**1.what does it mean for two views to be consistent? What kinds of inconsistencies can arise in multiview models?**

**2.What is an architectural model? What is its relationship to architecture? To design decision?**

An architectural model is a type of a [scale model](https://en.wikipedia.org/wiki/Scale_model) - a physical representation of a structure - built to study aspects of an [architectural design](https://en.wikipedia.org/wiki/Architecture) or to communicate design ideas.

Depending upon the purpose, models can be made from a variety of materials, including blocks, paper, and wood, and at a variety of scales.

Architecture is the art of designing buildings. Architectural models are used by architects for a range of purposes. [Ad hoc](https://en.wikipedia.org/wiki/Ad_hoc) models are sometimes made to study the interaction of volumes, or to get an idea of how they look from different angles and to explore ideas. They can be used to exhibit and sell a design to help visualise a design. A model may be useful in explaining a complicated or unusual design to the building team, or as a focus for discussion between the design teams such as architects, engineers and town planners. Models are also used as show pieces, for instance as a feature in the reception of a building, or as part of a museum exhibition such as scale replicas of historical buildings.

Some types of model include -

Exterior models are models of buildings which usually include some landscaping or civic spaces around the building.

Interior models are models showing interior space planning, finishes, colors, furniture and beautification.

Landscaping design models are models of landscape design and development representing features such as walkways, small bridges, pergolas, vegetation patterns and beautification. Landscaping design models usually represent public spaces and may, in some cases, include buildings as well.

Urban models are models typically built at a much smaller scale (starting from 1:500 and less, 1:700, 1:1000, 1:1200, 1:2000, 1:20 000), representing several city blocks, even a whole town or village, large resort, campus, industrial facility, military base and so on. Urban models are a vital tool for town/city planning and development. Urban models of large urban areas are displayed at museums such as the [Shanghai Urban Planning Exhibition Center](https://en.wikipedia.org/wiki/Shanghai_Urban_Planning_Exhibition_Center), [Queens Museum](https://en.wikipedia.org/wiki/Queens_Museum) in New York, the [Beijing Planning Exhibition Hall](https://en.wikipedia.org/wiki/Beijing_Planning_Exhibition_Hall), and the [Singapore City Gallery](https://en.wikipedia.org/wiki/Singapore_City_Gallery).

Engineering and construction models show isolated building/structure elements and components and their interaction.

A software architecture can be considered as the collection of key decisions concerning the design of the software of a system. Knowledge about this design, i.e. architectural knowledge, is key for understanding a software architecture and thus the software itself. Architectural knowledge is mostly tacit; it only exists in the heads of the creators. A problem is that this type of knowledge is easily lost. This phenomenon is called architectural knowledge vaporization and contributes to a number of problems that the industry is struggling with: expensive system evolution, difficult stakeholder communication, and limited reusability. The central theme of this thesis is how to reduce this vaporization of architectural knowledge. The focus is on one important form of architectural knowledge: architectural design decisions. This form is important, as the architecting process is all about making these key decisions. To reduce vaporization, this thesis explores a codification solution, in which these decisions are documented and modeled. For codification, the concepts one wants to codify must be understood well. To this end, a conceptual model of architectural design decisions is presented that explains the parts that this concept is made of and how it relates to other concepts. Based on this model, two approaches have been developed. The first, the Archium approach, is used for codifying, managing, and maintaining architectural design decisions in a forward engineering setting. The second approach is the Architectural Design Decision Recovery Approach (ADDRA) for recovering architectural design decisions in a reverse engineering setting. The Archium approach is evaluated in two ways. First, a study of how Archium deals with common use-cases for managing architectural knowledge is presented. Second, to address the issue of expensive system evolution, the Archium tool is compared with other tools using an evaluation framework. To evaluate ADDRA, the approach is applied on a case study. Both ADDRA and Archium help the architect with codifying architectural design decisions. This codification has two important consequences for software architecting. First, design decisions become explicit bridges between design and rationale. Second, once codified, a software architecture can be seen as set of architectural (design) decisions. Both consequences put a software architecture into a new perspective and deepen our understanding about what software architecting is all about and they also help to reduce architectural knowledge vaporization. As future work, we plan to investigate other types of architectural knowledge and their relationship to architectural design decisions.