Kinome Render: a stand-alone and web-accessible tool to annotate the human protein kinome tree

Human protein kinases play fundamental roles mediating the majority of signal transduction pathways in eukaryotic cells as well as a multitude of other processes involved in metabolism, cell-cycle regulation, cellular shape, motility, differentiation and apoptosis. The human protein kinome contains 518 members. Most studies that focus on the human kinome require, at some point, the visualization of large amounts of data. The visualization of such data within the framework of a philogenetic tree may help identify key relationships between different protein kinases in view of their evolutionary distance and the information used to annotate the kinome tree. For example, studies that focus on the promiscuity of kinase inhibitors can benefit from the annotations to depict inhibitor's specificity across kinase groups. Images involving the mapping of information into the kinome tree are common. However, producing such a figure manually can be a long arduous process prone to errors. To circumvent this issue, we have developed a web-based tool called Kinome Render (KR) that produces customized annotations on the human kinome tree. KR allows the creation and automatic overlay of customizable text or shape-based annotations of different sizes and colors on the human kinome tree. A stand-alone version is also available and can be run locally. The web interface can be accessed at: http://bcb.med.usherbrooke.ca/kinomerender

Kinome Render: a stand-alone and web-accessible tool to annotate the human protein kinome tree

Matthieu Chartier[1]*, Thierry Chénard[1]*, Jonathan Barker[2], Rafael Najmanovich[1]
[1]Département de biochimie, Faculté de médecine et des sciences de la santé, Université de Sherbrooke, Québec, Canada
[2]European Bioinformatics Institute, Wellcome Trust Genome Campus, Hinxton, Cambridge CB10 1SD, UK
*These authors contributed equally to the published work

Abstract

Human protein kinases play fundamental roles mediating the majority of signal transduction pathways in eukaryotic cells as well as a multitude of other processes involved in metabolism, cell-cycle regulation, cellular shape, motility, differentiation and apoptosis. The human protein kinome contains 518 members. Most studies that focus on the human kinome require, at some point, the visualization of large amounts of data. The visualization of such data within the framework of a philogenetic tree may help identify key relationships between different protein kinases in view of their evolutionary distance and the information used to annotate the kinome tree. For example, studies that focus on the promiscuity of kinase inhibitors can benefit from the annotations to depict binding affinities across kinase groups. Images involving the mapping of information into the kinome tree are common. However, producing such figures manually can be a long arduous process prone to errors. To circumvent this issue, we have developed a web-based tool called Kinome Render (KR) that produces customized annotations on the human kinome tree. KR allows the creation and automatic overlay of customizable text or shape-based annotations of different sizes and colors on the human kinome tree. The web interface can be accessed at: http://bcb.med.usherbrooke.ca/kinomerender. A stand-alone version is also available and can be run locally.

Introduction

The human genome codes for 518 protein kinases, also known as the human kinome, which represent a little less than 2% of all human genes (Manning, 2002). These kinases regulate multiple biological processes such as apoptosis, transcription, mobility of the cell and metabolism by catalyzing the covalent bonding of a phosphate group to an amino acid with a free hydroxyl group. The proliferation of cancer cells often involves alterations in kinase activity in signalization pathways. As such, human protein kinases represent important targets in drug design.

Panels of binding assays and bioinformatic analyses related to the human kinome produce large quantities of data, which are hard to analyze without appropriate visualization tools. One way of visualizing data is by annotating the kinome phylogenetic tree that originally accompanied the first analysis of the human kinome (Manning, 2002). This phylogenetic tree depicts the relationship between different members of the superfamily based on the sequence similarity of their catalytic domains. It became an iconic representation of the kinome due to its high graphic and artistic qualities. Annotated versions of this dendrogram have been used multiple times in the past in articles related to the human kinome (Karaman et al., 2008, Marsden & Knapp, 2008, Edwards, 2009, Fedorov, Müller, & Knapp, 2010). Some figures in these publications were created by hand in a laborious effort (A. Edwards, B. Marsden, & S. Knapp – personal communication). For the reason aforementioned, we have created Kinome Render, a tool that can be used via a web interface or that can be downloaded to run locally. KR

enables the annotation of the human kinome tree while reducing the amount of work involved and, more importantly, the risk of errors.

The tool works by overlaying annotations of specific kinases on a template of the human kinase phylogenetic tree. Annotations, which can be circles, polygons or text of different sizes and colors, are centered at the coordinates corresponding to the leaf of the tree that defines the kinase to be annotated. The annotations can be combined to represent practically any kind of data about the kinome, e.g. availability of known three-dimensional structures (Marsden & Knapp, 2008) or interaction maps between small-molecule inhibitors and kinases (Karaman et al., 2008).

