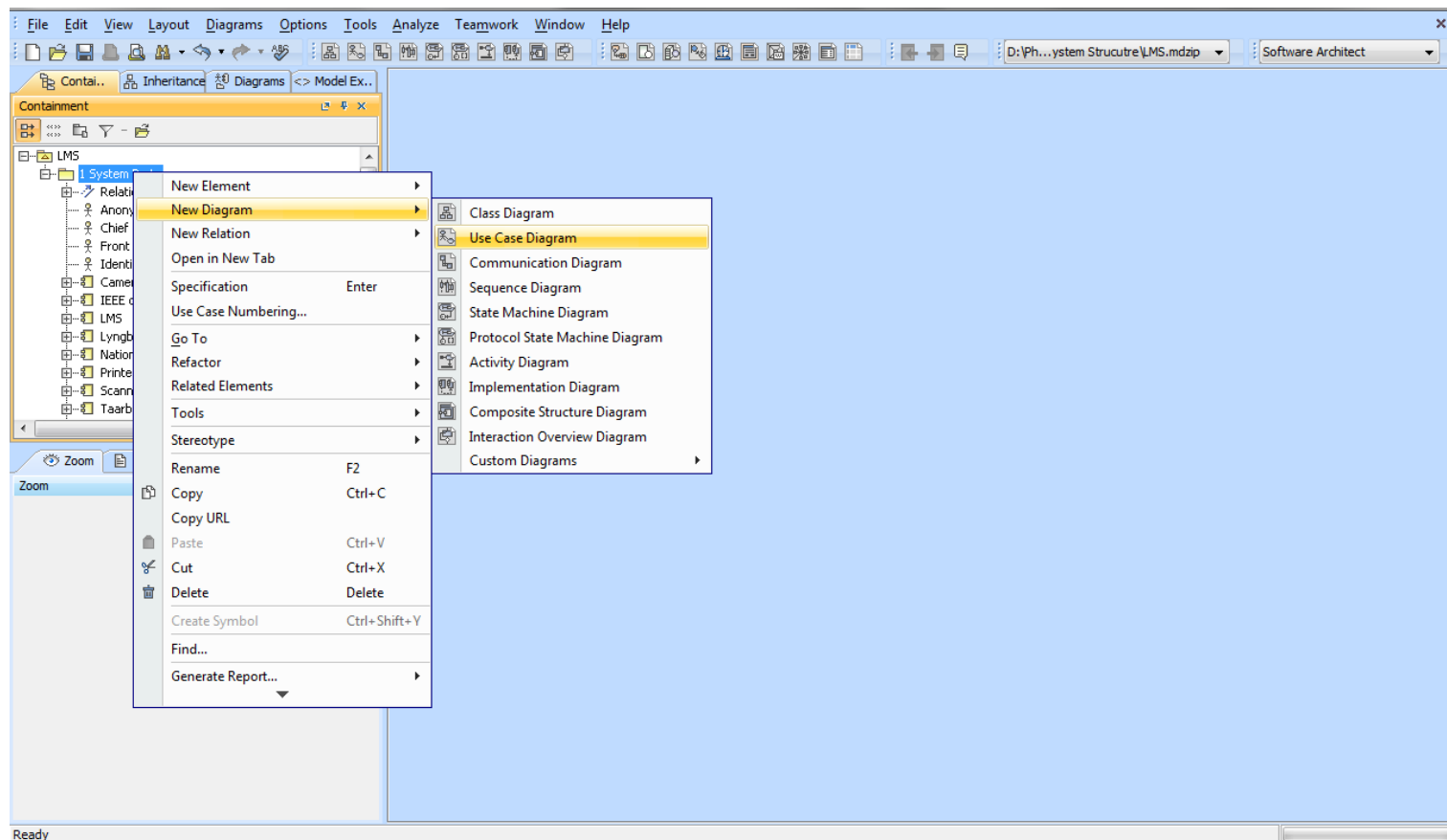
Step 4: Add associations



2. Domain Architecture Diagram

Step 1: Create the diagram

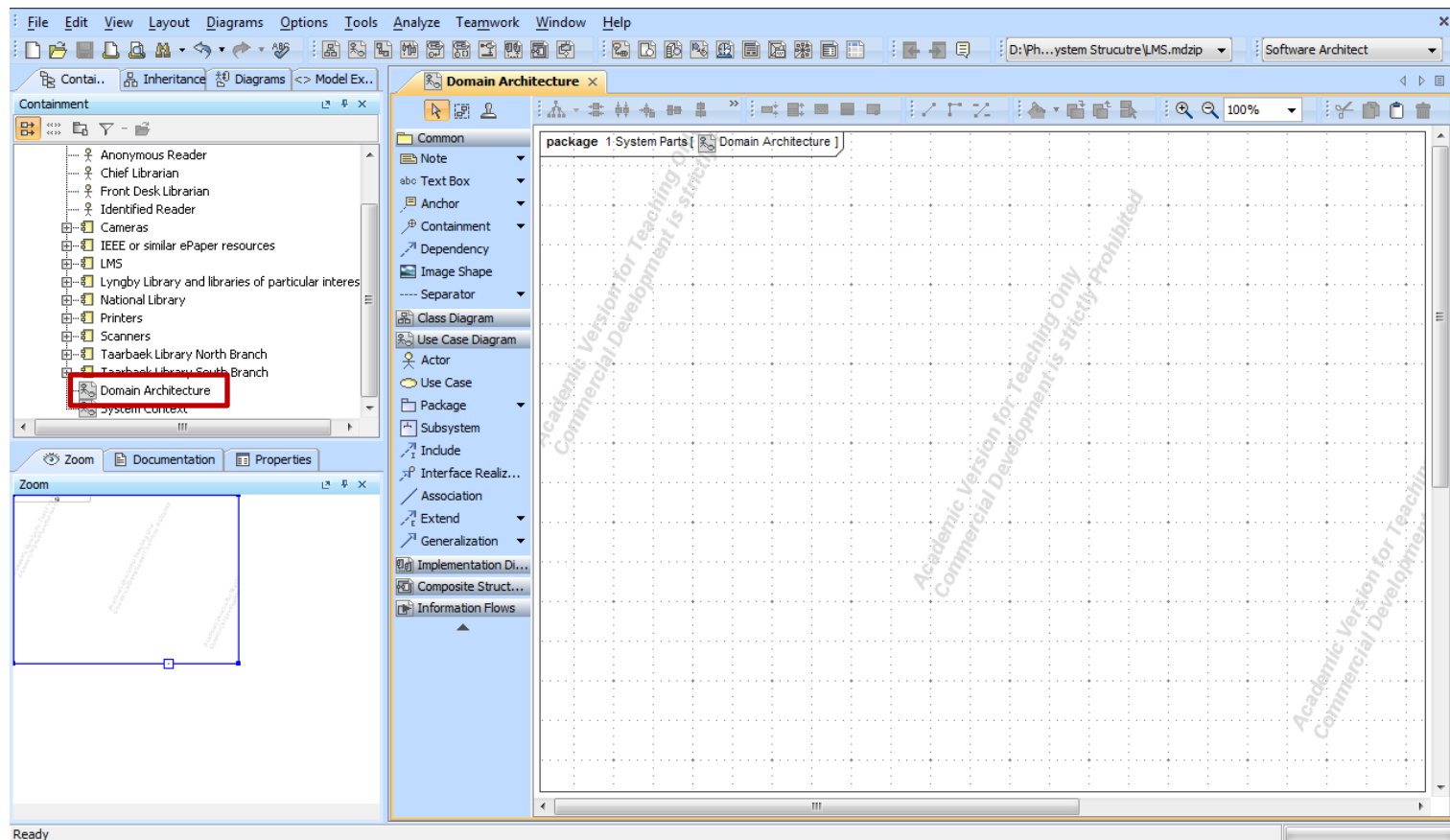
A new **Use Case Diagram** is representing the **Domain Architecture** is added to the System Parts package from its contextual menu.



