Chapter 12
Product Lifecycle Management Revisited

The scope of this chapter is PLM (Product Lifecycle Management). I point out some concerns with the traditional way of understanding PLM, and discuss potential benefits from tackling these concerns from the Activity Domain Theory (ADT) perspective.

PLM is a wide-ranging information system (IS) that contains product data and product related data. Sometimes PLM is referred to as the product’s “digital backbone” that defines all aspects of the product as it progresses through various phases of its lifecycle (see Figure 1).

PLM evolved from the need to manage large and complex data structures related to CAD (computer-aided design), CAM (computer-aided manufacturing), and CAE (computer-aided engineering) systems, usually referred to as CAx “authoring tools”. Before 1999, CAx-systems were managed with Product Data Management (PDM) systems. Around 1999, vendors of PDM systems started to include capabilities for document management, configuration/Bill-of-material (BOM) management, revision control, and workflow management. The term PLM was quickly adopted by vendors, who wanted to differentiate PLM from the prevalent CAx data centric focus of PDM (Lindenthal, 2008).

Today, most PLM-systems offer the following capabilities:

DOI: 10.4018/978-1-60566-192-6.ch012
Product structure management: structuring the product in various phases of its lifecycle: as-required, as-designed, as-planned, as-simulated, as-built, as-manufactured, as-delivered, and as-maintained.

Configuration control: configuring a specific product from general product structures containing many possible options. This is often done by specifying a set of parameters and rules that pinpoints a precise configuration of the product.

Revision control: managing various revisions of the same item.

Variant control: managing variants of the same product, for example, a car built to customer specific features.

Document management: managing various types of documents with revision control, definition of attributes, relations to other items, etc.

Classification: structuring a particular type of item into various classes. For example, a car manufacturer might want to structure “cars” into the classes “sports cars”, “load carrying cars”, or “low emission cars”.

CAX interfaces: tight interfaces to authoring tools.

Viewing: visualization of the product as 2-dimensional drawings or 3-dimensional solid geometries.

Work flows: defining work-flows for repetitive tasks such as change management, release management, and registration of products.

Authorization control: the allocation of rights to various roles concerning operations on the data items such as reading, writing, and deleting information.