The Kinome Render tool

Presentation

In KR, the text or shape-based annotations are printed at the position of kinases on the human kinome tree. To render the annotated tree, KR ultimately requires a text file written in KR syntax that contains all the information needed to draw the annotations. The web interface allows the creation of an annotated tree without any knowledge of the KR syntax. There are three ways to create an annotated tree using KR (summarized in Figure 1): (1) by running the stand-alone version locally, (2) by using the web interface directly, or (3) by uploading a KR file through the web interface. Using the stand-alone version requires a text file that lists the kinase(s) with their corresponding annotations as input. This file must be written in KR syntax, a flexible simple language described below. As mentioned, the web interface can be used in two ways. First, by creating an annotated tree from scratch, to help those not familiar with running stand-alone programs locally and / or with the KR syntax. Second, by uploading a file written in the KR syntax, which loads the annotations encoded in the file to the interface, allowing them to be edited. From there, additional annotations can be created and old ones edited. KR files uploaded to the interface must be plain text.

KR works on top of a database containing a total of 523 entries (Table S1) retrieved from Manning (2002). Each of the 523 entries correspond to a unique leaf of the original human kinome tree. From the initial 518 human kinases, 8 atypical kinases (BCR, FASTK, G11, H11, TAF1, TAF1L, A6, A6r) are part of a family with only one or two members and are absent from the original tree. Additionally, there are 13 kinases that have two distinct catalytic domains (JAK1, JAK2, JAK3, MSK1, MSK2, RSK1, RSK2, RSK3, Tyk2, GCN2, RSK4, SPEG, Obscn) and thus are present twice on the tree (their name is tagged with ~b). A list of corresponding names and synonyms, the Uniprot ID, group, family, subfamily, full protein sequence and kinase domain sequence are stored in the database for each entry. The kinase domain sequences of the atypical kinases are absent in Manning (2002) and thus not present in the database. The reason for compiling the protein sequences is explained below. The procedures to use KR via the web interface or to run it locally are described in the following sections.

The Kinome Render web interface

The KR web interface contains all the necessary tools to create an annotated human kinome tree. In the interface, users can add, modify or delete annotations and render the annotated tree in 5 different formats (PostScript, PNG, JPG, PDF, TIFF). At any time, the user can save the KR file containing all the created annotations. This file can be uploaded in the interface to restore a previous session and make modifications on the annotations.

In the interface, the kinases are sorted in alphabetical order and the user can browse through the list to select one or more kinases to annotate. The selection can also be performed in two other ways. The first method is by searching the database using any combination of name, Manning accession ID, Uniprot ID, group, family or subfamily. When searching by name, the query string is matched against the Manning name, the protein's full name (e.g. Mitotic checkpoint serine/threonine-protein kinase BUB1 beta) and a list of synonyms retrieved from the Uniprot database (Magrane & Consortium, 2011). The second method is by sequence alignment. To do so, the user must input a protein sequence in fasta format that will be aligned to the full protein sequences of the 523 entries using Fasta (Pearson & Lipman, 1988). The

results are sorted by E-value and the percentages of similarity and identity are displayed. At anytime, the user can hover over the kinase name to display a tooltip containing the kinase's full name and a list of synonyms.

Once the desired kinases have been selected, the user can customize the annotation that will be overlaid at the position of the selected kinase(s) on the phylogenetic tree. There are 3 types of annotations: (1) a custom text string (max. 25 characters), (2) a shape, or (3) the kinase's name. The custom text strings or the kinase names can be inside a box or underlined. The shape can be a circle or a polygon of any number of sides, all with 3 different styles: shape outline, color filled shape or color filled shape with a black outline. For all annotation types, the size and color can be adjusted. The color can be selected from a predefined list or by entering the RGB values (RGB values must be between 0 and 1). The combination of these annotation styling properties provide interesting data visualization possibilities that can help reveal trends across the entire human protein kinase family (see examples below). The formatting options can be modified or deleted at any time by clicking the *modify* or *delete* buttons of any annotation. A legend can also be created on the final image. To do so, the elements of the legend must be written in KR syntax (described in the next section) and inserted in the legend section of the interface.

The user can create a maximum of 523 annotations (one for each leaf of the tree). Once all desired annotations are created, the user can choose to display the names of all remaining kinases. The names will appear in color black and size 10. This format cannot be changed. If this option is not selected, only the backbone of the tree and the defined annotations will be displayed. The user has the option to provide a filename and to choose an output format (PDF, JPEG, PNG or TIFF). Note that a PostScript image is generated by default in addition to the chosen format. There are two templates to choose from: one with only the typical kinases tree and a second one with the typical kinases tree accompanied by a small sub-tree with the atypical kinase families in the bottom left. If any atypical kinases are used in the annotations, the interface will constrain the selection to the second template.

Once the disclaimer is accepted and the form submitted, a page will provide links to download (1) the final image in the user-selected format, (2) the PostScript image, and (3) a KR file. As mentioned above, the KR file can be saved for use with the stand alone version or uploaded in the interface in order to restore all previously created annotations to either make modifications, add or delete annotations from the tree.

We understand that users might have concerns about the privacy of their data. Some files are stored temporarily in our servers for the proper functioning of the interface. These files are deleted automatically after a 72-hour period. We also provide the option to immediately delete all related files.

