

What is Problem Domain and Solution Domain

Date: September 22, 2015 Author: Shey.shah00 0 Comments

Problem Domain:

Problem domain (or **problem space**) is an engineering term referring to all information that defines the problem and constrains the solution (the constraints being part of the problem). It includes the goals that the problem owner wishes to achieve, the context within which the problem exists, and all rules that define essential functions or other aspects of any solution product. It represents the environment in which a solution will have to operate, as well as the problem itself.

It is called scope of analysis, When collecting user stories or User Requirements whatever, how do you decide which ones are relevant? You keep in mind some higher-level statement of what the objectives are and what people think the problems are (if people didn't think there were problems, they wouldn't be paying someone to come up with a solution, would they?). Then it's either in, out or borderline. If it's borderline, you probably want someone to agree it's out (even if you want it to be in). In the mean time, and probably afterwards, you keep it as Something to think about.

Note that the customer for a software solution (the "problem owner") doesn't necessarily recognise the existence of a problem so much as an opportunity. An engineer sees a "problem domain" as being the set of circumstances for which s/he has to provide a solution; it's the engineer's problem, not necessarily the customer's.

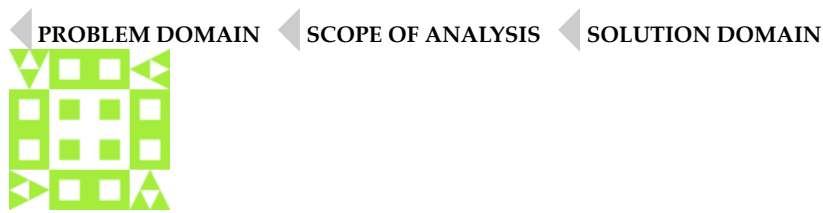
Solution Domain:

While the Problem Domain defines the environment where the solution will come to work, the solution domain defines the abstract environment where the solution is developed. The differences between those two domains are the cause for possible errors when the solution is planted into the problem domain.

In respect to a given problem (or set of problems), the **solution domain** (or **solution space**) covers all aspects of the solution product, including:

- The process by which the solution is arrived at;
- The environment in which it is constructed;
- The design, construction, testing, operation, and functions of the solution product itself.

Confusing the *problem* with the *solution* is one of the great dangers of IT projects, resulting in software that may be a very good solution to *some* problem, but not to the specific set of problems its users face. See the quotation from Gamma *et al.* under Problem domain



Published by Shey.shah00

A Software Engineer.Web Develper,Web Designer,Software Manager.Process Handler,programmer,Technology Lover.Advisar and Thinker of Tech and nature. View all posts by Shey.shah00

© 2022 SHAHWORLD

WEBSITE POWERED BY WORDPRESS.COM.