Alexander Lundqvist: DM For the domain model, the classes are representativ of reality rather than being a model of a system apart from the AccountingSystem and InventorySystem, which have no part . On the other hand, many of the associations give no additional information, for example printer isPartOf Registry. This gives no information of the purpose of the printer, only the location. It also has a few major links in the model, specifically cashier and registry, which together acts a spider-in-the-web.

Many of the other classes have very few associations and are far from the center which makes them very peripheral. Some of the associations have confusing names, ”isBoughtIn”, ”isStoredIn”, etc. The model itself is generally readable and the names follow convention. If provided as a model to interpret the scenario, one would understand the interaction, but would not be given any additional clarification compared to the requirement specification.

There are a couple of classes that are irrelevant or made irrelevant due to lacking associations. Printer and Change have associations that do not give relevant intel about their purpose or incorrect information about their associations. Adress is currently floating freely, which adds no new information to the model. Had it been an attribute instead, it would have provided relevant information. Item identifier is one of the classes that lack attributes, which is unfortunate as it is a class that would benefit majorly from them.

Currently the class does not add readability to the model, and the associations confuse more than they clearify. SSD The guard on the if-statements are used incorrectly. They give no information on what scenarios are considered, only state what is already specified in the ”alt” box.

The diagram lacks two lifelines, accounting system and inventory system. Since the newSale is within the alt, the system will constantly print the receipt. The return values are not used correctly, and all parts of the alternative flow is not considered. The included operations are not correct either.

It is difficult to understand the flow of the SSD and the fundamental structure of the diagram is incorrect. UML language is not followed, but rather includes a pseudo code approach without reason.