TracyML - A Modeling Language for Social Impacts of Product Life Cycles

Stefanie Betz, Andreas Fritsch, and Andreas Oberweis

Institute of Applied Informatics and Formal Description Methods (AIFB)
Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
{stefanie.betz,andreas.fritsch,oberweis}@kit.edu

Abstract. In order to analyse the impacts of production and consumption on social sustainability, Social Life Cycle Assessment (S-LCA) has emerged as an extensive and promising, yet still young methodology. S-LCA captures the whole life cycle of a product and aims to include all relevant affected stakeholders. It therefore requires large amounts of data and produces likewise complex results, which makes the presentation of findings challenging. We argue that conceptual models make the complexity of S-LCA study results easier to handle. In particular, Domain-Specific Modeling Languages (DSML) support users in structuring and communicating a complex problem domain. This paper presents TracyML, a DSML which provides transparency of social impacts associated with a product's life cycle. Special emphasis has been laid on the graphical notation and understandability of the resulting models. We have evaluated the utility of TracyML by performing interviews with domain experts.

Keywords: Domain-Specific Modeling Language, Product Life Cycle Modeling, Social Life Cycle Assessment

1 Introduction

Sustainable consumption and production is one of the grand challenges of sustainable development. It requires action from all members of society, be it international organizations, governments, businesses or individuals [15]. Sustainability is ultimately concerned with human well-being [33], but is generally understood as consisting of multiple dimensions (e.g. social, environmental, economic) that need to be considered [3]. It is indeed a "wicked problem", a challenge to be addressed, that requires one to consider multiple levels of (long-term) effects and affected stakeholders [3]. So, to improve a product's sustainability, there is a need for tools and methodologies that take a holistic perspective to provide the necessary information for directed action. In practical terms, improved sustainability could be achieved for example by replacing materials with more environmentally friendly alternatives or implementing auditing procedures to improve and monitor working conditions at suppliers. A holistic perspective then

In: C. Cabanillas, S. España, S. Farshidi (eds.):
Proceedings of the ER Forum 2017 and the ER 2017 Demo track,
Valencia, Spain, November 6th-9th, 2017,
published at http://ceur-ws.org

prevents negative environmental or social impacts simply being shifted along the supply chain or from one stakeholder group to another. The Social Life Cycle Assessment (S-LCA) methodology provides such an approach to assess impacts on human stakeholders and their well-being throughout a product's life cycle. Due to the (necessary) extensiveness of this approach, it produces complex results that are not easily processed and hard to compare [4]. It has been argued that visual representations help to convey information more effectively than text [2]. However, this is not true for any arbitrary graphical depiction – a diagram must be constructed in a way that supports understanding [18]. To this end, we have developed TracyML, a Domain-Specific Modeling Language for Social Impacts in Product Life Cycles. The design is based on cognitive and conceptional principles, in order to provide a modeling language that effectively communicates the results of S-LCA studies. For example, the language provides mechanisms to reduce the number of elements in a diagram. This refers to what Moody [22] describes as the principle of "Complexity Management", to not cognitively overload the human mind.

1.1 State of the Art in Social Life Cycle Assessment

S-LCA is an adaption of the Life Cycle Assessment (LCA) technique to the social dimension of sustainability. The rigorous approach of LCA led to its wide adaptation in the field of environmental sustainability [4]. In 2009, the Life Cycle Initiative condensed the latest research on S-LCAs [5] and compiled a guideline document (the guidelines) in order to initiate and inspire a broader use and further development of this still young methodology [4]. In the guidelines, S-LCA is described as "a social impact (and potential impact) assessment technique that aims to assess the social and socio-economic aspects of products and their potential positive and negative impacts along their life cycle" [4, p. 37]. The S-LCA technique was since used in several studies (e.g. [8], [20], [10]).

A generic product life cycle proceeds through the phases *extraction of raw materials*, *production* (of the product and its components), *distribution*, *use* of the product and finally *disposal or recycling* [4]. The guidelines identify workers, local community (the communities in proximity of a site), society (national and global), consumers and value chain actors (such as suppliers and competition) as potentially affected stakeholders of a product system. Each stakeholder category is further divided into subcategories, e.g. "child labor" and "fair salary" (among others) for workers. The methodological sheets for the guidelines (the methodological sheets) [6] list possible indicators per subcategory. Several paths for advancing S-LCA are pointed out in the guidelines: among other points, it is stated that models for the presentation of findings would be helpful to cope with the complexity of study results. In general, a S-LCA study requires large amounts of data and produces likewise complex results [4].

