Comparative Evaluation of Microarray-based Gene Expression Databases

Hong-Hai Do[†], Toralf Kirsten[†], Erhard Rahm[‡]

[†] Interdisciplinary Centre for Bioinformatics, [‡] Department of Computer Science
University of Leipzig
www.izbi.de, dbs.uni-leipzig.de

Abstract. Microarrays make it possible to monitor the expression of thousands of genes in parallel thus generating huge amounts of data. So far, several databases have been developed for managing and analyzing this kind of data but the current state of the art in this field is still early stage. In this paper, we comprehensively analyze the requirements for microarray data management. We consider the various kinds of data involved as well as data preparation, integration and analysis needs. The identified requirements are then used to comparatively evaluate eight existing microarray databases described in the literature. In addition to providing an overview of the current state of the art we identify problems that should be addressed in the future to obtain better solutions for managing and analyzing microarray data.

1 Introduction

With genomes of several organisms, especially the human genome, completely sequenced, the main focus of genomic research has shifted to using these sequences in order to understand how genes and ultimately entire genomes are functioning. Although all cells in an organism carry the same genetic information, only a subset of the genes is active, i.e. expressed, conferring unique properties of the cells in their specific conditions. Analyzing the behavior of the genes, i.e. whether and to what degree they are expressed, can help characterize and understand the functions of genes. In particular, it can be analyzed how the activity level of genes changes under different conditions such as for specific diseases, before and after the use of specific drugs, etc.

Various methods have been developed for detecting and measuring gene expression, including Northern Blotting [AK77], Differential Display [LP92], Reverse Transcription-Polymerase Chain Reaction (RT-PCR) [SW95], EST (Expressed Sequence Tag) Clustering [VE98], Serial Analysis of Gene Expression (SAGE) [VZ95] and microarrays [SS95, LD96, Na99]. Microarrays are quickly becoming the predominant approach because they allow performing expression analysis on a very large scale, i.e. to measure and study the expression of thousands of genes simultaneously. As a consequence, huge amounts of data are produced with every experiment. Moreover, the amount of data being produced is expected to explode with the falling cost of microarray technology.

Recently, several databases have been developed for storing and analyzing microarray data. Some of them have been reviewed in [GL01] with a focus on the databases' analysis capabilities. However, the broader requirements to build, maintain and use such a database in a flexible way are not sufficiently considered. Furthermore, most of the considered databases are not available to the public and/or have not been presented in scientific publications.

To obtain a better overview about the current state of the art in using database technology for gene expression analysis, we review the available microarray databases described in recent scientific publications. For this purpose, we first discuss the major requirements for managing microarray data. We cover important database-related issues that have been left open in [GL01], e.g. performance aspects, data integration, and the coupling of analysis/data mining with the database. We then use these criteria to comparatively evaluate eight database implementations and thus assess the current state of the art. We hope that our requirement analysis and evaluation helps identifying fruitful areas for future research and guiding the design and development of more powerful solutions for microarray data management.

The rest of the paper is organized as follows. In the next section we provide a short introduction to microarray technology. In Section 3 we present the main requirements for a microarray database. Section 4 compares the selected databases according to the identified requirements and criteria. In Section 5 we conclude and point to database-related problem areas for future work.

2 Microarray-based Gene Expression Measurement

The genetic information in the DNA is organized within two complementary strands consisting of sequences of four different nucleotides, Adenine (A), Thymine (T), Guanine (G) and Cytosine (C). Adenine and Guanine are the complements of Thymine and Cytosine, respectively. When two complementary sequences find each other, they hybridize (i.e. bind together). Gene expression is the cellular process that turns the genetic information of the DNA into proteins, which ultimately determine the morphology and functionality of a cell. Protein synthesis starts with the so-called transcription process, in which the genetic information of the DNA is transferred to the short-lived messenger RNA (mRNA). Measuring the mRNA abundance, i.e. *the transcription* or *expression levels*, in various tissues and under different environmental conditions can help understand the dynamic functioning of genes as well as their mutual influence in the regulatory network.

2.1. Microarray Principle

Microarrays are based on the same basic principle: the preferential binding of complementary, single-stranded nucleic-acid sequences. On a microarray (also called a *chip*), known sequences called *probes* are attached at fixed locations (*spots*). There are two variants of the microarray technology:

- *cDNA arrays (spotted arrays)*: This is the oldest microarray technology and was developed at Stanford University. It is based on immobilizing complementary DNA (cDNA) probes of length of 500~5,000 bases (nucleotides), each representing a gene, to a solid surface such as glass using robot spotting.
- *Oligonucleotide arrays*: These arrays use shorter sequences as probes, so-called oligos of 20~80 bases. Unlike in spotted arrays, a gene is represented by a set of oligos, i.e. a *probeset*. This technique was developed by Affymetrix, Inc.

2.2. Experiment Design

Figure 1 gives a schematic overview of a *microarray experiment*. In a cell, when a gene is expressed, mRNA transcripts are produced. In a microarray experiment, these transcripts, also called *targets* in the experiment context, need first to be isolated from the

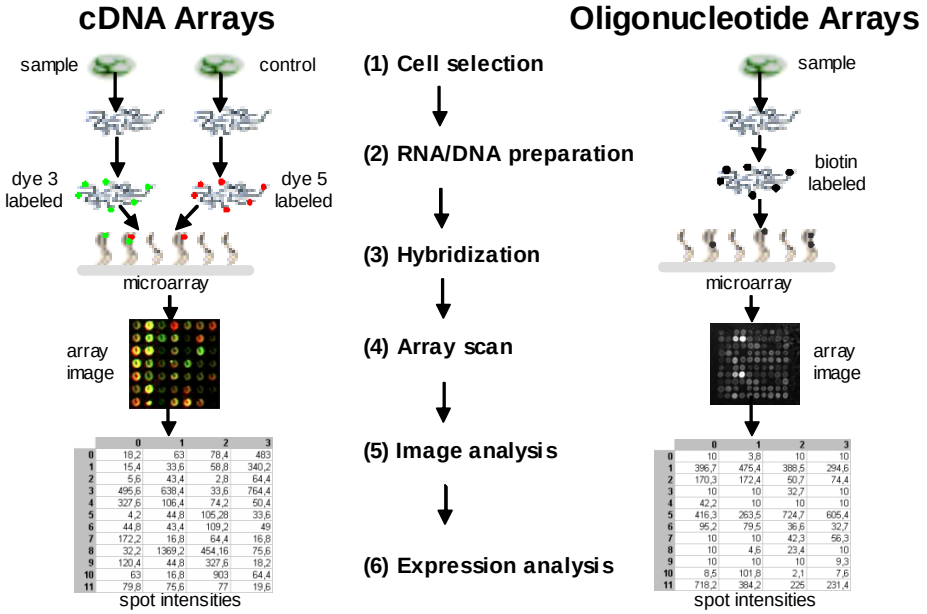

Figure 1. Schematic overview of a microarray experiment

sample of interest (Step 1), reverse-transcribed to cDNA, and tagged with a particular fluorescent dye to mark their origin (Step 2). In case of spotted arrays, transcripts from two samples, e.g. cells from a normal tissue for control and cells from a disease tissue for testing, are needed. When bathing the microarray in the target mixture, the *hybridization* process (Step 3) takes place and the available transcripts bind to the corresponding probes on the chips.

