



HAL
open science

Bioinformatics applications discovery and composition with the Mobyly suite and MobylyNet

Hervé Ménager, Vivek Gopalan, Bertrand Néron, Sandrine Larroudé, Julien Maupetit, Adrien Saladin, Pierre Tufféry, Yentram Huyen, Bernard Caudron

► **To cite this version:**

Hervé Ménager, Vivek Gopalan, Bertrand Néron, Sandrine Larroudé, Julien Maupetit, et al.. Bioinformatics applications discovery and composition with the Mobyly suite and MobylyNet. *Resource Discovery*, 6799, Springer Berlin Heidelberg, pp.11-22, 2012, Lecture Notes in Computer Science, <10.1007/978-3-642-27392-6_2>. <pasteur-03682347>

HAL Id: pasteur-03682347

<https://pasteur.hal.science/pasteur-03682347v1>

Submitted on 10 Feb 2026

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Copyright - All rights reserved

Bioinformatics applications discovery and composition with the Mobyly suite and MobylyNet

Hervé Ménager¹, Vivek Gopalan⁴, Bertrand Néron¹, Sandrine Larroudé¹, Julien Maupetit^{2,3}, Adrien Saladin², Pierre Tufféry^{2,3}, Yentram Huyen⁴, Bernard Caudron¹

¹Groupe Projets et Développements en Bio-informatique, Institut Pasteur
28, rue du Dr Roux, 75724 PARIS Cedex, France

²MTi, INSERM UMR-S 973, Université Paris Diderot (Paris 7), Paris, France

³RPBS, Université Paris Diderot (Paris 7), Paris, France

⁴Bioinformatics and Computational Biosciences Branch, Office of Cyber Infrastructure and Computational Biology, National Institute of Allergy and Infectious Diseases, National Institutes of Health, Bethesda, MD 20892, USA

Abstract. Given the sheer number of existing analysis tools and data sources, defining and running bioinformatics analyses is often challenging. We present here the **Mobyly framework**, a web-based environment to access such resources. It enables the use of local and remote bioinformatics services, seamlessly integrated within a homogeneous and web-accessible environment that focuses on usability. This framework offers the possibility of integrating a wide range of services that span from the traditional server-side command-line based programs to workflows and client-based visualization tools. By abstracting whenever possible the user from the details of syntactic compatibility and providing multiple classification tools, Mobyly provides an efficient web-based solution to run and chain bioinformatics analyses. The **MobylyNet** project promotes an organization that federates a network of small resources that shares its expertise within a community that spans various complementary domains of bioinformatics. Its most tangible expected result is the integration of the bioinformatics tools of the different nodes, providing interoperability, user assistance and quality management between its members.

Keywords: Web services, user interface, bioinformatics, metadata, workflow, service publication

1 Background

Performing bioinformatics analyses requires the selection and combination of tools and data to answer a given scientific question. Many bioinformatics applications are **command-line** only and researchers are often hesitant to use

them based on installation issues and complex command requirements. Some projects such as the EMBOSS package[19] have offered solutions to standardize the syntax and description of command-line based tools. Despite their success, such solutions are limited by their lack of adoption as a development framework by external programmers and by the reluctance of many biologists to use command-line based software. Furthermore, while dedicated communities help with the distribution of bioinformatics packages in specific operating systems[13] that tackle the issue of installation and maintenance of such software, they do not address issues such as user interface complexity.

A common workaround to these problems is to run programs on remote public web servers. However the cost of the development of **custom web interfaces** for bioinformatics programs, as well as maintenance and usability problems[20], reduce significantly the efficiency of this solution. . The rapid development of new tools and their continual improvement with new releases can make it difficult for research laboratories or institutes to choose the most suitable programs for their analyses without the help of computational experts. This can lead researchers to use outdated versions of software, based on their sole familiarity with them.

