

# Simplicity & Customisation: The Future of EA?

**Ian Mitchell, Designer of eaDocX, and BPMN trainer and user. EA user since v3.5**

Getting new groups of users to use EA can be a struggle. Let's face it, it's a complicated tool. It's got a big, complicated user interface, which seems to get more complicated with every release.

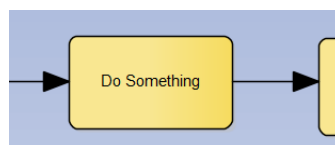
Those of us who have used it for many years, and have seen it evolve, seem to be able to visually 'filter out' those bits of the UI which we either don't use, or don't understand. So we perhaps forget that the first time a new user sees EA it can be a bit of a shock.

And the problem is getting worse, mostly because of the standards that Sparx have to implement which are introducing extra complexity. Take BPMN as an example.

BPMN is the OMG's 'official' UML-based process modelling language. It's a big improvement on the old-style UML flow chart (Activity Diagram), and this is where the problems begin. Old-style flow charts had the benefit of being a bit fuzzy at the edges – the basic flow is easy to understand, but dig a bit deeper, and the precise meaning was hard to nail down. So, BPMN came along, and tried to remove the fuzziness. Which it does. But it does this at the expense of increased complexity.

## A Simple Activity

Consider the simplest bit of BPMN: the humble Activity. The bit of the process description where someone does something:



What could be simpler? It's just a simple description of how someone 'does something'. And in the old-style flowchart, that's all there was to it.

But BPMN wants to add precision. So it lets us specify a few additional things about the Activity.

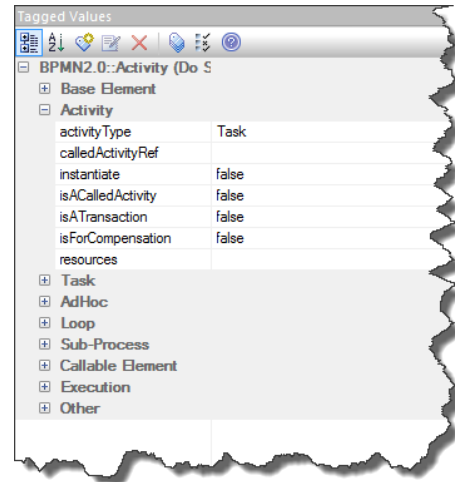
When I say 'a few', that's not quite true. Try 'a lot'. There are 51 additional values, in 9 groups which BPMN describes, and which Sparx correctly implements!

## Choosing Additional Values

Now those of you who are experienced BPMN users will be thinking ‘hold on – nobody uses all of them’, and you’d be right. In fact, even though I teach BPMN, there are fewer than 20 that I understand, and only 2 or 3 which I ever use.

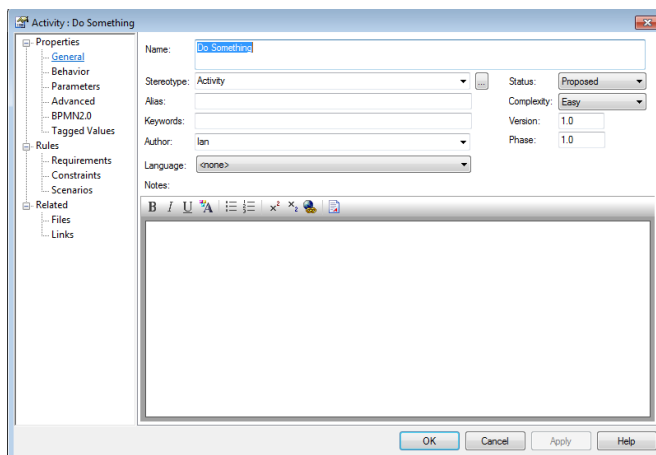
But EA is a fully UML and BPMN compliant tool. Sparx can’t be expected to know which of us will use which subset of these 51 attributes. Each of us probably has a slightly different sub-set. Sparx has to make ALL of them available.

Novice users can be confused by the complexity of ‘BPMN-as-implemented-in-EA’. It makes life hard for us experienced users as well, as we have to remember which 2-3 attributes to use, and where to find them.



## A Simple UI?

And it’s not just the additional attributes which BPMN requires which confuses the UI. There’s the EA implementation of this, which means all EA Elements share the same properties window:



We’re all used to this, but when a novice user opens up the details of their BPMN Activity, there’s nothing ‘BPMN Activity’ about it. It’s just the normal EA Properties window, with just a hint of BPMN-ness with the ‘BPMN2.0’ Properties tag.

And what about the ‘Constraints’, ‘Scenarios’, and ‘Requirements’ tags? Maybe our new user should capture their requirements there, or write some scenarios.

