Easy and user-friendly workflow management based on the data-morphing concept

Stéphane GRAZIANI
ISoft

.IQuick description

Based on the data-morphing, an innovating technology, ISoft Amadea Biopack allows biologists and physicians to define and publish on the web, without programming, their own pipeline of data processing for transcriptome, genome, proteome, etc. The platform offers simple and fast analysis modules using and linking instantaneously all algorithms and biological data banks with their own data. Data morphing homogeneous environment and engine performances open new perspectives to the bio-medical professionals that are able to test much more hypotheses and new protocols they would not have imagined before.

.IIIntroduction

In the last 10 years, with the emergence of large scale experimentation technologies (genome, transcriptome, proteome data, etc…) the databases providing bio-medical information have impressively grown and multiplied, so that they now represent a huge volume of data which format is furthermore very heterogeneous and trying to fit to a lot of quickly evolving scientific concepts. This quick evolution is due to the nature of bio-medical sciences, that every day discover new mechanisms in the biological complexes: these mechanisms need to be represented in the existing or in new databases, that were generally not primarily designed to integrate them, leading to the multiplication of data formats.

Even though a lot of data extraction and visualization tools are available to the bio-medical professional, he his in front of a new technological problem: instead of having a few genes (or other entities) of interest, bio-medical laboratories are in front of several thousands of experimental results that they can no more analyze "by hand", i.e. using the available interactive graphical interfaces: in front of the urgent need to automate these analyses, and to design automatic results filtering tools that provide the end-user with a volume of information he is able to handle and to interpret, bio-informaticians have started to create scripts in languages like perl, php, python, …, that were automating the execution of data extraction and analysis tools: now, they work towards the implementation of workflow management systems that allow full automation of what a bio-medical professional does manually when chaining several types of analysis and of information retrieval about intermediate results of these analysis steps, in order to reveal the information hidden in the huge input data. Several efforts, based on scripts languages, or providing graphical interfaces to help the design of a workflow, make it possible to define such workflows, by defining and executing the different tools that were previously adapted for their integration.

Here, we present a completely new approach of the way workflows can be designed, based on a specific data transformation and application pipelining technology, called "data-morphing". We have developed and tested this approach in research projects like the successful IST project “HKIS” (partially sponsored by the European Commission under number IST-2001-38153), aimed at designing an interactive and integrated platform for bio-medical professionals studying cancer or the French ExtraPloDoc project for automated biological bibliography analysis. As opposed to the other approaches, the data-morphing way takes into account the fact

that any workflow, as a way to analyze data, cannot be distinguished from the dataflow it generates, and, as such, provides the way to efficiently manage this dataflow, in an homogeneous manner, thus allowing any data source to contribute at any point of the analysis. The difference with existing tools is that this is not a query-based data integration, but rather a data-transformation-based one, which means that at any time, the whole datasets can be used to compute complex indicators from several databases together, instead of being limited to the results of distant database queries or simple biological data indexation systems like SRS. For example, it would allow to work on all combinations of Gene-Ontology annotations and known tertiary peptide folds in order to filter using combined functional and structural criteria.

After having described the principles of the data-morphing technology, we will explain how Amadea Biopack(our data-morphingplatform) allows to efficiently design integrated bio-medical data analysis tools thanks to its power of expression, and guaranties reproducibility, reusability and traceability of designed workflows. After detailing the different ways the problem of data heterogeneity is currently addressed, we will then describe our way to link together any bio-medical tools and data sources in an homogeneous manner, thanks to the concept of data heap and how data sources are made instantaneously available at any time of the analysis.

Among the numerous issues our platform can deal with, we can quote the following two for example, which will be used below to illustrate some points:

1) From a series of transcriptome chips data, find a list of metabolic functions that could discriminate the differentially expressed genes. A process giving a response to this question (in 1 minute for 16 chips) was designed in 1 hour.
2) Having the sequences and sometimes the identifiers of genes of interest, retrieve information about the known or predicted structure of the associated proteins, and try to discover some folds common to this list of genes. Based on existing operators, 30 minutes were sufficient to develop a process requiring only 5 minutes to answer this question. To add a custom protein structure analysis tool, a few minutes to 2 hours were needed, depending on the tool complexity.

