Introduction to the Key Questions in the Questionnaire

1 Introduction

The questionnaire would ask you to give answers for four questions/phases namely **Define Context**, **Quantify Requirements**, **Resolve Inconsistency**, and **Reason Trade-off**. Please take your time to think and concentrate on the tasks/questions, as we will record the completion time as well. We would also ask you to do so with and without our proposed tool (under different contents), so that we can assess the benefit/usefulness of the tool.

1.1 What do You Need to Do?

We kindly ask you to complete some questions in the following links in order (basically, one task + 1-5 questions each link):

1.1.1 Define Context

https://tnbat3ub4jr.typeform.com/to/U1h9Htft

https://tnbat3ub4jr.typeform.com/to/pcxnjw8F

https://tnbat3ub4jr.typeform.com/to/FZel1H1e

1.1.2 Quantify Requirements

https://tnbat3ub4jr.typeform.com/to/ZmJrFvfk

https://tnbat3ub4jr.typeform.com/to/gjR2o73j

https://tnbat3ub4jr.typeform.com/to/UQKghcrB

1.1.3 Resolve Inconsistency

https://tnbat3ub4jr.typeform.com/to/oGaNPoci

https://tnbat3ub4jr.typeform.com/to/TTml8q0x

https://tnbat3ub4jr.typeform.com/to/qF5YRPTZ

1.1.4 Reason Trade-off

https://tnbat3ub4jr.typeform.com/to/h7JVJPbR

https://tnbat3ub4jr.typeform.com/to/mwP4BzjM

https://tnbat3ub4jr.typeform.com/to/CLGUUaCR

Please ensure that, for all multiple choice questions, you have selected one option rather than simply clicking “OK”.

The detailed example and explanations can be found at the remaining of this document. Please read through them before start doing the questionnaire. Thank you very much for your help!

2 Define Context

Context refers to the particular environment under which the non-functional requirement should be applied. Those contexts are often external factors that cannot be controlled by the software system. When there are multiple requirements, your job is to extract the possible combination of the context

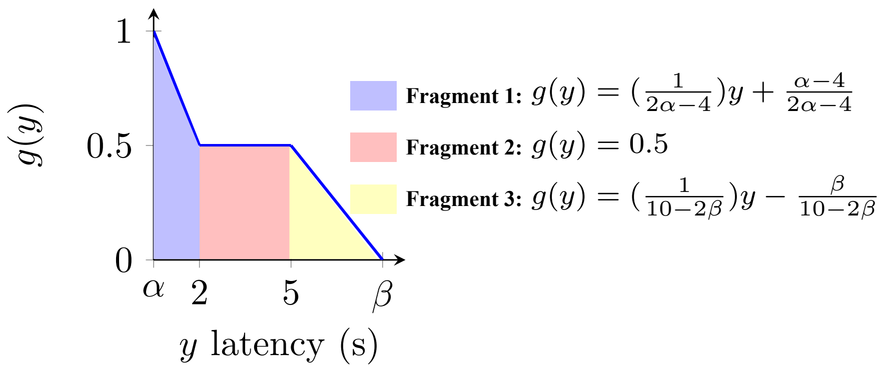


Figure 1: The function of an exampled proposition on *latency* formed by three fragments.

features so that these requirements can be tested therein. They can be connected using logic connector "AND" (for scenarios happen together) and/or "OR" (for mutually exclusive scenarios).

For example, considering the following statements:

Response time: `"The system response time must be no more than 2 seconds under read-only workload."`

Throughput: `"The system should support at least 1,000 users for CPU intensive requests."`

The context would be "`read-only workload`" AND "`CPU intensive requests`", as both of them need to be considered for different performance concerns at the same time.

3 Quantify Requirements

The given requirement may be imprecise in nature. Your job is to quantify the satisfaction score of the possible values of a performance concern with respect to a requirement.

