



CHALMERS
UNIVERSITY OF TECHNOLOGY



UNIVERSITY OF GOTHENBURG

A Framework for Applying an API-layered Architecture to e³ value Models

Bachelor of Science Thesis in Software Engineering and Management

JARNO KYTÖ
MONICA MURGESCU



The Author grants to University of Gothenburg and Chalmers University of Technology the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet.

The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let University of Gothenburg and Chalmers University of Technology store the Work electronically and make it accessible on the Internet.

Applying an API-layered Architecture to e³value Models

JARNO KYTÖ
MONICA MURGESCU

© JARNO KYTÖ, June 2018.

© MONICA MURGESCU, June 2018.

Supervisor: JENNIFER HORKOFF
Examiner: JAN-PHILIPP STEGHÖFER

University of Gothenburg
Chalmers University of Technology
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

A Framework for Applying an API-layered Architecture to e³value Models

Jarno Kytö
University of Gothenburg,
Gothenburg, Sweden,
guskytja@student.gu.se

Monica Murgescu
University of Gothenburg,
Gothenburg, Sweden,
gusmurmo@student.gu.se

Abstract—In this design science study three different layering options were created to find out which types of layering can improve modeled API ecosystems’ understandability and clarity. Two guides were also created to show how to layer e³value models, followed by a single guide focused only on splitting components into layers. The results indicated that the layering methods improved the understandability and clarity of the API ecosystem and that the difficulty of creating layered e³value models lies mostly in picking a layer for each component.

Keywords—e³value models, layering, API ecosystem

I. INTRODUCTION

Many companies are developing their own APIs (Application Programming Interface) to allow access to their data and services. These APIs can be used internally by companies, or be open to also third party developers. Opening the access to the APIs to third party developers can provide additional value from the new applications while giving the owner of the API the control of what assets can be used by the developers [1]. It is however not always obvious where and how an API provides value and fulfils business needs, as the APIs are interfaces. An API ecosystem can be visualized in an e³value model to help make sense of the flow of value.

e³value modeling is a type of requirements engineering and conceptual modeling used to better understand business ideas, their profit drivers, and how the business can provide value to customers through visualization [2]. The e³value models provide precise description and analysis of the technological and economic aspects of a business model to stakeholders [2] [3]. e³value models are good at visualizing API ecosystems because of their focus on strategic analysis, a simple visual syntax, and as they can help understand the value of APIs to potential users [4]. See Figure 1 for a simple example of an e³value model.

A problem with e³value models and other representations of such ecosystems [5] is that they can be very large in size and complex to understand for larger systems. It can be therefore difficult to identify the API components from the model and find important value flows. Layering an existing goal model (representing a software ecosystem) was found useful in identifying the role of the actors and seeing if any actors are missing [4]. Layering the e³value models was also tried in a previous study, but no easy way to do that was

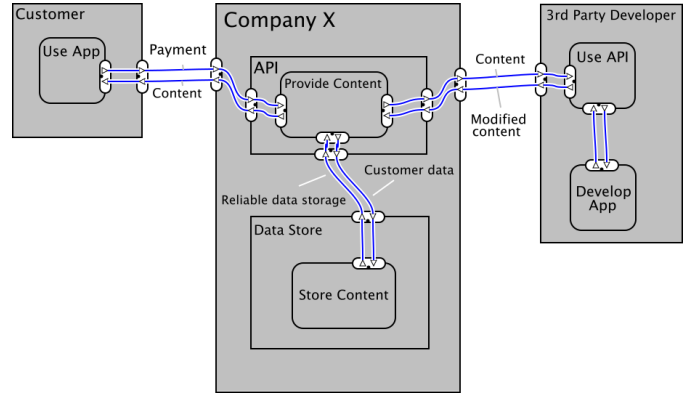


Fig. 1. An example e³value model

found [4], and therefore no solution to make the models more understandable and clear currently exists.

In this study we tried to find a way in which layering can be applied to e³value models: both what is the best way to layer them and how to split components into different layers. The layers we used in the study were developed as part of an API framework which divides an API ecosystem into four different layers [1]. These are described in more detail in Section II Related work. The use of these particular layers was found helpful in strategic analysis to understand API business value and usage [4].

What makes layering e³value models more complicated than other types of diagrams is containment (elements inside other components that might belong to a different layer than the other elements in the same component). An example of this can be seen in Figure 1 where the *Company X* actor has two components, API and Data Store and each belongs to a different layer. In addition to the containment, it can be hard to identify the API as it can appear as different types of elements, or just as a value flow. Also identifying the API might depend on the context the model is viewed from - for example changing the API of focus might move some components to a different layer as their interaction with a different API may be different compared to another API.

A. Companies

This thesis is a part of Chalmers University of Technology's and University of Gothenburg's Software Center ¹ API Strategies Project, that focuses on how APIs can provide value to businesses and customers. Five companies from different fields have agreed to participate in this study. The focus of the project is on their API governance and -strategies, with the thesis targeting e³value models.

These companies have previously visualized how their APIs provide value in an API ecosystem in the form of an e³value model [4]. The company descriptions are based on related work [6] that focused on the same companies:

- 1) Company A focuses on an API that was still in development during the study (out of multiple APIs to physical devices), with the focus being on the role of third party developers communicating with this API.
- 2) Company B focuses on a partially developed internal, but geographically distributed API. The role of the API is automating tasks and limiting risks by providing functionality without allowing direct access to the hardware.
- 3) Company C focuses on a partially operational internal API used for reusing common features across the company.
- 4) Company D focuses on multiple internal APIs managing access to different technologies, that are either active, or near retirement.
- 5) Company E focuses on an internal API still in development meant to ease access between their devices and mobile or cloud services.

This paper is organised as follows. Section II presents the related work on e³value models, visualization, and API ecosystems. In Section III, the research methodology is explained, including a short description of the three cycles, followed by validity threats. Section IV describes the three design science cycles in detail with their results. Finally, in Section V we present the discussion, and Section VI is conclusions.

II. RELATED WORK

A. Background

1) *e³value modelling*: The way in which e³value models can be utilised in modelling business needs has been written about and detailed in the literature [2] [3]. In e³value models, an e-Business model can be described as a value-network, where a business is composed by a number of actors creating value through value activities, which produces economic value [3]. The models, which show the enterprises, customers and the flow of value in terms of goods, services and money, can be used for analysing whether an enterprise will be economically viable [7]. They can be also used as a basis for analysing potential cash flow, and visualising customer needs [7]. Even the simplest form of e³value modelling can be used to help explain how customers gain value from a business idea, and a more complex one can show the details of value flow in a large organisation.

The elements forming the e³value ontology are listed and explained below, and illustrated in Figure 2:

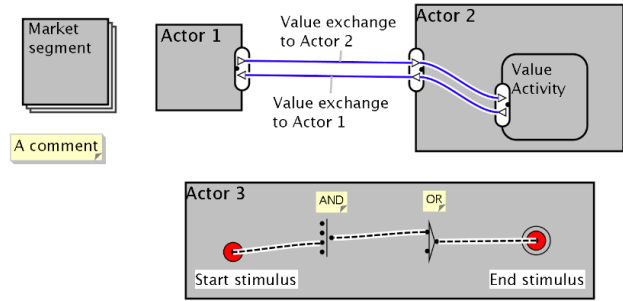


Fig. 2. e³value model legend

- Market segment: is used to show a number of actors that assign economic value equally. This could for example be used to show ‘Internet surfers’ in an internet cafe who pay for their connection by minutes [2].
- Actor: is an independent entity that could be used to represent a customer or a company [2].
- Value exchange: The blue lines are used to connect two value ports together. They represent the exchange of value from an actor to another. This value could for example be a service provided, or payment for the service [2].
- Value activity: is used to represent the action that creates the value. This could be for example providing internet access [2].
- Comment: can be used to leave notes or further explain something that might not be clear otherwise.
- Stimulus element: The red dots are used to represent different scenarios that can be started by an actor. These scenarios can have multiple possible endings.

Work on extending e³value models to provide additional or alternative visualization options has been limited. One paper presents how the the e³value model ontology can be mapped onto UML, to allow the use of UML tools to draw up e³value models [8].

2) *Layering*: Layering is a visualization technique which is considered one of the most efficient ways to reduce noise in a model and enrich the existing content [9]. Layering involves separating data and classifying it according to set criteria. Studies on UML class diagrams show that introducing layering to an existing system can give a “better understanding of the dependencies between the components of a system” [10].

To apply layering to the e³value models, a number of visualization principles have been considered to find the solution that is most efficient for them. The principles which will be used to extend the e³value models will be based on the graphical elements (horizontal and vertical position, shape, size, color, brightness, orientation and texture) which have been identified in literature as blocks through which visual representations can be built [11] (see Figure 3).

One assumption of layering is that it creates a hierarchy between the layers, that higher levels are dependent on the

¹<https://www.software-center.se>

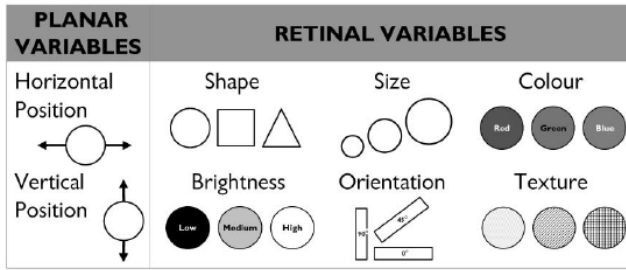


Fig. 3. Building blocks for visual notation [11]

lower layers [10]. Assumptions about the relationship between elements can also be drawn from the way in which elements are spatially organized in a diagram [11]. Creating unwanted associations and assumptions has to be taken consideration when layering the e^3 value models and picking the graphical elements.

B. Related project work in this project

A number of other studies have been completed as part of the Software Center’s project, which this thesis also belongs to. This includes creating the layering that would be applied to the models which represent the API ecosystem [1]. The four layers (see Figure 4) identified by the work are:

- Domain - the needs and events supported by application software. Examples of a Domain component are customer-components and people in the organisation.
- API Usage - the components that are using the API and usually gain value from it.
- API - the API or APIs inside the ecosystem.
- Business assets - these are components that the API gives access to. Examples are data, algorithms, or properties of a product.

These layers were used as the starting point for conducting the study in this thesis.

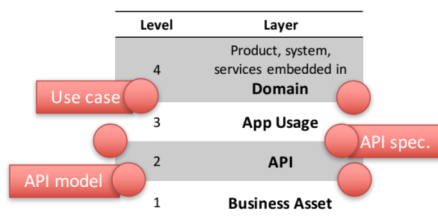


Fig. 4. Four-layer API architecture [1]

The e^3 value models for the participating companies have also been created in a previous sprint (sprint 13) of the same Software Center project [4]. In that study, the researchers mention that it was difficult to apply layers to these models.

Two other bachelor theses have been conducted as part of the same project, both dealing with goal models and how they can be applied to the API ecosystem [12] [13]. One of these

has already investigated adding layering to goal models [12] and the findings in their study indicated that the addition of layers to goal models had been perceived as beneficial by the participating companies.

III. RESEARCH METHODOLOGY

To find out how different kinds of layering visualization methods can be used to enhance e^3 value models and improve the comprehension of the role of APIs in them, the following research questions were formulated:

RQ1 What are effective ways to apply layering to an e^3 value model?

1. What type of layering makes it easier to understand the API ecosystem?
2. What type of layering makes it more clear to see which components interact with the APIs?
3. Considering ease of use, what methods can be used to layer e^3 value models?

RQ1 aims at answering how (or if) an e^3 value model can be layered to help in identifying the roles of the actors. With RQ1.1 different types of layering are compared to each other to find out if the layering can make the API ecosystem more easy to understand compared to the version without layering. The aim of RQ1.2 is to see if the layering types can make it more clear what components interact with the API compared to the version without layers. With RQ1.3 we aim to develop an easy to use method to transform a non-layered e^3 value model into a layered e^3 value model.

A. Design science research

We chose design science research as the research method because of its aim to solve problems by extending existing artefacts and/or creating new innovative ones [14]. A different research method that could have been considered instead was conducting a case study. A case study, however, has a goal of studying a phenomenon, providing an explanation for it or analyzing it as it is, whereas the aim of this study was to find out if we can create something new on top of the existing e^3 value models [15].