When considering web-based solutions, the dispersion of the services requires an additional effort to discover the services. Service providers who wish to facilitate the discovery of the resource can, in addition to traditional scientific communications and submission to generic web search engines and directories, consider **domain-specific registration tools**. Existing initiatives have recently focused on providing registries that enable the exploration of public bioinformatics web services such as the EMBRACE[17] and BioCatalogue[3] registries. The discovery in such systems is completely separate from the execution environment that will be used to define and run the analyses. Such systems delegate the solution for compatibility issues between different services (data formats, protocols) to the client layer. Additionally, being focused on public web resources, they cannot offer a solution that spans both public and restricted resources. The BioMOBY framework [24], in addition to a registry that stores the services, provides a standardized protocol to register, discover and run bioinformatics web services, solving the problem of syntactic compatibility by providing a way to describe the expected format of the required and produced data. However, the current registry has serious limitations, such as inconsistent service descriptions (mostly due to its public nature) and a lack of isolation between the logical (concepts) and the physical (formats) levels of the ontology. These limitations hinder the discovery possibilities it provides to client tools such as Remora [4] or MOWServ[14]. These issues, as well as a curation mechanism for the ontologies, have been discussed elsewhere [10].

The use of the metadata provided by web service registries or the publisher itself provides the required information for web-enabled bioinformatics service **clients** to enable the search for the most suitable tools to execute an analysis. Many existing systems have been developed over the last decade to exploit such resources, enabling the execution of remote or local services, either as local clients (e.g., Taverna [16], Kepler [2] or BioSide [9]) or web-based systems them-

selves (e.g., Galaxy [7][8], GenePattern[18]). These client systems offer powerful discovery and execution capabilities. A more detailed comparison of Mobylye with such solutions has already been published[15]. However, none of them succeeds in guiding the user in connecting semantically compatible tools that may require data format conversions.

The **Mobylye framework**[15] aims at providing a usable interface that offers access to many bioinformatics tools within a single integrated environment. It includes a complete set of tools that cover the description, publication, and execution of bioinformatics software in a "full web" environment. Additionally, as of version 1.0, Mobylye and MobylyeNet extend the range of integrated services from traditional web-wrapped programs to workflows and web-enabled client tools such as data viewers. Recognizing that usability of such systems is often limited by factors such as service discovery, format compatibility, or documentation, we designed and improved the the Mobylye system based on on extensive user studies as well as experience from the maintenance of the PISE[11] and RPBS[1] systems over several years.

As an alternative to the numerous problems generated by the use of distributed resources, a traditional approach is to setup centralized resource centers that aggregate all tools and data. However, such a strategy does not scale well with the growing diversity of the methods to operate, which can exceed the skills of the staff attached to such centers. The **MobylyeNet** project is an initiative to create a network of smaller platforms with specific skills. The services are published on a framework that favors interoperability between the sites, enabling the seamless integration of resources to run cross-domain protocols.

In the following section, we describe the service discovery mechanisms in Mobylye. We then discuss the benefits of the MobylyeNet approach over service publication, and conclude with the future work that focuses on improvement of the existing implementation.

2 Mobylye service discovery

The key element in the architecture of Mobylye is the description of the services. It covers multiple aspects of the tool or service, describing (1) the scientific task it performs, (2) how to use it, (3) the user interface of the submission form and the presentation of the results, (4) how to call it. This description, stored for each service as an XML file, provides an abstract description of what it achieves, using both natural language descriptions and controlled vocabularies that provide the required metadata which is exploited to facilitate service discovery. Mobylye provides an integration layer for different types of services, including:

1. Programs: software executables which run on the server side, are callable in a shell command line and return some results, which are stored as files.
2. Workflows: definitions for the coordinated execution of multiple tasks, using different services to process data.
3. Viewers: type-dependent visualization components for the data displayed in the Mobylye Portal. These custom interfaces can incorporate HTML-embeddable

components such as Java or Flash applets of Javascript code. For instance, using viewers, we can automate the inclusion of the Jalview[23] component to visualize multiple alignments wherever it is relevant in the portal, such as the results of multiple alignment programs like ClustalW or MUSCLE.