For example, suppose that the requirement "*the latency should be better than 5 seconds and ideally be better than 2 seconds*" can be quantified by three fragments to form its corresponding proposition:

1. `"a smaller latency` y `is more preferred when` $y \in [-\infty, 2)$" &
2. `"any latency` $y \in [2, 5]$ `is equally preferred at` 0.5" &
3. `"a smaller latency` y `is more preferred when` $y \in (5, \infty]$"

The corresponding function has been illustrated in Figure 1 where $g(y)$ is the satisfaction score. As can be seen, since we cannot compute $-\infty$ and ∞, we set the bounds as α or β which are the smallest and greatest values of latency measured during tuning. The $s_i = 0.5$ in *Fragment 2* sets the satisfaction score of joint points for the adjacent fragments at both ends; or otherwise, the corresponding s_{max} and s_{min} in *Fragment 1* or *Fragment 3* should be 1 and 0, respectively. The concatenated function g serves as what is used to measure satisfaction score.

There might be of course different way of quantifying the requirements, you might need to reason about their relationships. Please refer to the **Relationships between Requirements** section.

4 Resolve Inconsistency

Different stakeholders may come up with different requirement for the same performance concern, in that case, your job is to convince them to agree on a single requirement.

For example, two stakeholders might given the following for latency:

Stakeholder 1: `"The system latency must be no more than 3 seconds."`

Stakeholder 2: ``The system latency should ideally be better than 5 seconds and no worse than 2 seconds.''

The problem is: what is their relationship (when being quantified) and which should be chosen? Please refer to the **Relationships between Requirements** section.

5 Reason Trade-off

At some points after some measured performance, one might need to relax one or more requirement. Your job is to determine how exactly a requirement(s) can be refined/relaxed, and what will be the implication to the other requirement on some conflicting performance concerns.

For example with the following requirements:

``The system response time must be no more than 2 seconds under read-only workload.''

``The energy consumption should not be higher than 45W for CPU intensive requests.''

Now suppose that we want to relax response time for the benefit of energy consumption, what are the possible ways and what are their relationships? Please refer to the **Relationships between Requirements** section.

6 Relationships between Requirements

The key for finishing the jobs in the phases of **Quantify Requirements**, **Resolve Inconsistency**, and **Reason Trade-off** is how to reason about their relationships. Often, the non-functional requirement(s) can be refined according to the needs and this is often achieved in two ways: relaxation and tightening. The former means that the requirement(s) is made easier to be satisfied; while the latter implies that the satisfaction would become harder.

With the formal definition of propositions and the corresponding quantification in our tool, we are able to systematically refine a single or a set of propositions.

6.1 Single-Dimensional Refinement

Given two propositions p_i and p_j (together with their functions g_i and g_j) under an interval $[\alpha, \beta]$ for the single non-functional metric, we can define and generalize their refinement relationships within $[\alpha, \beta]$ as:

Single-Dimensional Refinement: p_i can be relaxed to p_j, or p_j can be tightened to p_i, if and only if $\int_{\alpha}^{\beta} g_i(y)dy < \int_{\alpha}^{\beta} g_j(y)dy : y \in [\alpha, \beta]$.

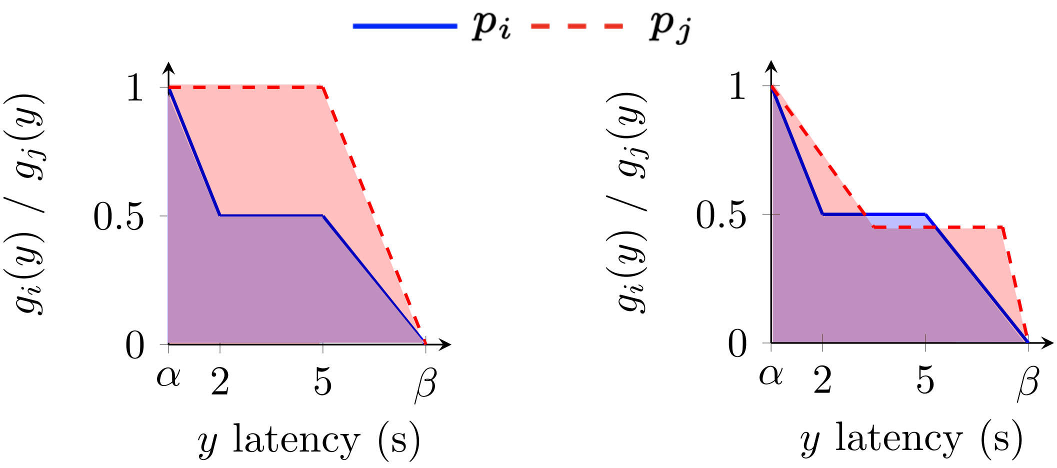
As shown in Figure 2, the idea behind this definition is straightforward: if the area covered by p_j is larger than that of p_i within $[\alpha, \beta]$, then it should be easier to satisfy in general as the overall satisfaction scores are higher over the same interval. The area $\int_{\alpha}^{\beta} g_i(y)dy$ serves as an indicator of the difficulty of satisfaction. Indeed, it is possible that after a refinement the areas that the two propositions cover are identical, in which case we say they are equally satisfiable, denoted as $p_i \equiv p_j$.