After the hybridization phase, the array is scanned to determine how much of each target is bound to each spot. The *scan* process delivers an image of the array showing spots with different color/brightness intensities depending on how many fluorescent targets are bound to the corresponding probes (Step 4). Finally, these intensities are measured and corrected against the background noise using some *image analysis* software to produce the expression level of each gene (Step 5). For oligonucleotide arrays, the intensities of the spots refer to oligos and are to be combined to produce a single intensity value for the corresponding gene.

To examine gene expression levels under various aspects, e.g. in different tissues or in a time series, an *experiment series* is to be conducted. In general, expression data is of a multidimensional nature: each measured expression level is a point in an *n*-dimensional space with dimensions such as the genes, gene functions, and the different conditions under which the genes have been studied. As we will see later, various analysis approaches can be employed in Step 6 to infer and interpret gene functions from expression data.

3 Database Requirements for Gene Expression Analysis

In this section we discuss the major requirements for microarray databases supporting gene expression analysis. In particular, we consider criteria from the following areas:

- *Data Characteristics*: We analyze which types of data need to be managed and their characteristics to be considered
- *Management of Annotation Data* describing the semantics of the expression data measured in the experiment
- *Data Integration*: In addition to the data generated by the microarray experiment itself, gene expression analysis should exploit annotation information available from public sources. We examine which data is useful and how it can be integrated
- *Data Interfaces* for data exchange (import, export)
- *Access Control* to avoid unauthorized database access in a multi-user environment
- *Data Normalization*: to improve the quality of gene expression measurements, which may suffer from noise due to various experimental fluctuations
- *Data Analysis*: which approaches are useful for gene expression data analysis
- *Tool Integration*: coupling of analysis algorithms and existing tools with the database

In the following we elaborate on these criteria in more detail.

3.1. Data Characteristics

Gene expression analysis requires various kinds of data, which are not only produced directly by the microarray experiment but can also stem from other sources. We distinguish between *Image*, *Expression* and *Annotation Data*. The latter is further divided into *Gene*, *Sample* and *Experiment Annotations*. Table 1 summarizes their characteristics and usage in gene expression analysis.

Data		**Source**	**Type**	**Characteristics**	**Usage**
Image Data		Array scan	Binary	large files	Generation of expression data
Expression Data		Image analysis	Number	fast growing volume	Visualization, statistical and cluster analysis
Annotation Data	Gene	External public sources	Text	regularly updated	Interpreting / Relating / Inferring gene functions
	Sample and Experiment	User input		user-specified, often free text	

Table 1. Relevant types of data and their characteristics

Image Data. Images are produced as large files in the array scan process. They represent the starting point for expression analysis. Because image analysis software may be changed or updated, both the images and their association with the generated expression data should be managed so that the previous analysis results can be reproduced and corrected. Access frequency is relatively rare and mostly read-only. Images may be stored within the database itself or in the file system with the file names or URLs kept in the database.

Expression Data. Expression data, i.e. numbers indicating gene expression levels, represents the core of a microarray database. It is of high volume and fast growing. Gene expression levels computed by different technologies, such as cDNA arrays, oligonucleotide arrays, as well as other non-array technologies like SAGE possess different semantics, and therefore are difficult to compare with each other without being normalized first. Unlike images, expression data is accessed more frequently. Typically, analysis poses high performance requirements due to the high data volume and the frequent need of interactive analysis requiring short response times. This asks for the use of advanced DBMS techniques such as materialized views, indexing and parallel processing that may have to be tailored to specific analysis needs.

Annotation Data. Annotations are metadata describing the expression levels measured in a microarray experiment, often in the form of textual descriptions. They help the user

in interpreting the detected gene expression levels, especially for inferring and relating gene functions. We distinguish between the following kinds of annotation data:

- *Gene Annotations*: Sequences placed on microarrays usually represent already known genes. Their annotations, e.g. names, currently known functions, location on chromosome, etc. are essential for interpreting the measured expression levels. Such information has been continuously collected, regularly updated and made available in various public data sources.
- *Sample Annotations*: This data describes how the targets have been extracted and prepared for hybridization. Moreover, it also includes biological descriptions, such as the source and characteristics of the sample, e.g. tissue and disease, any genetic and chemical manipulation and stimulation, any *in vivo* or *in vitro* treatments applied. For patient-related measurements personal characteristics such as age, sex and clinical status information can provide further important criteria for analysis.
- *Experiment Annotations*: This data describes primarily the technical process of the experiment. In particular it captures the protocols and parameter settings used by the machine and software for hybridization, for washing and scanning the array.

Typically gene annotation data has to be integrated from external public sources, while sample and experiment annotations need to be manually specified by the user for every new experiment. This leads to special requirements that are discussed in the following.

3.2. Management of Annotation Data

Because annotation data comes from heterogeneous sources, such as external databases and user input, it is essential to capture and organize annotation data in a uniform and flexible way so that it can be effectively used in analyzing expression data. Current databases often use free-text fields to capture annotation data, leading to two problems. First, because different sources and users often use different vocabularies, free-text fields tend to introduce large annotation discrepancies. Second, each free-text field potentially contains many terms or values, making them difficult to be queried in the database. Heterogeneous annotations make it difficult to identify comparable experiment results and to perform cross-experiment analysis.

As a result free-text fields should largely be avoided for annotations. Rather, annotations should be split into atomic items or categories with clearly defined semantics of simple data types, such as numbers of predefined units or values from a predefined list. The items as well as their values should be specified using a controlled *vocabulary*, which can either be specifically developed for local use only, i.e. a local vocabulary, or based on an existing standard, i.e. a standardized vocabulary. Moreover, the categories should not only be collected in flat vocabularies, but also organized into multiple levels, e.g. *taxonomies* and *ontologies*, to increase their expressiveness and support more focused analysis capabilities. A taxonomy, such as the gene function taxonomy of the GeneOntology (GO) Consortium [GO00], is a specialization/generalization hierarchy of categories, which are connected with each other by *is-a* relationships. Ontologies such as TAMBIS [BG99] often represent additional semantics, e.g. complex networks of categories.