2.1 Services description and classification

The first access key within the portal to the different services is their classification. This classification is specified by categorization elements which describe the task performed, as well as the software package they belong to and their "type" (command line-based program or workflow, see figure 1). Furthermore, this tree can be searched on a list of relevant description elements which enables to filter it. This double discovery mechanism enables a fine selection of the services. In addition to generating this human-readable interface, these files provide a way to expose the capabilities of the server (i.e., which are the tools that can be used) to remote APIs as well as to provide them as relevant site maps¹ to search engines. From this perspective, they serve as a source to generate a list which is equivalent to what is found within public web registries mentioned in the previous section. This possibility has been used to develop the playMOBY plug-in² that automates the publication of BioMOBY services using the service descriptions provided by Moby.

An aspect that should be noted is the relationship between the service which is published and the available "interfaces": given the fact that many programs do not perform one specific task, but rather offer multiple options that execute different operations, it is more relevant to separate these different tasks into separate program descriptions that may share common elements, but are presented as separate interfaces to the user. For instance, the ClustalW[22] program published on the Moby server of the Institut Pasteur is separated into (1) *clustalw-multialign* that performs sequence multiple alignments, (2) *clustalw-profile* that merges two multiple alignments by profile alignment, and (3) *clustalw-sequence* that sequentially adds the sequences of a profile to the multiple alignment of another profile.

2.2 Compatibility, data formats, chaining and reuse

A most common problem of bioinformatics resources is the availability of data in diverse formats to represent the same information [21]. The multiplicity of these formats, combined with the limited compatibility of the different tools, results in additional difficulties for users in service discovery and workflow composition:

1. The chaining of successive services, either interactive or predefined, is based on the compatibility between the types of an actual or expected result and

¹ see <http://www.sitemaps.org/>

² the playMOBY plug-in is available at <http://lipm-bioinfo.toulouse.inra.fr/biomoby/playmoby/>

The screenshot shows the Moby portal interface. At the top right, there is a user profile for 'hmenager@pasteur.fr (guest)' with options for 'set email', 'sign-in', 'Openid sign-in', 'sign-out', and 'refresh workspace'. The main navigation bar includes 'Welcome', 'Forms', 'Data Bookmarks', 'Jobs', and 'Tutorials'. The left sidebar contains a search bar and a service tree categorized by 'Programs' and 'Workflows'. The 'Programs' list includes services like '3D_structure', 'alignment', 'consensus', 'cons', 'megamerger', 'merger', 'differences', 'formatter', 'multiple', 'pairwise', 'structure', 'Analysis', 'assembly', 'combine', 'database', 'display', 'genetics', 'hem', 'map', 'migration', 'networks', 'networks', 'nucleic', 'pca', 'phylogeny', 'protein', 'recombination', 'sequence', 'structure', 'tests', 'tools', and 'unix'. The 'Workflows' list includes 'hmh_build_search', 'protein_distance_phylogeny', and 'protein_distance_phylogeny'. Below the service tree are sections for 'Data Bookmarks' and 'Jobs'. The 'Jobs' section shows a list of recent jobs with their names and completion times. The main workspace is currently displaying the 'MUSCLE 3.6' form. The form title is 'MUSCLE 3.6' and it includes a description: 'MUSCLE is a program for creating multiple alignments of amino acid or nucleotide sequences.' The form has 'Run' and 'Reset' buttons. It is divided into several sections: 'Inputs options' with a 'Sequences (-in)' section containing 'paste', 'db', and 'upload' buttons and a text area; 'Determining sequence type (-seqtype)' with a dropdown menu set to 'Automatic'; 'Optimization parameters' with fields for 'Maximum number of iterations (-maxiters)' (set to 16), 'Maximum number of trees (-maxtrees)' (set to 1), and 'Maximum time to run in hours (-maxhours)'; 'Find diagonals (faster for similar sequences) (-diag)' with a dropdown set to 'No'; 'The profile scoring function (for protein only) (-log-expectation (-E))' with a dropdown; 'Profile Alignments parameters' with 'paste', 'upload', and 'result' buttons and a text area; 'Profile 2' with 'paste', 'upload', and 'result' buttons and a text area; and 'Output Options' with an 'output format' dropdown set to 'fasta' and a checkbox for 'Preserve input order of sequences in output file (-stable)' set to 'No'.