Within such a context, we define strong single-dimensional refinement over $[\alpha, \beta]$ as:

Strong Single-Dimensional Refinement: p_i can be strongly relaxed to p_j ($p_i \rightrightarrows p_j$), or p_j can be strongly tightened to p_i ($p_j \leftleftarrows p_i$), if and only if $\int_{\alpha}^{\beta} g_i(y)dy < \int_{\alpha}^{\beta} g_j(y)dy : y \in [\alpha, \beta]$ while $\forall y \in [\alpha, \beta] : g_i(y) \leq g_j(y)$.

Similarly, a single-dimensional refinement may be weak:

Weak Single-Dimensional Refinement: p_i can be weakly relaxed to p_j ($p_i \rightarrow p_j$), or p_j can be weakly tightened to p_i ($p_j \leftarrow p_i$), if and only if $\int_{\alpha}^{\beta} g_i(y)dy < \int_{\alpha}^{\beta} g_j(y)dy : y \in [\alpha, \beta]$ while $\exists y \in [\alpha, \beta] : g_i(y) > g_j(y)$.



(a) Strong refinement $p_i \rightrightarrows p_j$

(b) Weak refinement $p_i \rightarrow p_j$

Figure 2: Refinement of the proposition on *latency*.

As in Figures 2a and 2b where p_i can be relaxed to p_j, it is intuitive to see that in strong single-dimensional refinements, any value of the non-functional metric will have higher (or at least equal) satisfaction score on p_j than that on p_i. In weak single-dimensional refinements, in contrast, there exists at least one non-functional metric value that is even more satisfied on p_i than on p_j, yet p_j leads to an overall higher satisfaction score (i.e., larger area).

6.2 Multi-Dimensional Refinement

Since there is one proposition per metric when multiple dimensions of non-functional metrics are involved, we are now dealing with a vector of propositions: $\overline{\boldsymbol{p}} = \{p_1, p_2, ..., p_n\}$; the corresponding functions also lead to a vector: $\overline{\boldsymbol{g}} = \{g_1, g_2, ..., g_n\}$. As a result, for a given vector of measured values over all non-functional metrics $\overline{\boldsymbol{y}} = \{y_1, y_2, ..., y_n\}$, we have $\overline{\boldsymbol{g}}(\overline{\boldsymbol{y}}) = \{g_1(y_1), g_2(y_2), ..., g_n(y_n)\}$. This entails a typical case of multi-objective comparison during the refinement: the satisfaction scores of two configurations under different proposition vectors $\overline{\boldsymbol{g}}_{\boldsymbol{i}}(\overline{\boldsymbol{y}}_{\boldsymbol{a}})$ and $\overline{\boldsymbol{g}}_{\boldsymbol{j}}(\overline{\boldsymbol{y}}_{\boldsymbol{b}})$ could be:

- $\overline{\boldsymbol{g}}_{\boldsymbol{i}}(\overline{\boldsymbol{y}}_{\boldsymbol{a}})$ dominates $\overline{\boldsymbol{g}}_{\boldsymbol{j}}(\overline{\boldsymbol{y}}_{\boldsymbol{b}})$, i.e., $\overline{\boldsymbol{g}}_{\boldsymbol{i}}(\overline{\boldsymbol{y}}_{\boldsymbol{a}}) \prec \overline{\boldsymbol{g}}_{\boldsymbol{j}}(\overline{\boldsymbol{y}}_{\boldsymbol{b}})$;
- $\overline{\boldsymbol{g}}_{\boldsymbol{j}}(\overline{\boldsymbol{y}}_{\boldsymbol{b}})$ dominates $\overline{\boldsymbol{g}}_{\boldsymbol{i}}(\overline{\boldsymbol{y}}_{\boldsymbol{a}})$, i.e., $\overline{\boldsymbol{g}}_{\boldsymbol{i}}(\overline{\boldsymbol{y}}_{\boldsymbol{a}}) \succ \overline{\boldsymbol{g}}_{\boldsymbol{j}}(\overline{\boldsymbol{y}}_{\boldsymbol{b}})$;
- or incomparable, i.e., $\overline{\boldsymbol{g}}_{\boldsymbol{i}}(\overline{\boldsymbol{y}}_{\boldsymbol{a}})$ and $\overline{\boldsymbol{g}}_{\boldsymbol{j}}(\overline{\boldsymbol{y}}_{\boldsymbol{b}})$ are nondominated.