2. Domain Architecture Diagram

Step 1: Create the diagram

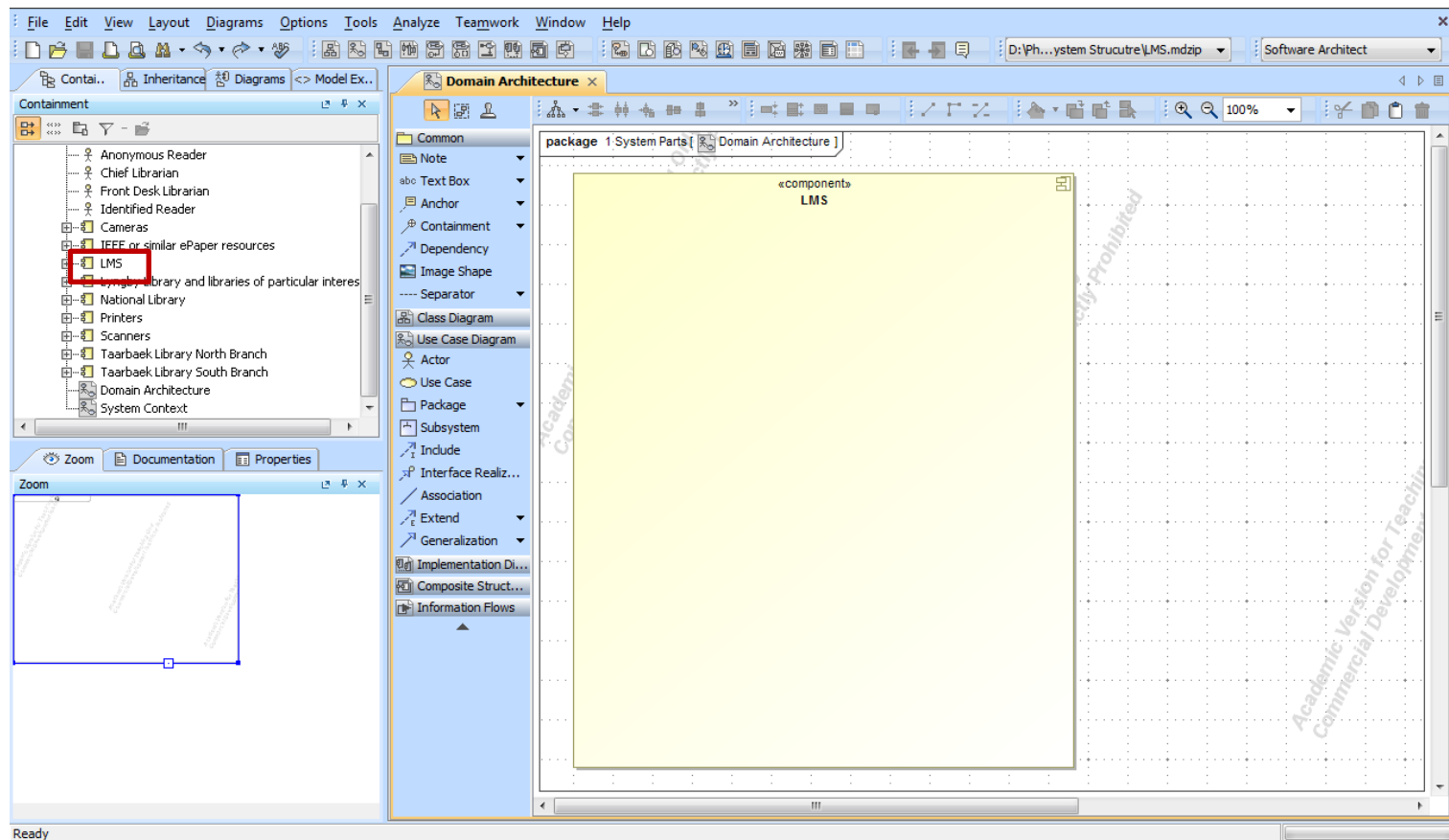
The created **Domain Architecture** diagram is highlighted in the project containment hierarchy in red below.



2. Domain Architecture Diagram

Step 2: Add the system

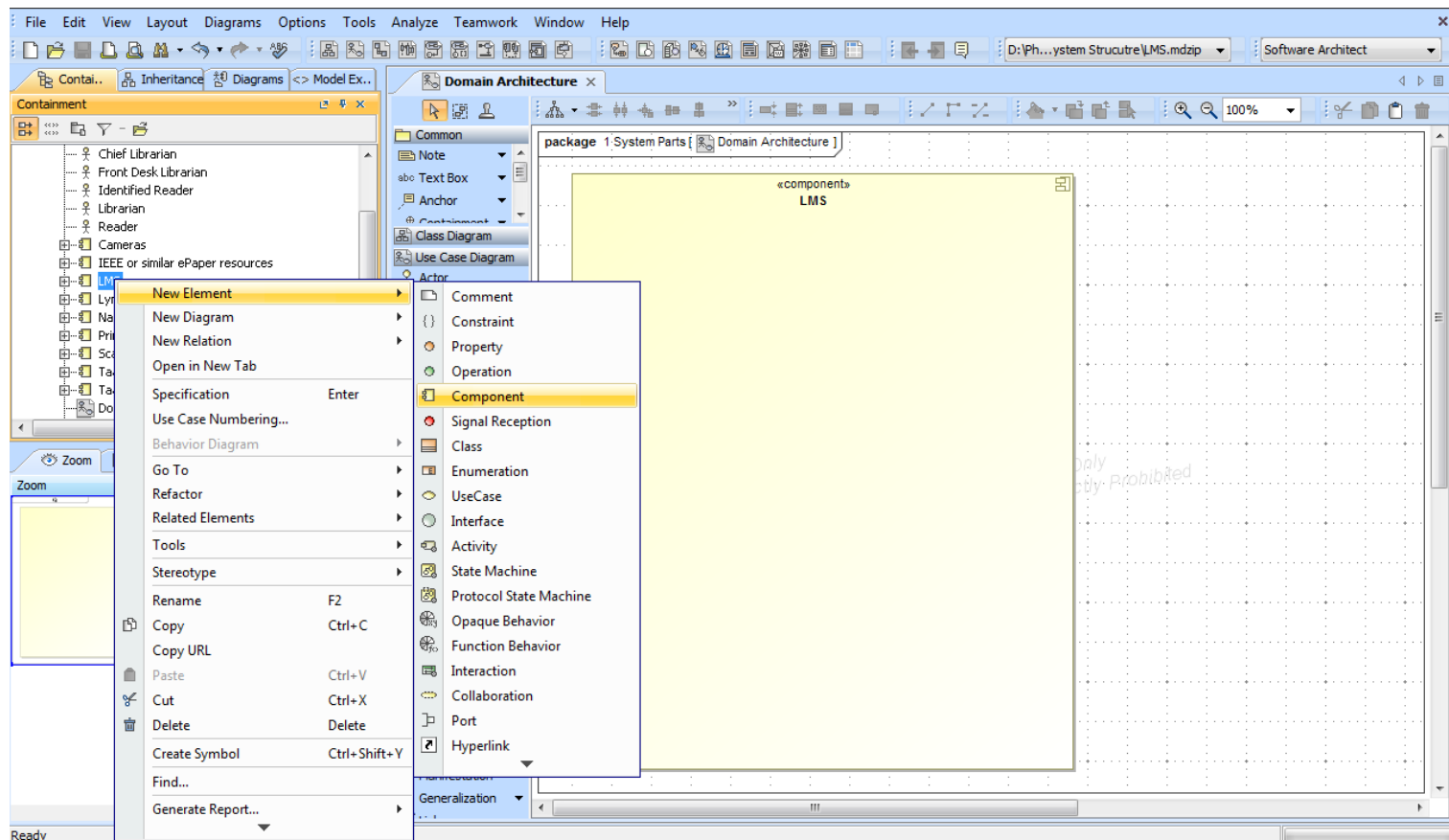
The **outermost domain (“the system”)** is added to the diagram by dragging it from the containment tree – in this example it is the LMS component highlighted in red.



