Response Letter to Reviewers' Comments

Title: *Aligning Data-Aware Declarative Process Models and Event Logs*
Conference: 19th International Conference on Business Process Management (BPM 2021)

We thank the reviewers for the thorough study and the helpful suggestions provided to improve our manuscript. We are happy to observe that the reviewers acknowledge that the paper presents an innovative idea and addresses an interesting and relevant topic. We have diligently worked in addressing all the comments, thus producing a revised version of the work, which tackles all the concerns raised by the reviewers. Due to the space limitations, we provide longer answers in this letter, and more compact descriptions are added to the paper. Within the revised version, for the reviewer's convenience, the (relevant) changed parts are highlighted, using a blue font color. The detailed actions taken to address all the individual points of the reviewers are listed below.

Best regards,

The authors
Giacomo Bergami, Fabrizio Maria Maggi, Andrea Marrella, Marco Montali

META REVIEW

In the paper, the authors study the problem of alignments for data-aware declarative process models. The paper presents a sound conversion to a data-agnostic trace alignment problem, and comes with a proof-of-concept implementation.

The main weakness of the paper is that the authors are in two minds about the paper. This becomes clear in the presentation of the paper. It presents a) a novel conversion of data-aware alignments into data-agnostic alignments, and b) it proposes the use of automated planning techniques. However, the paper does not address the connection between the two parts sufficiently. As the initial problem can be converted to an data-agnostic alignment problem, why are current techniques insufficient for the converted trace alignment problem?

A) We thank the PC chairs and the anonymous reviewers for asking questions allowing us to consolidate our paper. We genuinely appreciated these questions, as they allowed us to better explain key details that were previously neglected due to space constraints. Briefly, this article presents, for the first time, an approach for declarative data-aware alignment with trace repairments with data value imputation and/or modification covering both numerical and string data. In particular, this is done by reducing the data-aware alignment problem into a data-agnostic one.

Despite the resulting data-agnostic alignment via Automated Planning being equivalent to customary cost-based aligners (see **G**), our previous work [8] showed that the former outperforms the latter in such a scenario (in terms of computational performance and scalability) in the presence of models of considerable size, which is the case of this paper. In fact, as a consequence of the reduction of the data-aware alignment problem into a data-agnostic one, the automata-based process models used as input for our approach have several more transitions and states than in traditional alignment problems.

Since in [8] it has been also empirically proved that the existing literature declarative trace alignment techniques were not even able to terminate the alignment task and compute optimal alignments for models of increasing complexity, as we needed to show the feasibility of our approach, we decided to use the planning-based aligner for both our framework outline and the experiment section (see also **D).**

We rephrased several parts of our paper to better motivate the need to employ the planning-based alignment technique. Moreover, we explicitly pointed out that the existing literature solutions to declarative trace alignment are semantically equivalent to the planning-based ones, and therefore their only difference is a matter of computational performance and scalability.

Therefore, an important aspect of the motivation should be a comparison with existing techniques, which is currently missing in the paper.

B) To tackle this concern, we have significantly expanded the Related Work section.

The motivation and justification of the connection between the two parts should be improved to accept the paper at the BPM main conference.

To tackle this concern, we (i) added a new paragraph in the Introduction section that explains the reason of using Automated Planning techniques to solve the data-agnostic alignment problem (see **A**), and (ii) made the notation homogeneous throughout §5, and (iii) added explicit details on how the alignment problem is implemented as a planning problem. We also refer to answers **D)** and **J)** for further details.

REVIEWER #1

> *This paper proposes an approach to align data-aware declarative process models and event logs. For this, any activity with a data operation is renamed to an 'atom' in both the model and the log. For example, an activity A that sets some data value to 5 may be replaced by an atom p that signals that A has happened and has set the data value to a value between 4 and 10. These values 4 and 10 are found by checking conditions on the data value.*
>
> *The model is captured by a set of automatons, whereas a trace of the log is captured by an automaton. All automatons use these atoms, and can hence the trace automaton can then be aligned on the model automatons. If all automatons reach a final state, the trace fits the model.*
>
> *What I do not fully understand is the need to use Automated Planning techniques here. It seems that the alignment problem on the automatons is not really different from the well-known alignment problem, using synchronous moves, log moves (inserts in this paper), and model moves (deletes). As such, it seems that the paper introduces two ideas that seem orthogonal: How can we align data-aware declarative models to event logs, and how can Automated Planning techniques be used to find alignments. As such, it feels that this paper lacks a single focus. As a result, if the paper would be accepted, I believe that the authors should make clear why the known alignment techniques do not work here. If these do work, the section on the Automated Planning is very odd. To me, it seems like the paper should be split into two papers.*

