

New Developments in ModelDB

Thomas M. Morse¹, Rixin Wang², Luis Marenco², N. Ted Carnevale¹, Michael L. Hines¹, Gordon M. Shepherd¹.

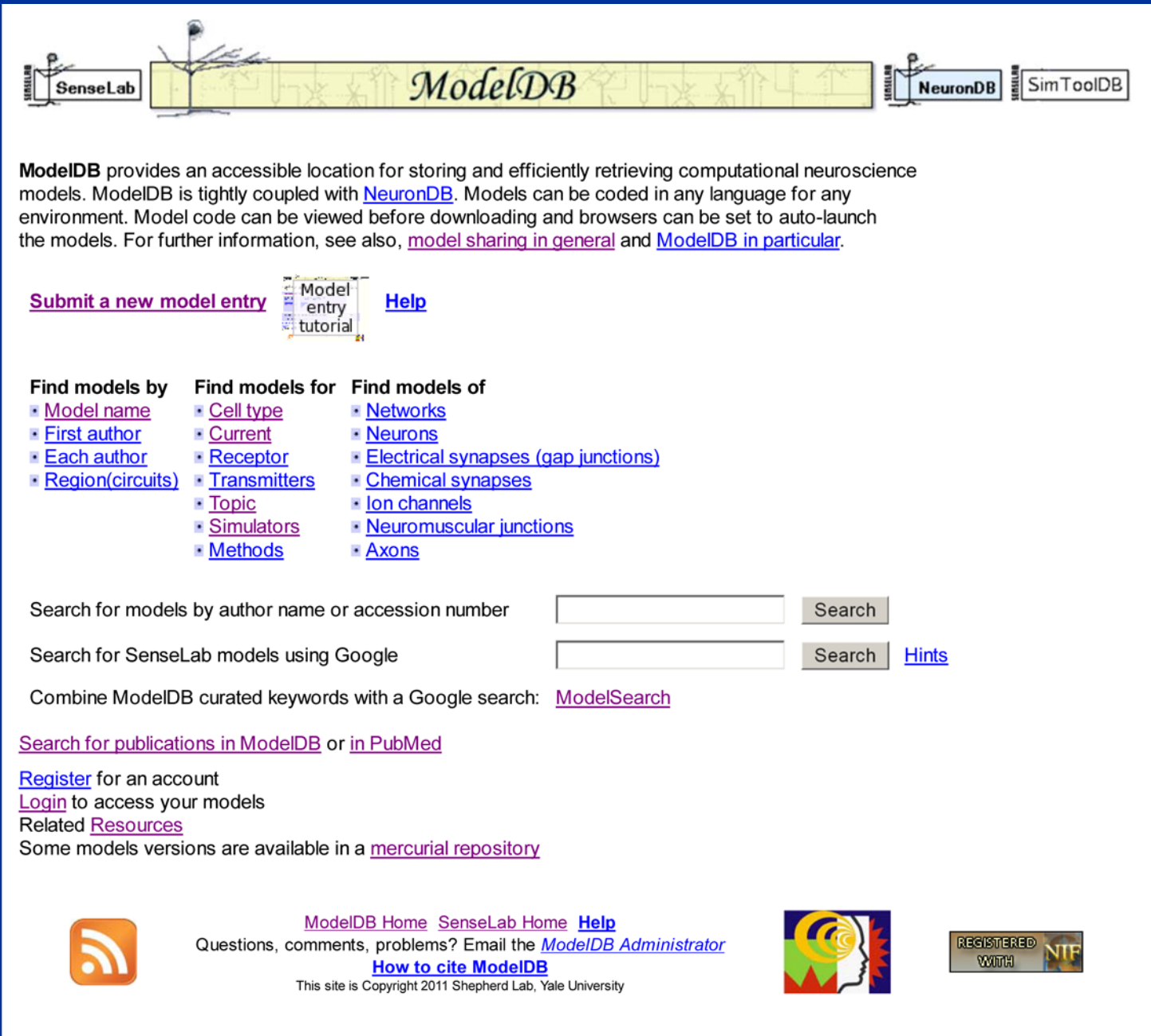
¹Department of Neurobiology, ²IYale Center for Medical Informatics, Yale University School of Medicine, New Haven, CT, USA. Contact e-mail: tom.morse@yale.edu

Abstract:

In the 12 months ending 10/31/2011, ModelDB grew by an additional 75 entries: 24 networks, 34 neurons or other electrically excitable cells, and 17 subcellular (dendrite, synapse, axon, channel) or methods models. This has led us to develop new tools to help ModelDB users probe the > 660 models now in the database. Our latest is ModelSearch, a search engine that combines an open-ended Google search with the precise control of ModelDB curated keywords. It enables users to search for models that contain noncurated terms and restrict results to those that match curated keywords. Noncurated terms are entered into a Google search box, and curated terms are selected from pop-up lists of ModelDB keywords. This search engine was developed in response to external requests, most of which were to find Neuron simulator models with constraints. Example constraints were network models and python code, or models that contained integrate and fire neurons. Each of these constraints contained uncurated terms (python Neuron code, integrate and fire neurons). Google alone however generates too many false positive results from the term "Neuron," which often signals the presence of a particular cell type in a model rather than the Neuron simulator. Tutorial examples will be presented in a live demonstration.