2. Domain Architecture Diagram

Step 3: Add components

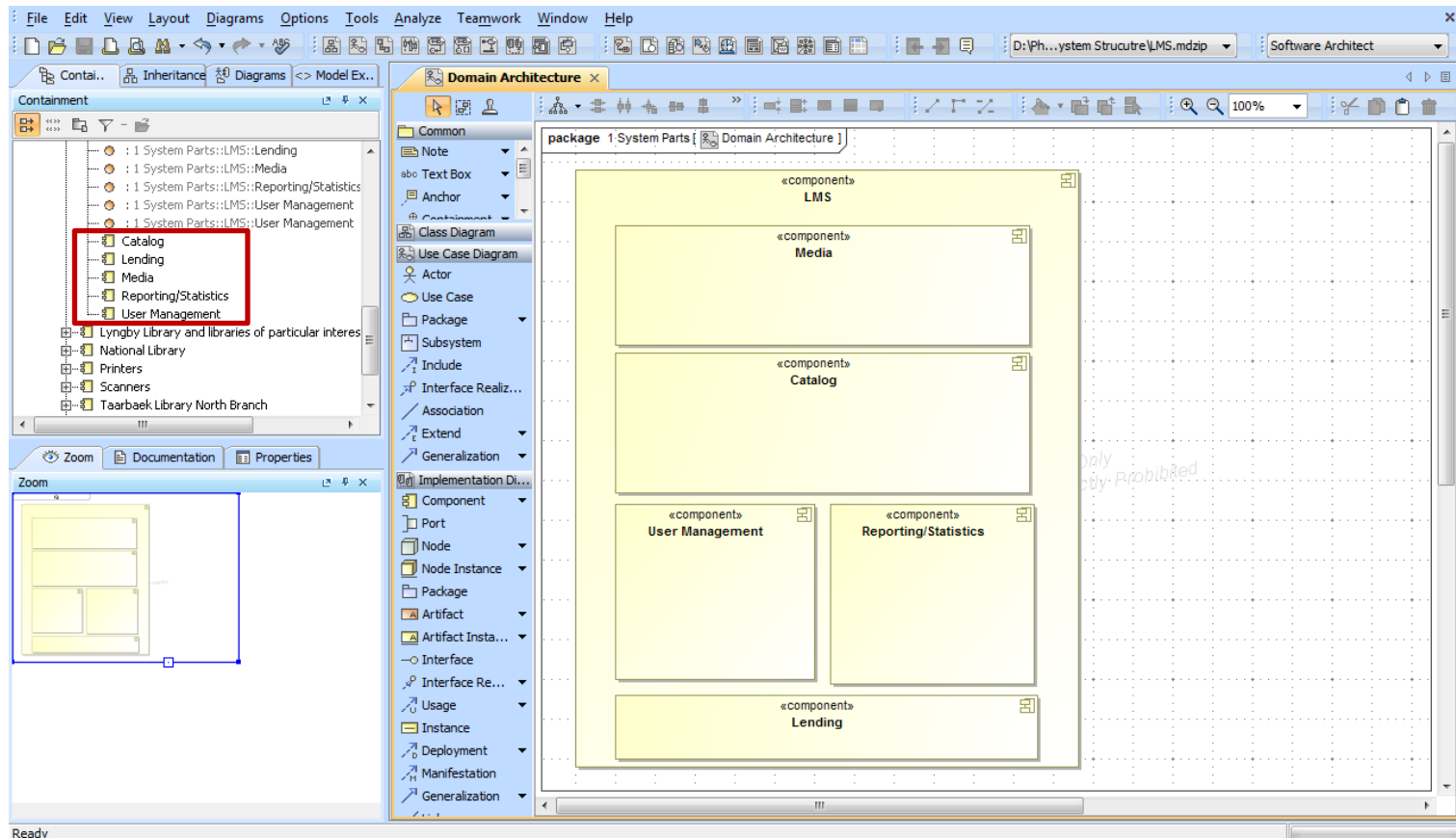
Components representing **sub-domains** are added from the context menu of the system (LMS). They must be nested under it in the containment hierarchy.



2. Domain Architecture Diagram

Step 3: Add components

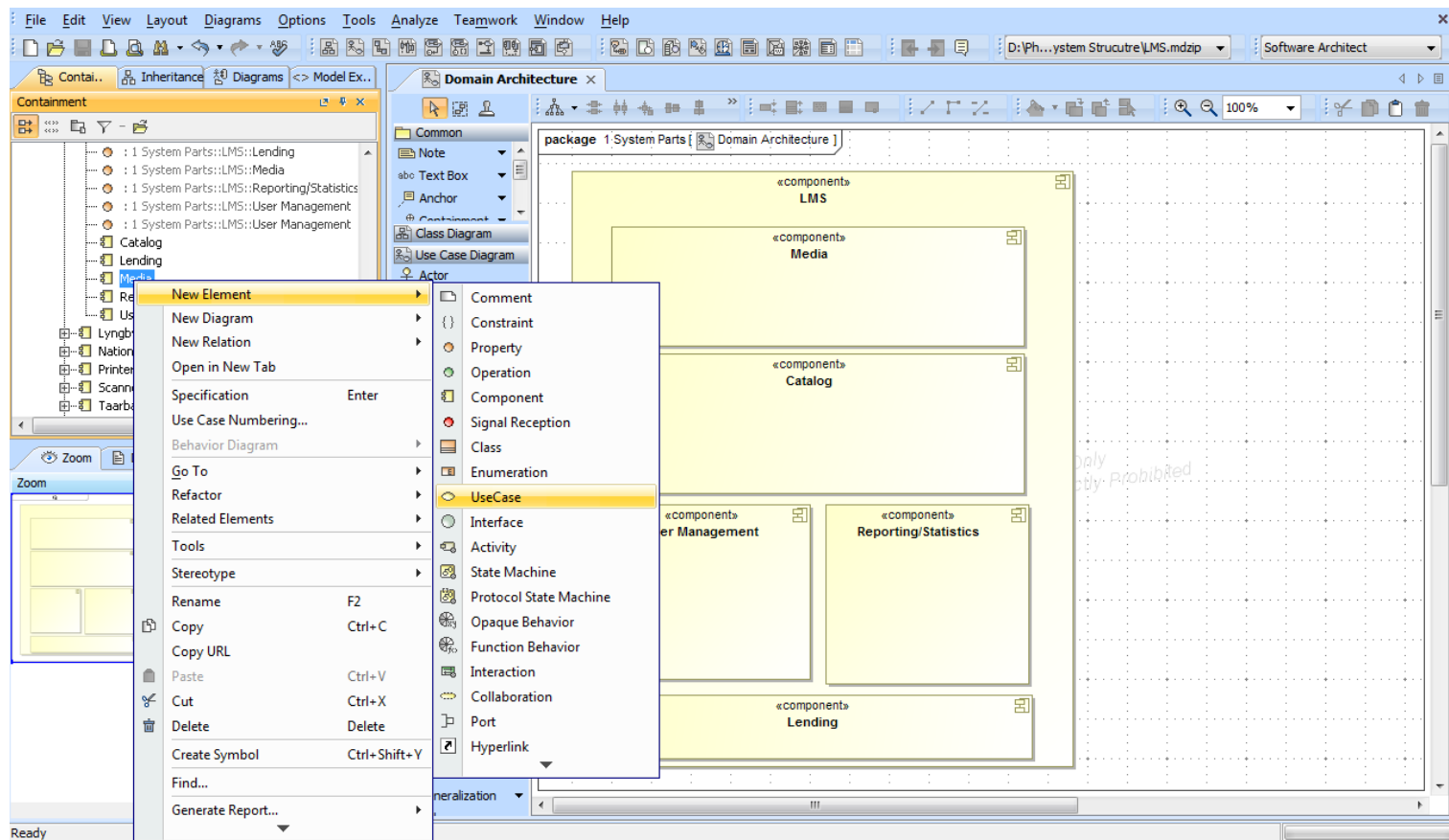
The sub-domains nested in the containment hierarchy under the outer domain are highlighted in red below.



