Title: Design Guidelines for Domain Specific Languages.

Link: http://www.dsmforum.org/events/dsm09/Papers/Karsai.pdf

What are the problems/research questions addressed by this article?

--->
Now-a-days, the need for new languages for various growing domains is strongly increasing. It is so complex for a developer to design a new domain specific language and its too time consuming. Existing tool support for designing a new language just focuses on simplification of technical aspects but lacks in principles of a good language design. The author talked about the guidelines to be followed for better designing of a domain specific language by relying on their own experience and as well as on existing general purpose languages.

What are the existing solutions for this research question/problem?

--->
As the need for new languages for various growing domains is strongly increasing. Fortunately, also more sophisticated tools exist that allow software engineers to define a new language with a reasonable effort. we developed the frameworks MontiCore and GME which support the definition of domain specific languages. Developers use these frameworks for designing several DSLs for a variety of domains, e.g., a textualversion of UML/P notations and a language based on function nets in the automotive domain.

What is the research method [s] they have used?

--->
The authors have compiled this list from literature and from their own experience. Guidelines might change during time as developers gather more experience, tools become more elaborate, and taste changes. Maybe some guidelines are not relevant anymore in a few years, as some guidelines from the 1970’s are less important today.

What is their proposed solution ?

-->
In this paper 26 guidelines have been discussed that should be considered while developing domain specific languages.
"Identify language uses early", "Ask questions.”, “Make your language consistent.”, “Decide carefully whether to use graphical or
textual realization.”, “Compose existing languages where possible.”, “Reuse existing language definitions.”, "Reuse existing type systems.” are some of the proposed guidelines.

What are three future directions from this article?

--->
Other guidelines are needed for successfully integrating DSLs in a software development process, deploying it to new users, and evolving the syntax and existing models in a coherent way.
Some of the guidelines have to be discussed in certain domains, because they might not have the same relevance and as discussed many guidelines contradict each other and the language developer has to balance them appropriately.

Concepts that you learnt from this paper?

--->
reuse of existing artifacts versus the implementation of a language from scratch, organizational structures for models.