The study was conducted in three cycles. The aim of the first cycle was to investigate possible ways to layer an e^3 value model and evaluate which types of layering would make it easier to understand an API ecosystem as a whole (RQ1.1), and make it more clear to see which components interact with the APIs (RQ1.2). Based on the feedback of the first cycle, we planned the second cycle to investigate if we can create a potentially easy to use method by using software tools to move from a traditional e^3 value model into a layered version (RQ1.3). In the final cycle, we then tried to find an easy method for separating the components of a traditional e^3 value model into the four different layers used (RQ1.3).

The evaluation of the outcome of each cycle was done via online surveys using Google Forms, and interviewing one person at the end of the third cycle. The surveys were used to gather quantitative- and qualitative data from as many company API governance representatives as we could reach

from the five participating companies. Online surveys were used because of time limitations of the company representatives, and it being the easiest way for them to give feedback to us. The companies were also located in different regions around the Nordic countries. The participants of this study were 14 people from the five companies, 1-3 participants from each company. These people were members of the API governance teams in their companies, and have been working with the larger Software Center project for a longer time. They were thus selected for this study as well, and the same participants were used to gather data for each cycle. Due to the constraints of the larger Software Center project, we ended up having different data collection methods for each design science research cycle that was had. The reason for this was the limited time that we could directly interact with the participants to receive answers to our questions.

B. Design science cycles

1) *Cycle 1*: During the first cycle we came up with four visualisation options, rejected one of them and evaluated the other three with the companies. The three visualisation options were presented in a focus group meeting where the company representatives were participating via an online conference call. The visualisation options are presented in detail in section IV Results.

The visualization options were evaluated via an online survey sent out to the participants, and to those who had not been able to attend the focus group. The survey was sent to a total of 14 people, who are working with API governance in the participating companies, there being 1-3 people per company. We got four responses to the survey giving us a response rate of 28.6%. We asked the participants to evaluate three different visualization options. In the questions they were asked to assess if the understandability of the API ecosystem was easier or not, and if it was more clear or not what components in the model interact with the API. They were also asked which one of the visualization options they preferred and how they compared to each other. The full list of questions can be viewed in Appendix A. Our plan for this cycle was to ask questions during the focus group in the style of group interview to get qualitative data on what the participants thought about the visualization options. There was, however, not enough time to have this group interview, and we ended up sending the interview questions as a short survey with qualitative questions to the participants instead.

2) *Cycle 2*: During the second cycle, the feedback from the previous cycle was used to create two methods to layer e^3 value models. These methods were presented as two guides which were emailed to the company representatives, as no focus group meeting was scheduled for this cycle. These guides are described in detail in Section IV Results.

The evaluation was done via an online survey, similar to the one in the first cycle. The survey was sent to 14 people, out of which 5 replied to the survey and 2 added in comments separately via email. This gives it a response rate of 35.7%. We asked the participants to mention what they thought of the

two guides, if they made it easier to transfer an e^3 value model to a layered e^3 value model and if they would use this method for that purpose. The full survey questions can be found in Appendix B.

3) *Cycle 3*: In the third cycle another focus group meeting was held with a number of company representatives attending via a video conference call. We presented an updated version of the methods from cycle 2, which focused only on how to decide in which layer the components of the API ecosystem belong to. For the evaluation, we had again planned to have a survey after the focus group meeting. The focus group meeting turned out to go faster than expected and we had the chance to ask our interview questions live. One person participating in the focus group answered the questions presented in the meeting (as part of the interview), and one person (out of the 14 it was sent to) answered the online survey, giving it a response rate of 7.1%. The questions asked if this method made it easier to identify the layers in an e^3 value model, if they would follow this specific method, and if they would change anything about it. The full survey can be viewed in Appendix C.

C. Threats to validity

The following threats to validity were discovered while conducting the study, and are presented together with the mitigation strategies used to avoid them:

1) *Construct validity*: One of the first threats discovered was mono-method bias, as surveys were the main means through which data was collected for the evaluation stage of each cycle. It was possible that this type of data collection method would not provide enough data to properly evaluate the entire cycle. To mitigate this, we have tried to have the surveys be detailed enough to provide enough qualitative data to cover the criteria by which the artefact was going to be evaluated [16].

Another threat to construct validity is possibly not taking into consideration the negative side effects of applying layers to the e^3 value models. For example an unintended consequence could be the increased complexity of maintaining the models, where adding a new component to the model after layering would force one to also think about what layer does that component go into. To mitigate this, the surveys used for evaluation attempted to find out if any specific solution also had some negative side effects that might not have been considered by us [16].

2) *Conclusion validity*: Reliability of measures was another threat discovered. The surveys used for the evaluation phase might not give as much qualitative information as for example interviews would, as during an interview the interviewee might want to explain his/her decisions in more detail than in a survey. To mitigate this, the surveys were presented to the thesis supervisor before being sent out to the company representatives [16].

3) *External validity*: The study is conducted in a limited environment with only five companies participating which are all part of the same larger project. It is therefore difficult to

evaluate how the layering solution can be generalized to other settings, but the fact that there were five companies involved in total still provided a varied amount of perspectives and data [16].

Two additional threats to validity were the low response rate to the surveys and the fact that the evaluation methods had to change between the different cycles. These were, for the most part, unavoidable, as there was no way to convince the participants to give additional replies and the evaluation methods were constrained by the bigger project in which this thesis was taking place.

IV. RESULTS

A. Understandability of the API Ecosystem

The following section aims to answer the question posed by Research Question 1.1. To try to find out if a certain type of layering makes it easier or more difficult to understand the API ecosystem in a e³value models, three different visualization methods were created in the first design science cycle using the companies' original e³value models (see Figure 5 for the original model and see Appendix D, E, F, G, for additional example images for all of the companies):

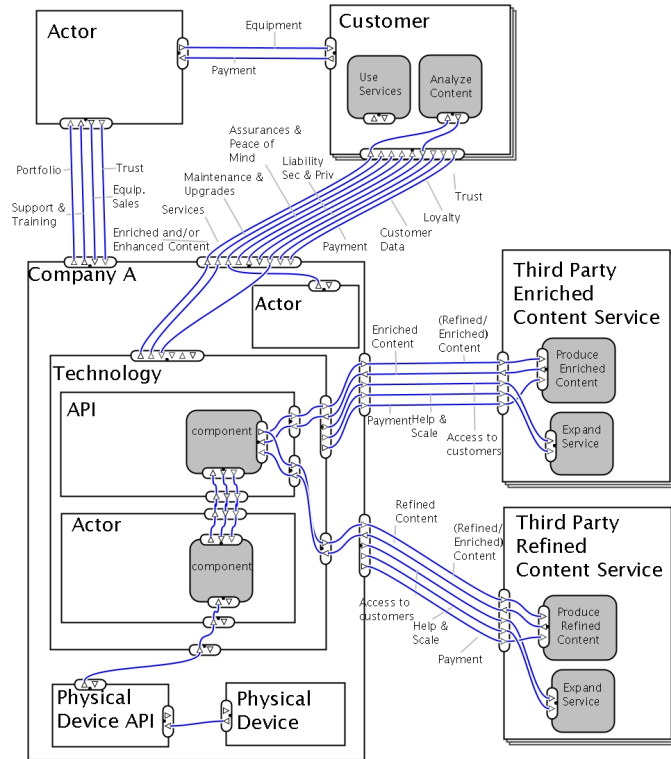


Fig. 5. Company A Original model

When looking into what were the possible visualization options for the first cycle, we looked at the visual variables defined by related work (see Figure 3) and tried several options manually. The horizontal and vertical position worked well for e³value models, as it was possible for us to separate the components into groups, depending on which layer they

belonged to. Size and shape were left out, as some of the components are already naturally larger than others, and the e³value model ontology already defines specific shapes that we could not change to indicate which layer a component belonged to. Texture, color and brightness worked similarly when it came to the e³value models, and we ended up using color-blind friendly colors, as the components' texts were still easy to read after applying the colors.

The resulting visualization options were:

- 1) In option 1 (V1), all of the layers were placed in a vertical [11] tower starting from Domain and going down to the Business assets (see Figure 6). This option is not meant to model a hierarchy of layers where one layer is dependent on the one below it (as can be done in traditional layering) [10]. Every layer can provide value to any other layer in the ecosystem.

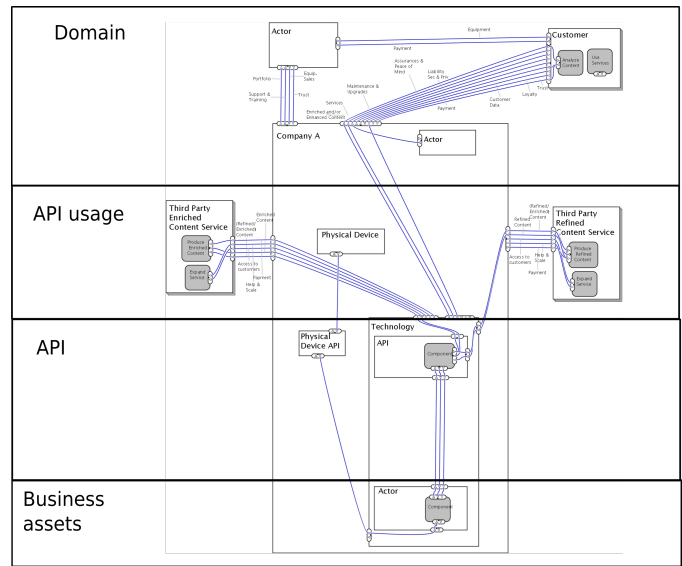


Fig. 6. Company A Visualization 1

- 2) Option 2 (V2) focused on adding a horizontal dimension [11] to the layers by placing them in a 2x2 grid, with the intention of preventing the lines from overlapping when components from the Domain layer had to provide value to those in the Business Assets layer, and vice-versa (see Figure 7). Similar to the first visualization option, ordering the layers both horizontally and vertically is not meant to provide any hierarchical ordering of the layers.
- 3) Option 3 (V3) focused on the variable color [11], each component of the e³value model was colored differently to show in which layer it belonged (see Figure 8). Colors were considered appropriate for this visualization option as they can separate elements without also creating an ordering as far as importance is concerned [17].
- 4) A fourth option (V4) was considered using texture [11] as its main way to separate layers, but was discarded and not presented as it was considered too cluttered (see Figure 9).