Running Kinome Render locally

The KR program behind the web interface is a Perl script that can be used as a stand-alone version. It generates a PostScript file that defines the final image. If other formats are required, this PostScript file should be converted using the appropriate programs. For example, this conversion is done automatically with some programs like Preview (Mac) or Adobe's Acrobat Reader (Windows or Mac). As previously mentioned, the KR file can also be uploaded directly to the web interface in order to render the image in multiple formats. The stand-alone version of the program takes as argument (1) a text file containing the annotations written in KR syntax (described below), (2) the template to be used (if its not template 1), and (3) the output location and file name. The templates are in the downloadable Kinome Render folder, available for download at http://bcb.med.usherbrooke.ca/kinomerender.php.

The KR format works with small one-line commands listed in Table 2. The first obligatory command for all annotations is the "*at kinaseName*" replacing *kinaseName* with the proper name (column "Name" in Table S1). This indicates to which kinase assign the annotation. By default, the color is black and the scale is 10. This can be changed using the "*color*" and "*scale*" commands. KR draws annotations in the same order they are written in the file. For this reason, annotations with the biggest scales should be written first so as not to hide smaller annotations in their vicinity. If a file is uploaded to the Interface, annotations are automatically sorted in descending order.

The annotation type is defined on the line that follows after the position, scale and color have been defined as described above. If the annotation to be drawn is a text, simply write "*text string*" replacing *string* with the desired label. Special characters such as Greek letters can be used. For example, to insert

an α simply write "[alpha]" (see Table 1 for a list of symbols and their corresponding code). If the text needs to be underlined or boxed, the commands "*underlined*" or "*boxed*" must be written immediately on the following line. Shapes (circle or polygon) can also be drawn with three different styles: (1) a shape's outline, (2) a color-filled shape or (3) a color-filled shape with black border. The syntax for each shape type is listed in Table 2. Some examples on how to use these commands appear below.

KR also allows the creation of legends. To do so, the user has to write the one-line command "*legend*" and, following this line, add the information of each element of the legend in the same format as the annotation. The "*next-line*" command can be used to jump to the following line of the legend. The "*space*" command can be added to concatenate a space at the current position. The legend can be enclosed in a box by adding the "*legendBox*" command at the very end of the legend. Refer to examples below for more details on how to use legends.

Finally, if the user wants all the unannotated kinases to be labelled with their respective name in black, size 10, the line "*remainder*" must be added at the end of the text file.

Examples

Ex.1 - Quick start. This example shows the basics of the KR syntax (Supplementary File S1). We will annotate the PINK1, MPSK1 and BIKE kinases, each with a different color, size and annotation type.

at PINK1
color 1 0 1
scale 30
polygon-filled 3

at MPSK1
color 0.92 0.08 0.08
scale 20
circle-lined

at BIKE
color 0 0.67 0
scale 10
text BIKE

A file submitted with the above code will yield what we see in Figure 2. Note that the whole kinome tree is rendered in the final figure but only a fraction is shown for simplicity. Also note that the annotations are written with the biggest scales first. Even though the annotations do not overlap in the final figure, this is a good practice.

Ex. 2 - Draw legends. For this example, we will draw a legend for the example above. The code below yields a legend (Figure 3). Again, the whole tree is generated but only the legend is shown in Figure 3 for simplicity. Supplementary File S2 contains the code of example 1 concatenated to example 2.

legend
scale 20

color 1 0 1
polygon-filled 3
space
text PINK1 is annotated with a pink triangle

next-line
color 0.92 0.08 0.08

circle-lined
space
text MPSK1 is annotated with a red circle

next-line
color 0 0.67 0
text BIKE is annotated in green

legendBox

One interesting thing to note in the last example is that the scale was set only once at the beginning. This will make all objects declared afterwards render at scale 20. The same thing is applicable to the color. For example, if we want all annotations or legend elements of our tree to be in orange, we can declare once at the beginning of our code "*color 1 0.65 0* ". Note that kinases annotated with the "*remainder*" command will always be black. The "*next-line*" command is used between each element of the legend. The "*space*" command is used to concatenate a space between two elements on the same line of the legend. The command "*legendBox*" is used at the very end of the legend to draw a box around the legend.

Advanced example 1. In this example, we combine information about the availability of PDB structures of kinase domains with the list of kinases studied by Karaman et al. (2008) (Figure 4). We use a color code to distinguish three possibilities: (1) protein kinases studied by Karaman (blue), (2) protein kinases with a PDB structure representing the kinase domain (red), and (3) protein kinases both studied by Karaman and with a PDB structure (green). The rest of the kinases are annotated in black using the "*remainder*" command. The KR file used to render this figure is available in the supplementary data (Supplementary File S3).

Advanced example 2. Karaman et al. (2008) contains many figures that use the human kinome tree to depict the promiscuity of a variety of drugs. In these figures, the size of the circles represent the affinity of the drug for a particular protein. The KR file for this figure is available in the supplementary data (Supplementary File S4). Processing this file with KR yields what we see in Figure 5. Visualizing the results using such technique allows to detect important information. It can reveal if a drug is promiscuous and if it has a propensity to bind a specific family. To render the figure, the binding data for Sunitinib was used. We can quickly see that the inhibitor Sunitinib is very promiscuous and binds very strongly to members of the TK family.

