BIOINFORMATICS APPLICATIONS NOTE Vol. 24 no. 5 2008, pages 738–740
doi:10.1093/bioinformatics/btm559

Databases and ontologies

KUTE-BASE: storing, downloading and exporting MIAME-compliant microarray experiments in minutes rather than hours

Sorin Draghici[1,*], Adi L. Tarca[1,2,3], Longfei Yu[1], Stephen Ethier[3] and Roberto Romero[2]

[1]Department of Computer Science, Wayne State University, 431 State Hall, Detroit, MI 48202, [2]Perinatology Research Branch-NIH/NICHD, 4 Brush, 3990 John R and [3]Barbara Ann Karmanos Cancer Institute, 110 Warren Avenue Detroit, MI 48201, USA

Received and revised on October 14, 2007; accepted on November 2, 2007

Advance Access publication November 17, 2007

Associate Editor: John Quackenbush

ABSTRACT

Motivation: The BioArray Software Environment (BASE) is a very popular MIAME-compliant, web-based microarray data repository. However in BASE, like in most other microarray data repositories, the experiment annotation and raw data uploading can be very timeconsuming, especially for large microarray experiments.

Results: We developed KUTE (Karmanos Universal daTabase for microarray Experiments), as a plug-in for BASE 2.0 that addresses these issues. KUTE provides an automatic experiment annotation feature and a completely redesigned data work-flow that dramatically reduce the human–computer interaction time. For instance, in BASE 2.0 a typical Affymetrix experiment involving 100 arrays required 4 h 30 min of user interaction time forexperiment annotation, and 45 min for data upload/download. In contrast, for the same experiment, KUTE required only 28 min of user interaction time for experiment annotation, and 3.3 min for data upload/download.

Availability: http://vortex.cs.wayne.edu/kute/index.html

Contact: sod@cs.wayne.edu

1 INTRODUCTION

The BioArray Software Environment (BASE) has become a very popular repository for microarray data, as suggested by the large number of installations world wide (Saal *et al*., 2002). Our experience in managing medium to large microarray studies, revealed that although BASE is a very flexible data management system, certain aspects of it could still be improved. In our study of BASE 1.x/2.0, we identified a number of issues that can make it inefficient and time consuming. These issues have been addressed with a work-flow redesign, as well as a number of other modifications and additions, engineered together as a plug-in for the existing BASE 2.0 (henceforth BASE). These modifications led to significant improvements in the overall efficiency of the system. The issues described in this article are still pertinent even for the latest release of BASE that is 2.4.

*To whom correspondence should be addressed.

2 ENHANCEMENTS PROVIDED BY KUTE-BASE

2.1 Automatic experiment annotation

A first area in which improvements can be madeis related to the experiment annotation. In BASE 2.0, assuming an ideal framework in which all protocols, as well as all hardware and software information are already available in the system, annotating a single-channel Affymetrix experiment involving 50 arrays requires over 2 h of human–computer interaction.

This is because for every single arraythe user has to create and annotate items such as Biosource, Sample, Extract, Labeled Extract, Hybridization, Scan and Raw bioassay. Typical fields that need to be filled for each of these seven items are: dates, protocols, name of hardware and software used, etc. Overall, there are approximately 35 fields to be filled for every array.

The user is required to specify the content for all fields even though there might be a lot of redundancy between items of the same type (e.g. between arrays, samples, etc.). In fact, a sound experiment design would *require* the researcher to minimize the variability introduced by nuisance factors, such as the variability introduced by using different protocols, in order to maximize the statistical power. Thus, in most experiments, large batches of arrays are very likely to share the same protocols for mRNA extraction, labeling, hybridization, etc. BASE 2.4 takes advantage of this redundancy by having some default values for each experiment. This helps but more can be done.

KUTE-BASE takes advantage of this redundancy by automatically creating most of the necessary annotation items. This is done by: (i) assuming a one-to-one correspondence between the experiment items, (i.e. Biosource-1 will be linked by default to Sample-1, Extract-1, etc.) and (ii) using a naming convention. The assumption here is that all items in the microarray processing pipeline (e.g. Samples, Extracts, Labeled extracts, Hybridizations,Scans and Raw bioassays) that are associated with a given experimental unit share the same name in addition to their extensions. For example, if the name of a

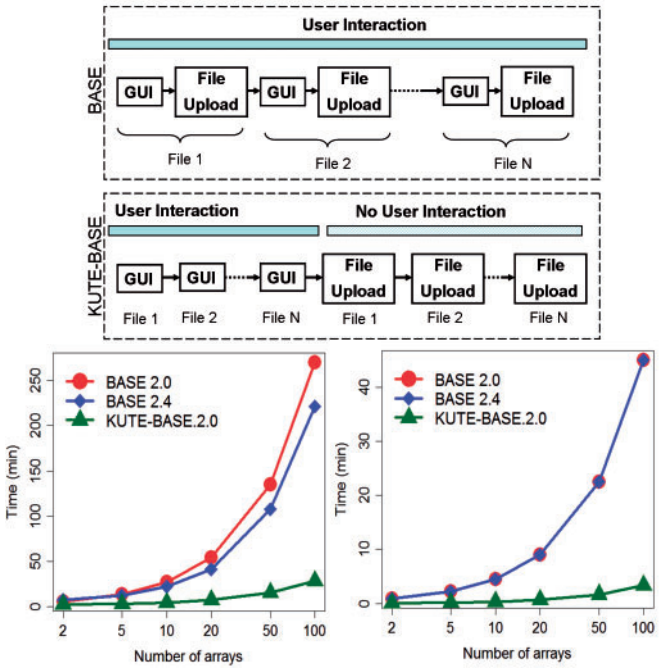

Fig. 1. The work-flow redesign in KUTE vs BASE (top) and its effect on the user-interaction time for experiment annotation (bottom-left), and data upload/download (bottom-right). The average transfer speed reported by BASE was 800KB/s.