D) We thank the reviewer for this observation, as it allows us to better clarify the paper's contribution. Automated planning has been chosen to show that the conceptual framework can be translated into an operational framework by taking existing solid techniques and extending them appropriately. As stated in the introduction, we exploited the SymBA*-2 planner, which was effectively adopted in our previous work for trace alignment tasks [8]. In [8], we showed that such a solution outperformed the state-of-the-art declarative aligners (in terms of computational performance and scalability) in the presence of models of considerable size, which is the case of this paper.

As a matter of fact, in §5.2 we now acknowledge the fact that such aligners could be also used in place of SymBA*-2:

> *Similarly to customary cost-based trace aligners, each of these operations [deletion, insertion, replacement - A.D.] has an associated cost, [...]*

Still, the augmented automata introduced in the present paper have several more transitions, as the atomization steps increase the number of possible labels as well as the overall number of states. In [8], it has been also empirically proved that the existing trace alignment techniques were not even able to terminate the alignment task and compute optimal alignments for models of increasing complexity. Therefore, as we needed to show the feasibility of our approach, we decided to use the planning-based aligner for both our framework outline and the experiment section.

As stated in the experiment section, SymBA*-2's heuristics efficiently handle the aforementioned increase in complexity with acceptable performance in case of a reasonably large number of data-aware constraints. We added such discussion in §4 as follows:

> *We also want to show that our conceptual framework can be translated into an operational framework by taking existing solid techniques and extending them appropriately. By reducing the data-aware alignment problem to a data-agnostic one, we choose to operationalize it using automated planning techniques, as our previous work [8] already showed that such a strategy*

outperforms customary cost-based trace aligners in terms of computational performance and scalability.

Minor comments:
p.1: "in the context data-aware" => "in the context of data-aware".
p.2: LTL_f appears 'out of the blue'. Please introduce this before.
p.4: As a side note: Odd that there is no 'past' version for G.
p.5: "on the footsteps" => either "in the footsteps" or "on the footpath".
p.6: Here, LTL_f is introduced, which is too late.

As kindly suggested by the reviewer, we explain the LTL_f acronym in the introduction and we provide a reference for it.

p.6: "such that \sigma_i |= t_i": Unclear, what does this mean?

E) It means that any trace σ composed of events σ_i can be expressed as a finite sequence of predicates t_i such that t_i is satisfied by σ_i.

Ex.1 (continued), last line: This seems incorrect. The F clause now also accepts if, say, B.x = 1, whereas the original F clause did not. I guess the or-clause should be in the G clause (and not in this F-clause): G(not(C) ∨ F(B ∧ (0 < B.x <= 3) ∨ B.x > 3))). This matches the decomposition into the three intervals much better.

F) At this step, the only change that is performed is on the data predicates, so the satisfiability is unaltered. On the other hand, this is just a preliminary decomposition step. Please refer to the subsequent parts of Example 1 for the final decomposition and atomization producing Σ as required by the alignments. In such, we also consider the y variable jointly with x.

p.9: [i]t seems replacements are not used to augment the trace (see next page). Why introduce them here?

G) In this section, we introduce the mathematical notation for repair sequences generated by the PDDL. Despite replacements might be seen as a syntactic sugar for deletions immediately followed by insertions, the definition of such a constraints has direct implications on the final outcome of the alignment: as we observed in the paper, replacement operations might be favored instead of insertions or deletions by assigning to the latter an inferior cost. This notation also allows us to gracefully merge the two main contributions of the paper (see **J**).

Def.1: Unclear what is being defined here. Reads more like a theorem. Given a trace t and and a model M, a repair sequence r is a sequence of operations (deletion, insertion, replacement) such that r(t) conforms to M. The repair sequence r is called optimal if there exists no repair sequence r' such that the costs of r' are lower than the costs of r. The conformance checking problem is to find an optimal repair sequence for trace t and model M. (?)