.IIIData morphing

The data-morphing technology is for data tables the equivalent of image morphing for pictures: it describes data transformation experiments as successions of single steps, called "operators" of data transformations. These single steps can be very simple, like selection / creation of fields (columns) or records (rows), or more complex, giving access to powerful tools and efficient acting on table contents or structure : aggregation, binarisation, verticalisation, string splitting, string comparison, etc…. The design of complex data transformations is easy thanks to an interactive graphical editor (the studio : see Figure 1). But this graphical display is not only an interface used to simplify the creation of an integration script that would, after its full design, be run: on the contrary, each single step is functional and executed in real-time as soon as it is added, thus allowing an incremental bottom-up design: this way, “specifying is building”. This crucial difference with other programming-based approaches (Perl, Xquery, …) avoids important time loss due to specification defaults. Furthermore, at any time, intermediate results of each transformation step are always accessible, and when viewing an output that was not yet computed (or one step upwards of which has changed), all steps upwards are, if needed, recomputed as to guarantee reproducibility and reusability of the process designed this way.

This bottom-up approach to build data analysis pipelines is largely facilitated by

the self-extension capacity of the data-morphing engine, allowing the creation of a new building block ("operator") corresponding to a data transformation, with an input and an output. This way, it is possible to define different sets of operators, depending on the "level" of the application, to be defined: For example in the second issue, we chain *data access* operators with specialized *protein structure analysis* and *sequence comparison* operators, and finally we apply an operator to do a *statistical test* to obtain the discriminating peptide folds. All this can build a "Gene Structure analysis" operator that can be used as a new building block, etc.

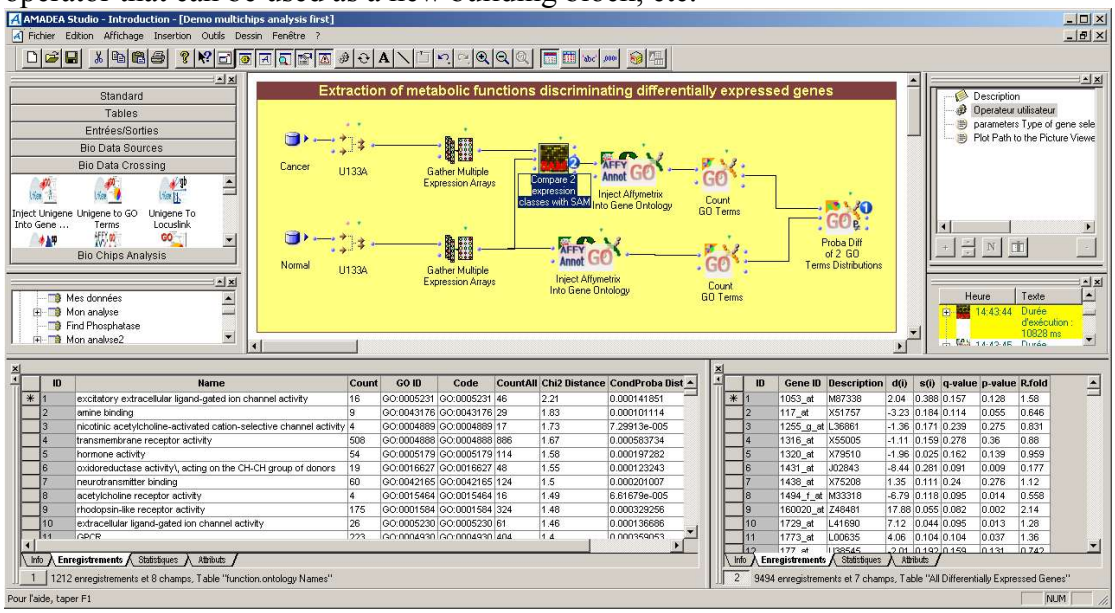

Figure 1: One response of issue 1, in Amadea Biopack Workflow Development Studio (see the middle frame): Starting (left) from expression data of normal and tumor tissues, normalize the data, obtain a list of differentially expressed genes by calling SAM, obtain from Gene Ontology the list of metabolic functions associated to these genes, and get the functions best discriminating these genes using a statistical test (right) : The bottom grids give the results at any point of the graph.