Fig. 1: Moby Portal overview.

The first part of the left menu is a searchable and configurable service tree. This tree can be configured to sort services based on their classification or on their software package, and to either mix or separate the different types of server-side services it displays (programs and workflows). Additionally, the search box provides a way to filter this tree based on keywords.

The user workspace (data bookmarks and jobs) can be navigated using the two menus located below the services tree.

The central part of the portal is organized using tabs which let users navigate between the different services and their personal workspace. The current view shows the form used to compute a multiple alignment using MUSCLE.

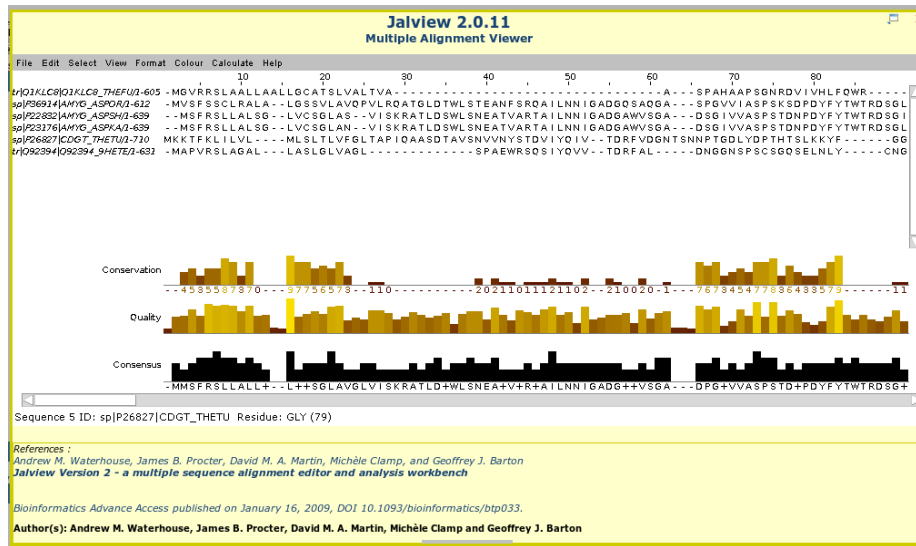


Fig 3: **Jalview viewer**. Any compatible multiple alignment displayed in the portal can be accessed using this applet.

to the start of the next job. Additionally, direct access to a relevant viewer, Jalview, is automatically displayed (see figure 2).

- the **history reload suggestions** (see figure 4) offers the user to load data that have been previously bookmarked in his workspace into a new program that processes such data. Similarly, if the configuration allows the appropriate conversion, data bookmarks incompatible in their native format with the program can be directly and seamlessly loaded.

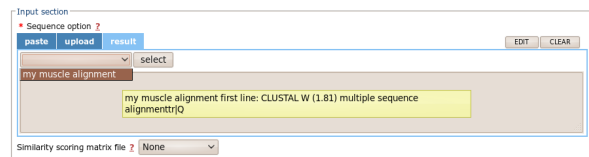


Fig 4: **history reload suggestion** in a program (EMBOSS:showalign) form in the Mobyly Portal

The result of these implicit mechanisms is that the semantic compatibility, which defines an "abstract" level of the analysis chain, is only accepted if an

executable format conversion component is available. This can be very helpful to identify relevant service composition patterns.

The chaining suggestion mechanism is also used within the BCBB Mobyly Workflow Interface (see figure 5) to guide users during the construction of a predefined workflow. By clicking on the result/output of a given task in the graph representation of the workflow definition, users are presented with a filtered list of the available next steps in the left menu. Dragging and dropping the selected program will automatically link the result to the next step.

