jCGP, JGP, and KGP have cooperatively worked to define RootLGR for CJK TLD labels. Here CJK means Chinese, Japanese, Korean languages, which have Han (also called as Kanji in Japanese and Hanja in Korean) script in common. As these three languages have different rules in using the same script, RootLGRs for CJK are different and thus the allocatable labels are different and this fact hinders the consistent and satisfactory definition of CJK RootLGRs.

This letter is to request ICANN to devise a TLD application/evaluation process to complement the function of RootLGRs to solve the above-mentioned difficulty.

<Problem statement>

Language Generation Panels intensively work on the definition of RootLGRs for their respective languages. For a language whose character set (repertoire) contains variant characters, the number of variant labels of a specific label may be more than one - even thousands or more. For such cases, reduction of the number of allocatable labels to a moderate number is requested by Integration Panel (IP). However, in some languages (such as Japanese), arbitrary combinations of characters within its repertoire is allowed as living words. In such cases, no reasonable set of rules will not exist to reduce the number of allocatable labels.

Even in the case where rules can be defined as general principles, exceptions usually exist for any language. For example, Chinese has general principles demanding “traditional and simplified characters cannot be mixed in a word”. However some words may have comingled use of traditional and simplified characters, especially when traditional proper noun is followed by a simplified attribution, such as ‘國立臺北大学’, where國 and 臺 are traditional while 学 is simplified. Sometimes more than one variant labels that contain comingled use of simplified and traditional characters are wished to be registered. Even in the case “allocatable labels must consist of all-traditional characters or all-traditional characters” is forced, if a traditional character has variants, more than one variant labels may be wished to be registered. \*\*\*\* more realistic examples are desirable \*\*\*\*

The above issue becomes more serious when scripts are shared by more than one languages and variants are defined in some of the languages. In this case, sets of labels that are not variant at all in one language are considered as variants by passively importing foreign variant definition.

<possible solutions>

The above situation can be solved either by

[Case-1] no variants are defined (all characters are independent) for all the language RootLGRs sharing the script

[Case-2] reduction of allocatable labels can be waivered

[Case-3] the number of ‘allocatable’ labels that are actually delegated is limited even if the number of allocatable labels generated by RootLGR is big (e.g., more than 10).

[Case-4] labels that are not attributed ‘allocatable’ by RootLGR can be delegated in special cases

[Case-1] should be avoided if one or more characters are strictly regarded as the same character in some language. This is needed to avoid improper usage of domain names such as phishing.

[Case-2] is a simple solution. However, IP has delivered its views that it is not desirable if thousands of variant labels remain as candidates to be delegated and to be in root DNS in the future. Such fear may become true if allocatable (but not yet delegated) TLD labels will be delegated for (almost) free of charge without limitation.

[Case-3], however, will solve (at least relax) the situation of [Case-2]. In this case, string application/evaluation process solves the situation in combination with RootLGR. It should be assured that the upper-limit number of the delegated variant labels by the application/evaluation process (not by RootLGR).

[Case-4], on the contrary, will solve the situation when [Case-2] is not employed. I.e., [Case-4] solves the situation where RootLGR can generate only a limited number of allocatable variant labels that may not include the variant labels wished by the TLD applicant. To implement [Case-4], string application/evaluation process should wrap the parallel execution of RootLGRs, each of which tales a wished variant label as input and yields allocatable labels including the original input label at least.

In summary, [Case-1] is not desirable from the viewpoint of generation panels because some languages clearly have variant characters. If [Case-2] is not allowed, [Case-3] and/or [Case-4] are the options to solve this problem. Both Case-3 and Case-4 requires an application/evaluation process that wraps RootLGR and solves the problem in combination with RootLGR. The details of [Case-3] and [Case-4] are described below.

<sample scenario of [Case-3] : limitation on the numbers of delegated TLDs>

When a TLD applicant applies for a TLD string, RootLGR takes the string as input and generates a set of variant labels. And if the corresponding language RootLGR intrinsically defines all the characters as independent and all the variants are imported from other language(s), all those variant labels are marked as ‘allocatable’.

After that, if the same applicant try to apply for an ‘allocatable label’, the label will be delegated. This kind of application and delegation can be repeated at most N times. Here, N is a constant number, 4 for example.

<sample scenario of [Case-4] : limitation on delegated TLDs by applicant declaration>

RootLGR generates very limited number of ‘allocatable’ labels including the original input label, and the other labels are marked ‘blocked’ (or ‘prohibited’). In the case if one or more such blocked labels are requested to be delegated by the applicant at the same time of the original application, the complement process for RootLGR generates allocatable variant labels that are the union of labels that applicant specifies and their allocatable variant labels. This allows the applicant to have all the labels of his/her choice to be delegated, if (s)he knows all the variant labels (s)she wants.

The process takes one or more original labels at the same time and generates variant labels that are categorized into following three groups.

1: allocatable variant labels, including original labels

2: invalid variant labels, which are variant labels that include characters out of repertoire

3: blocked labels, which are all variant labels that are not allocatable or invalid

This process is useful especially when RootLGR marks ‘block’ to a subset of variant labels but the TLD applicant needs one or more ‘blocked’ labels to be delegated. For example, let’s consider the case where mixed use of simplified Han characters and traditional Han characters in a single TLD string is prohibited. In this case, 国立大学 and 國立大學 are marked ‘allocatable’ but ‘国立大學’ and ‘國立大学’ are marked ‘block’ by RootLGR, because 国 and 学 are simplified while 國 and 學 are traditional. But, for some justifiable reasons, an applicant may want to have both ‘国立大學’ and ‘國立大学’ delegated. \*\*\*\*\* needs examples with justifiable reasons \*\*\*\*

This situation can be solved by introducing a process that wraps one or more RootLGR execution, each of which works with each original input label. Such a process takes one or more original input labels and generates allocatable/blocked/invalid labels as output, where original input labels are marked as ‘allocatable’ in the output. Refer to the figures of this proposed process.



<conclusion>

In general, the applicant knows which string he/she wants to use when he/she applies for TLDs, and that’s why he/she applies for a TLD of such a string. And in some cases, applicant wants to use one or more variant labels when he/she applies for a TLD. Therefore, in general, if applied-for labels are all marked as ‘allocatable’, he/she will not demand further delegation of more variant labels in the future. This is the background of this request.