The database representation for annotations should take into account that the relevant items/categories and vocabularies change over time, e.g. if the experiments change their biological focus. Figure 2 illustrates two representation schemes for annotation data, a straightforward relational approach and the so-called *Entity-Attribute-Value* (EAV) ap-

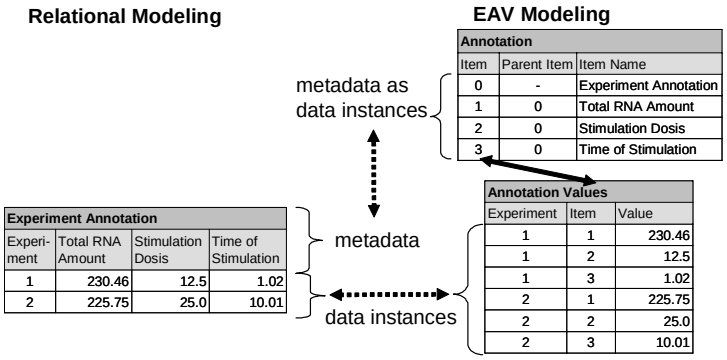

Figure 2. Relational vs. EAV approach for modeling annotation data

proach [NB98]. In the former method, annotation categories are modeled as attributes of tables and the captured values are stored as instances. This simple approach makes it easy to control the correctness of annotation values and to use them for data analysis. However, it is only suitable for annotations that rarely change over time because the database schema has to be modified whenever different annotations are needed. Moreover, hierarchically organized categories need additional support.

On the other side, the EAV method is more flexible and robust against changes in annotation categories. Here, each annotation category or item appears as a uniquely identified instance in a metadata table, the *Annotation* table, which can be easily extended to capture more items. Furthermore, the items can also easily be organized in a hierarchical way. The captured values for each item are stored in another table, the *Annotation Value* table. The price for the flexibility of the EAV approach is that queries are hard to formulate and that the two tables always need to be joined to associate the items with their values. This drawback however may be reduced by building materialized views to simplify and speed up frequent queries involving annotation data.

3.3. Data Integration

We first discuss the challenges directly resulting from the information need of gene expression analysis and then the mechanisms for integrating gene annotation data.

Integration Requirements. Gene annotation data is stored in various public data sources accessible on the web. For instance, gene sequences placed on an array often stem from a sequence database, such as GenBank [WC02], which maintains all known nucleotide and protein sequences with different annotations, e.g. on bibliographic references, organisms, coding regions, repeat regions, mutations, etc. To characterize the functions of genes, their expression patterns should be related with the functions of their products, e.g. proteins. SwissProt [BA00] is a curated protein sequence database providing for each protein sequence extensive annotation information, such as functional descriptions, structures, associated diseases etc. Moreover, the genes can be examined in a broader context, namely in the network of interactions between their proteins. This information can be exploited from KEGG [KG00], a collection of pathway maps computerizing the network information of molecular interactions, such as metabolism, signal transduction, cell cycle, etc.

Since a gene is often represented by multiple sequences in GenBank, annotations at the sequence level are often impractical for functional gene analysis. Hence a major requirement is to integrate sequence-level annotations from different sources to provide

gene-oriented annotations, e.g. for gene expression analysis. This is already supported by several databases. LocusLink and RefSeq [PM01] are the sources of choice for curated annotations and sequences of known genes. Other databases such as UniGene [WC02], TIGR [QC01] and Ensembl [HB02] have been constructed using different automatic gene prediction algorithms and provide computed annotations for the predicted genes. The Human Genome Browser (HGB) [KS02] maps the sequence of the genes maintained by different databases/predicted by different algorithms uniformly onto the genome, providing a powerful visual mean for comparing genes from different databases as well as for relating the genes with other annotations, like tandem repeats, CpG islands, homology between species, which can also be mapped onto the genome. Finally, microarray vendors also provide annotations for the genes on their own chips. For example, Affymetrix users can exploit the NetAffx database for probeset annotations for all Affymetrix chips [LL02]. The annotations include information integrated from LocusLink, UniGene and SwissProt, as well as various in-house computed annotations.

Typically, each database uses proprietary gene identifiers so that the same gene may be found under different identifiers in different sources. Furthermore, vendor-proprietary gene identifiers such as Affymetrix probesets are unknown in public annotation databases and not suitable for referencing in scientific publications. Therefore, an essential requirement in integrating gene annotation data is to relate the corresponding genes between public annotation databases with the proprietary genes of microarray vendors.

Integration Mechanisms. Traditional approaches for data integration are *Virtual* and *Materialized Integration*. In the former approach the data is retrieved from the corresponding sources when it is needed, while the latter locally replicates the data from the external sources. We further differentiate between two variants of the virtual approach, namely *Web Link* and *Federated* integration.

- *Web Link Integration*: This approach is followed by most current databases and only stores the *accession keys*, the unique keys to access data entries in the external sources. Using accession keys, web links can be built automatically, allowing the user to navigate to the corresponding source in order to obtain annotation information for the genes of interest. While requiring only little integration effort, this approach shows significant limitations. Firstly, it is not possible to consider several genes, for example in an identified gene cluster, at the same time. Secondly, and more importantly, it is not possible to directly relate the annotations and expression of genes for database queries or data mining.
- *Federated Integration*: In this variant of virtual integration, the schemas of the relevant sources have first to be integrated to a global schema. Determining a consistent global schema is a major problem due to typically large degrees of semantic heterogeneity between different sources, despite the availability of some global taxonomies (GO etc.). Furthermore, a complex mediator software is needed supporting queries against the global schema by executing relevant subqueries at the respective data sources and combining their results. The approach also suffers from the only rudimentary query capabilities of public sources, typically based on string/pattern matching of text. Furthermore, strategies to automatically deal with possibly dirty and overlapping data between different sources have to be developed and incorporated in the query processing engine. Moreover, query processing depends on the availability and performance of the corresponding sources. On the positive side, the data itself needs not be replicated and the most up-to-date data can be retrieved and analyzed.

- *Materialized Integration*: This approach corresponds to data warehousing and requires extensive preprocessing effort. Not only the source schemas have to be integrated, but the data has also to be extracted from the single sources, transformed, cleaned, and then uniformly stored in the microarray database (warehouse), together with expression measurement data. As external sources are regularly updated, automatic techniques are needed to refresh the local data on a continuous basis. Once the data has been integrated, the warehouse approach promises significant advantages because all relevant data is directly accessible for analysis. This can help to provide both good performance and extensive analysis capabilities.

3.4. Data Interfaces

We first discuss data exchange interfaces w.r.t. other databases and software tools. We also discuss security aspects, i.e. how to control data access of users.

Data Exchange. Because experiments may be continuously conducted by different users in different labs, a public database should provide the users with possibilities to import, i.e. submit, their data for management and analysis. Furthermore, some users may want to export the data so that they can use external tools or programs for data analysis. Hence, interfaces for both import and export are required.

The most common way for data exchange is to support a particular flat file format. Expression data can be easily organized in a matrix, the gene expression matrix [BH01], with rows representing genes and columns the investigated samples. Hence, tab-delimited files represent a straightforward way to exchange expression data. This file format has an essential disadvantage that it does not include the corresponding experiment and sample annotations. In contrast, annotation can be easily specified using XML which has already been widely used to exchange data over the web. Several efforts have developed proposals for a standard XML format for microarray data, e.g. MAGE-ML[1], GEML[2], and GeneXML[3].

Access Control. The control of user access to expression data is an important criterion for the acceptance of a microarray database due to two main reasons. First, the experimenter usually wants to hold back his/her expression data and analysis results until they have been published in some journal or conference contribution. Second, annotation data may contain sensitive person-related information, such as patient and clinical data.

