Dewayne E. Perry

Systems and Software Research Laboratory
Bell Laboratories
600 Mountain Ave
Murray Hill NJ 07974
dep@research.bell-labs.com +1.908.582.2529

Position Statement

1. General comment on the Product Line Reference Architecture approach

- too solution specific, not enough emphasis on the domain itself

you may well have the components of the system organized as your current reference model suggests in your component repository, but the reference architecture itself should be organized around the domain (ie, customer focused) that the product line addresses.

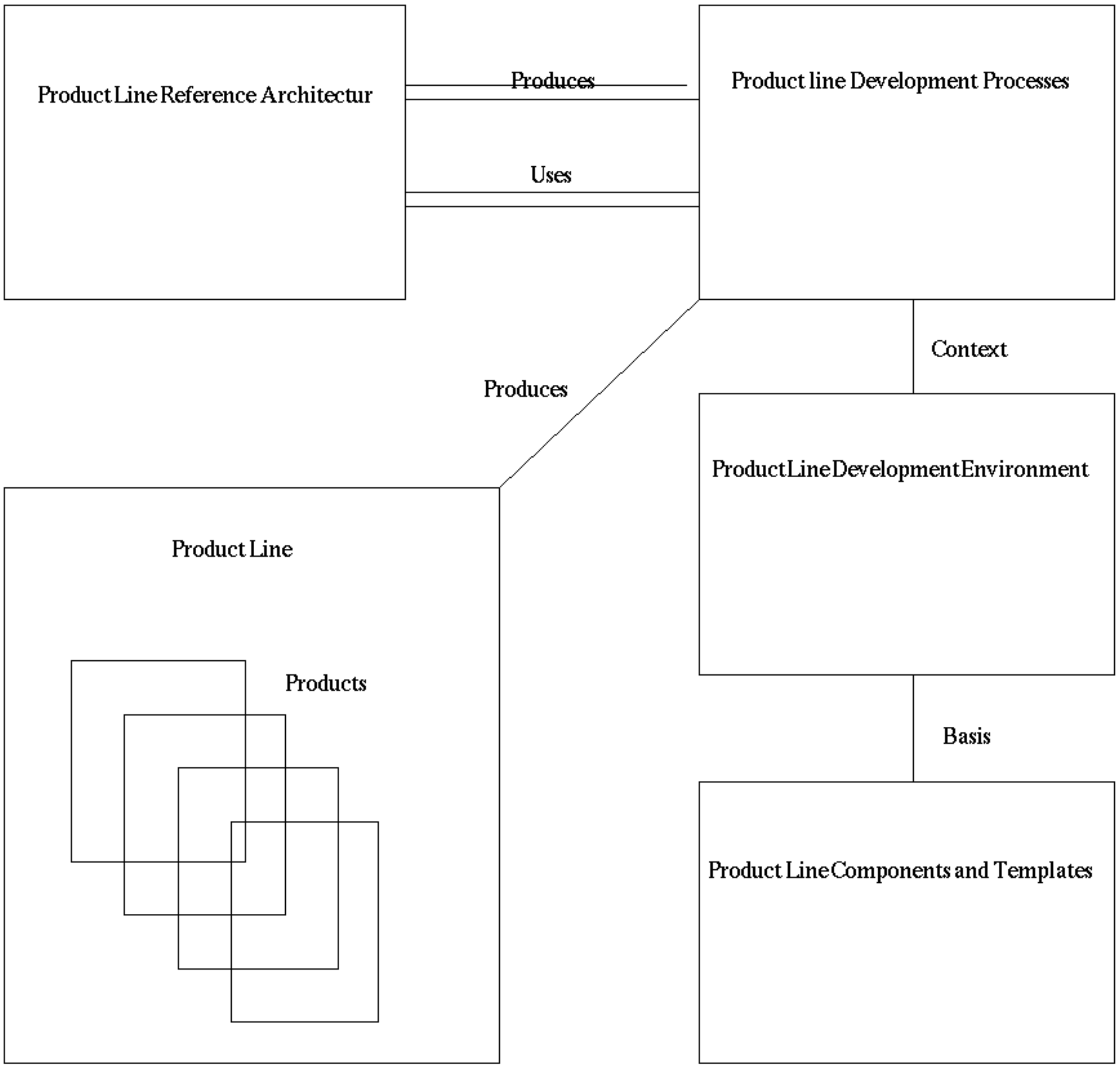
consider the typical process centered environment architecture description: gui interface component, tool set component, object store, process engine, etc. that certainly describes the components in the solution space, but tells almost nothing about the actual domain of process support. for that you need components such as: process definition components, process instantiation, process administration, process execution, process state mgmt, process history mgmt, process guidance, etc. these latter components deal with the domain that of the system, the former are there and are used to implement the domain specific architecture.

- not quite enough abstraction and emphasis on commonality.

consider the example of messaging in the middleware application services: you have four domain specific components: voice, fax, text and multimedia, each with their own subarchitecture which are almost identical. you should collapse these into one uniform subarchitecture. the advantage of this is that it shows the domain shape of the subarchitecture and where things are identical and where they are outwardly similar but inwardly different.

this points out in important distinction: those components which are shared and those which have the same shape (ie, interface, or general behavior) but which must be implemented distinctly for the different subdomain.

2. Product Line Business Processes to build the product line reference architectures and use them to build product lines and products


Product Line Reference Architectur
Produces
Product line Development Processes
Uses
Context
Produces
Product Line Development Environment
Product Line
Products
Basis
Product Line Components and Templates

the general approach is to have a set of product line processes first to define the product line reference architecture and the individual product reference architectures and then use them with an appropriate product line development environment together with the shared components and component templates (for those components which must be either individually built or customized for the particular sub-domains.

3. Reference Architecture VS Implementation Architecture

there are several ways in which this might be approached. the way i like the best is not to think of an implementation architecture at all, but think of either refining the architecture in to a number of levels (which is could be considered a slight of hand trick i suppose - in some sense it is providing a more detailed view of the high level architecture, its implementation if you will) or leaving the implementation to the designers (which is of course their job)

part of it is a question of how much do you want to put into the architecture and how much you want to put into the design and implementation. i think you should keep the architecture description in the business domain - the domain specific architecture that it is. but i also think that having a layered architecture is also a good thing - but that i think should still be in the domain specific variety.

what you might do is to suggest existing components that satisfy the various domain specific architectural components. i would view this not as architectural but as implementation guidance. your architecture is build from either existing components, instantiating well-defined domain elements, or building new pieces.

how you go about doing that is part of the overall implementation process for the product line (or, better put, the domain specific) architecture. i think it is here that you make the correspondence between the architecture and the pieces that are used to implement it. it is in the design and implementation processes where this mapping should occur, not in the architecture. this is why i shy away from talking about an `implementation architecture'.