2. Domain Architecture Diagram

Step 4: Add use cases

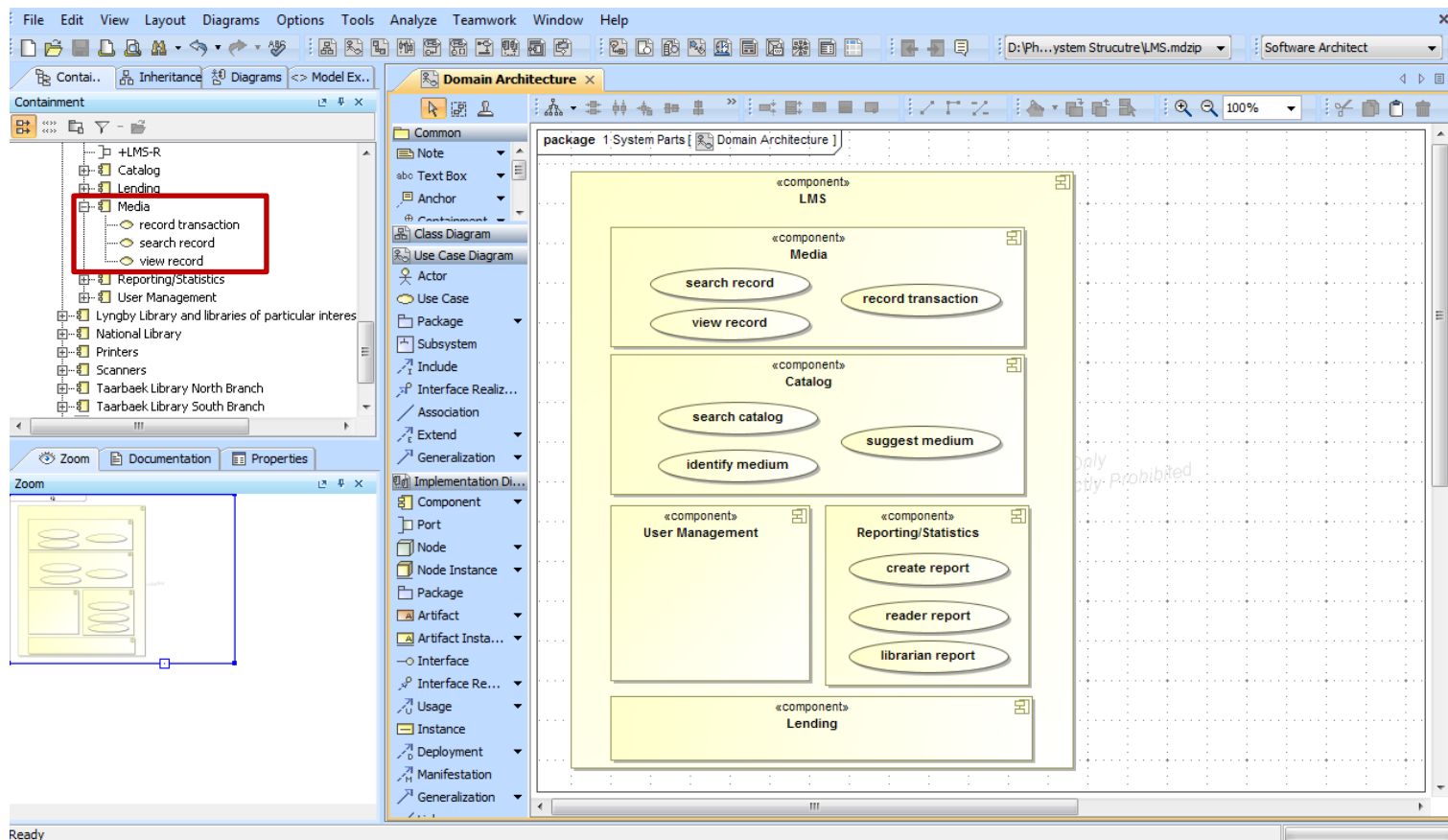
Use cases are added from the context menu of the sub-domain which realizes them. They must be nested under this domain in the containment hierarchy.



2. Domain Architecture Diagram

Step 4: Add use cases

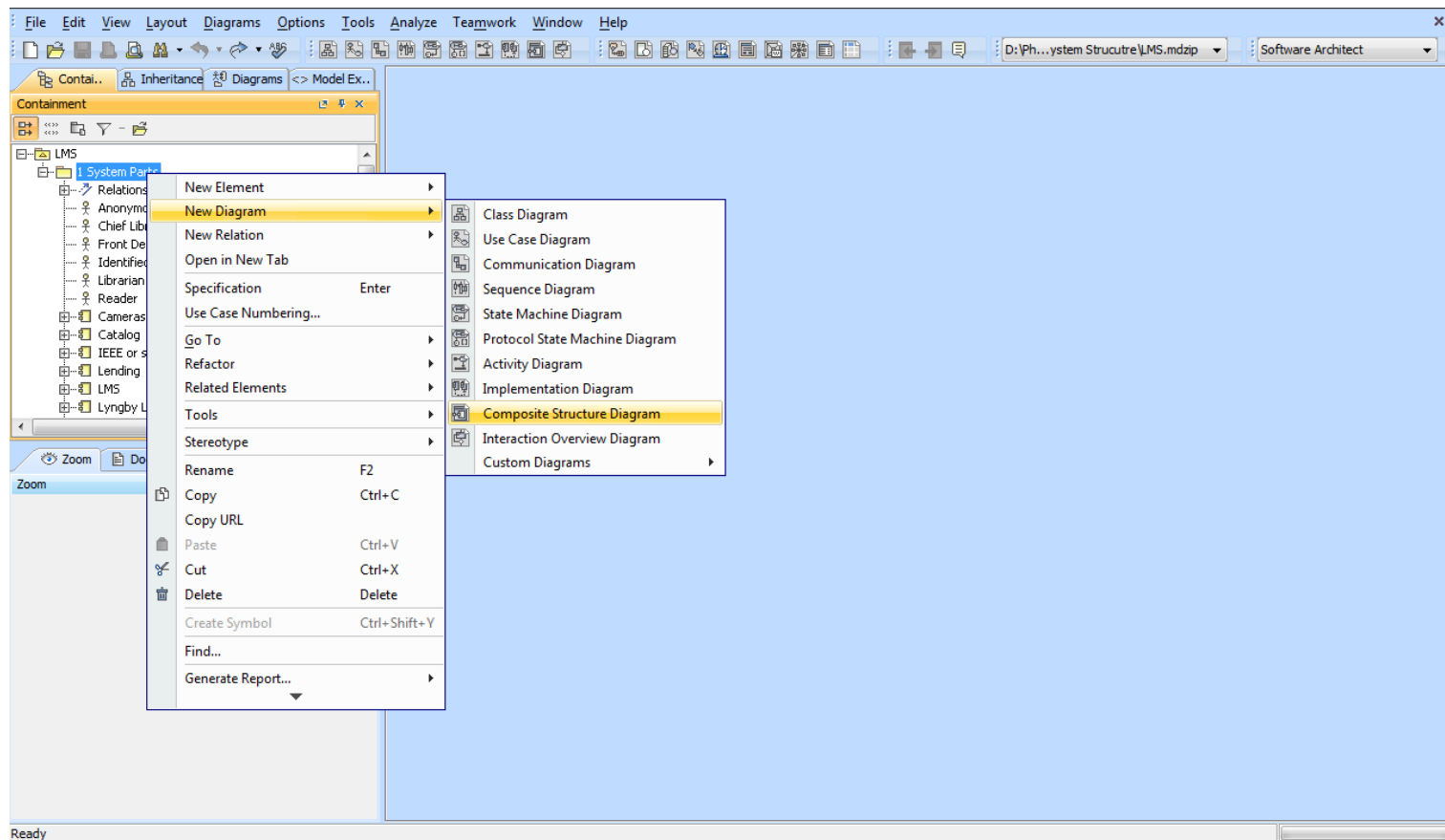
The use cases nested in the containment hierarchy under the Media domain are highlighted in red below. They are also visible on the diagram.



3. System Structure Diagram

Step 1: Create the diagram

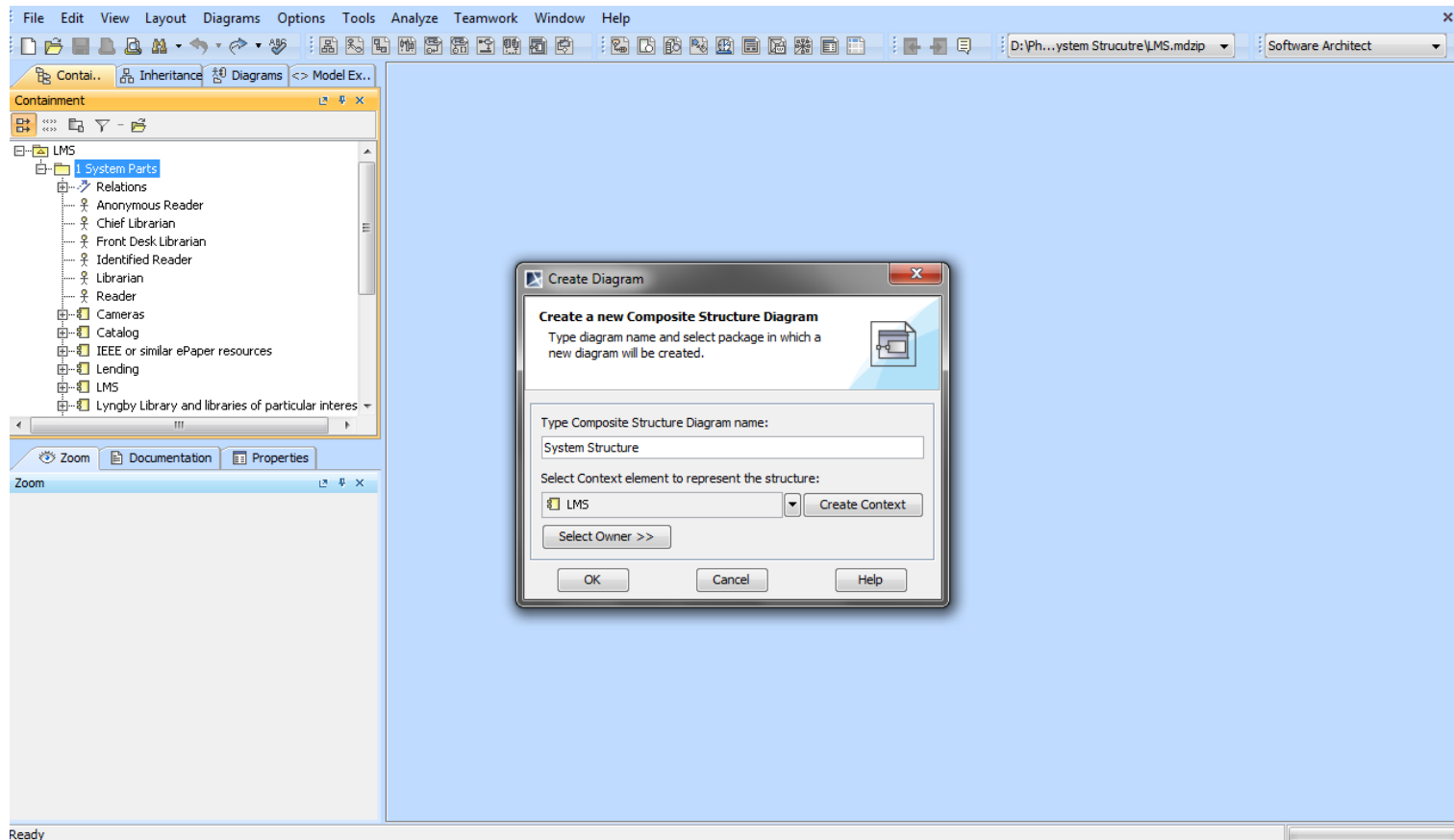
A new **Composite Structure Diagram** representing the **System Structure** is added to the System Parts package from its contextual menu.