As usual, access control has to consider the individual users, the available data as well as the access rights or functions. This may be achieved with the authorization concept of the underlying DBMS or by a specific implementation. With respect to the *users*, the database should provide some mechanisms for building a hierarchy of individual users, groups, and roles. Regarding the *data*, different levels of granularity should be distinguished, such as expression data of an experiment/experiment series or annotation data. Finally, different access *functions* should be supported such as data import, export, or performing certain analysis types.

3.5. Data Normalization

Raw expression data produced by the image analysis process still contains noise. In particular, each step in target and probe preparation, in the hybridization, wash and scan

[1] http://www.mged.org/Workgroups/MAGE/mage-ml.html
[2] http://www.rosettabio.com/products/conductor/geml/default.htm
[3] http://www.ncgr.org/genex/genexml.html

process represents a source of fluctuations which can influence the determination of expression data in different ways. For example, the efficiency of the hybridization reaction depends on a number of experimental parameters, such as temperature, time, and the overall amount of available mRNA. Since the reliability of expression patterns derived from array data is essential for their interpretation, a data normalization step aiming at reducing the effects of such fluctuations is necessary.

Currently, many strategies have been proposed for normalizing data from a single experiment or from an entire experiment series. Descriptions and evaluations of the various strategies for cDNA and oligonucleotide arrays can be found in [SB00] and [HB01], respectively. For a single experiment, common normalization strategies perform a division by a constant approximately determined by average intensity either of all spots on the array (ratio vs. total) or of a few control genes (ratio vs. control), such as the so-called housekeeping and spiked genes, whose expression behavior is already known or predictable. For multiple experiments, one approach is to normalize them against one reference experiment which has been conducted for a control sample. After being adjusting to a common standard, the results from different experiments can be compared.

Because there is still no agreed-upon standard procedure, the common normalization strategies should be provided so that the user can choose to pre-process his/her expression data. Moreover, not only the normalized, but also the raw expression data should be stored in the database, so that a re-normalization can be performed later on, e.g. for testing and evaluating novel normalization strategies.

3.6. Data Analysis

The analysis process takes the normalized expression data and tries to derive the relationships between the genes and samples. Most methods for gene expression analysis have already been developed and used in other areas, especially data warehousing, data mining, and statistics. We differentiate between the following families of analysis approaches:

Querying/Reporting. This standard database access allows the user to navigate in the database and to retrieve a subset data of interest for further study or visualization. To simplify the construction of frequent queries and speed up their execution, canned queries and reports should be supported. They are pre-defined database queries, which are stored so that they can be executed at any time with different user-specified parameter values. For example, canned queries can be defined to filter genes based on specific expression level thresholds and/or functional annotation.

Online Analytical Processing (OLAP). OLAP has been widely used in data warehousing to analyze multi-dimensional data; recently, the use of OLAP has also been proposed for the biotech domain [Hu01]. Because of the multidimensionality of gene expression data, this technique represents a promising approach to gene expression analysis. Assuming a proper representation for dimensional annotations, the user may interactively navigate through different levels in the hierarchy of a dimension, such as the GO function hierarchy of genes, to obtain and compare summarized information about gene expression patterns.

Data Mining. Data mining supports the detection of interesting patterns in large data sets and has commonly been used for analyzing expression data. There are unsupervised approaches, e.g. clustering, as well as supervised schemes, e.g. classification methods.

- *Clustering*: represents the most common analysis method for expression data. The goal of clustering is to group together objects, i.e. gene or samples, with similar properties. So far, many algorithms, such as hierarchical, K-mean clustering algorithms, and Self-Organizing-Maps (SOM) [BR02], have been developed and successfully employed to analyze expression data. Typically, genes are clustered to identify co-regulated and functionally related genes. Furthermore, clustering can also be performed for samples. Samples with similar expression patterns may constitute some new, previously undefined subgroups, e.g. for diseases like tumors. These findings can be useful for designing treatment procedures for different groups of patients.
  Clustering is often accompanied by dimension reduction methods, which can either identify and disregard the less informative dimensions or establish a new smaller set of dimensions as combination of the original dimensions. These methods include Multidimensional Scaling (MDS), Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) [Do02].
- *Classification*: This supervised approach is often based on machine learning and aims at assigning predefined classes of known characteristics and functions to given expression patterns. Popular classification methods include linear discriminants, decision trees, and Support Vector Machines (SVM) [BG00]. Typically, such classifiers are first trained on a subset of data, for which classification is already known, and then tested to find classification for another subset of data.

Statistics. Statistical methods are emerging to account for multiple sources of variation when trying to pool information from many microarrays and to identify genes exhibiting significant differential expression. The ANOVA approach [KM00] decomposes the appropriately transformed expression measurement as a linear combination of effects from different sources of variation. Also in this context, several other statistical techniques have also been employed, including the permutation-testing and p-value adjustment [DY02], the t-test and Wilcoxon test [Pa02].

Visualization. All analysis methods need to summarize the results in a comprehendible way for human interpretation. Using different techniques, such as scatter plots, dendrograms, charts and graphs, a large amount of data can be surveyed and examined simultaneously. In particular, visualization is needed to display the results of clustering.

3.7. Tool Integration

Typically, data analysis has to be performed iteratively and in interaction with the user. This requires a close integration of the analysis methods with the database. For the various types of analysis discussed above, many algorithms and visualization support are already available in powerful software tools. These tools should be usable together with the microarray database which also helps to limit development effort. We distinguish between three forms of tool integration.

Loose Integration. In this scenario, only little integration effort is needed. The user uses the export interface of the database to export a subset of data of interest, typically gene expression levels, to a flat file, which is then imported in the corresponding tools. Its main drawback however consists in the lack of annotation data in those tools for interpreting the expression patterns. Even so, for many proprietary tools it represents the only way to analyze expression data.

Transparent Integration. This approach can be employed to integrate tools which provide some API to their functionalities. A single user interface can be built covering

multiple tools addressing different steps in gene expression analysis. The communication between the tools and the database can be based either on direct database queries or flat file export and import, which is however hidden from the user.

Tight Integration. As opposed to loose integration, this approach requires analysis algorithms to work directly on the database. It represents the most promising integration form because it can exploit all available data in the database and achieve the best performance. However, it implies a high implementation effort to re-develop the approaches as new database applications and to tune the database, or to directly integrate the analysis approaches into the DBMS, e.g. as stored procedures or special type extensions.

4 System Evaluation

According to the introduced criteria and requirements we compare in this section several public microarray databases, which we could identify from recent scientific literature. Being presented and discussed in scientific publications, their approach has actually been approved by the research community and should show impact on future work. However, we have encountered a number of systems, such as GEO [ED02], Gene Expression Atlas [SC02], HugeIndex [HW02], yMGV [CD02], READ [BK02], and SGD [BJ01], which are still at early stage of development and/or have not been described with sufficient detail for our purpose. Therefore, we do not consider those databases but only focus on 8 databases listed in Table 2. In addition to the information from the publications, we also test and consider the functionalities provided by the websites of the corresponding databases.

