

Colour Your Knowledge

Ways to use colour in your EA models and documents

Ian Mitchell, ian@eadocx.com

When my children were small, each evening I used to ask them what they had done at school. Their answer? “Stuff”. One day, they asked me what I’d done at work. As I was a Business Analyst working on the high-level specification of a fixed-mobile convergence switching system, using UML and EA, I thought I need to simplify my reply a bit.

“I do colouring”

I showed them what I’d produced that day: a big Use Case diagram, with lots of stick-people, coloured ovals and arrows. They weren’t impressed. They haven’t asked me since.

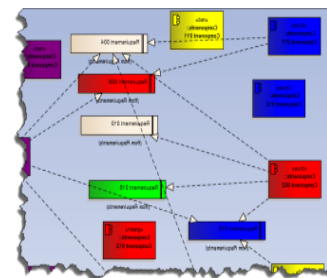
In my job as a mentor to EA projects, I look at dozens of models and hundreds of diagrams each year, and one thing is very noticeable. EA does a great job in making modellers stick to UML, BPMN or SysML notations, but the use of colour varies hugely. Sometimes I see diagrams where the meaning is clear at a glance. Other might have been scribbled by a 3 year old, excited by their first crayons.

So I thought it was time to do a quick review of *why* we might want to use colour in our models, and also *HOW* we can implement sensible colouring in EA.

Colour – why we need it.

Looking through my collection of EA models from around the world¹ they seem to use colour in just a few ways:

- **Randomly** – just to make things look ‘interesting’. I don’t see this a lot, but it’s a common rookie mistake. Someone finds the F4 option on the diagram, changes a few background colours, and hey! – doesn’t that look good!
I’m sure *you* wouldn’t do this, so I’ll say no more about it.
- **To show STRUCTURE.** This is probably the most common use of colour, and the most useful. Colour is used to indicate the nature of the thing. ‘Things’ in the diagram which share the same set of properties share the same colour. The UML stereotype is the normal way to group these similar ‘things’, though boundaries and swimlanes are also useful.



¹ perhaps the beginnings of a small museum?

- **To show STATUS.** So ‘open’ issues are in one colour, closed ones in another. Strictly speaking, this is a sub-set of the Structure case above, but whereas the **stereotype** of a ‘thing’ probably won’t change, we would expect that its **status** probably will.
- **To draw the attention of specific users.** This is the author of the diagram talking directly to the reader: “Look – this bit is important”. Hard to do in EA, as there may be all kinds of readers, both now and also in the future. So this use of colour means changing the output of the model, rather than the model itself.

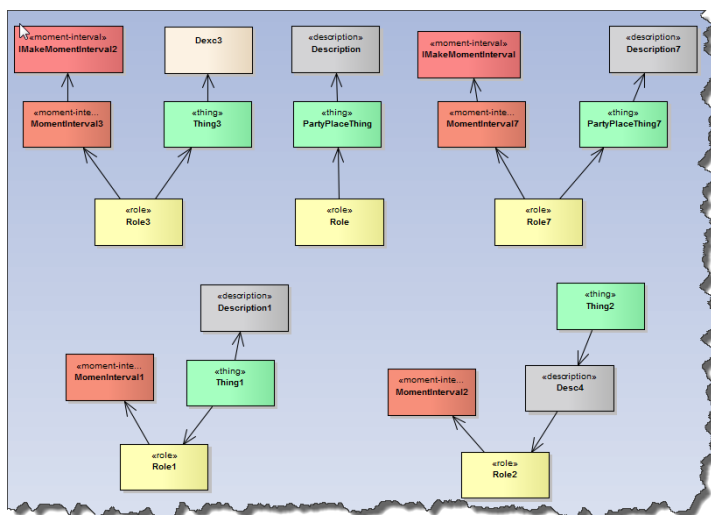
In this paper, I’ll look at how to make good use of colour in your EA diagrams and documents, which will help you to communicate the knowledge you have in EA more effectively.

In each section, I’ll look at techniques for using colour, then how to apply those techniques using EA.

1 - Colour as Structure

In *Java Modelling in Color with UML* [1], Coad, Lefebvre and De Luca describe an approach to using a standard set of colours for what some common types of classes which they find in Domain (Class) models.

Now you might not be using these types of classes in your models, but the principles they use are more generally useful. For example in the diagram below, taken from [1]



Even knowing nothing about the example, you can see that the clusters of classes all follow a similar pattern. All except the bottom right hand one.

Even knowing nothing about what the colouring means, that one is clearly different from the others, which will be clear to any reader.