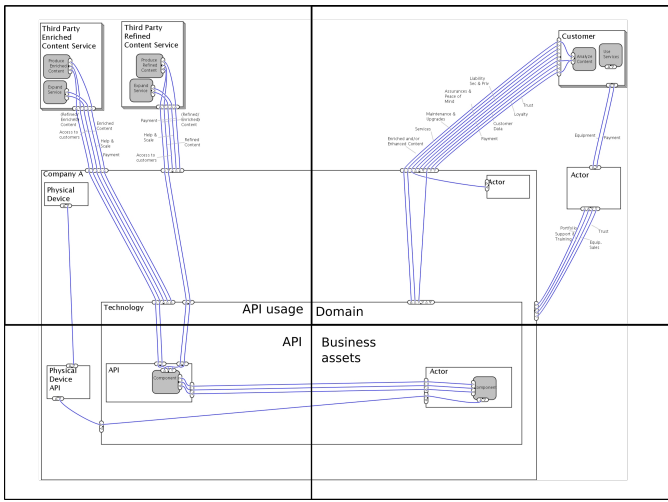


Fig. 7. Company A Visualization 2

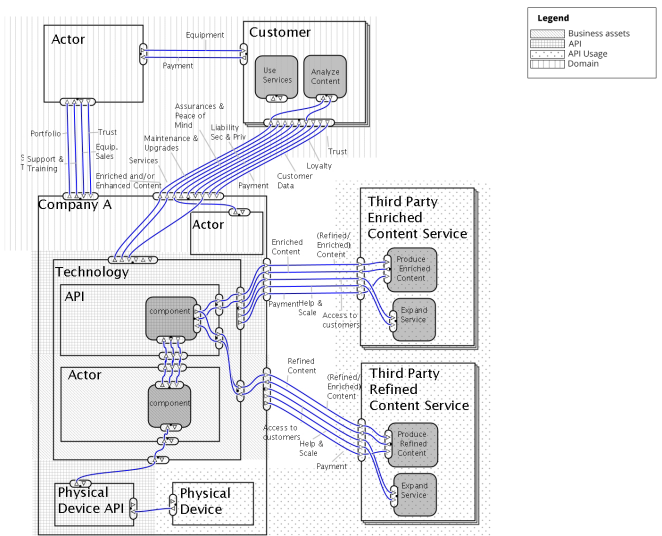


Fig. 9. Company A Visualization 4 (not presented and evaluated)

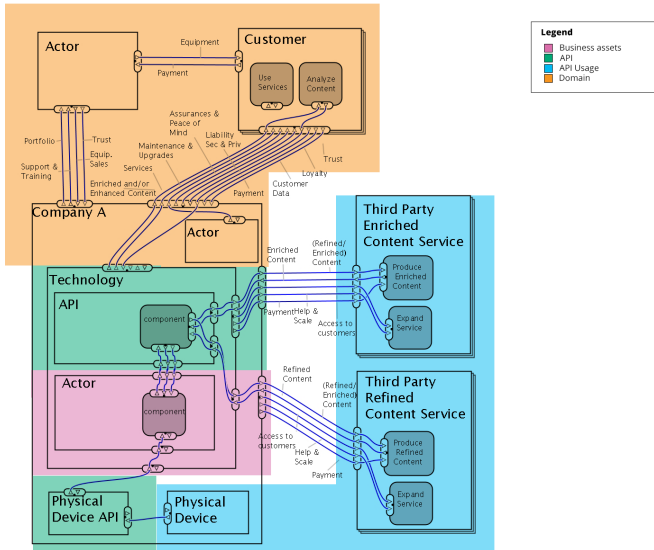


Fig. 8. Company A Visualization 3

One difficulty that arose when creating the visualization options was when the issue of containment came up. Containment, in this case, refers to e^3 value model actors which are contained in other actors. In some cases, the outside actor could belong to a different layer than the inside actor. To solve this, the inside actor was the one that took priority, in most cases.

Another issue was that sometimes the API is not represented as an actor in the diagram. There can be cases where it is represented as a value flow. In these cases, to allow the diagram to properly show where the API component is located, it was necessary to add an API actor on top of the value flow.

To answer the research question, the three visualization options were evaluated through surveys. The options were first evaluated separately, followed by one question that compared them to each other. The first question in each section asked about how the different visualization options affected the

understandability of the API ecosystem, and why: “Has the visualisation in the first/second/third option made it easier to understand the API ecosystem? Why, why not?”

The answers to this question have been summarized in Table I and the full results of the survey can be seen in Appendix L. To summarize these results, the answers to this open question were categorised and counted based on how many people answered similarly for each visualisation option.

TABLE I
SUMMARY OF QUESTION 1 RESULTS FOR CYCLE 1

Visualization	Number	Comment
V1	3 out of 4	easier to understand
	1 out of 4	easier to understand for smaller ecosystems, but not for larger ones
V2	2 out of 4	easier, but the 1st visualization was better
	1 out of 4	easier to understand
V3	1 out of 4	gives an interesting angle to the value chain
	2 out of 4	easier to understand
	1 out of 4	it did not give any additional information
	1 out of 4	it is not as good at the original model

B. Clarity of the API ecosystem

The following section aims to answer the question posed by Research Question 1.2. The same three visualization options of cycle 1 were evaluated using questions in the same survey as those for RQ1.1. The aim was to find out if the layered versions made it more clear or less clear to see what components interact with the APIs in a e^3 value models.

The second question in each section of the survey was: “Does the first/second/third visualization option make it more or less clear what components interact with the API? Why, why not?”

The answers to this question have been summarized in Table II and the full results of the survey can be seen in Appendix M. To summarize these results, the answers to this open question were categorised and counted based on how many people answered similarly for each visualisation option.

TABLE II
SUMMARY OF QUESTION 2 RESULTS FOR CYCLE 1

Visualization	Number	Comment
V1	2 out of 4	less clear
	1 out of 4	more clear for simple API ecosystems
	1 out of 4	more clear
V2	2 out of 4	more clear
	1 out of 4	as clear as the original model
	1 out of 4	less clear
V3	3 out of 4	less clear
	1 out of 4	more clear

C. Additional results about the visualization options

In addition to the questions presented above, the survey also asked additional questions about which visualization option was preferable and why. One participants also commented that the first visualization was helpful in finding missing business values and another participant said that the layering will help to understand where to look for specific things.

For the second visualization, three participants pointed out that they were not familiar with this type of layout, but one person pointed out that it gave it an interesting angle. “A bit confusing when not used to it. Alternative 1 is more intuitive, but this one might give better use of screen space with less crossing of lines.”

The third visualization was considered easy to understand by two people because it was “easy to navigate and remember as opposed to monochrome uniform boxes.” However, another two participants did not find that this gave any extra information compared to the other two visualization types and one of them found the grouping “less supportive than the original, since the grouping hides the details of the value model.”

D. Methods to apply layering to e³value models

The following section aims to answer the question posed by Research Question 1.3. To find out what way method is considered easier to use, two guides showing two different ways in which an e³value model can be layered were created:

- The first guide was a set of straightforward steps using the e3 editor tool to try to effectively transform a traditional

e³value model in to a vertical stack style layered model. It contained three parts, which showed how to separate the components into layers, move the components into their correct position and overlay the layers on top of them, and

- The second guide, also containing three parts, but using an intermediate step. The intermediate step involved creating an e³value model where each of the components was colored according to the layer it belonged to, and this intermediate model was then used as a guideline when applying the layers.

The full guides can be seen in Appendix I and J and the parts have been summarized in table III.

To evaluate these two methods, the respondents were asked to rate and comment on the usability of the guides. The answers to these question have been summarized below and the full results of the survey can be seen in Appendix M.

The respondents were asked to rate on a scale from 1 to 4 how likely they are to utilize these methods, how useful the methods are, and how easy to use they are. The responses have been presented in Figures 10, 11 and 12.

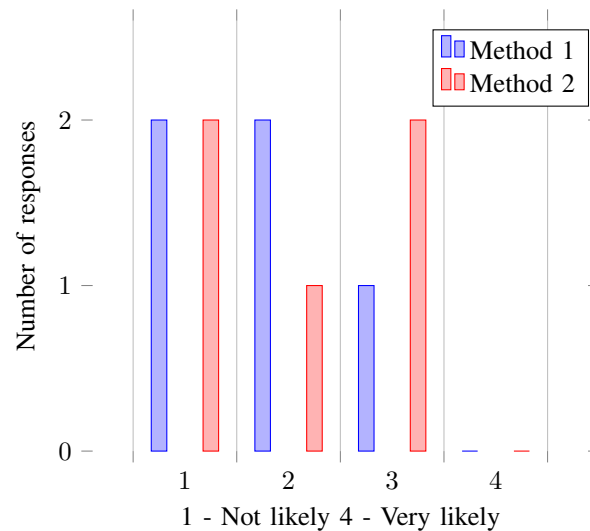


Fig. 10. How likely are you to utilize this method of layering e³value models?

When asked why they would utilize the guides or not, and what they would like to see different, many comments pointed out that they were too tool specific and did not provide enough details:

- “The method description is almost missing. It is basically captured in these few statements.”
- “A focus on the actual activities, how to identify which components belong to a certain layer. Identify what is missing and what is superfluous. Hence a description of the process, not the tools used to realize the process.”
- “I find the lack of process description support troublesome, while the tooling description gets in the way of the real value creation.”

TABLE III
OVERVIEW OF METHODS CREATED FOR CYCLE 2

Part	Method 1	Method 2
Part 1	Familiarize yourself with the original model.	Familiarize yourself with the original model.
	Check description of the layers, identify the API(s) and decide which layer the components belong to.	Import model picture into a photo editing tool. Decide what component belongs to what layer and draw colored boxes on top of the components, using different color for different layers.
Part 2	Reorganize the model in e ³ value editor tool by moving components around to roughly four different rows, each row representing a different layer.	Use the resulting image to help you reorganise the model in e3 editor by moving components around to roughly four different rows, each row representing a different layer.
Part 3	Export the model as an image and draw layer lines in a photo editing tool.	Export the model as an image and draw layer lines in a photo editing tool.

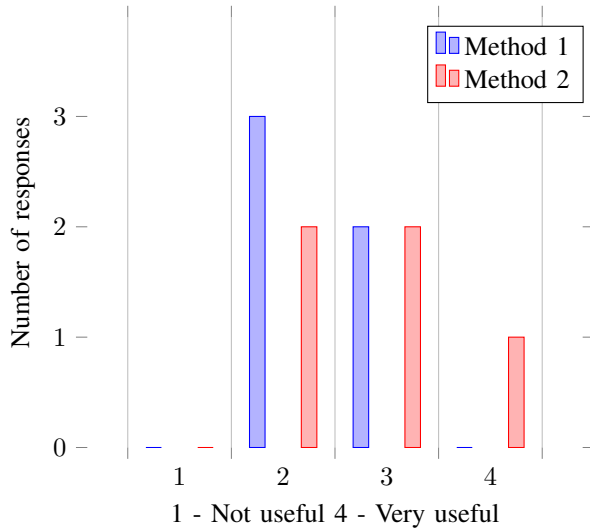


Fig. 11. How useful do you think this method is for creating a layered e³value model?

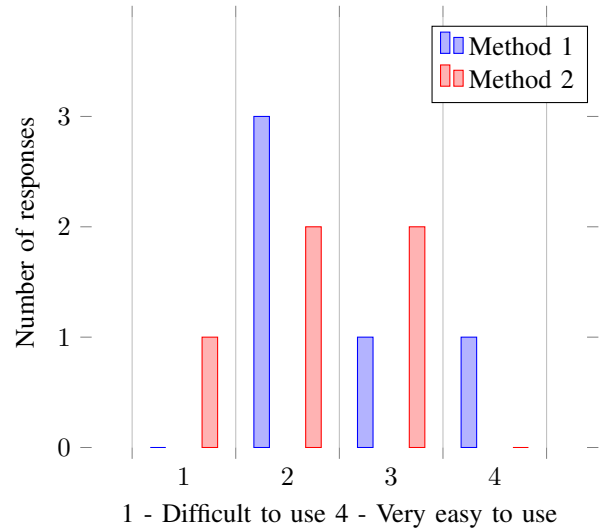


Fig. 12. How would you rate the ease of use of this method?

E. Categorizing the components into layers

The following section aims to answer the question posed by Research Question 1.3. As part of the third cycle, the focus was on creating a single guide that would help in figuring out in which layer each component of the API ecosystem belonged to. The feedback from the second cycle indicated that the respondents were more interested in knowing how to assign the individual components into their correspondent layer, than in the more tool focused approach of the two previous guides.

To complete this task, we found common elements which help in recognizing what layer each component belongs to. Related work described the API Framework layers in more broad terms, so this had not been done yet in such detail. As a result, an updated guide was created with steps on how to identify which components go to each layer. The guide had four major steps, each step explaining how to choose components into a single layer, with visual examples meant

to assist in picking the correct layer as well. See Appendix K for the steps in the guide dealing with identifying the layer for each components. The more detailed information on how to separate components into their corresponding layer will be useful for the companies in the future, when the bigger project is working on making modeling methods transferable for them. Right now creating an e³value model, and applying the layering to it requires some knowledge on how they work. These steps can be summarized as:

- 1) start by identifying the API of focus. It might not necessarily be named as simply as an 'API', but could be some kind of an interface component instead, or just a value exchange. If there is no actor- or value activity component representing an API, but it is simply a value exchange, create a new actor in the middle of that value exchange. Name this actor and mark it as the API of focus, then
- 2) identify the business asset that the API gives access to. These are often things like data, algorithms, or properties

of a product. In many cases, the business asset components provide value to the APIs (have a value exchange flow to the API component). Then

- 3) identify the API usage components which gain value from the API. API usage components usually only gain value from the API component and do not necessarily return any value back. And finally
- 4) identify Domain components, which should consist of everything that remains. These are usually customers, or people working in the organisation.