H) We have reformulated the definition of the conformance checking problem that we intend to solve in the paper. By doing this, we motivate the required formalisms that will follow.

Def.1: "it exists" => "there exists".

The typo was fixed according to the reviewer's suggestion.

p.10: A delete operation corresponds to a move on model, whereas an insert operation corresponds to a move on log. This suggests that we now have the standard problem of aligning a trace on a model, and could use the standard alignment techniques. Is this correct? If so, using the planner as suggested is an option, but standard alignment techniques could also be used, right?

Please refer to our previous answers **G)** and **D)**, where we explained how the alignment-based planning technique can be replaced with a customary state-of-the-art cost-based aligner, even if the latter are not always suitable to deal with the high complexity of the problem.

Fig.4: The arc from s_5 to s_6 is labeled p_7. I think this should be p_8.
p.10: No replacements

We acknowledge the typo. We fixed it as suggested.

REVIEWER #2

The paper introduces an alignment-based approach for data-aware Declare models using planning tools. To this purpose, data-aware constraints are converted in data-agnostic automata which can be enriched with repair operations which can lead the planning to suggest optimal alignments.

The topic of (data-ware) Declarative alignments is interesting, and the paper has a clear elaboration. The examples guide the reader through the formalisms which underpin the approach well. The authors have clearly considered the topic to its fullest details.

Thanks for the positive consideration of our research.

My main concern is that within the scope of BPM, the planning part could use with a bit more detail. E.g. Section 3 mentions the use of a STRIPS fragment without any context. [...]

I) We agree with the reviewer. The use of STRIPS is not required to the understanding of the paper and can create confusion. Therefore, we have revised the section discussing the planning part.

[...] Next, Section 5.3 does not link that well with 5.1/5.2, and the predicates are understandable but not intuitive given the prior formalisation. It is a hard balancing act, but this could confuse some non-informed readers, also concerning the suitability of planning. [...]

J) We deeply thank the reviewer for this suggestion. To solve this issue, we extended §5.3 by describing how the planning strategy generates the repair strategy that is going to be exploited in §5.4 to repair the trace. Therefore, we imported in §5.3 the notation for the repair strategy from §5.2 and in §5.3 onwards we changed the sigmas and phis with t-s and e-s, so as to uniform the mathematical notation with the one for PDDL.

[...] Can this be formulated as other optimisation exercises as well? The related work section also does not line up the latest in planning to quickly get a grasp of the field, except for the recent usage of it in the declarative process modelling community.

K) We thank the reviewer for the suggestion. To contextualize the paper in the state-of-the-art literature of Automated Planning, we explicitly specified in the Introduction section that we exploited the SymBA*-2 planner, which is the winner of the sequential optimizing track at the 2014 Int. Planning Competition.

From Section 5.4, it is not completely clear whether the planning optimises just the single trace, or the cost for the whole model as the planning problem focuses on a goal condition which lines up with a (single) constraint automaton. Given that the experimental evaluation deals with full Declare models, it would have been nice to have a better insight into how the overarching output could inform the improvement/repair of the whole model with an example.

M) In the present paper, we are interested in repairing the trace so as to make it conformant to the model, as we assume that models represent the golden standard while the traces might contain the anomalies to be detected. Still, we repair a trace by taking the model as a whole via its automaton representation. To tackle the reviewer's concern, we rephrased the first sentence of §5.4 to clarify our intent.

The paper is very well written, only a few apparent typos were present:
- Abstract: '…in the context data-aware declarative…' : rephrase
- Page 2: can can: repetition
- Page 3: two constraints could be not in conflict: rephrase
- Page 5, very minor: walking on the footsteps: walking in
- Page 6: post mortem – just post analysis? Sounds quite strong

We thank the reviewer for the suggested improvements to the article. We fixed all of the changes: post mortem was rephrased as "[when] traces reach [their] completion".

Overall, the paper has a high degree of maturity and clearly illustrates its strengths and can be considered for presentation.

Thank you.