Conclusions

Efficient visualization of data is a key step in every scientific study. Human kinases are interesting and widely studied drug targets. The phylogenetic tree produced for these human kinases has become an iconic figure widely used in a large number of studies and can help reveal important trends. We have implemented a tool called Kinome Render that allows simple large-scale customized annotations of the human kinome tree. The method reduces the risk of errors prone to occur with hand made annotations. A user-friendly web interface was developed to make the tool easily accessible. We believe that KR can be a useful tool for anyone studying human kinases to quickly create annotated figures.

Availability

To access the interface and download Kinome Render, please visit http://bcb.med.usherbrooke.ca/kinomerender. To publish a figure obtained with KR, you must acknowledge in the following way: "Illustration reproduced courtesy of Cell Signaling Technology, Inc. (www.cellsignal.com)".

Acknowledgements

The authors would like to thank María Inés Zylber for the critical reading and editing of the manuscript. The phylogenetic tree of the human kinome is reproduced with permission of *Science* and Cell Signaling Technology, Inc.

References

Edwards, A. (2009). Large-scale structural biology of the human proteome. *Annual review of biochemistry*, *78*, 541–568. doi:10.1146/annurev.biochem.78.070907.103305

Fedorov, O., Müller, S., & Knapp, S. (2010). The (un)targeted cancer kinome. *Nature chemical biology*, *6*(3), 166–169. doi:10.1038/nchembio.297

Karaman, M. W., Herrgard, S., Treiber, D. K., Gallant, P., Atteridge, C. E., Campbell, B. T., et al. (2008). A quantitative analysis of kinase inhibitor selectivity. *Nature Biotechnology*, *26*(1), 127–132. doi: 10.1038/nbt1358

Magrane, M., & Consortium, U. (2011). UniProt Knowledgebase: a hub of integrated protein data. *Database*, *2011*(0), bar009–bar009. doi:10.1093/database/bar009

Manning, G. (2002). The Protein Kinase Complement of the Human Genome. *Science (New York, N.Y.)*, *298*(5600), 1912–1934. doi:10.1126/science.1075762

Marsden, B. D., & Knapp, S. (2008). Doing more than just the structure-structural genomics in kinase drug discovery. *Current opinion in chemical biology*, *12*(1), 40–45. doi:10.1016/j.cbpa. 2008.01.042

Pearson, W. R., & Lipman, D. J. (1988). Improved tools for biological sequence comparison. *Proceedings of the National Academy of Sciences of the United States of America*, *85*(8), 2444–2448.

Figure 1. There are 3 ways to create an annotated tree. (1) Write a KR file and use it as input to the stand-alone version of the program on a local machine. (2) Write a KR file and upload it to the web interface. This allows the edition of the annotations in the file using the web interface as well as to create new annotations. (3) Start directly from the web interface, create annotations and generate the annotated tree. You can save the KR file at any time while using the interface. You can save and restore your session on the interface with this file (step 2).

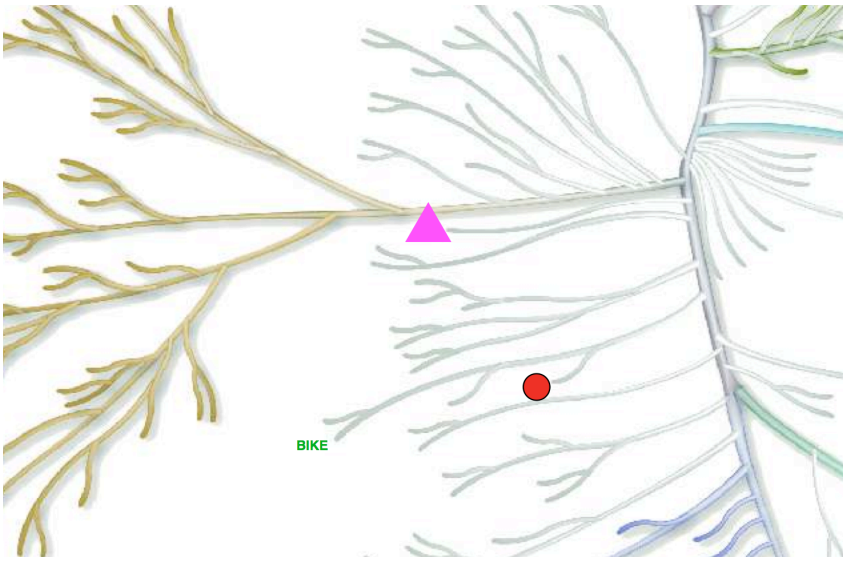

Figure 2. PINK1, MPSK1 and BIKE kinases labeled using different annotation types.

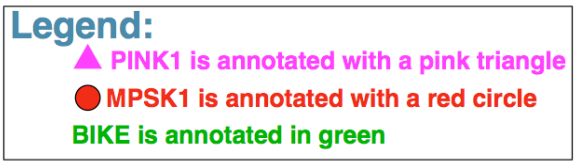

Figure 3. A legend example for Figure 2 to illustrate how to create legends.