To systematically measure the difficulty of satisfaction under multi-dimensional propositions and calculate the relationships therein, we extend the notion of hypervolume indicator:

$$H(\overline{\boldsymbol{g}}, \boldsymbol{\mathcal{Y}}) = \lambda(\bigcup_{\overline{\boldsymbol{y}}_{\boldsymbol{k}} \in \boldsymbol{\mathcal{Y}}} \{\overline{\boldsymbol{z}} | \overline{\boldsymbol{g}}(\overline{\boldsymbol{y}}_{\boldsymbol{k}}) \prec \overline{\boldsymbol{g}}(\overline{\boldsymbol{z}}) \prec \overline{\boldsymbol{r}}\}) \tag{1}$$

whereby $\overline{\boldsymbol{r}}$ is the reference nadir point and $\overline{\boldsymbol{g}}$ is the given vector of functions that calculates the satisfaction scores over all dimensions of non-functional metrics. λ is the Lebesgue measure and we are interested in the total volume of the hypercube covered by all points (vectors of scores) in $\boldsymbol{\mathcal{Y}}$.

As shown in Figure 3, since in our case, the objective values are the (normalized) satisfaction scores ranging between 0 and 1 while preferring a greater value, we use $\{-0.1, -0.1, ..., -0.1\}$ as the reference point, thus the greater the H, the larger area/volume would be covered by a vector of propositions, and hence the better. By transforming the non-functional metric values via the functions in a proposition vector, we incorporate specific preferences into the indicator while the hypervolume values of the same points across different proposition vectors are comparable as each dimension represents a satisfaction score.

Using hypervolume, we can define multi-dimensional refinement similar to the case of a single dimension. However, since we often do not know the relationships between the non-functional metrics,

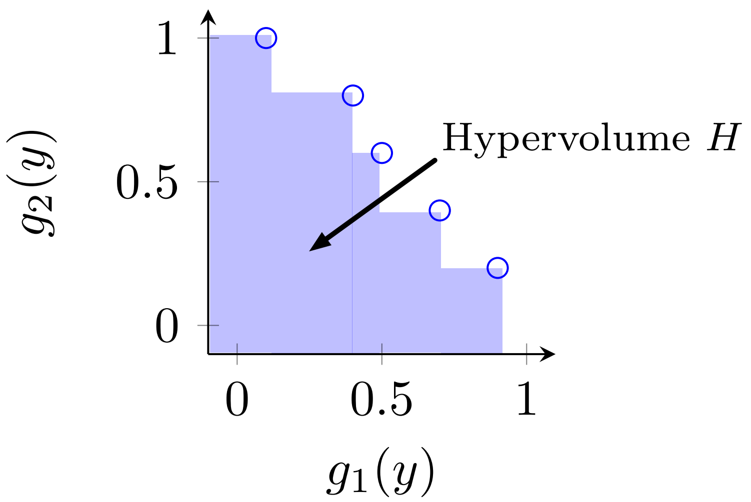
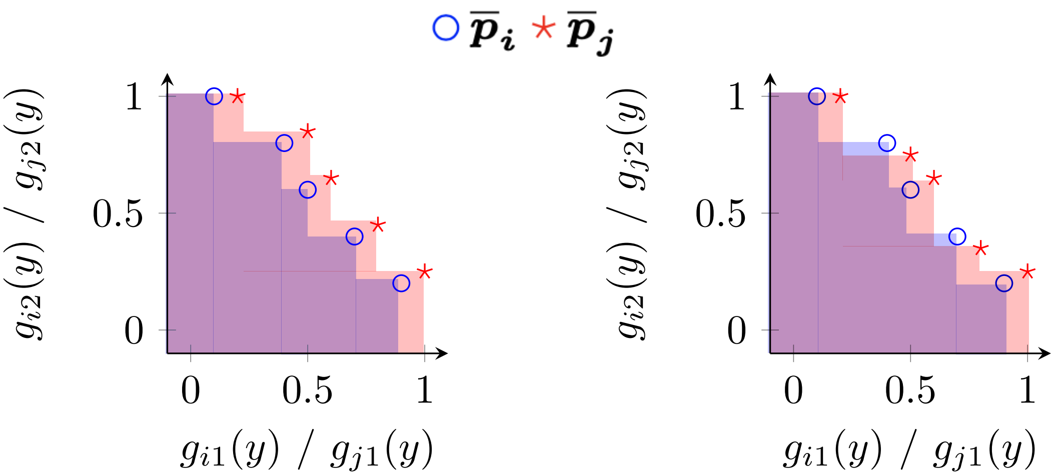