Databases	Organization	References
ArrayDB	National Human Genome Research Institute (NHGRI) *http://genome.nggri.nih.gov/arraydb*	[ER98]
ExpressDB	Harvard University *http://arep.med.harvard.edu/ExpressDB*	[AR00]
GeneX	National Center for Genome Resources (NCGR) *http://genebox.ncgr.org/genex*	[MC01]
GIMS	University of Manchester *http://www.cs.man.ac.uk/~norm/gims*	[CP01, PK00]
M-CHIPS	German Cancer Research Centre (DKFZ) *http://www.mchips.de*	[FH02, FH01]
RAD2	University of Pennsylvania *http://www.cbil.upenn.edu/RAD2*	[SP01]
SMD	Stanford University *http://genome-www4.stanford.edu/MicroArray/SMD*	[SH01]
YMD	Yale University *http://info.med.yale.edu/microarray*	[CW02]

Table 2. Microarray databases in the evaluation

4.1. Technical Implementation

Table 3 shows an overview of the databases according to their technical implementation. While all databases are accessible over the internet, they are intended to store and support analysis of microarray data generated by local labs. However, external users can pose queries to the data, which has been made available to the public. Only very few projects, in particular, ArrayDB, ExpressDB, GeneX and SMD, are open-source, allowing a local installation. All databases in our evaluation make use of DBMS technology, which is in most cases a commercial relational DBMS. GIMS represents an exception by using POET, an object oriented DBMS. All databases, except for GIMS, provide web interfaces for data access, which have been implemented using common web technologies, such as

Perl and Javascript. For data analysis routines, some databases also use special programming environments, such as the statistic language R and the scientific software package MatLab.

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
Access	no public data submission but public queries							
Open source	Yes			No			Yes	No
DBMS	Sybase	Sybase	PostgreSQL, Sybase	POET	Post-greSQL	Oracle	Oracle	Oracle
Program languages	Perl, Java	Perl, Javascript	Perl, Java, R	Java	C, Perl, MatLab	Perl, Java	Perl, C	Perl
GUI	Web	Web	Web	Java	Web	Web	Web	Web

Table 3. Technical implementation of the databases

4.2. Supported Kinds of Data

Table 4 shows the different types of data managed by each database. Only few databases, in particular ArrayDB, SMD, and YMD, consider storing array images for later reference and analysis. The images are not managed by the DBMS, but in file systems. The DBMS only stores the path to the image files, which is to be specified explicitly.

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
Images	in file system	no	no	no	no	no	in file system	in file system
Expression Data	cDNA	cDNA, Oligo, SAGE	cDNA, Oligo, SAGE	cDNA (public data sets from SMD)	cDNA, Oligo, SAGE	cDNA, Oligo, SAGE	cDNA	cDNA, Oligo

Table 4. Images and expression data managed by different databases

cDNA arrays are supported by all databases. Several databases are able to store expression data produced by other technologies. The new emergence of the oligonucleotide arrays has already been taken into account by several solutions. In particular, expression data from Affymetrix arrays can be managed by ExpressDB, GeneX, RAD2 and YMD. Furthermore, a few databases including ExpressDB, GeneX, and RAD2 also support SAGE expression data. Currently, SMD represents the biggest microarray database with more than 538 million expression data points from 25 thousands experiments (as of July 2002). However, despite the huge data amount and the requirement for query performance, no experience has been reported so far concerning the use of advanced DMBS techniques, such as materialized views and parallel processing.

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
Sample Ann.	tissue, cell type	1 free-text field	sex, age, tissue, dev. stage, ...	no	very comprehensive lists of annotation information	sex, age, disease, dev. stage, ...	sex, age, status, ethnicity, ...	no
Experiment Ann.	array printing, environmental conditions	1 free-text field	hardware and software parameters	no		RNA amplification, labeling protocol, scan parameters	yes	no

Table 5. Sample and experiment annotations

Because gene annotation represents data to be integrated from external sources, we will discuss it later in the next section together with the data integration mechanisms. Table 5 shows the information which currently can be specified for sample and experiment annotation. Most databases allow the specification of some information, such as sex, age, tissue, developmental stage, to annotate the sample being examined as well as the protocols, hardware and software parameters used to conduct the experiment. However, the degree of details varies drastically from database to database. Some databases,

such as GIMS and YMD, completely lack the possibility to specify sample and experiment annotations.

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
Capturing	no controlled vocabularies	no controlled vocabularies	local vocabularies	-	local vocabularies	standard vocabularies	local vocabularies	-
Modeling	relational	relational	relational	-	EAV	relational	relational + EAV	-

Table 6. Capturing and modeling/storing sample and experiment annotation

Table 6 shows the techniques used in current databases for capturing and modeling sample and experiment annotation. Typically, free text fields are provided to specify their own descriptions. Especially, ExpressDB provides a single text field for completely annotating the sample and the execution of the experiment, respectively. In contrast, M-CHIPS does not allow any free text fields and employs comprehensive lists of strictly defined annotation items, enforcing the user to specify very detailed information about the sample and the experiment. Because of the pre-definition of all annotation items and their values, annotation data in M-CHIPS is uniform across different experiments and can be exploited for statistical analysis. Other databases try to limit the negative effects of free text by enforcing controlled vocabularies when applicable. While GeneX and SMD use vocabularies developed by local users, RAD2 exploits standard vocabularies employed in other sources, in particular NCBI Taxonomy, MGD mouse anatomy and KEGG disease table.

Typically, the databases use a standard representation with fixed attributes for sample and experiment annotations. Only M-CHIPS and SMD follow the EAV approach and hence are more flexible in case new annotation information is to be captured. While M-CHIPS strictly applies the approach to store its annotation data, SMD employs both approaches and only uses EAV when necessary.

4.3. Data Integration

Table 7 shows the public data sources, which have been integrated with the current microarray databases. We observe that web linkage represents the most commonly used integration mechanism. Despite the limitation that the annotation data residing in the external source cannot be programmatically involved to analyze the local expression data, this approach requires almost no integration effort and is the fastest way to make a database solution ready for public use. Among others, UniGene, GenBank, SwissProt and KEGG represent the mostly referenced sources.

On the other side, the federated approach of virtual integration has not been exploited by any current databases. Similarly, the materialized approach also has found only very little use. In particular, still very limited gene annotation data has been integrated and

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
Web Link	UniGene, dbEst, GenBank, KEGG	BIGED	SGD, MGD, dbEST, GenBank, KEGG, SwissProt	no	yes	GenBank, AllGenes, KEGG	dbEST, GeneMap, LocusLink, SwissProt,	UniGene DRAGON, SOURCE
Federated	no							
Materialized	no	names, functional groups for yeast (MIPS)	no	functional groups for yeast (MIPS)	GO functions	no	GO functions (SGD), gene names (WormPD), UniGene	no
Auto. Update	-	no	-	no	no	-	yes	-

Table 7. Gene annotation data integrated from external public databases

replicated in the current microarray databases. Mostly, the data has been imported just once, as for example in ExpressDB and GIMS, and no mechanism for continuously updating them is provided. So far, SMD represents the single effort to comprehensively integrate gene annotation data from different sources while providing mechanisms to automatically keep the local data updated with the sources.