1.2 The Potential of DSMLs for S-LCA

Information Design has been described as "the art and science of preparing information so that it can be used by human beings with efficiency and effectiveness" [17, p. 15]. Its goal is "to present the right information [...] in the most effective and efficient form" [17, p. 16]. Visual modeling languages can be seen as a manifestation of Information Design. They have been developed as a tool to bridge the gap between end users and the implementation of an information system [2]. According to Moody [22], visual languages bear the potential to convey information more effectively (especially to non-technical users) than text. This can be attributed to two characteristics of a visual language: (1) The possibility to spatially arrange graphical and textual elements. (2) Visual representations can be processed in parallel by the human mind (see [21]). Moody's core proposition is that this effectiveness is not an intrinsic feature of visual notations, but must be "designed into them" – creators of modeling languages should therefore no longer consider the design of the visual syntax subordinate to the design of the semantics [22]. To this end, he establishes "principles" for designing effective visual notations, which have been integrated in the methodical approach that we used for TracyML (see Sect. 2).

Domain-Specific Modeling Languages (DSMLs) – in distinction to general purpose modeling languages such as ERM or UML – are modeling languages designed for specific purposes [11]. According to Frank [14], they have gained popularity over the recent years for several reasons: (1) No restriction to a basic vocabulary; DSMLs provide the possibility to describe large models with domain-level concepts. (2) They may improve model quality and integrity. (3) There is freedom to design a specialized graphical notation that helps to improve clearness and comprehensibility. (4) Specific concepts may support users with structuring a domain for a certain purpose. For the purpose of this work, the main reason for designing a DSML is that it gives a certain freedom in graphical notation that allows for appealing and effective visualizations,[1] while still avoiding the pitfalls of an arbitrary graphical depiction (e.g. inconsistency). Furthermore, a well defined syntax and semantic allows for a straightforward implementation of the modeling language into a software system.

2 Applied Research Methodology

TracyML, as a design artifact, is intended to provide utility [25] for S-LCA researchers as well as for practitioners in the field of sustainable production and consumption. NGOs and firms that strive to improve product sustainability can use it as a tool to provide transparency on the product's social impacts.

When designing the language we applied a method proposed by Frank [14] to ensure research rigor. The main design phases are: (1) *Clarification of Scope and Purpose* (2) *Analysis of Generic Requirements* (3) *Analysis of Specific Requirements* (4) *Language Specification* (5) *Design of Graphical Notation* (6) *Development of Modelling Tool* (7) *Evaluation and Refinement.* In this paper, we will

---

[1] See [22] for a critique of common GPML notation.

present the scope, purpose and the identified specific requirements for the language in Sect. 3 (Requirements for TracyML) and the final language specification and graphical notation in Sect. 4 (Language Specification). To ensure practical utility, we have collaborated with two NGOs that work on improving the social sustainability of information and communication technology (ICT) hardware products. They provided the necessary data and use-case (see Sect. 5 for a brief description) for the definition of the language requirements and final evaluation. Section 6 presents the results of our evaluation and a discussion of the language. Finally we will relate our work to other research approaches in the field of DSML and Sustainability (Sect. 7) and conclude with an outlook (Sect. 8).

3 Requirements for TracyML

The objective for TracyML is to facilitate the transparency [2] of social impacts associated with product life cycles. This to done by enabling the modeler to create detailed models of social sustainability characteristics of the life cycle of a product. Emphasis is laid on the ease of use and comprehensibility for non-technical users (be it in the role of a modeler or viewer of a diagram). Thus, we give special attention to the graphical notation [22]. The domain and context of the prospective usage of TracyML is the documentation and communication of the results of a product's S-LCA. Intended users of the language are stakeholders of firms who want to understand, manage and communicate the sustainability of their products. In the role of a modeler this could be a supply chain manager or customer relationship manager. In the role of a viewer, the intended user group would be any possible customer of the product. The diagrams could also be used to communicate with other stakeholders such as governments, NGOs or suppliers.

After defining the scope and the desirable properties, a list of requirements has been developed. Frank [13] proposes a set of formal and pragmatic requirements for DSMLs. While formal requirements contribute to the correctness and completeness of a DSML, pragmatic requirements refer to the perspective of a prospective user (simplicity, comprehensibility and convenience of use) and the modeled domain. The discussion of these generic DSML requirements for TracyML is available under `www.gotracy.org/workingpapers/tracyml_specification.pdf`. Within the scope of this paper, we present the specific requirements for TracyML, that were defined and refined based on reflection about the challenges inherent to the domain of S-LCA. This involved creating preliminary diagrams using the use case data (see Sect. 5). Additionally, basic S-LCA literature like the aforementioned guidelines [4] was taken into account. The following list also explains the rationale for the defined specific requirements:

R1 *The language should provide generic concepts to capture all production steps* from the extraction of raw materials to the final assembly of a product. While the focus is on ICT hardware products, the modeling concepts should be at a generic level that allows for modeling of arbitrary tangible products.

