Rick's Rule, The PTO's Long Standing Failure to Address Defects in its Public Facing Electronic Interfaces

By Rick Neifeld, Neifeld IP Law, PLLC

There are some things about the USPTO which are exceptional. One thing that I find exceptional is the level of professionalism in the USPTO when dealing with substantive patent law issues, and particularly at the upper echelons of authority.

However, down in the guts of the USPTO, there are things that languish. I speak, in particular of the public facing data systems of the USPTO. It has been almost two decades since I first voiced my concerns on this issue and voiced them to USPTO officials whom I thought might act to rectify the obvious deficiencies. After about the third time I brought these things to the attention of USPTO officials, without affecting any change, I gave up. Yet I continued to identify these problems in patent attorney online forums.

You may ask, why this article is called "Rick's Rule." The answer is that I have complained about these issues so many times in public forums, that folks started calling the problem, "Rick's Rule." My colleague, Carl Oppedahl, recently explained the genesis of Rick's Rule. He stated:

> I will note that this is a specific example of the general rule which I think is by now known as "Rick's Rule". I think it was Rick Neifeld who articulated a rule that USPTO should comply with but has consistently failed to comply with.
>
> The rule, stated in practical terms, is that USPTO ought to design its systems so that you can copy and paste any important number from any USPTO system into any other USPTO system and it should work.
>
> There are several alternative formulations of Rick's Rule:
>
> • It is not acceptable design if any USPTO system pukes when a user pastes into the system an important number that the user has copied from any other USPTO system.
>
> • It is not acceptable design if any USPTO system fails to respond correctly when a user pastes into the system an important number that the user has copied from any other USPTO system.
>
> From the computer programming point of view, Rick's Rule could be easily implemented by competent computer programmers.
>
> That's what regular expressions are for. Stuff like this is trivially easy to accomplish using regular expressions. A regular expression can trim whitespace, which is something that the USPTO developers seem to be largely incompetent to do. A regular expression can pad leading zeroes if that is what is needed. A regular expression can insert a dash or a virgule or some commas, or can excise a dash or a virgule or some commas.
>
> You could probably put ten competent computer programmers on this project on a Saturday morning and give them suitable ethnic food and caffeine-bearing beverages and they could correct the flaws relating to this issue on every external-facing USPTO system in the course of a single weekend. And

they would have part of Saturday and all of Sunday left over.

Unfortunately that is not how the USPTO works, on at least two important levels of analysis.

First, it is very clear that no developer at the USPTO ever thinks about what it is like for the paying customer to do things. No developer at the USPTO ever asks "can the paying customer copy a reel and frame number from USPTO system A and paste it into USPTO system B and will it work?"

Second, the USPTO does not have any resources or processes in place to permit sensible corrections of design defects in the way that I described. To fix even the simplest design flaw, it is necessary to write up a super detailed bidding document, the kind of document that guarantees that there are only half a dozen companies (Northrup Grumman, Lockheed Martin, Boeing) that are even capable of putting in a bid. No normal computer programmer or group of computer programmers could ever hope to bid in the cumbersome mechanisms that USPTO uses for things like this. [List serve email, subject line, "Re: [Patentcenter] Patent Center - Assignment Reel/Frame Numbers and Leading Zeros", dated "Fri 12/4/2020 1:59 PM", posted at "https://oppedahl-lists.com/mailman/private/patentcenter_oppedahl-lists.com/2020-December/000474.html"]

In the hopes that someone in a position of influence of the USPTO's IT infrastructure will take heed, here is a list of issues that fall under "Rick's Rule."

1. Lack of "regular expressions" (see https://en.wikipedia.org/wiki/Regular_expression) across the USPTO's customer facing interfaces.

This is particularly annoying because the USPTO clearly knows how to regularize expressions and does so when it directly affects their own bottom line. As in https://fees.uspto.gov/MaintenanceFees which is the interface for users to pay maintenance fees. In this interface, it does not matter whether your application number includes slashes and/or commas or padding before or after the number. The interface works regardless of these irregularities of expressions.

In contrast, the data lookup interfaces that do not directly result in PTO funds, fail to take data in various forms, and particularly in the forms that other USPTO interfaces present.

For one example, when one looks up a patent application in PAIR, the record shows a bunch of data. For example, it shows the application publication number, such as "US 2001-0000044 A1". That is a form of the publication identifying the country ("US"), and that the publication is pre grant (A1), following conventional rules used worldwide. Yet back at the "http://patft.uspto.gov/netahtml/PTO/srchnum.htm" interface (for retrieving published patent applications based upon their publication numbers), the PTO server pukes on "US 2001-0000044 A1". And likewise, the USPTO's assignment interface, when set to retrieve assignments in response to publication numbers, see https://assignment.uspto.gov/patent/index.html#/patent/search, also pukes on "US 2001-0000044 A1".

2. Failure of "http://patft.uspto.gov/netahtml/PTO/srchnum.htm" to set the focus to the only data

entry field present on this page, the one in which to enter a patent publication number. So, when a user navigates to this URL and starts to type in, or pastes data in the clipboard, nothing happens. This is because the page fails to set the *focus* to the only place in the page into which data can be entered; the publication number input box. And the fix is apparently trivial. So trivial that I posted a corrected page which does set the focus upon navigating to the page. Compare the USPTO's page "http://patft.uspto.gov/netahtml/PTO/srchnum.htm" to my page "https://neifeld.com/PGPs%20(AutoFocusExceptForIE).htm."