A survey and interview were conducted to gather feedback on this method, and the respondents were asked: “Does this method make it easier to identify the layers in an e^3 value model?”

To that question, two respondents out of two says that yes, it is easier. When asked if they would use the method however, the results were split, with one person saying that they would use it, while another mentioning that they would not use it unless there was adequate tool support for it.

The full results of the interview and the survey can be seen in Appendix O and P.

V. DISCUSSION

A. Choosing a visualization option

During the first cycle, three visualization options were presented to the companies. Based on the feedback received, a majority of respondents indicated that they found the first one (the classical vertical stack layering view) to be the one that made the model to be easiest to understand.

The visualization option with the 2x2 grid was perceived to perhaps give better use of screen space compared to the first visualization option and bring more clarity to the value flows but was perceived unusual or odd, and less intuitive than the 1st visualization option by the participants and we chose to not continue with this visualization option. It is likely that the different placement of layers made it so that there would be an additional element that the viewer had to get used to before understanding the API ecosystem. This could explain their hesitation about the visualization option. Perhaps the familiarity of the first visualization option made it easier to focus on the important aspects of the API ecosystem, compared to the second visualization option where the participants had to first learn how to correctly view how the layering works.

On the other hand, the colored version did not provide all of the information that the respondents were expecting. It was thought to be easy to understand, but not to provide extra information compared to the original e^3 value models, and might make the information about the value flows more difficult to see. Some of the answers indicated that it would work better as an intermediate step in layering the e^3 value models, as opposed to a complete layered diagram. Based on that comment, the colored version was then utilized in one of the guides of the second cycle showing how to apply the layers to an e^3 value model.

The results show that each visualization option had at least one respondent saying that the ecosystem was easier

to understand. The first option was the one that got the most amount of support, with the rest having comments that indicated various shortcomings for that visualization option. It can therefore be considered that the answer to RQ1.1 is that visualization 1 (the classical layering stack) makes it easier to understand the API ecosystem.

The results also show that each visualization option had at least one respondent saying that it was easier to understand what components interact with the API. The analysis of the results indicates that the second visualization option might have provided the most clarity, with the first option receiving a similar amount of positive and negative feedback. The answer to RQ1.2 is that either the first or the second visualization option provide the most clarity, but the limited amount of responses make it difficult to create a meaningful distinction between the two.

This confirms that introducing layering to an existing system can give better understanding of the components of a system, as has also been suggested in existing literature [10]. Based on one respondent’s answer, this is likely due to the layering giving cues where to find the information that someone is looking for. Having a set structure to a model also likely reduces noise and can be used to enrich the models [9].

B. Creating a method to layer e^3 value models

The feedback received indicated that the respondents were not interested in a tool-specific solution on how to create the diagrams, but were rather interested in a guide to help them identify which component belonged to which layer instead. One of the difficult aspects of this task is that the exercise works best with models which are complete, with all of the value flows consistently labeled, and all of the guides have these as a prerequisite. Presumably, this prerequisite would not be a problem for someone that is actively working with the API ecosystem, who understands what component is what.

The results answer RQ1.3 by indicating that a tool-specific method is not what they were looking for, nor did it make the process easier. Instead, the easy to use method to layer e^3 value models simply contained the description of the layers, as well as pointers on how to differentiate between each layer and to pick where each component belonged. The tools to complete this task were left to the discretion of the person layering the e^3 value model. A solution to the difficulty of choosing what API framework layers the components go to could be further clarifying the description of the layers and giving some examples of typical components, as this was shown to be useful in the results received for the third cycle guide, which tried to exemplify the layers as much as possible, based both on experience, and on related work within the project [1].

One aspect that would impede on the method being easy to use is that respondents indicated that improper tooling support would make them not inclined to use this, as they would wish for a reversible process, where they could inspect one layer at a time and go back to a non-layered model with a push of a button. All respondents indicated that they would be more interested if there was a single tool support to apply the API

ecosystem layers to the e^3 value models. This can be a point that can be taken into consideration for future work, where there would be a tool in which all the steps of the layering an e^3 value model could be done. For the time being, the easiest perceived way remains pen and paper.

This need for a single tool support for layering a model has been mentioned before in another study for UML diagrams [10]. The study suggests that even though the process of layering of components is difficult to automate, a tool could still prove useful by trying to layer a model by following certain rules, and finally asking a human to confirm or reject them. We believe that this kind of a tool could help in creating layers for models which share the same properties as e^3 models (in particular, containment).

One problem that could impact the results is the different data collection methods that were used to evaluate the artifacts and the limited response rate. If there had been chance to have the interviews during the focus groups, perhaps we could have gotten more detailed data about the different visualization options and the guides, than we did now using the surveys. The participants didn't necessarily have the time to answer our survey, perhaps leading to a lower response rate, whereas more of the participating people could have answered had we had the chance to ask the interview questions during the focus group. This makes it difficult to generalize our results as we cannot be sure if this is the common take on layering the e^3 value models, or just the opinion of a few participants. This is partly due to us being constrained by the larger Software Center study as there were many different studies going on simultaneously, and the participants did not necessarily have the time to answer our surveys.

In future work, the results of this study can also be used to apply layers to e^3 value models, or different models with containment. The results show that creating a structure with vertical layering in these cases, makes models with this characteristic more easy to understand by separating the model into well-defined categories, and possibly making the interactions between components more clear. As this is part of a bigger project, the expectations are that the layering method proposed will be used by the participating companies as part of their future work with e^3 value models, with the other researchers of the project helping them to transfer the methods to different use cases. One finding of the larger project that came up during the planned project meeting indicated that having one person on the team who understands the models is vital in transmitting the knowledge to the other members of the team.

VI. CONCLUSION

In this study we tried to find an effective way to layer an e^3 value model and an easy to use method for transforming a traditional e^3 value model into a layered version. We found that given three different options to evaluate, the participants preferred the traditional layering option. Following that, we presented two guides on how to create a layered e^3 value model, but they were considered too tool specific. A final cycle was conducted where the focus was on creating a guide to help

identify and separate components of a e^3 value model into four different API framework layers. In future studies these findings can be expanded in two ways:

- 1) adequate tool to create layered e^3 value models from beginning to end, and to be able to dynamically focus on a single layer, and
- 2) applying these findings to layering other types of models with similar characteristics to e^3 value models, namely containment.

ACKNOWLEDGMENT

We would like to thank our supervisor, Jennifer Horkoff, for the patience and guidance offered during the time when this study was being conducted. We would also like to thank the company partners who took the time to answer our surveys and interviews and provided us with the data that we needed to conclude the study.

REFERENCES

- [1] J. Lindman, I. Hammouda, J. Horkoff, and E. Knauss, "Emerging perspectives to api strategy," 2017, (In submission).
- [2] J. Gordijn, "E-business value modelling using the e3-value ontology," in *Value creation from e-business models*. Elsevier, 2004, pp. 98–127.
- [3] F. A. Bukhsh and P. D. A. Silva, "Modeling e-business customization with e3value modeling," in *Frontiers of Information Technology (FIT), 2016 International Conference on*. IEEE, 2016, pp. 187–192.
- [4] J. Horkoff, J. Lindman, I. Hammouda, E. Knauss, J. Debbiche, M. Freiholtz, P. Liao, S. Mensah, and A. Strömberg, "Modeling support for strategic api planning and analysis," *9th International Conference on Software Business*, 2018, (In submission).
- [5] J. Joshua, D. Alao, S. Okolie, and O. Awodele, "Software ecosystem: features, benefits and challenges," *International Journal of Advanced Computer Science and Applications*, vol. 4, no. 8, 2013.
- [6] J. Horkoff, J. Lindman, I. Hammouda, and E. Knauss, "Experiences applying e3 value modeling in a cross-company study," 2018, (In submission).
- [7] J. Gordijn, E. Yu, and B. Van Der Raadt, "E-service design using i* and e3value modeling," *IEEE software*, vol. 23, no. 3, pp. 26–33, 2006.
- [8] C. Huemer, A. Schmidt, H. Werthner, and M. Zapletal, "A uml profile for the e3-value e-business modeling ontology," 2008.
- [9] E. R. Tufte, *Envisioning information*. Graphics Press, 1990.
- [10] V. Kumar and J. Deka, "Organizing uml class diagrams in layers," in *Information and Communications Technology, 2005. Enabling Technologies for the New Knowledge Society: ITI 3rd International Conference on*. IEEE, 2005, pp. 39–55.
- [11] D. Moody, "The "physics" of notations: toward a scientific basis for constructing visual notations in software engineering," *IEEE Transactions on Software Engineering*, vol. 35, no. 6, pp. 756–779, 2009.
- [12] J. Debbiche, A. Störberg, and P. Liao, "Applying goal modeling to api ecosystems: A cross-company case study," 2017, bachelor of Science Thesis in Software Engineering and Management.
- [13] F. M. Bedru, M. Freiholtz, and S. Mensah, "An empirical investigation of the use of goal and process modelling to analyze api ecosystem design and usage workflow," 2017, bachelor of Science Thesis in Software Engineering and Management.
- [14] R. H. Von Alan, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS quarterly*, vol. 28, no. 1, pp. 75–105, 2004.
- [15] D. E. Perry, S. E. Sim, and S. Easterbrook, "Case studies for software engineers," in *Proceedings of the 28th international conference on Software engineering*. ACM, 2006, pp. 1045–1046.
- [16] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in software engineering*. Springer Science & Business Media, 2012.
- [17] E. R. Tufte, *The Visual Display of Quantitative Information*, 2nd ed. Cheshire, CT: Graphics Press, 2001.

APPENDIX A
CYCLE 1 SURVEY QUESTIONS

- 1) Alternative view 1 (classical layering)
 - Has the visualization in the first option made it easier or not to understand the API ecosystem? Why, why not? [open question]
 - Does the first visualization option make it more or less clear what components interact with the API? Why, why not? [open question]
 - Other comments? [open question]
- 2) Alternative view 2 (2x2 grid)
 - Has the visualization in the second option made it easier or not to understand the API ecosystem? Why, why not? [open question]
 - Does the second visualization option make it more or less clear what components interact with the API? Why, why not? [open question]
 - Other comments? [open question]
- 3) Alternative view 3 (colors)
 - Has the visualization in the third option made it easier or not to understand the API ecosystem? Why, why not? [open question]
 - Does the third visualization option make it more or less clear what components interact with the API? Why, why not? [open question]
 - Other comments? [open question]
- 4) Layering in e³value models
 - How do the the models compare to one another? Which one of the layered models do you prefer? Why? [open question]

APPENDIX B
CYCLE 2 SURVEY QUESTIONS

- 1) Method one
 - How likely are you to utilize this method of layering e³value models? [scale from 1(not likely) to 4 (very likely)]
 - Why or why not would you utilize this method? [open question]
 - How useful do you think this method is for creating a layered e³value model? [scale from 1(not useful) to 4 (very useful)]
 - How would you rate the ease of use of this method? [scale from 1 (difficult to use) to 4 (very easy to use)]
 - What would you like to see different? [open question]
 - Other comments? [open question]
- 2) Method two
 - How likely are you to utilize this method of layering e³value models? [scale from 1(not likely) to 4 (very likely)]
 - Why or why not would you utilize this method? [open question]
 - How useful do you think this method is for creating a layered e³value model? [scale from 1(not useful) to 4 (very useful)]

- Does the intermediate step with colors make it easier to build the final layered model? [Yes/No question]
 - How would you rate the ease of use of this method? [scale from 1 (difficult to use) to 4 (very easy to use)]
 - What would you like to see different? [open question]
 - Other comments? [open question]
- 3) Both methods
 - Was there anything that made the layering difficult in these methods? [open question]
 - Would you prefer if there was software tool support to help in building the layered models? (a single tool that could do all required tasks in building a layered e³value model) [Yes/No question]
 - Other comments? [open question]

APPENDIX C
CYCLE 3 SURVEY QUESTIONS

- 1) Does this method make it easier to identify the layers in an e³value model? [open question]
- 2) Would you follow this if you tried to apply layering to e³value models? [open question]
- 3) What would you change in the method? Other comments? [open question]