---

[2] By "transparency" we mean the understandable disclosure of information.

R2 *The language should provide mechanisms to deal with data gaps.* Data availability and quality is one of the current challenges for S-LCA [4]. For example, even a large company like Apple struggles with providing transparency about their complete supply chain [16]. The use case supply chain data, while being extensive, also showed some gaps, like unknown production sites of a component or missing supply chain branches.

R3 An organization might undertake actions to alleviate negative social impacts. *The language should provide concepts for reporting on these actions.* The integration into the modeling language should *account for some flexibility*, as these actions might differ significantly for different use cases.

R4 Since the model can get very large, it would be useful *to be able to summarize parts of it.* This less detailed view should still give insight into the sustainability characteristics of the hidden components.

R5 *When aggregating sustainability information, it must still be clear which aspects of social sustainability are considered.* The understanding of social sustainability might differ depending on cultural or political background - to counteract a possible bias, the S-LCA guidelines are based on international agreements [4]. But still, to avoid misinterpretation, it must be clearly stated which aspects are part of the aggregation.

R6 The integration of different sustainability aspects is initially restricted to the stakeholder group 'workers', as proposed by the S-LCA methodological sheets. In parallel to the reasoning for **R5**, *the integration of these aspects should be flexible enough to be altered or extended at a later stage.*

R7 *The language should provide a way to include information on production sites*, i.e. where a component or material was extracted/produced. The assessment of social impacts is usually location-based. This could be concrete production sites (if known) or areas (countries, regions) for a generic assessment [4].

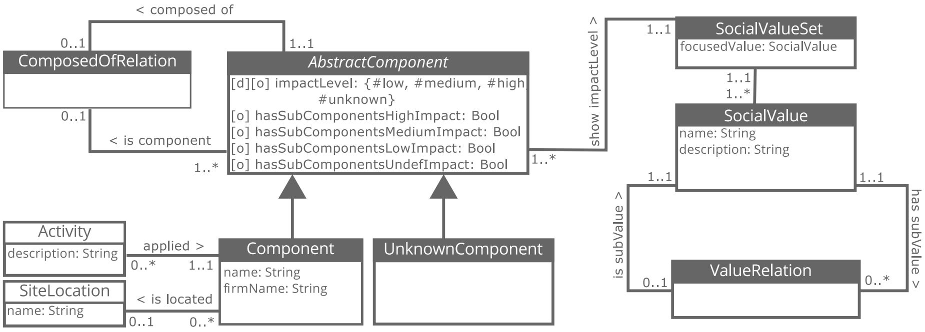
Overall, TracyML is limited to social sustainability aspects. Additionally, the initial design is currently restricted to the stakeholder group 'workers' and the life cycle phases 'extraction of raw materials' and 'production' (see R1). Extensions of the language to further aspects of social sustainability and stakeholder groups or even further dimensions of sustainability are conceivable for later evolution stages of TracyML. In the following, we discuss the language specification.

4 Language Specification

The syntax of TracyML is specified in a meta model. In general, one distinguishes meta type level (M2), type level (M1) and instance level (M0) [12]. A meta model is then on level M2 and specifies a modeling language, while a model is defined on level M1. In the context of enterprise modeling, a meta model could specify concepts like "Process" or "Organisational Unit". The model, on level M1, would then specify, say, a "sales process" carried out by a "sales department". A concrete sale that happened on a specific time is then on instance level M0. This strict separation between the abstraction levels is not always appropriate [12]. It is possible that in some cases an element of a model appears as concrete instance on level M0, but at the same time this concrete instance is also depicted in the model (on level M1). Think about a transportation network model (at the M1 level) that needs to include concrete cities. These cities would be located on instance level M0 [11]. This mapping of different abstraction levels in the model (M1) has been discussed in detail by Frank [12]. He states that this mapping on the different level is not always appropriate and depends on the use cases. This is true for our use case. Regarding the domain and use case targeted with TracyML, there is a also a need to model instances: locations (i.e. production sites in countries or regions, see **R7**) and concrete activities (**R3**). As discussed above, countries are a typical use case for instance-level concepts. Concerning activities, the intention is to provide the possibility for organizations to report on concrete activities, like an audit, that were performed at a specific production site. Thus, a model would also include instances of activities – possibly combined with an additional description of this activity (e.g. date, time, results of an audit) in an information system.