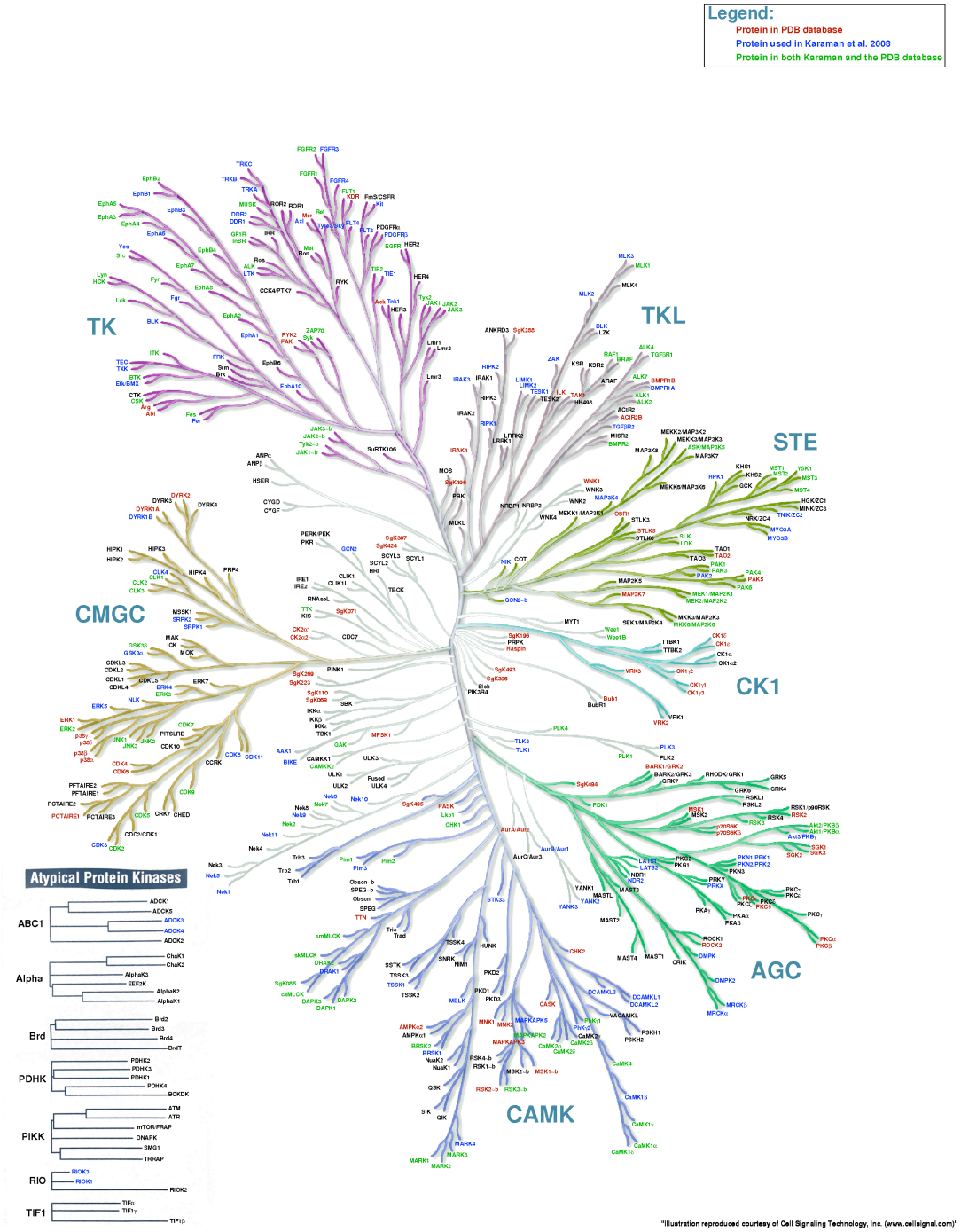

Figure 4. Phylogenetic tree created by Kinome Render showing protein kinases studied by Karaman *et al*., 2008 (blue), protein kinases with a PDB structure representing the kinase domain (red) and protein kinases both studied by Karaman and with a PDB structure (green).