APPENDIX D
(ARTEFACT) VISUALIZATION OPTIONS: ORIGINAL
MODELS

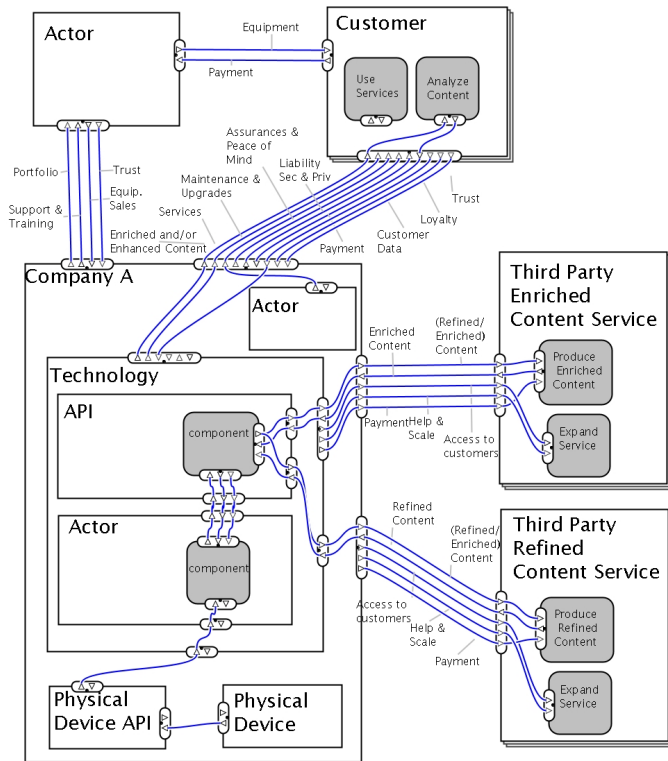


Fig. 13. Company A Original model

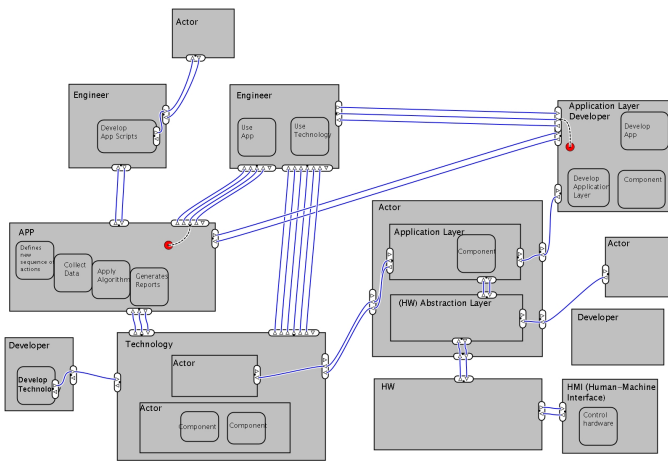


Fig. 14. Company B Original model

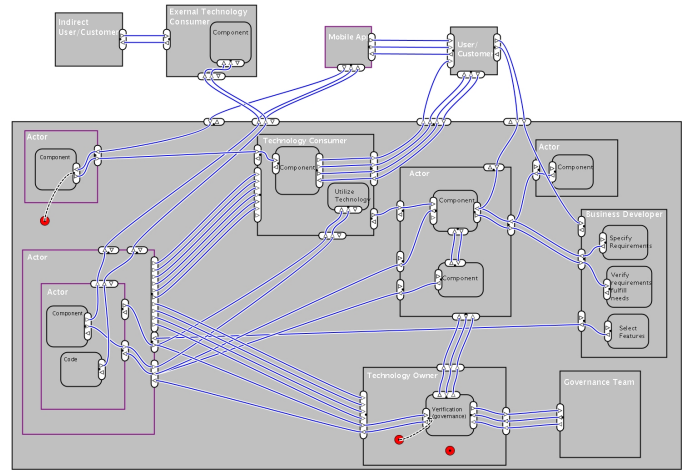


Fig. 15. Company C Original model

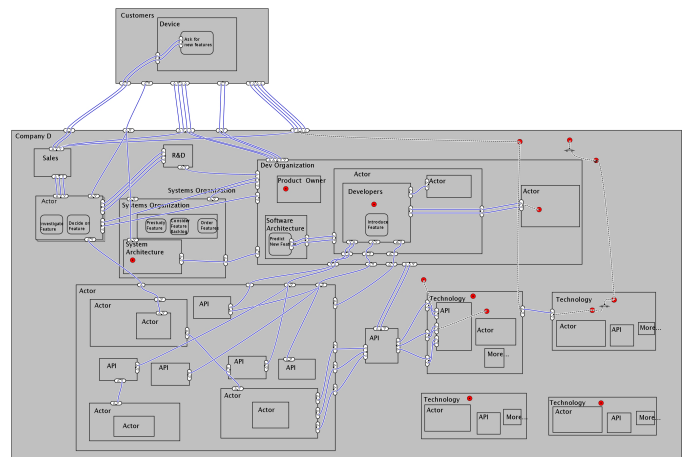


Fig. 16. Company D Original model

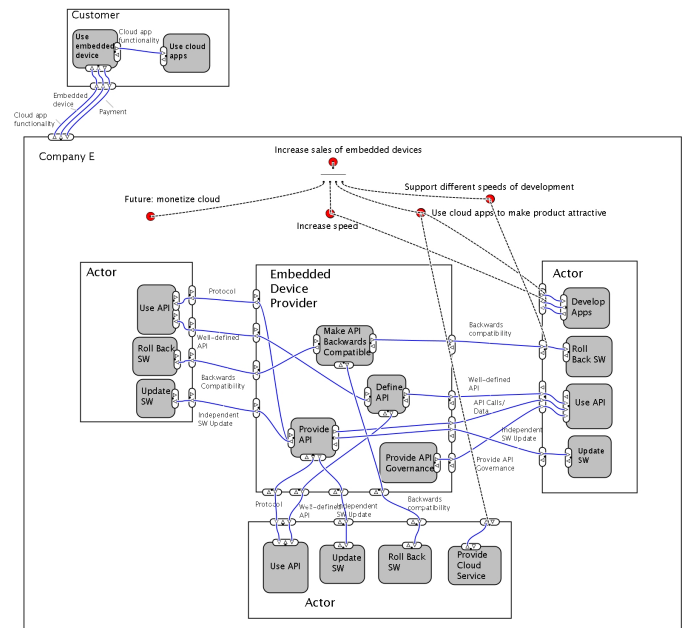


Fig. 17. Company E Original model

APPENDIX E
(ARTEFACT) VISUALIZATION OPTIONS: VISUALIZATION 1

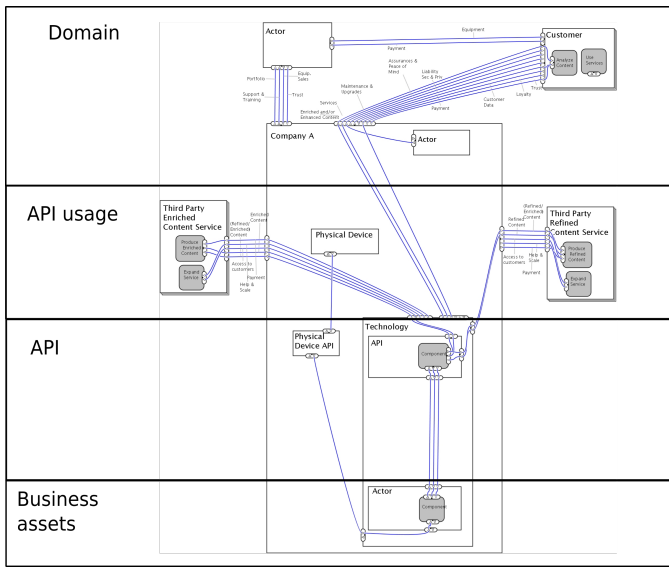


Fig. 18. Company A

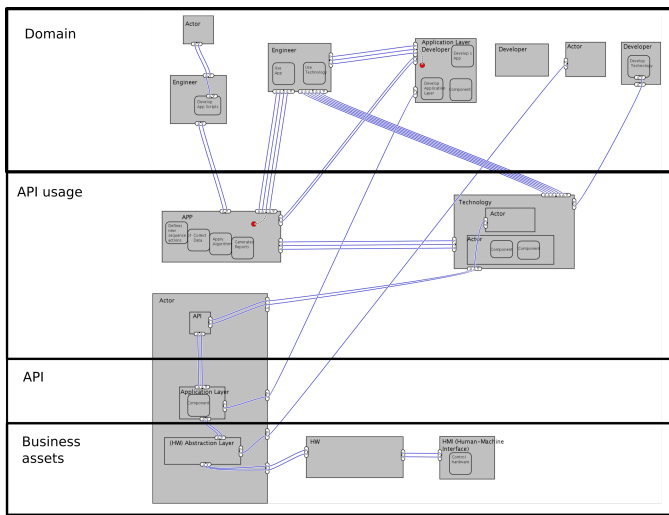


Fig. 19. Company B

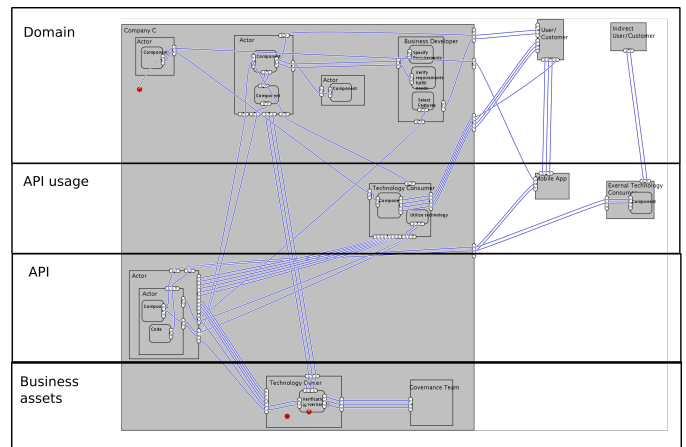


Fig. 20. Company C

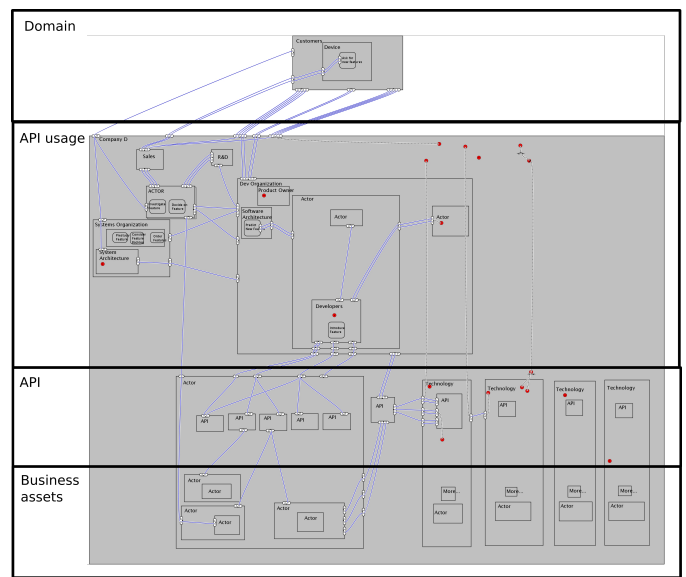


Fig. 21. Company D

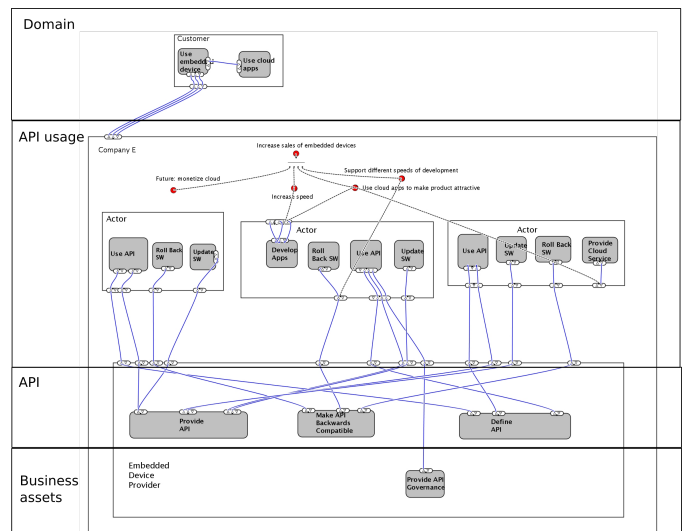


Fig. 22. Company E

APPENDIX G
(ARTEFACT) VISUALIZATION OPTIONS: VISUALIZATION 3

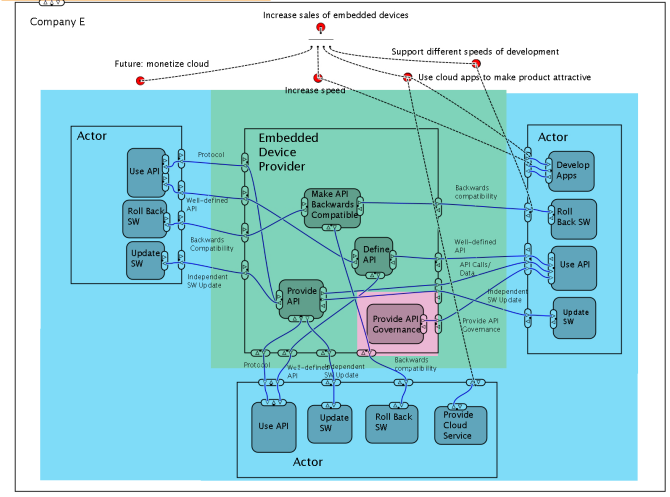
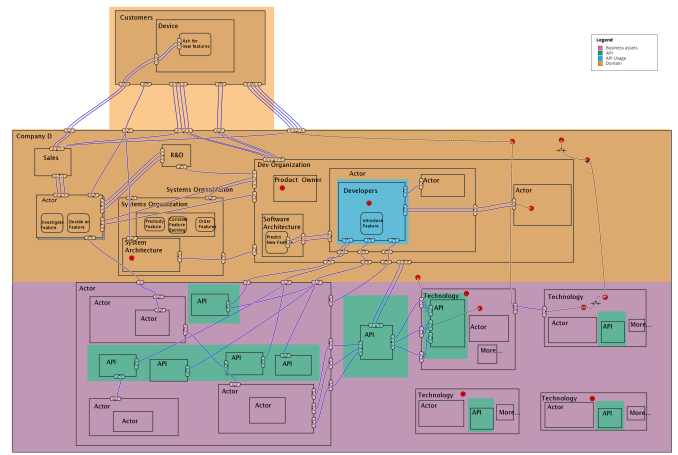
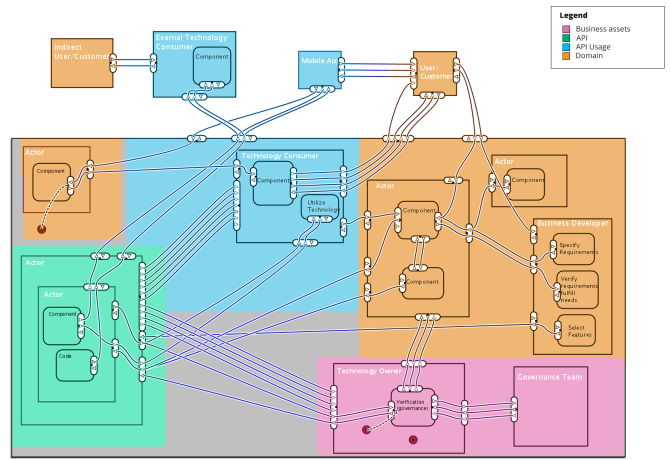
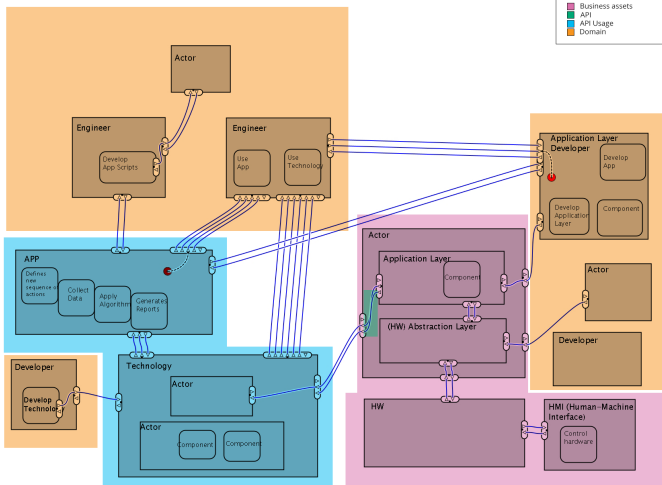
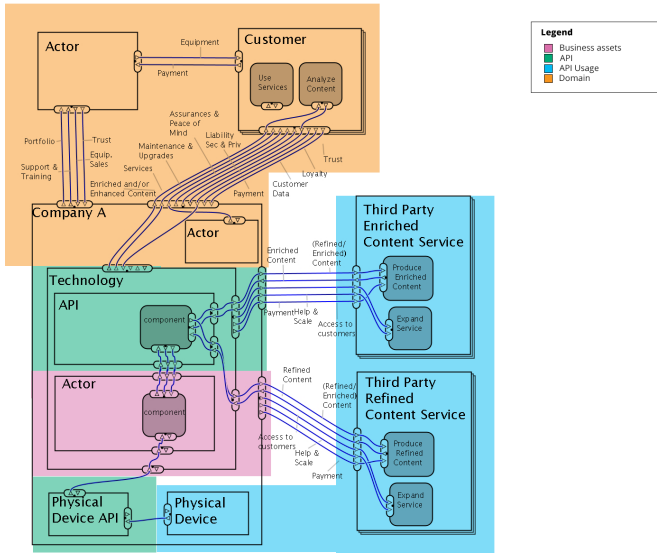


Fig. 32. Company E

APPENDIX H
 (ARTEFACT) VISUALIZATION OPTIONS: VISUALIZATION 4
 (DROPPED)

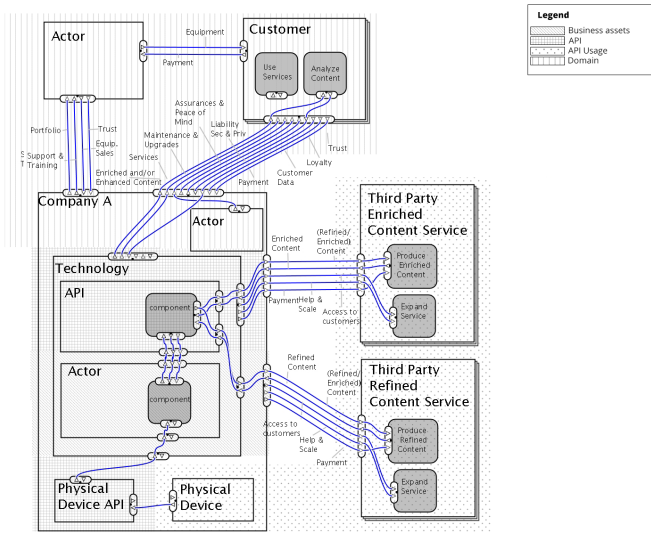


Fig. 33. Company A

APPENDIX I

(ARTEFACT) LAYERING METHODS: GUIDE FOR CREATING LAYERED e³VALUE MODELS VERSION 1

This guide is meant to present a method for applying the four-layer API ecosystem framework to e³value models. The guide assumes that there is an existing e³value model on which the layering will be applied and will not cover the steps used in creating this original model. A single model has been chosen as an example for the purpose of this guide, but the guide is meant to apply to any existing model.

Overview of the method

This method is divided into three parts:

- 1) In Part 1 the original (non-layered) model is used to identify one-by-one to which layer each component belongs to. The identification should start with the API layer, followed by the API usage layer, and then the Domain and Business assets components.
- 2) In Part 2, the components have to be placed in their correct positions, according to the previously decided layers. The layers are placed in a vertical stack, with the Domain layer at the top, going down towards the API usage, API and then Business assets at the bottom of the stack. To create this structure, all components could be placed in four rows, with each row representing the layer associated with the components.
- 3) In Part 3, the boxes representing the layers are drawn on top of the image resulting from the previous part. The layers are then labeled.

Currently there is no software tool that can complete all of the layering steps by itself. Any tool (analog or digital) can be used to complete the parts mentioned above. As far as software is concerned, it is likely that a program that can build e³value models will be needed, as well as an image processing tool. To exemplify in more detail how the layering can be performed, this guide uses the following tools:

- the e³value editor² for creating and editing the e³value models
- and Inkscape³ for applying the layers to the resulting images.

Part 1: Identifying the components for each layer

This section does not specify any recommended tools and can be completed in what way the user deems best, be it pen and paper or using digital tools.

- 1) Take a look at the original model and familiarize yourself with what each component does and what it connects to.
- 2) Check the description of the four layers (Domain, API Usage, API, and Business assets) as presented in related work [1].
- 3) Identify the API component(s) in the model.
- 4) Based on the previously identified API, identify the components that utilize the API and that are part of the API usage layer.

- 5) The components which remain can be separated into either the Domain or the Business assets layer.

Part 2: Reorganizing the model according to the layer
This section requires the e³value editor tool.