The meta modeling language MEMO MML provides features to account for such conceptional difficulties (see [12]) and is therefore utilized to specify the meta model for TracyML. The meaning of the relevant concepts of MEMO MML is explained alongside the language concepts in the following section.

4.1 Language Meta Model



C1: no cyclic ComposedOfRelation for *AbstractComponent*
C2: no cyclic ValueRelation for SocialValue
C3: impactLevel of UnknownComponent and of any Component that has no SiteLocation is always 'unknown'

Fig. 1: Meta model for TracyML.

The essential concept in the meta model (Fig. 1) is `Component`. A `Component` represents a tangible product or a constituent part or material that is used in the production of it. The concept `UnknownComponent` is introduced to account for the fact that a component might consist of further, unknown components (see **R2**). `AbstractComponent` is then an abstraction of unknown components and regular components. It has no notational correspondence and is denoted as abstract by the *cursive* font in the meta model. The arrows connecting `Component` and `UnknownComponent` with `AbstractComponent` represent the inheritance relationship. Note that the `Component` concept is semantically overloaded on purpose to focus on the social sustainability characteristics rather than details of the production process. Therefore, TRACYML does not distinguish between different kinds of materials like auxiliary materials or operating materials (i.e. material that is necessary for the production but not constituent part of the final product). Whether or not to model such materials in a TRACYML model is for the prospective modeler to decide, but should be made clear to the viewer. This is part of the decision where to stop the underlying assessment.

Each `AbstractComponent` may be composed of none or many subcomponents (`ComposedOfRelation`). In a diagram, each component shows an `impact Level` concerning one focused `SocialValue`, which is dependent on the location of the production site. This location information can be *derived* ([d] in the meta model) from the modeled component via its connected `SiteLocation`. Based on this information, one can *obtain* ([o]) the `impactLevel` from an external data source, e.g. a database that contains assessment data. `SiteLocation` is on type level - therefore represented with white background. This means that in a TRACYML diagram, always a concrete instance of a location (e.g. “Germany” or “Philippines”) is modelled (see 5 for the rationale). Additionally, cyclic “composed of” relationships of `AbstractComponents` are not allowed (Constraint C1).

The `0..1` multiplicity for `SiteLocation` accounts for the fact that the location of a production site might be unknown. For an `UnknownComponent` and for `Components` with no connected `SiteLocation` the `impactLevel` is always unknown (C3). The attributes `hasSubComponentsHighImpact`, `-MediumImpact` etc. are true whenever any subcomponent shows a high, medium, low or unknown impact level. These attributes are flags for a variant of the `Component` symbol, that summarizes all subcomponents – it is described in the following section. Per known component one can specify `Activities` that are performed by an organization in order to address social impacts.

The `SocialValue` and `ValueRelation` concepts set a model into the larger context of social sustainability. A social value would be for example the absence of child labour or non-excessive working hours. These social values can be grouped together thematically e.g. to capture all values concerned with the stakeholder group workers [4]. An aggregation over these thematically grouped social values would then be a social value in itself, which is expressed by the `ValueRelation` concept. The social dimension of sustainability can also be seen as consisting of several social values [26]. As for the `AbstractComponents`, there should not be cyclic relationships between the `SocialValues` (C2).

Finally, any TRACYML diagram is put into the context of one set of social values (here, the social values concerned with working conditions). Each diagram shows the impact levels of each component concerning only one `focusedValue`. This is due to the fact that the user should not be overloaded with information [22]. A software tool could allow a user to switch between the different social values.

Dialect: TracyML *strict*: In the design of TracyML, one can identify a conflict between the desirable properties of a DSML “completeness & correctness” and “ease of use” (see [13]): a diagram that does not explicitly model information gaps (see **R2**) is prone to misunderstanding. A viewer of a diagram cannot see whether a diagram shows all relevant known components down to the initial raw materials. To guide the modeler in creating more complete diagrams, we set up the following optional constraint: Only `UnknownComponents` or `Components` that are raw materials can be “leaves” of the modeled “graph”.