sample is *MYO*, then, when users annotate experiments by using the *Kute-Express* feature, the system will assign default names, *MYO.e1*, for Extract, *MYO.le1* for Labeled extract, *MYO.h1* for Hybridization, *MYO.s1* for Scan and *MYO.rb1* for Raw bioassay. Note that these conventions do not prevent the users from subsequently assigning arbitrary names, preserving therefore the flexibility offered by the original BASE.

2.2 Redesign of data upload/download work-flow

Another important factor affecting the raw data upload in BASE is the inter-twining between the human interaction and the data transfer. This is especially important for microarray technologies producing a large raw data file for each array (e.g. Affymetrix). As shown in Figure 1, BASE 2.x requires the user to specify a file name, after which the respective file is uploaded. The upload of such a file may require between tens of seconds and minutes, depending on the system and connection speed. This amount of waiting time is not sufficient for the user to switch to some other task during any one particular file upload. However, when cumulated over the entire data set involving hundreds of arrays, this waiting time can sum up to several hours.

In KUTE, this work-flow has been redesigned as shown in Figure 1. Here, the user interaction with the system is disentangled from the file transfers. The user interacts with the computer for a few minutes only, providing the file names, after which the tens or hundreds of files necessary can be automatically uploaded into BASE system without further user intervention. Raw data download may also be needed to perform various analysis using other software tools that are not integrated with BASE. Instead of downloading each data file one by one, KUTE allows to download all raw data into a single archive (zip) file. This feature is currently implemented for Affymetrix data only, but can be extended to other platforms as well.

3 RESULTS

KUTE implements both the automatic experiment annotation as well as the batch upload/download of raw data files, minimizing the human–computer interaction. *KUTE-Express* is a feature that allows the user to specify the names of the samples in the Sample section of the GUI. The system generates all required items (Samples, Extracts, etc), and annotates them with the default values. If the Affymetrix platform is used, the *Affymetrix File Batch Uploader* is a better choice, since the user can specify the CEL files to upload after the experiment annotation. The sample names will be directly derived from the.CEL file names. Unlike the conventional BASE 2.x work-flow, this process requires only minimum human intervention that saves a considerable amount of time.

The effect of using the KUTE features on the user interaction time is also shown in Figure 1. The user interaction time is defined as the time a user is requiredto spend in front of the computer. The overall interaction with the database was split into three phases: experiment annotation, data upload and data download. The experiment annotation phase comprises all steps necessary to build the experiment structure (create and annotate samples, extracts, labeled extracts, etc.). The data upload phase includes browsing for the files in the local file system and associating them with the appropriate entries in the database. The data download phase includes the time necessary to navigate the database in order to specify which raw data files one wishes to download from the database tothe local machine. Both upload and download of raw data files associated to the raw bioassays, require the same user interaction: a file selection and a confirmation step. Hence, there is only one value reported for the upload/download time. The separation of the user interaction from the file upload/download process and the automatic experiment annotation dramatically reduced the human–computer interaction time. For instance, in BASE, an experiment involving 100 arrays processed with the same sample extraction, sample preparation, scanning and hybridization protocols required 4 h 30 min of user interaction time for experiment annotation, 45 min for data upload/download. In contrast, for the same experiment KUTE required only 2.2 min of user interaction time for experiment annotation and 3.5 min for data upload/download.

The substantial differences are explained by the very different work-flows as well asby the addition of the automatic experiment annotation feature. In BASE 2.0, all processing is completed at the end of each phase but the user is forced to remain in front of the computer for the entire duration (many hours in most cases). In KUTE, the user is required to remain in front of the computer only as long as necessary to provide all required information (minutes in most cases) but not all processing is completed when the user leaves the

machine. Even though the computer continues to do a lot of background processing long after the user is gone, the most expensive resource—the highly qualified human—is now available for other tasks.

Conflict of Interest: none declared.

ACKNOWLEDGEMENTS

This work has been partially supported by the following grants: NSF DBI-0234806 and CCF-0438970, NIH 1R01HG003491, 1U01CA117478, 1R21CA100740, 1R01NS045207, 5R21EB 000990 and NCI 2P30 CA022453. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF or NIH. This research was supported, in part, by the Intramural Research Program of the National Institute of Child Health and Human Development, NIH/DHHS. This research was supported, in part, by the Intramural Research Program of the National Institute of Child Health and Human Development, NIH/DHHS.

REFERENCE

Saal,L.H. *et al*. (2002) BioArray Software Environment (BASE): a platform for comprehensive management and analysis of microarray data. *Genome Biol*, **3**, 1465–6914.