- 1) Open the e³value editor tool and open the .xsvg file corresponding to the original e³value model.
- 2) Make sure you can see the whole model (View - Zoom to fit / Zoom out)
- 3) Enlarge the canvas downward by enlarging one of the boxes. Clicking on the box brings out green dots on the borders where you can drag to enlarge
- 4) Move the components into roughly 4 different rows. Put Domain components on the top row, then API usage components in a row below it and in the last two rows there are API components, and Business asset components. Try to make the components not overlap each other.
- 5) Move the value lines as needed. This can be done by dragging a value interface on to a different place on the border of the component.
- 6) The completed model can be exported as an .svg file for further processing from the Tools - SVG File menu. Note: If parts of the model are missing when completing this step, then the entire image needs to be moved down in the editor to allow some space between the topmost elements and the title bar. Then the step needs to be repeated.

Part 3: Adding layers on the model

This section requires Inkscape to be used and three tools will be used. These are:

- the Pointer tool, used for resizing and moving objects;
- the Rectangle, used for drawing boxes, and
- the Text tool, used for creating text fields in which information can be written.

In addition to those, the “Fill and stroke” dialog which can be found in the Object menu will also be utilized for adjusting the colors and outlines of the layers.

- 1) Open the resulting .svg file (from Part 2) in Inkscape.
- 2) Using the Rectangle tool, draw a box around the components that form one of the layers. The box has to have a transparent fill, and a The Text tool can be used to label the layers. This has to be done four times for each individual layer.
- 3) Once the layering has been completed, the file can be saved as an .svg via the File - Save As menu.

²<https://www.e3value.com/software/>

³<https://inkscape.org/en/>

APPENDIX J

(ARTEFACT) LAYERING METHODS: GUIDE FOR CREATING LAYERED e³VALUE MODELS VERSION 2

This guide is meant to present a method for applying the four-layer API ecosystem framework [1] to e³value models. The guide assumes that there is an existing e³value model on which the layering will be applied and will not cover the steps used in creating this original model. A single model has been chosen as an example for the purpose of this guide, but the guide is meant to apply to any existing model.

Overview of the method

The method presented in this guide is an extension of the one in Version 1. It involves an additional step in the first part where an intermediate diagram is created by coloring the components according to their layers. This step was chosen as a result of feedback from the previous iteration where multiple layering methods were presented. The colored version was perceived as an effective way of visualizing the layers, and is therefore going to be used as a guideline for preparing the following parts.

This method is divided into three parts:

- 1) In Part 1 the original (non-layered) model is used to identify one-by-one to which layer each component belongs to. The identification should start with the API layer, followed by the API usage layer, and then the Domain and Business assets components. The next step involves coloring the original model using four different colors, each color representing one of the layers.
- 2) In Part 2, the components have to be placed in their correct positions, according to the previously decided layers and using the colored model as a guideline. The layers are placed in a vertical stack, with the Domain layer at the top, going down towards the API usage, API and then Business assets at the bottom of the stack. To create this structure, all components could be placed in four rows, with each row representing the layer associated with the components.
- 3) In Part 3, the boxes representing the layers are drawn on top of the image resulting from the previous part. The layers are then labeled.