This requires introducing the concept `RawMaterial`, and it adds another layer of complexity that makes it harder for a novice user to pick up the modeling language. Thus, we developed the optional dialect TracyML *strict* (e.g experienced users might opt for the dialect) to account for this conflict. The dialect does further not allow subsuming `Components`. As we want to keep the dialect as compatible as possible with the basic TracyML language, we did not define a new symbol for `RawMaterial`: TracyML *strict* provides a spatial area `RawMaterialArea` – any `Component` that is arranged within this area changes its semantic to `RawMaterial`. The modeled graph is then still valid independent from the used dialect, which would not be the case when `RawMaterials` were represented by their own symbol. This way, a modeler can switch freely between both dialects, as long as the subsumed variant of `Component` is not used. A corresponding modeling tool could account for this and automatically “retract” or “extend” subsumed components when switching dialects. See Fig. 5 for an example diagram that uses TracyML *strict*.

4.2 Graphical Notation

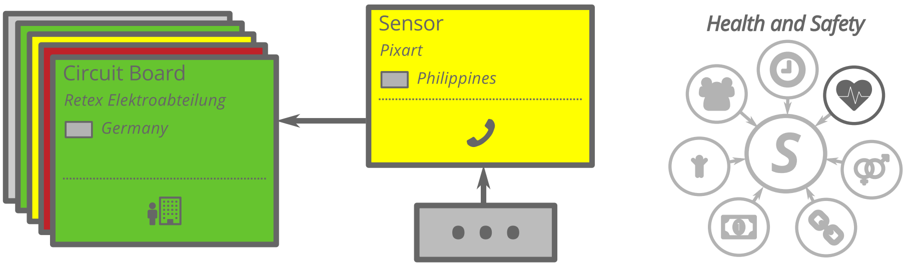


Fig. 2: Simplified exemplary TracyML diagram.

Figure 2 presents a simple exemplary diagram to explain the use of the notation. It depicts a component ’sensor’ that is a subcomponent of a ’circuit board’. The circuit board shows the variant of the `Component` symbol depicting a stack to denote that it has further subcomponents that are not in the focus of the example diagram. Further subcomponents that are used in the production of the sensor are not known, which is represented by the `UnknownComponent` symbol. TracyML uses color to convey information concerning the severeness of social impacts. In this diagram, the impacts concerning the `SocialValue` health and safety are disclosed. The circuit board is assembled in Germany where the potential impact (e.g. work related injury) is relatively low (green color). The production of the sensor in the Philippines shows a medium impact level (yellow color) for health and safety. Further diagrams for other social values are available, as shown with the grayed out circles. There exists also an aggregated social assessment for these values, which is shown with the bigger S-circle in the middle. Finally, one can see that the production site of the circuit board was visited to assess the working conditions. The firm producing the sensors was so far only contacted (phone icon) in order to prepare further steps to improve working conditions.

Figure 3 shows the `Activity` icons that we propose to be used with the `Component` symbols. They have been inspired by the use case in the running example: in order to improve social sustainability, the organization tries to get in direct contact with lower tier suppliers as a first step (a). Some suppliers' sites could also be visited to assess working conditions (b). Another exemplary `Activity` is negotiating for better working conditions (c). The proposed `Activities` and corresponding icons are to be seen as examples, rather than strict specifications, as the activities might change for different products and supply chains. Therefore, further `Activities` to address other use cases can be added relatively easily by defining new corresponding icons.

(a) Contact (b) Visit (c) Negotiation

Fig. 3: TracyML `Activity` icons.

The semantics of different `SocialValues` are expressed by icons to help understanding the diagram. Figure 4 lists the icons that we propose to express the social values concerning the stakeholder group workers. The structure of TracyML allows for easy extension to further aspects of social sustainability by defining corresponding `SocialValue` icons at a later stage.[3]

5 Example Application

Nager IT (`www.nager-it.de`) is a German start-up that works on one product – a computer mouse – to continually improve its design and supply chain in terms

---

[3] The icons for TracyML `Activities` and `SocialValues` were selected from the font toolkit Font Awesome (`http://fontawesome.io`). Additionally, we evaluated the understandability of the `SocialValue` icons. The results are available under `www.gotracy.org/workingpapers/tracyml_icon_evaluation.pdf`.

(a) Freedom of Assoc. (b) Child Labour (c) Fair Salary (d) Working Hours

(e) Forced Labour (f) Equal Opp. (g) Health and Safety (h) Social Security

Fig. 4: TracyML `SocialValue` icons.

of social and ecological sustainability. Transparency concerning the current state of the project is a priority.[4] Being a relatively simple ICT product in terms of the number of components (still, the known part of the supply chain encompasses around 100 components or raw materials), it is a good starting point to address the complex challenge of social sustainability. The firm provided an illustration of the supply chain that served as inspiration for a preliminary diagram during the design of TracyML and as data source for an exemplary S-LCA study.

