



Yvonne Dittrich's classification after ©Jeanette Eriksson (work in	Four categories of end user adaptability
	configuration through data
	putting together predefined objects (e.g. plug-ins)
	problem domain specific tailoring interface with code generation
	coding add-ons

access to code would not count as end-user adaptation.

Product	kind of flexibility	how are they realised	what about upgrades	limitations
Fleet Planner	configuration of customer specific fields import/export of data	meta data macro language	control over meta data backward compatibility	fixed data model fixed domain
TIA	configuration of customer specific fields business rules in the data base 'user exits' (new logic and replace logic) customisation of code base	metadata tables metadata and new code new code and code replacement coding	translation of metadata manual upgrade of user side code	everything is possible
Web Identity Management	plug-in interfaces to third party	supporting a number of standards	problem, keep it simple goal: not to bother the user	new features?
Hearing Aid Customisation	the flexibility is with the hearing aid	?	with a new hearing aid a new software is distributed	
Microsoft Dynamics NAV og AX	configuration (meta data) and coding	total solution split into kernel and application. application is open and shipped. integrated development environment offer full customization of multi-layered app in meta data and code. no code generation	tool supported upgrade with manual conflict handling	few limitations. challenge of flexibility vs. upgradeability
Scott Hill's historic overview	Factor	Size	Method	Distribution Model
	factors: size of product	>10000	no structured methods	fixed
		between 10000 and 100000	structured methods	configuration or shipping code
		bigger and until several million lines of code	its just difficult	the customer does not see the code