What the authors of this modelling approach have done is

- defined a set of ‘types’ of class, which are useful in their modelling world
- chosen colours for each one of those types. Note they use pale colours, not primary ones
- ..then they use those colours whenever they show those classes

This then means that the reader of any of their diagrams can get an instant impression of the structure of a diagram with a single glance.

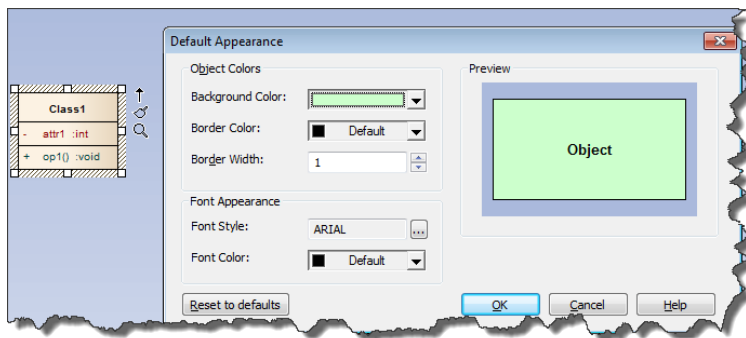
Using colour to show structure in this way isn't just for fun. It can help you to spot errors, and help your readers to see the overall shape of your diagrams before they dive into the detail.

Adding Colours to Elements

As is so often the case in EA, there are lots of ways to apply colour to Elements². Some good, some bad.

Let's start with the bad.

You have a diagram – a class diagram – and you'd like to help your readers by applying some colouring. The obvious solution is to select the item, hit F4, and change its background colour.



Easy.

..but you have just set-off down a long, dark road.

Your class will now change colour. And even if you put the class into another diagram, EA will keep the same colouring.

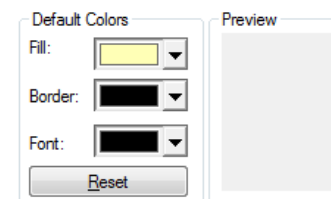
BUT, you're going to have to do

this every time you create a new class, AND remember what your colour scheme was.

A Good Approach

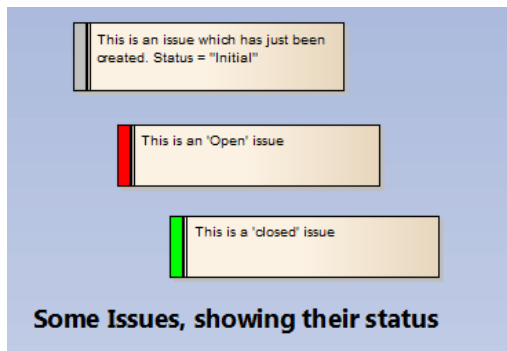
On your diagram, select a class with the stereotype you want to colour. Then select **Settings | UML Types**, and choose a colour for EVERY class with this stereotype.

So now, every new class with this stereotype will get the right colour. Not revolutionary on its own, but you'll soon start to see patterns in the structure of your models, and it will help make sure that you use your stereotypes consistently.



² An EA Element is any EA 'thing' which can appear in the Project Browser. So Requirement, Class, Use Case and Interface are all examples of EA Elements. EA 'things' which are not Elements are e.g. Glossary entries, Project Issues or Internal Requirements or Constraints.

2 - Colour as Status



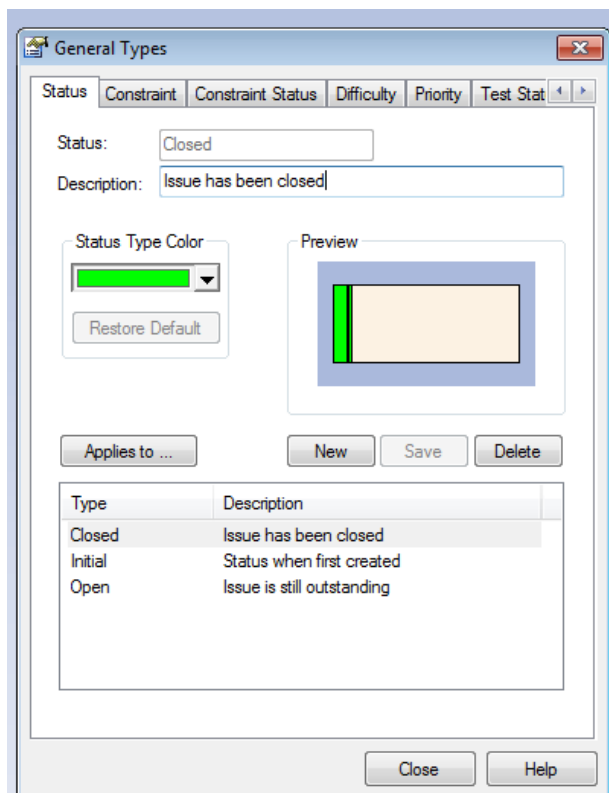
This is similar to the 'Colour as Structure' approach above, but is more dynamic. It would be odd to have a class with one stereotype suddenly become another, but if we base the colour of something based on its State (or status attribute in EA terms) then we *expect* it to change colour.