In general, data sources for S-LCA studies can be governmental and non-governmental organizations, corporate websites, sustainability reports or literature and internet research [8]. We performed a generic S-LCA study for the computer mouse utilizing regional data, rather than site-specific data. This has the advantage that a generic approach is potentially applicable to any product rather than only the use case product presented above. For each social value one or two indicators were defined, for example *Incidence of long working hours* based on the ILO study “Working time around the world” [19]. Finally, the data was classified via a scoring system to identify low, medium and high impacts. A detailed description and discussion for each of the indicators and the scoring system is available in a report under `https://gotracy.org/workingpapers/slca_study.pdf`.

Available supply chain data and the results of the S-LCA study were then used to create informative TracyML diagrams. Figure 5 shows the resulting diagram for the solder wire that is used in the production of the computer mouse. The focused social value is *Health and Safety*. This diagram also exemplifies the use of the *strict* dialect of TracyML as it shows an area that denotes all contained `Components` as raw materials, that are either extracted, or in the case of residual solder, industry waste and e-waste taken back and recycled by the corresponding firm. One can see that the origin of some raw materials like copper or mineral oil is not known. The recycling or production of residual solder, secondary tin and wax takes place in Germany and bears a relatively low negative impact on health and safety. The same is true for secondary tin, which is produced in Belgium. The production of kolophonium shows medium

---

[4] See `www.nager-it.de/projekt`.

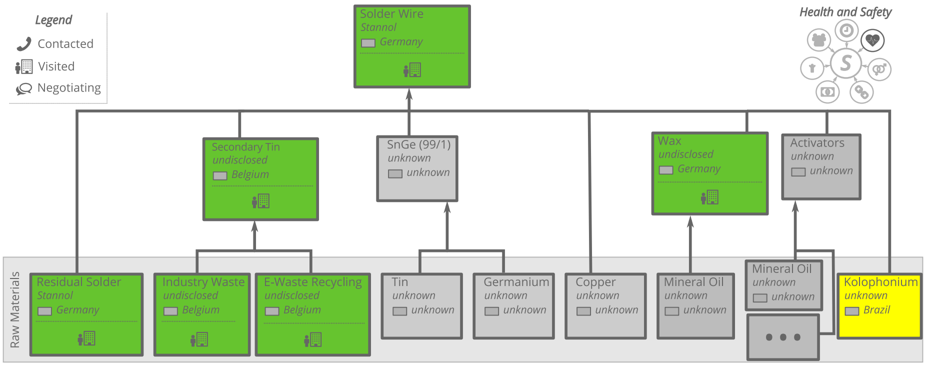


Fig. 5: TracyML diagram for the solder wire that used for the computer mouse.

impact according to the S-LCA study.[5] One can further see that the production of activators uses mineral oil, but also other raw materials that are not known.

6 Evaluation and Discussion

We have conducted an evaluation of TracyML using semi-structured interviews with experts in the field of sustainable production in order to assess and evaluate the results of our work [30]. We interviewed two active members of Fairlötet, a NGO that aims to reduce the negative social and environmental impact of solder wire production and one member of Nager IT, who also provided the use case as mentioned above.

With regard to the understandability, the interviewees appreciated the resulting diagrams as a way to quickly identify "social hotspots' and that the diagrams provides a good overview. Additionally, the consistency of the diagram is to be seen as a plus especially with regard to future extensions and implementations of the DSML. Moreover, the interviewees were in agreement with the different social values currently represented in TracyML and the way they are expressed by icons to help understanding the diagram. With regard to the easy-to-use focus the interviewees stated that they very much value the simplicity of the language (e.g. everything can be added through the one concept `Component`). Additionally, one of the NGOs is eager to use the proposed DSML once it is properly implemented in a software system. Two of the interviewees pointed out that it might be a problem that the assessment of the impact shown in the highest listed component could be seen as an aggregation and as such as an overall assessment. Also, the efficient and understandable layout of large models has been discussed as an issue. One way we plan to address this, is the implementation of TracyML in a software tool that allows for dynamic contraction and expansion of `Component`s (see the discussion of the notational variant of `Component` in Sect. 4).

---

[5] Considering that the underlying assessment is generic, i.e. only on country-level, one could also talk of "potential impacts"

7 Related Work

We did not find any comparable DSML designs that are directly concerned with social sustainability and product life cycles. But one can see DSMLs for modeling business risks [31], business performance [32], or goals [7] as related to TracyML. These DSMLs were also developed using a similar design approach.