3. System Structure Diagram

Step 1: Create the diagram

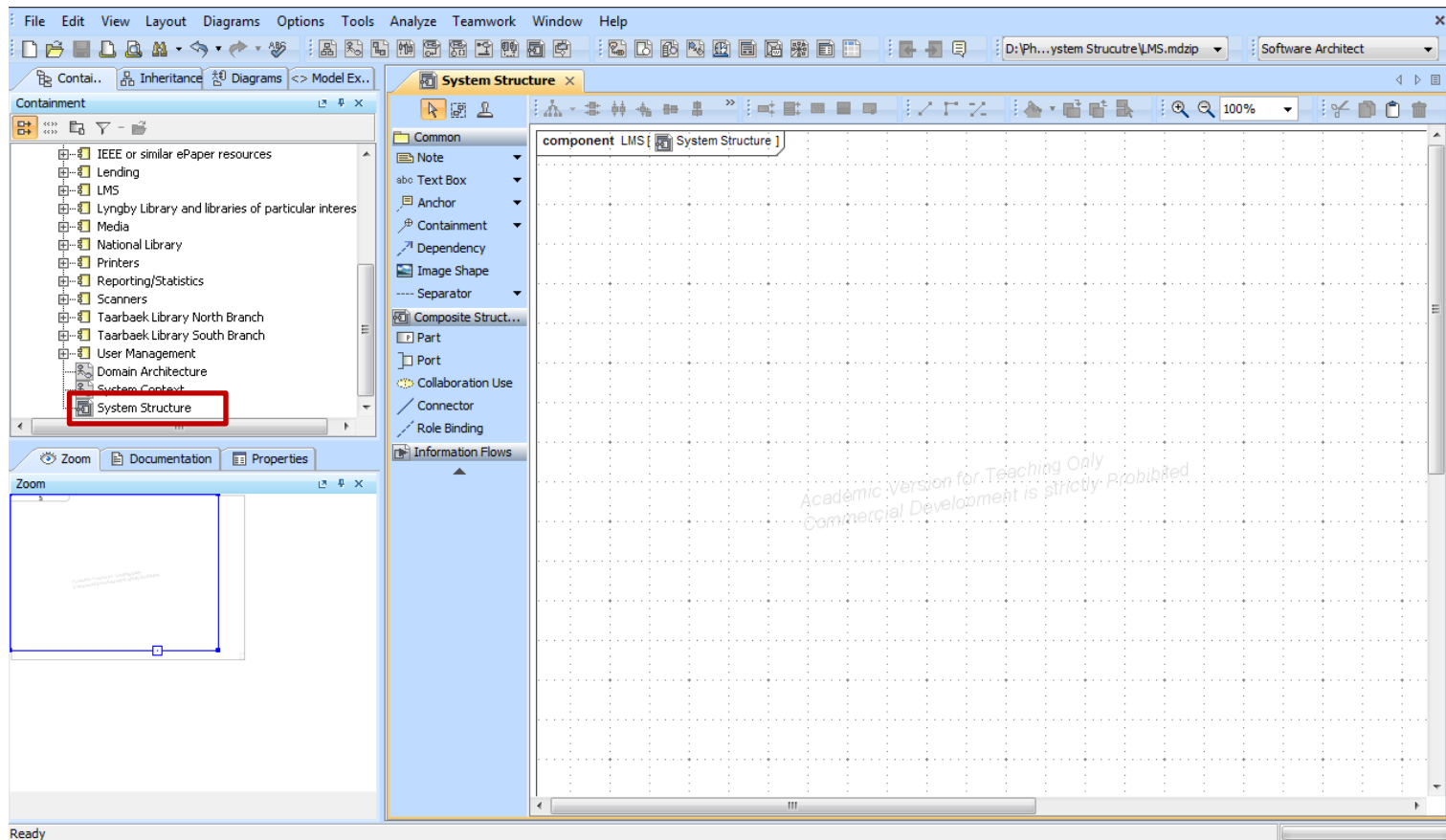
MagicDraw prompts for a name and context of the new diagram. The context must be specified as the system being modeled (in this example the LMS component).



3. System Structure Diagram

Step 1: Create the diagram

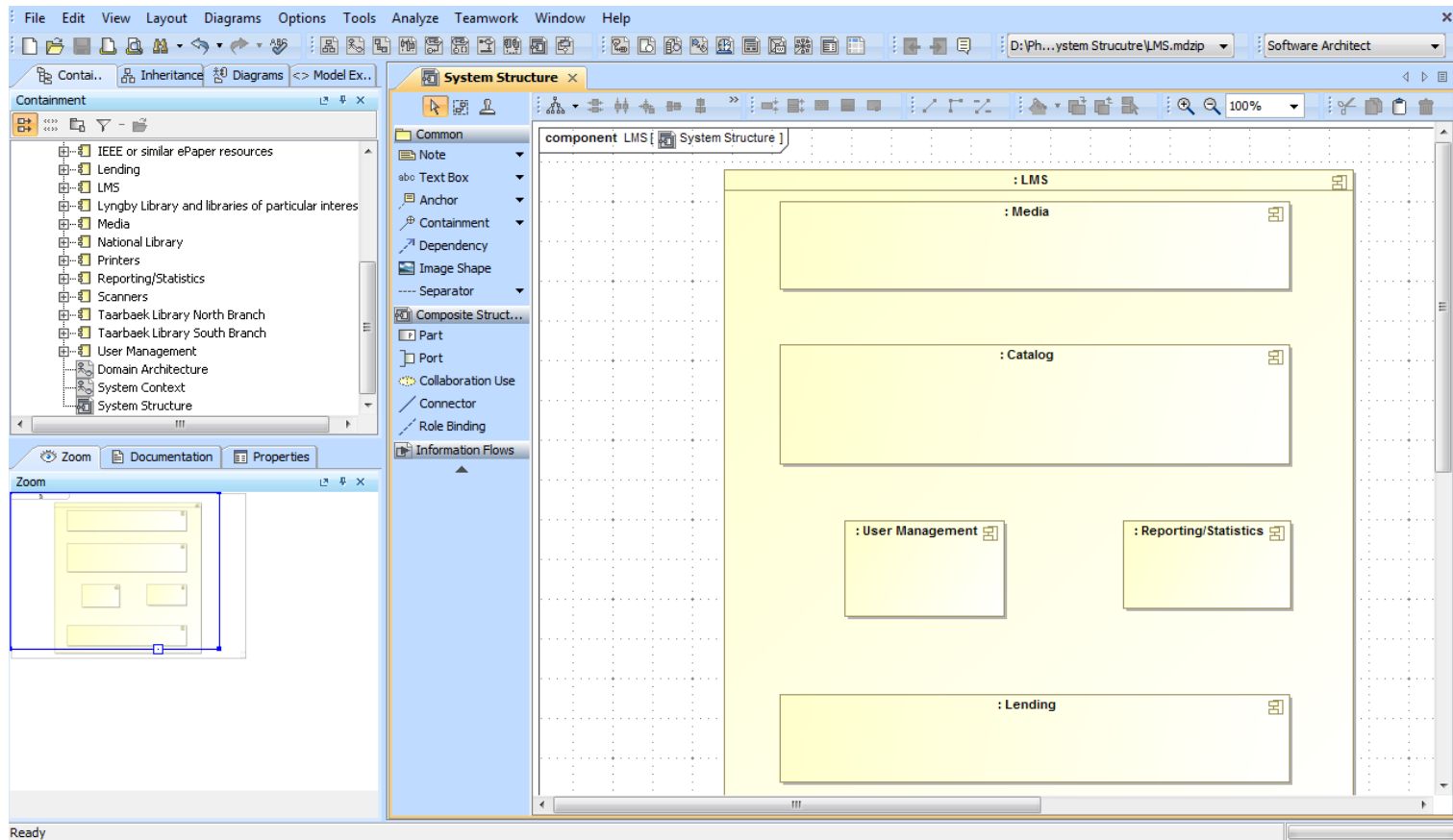
The created **System Structure** diagram is highlighted in the project containment hierarchy in red below.