In this example, we have used a feature of some kinds of EA element to have their colour change based on their status.

Adding Colours to States

You can set up your own colours for your own set of states.

In EA, select **Settings | Project Types | General Types**:



Here, we have created three different status values, and given them colours.

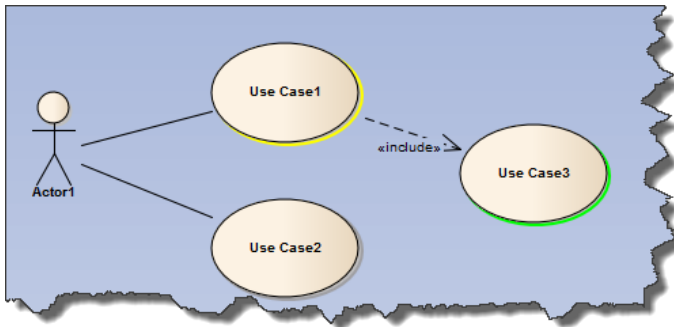
We have also used 'Applies to' to restrict which kinds of EA elements will be given these status values & colours.

It's worth spending some time, even in the early days of your modelling, to choose some status values and colour which are meaningful in your world. Just the act of thinking about it will start to add structure and clarity.

The EA Elements which look good when using this approach are those which have built-in shapes with a coloured bar at the left-hand side: Requirements, Changes, Issues, Risks.

But it can also apply to other elements as well.

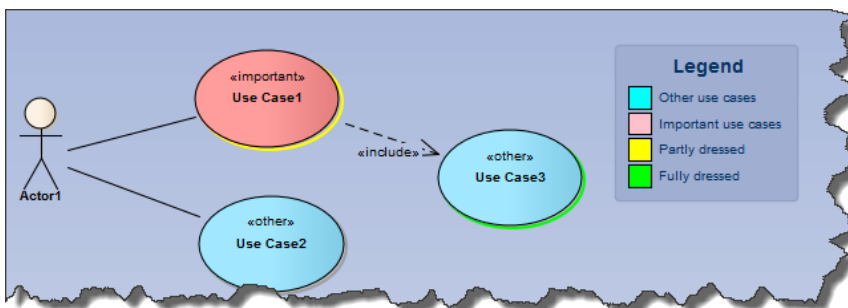
EA doesn't *fill* other kinds of elements with the status colour, just gives them a coloured *shadow*.



We have chosen some stereotypes for our Use cases, to show the reader which ones are ‘initial’, ‘partly-dressed’ or ‘fully-dressed’.

Not as obvious as filling-in the use cases, but we can use both fill AND shadow.

The effect is rather hard to spot, but it the combination of structure (stereotype) based colouring and status-based is now starting to convey lots of additional information:



As we are now using LOTS of colours, we can’t expect our readers to remember them all, so we have used a Legend to explain them.

In case you haven’t found the ‘legend’ feature, it the icon on the main toolbar which looks like:



You can also copy & paste these legends from one diagram to another, so there’s no excuse for not including them. And, if you ‘paste as link’ the legend, then changes to one legend will update all the others.

3 - Drawing the Reader's Attention

So far, we have looked a couple of ways to change the diagrams in your EA model using colour. These changes will apply to *anyone* reading the model.

But what if we want to draw the attention of some specific readers to specific aspects of the model? We don't want to change the underlying model. We want to change the *view* which some readers see, by changing the documents which we give them.

Here, we will be using the eaDocX document generator, which gives us lots of ways to add colour to our documents, in ways which will help our readers. And we are moving beyond just colouring diagrams.

Conditional formatting

Using rules to change the colour of a cell may be familiar to Excel users. These are simple rules which change the appearance of a cell. For example, *'if the value is less than zero, colour the cell in blue'*.

When you generate documents from an EA model using eaDocX, you can also add Conditional Formatting rules. These can help your readers – and you! – to spot interesting information, or errors:

1 Component Summary

Ref	Component	Author
CMP07	Component7	Ian
CMP08	Component8	Ian
	Component9	Ian
CMP06	Component6	Ian
CMP10	Component10	Ian
CMP04	Component4	Ian
CMP05	Component5	Jackie

In this example, we have added two kinds of rules, which are applied each time we print a list of Components:

1 – Colour missing 'Ref' (Alias) values in pink, to show where there is missing data. This for quality control. This highlights an area where our model hasn't obeyed our own rule, and needs changing.