Some modeling approaches concerned with sustainability can be found in the area of (business) process languages. These approaches can be roughly[6] separated into approaches concentrating on methodologies and frameworks to integrate sustainability into business process lifecycles (e.g. [29], [27], [24]), improving process sustainability using patterns (e.g. [23]), enabling and applying sustainability performance measurement on processes (e.g. [9], [1]), and finally designing and developing process languages for sustainability (e.g. [28]). Overall we have realized that social sustainability is an often-overlooked dimension. To the best of our knowledge in the existing approaches we found only two of them with a focus on social sustainability (see [9], [1]). The work [9] introduces a Capability Maturity Model including measures. But the proposed model only provides suggestions and misses underlying KPIs. Additionally, this approach is not presenting any process modeling languages. In [1] the authors apply techniques of sustainable performance measurement on a use case. The researchers used structured interviews and document review to determine the sustainability performance measures used by the airline. Here again, the focus is not on modeling. In [28] the authors develop an extended BPMN to measure the carbon footprint of an individual process and apply the extended notation to a case study. The extension is focusing only on environmental sustainability with a special focus on energy consumption. To summarize, none of the presented approaches is focusing on modeling a product's social sustainability. Moreover, none of the presented approaches provides a generic S-LCA serving as a data basis for modeling sustainability.

8 Conclusion and Outlook

In this paper we have presented TracyML, a DSML to provide transparency of social impacts associated with a products life-cycle. TracyML allows modeling the results of a S-LCA assessment. Thus, it provides an approach for visualizing a product's sustainability. By including the concept `Activity`, a firm is further enabled to communicate the actions it performs in order to address possible negative social impacts. We have demonstrated its applicability in a use case and we have evaluated the approach using an expert evaluation. We are aware that it can be a problem to gather all the data needed to model the social sustainability. Not only because of data being not available but also because the data may not be in the correct or needed format. To reduce this problem we included the element `UnknownComponent` as a placeholder when data is not available.

---

[6] Some of the analyzed work are overlapping approaches.

For future work it is still an open question, whether TRACYML can and should be expanded to further life cycle stages and stakeholder groups. While the inclusion of further stakeholder groups would easily fit into the conceptual frame, it is not clear how one could address for example the recycling and disposal phase. Maybe the way to improve sustainability here is rather a question of how to ensure and control final socially and environmentally sustainable recycling. Thus, we are envisioning the development of interfaces between different systems that are specialized on certain aspects of sustainability, in order to cover the full life cycle of a product.

References

1. Alemayehu, W., vom Brocke, J.: Sustainability Performance Measurement – The Case of Ethiopian Airlines. In: BPM 2010 Workshops, LNBIP, vol. 66, pp. 467–478. Springer (2011)
2. Avison, D., Fitzgerald, G.: Information systems development: methodologies, techniques and tools. McGraw Hill (2003)
3. Becker, C., Chitchyan, R., Duboc, L., Easterbrook, S., Mahaux, M., Penzenstadler, B., Rodriguez-Navas, G., Salinesi, C., Seyff, N., Venters, C., Calero, C., Kocak, S.A., Betz, S.: The karlskrona manifesto for sustainability design. arXiv preprint arXiv:1410.6968 (2014)
4. Benoît, C., Mazijn, B. (eds.): Guidelines for Social Life Cycle Assessment of Products. United Nations Environment Programme (2009)
5. Benoît, C., Norris, G.A., Valdivia, S., Ciroth, A., Moberg, A., Bos, U., Prakash, S., Ugaya, C., Beck, T.: The guidelines for social life cycle assessment of products: just in time! Int. J. Life Cycle Assess. 15, 156–163 (2010)
6. Benoît Norris, C., Traverso, M., Valdivia, S., Vickery-Niederman, G., Franze, J., Azuero, L., Ciroth, A., Mazijn, B., Aulisio, D.: The methodological sheets for sub-categories in social life cycle assessment (S-LCA). UNEP/SETAC (2013)
7. Bock, A., Frank, U.: MEMO GoalML: A context-enriched modeling language to support reflective organizational goal planning and decision processes. In: ER 2016, LNCS, vol. 9974, pp. 515–529. Springer (2016)
8. Ciroth, A., Franze, J.: LCA of an Ecolabeled Notebook. GreenDelta, Berlin (2011)
9. Cleven, A., Winter, R., Wortmann, F.: Managing process performance to enable corporate sustainability: A capability maturity model. In: vom Brocke, J., Seidel, S., Recker, J. (eds.) Green Business Process Management, pp. 111–129. Springer, Berlin (2012)
10. Ekener-Petersen, E., Finnveden, G.: Potential hotspots identified by social LCA – part 1: a case study of a laptop computer. Int. J. Life Cycle Assess. 18, 127–143 (2012)
11. Frank, U.: Outline of a Method for Designing Domain-Specific Modelling Languages. Tech. Rep. 42, ICB-Research Report (2010)
12. Frank, U.: The MEMO meta modelling language (MML) and language architecture. 2nd Edition. Tech. Rep. 43, ICB-Research Report (2011)
13. Frank, U.: Some guidelines for the conception of domain-specific modelling languages. In: EMISA. vol. 190, pp. 93–106 (2011)
14. Frank, U.: Domain-Specific Modeling Languages: Requirements Analysis and Design Guidelines. In: Reinhartz-Berger, I., Sturm, A., Clark, T., Cohen, S., Bettin, J. (eds.) Domain Engineering, pp. 133–157. Springer, Berlin, Heidelberg (2013)