Figure 3: Hypervolume on some configurations measured by two-dimensional propositions.



(a) Strong refinement $\overline{\boldsymbol{p}}_{\boldsymbol{i}} \rightrightarrows \overline{\boldsymbol{p}}_{\boldsymbol{j}}$

(b) Weak refinement $\overline{\boldsymbol{p}}_{\boldsymbol{i}} \rightarrow \overline{\boldsymbol{p}}_{\boldsymbol{j}}$

Figure 4: Refinement of propositions on two non-functional metrics.

such a definition needs to be specific to a set of given configurations rather than being generalizable. Formally, given two n-dimensional vectors of propositions, $\overline{\boldsymbol{p}}_{\boldsymbol{i}}$ and $\overline{\boldsymbol{p}}_{\boldsymbol{j}}$, together with a set of non-functional metric values achieved by some configurations (denoted as $\boldsymbol{\mathcal{Y}} = \{\overline{\boldsymbol{y}}_{\boldsymbol{1}}, \overline{\boldsymbol{y}}_{\boldsymbol{2}}, ..., \overline{\boldsymbol{y}}_{\boldsymbol{m}}\}$ where $\overline{\boldsymbol{y}}_{\boldsymbol{m}}$ is a vector that contains the values of all non-functional metrics considered), the refinement relationships between the proposition vectors on $\boldsymbol{\mathcal{Y}}$ is:

Multi-Dimensional Refinement: $\overline{\boldsymbol{p}}_{\boldsymbol{i}}$ can be relaxed to $\overline{\boldsymbol{p}}_{\boldsymbol{j}}$, or $\overline{\boldsymbol{p}}_{\boldsymbol{j}}$ can be tightened to $\overline{\boldsymbol{p}}_{\boldsymbol{i}}$, if and only if $H(\overline{\boldsymbol{g}}_{\boldsymbol{i}}, \boldsymbol{\mathcal{Y}}) < H(\overline{\boldsymbol{g}}_{\boldsymbol{j}}, \boldsymbol{\mathcal{Y}})$.

whereby $\overline{\boldsymbol{g}}_{\boldsymbol{i}}$ and $\overline{\boldsymbol{g}}_{\boldsymbol{j}}$ are the corresponding vector of functions that compute the satisfaction scores. As shown in Figure 4, despite the conflicts between satisfying the requirements for distinct non-functional metrics, a vector of propositions ($\overline{\boldsymbol{p}}_{\boldsymbol{j}}$) are said to be satisfied more than the other vector ($\overline{\boldsymbol{p}}_{\boldsymbol{i}}$) in general if the volume it covers in the high-dimensional space of the satisfaction scores is larger given the same points of configurations. Again, when the volumes covered by two vectors of propositions are the same even though they contain distinct propositions, they are said to be equally satisfiable, or $\overline{\boldsymbol{p}}_{\boldsymbol{i}} \equiv \overline{\boldsymbol{p}}_{\boldsymbol{j}}$. As such, similar to the 1-dimensional case, we can define strong multi-dimensional refinement as:

Strong Multi-Dimensional Refinement: $\overline{\boldsymbol{p}}_{\boldsymbol{i}}$ can be relaxed to $\overline{\boldsymbol{p}}_{\boldsymbol{j}}$ ($\overline{\boldsymbol{p}}_{\boldsymbol{i}} \rightrightarrows \overline{\boldsymbol{p}}_{\boldsymbol{j}}$), or $\overline{\boldsymbol{p}}_{\boldsymbol{j}}$ can be tightened to $\overline{\boldsymbol{p}}_{\boldsymbol{i}}$ ($\overline{\boldsymbol{p}}_{\boldsymbol{j}} \leftleftarrows \overline{\boldsymbol{p}}_{\boldsymbol{i}}$), if and only if $H(\overline{\boldsymbol{g}}_{\boldsymbol{i}}, \boldsymbol{\mathcal{Y}}) < H(\overline{\boldsymbol{g}}_{\boldsymbol{j}}, \boldsymbol{\mathcal{Y}})$ while $\forall \overline{\boldsymbol{y}}_{\boldsymbol{a}} \in \boldsymbol{\mathcal{Y}} : \exists \overline{\boldsymbol{y}}_{\boldsymbol{b}} \in \boldsymbol{\mathcal{Y}} : \overline{\boldsymbol{g}}_{\boldsymbol{j}}(\overline{\boldsymbol{y}}_{\boldsymbol{b}}) \prec \overline{\boldsymbol{g}}_{\boldsymbol{i}}(\overline{\boldsymbol{y}}_{\boldsymbol{a}})$.

Likewise, the refinement over multiple dimensions of propositions can also be weak:

Weak Multi-Dimensional Refinement: $\overline{\boldsymbol{p}}_{\boldsymbol{i}}$ can be relaxed to $\overline{\boldsymbol{p}}_{\boldsymbol{j}}$ ($\overline{\boldsymbol{p}}_{\boldsymbol{i}} \rightarrow \overline{\boldsymbol{p}}_{\boldsymbol{j}}$), or $\overline{\boldsymbol{p}}_{\boldsymbol{j}}$ can be tightened to $\overline{\boldsymbol{p}}_{\boldsymbol{i}}$ ($\overline{\boldsymbol{p}}_{\boldsymbol{j}} \leftarrow \overline{\boldsymbol{p}}_{\boldsymbol{i}}$), if and only if $H(\overline{\boldsymbol{g}}_{\boldsymbol{i}}, \boldsymbol{\mathcal{Y}}) < H(\overline{\boldsymbol{g}}_{\boldsymbol{j}}, \boldsymbol{\mathcal{Y}})$ while $\exists \overline{\boldsymbol{y}}_{\boldsymbol{a}} \in \boldsymbol{\mathcal{Y}} : \nexists \overline{\boldsymbol{y}}_{\boldsymbol{b}} \in \boldsymbol{\mathcal{Y}} : \overline{\boldsymbol{g}}_{\boldsymbol{j}}(\overline{\boldsymbol{y}}_{\boldsymbol{b}}) \prec \overline{\boldsymbol{g}}_{\boldsymbol{i}}(\overline{\boldsymbol{y}}_{\boldsymbol{a}})$.

Intuitively, when the hypervolume differs, a multi-dimensional refinement, e.g., $\overline{\boldsymbol{p}}_{\boldsymbol{i}}$ is relaxed to $\overline{\boldsymbol{p}}_{\boldsymbol{j}}$, is strong when the satisfaction scores of all points in $\boldsymbol{\mathcal{Y}}$ over $\overline{\boldsymbol{p}}_{\boldsymbol{i}}$ are dominated by the score of at least one point from $\boldsymbol{\mathcal{Y}}$ over $\overline{\boldsymbol{p}}_{\boldsymbol{j}}$. In contrast, the refinement is weak if there is at least one such point whose satisfaction scores over $\overline{\boldsymbol{p}}_{\boldsymbol{i}}$ is nondominated to those of the points over $\overline{\boldsymbol{p}}_{\boldsymbol{j}}$.

6.3 How to Refine in Trade-off?

With the above theoretical foundation, we can draw the following actions for single-dimensional refinement:

1. Modify the parameter(s), i.e., interval and/or $\overline{s}$, in the proposition.
2. Modify, add, or remove the fragment(s) in the proposition.
3. Change on both (1) and (2).

Actions for multi-dimensional refinements are essentially derived from those of single-dimensional refinement: they are realized by conducting refinement on one or more propositions individually, which eventually affects the overall difficulty of satisfaction on the vector of propositions.