Currently there is no software tool that can complete all of the layering steps by itself. Any tool (analog or digital) can be used to complete the parts mentioned above. As far as software is concerned, it is likely that a program that can build e³value models will be needed, as well as an image processing tool. To exemplify in more detail how the layering can be performed, this guide uses the following tools:

- the e³value editor⁴ for creating and editing the e³value models
- and Inkscape⁵ for applying the layers to the resulting images.

Part 1: Identifying the components for each layer

⁴<https://www.e3value.com/software/>

⁵<https://inkscape.org/en/>

This section requires the use of both the e³value editor, and Inkscape. Three tools will be used in Inkscape, which have been named:

- the Pointer tool, used for resizing and moving objects;
- the Rectangle, used for drawing boxes, and
- the Text tool, used for creating text fields in which information can be written.

In addition to those, the “Fill and stroke” dialog which can be found in the Object menu will also be utilized for adjusting the colors and outlines of the layers.

The colorbind colors (in RGBA format) that are suggested for the four layers are:

- Green: R-0, G-168, B-117, A-150
- Blue: R-0, G-185, B-242, A-150
- Orange: R-247, G-148, B-29, A-150
- Pink: R-218, G-111, B-171, A-150

- 1) Take a look at the original model and familiarize yourself with what each component does and what it connects to.
- 2) Check the description of the four layers (Domain, API Usage, API, and Business assets) as presented in related work.
- 3) Open the e³value editor tool and open the .xsvg file corresponding to the original e³value model.
- 4) The original model can be exported as an .svg file for further processing from the Tools - SVG File menu. Note: If parts of the model are missing when completing this step, then the entire image needs to be moved down in the editor to allow some space between the topmost elements and the title bar. Then the step needs to be repeated.
- 5) Open the resulting .svg file (from Part 2) in Inkscape.
- 6) Identify the API component(s) in the model.
- 7) Using the Rectangle tool drag and pull to draw colored boxes on top of the relevant components.
- 8) Based on the previously identified API, identify the components that utilize the API and that are part of the API usage layer. Repeat step 7 with a different color.
- 9) The components which remain can be separated into either the Domain or the Business assets layer. Repeat step 7 with a different color for each layer.
- 10) Once the coloring has been completed, the file can be saved as an .svg via the File - Save As menu and then used as a guideline for the following sections.

Part 2: Reorganizing the model according to the layer

This section requires the e³value editor tool.

- 1) Open the e³value editor tool and open the .xsvg file corresponding to the original e³value model.
- 2) Make sure you can see the whole model (View - Zoom to fit / Zoom out)
- 3) Enlarge the canvas downward by enlarging one of the boxes. Clicking on the box brings out green dots on the borders where you can drag to enlarge.
- 4) Move the components into roughly 4 different rows. Put Domain components on the top row, then API usage components in a row below it and in the last two rows there

are API components, and Business asset components. Try to make the components not overlap each other.

- 5) Move the value lines as needed. This can be done by dragging a value interface on to a different place on the border of the component.
- 6) The completed model can be exported as an .svg file for further processing from the Tools - SVG File menu. Note: If parts of the model are missing when completing this step, then the entire image needs to be moved down in the editor to allow some space between the topmost elements and the title bar. Then the step needs to be repeated.

Part 3: Adding layers on the model

This section requires Inkscape to be used. The tools used have been presented at the beginning of part 1 of this guide.

- 1) Open the resulting .svg file (from Part 2) in Inkscape.
- 2) Using the Rectangle tool, draw a box around the components that form one of the layers. The box has to have a transparent fill, and a The Text tool can be used to label the layers. This has to be done four times for each individual layer.
- 3) Once the layering has been completed, the file can be saved as an .svg via the File - Save As menu.

APPENDIX K
(ARTEFACT) GUIDE TO SEPARATE COMPONENTS INTO
LAYERS

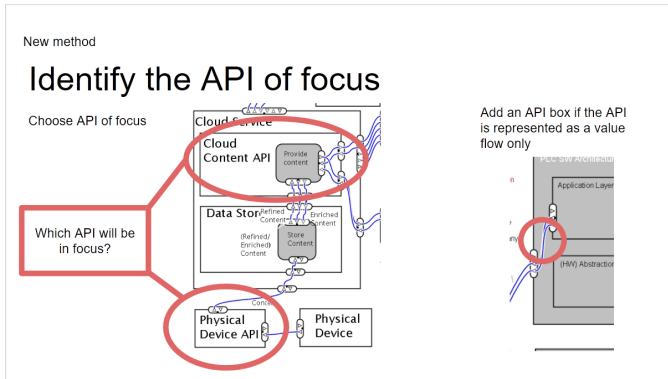


Fig. 34. Guide: Identify the API of focus

New method

Identify Domain components

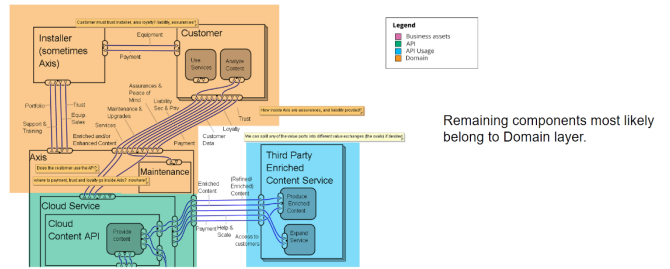
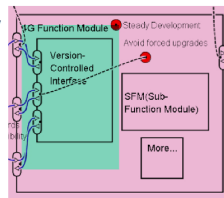


Fig. 37. Guide: Identify Domain components

New method

Identify the business assets

- These are components that the API gives access to. Usually things such as data, algorithms, or properties of a product.
- In a lot of cases the business asset provides value to API and doesn't necessarily get any value from API.



Green = API
Pink = Business assets

Fig. 35. Guide: Identify the business assets

New method

Identify API Usage components

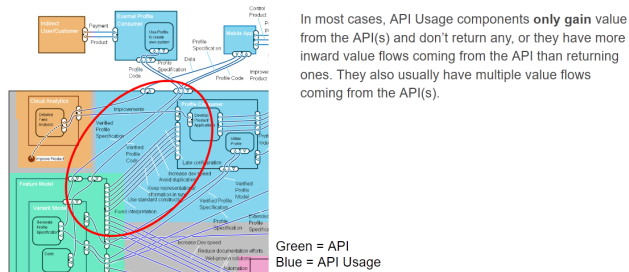


Fig. 36. Guide: Identify the API usage components

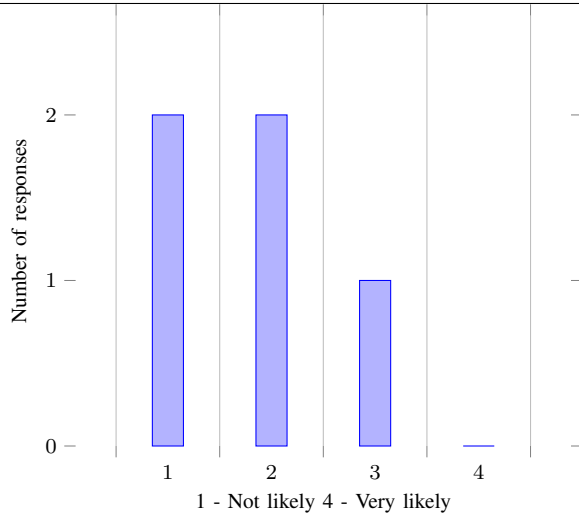
APPENDIX L
CYCLE 1 SURVEY RESULTS

<p>Alternative view 1 (classical layering)</p> <p>Q: Has the visualization in the first option made it easier or not to understand the API ecosystem? Why, why not?</p> <p>The architecture is easier to understand but it also shows if the model miss some business values.</p> <p>Easier for small/simple eco systems. Not much help for larger ones.</p> <p>Yes, different stakeholders are visible</p> <p>Yes, I found it beneficial to conduct a value based dicussion when the values are grouped related to what domain they belong to.</p> <p>Q: Does the first visualization option make it more or less clear what components interact with the API? Why, why not?</p> <p>Not really. This view is more the technical view about dependencies but we more like to highlight the value chains.</p> <p>Clearer for simple eco-systems.</p> <p>No, components are not clearly defined</p> <p>Clearer. Since the API as such is lift forward</p> <p>Q: Other comments?</p> <p>no</p> <p>I guess it makes most sense if you have many small eco systems that you want to compare. Then the layering will help understand where to look for specific things.</p> <p>Easy to see separation between layers</p>	<p>Alternative view 2 (2x2 grid)</p> <p>Q: Has the visualization in the second option made it easier or not to understand the API ecosystem? Why, why not?</p> <p>This view gives an interesting angle of the value chain. The customer is interested in the value of the asset and the API is only a mean to get to this asset.</p> <p>A bit confusing when not used to it. Alternative 1 is more intuitive, but this one might give better use of screen space with less crossing of lines.</p> <p>A bit easier but not as good as in alternative view 2, not so used to this layout</p> <p>Same as for first view</p> <p>Q: Does the second visualization option make it more or less clear what components interact with the API? Why, why not?</p> <p>Yes it does. You can more easy see the lines from the customer to the assets meaning showing the true value for the customers.</p> <p>I would say it is about equal to a layout based on association (which is how I would describe the original models). Might become worse if there are many APIs.</p> <p>No, same as for alternative 1</p> <p>Same as for first view</p> <p>Q: Other comments?</p> <p>no</p> <p>Compared to alternative 1 it seems not add more info but if the layout makes it cleaner compared to alternative 1 it may be an optional way to present it.</p> <p>I found the first view slightly more beneficial. Most likely because I am used to layering, and that layers as such provide a value besides the pure grouping, which is not satisfied in alternative 2.</p>
<p>Layering in e³value models</p> <p>Q: How do the the models compare to one another? Which one of the layered models do you prefer? Why?</p> <p>I prefer view 2 since this gave a clear view of the business and the technical side of the value chains.</p> <p>I would use the first one for presenting to uninitiated people, while the second variant would probably be better for internal communication between people who are used to it. The third one adds another dimension and is not mutually exclusive to the other two, but it could conflict with other uses of colors.</p> <p>Alternative 1, I think it is good to clearly see the layers when working and presenting the model.</p> <p>View one, besides providing clearer relations, the layerign as such adds a valuable dimension not provided in view 2.</p>	<p>Alternative view 3 (colors)</p> <p>Q: Has the visualization in the third option made it easier or not to understand the API ecosystem? Why, why not?</p> <p>For me, this did not give any extra information.</p> <p>Easier. Blobs of color are easy to navigate and remember as opposed to monochrome uniform boxes.</p> <p>Yes, it shows which parts are related to a specific layer</p> <p>I find this view less supportive than the original, since the grouping hides the details of the value model.</p> <p>Q: Does the third visualization option make it more or less clear what components interact with the API? Why, why not?</p> <p>Not to me. I think the other two was more clear to see value lines between different layers.</p> <p>Clearer. See above.</p> <p>No, same as for alternative 1 and 2</p> <p>Less clear</p> <p>Q: Other comments?</p> <p>Depends on the fact that no colors are otherwise used in the model, which would be a deal breaker for me. (A more sensible representation of E3 models would already use colors and/or patterns to distinguish between components/groups/roles etc rather than the questionable combination of outlines and arrows.)</p> <p>Good if you prioritize a good layout of the model, maybe hand-made, and still want to show different layers.</p> <p>I find this view a valuable early working view on the way to create view 1 or 2.</p>

APPENDIX M
CYCLE 2 SURVEY RESULTS

Method 1

Q: How likely are you to utilize this method of layering e³ value models?



Q: Why or why not would you utilize this method?