REVIEWER #3

This very nicely executed paper provides a valuable extension to our knowledge and tools set in the area of conformance checking between process execution traces and declaratively specified process models. More specifically, the paper develops a framework for checking conformance between execution traces and data-aware declarative process models specified in a data-aware variant of DECLARE. Unlike previous work on conformance for data-aware DECLARE, the framework in the current paper includes an implemented algorithmic approach for "repairing" a trace if it does not comply with the process model. The framework relies on (a) using a "cellular decomposition" of the space of continuous data values to reduce things to a propositional context, (b) mapping the propositional version of the DECLARE process model into a constraint automata, (c) constructing a second automata that captures the idea of repair steps (insert, delete, and a form of replace) against a given trace, and (d) building a planning problem based on simultaneous traversal of the 2 automata. The paper also provides some benchmarking experiments, showing the practical utility of the framework on modestly sized traces (up to length 30) and a modest number of constraints (up to 10).

Minor questions:

1) in Section 3.2 you say that your atoms have the form "A.k R c", where c is a constant. Would your framework easily extend to the case where you have atoms of the form "A.k R A'.k' " ? I think that your \Sigma-encoding (what I might call "cellular decomposition") can be extended to create a propositional representation. Would other parts of the construction also carry through? (One sticking point might be for the case of inserting steps and inventing values).

Please refer to answer **P)**, as we merged the concerns by reviewer #3 with the ones by #4. At the present stage of the paper, value invention is carried out when repairing a trace by adding new values or altering existing ones within event payloads.

> *2) in Section 4 you say that each event trace should be associated to just one activity label. But, if I understand your running example you seem to have both B and C as "labels", which I take to be "activity labels". Is there an inconsistency here? Also, if you require that there is just one activity label, then why bother with them? [...]*

N) We reformulated the beginning of Section §4 to make explicit that ``each event'' (so, each event which is part of the trace) should contain just one label. Thus, each trace might be composed of events, each of them having a distinct label.

> *[...] Also, if I wanted to use multiple activity labels then I could simulate that by using a single dummy activity label and using a field in the payload to hold my multiple activity labels, right?*

O) We thank the reviewer for this observation, which is correct. Multiple activity labels are also supported by our previous work on extending the property graph data model for big-data scenarios [5]. We added it as a reference motivating the reviewer's claim.

REVIEWER #4

> *The paper contributes to the concept of alignments used in conformance checking tasks. The proposed approach provides a data-aware alignment strategy for data-aware declarative process models including a trace repair strategy existing approaches usually omit. After the problem is reduced to a data agnostic trace alignment problem the authors describe a solution based on Automated Planning techniques from the AI research field. The approach is evaluated via a proof-of-concept implementation and regarding computation times to sketch the approach's scalability with respect to the size of the model and the noise in the traces.*
>
> *The paper is well-structured and is built upon a clearly formulated problem statement. A mostly informative description of necessary background knowledge is provided, too. The clearly formulated working assumptions are important and answer many questions that might arise in advance, hence, making the approach (more) convincing. The description of the approach is based on a strong and, as far as I can assess this, sound formalization. Additionally, the technique is accompanied with a proof-of-concept implementation that was setup for a performance measurement giving informative insights regarding the dependencies of computation time to influencing factors like trace length, number of constraints and repair costs.*

We are grateful to R4 for the positive reflection about our paper, and the very nice and correct highlight on its importance.

> *However, though I am mainly convinced that the paper is a valuable contribution to the BPM research domain, I also have some points of criticism, which are explained in the following.*
>
> *Criticism:*
> *- Related Work: While the discussion about applicability of alignment-based approaches for conformance checking with procedural models is informative and concise, the discussion of related*

> *approaches for declarative models comes up a bit too short. The authors mention two approaches directly related to the proposed approach that "only" lack repair strategies. It would be interesting to know, if those approaches are not able to provide repair strategies or if the authors of the cited publications simply left this out of scope. In case of the latter, it is questionable to "discard" those approaches without a thorough comparison. If the underlying formalism does not allow for repair strategies instead, this should be stated clearly in section 2.*

We significantly extended the related work section to explain in detail why existing techniques are not suitable to solve trace alignment based on data-aware Declare (see **S**).