It is of course possible to adopt a more current top-down approach: from conception to implementation. Generally, data-morphing composite building-blocks are built bottom-up, while final scenarios use them in a top-down manner.

Finally, there is a fundamental difference between this data-morphing approach and usual database- and/or XML-based applications: effectiveness. Databases (and SQL, Xpath, XQuery) are designed and optimized to store data and quickly retrieve it, but not to perform complex data transformations. XML, very well suited to generic information exchanges (like web-services), was not designed to be used as intermediate data results in large data processing application, because it implies very large files (10-100 times larger than an optimized representation), long to read and interpret at any step of the analysis. On the contrary, the data-morphing technology uses its own data representation, optimized for these complex data analyses: as such it is 10-100 times faster than SQL queries , perl applications or XML-transformation-based workflows. For instance, the ExtraPloDocs bibliography analysis project showed that the same workflow could be executed in some hours with XML-based tools integration, and some minutes with our data-morphing implementation. The workflow design time was also much longer with the XML approach. This efficiency of data-morphing allows simplifying incremental designs, as results are obtained in real-time, and several sets of parameters can be tested successively at no additional cost.

The originality of the data-morphing for chaining tasks is that the difficulty is no more data formatting, which is very straight-forward, but rather the "real" experimental problem: "how shall the tool be tuned to give the best results in a given context?".

.IVWorkflows using data-morphing

Since it is possible to embed calls to external applications into single operators of the data-morphing chain, we have designed a very user-friendly and interactive workflow definition platform, that allows to create complex integrative analysis protocols in a seamless manner, linking different tools in the same pipeline.

As mentioned in the introduction, the design of a workflow is tightly bound to the design of the associated dataflow: without a dataflow, no workflow can run because of a lack of data to treat. Without a workflow, the data cannot be modified or analyzed. One of the consequences of this work-/dataflow duality is that the representation and the format of the data circulating all along a workflow is almost as important as the different treatments it is undergoing. Thus, workflows as they can be defined on our platform (to be compared to the generic definition given on the NETTAB web site : "a computerized facilitation or automation of a business process, in whole or part") include data-transformation capabilities and correspond to the chaining of tasks, each of which, in an homogeneous data access environment, has an effect on this environment that will determine the next task to launch. In this definition, the environment is clearly very important, and corresponds in fact to the data flow, implemented as a data heap (see below) : the state of the environment will determine the result of the data analysis process. Other workflow management systems often take the dataflow into account by using the XML/XQuery standard. The problem with this approach is that it is quite long to specify the correct XML format used to allow cross-communication of different tools, and each time a new tool needs to be integrated, a new "bridge" with other tools needs to be made.

In our approach, we take full advantage of the data-morphing simplicity and power of expression to easily connect any integrated tool with any other tool, or with any biological data source. This transparent connection is due to the fact that any data (sources, tools inputs or results, properties, …) in the data-morphing pipe-line, is represented in the same and most simple way: tables in the data heap (see below) and tools connection is reduced to simple data-morphing steps, which generally results into a trivial task.

.1 Integrating a new tool: basic principles

We describe here the integration of a locally executed tool, but the same principles are applied to distributed / web based tools since data-morphing operators can perform web queries.

- First step, the embedding phase itself. The principles of embedding are very simple: the input table is exported in the right format for the tool to embed, then the tool is executed with an automatically computed command-line, and finally, the files produced by this execution are imported as result tables into the data-morphing pipeline. Our tools-embedding-toolkit including an operator that is sufficient to embed the majority of external tools, greatly facilitates this task. Standard output and error streams can also be transmitted to the pipeline.

Export
Of data prepared for the tool

Exécution
Of the tool

Import
Of the data generated by the tool

- The second phase is optional and needed only when the resulting table has a complex format: in this case, the result is transformed into simple tables, which information will be easy to exploit in the pipe-line.

Eg, integrating Blast means to create FASTA files (query and/or databases), launch Blast, import and simplify the result with simple data-morphing steps.