3. System Structure Diagram

Step 2: Add system and subsystems

The system and sub-systems are dragged to the diagram from the containment hierarchy. They have already been created for the Domain Architecture diagram.



3. System Structure Diagram

Step 3: Add actors

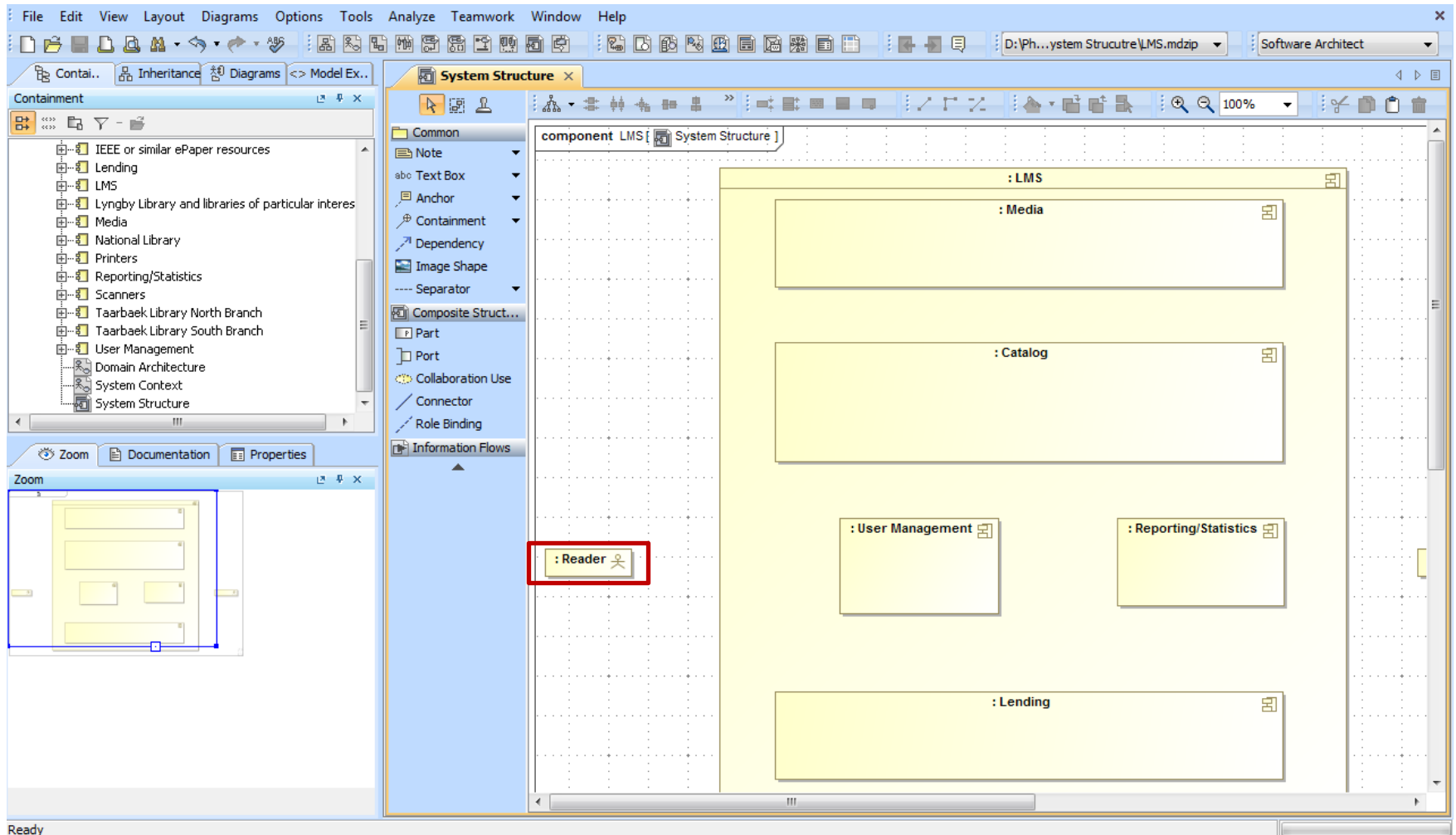
Actors that interact with the system are dragged to the diagram from the containment hierarchy.

They have already been created for the System Context diagram.

Notice the different concrete syntax (shape) that actors have in a Composite Structure diagram. Instead of stick figures they are now represented as rectangles, as shown by the Reader actor highlighted in red in the following screenshot.