15. General Assembly: Transforming our world: the 2030 Agenda for Sustainable Development. available from undocs.org/A/RES/70/1 (2015)
16. Habert, T.: Ethics and the Supply Chain. SAGE Business Researcher (2016), available from http://businessresearcher.sagepub.com/sbr-1775-99621-2728048
17. Horn, R.E.: Information Design: Emergence of a New Profession. In: Jacobson, R. (ed.) Information Design, pp. 15–33. MIT Press, Cambridge, London (2000)
18. Larkin, J.H., Simon, H.A.: Why a diagram is (sometimes) worth ten thousand words. Cognitive science 11(1), 65–100 (1987)
19. Lee, S., McCann, D., Messenger, J.C.: Working Time Around the World. International Labour Office, Geneva (2007)
20. Lehmann, A., Russi, D., Bala, A., Finkbeiner, M., i Palmer, P.F.: Integration of Social Aspects in Decision Support, Based on Life Cycle Thinking. Sustainability 3, 562–577 (2011)
21. Mayer, R.E., Moreno, R.: Nine Ways to Reduce Cognitive Load in Multimedia Learning. Educational Psychologist 38(1), 43–52 (2003)
22. Moody, D.: The Physics of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering. IEEE Trans. Software Eng. 35(6), 756–779 (2009)
23. Nowak, A., Leymann, F., Schleicher, D., Schumm, D., Wagner, S.: Green Business Process Patterns. In: PLoP'11. pp. 6:1–6:10. ACM (2011)
24. Nowak, A., Leymann, F., Schumm, D., Wetzstein, B.: An Architecture and Methodology for a Four-Phased Approach to Green Business Process Reengineering. In: ICT-GLOW 2011, LNCS, vol. 6868, pp. 150–164. Springer (2011)
25. Österle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., Loos, P., Mertens, P., Oberweis, A., Sinz, E.J.: Memorandum on design-oriented information systems research. European Journal of Information Systems 20, 7–10 (2011)
26. Penzenstadler, B., Femmer, H.: A generic model for sustainability with process- and product-specific instances. In: GIBSE'13. pp. 3–8. ACM (2013)
27. Pernici, B., Ardagna, D., Cappiello, C.: Business Process Design: Towards Service-Based Green Information Systems. In: E-Government Ict Professionalism and Competences Service Science, IFIP, vol. 280, pp. 195–203. Springer (2008)
28. Recker, J., Rosemann, M., Hjalmarsson, A., Lind, M.: Modeling and Analyzing the Carbon Footprint of Business Processes. In: vom Brocke, J., Seidel, S., Recker, J. (eds.) Green Business Process Management, pp. 93–109. Springer, Berlin (2012)
29. Rozman, T., Draghici, A., Riel, A.: Achieving Sustainable Development by Integrating It into the Business Process Management System. In: EuroSPI 2015, CCIS, vol. 543, pp. 247–259. Springer (2015)
30. Seaman, C.B.: Qualitative Methods in Empirical Studies of Software Engineering. IEEE Trans Software Eng 25(4), 557–572 (1999)
31. Strecker, S., Frank, U., Heise, D., Kattenstroth, H.: MetricM: a modeling method in support of the reflective design and use of performance measurement systems. Inf Syst E-Bus Manage 10(2), 241–276 (2012)
32. Strecker, S., Heise, D., Frank, U.: RiskM: A multi-perspective modeling method for IT risk assessment. Inf Syst Front 13(4), 595–611 (2011)
33. World Commission on Environment and Development: Our Common Future. available from undocs.org/A/42/427 (1987)