4.4. Data Interfaces

In this section we examine how interfaces for data exchange and user access have been implemented by the current databases.

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
Import	tab-delimited	tab-delimited	tab-delimited, GeneXML	tab-delimited	tab-delimited	tab-delimited	tab-delimited, Genepix, Scanalyze	tab-delimited, Genepix, GPMerge
Export		no		no	no	no	tab-delimited, TreeView	tab-delimited, Excel, CLUSTER
Automation	directory scan	no	no	no	no	no	batch import	batch import

Table 8. Data exchange interfaces and mechanisms

Data Exchange. Table 8 shows the interfaces provided by the corresponding databases for data exchange. Data exchange mostly addresses gene expression levels, so the ASCII tab-delimited file format is widely supported for both import and export.

SMD and YMD allow direct import of data from proprietary files produced by particular image analysis software such as Genepix and Scanalyze. Furthermore, they also support exporting data to the formats required by some analysis tools, such as TreeView, CLUSTER and Excel. GeneX represents the first effort so far to use XML as exchange format. In particular, it allows microarray data, i.e. both annotation and expression data, to be imported and exported using the proprietary format GeneXML.

Mostly the user has to manually initiate the import and export process from the database website. Automation for import is provided only by ArrayDB which can automatically scan a specified directory for import files. SMD and YMD also support import of multiple files which however have to be specified first on the database website.

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
User/group	application-based	application-based	application-based	-	separate databases for each group	application-based	application-based	application-based
Granularity	experiment		experiment	-	experiment series	experiment	experiment	experiment series
Function	No							

Table 9. Control of data access

Access Control. Table 9 shows how the data access is implemented in current databases. Typically, the databases implement their own user and group hierarchies (at application level) and do not make use of the DBMS-provided user/group concept. M-CHIPS represents an exception by providing each group of users, which perform similar experiments, with a separate logical database.

Typically, the finest granularity of data that can be assigned to the user is the experiment, which consists of both annotation and expression data. A distinction between these two kinds of data for access control is not yet implemented in any database. A few databases, such as YMD, assign an entire series of related experiments to a user. This is also the case with M-CHIPS, as a group-specific database in M-CHIPS also represents a series

of related experiments. Finally, no database supports the restriction of functions that can be performed by a particular user on the assigned data set.

4.5. Data Normalization

Table 10 shows the normalization strategies supported by the different databases. Several databases, such as ArrayDB, ExpressDB and GIMS, do not implement any normalization strategies, leaving to the user the task to normalize the data before uploading it. Hence, the user has to be aware about whether the data stored in the database has been normalized, and if so, using which strategy.

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
Normalization methods	no	no	average, ratio vs. control	no	robust affine-linear regression vs. control	ratio vs. total, ratio vs. control	2 strategies with scaling factors	no
Expression data	-	-	raw + normalized	-	raw + normalized	raw + normalized	raw + normalized	-

Table 10. Supported normalization strategies

On the other side, a few databases, e.g. SMD, M-CHIPS and RAD2, provide integrated strategies and allow the user to choose the strategy to normalize the data being uploaded, although the strategies, apparently tailored to the characteristics of the local expression data, are very different between the databases. They also store both the raw and normalized expression data, allowing a re-normalization using another strategy.

4.6. Data Analysis

We now examine the facilities provided by the databases for data analysis, in particular Querying/Reporting, Data Mining and Statistics and finally, Visualization. So far, no database makes use of OLAP technologies. We also indicate how the analysis tools have been integrated with the database.

Querying and Reporting. Table 11 shows the querying and reporting facilities provided by the single databases. All databases support a query tool, mostly web-based, which all operate directly on the DBMS (i.e. tight integration). The common approach, as followed by various databases, such as ExpressDB, SMD, YMD, and RAD2, is first to allow the user to select or search for the experiments of interest, and then to filter the relevant genes by specifying search criteria for the thresholds, the intervals of expression values or gene annotation information, such as name, organism, and disease. In contrast to the simple HTML-based in other databases, ArrayDB provides a comprehensive, integrated graphical tool with more interaction options for user queries.

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
Software Tools	ArrayViewer, MultiExperimentViewer (Web)	Web	Web	Java	Web	Web	Web	Web
Integration	tight integration							
Functionalities	selecting, filtering experiments, filtering genes	selecting experiments, filtering genes	selecting, filtering experiments	canned queries	filtering genes	selecting experiments, filtering genes, canned queries	selecting experiments, filtering genes	selecting experiments, filtering genes

Table 11. Querying and reporting facilities

GIMS and RAD2 allow defining and storing canned queries to answer frequently asked questions. For example, GIMS provides queries for detecting relationships of gene ex-

pression to the gene structure (distribution of introns and exons), to the location of the gene products in the cell, and to the location of the genes on chromosome.

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
Software Tools	no	proprietary	RClust, Eisen, CyberT (Web)	no	proprietary	no	XCluster (Web)	no
Integration		loose	transparent		tight		transparent	
Data mining	no	condition clustering using Pearson correlation	hierarchical, K-means, permutation-based, PCA	no	correspondence analysis, hierarchical clustering	no	hierarchical, K-means, SOM, SVD	no
Statistics	no	no	t-tests, Bonferonni correction, Bayesian variance estimation	no	no	no	no	no

Table 12. Implemented data mining and statistical methods

Data Mining and Statistics. Table 12 shows the data mining and statistical methods currently implemented in the different databases. We can observe that GeneX, SMD and M-CHIPS offer the most comprehensive facilities for data mining, allowing the user to perform various clustering methods, such as the hierarchical and K-means algorithms. While for M-CHIPS, dedicated analysis tools have been developed to operate directly on the database, i.e. tightly integrated, GeneX and SMD transparently integrate the existing clustering tools under their web interface. The user can first identify a data set of interest using the query tool and then immediately specify a data mining method to analyze the data set. The data set is automatically extracted to a file and fed to the data mining tool. ExpressDB also includes a clustering tool, which is, however in contrast to those in GeneX and SMD, only loosely integrated. The data has first to be manually exported from the database, transformed and then imported to the tool for analysis. Similar to ExpressDB, ArrayDB, GIMS and YMD do not have any integrated clustering algorithms. Unlike data mining, integrated statistical analysis has not been supported widely yet. Only GeneX has an integrated tool, CyberT, for performing different statistical tests, such as t-tests, on expression data.