3. System Structure Diagram

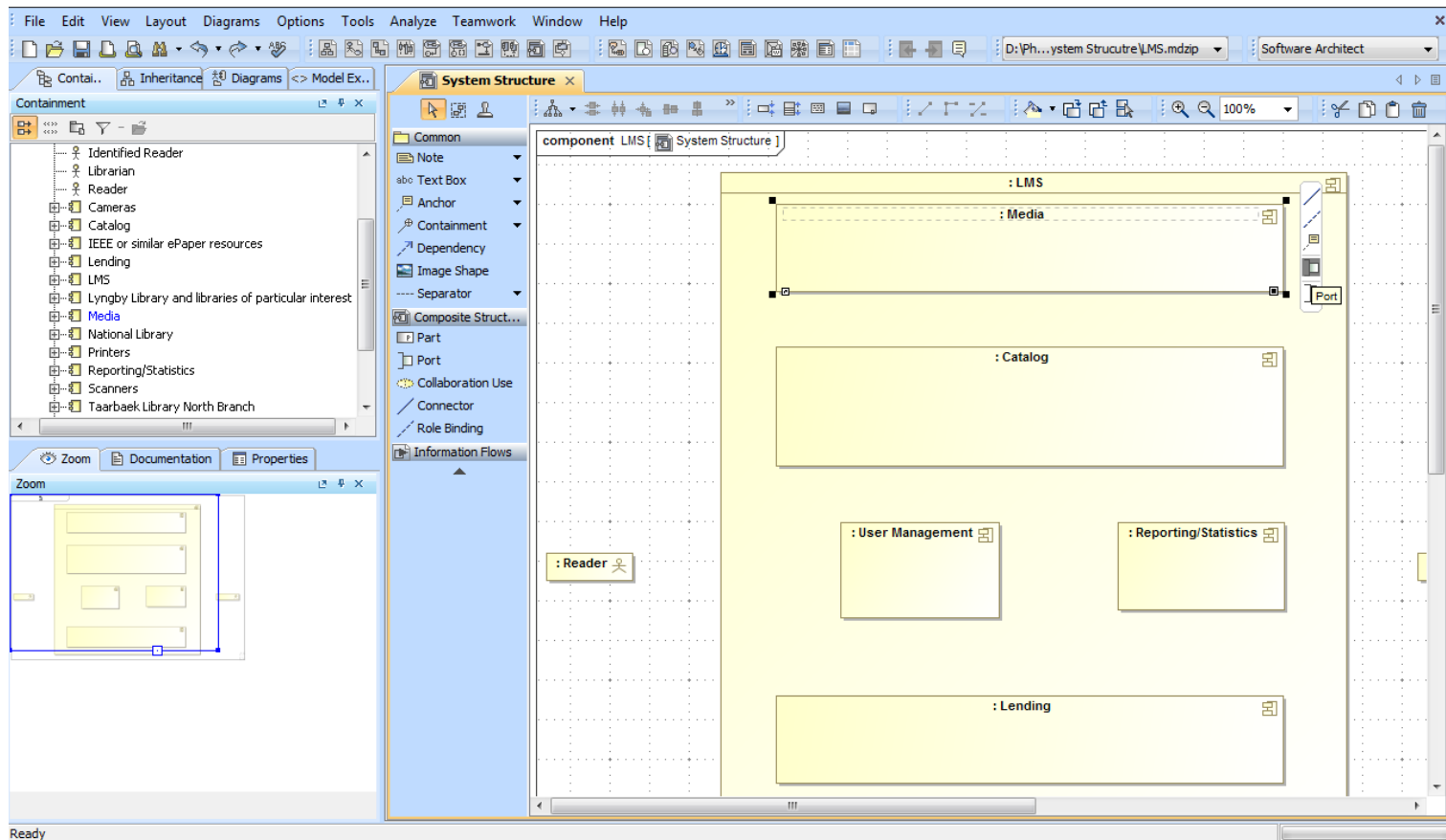
Step 3: Add actors



3. System Structure Diagram

Step 4: Add ports

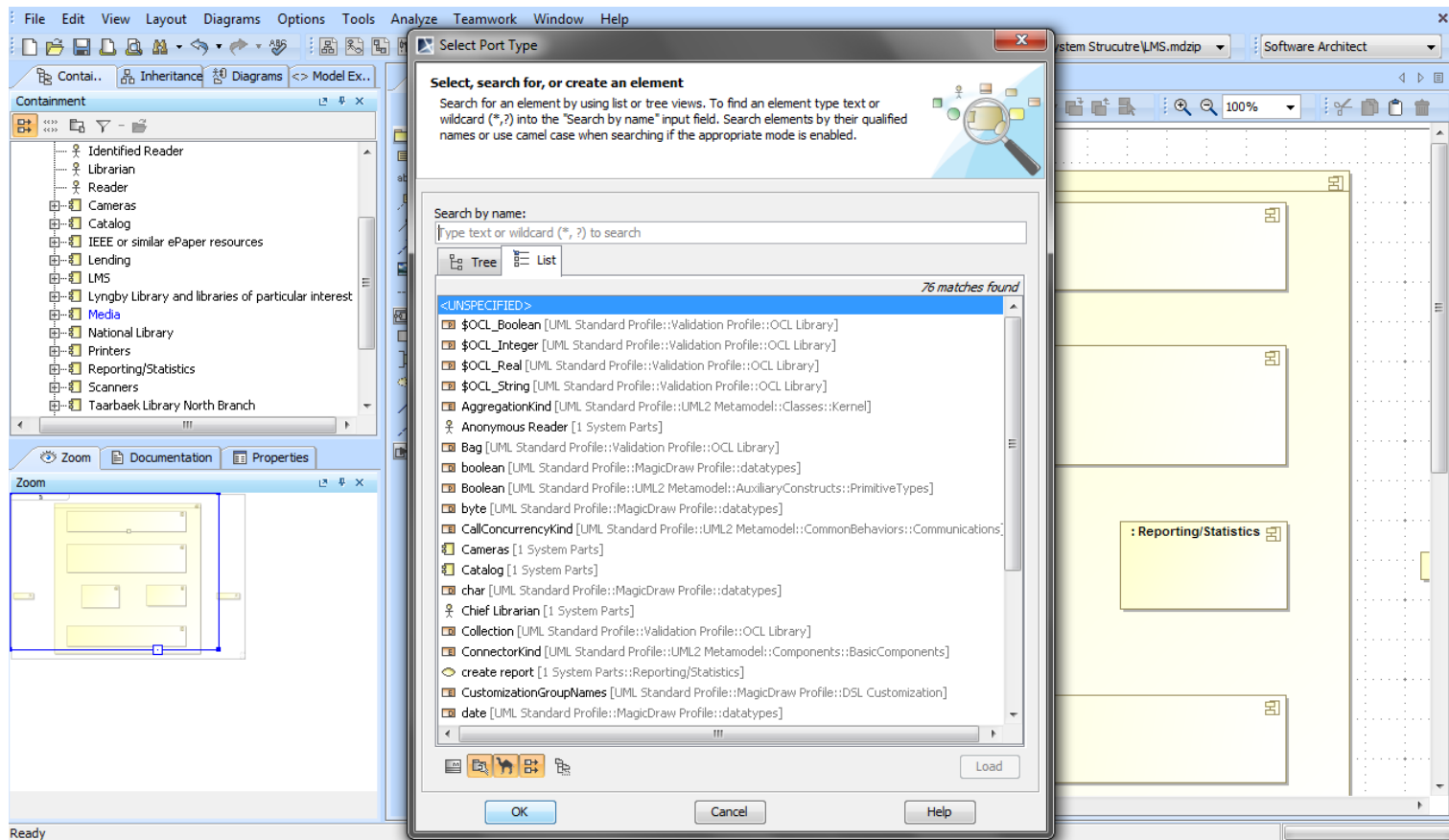
Ports are added by clicking on a system in the diagram and selecting the port entry from the pop-up menu that appears on screen.



3. System Structure Diagram

Step 4: Add ports

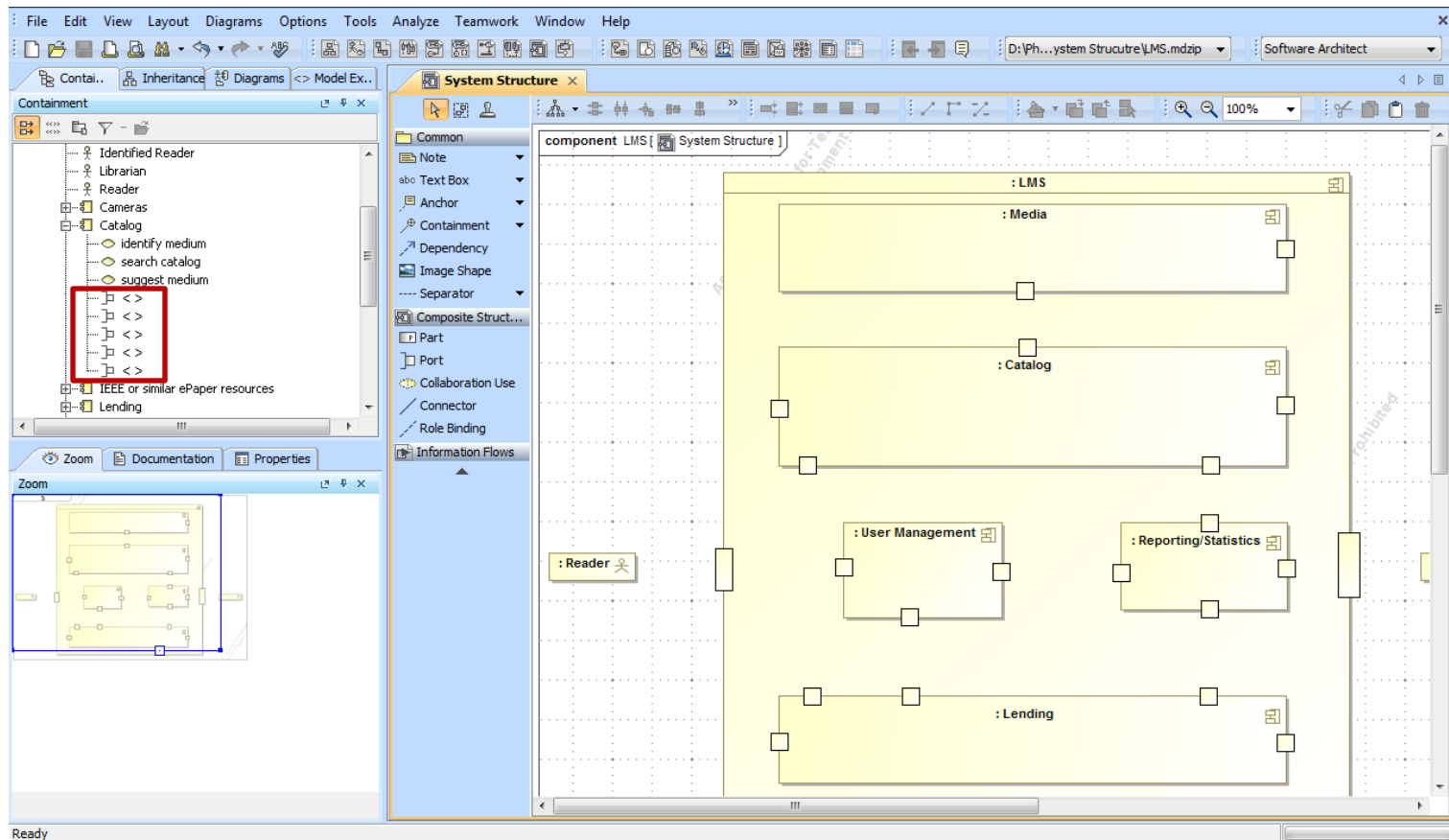
A dialog prompting for a port type will pop up. You may select <UNSPECIFIED> for the purpose of this diagram.