> *- Section 3.2: Though I like the concise definition of MP-Declare, when referring to activation and target conditions one might ask for the correlation conditions, too. Is it possible to integrate them in the suggested approach, too, or is this a drawback? If so, this should be clearly mentioned as a limitation in the overall evaluation, since it restricts application scenarios to those that do not require correlation conditions.*

P) This is actually a limitation of our work. However, as witnessed by other works (also in the context of multi-perspective conformance checking based on procedural models), taking into consideration correlations in the trace alignment problem would lead to a state space explosion when searching for the optimal alignment. We provide some possible directions for future work to deal with correlation conditions

> *- Section 4, assumption b): "we restrict the space of the possible alignments of the log trace repairs to the traces generated by the automaton representation of the Declare model" -> It was announced that the assumptions (and restrictions) can be inferred from the literature. However, assumption b) lacks a corresponding reference. Furthermore, to me it is not clear what consequences arise from this restriction. Actually, this whole sentence part is hard to read and I want to encourage the authors to simplify it for reasons of comprehensibility.*

Q) We added the reference to the paper [8] from where we actually drew our assumption. We also simplified the sentence as suggested: the aim is to restrict the possible repairs of a given trace to only the traces that are both conformant to the Declare model and are generated by its associated automaton. Quoting from the revised paper:

> *b) we restrict the space of the possible alignments of the log trace repairs to the traces generated by the automaton representation of the Declare model [8];*

> *- p. 9: "E.g., by assigning a higher cost to insertions and deletions and a lower one to replacements, we will favor replacements when possible." -> I wonder how appropriate costs for the three operations can be determined. Is this arbitrary? I am also a bit concerned about uniqueness. Let us consider, for instance, the response constraint (and let us stick to plain Declare): response(A,B). This means a trace like AAAAC would violate the constraint. In that case it is easy to see that it can be repaired by inserting a B. But what about the trace AC? I could delete A or insert a B. From an algorithmic perspective it must be defined whether the insertion or the deletion is applied. One way would be to choose different costs for the two operations. This should be briefly clarified (probably at the end of section 5.2).*

R) We thank the reviewer for this clarification request, thus allowing us to better explain our solution. As customary in traditional trace alignments [2], preferred repair strategies are always modeled by assigning different costs to insertions, deletions, and replacements (see also **D**). This depends on the domain in which the process is executed and on the specific semantics of the activities that need to be inserted or deleted. For example, in a healthcare process, it could be more costly inserting an activity related to the treatment of an allergy than removing an activity related to the administration of a medicine that provokes that allergy. We expanded our example to better describe the different possible repair solutions as suggested.

> Figure 4 shows the automaton augmented with the repair operations $\mathcal{A}^{+}_{\varphi_{\mathcal{M}}}$ obtained for the model $\mathcal{M}$ from Example \ref{ex:first}. Intuitively, $\mathcal{A}^{+}_{\varphi_{\mathcal{M}}}$ accepts all the string sequences conformant to the model and have been obtained by adding/removing the missing/wrong atoms to/from t_{σ}, where atomic operations are explicitly marked. As required, both augmented automata do not accept $t_{\sigma'} = p_5$ C C. If insertions are associated to the lowest cost, the best alignment strategy adds p_8 at the end on the trace; by explicitly marking such repair with ins_p_8, the augmented automata now accept $\hat{t_{\sigma'}} = p_5$ C C ins_p_8. On the other hand, if replacements are associated to the lowest cost, the best alignment strategy would replace the last \texttt{C} with p_8, thus requiring to first delete \texttt{C} and then insert p_8; the resulting $\hat{t_{\sigma'}} = p_5$ C del_p_8 ins_p_8 is also accepted by both automata.

> *- The main flaw of this (apart from this well-written and sound paper): The paper rather directly jumps into an explanation of how to transform the model-trace alignment problem into a planning problem. I completely miss any discussion of alternative solution strategies and, consequently, advantages and drawbacks of the proposed approach, too. What about transforming the problem into a satisfiability problem and applying optimized SAT solvers instead? It is clear that space is limited but if a new tool is suggested it is essential to comment on alternative solutions and reasons why the strategy chosen is superior.*

S) We thank the reviewers for this observation, that gave us the opportunity of clarifying the novelty of our contribution. In fact, some of the discussion of the Related Work was previously removed due to the lack of space. We re-introduced such comments in the Related Work section. We explain more into detail why existing approaches are not suitable to cover all the aspects that we cover in the present paper.

In addition, we wanted to mention that we are currently working on a separate research thread in which we use SAT solvers to address the conformance checking problem. However, similarly to other state-of-the-art approaches described in the related work section [7], the conformance checking problem addressed with a SAT solver can only provide a numerical characterization of the degree of conformance without building an alignment of a deviant trace. In fact, to do this, we need to solve an optimization problem with respect to a specific cost function. On the other hand, other equally interesting problems can be solved with SAT such as LTL Query Checking and online monitoring of LTL rules.