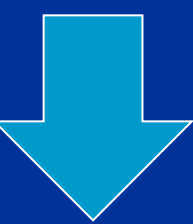
Introduction:

For each tutorial search first select the ModelSearch link from the ModelDB home page:



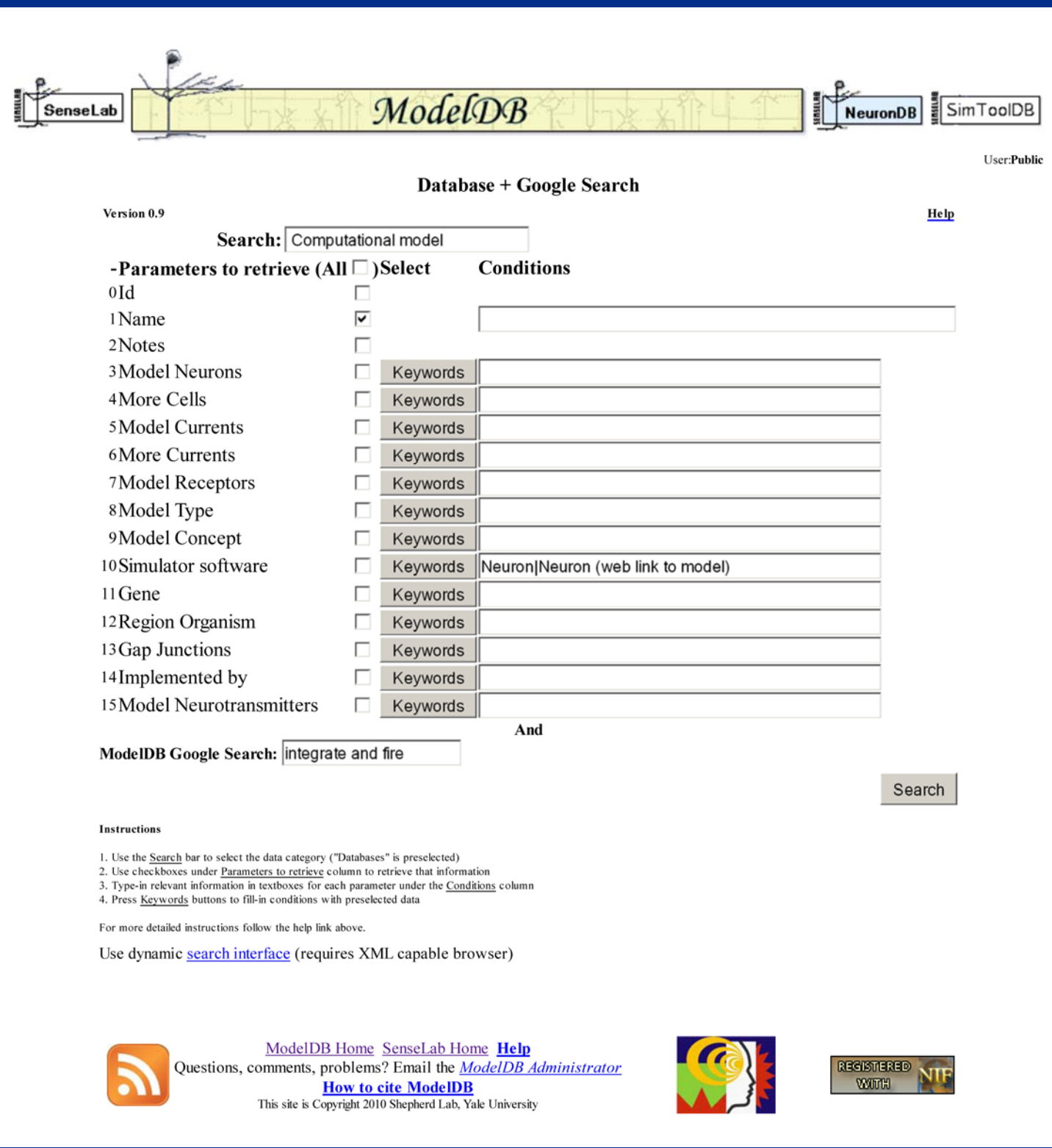
Example 1:

Search for NEURON simulator models which contain both python code and are network models. In the ModelSearch page shown below click the Keyword button next to Model Type and select network in the pop up list and close the box. Similarly select the Neuron simulator and Neuron simulator (web link to model) options by clicking on the Keywords button next to Simulators. type in python and press enter or click the search button.