	ArrayDB	ExpressDB	GeneX	GIMS	M-CHIPS	RAD2	SMD	YMD
Software Tools	ArrayViewer, MultiExperiment-Viewer (Web)	MS Excel	RClust, Eisen (Web)	proprietary (Java)	proprietary	no	XCluster, TreeView (Web)	no
Integration	tight	loose	transparent	tight	tight		transparent	
Visualization	zoomable spot map, intensity graph	cluster image	clickable dendrograms, cluster trees	graphical browsers for protein interaction	correspondence analysis biplot	-	zoomable spot map, clickable cluster maps	-

Table 13. Visualization features

Visualization. In Table 13 we present the most remarkable visualization features of the databases. Typically, the clustering tools, which are only integrated in GeneX, SMD, and M-CHIPS, also possess the functionality to visualize their results, the cluster maps. In GeneX and SMD, the maps are clickable so that the user can directly navigate from the cluster result to the genes of interest and their annotation. ExpressDB uses MS Excel to offline visualize the clustering results. GIMS provides a Java-based user interface for browsing and navigating along the protein-protein interaction network. Only ArrayDB and SMD integrate array image with gene expression analysis. Here the user can zoom to individual spots to verify intensity values and obtain other spot-related metadata.

Databases	Advantages	Drawbacks
ArrayDB	• comprehensive graphical query tool	• cDNA array-specific expression data • no local gene annotations • no integrated clustering and statistics
ExpressDB	-	• limited sample and experiment annotation • no integrated data analysis
GeneX	• transparently integrated analysis functionalities (clustering and statistical) • XML (GeneXML) as exchange format	• no local gene annotations
GIMS	• comprehensive library of canned queries	• no integrated clustering and statistics
M-CHIPS	• enforcing local controlled vocabularies in capturing user-specified annotation data • EAV-based management of sample and experiment annotation data	• no local gene annotations
RAD2	• integration of various standard vocabularies for sample annotations • canned queries	• no local gene annotations • no integrated clustering and statistics
SMD	• transparently integrated cluster analysis • materialized integration of gene annotation data and update automation	• cDNA array-specific expression data
YMD	-	• no local gene annotations • no integrated clustering and statistics

Table 14. Main advantages and limitations

4.7. Comparative Discussion

Table 14 summarizes the major advantages and drawbacks we have observed for the various systems. Most databases are able to manage expression data generated using different technologies. Exceptions are ArrayDB and SMD, which focus on cDNA microarray technology. The use of annotation data varies drastically from database to database. For sample and experiment annotation, mostly free-text fields are provided. ExpressDB, for example, uses a single description field for capturing sample and experiment annotation, respectively. A few databases however try to enforce controlled vocabularies (M-CHIPS for locally developed vocabularies to specify sample and experiment annotations, RAD2 to integrate and use standard vocabularies). Usually, a standard relational approach is employed to represent annotation data. The more flexible EAV approach is only supported by M-CHIPS and SMD. SMD integrates and locally replicates gene annotation data from some public sources. Other databases, such as GeneX, M-CHIPS, RAD2 and YMD, provide links to external sources but do not locally store gene annotation data.

The common exchange format for microarray data is the tab-delimited file. So far GeneX represents the only effort to employ XML for both data import and export, which allows to exchange both expression and annotation data. Data access is typically controlled at the experiment level. So far, no distinction has been made between annotation and expression data for access control.

Finally, the databases widely differ in their data analysis facilities. ArrayDB provides a comprehensive graphical query tool for interactive investigation of the gene expression, while other databases offer rather simple web-based query forms. Only GIMS and RAD2 support canned queries. Most databases, in particular ArrayDB, ExpressDB, GIMS, RAD2 and YMD, do not yet support clustering and statistical analysis methods. In contrast, GeneX and SMD exhibit comprehensive facilities for data analysis. In these databases, the different tools are transparently integrated under a uniform user interface, providing a relatively convenient and powerful analysis framework.

5 Conclusions and Future Work

Recently, microarray technology has emerged as a revolutionary technique in molecular biology, allowing to study the expression levels of thousands of genes simultaneously. This massive parallelism has led to an explosion of valuable data to be managed and analyzed. So far, several databases have been developed for microarray data but the current state of the art in this field is still early stage. In this paper, we comprehensively analyzed the requirements for microarray data management. We considered the various kinds of data involved as well as data preparation, integration and analysis needs. The identified requirements are then used to comparatively evaluate eight existing microarray databases described in the literature.

Based on the observed strengths and weaknesses of the current databases we can identify the following major problem areas to be addressed in future research:

- *Data Integration*: Web link-based data integration, as usually implemented in current databases, is not sufficient to support gene expression analysis. Advanced approaches such as federated or materialized integration promising more comprehensive analysis of microarray data, have not yet been investigated in this context. Their implementation poses significant challenges w.r.t. schema and data integration, schema matching, and data cleaning. While techniques from other data integration areas are likely to be useful, the specifics of the bioinformatics domain need to be considered for viable solutions, e.g. to deal with the characteristics of the public sources, such as limited query capabilities, overlapping data, use of different vocabularies etc.
- *Data Analysis*: Current databases only provide simple analysis approaches and do not sufficiently exploit annotation data thereby making only suboptimal use of the obtained expression data. For instance, the multidimensionality of expression data and the typically hierarchical nature of (annotation) dimensions have largely been ignored so far. Hence, the applicability of OLAP technologies for interactive analysis, for which various powerful tools are already available, needs to be explored. This necessitates expression and annotation data be clearly defined and modeled, which also requires further research. Moreover there is a large spectrum of data mining approaches but yet there is no systematic evaluation of their relative strengths and weaknesses w.r.t. gene expression analysis.
- *Performance Optimization*: To achieve high performance for interactive (OLAP) queries and data mining on large amounts of expression data, the use of advanced DBMS technologies such as materialized views, parallel processing, and indexing is to be evaluated. Especially for interactive data mining purposes, new approaches in these areas are likely necessary to achieve short response time.

At the University of Leipzig we have started a project to build a microarray data warehouse for local user groups that aims at taking the discussed requirements into account.

Acknowledgements
We thank the anonymous reviewers for their helpful comments. This work is supported by DFG grant BIZ 6/1-1.

References

[AK77] Alwine, J.C., D.J. Kemp et al: Method for Detection of specific RNAs in Agarose Gels by Transfer to Diazobenzyloxymethyl-paper and Hybridization with DNA Probes. Proc. National

Academy of Science 74, 1977

[AR00] Ach, J., G.M. Rindone: Systematic Management and Analysis of Yeast Gene Expression Data. Genome Research 10, 2000

[BA00] Bairoch, A., R. Apweiler: The SwissProt Protein Database and its Supplement TrEmbl in 2000. Nucleic Acids Research 28(1), 2000

[BF99] Beißbart, T., K. Fellenberg et al: Processing and Quality Control of DNA Array Hybridization Data. Bioinformatics 16:11, 2000

[BG99] Baker, P.G., C.A. Goble et al: An Ontology for Bioinformatics Applications. Bioinformatics 15(6), 1999

[BG00] Brown, M.S., W.N. Grundy et al: Knowledge-based analysis of microarray gene expression data by using support vector machines. Proc. National Academy of Science 97, 2000

[BH01] Brazma, A., P. Hingamp et al: Minimum Information about a Microarray Experiment (MIAME) – Toward Standards for Microarray Data. Nature Genetics 19, 2001