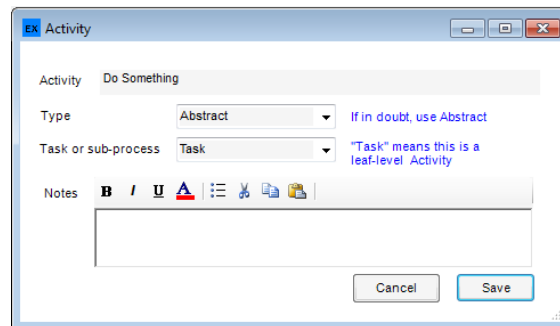
Scenarios? For a BPMN Activity? What? Maybe it’s defined in the BPMN Specification, but it makes no sense in any normal usage of BPMN. The user just has to know to filter-out these bits, and go straight to the ‘BPMN2.0’ tag, and the simple list of 51 additional attributes!

What would be great would be a way to hide the complexities of ‘BPMN-as-implemented-in-EA’ from users, and show them just what they NEED to use, rather than what the BPMN standard and the EA implementation allow.

## An Answer

So, having struggled for several years with BPMN/EA, I came across a new tool called eaForms.

This allowed me to create simplified properties windows for EA elements. So, if I don't like how the EA Properties window for a BPMN Activity looks, then I can create my own:



Much less frightening! I can pick which standard EA attributes I want to see, which BPMN tagged values, and even which tagged-values of my own. I can add my own explanations, and call attributes by new names which my users will find easier to understand.

I can even let users chose and create links to other elements, all from the same form, without needing to use a diagram. So, I might want to link my BPMN Activity to, say, a Business Rule, which the Activity implements.

This is a much less frightening window than the EA default because I have applied my own view about what my users need to specify for a BPMN Activity. This may not be the same as yours. I may also change my view of what I'd like them to specify, as they get happier with EA and BPMN. In which case, I can just create a more elaborate form. Remember, the information about the activity is still stored in my EA model, with all the BPMN attributes available if I ever want them.

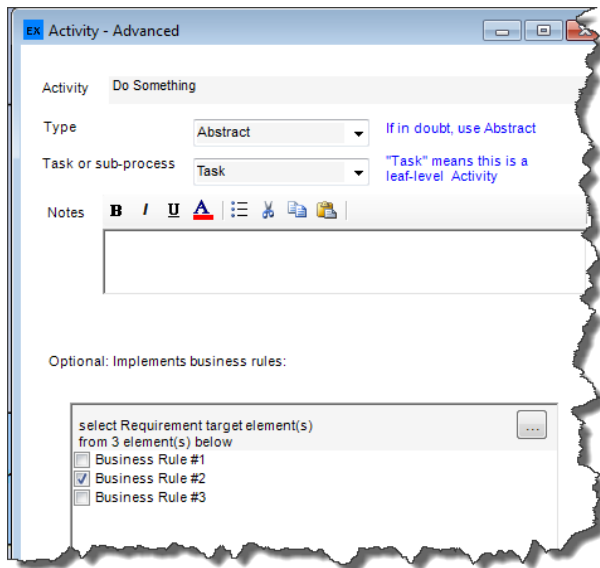
### **Implications of a Customizable EA User Interface**

This isn't just an exercise in 'dumbing-down' the EA user interface. I think eaForms has wider implications.

It makes EA more appealing to more users, getting them over the initial shock of the 'vanilla' UI, and maybe makes the transition from Visio quite a bit simpler.

But even for the experienced EA and BPMN user, it still has great value. Where a model has many users, keeping all those users modelling in the same style is a constant challenge. It takes time away from more productive tasks for key people. And being a "modelling-style policeman" is a really dull job.

Now I can embed our local modelling standards in a series of bespoke forms, give those forms to all the users, and they don't have to do anything to get them: just click on an element, and up comes the form. And those customisations aren't restricted to which attributes the user sees. For example:



This is a new version of my Activity form. Here, I decided that the link between an activity and a <<Business Rule>>Requirement is useful information. So, I can let the user choose which business rules get implemented by this Activity.

The user doesn't need to know that business rules are in fact <<Business Rule>> stereotypes of Requirements, or that the Activity and the rule are linked with a UML 'Realization' connector. That's all done by the form.

## The Future of EA Customisation

EA already has a great array of customisation features. We can create new stereotypes, give them different appearances with graphics or shape scripts, create new diagram and toolbox types, and wrap all these changes up into our own MDGs. We can even add to the EA help, with our own content.

And now we can add our own user interface as well.

The way now seems to be open for the EA user community to create our own EA versions, which have highly bespoke implementations of different model types, using all the customisations which are now available. So we might, for example, create an 'EA for BPMN', which has a practical sub-set of the BPMN standard, implemented as simpler set of diagrams and elements, and with links to non-BPMN elements like business rules, stakeholders or TOGAF/Archimate ideas. And all delivered via a user interface which understands what's important, and hides what's not.

This should make EA appeal to a whole new set of users, who are currently seeing only a generic tool with generic solutions, but who can now get a tool which matches their needs in great detail.

Seems to me that we're entering a new era of EA customisation. So what are you waiting for?

For more information, see [eaForms on eaWorkplace](#).