Not using e³ value models at all right now

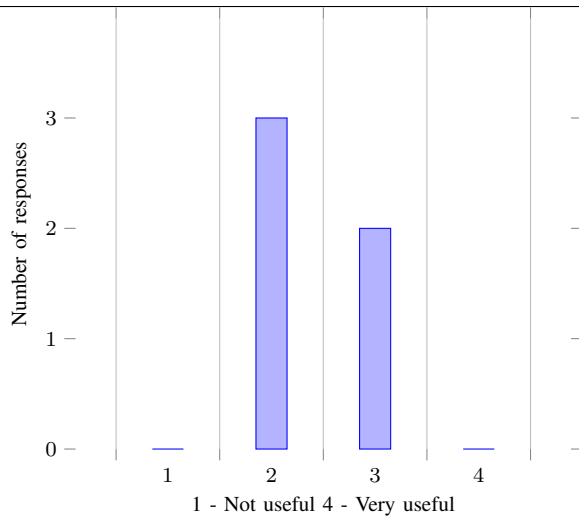
The value of the model comparing to the efforts for creating it could be higher.

The method description is almost missing. It is basically captured in these few statements. Identify API, identify components utilizing the API, the remaining components can be separated into domain and business asset. Reorganize the components according to the layering schema.

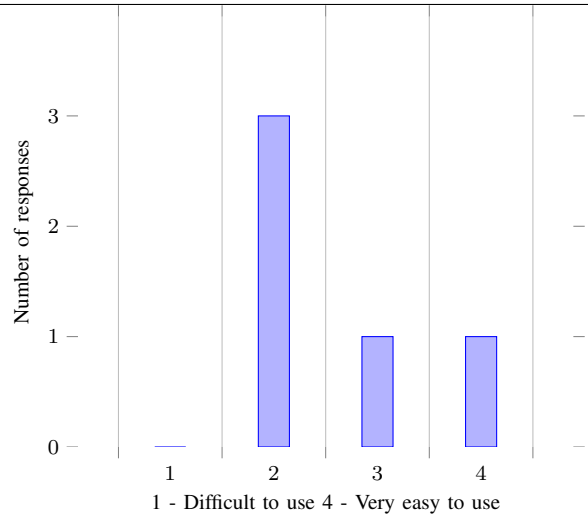
You identify and move at the same time that is harder

If purpose is presentation, I would make a simplified picture in inkscape or yed. If purpose is detailed documentation of an e³ value model, I would want to preserve the original format.

Q: How useful do you think this method is for creating a layered e³ value model?



Q: How would you rate the ease of use of this method?



Q: What would you like to see different?

One tool would be easier. Do I need a specific method to layer the e³ model? Somehow the "many" connections from one block to another could be simplified.

A focus on the actual activities, how to identify which components belong to a certain layer. Identify what is missing and what is superfluous. Hence a description of the process, not the tools used to realize the process. Toolign guide is of course useful, but very limited usage in case other tool is used. Hence the body needs to be in the process description, exemplified with models (and then it is the models that are in focus, not how the models are created using a specific tool)

First focus on type of value then move the boxes

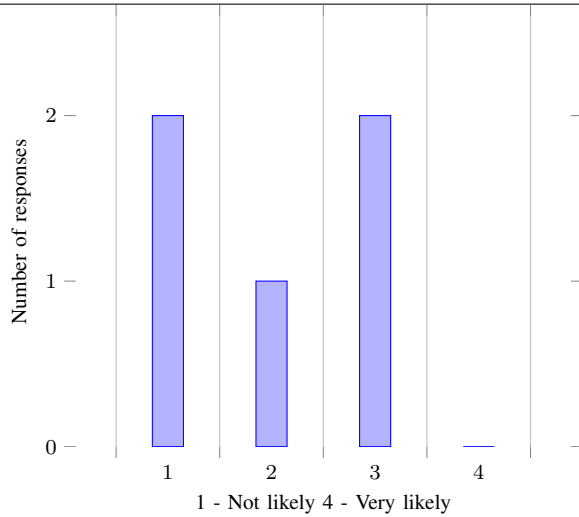
A single tool, preferably something that I am using for other similar tasks. (A plugin in inkscape would do for example.)

Q: Other comments?

The method is depending on a value model with good quality and completeness

Method 2

Q: How likely are you to utilize this method of layering e³value models?



Q: Why or why not would you utilize this method?

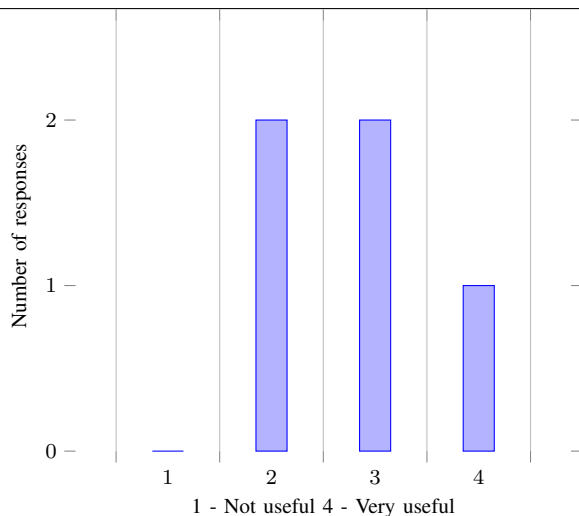
Same answer as for v1

Basically same issue as with on, but in this guide 2 even more empazis is on the tooling, instead of the process.

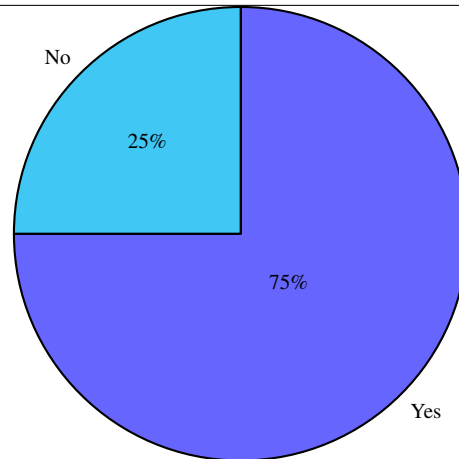
Good method to think on the correct thing

I could consider using the colors step this to highlight something in e³value models for a presentation or summary document.

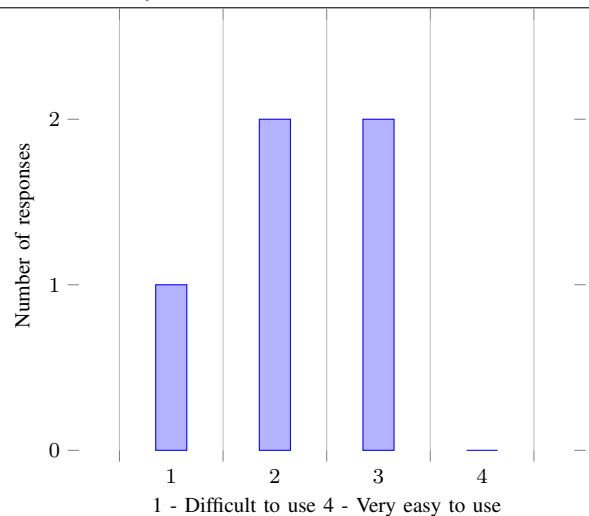
Q: How useful do you think this method is for creating a layered e³value model?



Q: Does the intermediate step with colors make it easier to build the final layered model?



Q: How would you rate the ease of use of this method?



Q: What would you like to see different?

I like the colors, but in this case feels like to much work in the tool for me to try out/put the extra time to do it.

Any kind of grouping is of course valuable during reorganization. However, it is not so muck the focus for teh modelling part where domain and layer highlighting is adding value, but in the resulting model potentially and hopefully read by many. For the person/architect doing the work I would suspect the real work is done on a white board with yellow stickers where it is easy to move the classes (on the stickers) and regroup them as needed.

See comments on method 1.

Q: Other comments?

The method is depending on a value modell with good quality and complete-ness

Too many frames. A general principle I try to follow for graphical presentation is to avoid drawing frames around stuff. It is much more costly for the human brain to understand a grouping by interpreting a border compared to natural concepts such as proximity, color, motion vectors. So part 3 is counterproductive.

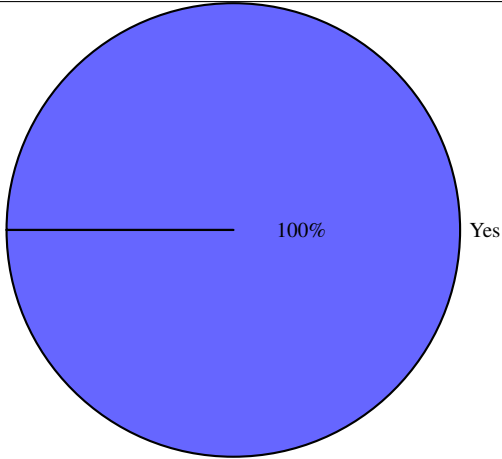
Both methods

Q: Was there anything that made the layering difficult in these methods?

Sounds strange but there were too many mouse clicks.

I find the lack of process description support troublesome, while the tooling description gets in the way of the real value creation.

Q: Would you prefer if there was software tool support to help in building the layered models? (a single tool that could do all required tasks in building a layered e³value model)



APPENDIX N
CYCLE 2 SURVEY ADDITIONAL RESPONSES

For me both descriptions fails to meet the needs, simply because the descriptions focus more on tooling support, and almost totally miss describing the process as such. Tooling support is of course necessary, but very tool specific, while the process description should be universal supporting any tool, including a pen and a whiteboard. Hence tooling issues are better fit for an appendix, while the process description should preferably be tool agnostic.

I think identifying which components belonging to each layer is the important method. How the tool can be used is difficult for me to give feedback on since I have never used it.

APPENDIX O
CYCLE 3 SURVEY RESULTS

Updated method

Q: Does this method make it easier to identify the layers in an e³value model?

Yes

Q: Would you follow this if you tried to apply layering to e³value models?

Yes

Q: What would you change in the method? Other comments?

No suggestion

APPENDIX P
CYCLE 3 FOCUS GROUP INTERVIEW

Q - Interviewers, A - Company representative, S - Thesis supervisor

Q: So we had some questions we would like to ask to sort of evaluate this type of identifying the components. And so, the first question is basically do you believe if there was this method that you would use, that I tried to describe earlier, that it would make it easier to identify the different layers or distribute the components into different layers compared to sort of more ad-hoc style of just looking at the model and trying to identify them?

A: Yeah, I think that it's easier. Yeah, but I'm, I, when I have to show this model to other people I have elected a particular layer and only drawn that. So I don't know how important it is to be able to differentiate between the layers, in the same. It's good to be able to express this as layers, of course, and to have them all in the same model, but when you are to present to someone who is completely uninitiated probably are only concerned about a particular layer. Anyway, that's my view.

Q: And also the next question is, if you had to do this exercise, would you, do you think you would use if you had this as a paper version basically. So would you use these guidelines, or not? I make it sound like the first question. But I guess is, the first one is would it make it easier, and this is would you use it, or not?

A: Unless there is support in the e³value model tool, I probably wouldn't use it. Because I would use that, that model as for documentation purposes to have all the data in one document and I would want to be able to change it and keep it updated and so on. I think, if I understand it correctly you, it's not a reversible thing that you do.

S: Yeah, you can't take it away, yeah, because it's done outside the tool for now. But it could, I mean theoretically it could be done in the tool such as that, you know, see the layers, you don't see the layers. It's just that at the moment the tool doesn't support that. But that's good feedback.

A: If it was supported then it would be definitely [unintelligible].

Q: Yeah. Last one is if you would change anything basically. Anything you would like to see different, or any feedback, other comments?

A: Nothing that I can think of right now.

S: Yeah, but I think your first comment was really good. Some sort of support, not just for layering, but for viewing one layer at a time basically. And for that it's important to be able to distinguish the layers, but also to remove that distinction, sort of to have a dynamic visualization of the layers. So I mean even if you don't implement it, that's a good requirement to have for whatever you finally give.

A: I did a presentation, I only showed the domain layer, and then I [unintelligible] the one that was of interest to the internal components and business assets and stuff like that. So if the tool would support that kind of thing, that would be good.

S: That's good. How did you present it? Did you use the images resulting from the e³value tool or did you use a different representation?

A: No, I drew them in a vector program. In a graphics program. So they didn't really follow the model, they were not correct in that sense, but I stripped out a lot of the information. But it would have been nice to have that information when there were questions about particular things, it would have been nice to be able to show the details.

Q: Thank you, that's all that we have.