3. System Structure Diagram

Step 4: Add ports

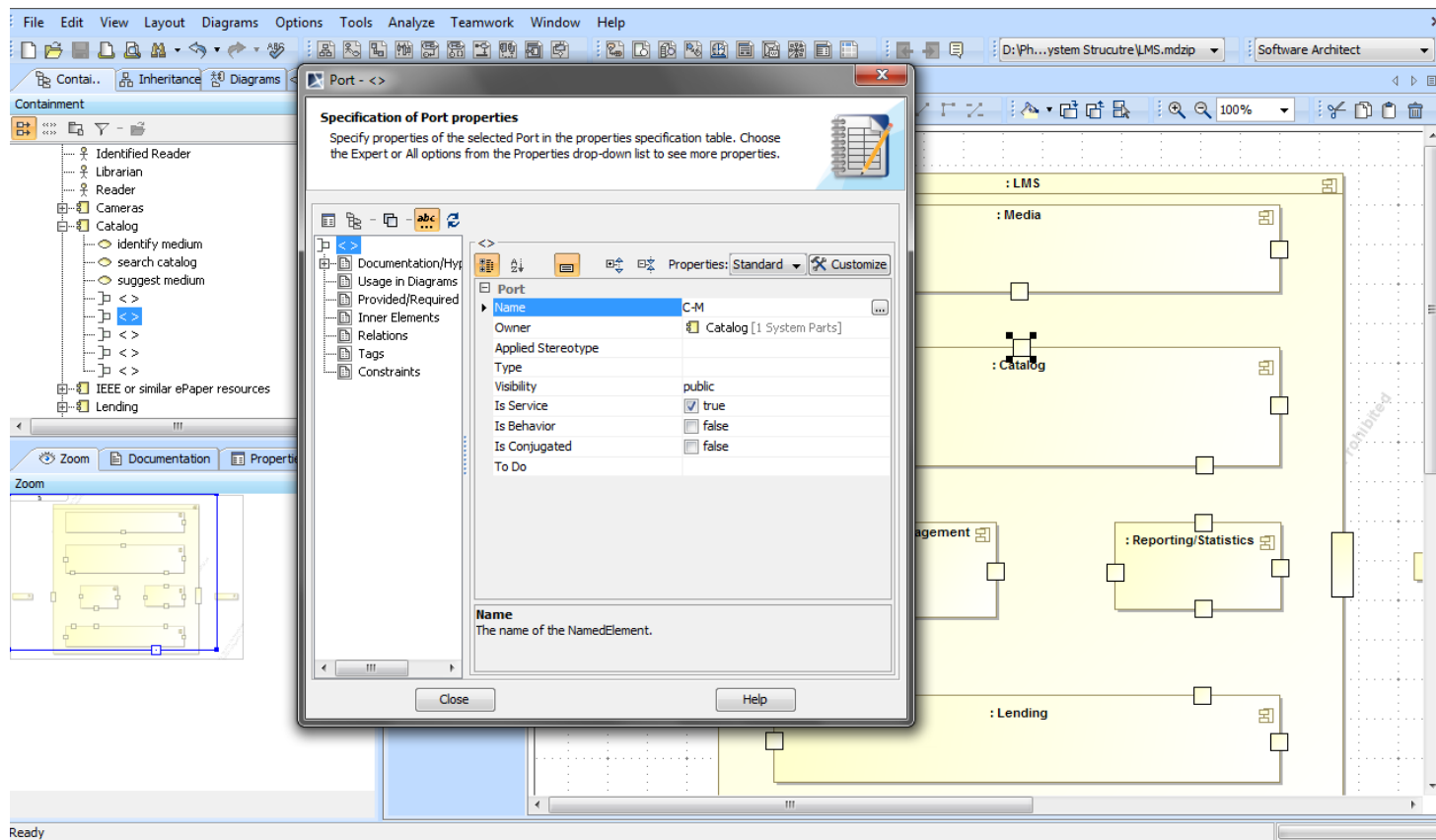
The created ports are visible on the diagram. The ports of the Media sub-system are highlighted in red in the containment tree.



3. System Structure Diagram

Step 5: Name ports

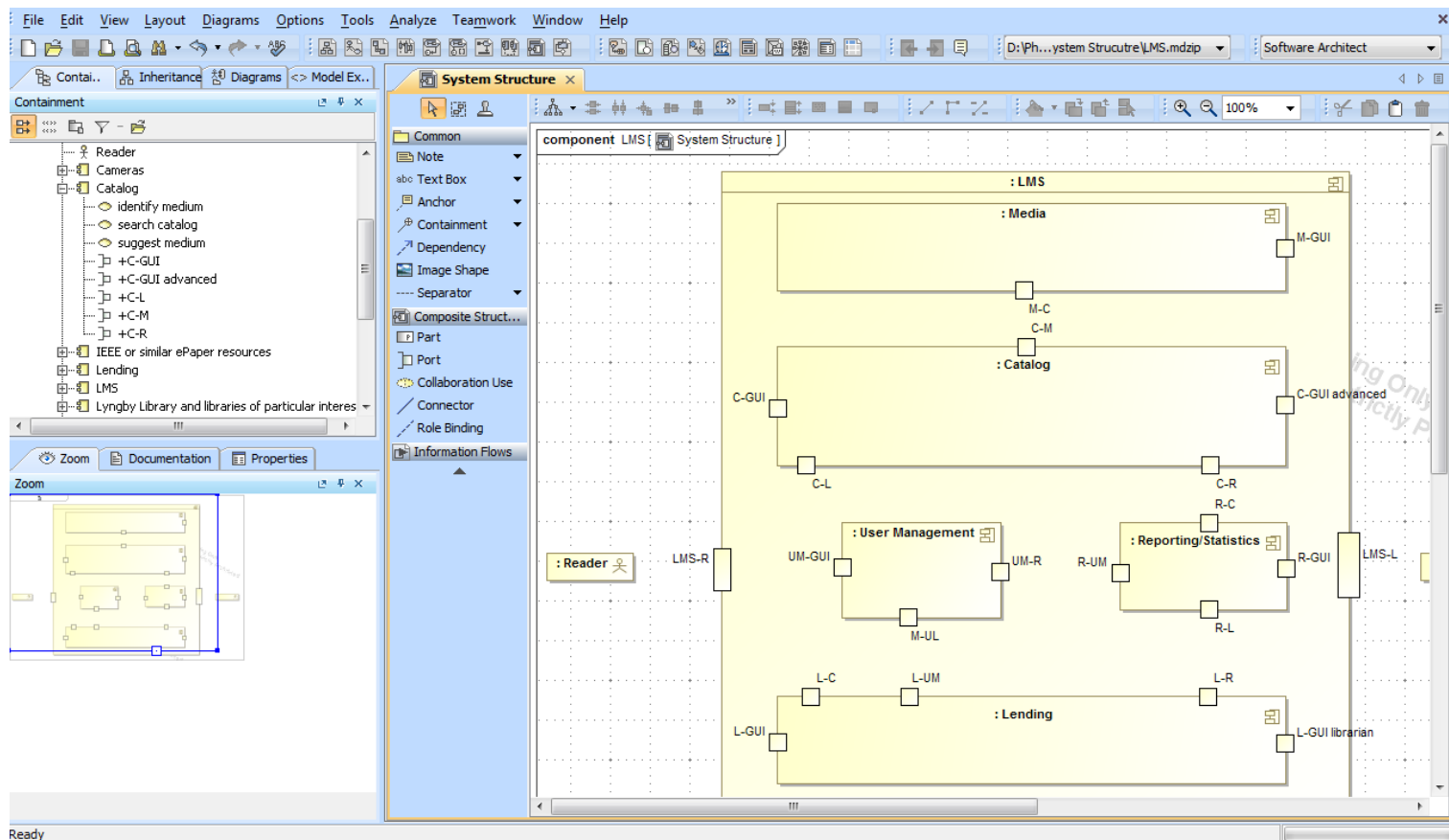
Names are set for ports using the Specification entry in each port's context menu.



3. System Structure Diagram

Step 5: Name ports

The example diagram with all ports named is displayed below.



3. System Structure Diagram

Step 6: Add connectors

Finally, **Connectors** are added between ports using the toolbar button highlighted in red below.