> *Minor remarks:*
> *- p. 1: "while the former fully enumerate the set of all the possible allowed traces, the latter provide a compact process representation" -> This is a bit oversimplified. The first part is at least language-dependent. In BPMN, for instance, we have a concept of swimlanes but they lack a clear definition of their execution semantics. Hence, the model is in that regard underspecified, which means that the model is not able to enumerate the set of all possible allowed traces. Considering the data perspective, it is even worse. Additionally, if declarative models are compact depends on the processes they represent. For instance, a simple sequence of two activities can be represented in a compact way in BPMN but requires comparatively many constraints. I want to encourage the authors to reformulate this sentence.*

T) We rephrased the sentence by removing any reference to the compactness of declarative models' representation.

> *- p. 5: What are numeric fluents? I'm not an expert in Planning Problems but this seems to be a specific term of the PDDL language and, consequently, should be introduced to improve comprehensibility.*

U) A (numeric) fluent is a variable which applies to zero or more objects and maintains a value throughout the duration of the plan. Numeric fluents can be altered through the effects of both actions: a numeric fluent is used in our scenario for keeping track of the alignment cost. We added this clarification in the text. Due to the lack of space, we simply referred to those as "global numeric variables".

> *At the end of this section: "We also freely assume that missing values are represented with specific values, such as an empty string, -1, or ∞, depending on the type." -> Declaring a set of maybe valid values (an empty string could be meaningful, too), is on the one hand valid but on the other hand questionable since it might lead to misbehavior in practical applications. Hence, it should be mentioned as a limitation of the approach in the overall evaluation.*

V) We brought this notion from a data-cleaning perspective, where missing information is seen as a data error and should be imputed with some real-world values. As our implementation relies on POSets, we can always generalize the argument by assuming that any missing value could be represented by a minimal element being neither a specific float nor a string value. We rephrased our claim as per the anonymous reviewer's suggestion. Please also observe that such extension does not alter the experiments that we already run, as no missing data was generated by the log generator: that might only require a slightly re-configuration of the tool's implementation. We then rephrased our claim as such:

> *As both strings and floating point numbers have non-strict partial orders and are hence posets, we can also freely assume that missing values are represented as an extra minimal element ε being neither a string nor a specific floating point number.*

> *- p. 13: Until this page I have tried to figure out what the notion of a "goal condition" is and which role it plays for the proposed alignment approach. Maybe this could be clarified (probably with a simplified summary) earlier in the paper.*

W) Planning systems are an Artificial Intelligence technology showing how to reach a prefixed goal configuration given an initial world by exploiting a set of actions that change the initial world so as to reach the goal configuration. We rephrased such an explanation at the beginning of §3.3. The sequence of actions resulting from the execution of a PDDL plan provides the repair sequence that is going to be exploited to repair the trace, thus describing how the trace should be possibly transformed to be accepted by the model. In particular, the initial world configuration contains the description of the two augmented automata, thus including their initial and final states. On the other hand, actions are the alignment strategies that are enacted to reach the final goal configuration, i.e., reaching a final state in both automata. As now stated at the end of §5.3:

> *As the goal requires that in both augmented automata an accepting state is reached, the actions will encode the strategies to visit both automata by exploiting their transition functions, while assigning different alignment costs to each of the strategies. When the goal is reached, the resulting action sequence (where **sync**s are stripped) represents the repair sequence ϱ that we are going to exploit in the next section.*

Spelling, grammar, typos, etc.:
- Abstract: “in the context data-aware declarative process models” -> “in the context of data-aware declarative process models”
- p. 2: “After representing the data-aware” -> formatting issue (consider “overfull hbox” warnings in your LaTeX editor)
In the next line: “using a data agnostic LTL…” -> “… data-agnostic”
p. 5: “should be associated to just one activity label” -> “… associated with just one …”

All the remarked typos were fixed as suggested.

Fulfillment of Open Science Principles (incl. transparency, reproducibility, and replicability):
All research artifacts are available via a given link to a Google Drive folder. A Readme and a provided video enable reviewers and researchers to reproduce the results included in the paper in a transparent way.

Thank you.