3. Lack of identification of fees to petitions to petitions.
At least in EFS web (which will eventually be replaced by PatentCenter), the petition fee payment page only allows selection of certain classes of petitions by "Group," instead of identify the type of petition the user is attempting to pay for.

For example one can and must select "1462 $420 Petition Fee - 37 CFR 1.17(F) (GROUP I)" to pay the fee for certain petitions. However, users identify petitions by 37 CFR rule numbers with which they are associated, and with the names of those rules, and not within non-descriptive "GROUP" numbers. Like a 37 CFR 1.181 petition, a 37 CFR 1.182 petition, and a 37 CFR 1.183 petition, which specify rule numbers. And like a petition to accord a filing date, which specifies the desired relief. No practitioner thinks of their petition as a "Group I" petition, because such a designation is meaningless.

In order to determine which "Group" to check in the "Petition Filing Fees" section of the "Calculate Fees" tab in EFS, a user must find and inspect a copy of 1.17 to see, for example if the name of their petition appears in the "1.17(F) (GROUP I)" section. In fact, 1.17(f) does not identify any "GROUP," such as "(GROUP I)." In other words, the "GROUP" association is not a rule, but a concept made up by the USPTO for its convenience in grouping fees, apparently for the convenience of their internal accounting department's designation of internal fee codes. What rule 1.17(f) states is clarifying. (Also note there is no "1.17(F)" but instead only a Rule 1.17(f)" states the following:

> (f) For filing a petition under one of the following sections that refers to this paragraph (f): ***
> §1.36(a)—for revocation of a power of attorney by fewer than all of the applicants.
> §1.53(e)—to accord a filing date.
> §1.182—for a decision on a question not specifically provided for in an application for a patent.
> §1.183—to suspend the rules in an application for a patent.
> §1.741(b)—to accord a filing date to an application under §1.740 for an extension of a patent term.
> §1.1023—to review the filing date of an international design application.

This section identifies the rule number and corresponding names of the relief requested by each petition. "§1.36(a)" refers to 37 CFR 1.36(a), a rule. The text "revocation of a power of attorney by fewer than all of the applicants" identifies the relief requested by a rule 1.36(a) petition, by

naming the relief. These are the things a customer of the USPTO would recognize, when trying to pay fees for a petition. Yet, EFS fails to identify these rules and names of petitions, even though these are the names the public uses when naming petitions. So, for example, when a member of the public prepares a petition to "accord a filing date" pursuant to 37 CFR 1.53(e), they look in EFS for the fee to pay a petition to "accord a filing date" pursuant to 37 CFR 1.53(e). Yet no such information appears in EFS's "Calculate Fees" page. Instead, this page appears to be designed for the convenience of the USPTO's accounting department, merely by grouping petitions with the same fee into one fee code and calling that group, "GROUP I".

4. Idiotically programmed patent and patent publication search retrieval logic. Part of the USPTO's mission is to disseminate data about inventions to promote the constitutional mandate of promoting the useful arts. So the USPTO provides a US issued patent search interface, and a US published patent applications search interface to achieve that goal. Both interfaces work reasonably well to return a list of publications and titles, in response to any search query. For complicated search logic, the query can take some time, for example 1 to 2 minutes. But the result is a list of patents or published applications and their titles, in which each member in the list is linked back to the actual text of the patent or published application. And the USPTO's data servers can serve up a patent or a publication very quickly, in fractions of second. They can do that by querying by the patent or published application number. It is easy for a computer system to look up a number and return to the user a file directly linked to that number. So for example when a user enters into the patent number lookup page http://patft.uspto.gov/netahtml/PTO/srchnum.htm a patent number, the USPTO server returns the file for that patent in a fraction of a second. So you would think that a user examining any member in such a list, by clicking on it, would result in the USPTO server finding the file for the patent and returning that file in a fraction of a second. But it is does not.

In fact, clicking on any patent or publication number in the list provided by the USPTO server in response to a search query can take a very long time to return the file for the patent or publication number. In fact, it takes as long as the original search query to return the list of patents or publications. Apparently, this occurs because the USPTO server's hyperlink in each patent or publication in the list provided to the user's computer, is a link to rerun the original search query, and not a link to the file for the patent or publication number. Apparently, clicking on a patent or publication number in the list resulting from search reruns the original search logic again to retrieve the the text of the patent or publication number, and therefore takes about as long, such as the 1 to 2 minutes for the example above.

Incredibly, the USPTO chews up its own server CPU time, unnecessarily spinning its wheels, to redo complicated search logic, each time such a list member is clicked. To be clear, the USPTO search results appear to contain contains requiring rerun of the original search logic, links to the simple number lookup based upon the identified patent or publication. That both chews up USPTO server CPU time unnecessarily, and fails to timely serve up the resulting patent text to the user. Unnecessarily making a public user's review of search results inefficient.