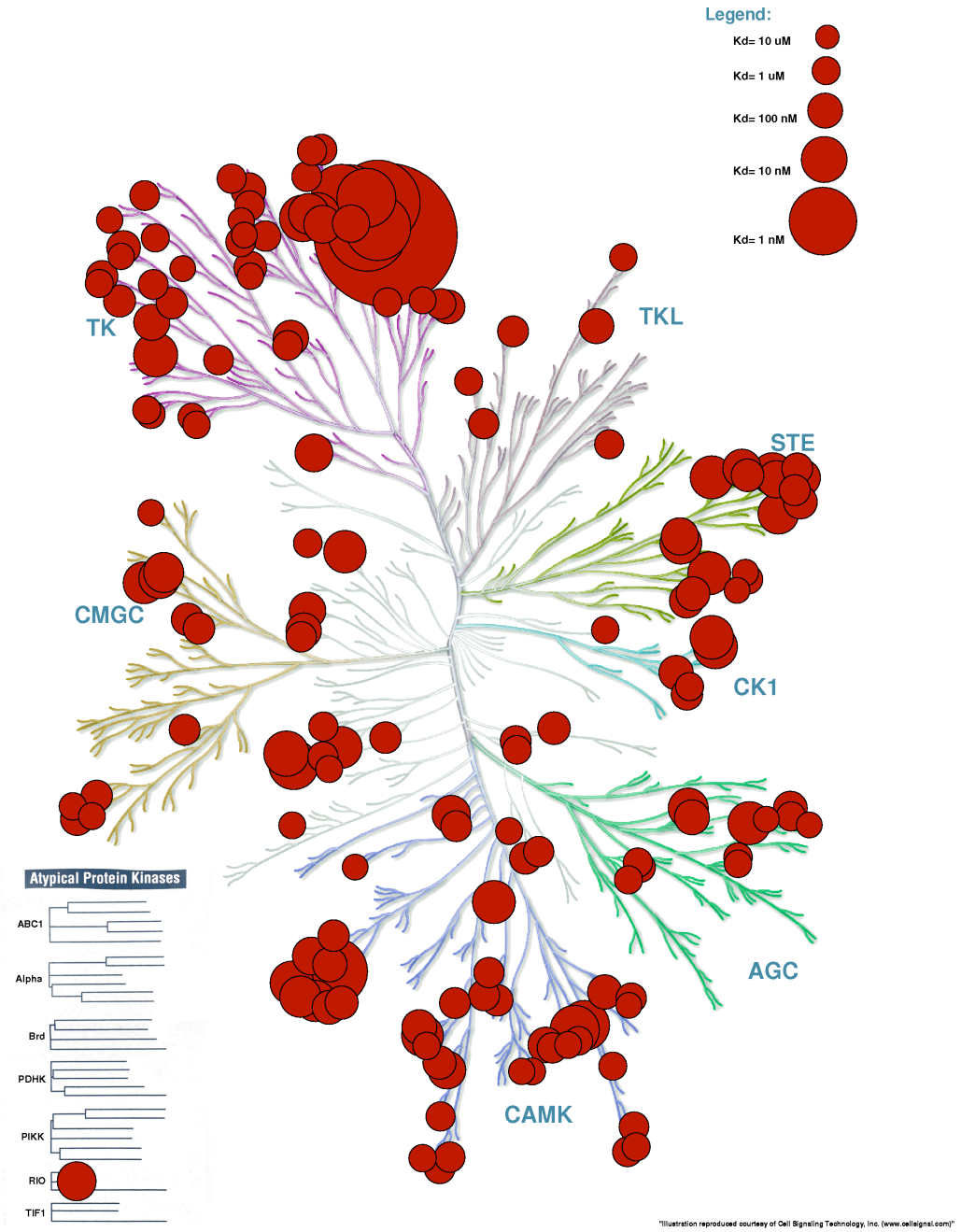

Figure 5. Recreation of the figure representing the affinity of Sunitinib for different kinases studied by Karaman *et al*., 2008 (using data of the article). The kinases for which Sunitinib has an affinity < 3 µM are annotated with a red circle. The bigger the circle, the higher the affinity.

Table 1. Special symbols can be printed by writing the code in the text string.

Code	Symbol	Code	Symbol
[alpha]	α	[sigma1]	ς
[beta]	β	[upsilon]	υ
[chi]	χ	[tau]	τ
[delta]	δ	[xi]	ξ
[Delta]	Δ	[psi]	ψ
[epsilon]	ε	[Psi]	Ψ
[phi]	ϕ	[zeta]	ζ
[gamma]	γ	[intersection]	∩
[Gamma]	Γ	[union]	∪
[eta]	η	[angle]	∠
[iota]	ι	[equivalence]	≡
[lambda]	λ	[plusminus]	±
[kappa]	κ	[lesserequal]	≤
[mu]	μ	[greaterequal]	≥
[nu]	ν	[diamond]	♦
[pi]	π	[heart]	♥
[Pi]	Π	[spade]	♠
[rho]	ρ	[club]	♣
[theta]	θ		
[Theta]	Θ		
[theta1]	ϑ		
[omega]	ω		
[Omega]	Ω		
[sigma]	σ		
[Sigma]	Σ		

Table 2. One-line commands of the Kinome Render format language.