Of course, a plugin architecture also provides bioinformaticians with an API offering tightened integration of their tools, but the previously described loose integration has the great advantage to avoid any programming step to the user.

.2 A new way to meet the data heterogeneity challenge: the data heap

As we described in the introduction, the quick evolution of bioinformatics databases, tools and biological concepts poses the challenge of being able to access to and link these data together, even when they change in large proportions.

The classical ways to solve this problem is, either through mediators connected to distant databases, or building a large data warehouse, to try and build a data model able to store or at least to represent all the concepts one wants to use together: the semantics of the data is integrated in the structure used to store this data, because this structure somehow presupposes the future use of the data. This works well, provided that the time to adapt the wrappers of the mediator or the structure of the database is low. But biological data sources often undergo very deep structural changes, that cannot be quickly integrated into such models, and frequently imply the redesign of the models: with such an approach, being permanently connected to biological data formats stays a complicated challenge and needs important resources.

The data-morphing technology introduces a new way to handle bio-informatics data: instead of trying to structure the data a priori, we prefer to completely unstructure it, reducing all data sources to the set of all "atomes" of information, gathered in a set of tables called the "data heap". This data heap is the same format as the basic format of all intermediate results of the data-morphing engine, so that accessing any bio-medical information from the data heap is instantaneous, whatever the complexity and the volume of the data. This way, we do not need any specific modelization of data and the semantics is simply given at the step where the data is used, which is possible because of the rapidity of the data-morphing engine: the bio-medical professional can consider and use the information he needs in the sense he chooses, either through "standard" data-linking operators or defining new ones.

This simplified data representation based on data tables with no structural constraints and no pre-defined links allows unprecedented performances of the data-morphing engine, thanks to optimization mechanisms like data caching and specialized memory management procedures.

It is noticeable that this approach also allows to give instantaneously access to any bio-medical data sources, either public (Swissprot, genbank, PDB, …) or private (Affymetrix chips definition files, proprietary experimental data sources, …).

The advantage is that, when data sources undergo changes, one simply needs to change the processes involving the modifications, without having to modify anything

else: thanks to the data-morphing power of expression, this is generally only a matter of hours. For example, when the NCBI decided to replace Locuslink (that is by definition at the heart of a majority of data linking operators) by Entrez Gene, one person adapted all data sources and crossing operators in half a day.

.VImmediate publication on a web server

Once a data analysis workflow has been designed and tested, it is possible to make it immediately accessible through interactive web pages on an http server, simply by defining the list of intermediate points of the transformations that should be visualized as tables, charts, raw HTML, etc. and the list of variables that may be changed by the user (text fields, combo-boxes, checkboxes, etc.). This way, any workflow can be deployed on a server in minutes, and the generated web pages can have a good-looking aspect thanks to a template mechanism, allowing the user to design his own pages with any HTML editor, independently from data-flow considerations.

Thanks to this web interface features, the application can be made accessible to many people who do not need nor want to understand the technical details of the workflow, but need to have an interactive access to it, in order to test several parameters / experimental conditions and rapidly visualize the results for each test configuration.

.VIAn efficient and easy-to-use workflow management system

Thus, putting all these points together, we present here an interactive, user-friendly, powerful, evolutive and adaptable workflow management environment that provides seamless integration capabilities and a simple response to the problem of data and tools heterogeneity and constant evolution. This tool, by its graphical design and the bottom-up approach, is a strong support for methodologies of analysis.

Moreover, once the workflow was designed, other modules provide immediate creation of web-based interfaces giving to others access to the workflows, so that they can pilot the different steps of these workflows from their navigator.

It is also noteworthy that the interactivity of the tool makes it a very good communication platform around which several different experts can work together, quickly prototyping an application that they will later refine, each one implementing the modules corresponding to his know-how: in front of it, it is easy to test hypotheses in real-time and to envisage new ways to explore.

Thus, as demonstrated in the HKIS project on cancer-related bio-medical analyses, our platform is a breakthrough in the bioinformatics field, because it gives a solution to the problem of data and tools heterogeneity and volume, allowing the bio-medical professional to immediately access to any information or treatment he needs, and to use them to build new analyses workflows in an incremental manner.