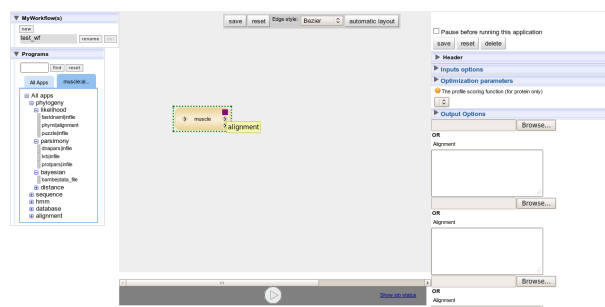


Fig. 5: **program filter in the BCBB Mobyly Workflow Interface.** Here we get the list of programs that can process the alignment produced by the first task, a MUSCLE job.

Here are the most prominent benefits of this mechanism:

1. For users, the didactic value represented by filtering only the relevant operations on a given data type helps to identify all the applications that are associated with that data type.
2. For administrators, the reduction in time needed to identify errors caused by the submission of badly formatted data to the services. Given the lack of stability and continuous improvement of bioinformatics tools, this can prevent painful maintenance operations caused by inappropriate data types.

3 MobylyNet resource sharing

A traditional approach to the need for access to bioinformatics services is to setup centralized resource centers that aggregate all tools and data. However, such a strategy does not scale well with the growing diversity of methods to operate, which can exceed the skills of the staff attached to such centers. Service providers such as Mobyly administrators are often asked to integrate new programs very quickly, in addition to answering the specific needs of a user. Issues related to the installation, maintenance, and scientific support for such programs are thus very commonplace.

As an alternative, the MobyNet organization promotes a network of smaller platforms. These platforms contribute their computing resources as well as their scientific expertise to ensure user support. The services are published on a framework that favors interoperability between the sites, enabling the seamless integration of resources to run cross-domain protocols. MobyNet is a federation of web portals running the open-source Moby framework. Its goals are:

- to integrate into a single framework (Moby) the services located on different nodes.
- to initiate a large application spectrum framework, covering complementary aspects of bioinformatics.
- to guarantee the interoperability of the services.
- to setup a confidence network, spanning over the specific skills of each platform, by enabling quality management.

The technical basis of the MobyNet initiative is the Moby Network functionality. It provides Moby server administrators the ability to share resources, by exporting and importing programs between servers: while users still access the same portal, the jobs corresponding to some programs can be remotely executed and stored. The administrator of a server can:

- import programs from other servers, allowing these programs to be displayed as available (through remotely-executed programs). A user can submit jobs to execute this program, and the jobs will be seamlessly forwarded to the remote server which hosts the program.
- export programs. This allows administrators from other Moby servers to import these programs into their portal, hence forwarding the corresponding jobs to the execution server.

Within this configuration, help requests for a given program are sent to the "service publishing" platform which actually runs it, contributing to the above-mentioned sharing of expertise.

The MobyNet currently comprises nine sites. Participating sites are either providers of services and/or developers of the core MobyNet framework. These platforms are distributed throughout France. Their areas of expertise include genomics, microarrays, sequence analysis, phylogenetics, structural bioinformatics and cheminformatics, with diverse focuses such as microorganisms, plants, pharmacology, or cancer. This network is also open to international contributions.

4 Conclusion

We presented here the Moby software suite, and its derived project, the MobyNet initiative. Both provide solutions to the question of resource integration in bioinformatics, including mechanisms that facilitate the discovery of new services, which is a key element to address when publishing bioinformatics resources to

biologists. The development and the evolutions of the Mobyly system are based on a user-centered approach, which has already been discussed in detail [15, 12], and has largely contributed to the wide adoption of the platform by both users and portal administrators.

It should be noted that, for Mobyly as for most of the other existing bioinformatics resources integration portals, the most costly task remains the description of the resources, i.e., the authoring of *service descriptions*. The existence of service description metadata in some software suites, such as ACD files in EMBOSS, allows the automation for some steps of these tasks. However, in the absence of a widely-adopted standard to describe the integrated resources, this description remains an almost, if not an entirely, manual work. Recent works to establish such standards, such as the EDAM ontology[17], could however provide a very valuable information source for this process.

