Reasoning tool proof-of-concept

In November 2017, applicants invited for a virtual interview will present:

1. a 10-month plan to build a reasoning tool and
2. their proof-of-concept software

The proof-of-concept software will serve as preliminary data to give the review panel a sense of the approach that will be taken, the quality of the software and speed with which development is able to progress.

Proof-of-concept expectations

A key feature of any proposed reasoning tool is the ability to automate the process of answering a question. Currently, a question such as "*how does this drug induce a clinical outcome?"* could be answered by manually identifying the relevant data sources and data types followed by manually constructing API calls or SQL queries to retrieve the appropriate subsets of data. This would be followed by joining one or more datasets and then finally presenting the data or some computed output as the solution. Examples of such manually-constructed workflows can be found in the form of Python notebooks developed as part of the current project [link and link].

Proof-of-concept software is expected to provide reasonable and relevant answers for specific classes of research questions (see proof-of-concept questions below). Note that a specific input format for the reasoning tool is not being prescribed. While the questions, such as above, will be specified in plain English, the reasoning tool may parse it in any way that is appropriate (including manually), though automated and semi-automated approaches will be preferred. Operation of proof-of-concept software may require the assistance of the applicant to initiate its operation, but must be transparent to NIH staff. Once initiated, software is expected to complete its tasks without further operator input within a working period of twelve hours on cloud computing instances that will be provided by NCATS. The initial proof-of-concept may implement access to novel resources and algorithms, but development after November must be interoperable with existing components of the Translator (see *Further Information* below).

Prior to its demonstration in the virtual presentation, applicants are strongly encouraged to provide the objective review panel access to documentation or source code for those parts of the software responsible for substantive aspects of its operation and document sources of knowledge used. Accessibility of code will be a consideration of the evaluation (see *Collaboration, Sharing, and Intellectual Property Expectations* below).