2 – Colour all 'Jackie's' components in green. Perhaps these are worth closer inspection?

For readers who are busy – basically all of them – this shows that you are respecting the time they spend reading your documents, by helping them to see what's important.

And they also help you to make your models consistent, by implementing your own quality rules in a way which is easy to spot.

Making Tables Interesting

As our documents get bigger, it becomes more important to break-up the document with colours which will guide the reader. A document which is just endlessly boring lists of things will soon cause most readers to stop paying attention. So we need to help them see what's what.

In the 'Structure' examples above, we used colour to show the 'type' of thing. Why not use the same approach with tables of data?

So this:

Name	Stereotype	Last_Updated
Role2	role	15/04/2013
Role3	role	15/04/2013
Role7	role	15/04/2013
Role1	role	15/04/2013
PartyPlaceThing	thing	15/04/2013
PartyPlaceThing7	thing	15/04/2013
Role	role	15/04/2013
Thing1	thing	15/04/2013
Thing2	thing	15/04/2013
Thing3	thing	15/04/2013
MomentInterval7	moment-interval	15/04/2013
Dexc3		15/04/2013
MIDetail	mi-detail	15/04/2013
MIDetail1	mi-detail	15/04/2013
Description7	description	15/04/2013
Desc4	description	15/04/2013
Description	description	15/04/2013
Description1	description	15/04/2013
MomentInterval	moment-interval	15/04/2013
MomentInterval2	moment-interval	15/04/2013
MomentInterval3	moment-interval	15/04/2013
Momeninterval1	moment-interval	15/04/2013
MIDetail2	mi-detail	15/04/2013

Becomes this:

Name	Stereotype	Last_Updated
Role2	role	15/04/2013
Role3	role	15/04/2013
Role7	role	15/04/2013
Role1	role	15/04/2013
Role	role	15/04/2013

Name	Stereotype	Last_Updated
PartyPlaceThing	thing	15/04/2013
PartyPlaceThing7	thing	15/04/2013
Thing1	thing	15/04/2013
Thing2	thing	15/04/2013
Thing3	thing	15/04/2013

Name	Stereotype	Last_Updated
MomentInterval7	moment-interval	15/04/2013
MomentInterval	moment-interval	15/04/2013
MomentInterval2	moment-interval	15/04/2013
MomentInterval3	moment-interval	15/04/2013
Momeninterval1	moment-interval	15/04/2013

In this case, each different stereotype has been printed using a different coloured Word table.

The same techniques can also be applied to other kinds of EA tables.

The Relationship Matrix allows us to show relationships between two sets of data in a simple 2-d table.

The basic EA Relationship Matrix just shows that there IS a relationship, and what direction it runs.

We can put more information into our EA model, by, for example, stereotyping relationships with RACI stereotypes, showing who is Responsible, Accountable, Consulted or Informed about each Requirement.

	Requirement 1	Requirement 10	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement 6	Requirement 7	Requirement 8	Requirement 9
Actor #1					↑					
Actor #2		↑					↑		↑	
Actor #3			↑	↑						↑
Actor #4	↑	↑						↑		
Actor #5						↑				

	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement 6	Requirement 7	Requirement 8	Requirement 9	Requirement 10
Actor #1				R						
Actor #2						I		C		A
Actor #3		A	C						C	
Actor #4	I						A			R
Actor #5				R						

When we apply some conditional formatting to the output, this can then make for a much richer and clearer explanation of the data.

Here we have replaced the 'Informed' stereotype with just 'I', and coloured each cell.

Readers can now see a glance who is responsible for what, and which Requirements have nobody responsible at all!

Summary

In this paper we have looked at some simple techniques for enhancing the readability and quality of your EA models using colour. We have looked at

- colouring based on Structure,
- based on Status, and
- colour to help particular readers see what is important.

Applying these techniques, using sensible, light-touch colours consistently, will help you to make the most of the work you put into your EA models, and transfer the knowledge you have put into EA to a wider audience.

If you have any other tips for using colour in EA, please send them to the author, and I may include them in future versions of this paper.

About the Author

Ian Mitchell has been an EA user since v3.5, over 10 years ago. He's a business analyst leader, mentor and trainer, and speaks regularly at industry conferences. He is also the designer of the popular EA document generator [eaDocX](#), and teaches Enterprise Architect for [Ability Engineering](#) in the UK.

See <http://www.eadocx.com>

References:

[1] Peter Coad, Eric Lefebvre, Jeff De Luca: **Java Modeling In Color With UML:** Enterprise Components and Process, Prentice Hall, 1999, ISBN 0-13-011510-X