Integrating other types of services in Mobyly, such as web-based external resources, is another evolution we hope to implement, with the development of a client layer for BioMOBY or web services. Finally, our current work also focuses on the extension of the workflow authoring system, and its integration into the Mobyly portal to enable the creation of workflow definitions using the history of the user.

5 Availability

The Mobyly system is distributed under the terms of the GNU GPLv2 on the project website (<https://projets.pasteur.fr/projects/show/mobyly>). It is already available on many public servers.

The currently developed BCBB Mobyly Workflow Interface is available as a prototype on <http://exon.niaid.nih.gov/mobylyWorkflow/>.

More information about the MobylyNet project can be found on <http://mobylynet.rpbs.univ-paris-diderot.fr>.

6 Acknowledgements

The MobylyNet project is funded by the IBISA (<http://www.ibisa.net>) initiative (French National coordination of the Infrastructures in Biology, Health and Agronomy). The Mobyly/BCBB collaboration is funded by the NIH-Pasteur partnership <http://nihpasteurpartnership.niaid.nih.gov>.

7 Contact:

mobyly-support@pasteur.fr, mobyly-support@rpbs.univ-paris-diderot.fr.

References

1. Alland, C., Moreews, F., Boens, D., Carpentier, M., Chiusa, S., Lonquety, M., Renault, N., Wong, Y., Cantalloube, H., Chomilier, J., et al.: RPBS: a web resource for structural bioinformatics. *Nucleic Acids Research* 33, W44 (2005)
2. Altintas, I., Berkley, C., Jaeger, E., Jones, M., Ludascher, B., Mock, S.: Kepler: An extensible system for design and execution of scientific workflows. In: *SSDBM '04: Proceedings of the 16th International Conference on Scientific and Statistical Database Management*. p. 423. IEEE Computer Society, Washington, DC, USA (2004)
3. Belhajjame, K., Goble, C., Tanoh, F., Bhagat, J., Wolstencroft, K., Stevens, R., Nzuobontane, E., McWilliam, H., Laurent, T., Lopez, R.: Biocatalogue: A curated web service registry for the life science community. In: *Microsoft eScience conference* (2008)
4. Carrere, S., Gouzy, J.: REMORA: a pilot in the ocean of BioMoby web-services. *Bioinformatics* 22(7), 900–901 (2006), <http://bioinformatics.oxfordjournals.org/cgi/content/abstract/22/7/900>
5. Edgar, R.C.: MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research* 32(5), 1792–1797 (2004), <http://nar.oxfordjournals.org/content/32/5/1792.abstract>
6. Felsenstein, J.: PHYLIP - Phylogeny Inference Package (Version 3.2). *Cladistics* 5, 164–166 (1989)
7. Giardine, B., Riemer, C., Hardison, R.C., Burhans, R., Elnitski, L., Shah, P., Zhang, Y., Blankenberg, D., Albert, I., Taylor, J., Miller, W., Kent, W.J., Nekrutenko, A.: Galaxy: a platform for interactive large-scale genome analysis. *Genome Res* 15(10), 1451–1455 (October 2005), <http://dx.doi.org/10.1101/gr.4086505>
8. Goecks, J., Nekrutenko, A., Taylor, J., Team, T.G.: Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biology* 11(8), R86 (2010), <http://genomebiology.com/2010/11/8/R86>
9. Hallard, M., Picouet, P., Rodriguez, L.F., Bigaret, S.: Bioside : faciliter l'accès des biologistes aux ressources bioinformatiques. In: *JOBIM 2004 : 5èmes journées ouvertes biologie informatique mathématique*, 27-30 juin, Montréal, Canada. pp. 64 –. IASC - Dépt. Intelligence Artificielle et Systèmes Cognitifs (Institut TELECOM ; TELECOM Bretagne) (2004)
10. Lacroix, Z., Aziz, M.: Resource descriptions, ontology, and resource discovery. *IJMSO* 5(3), 194–207 (2010)
11. Letondal, C.: A Web interface generator for molecular biology programs in Unix. *Bioinformatics* 17(1), 73–82 (2001)
12. Letondal, C., Amanatian, O.: Participatory design of pipeline tools and web services in bioinformatics. In: *Requirements Capture for Collaboration in e-Science Workshop*. National e-Science Center (January 2004)
13. Moeller, S., Krabbenhoeft, H., Tille, A., Paleino, D., Williams, A., Wolstencroft, K., Goble, C., Plessy, C.: Community-driven computational biology with debian and taverna. In: *BOSC 2010* (July 2010)
14. Navas-Delgado, I., Rojano-Munoz, M.d.M., Ramirez, S., Perez, A.J., Andres Leon, E., Aldana-Montes, J.F., Trelles, O.: Intelligent client for integrating bioinformatics services. *Bioinformatics* 22(1), 106–111 (2006), <http://bioinformatics.oxfordjournals.org/cgi/content/abstract/22/1/106>