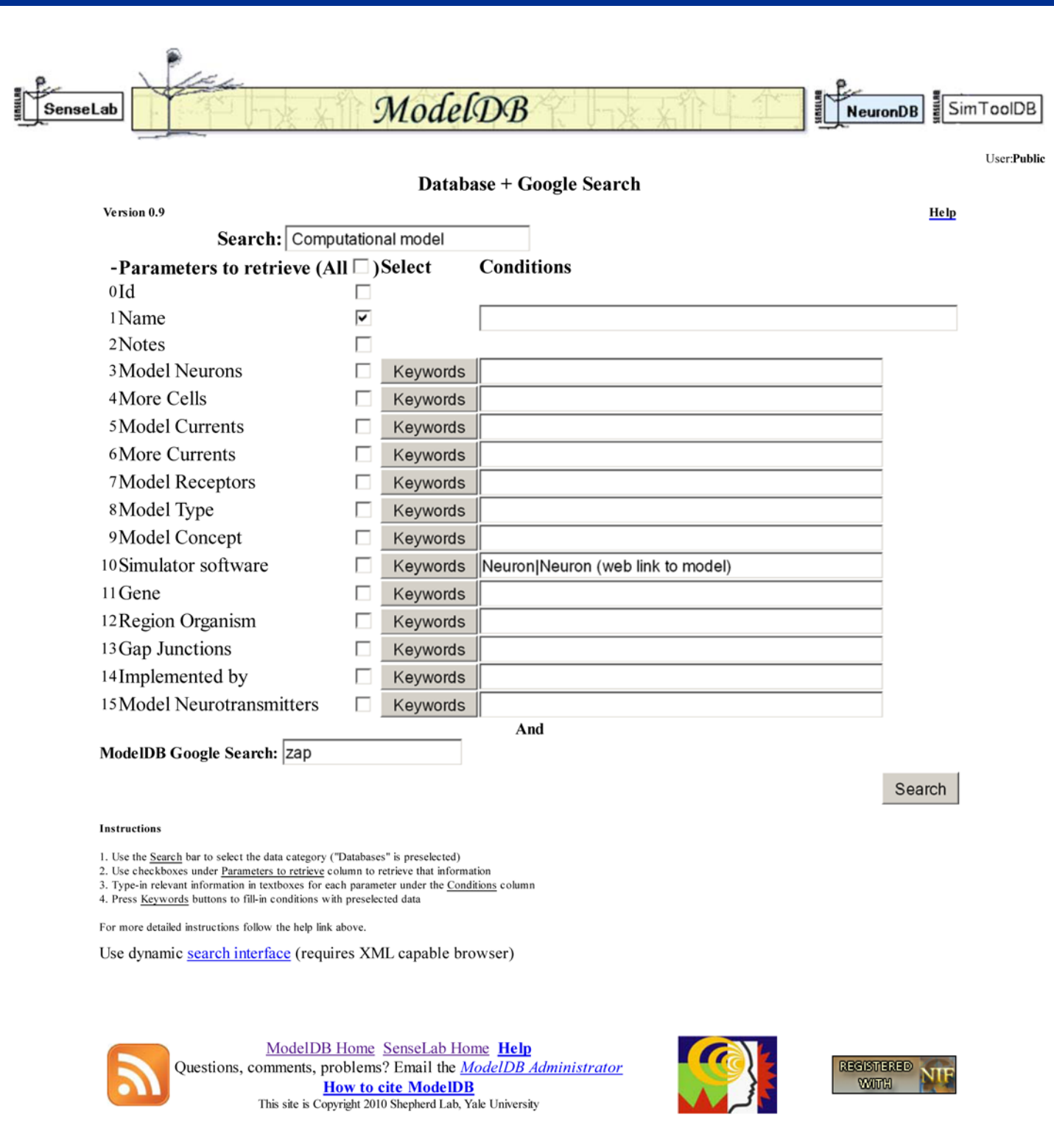
Example 2:

Search for NEURON simulator models which also contain integrate and fire neurons. In the ModelSearch form again select the NEURON simulator. Type "integrate and fire" into the Google search box and press enter.



Example 3:

Search for Neuron simulator models that contain the zap stimulus. The zap stimulus is an injected current that sweeps over a range of frequencies, e.g. from 0 to 100 Hz. As before select the NEURON simulator and enter zap into the Google search box and press enter.



The search results are collected into groups under a link to each model entry. The file and path to the file and google paragraph are provided.

Technical notes: How ModelSearch works:

ModelSearch works by first storing the result from searching ModelDB's database for the Keywords selected by the modeler, and then storing the result of the entire Google search. These stored results are then combined: Only those model entries are retained in the search results which were found to be present in both the attributes (ModelDB curated keywords) and google searches. The reported results have the hits collected under each model. The google links are transformed into a path to the file within the model entries file archive and the google paragraph is provided below the link.