[BJ01] Ball, C.A., H. Jin et al: Saccharomyces Genome Database Provides Tools to Survey Gene Expression and Functional Analysis Data. Nucleic Acids Research 29(1), 2001

[BK02] Bono, H., T. Kasukawa et al: READ – RIKEN Expression Array Database. Nucleic Acids Research 30(1), 2002

[BR02] Brazma, A., A. Robinson, J. Vilo: Gene Expression Data Mining and Analysis. In Jordan, B. (Ed.): DNA Microarrays - Gene Expression Applications, Springer, 2002

[BV00] Brazma, A., J. Vilo: Gene Expression Data Analysis. FEBS Letters 480, 2000

[CD02] Crom, S.L., F. Devaux et al: yMGV – Helping Biologists with Yeast Microarray Data Mining. Nucleic Acids Research 30(1), 2002

[CP01] Cornell, M., N. Paton et al: GIMS – A Data Warehouse for Storage and Analysis of Genome Sequence and Functional Data. Proc. 2nd IEEE International Symposium on Bioinformatics and Bioengineering, 2001

[CW02] Cheung, K.H., K. White et al: YMD: A Microarray Database for Large-scale Gene Expression Analysis. Proc. American Medical Informatics Association 2002 Annual Symposium

[Do02] Dopazo, J.: Microarray data processing and analysis. In Lin, S.M, Johnson, K.F. (Ed.): Microarray Data Analysis II, 43-63, Kluwer Academic, 2002

[DY02] Dudoit, S., Y.H. Yang et al: Statistical Methods for Identifying Differentially Expressed Genes in Replicated cNDA Microarray Experiments. Statistika Sinica 12(1), 2002

[ED02] Edgar, R., M. Domrachev, A.E. Lash: Gene Expression Omnibus - NCBI Gene Expression and Hybridization Array Repository. Nuleic Acids Research 30(1), 2002

[ER98] Ermolaeva, O., M. Rastogi et al: Data Management and Analysis for Gene Expression Arrays. Nature Genetics 20, 1998

[FH01] Fellenberg, K., N.C. Hauser et al: Correspondence Analysis Applied to Microarray Data. Proc. National Academy of Science 98(19), 2001

[FH02] Fellenberg, K., N.C. Hauser et al: Microarray Data Warehouse Allowing for Inclusion of Experiment Annotations in Statistical Analysis. Bioinformatics 18(3), 2002

[GL01] Gardiner-Garden, M., T.G. Littlejohn: A Comparison of Microarray Databases. Briefings in Bioinformatics 2(2), 2001

[GO00] The Gene Ontology Consortium: Gene Ontology - Tool for the Unification of Biology. Nature Genetics 25, 2000

[HB01] Hill, A.A., E.L. Brown et al: Evaluation of Normalization Procedures for Oligonucleutide Array Data Based on Spiked cRNA Controls. Genome Biology 2(12), 2001

[HB02] Hubbard, T., D. Barker et al: The Ensembl Genome Database Project. Nucleic Acids Research 30(1), 2002

[Hu01] Huyn, N.: Scientific OLAP for the Biotech Domain. Proc. 27th Intl. VLDB Conference, 2001

[HW02] Haverty, P.M, Z. Weng et al: HugeIndex – A Database with Visualization Tools for High-density Oligonucleotide Array Data from Normal Human Tissues. Nucleic Acids Research 30(1), 2002

[KG00] Kanehisa, M., S. Goto et al: The KEGG Databases at GenomeNet. Nucleic Acids Research 30(1), 2000

[KM00] Kerr, M.K, M. Martin, G.A. Churchill: Analysis of Variance for Gene Expression Microarray

Data. Journal of Computational Biology 7(6), 2000

[Kn02] Knudsen, S.: A Biologist Guide to Analysis of DNA Microarray Data. Wiley-Interscience. New York, 2002.

[KS02] Kent, J., C.W. Sugnet et al: The Human Genome Browser at UCSC. Genome Research 12, 2002

[LD96] Lockhart, D.J., H. Dong et al: Expression Monitoring by Hybridization to High-density Oligonucleotide Arrays. Nature Biotechnology 14, 1996

[LL02] Liu, G., A.E. Loraine et al: NetAffx – Affymetrix Probeset Annotations. Proc. ACM SAC 2002

[LP92] Liang P., A.B. Pardee: Differential Display of Eukaryotic Messenger RNA by means of the Polymerase Chain Reaction. Science 257, 1992.

[MC01] Mangalam, H., G. Chen et al: GeneX: An Open Source Gene Expression Database and Integrated Tool Set. IBM System Journal 40(2), 2001

[Na99] Nature Genetics Supplement 21(1), 1999

[NB98] Nadkarni, P.M., C. Brandt: Data Extraction and Adhoc Query of an Entity-Attribute-Value Database. Journal of the American Medical Informatics Association 5, 1998

[Pa02] Pan, W.: A Comparative Review of Statistical Methods for Discovering Differentially Expressed Genes in Replicated Microarray Experiments. Bioinformatics 12, 2002

[PK00] Paton, N.W., S.A. Khan et al: Conceptual Modelling of Genomic Information. Bioinformatics 16(6), 2000

[PM01] Pruitt, K., D.R. Maglott: RefSeq and LocusLink – NCBI Gene-centered Resources. Nucleic Acids Research 29(1), 2001

[QC01] Quackenbush, J., J. Cho et al: The TIGR Gene Indices: Analysis of Gene Transcript Sequences in Highly Sampled Eukaryotic Species. Nucleic Acids Research 29(1), 2001

[SB00] Schuchhardt, J., D. Beule et al: Normalization Strategies for cDNA Microarrays. Nucleic Acids Research, 28(10), 2000

[SC02] Su, A.I., M.P. Cooke et al: Large-scale Analysis of the Human and Mouse Transcriptomes. Proc. National Academy of Science 99(7), 2002

[SH01] Sherlock, G., T. Hernandez-Boussard et al.: The Stanford Microarray Database. Nucleic Acids Research 29(1), 2001

[SP01] Stoeckert, C., A. Pizarro, et al: A Relational Schema for Both Array-based and Sage Gene Expression Experiments. Bioinformatics 17(4), 2001

[SS95] Shena, M., D. Shalon et al: Quantitative Monitoring of Gene Expression Patterns with a complementary DNA Microarray. Science 270, 1995

[SW95] Somogyi, R., X. Wen et al: Developmental Kinetics of GAD Family mRNAs Parallel Neurogenesis in the Rat Spinal Cord. Journal of Neuroscience 15(4), 1995

[VE98] Vasmatzis, G., M. Essand et al: Discovery of Three Genes Specifically Expressed in Human Prostate by Expressed Sequence Tag Database Analysis. Proc. National Academy of Science 95, 1998

[VZ95] Velculescu, V.E., L. Zhang et al: Serial Analysis Of Gene Expression. Science 270, 1995

[WC02] Wheeler, D.L., D.M. Church et al: Database Resources of the National Center for Biotechnology Information: 2002 Update. Nucleic Acids Research 30, 2002