Command	Description	Value
at (value)	specifies the kinase to which assign the annotation	name (see Table S1) i.e. Abl
color (value)	sets the color of the annotations	RGB values between 0 and 1 i.e. 1 0.67 0.5
scale (value)	sets the size of the annotations	integer (default 10)
text (value)	prints a string of text	string (max. 25 characters, see Table 1 for special characters)
circle circle-filled circle-lined	draws a circle draws a color-filled circle draws a color-filled circle with a black outline	N.A.
polygon (value) polygon-filled (value) polygon-lined (value)	draws a polygon of (value) sides draws a color-filled polygon draws a color-filled polygon with a black outline	integer (e.g.: polygon 4 will draw a square)
boxed	encloses the annotation in a box	N.A.
underlined	underlines the annotation	N.A.
remainder	prints the names of all unannotated kinases (add this command once at the last line of your file)	N.A.
legend	declares that the following lines describe the legend	N.A.
space	adds a trailing space in an element of the legend	N.A.
next-line	jumps to the next line in the legend	N.A
legendBox	inserts a box around the legend; must be placed in the last line of the legend section	

Figure 1

Kinome Render workflow

Figure 2

KR annotation example 1

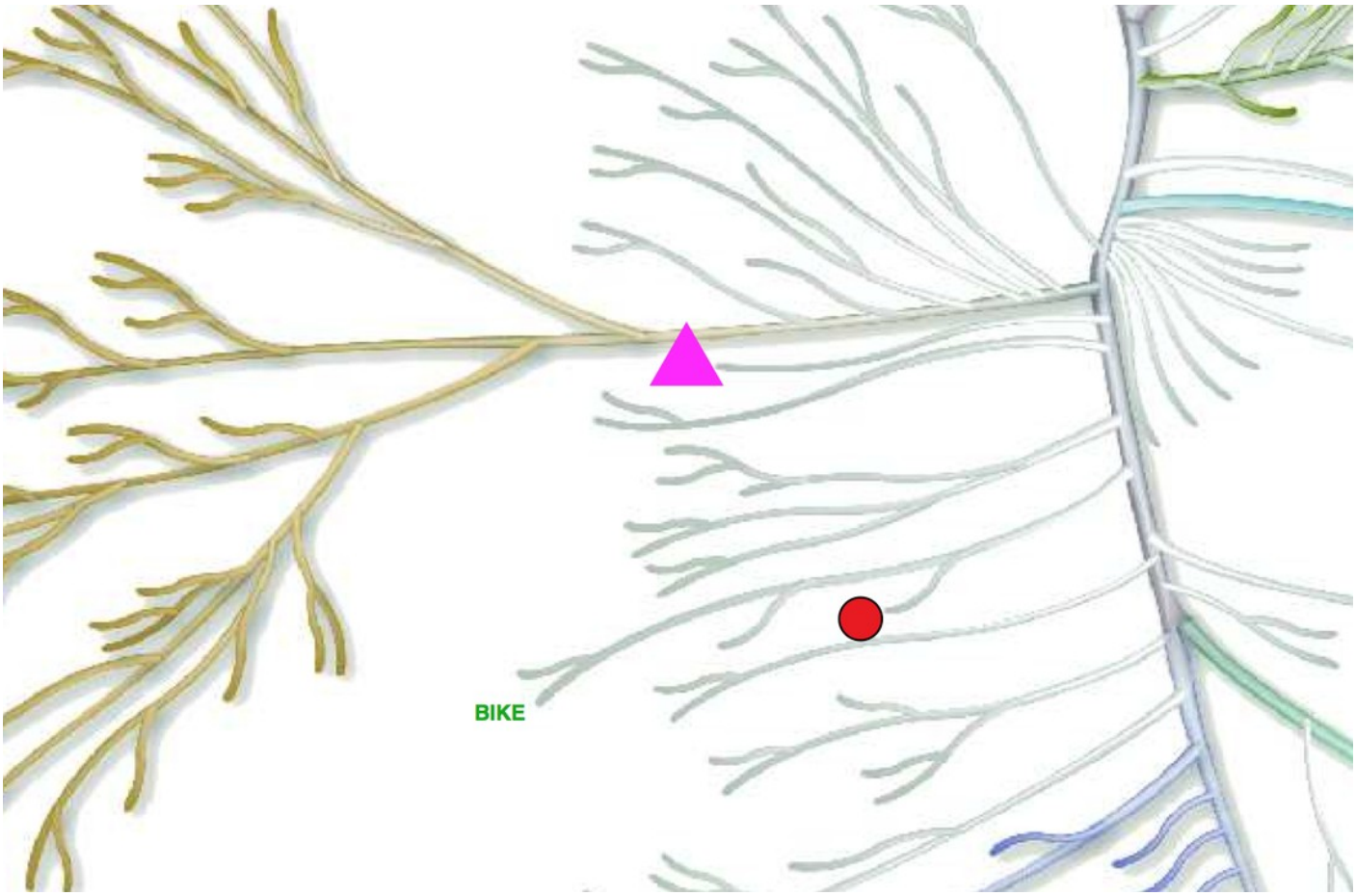

Figure 3

KR annotation legend

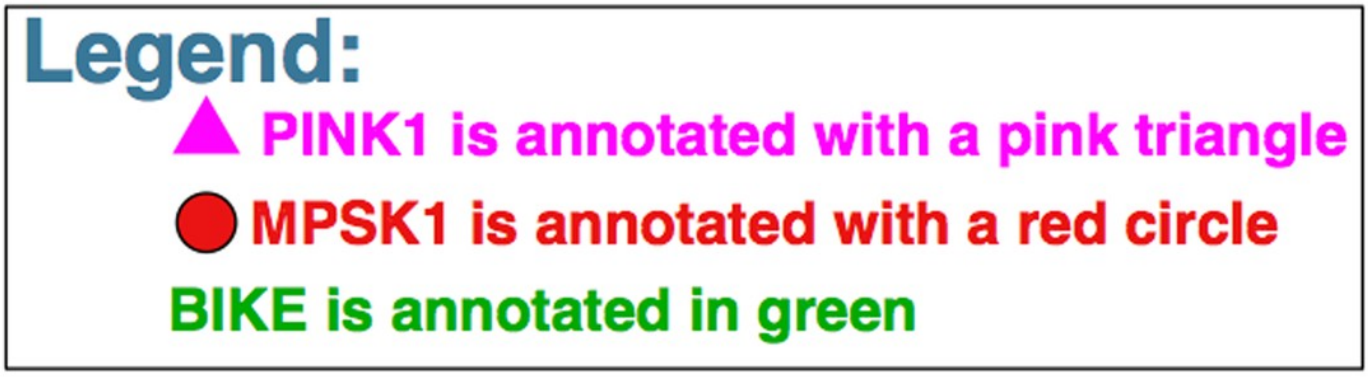

Figure 4

Binding affinities for Sunitinib

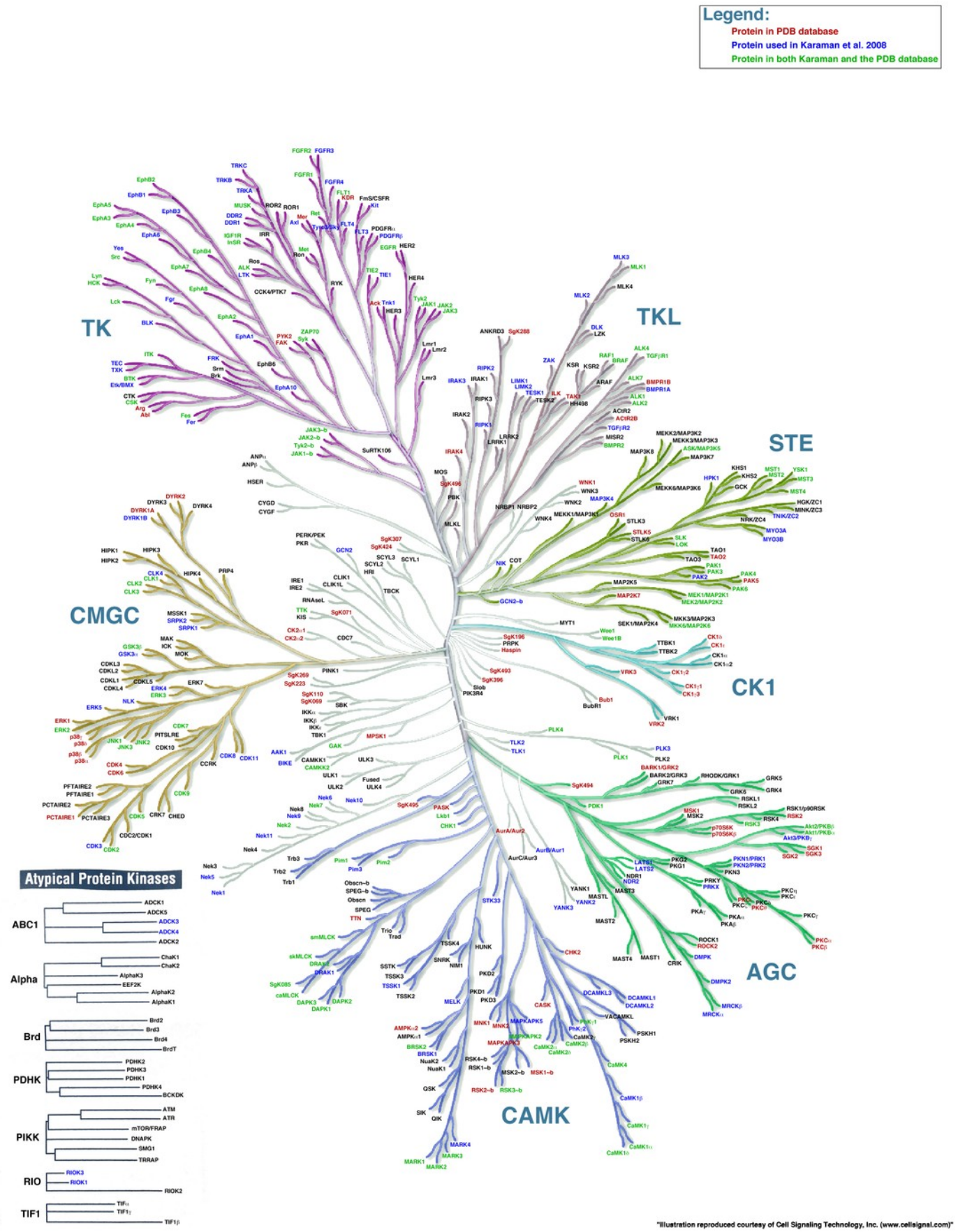

Figure 5

KR image showing kinases in PDB and tested by Karaman *et al.*