15. Néron, B., Ménager, H., Maufrais, C., Joly, N., Maupetit, J., Letort, S., Carrere, S., Tuffery, P., Letondal, C.: Moby: a new full web bioinformatics framework. *Bioinformatics* 25(22), 3005–3011 (2009), <http://bioinformatics.oxfordjournals.org/content/25/22/3005.abstract>
16. Oinn, T., Addis, M., Ferris, J., Marvin, D., Senger, M., Greenwood, M., Carver, T., Glover, K., Pocock, M.R., Wipat, A., Li, P.: Taverna: a tool for the composition and enactment of bioinformatics workflows. *Bioinformatics* 20(17), 3045–3054 (2004), <http://bioinformatics.oxfordjournals.org/cgi/content/abstract/20/17/3045>
17. Pettifer, S., Ison, J., Kalaš, M., Thorne, D., McDermott, P., Jonassen, I., Liaquat, A., Fernández, J.M., Rodriguez, J.M., Partners, I., Pisano, D.G., Blanchet, C., Uludag, M., Rice, P., Bartaseviciute, E., Rapacki, K., Hekkelman, M., Sand, O., Stockinger, H., Clegg, A.B., Bongcam-Rudloff, E., Salzemann, J., Breton, V., Attwood, T.K., Cameron, G., Vriend, G.: The EMBRACE web service collection. *Nucleic Acids Research* 38(suppl 2), W683–W688 (2010), http://nar.oxfordjournals.org/content/38/suppl_2/W683.abstract
18. Reich, M., Liefeld, T., Gould, J., Lerner, J., Tamayo, P., Mesirov, J.: GenePattern 2.0. *NATURE GENETICS* 38(5), 500 (2006)
19. Rice, P., Longden, I., Bleasby, A.: EMBOSS: The European Molecular Biology Open Software Suite. *Trends in Genetics* 16(6), 276–277 (June 2000), [http://dx.doi.org/10.1016/S0168-9525\(00\)02024-2](http://dx.doi.org/10.1016/S0168-9525(00)02024-2)
20. Schultheiss, S., Muench, M., Andreeva, G., Raetsch, G.: Persistence and availability of web services in computational biology. *PLoS Comp Biol* (2010 (under review))
21. Stein, L.D.: Creating a bioinformatics nation. *Nature* 417(6885), 119–120 (May 2002), <http://dx.doi.org/10.1038/417119a>
22. Thompson, J.D., Higgins, D.G., Gibson, T.J.: CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res* 22(22), 4673–80 (Nov 1994)
23. Waterhouse, A.M., Procter, J.B., Martin, D.M., Clamp, M., Barton, G.J.: Jalview Version 2—a multiple sequence alignment editor and analysis workbench. *Bioinformatics* (Oxford, England) 25(9), 1189–1191 (May 2009), <http://dx.doi.org/10.1093/bioinformatics/btp033>
24. Wilkinson, M., Schoof, H., Ernst, R., Haase, D.: BioMOBY Successfully Integrates Distributed Heterogeneous Bioinformatics Web Services. The PlaNet Exemplar Case. *Plant Physiol.* 138(1), 5–17 (2005), <http://www.plantphysiol.org/cgi/content/abstract/138/1/5>