Concept Instance Sketching and Design for a Biological Database Framework

Alan McCulloch[1], Pieter Demmers[2], Jason Mitchell[1], David Townley[3], Paul Smale[1], Russell Smithies[1], Craig Miskell[1], Anar Khan[1], Nauman Maqbool[1]

[1] AgResearch, Invermay Agricultural Centre [2] Crop and Food Research, Invermay / Nutrigenomics New Zealand [3] CSIRO/Sheep genomics

alan.mcculloch@agresearch.co.nz

Abstract

We developed a biological database framework based on a collection of concept instance sketches expressed using a novel extended hypergraph notation, and applied this framework to the data warehousing requirements of a number of research projects in the fields of genomics, genetics, nutrigenomics and molecular biology. The resulting framework and its instances have been implemented in a Postgres database, with a Python object oriented API and web-based interface, and a number of production systems have been built and operated using the framework, and continue to be developed.

Definition of a data framework

1. A database schema, that we know in advance will be extended and modified over the project lifetime

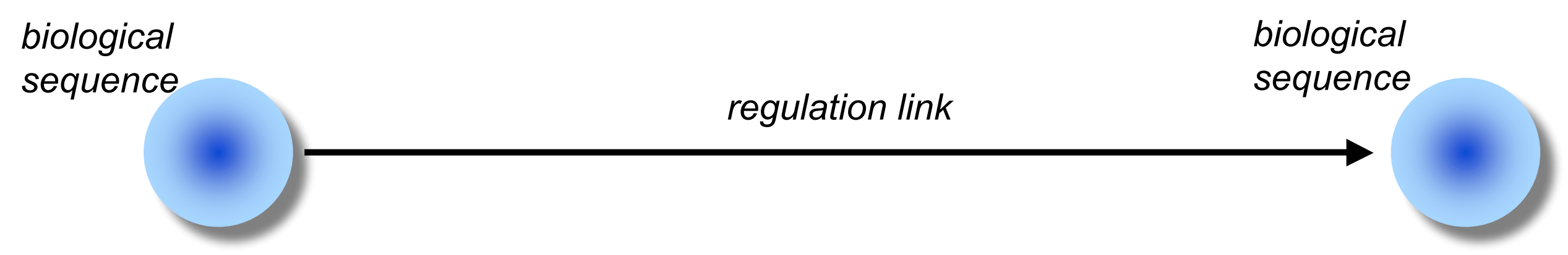
2. A set of rules for deciding which changes to the schema are permitted according to the framework.

Modelling languages cannot completely specify a data framework

Modelling languages (such as Entity Relationship diagrams) are used to formally and completely specify data models, but cannot completely specify data frameworks as these include statements outside the model itself, about what changes and extensions to the model are to be allowed.

Concept instance sketching : design by example

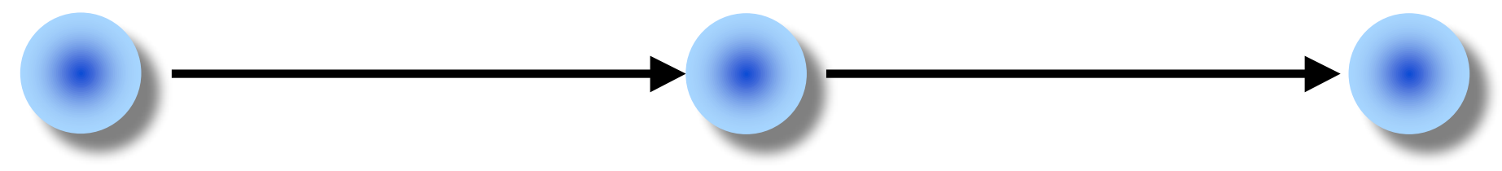
Concept instance sketching is the graphical depiction of instances of data objects and the relationships between them. :



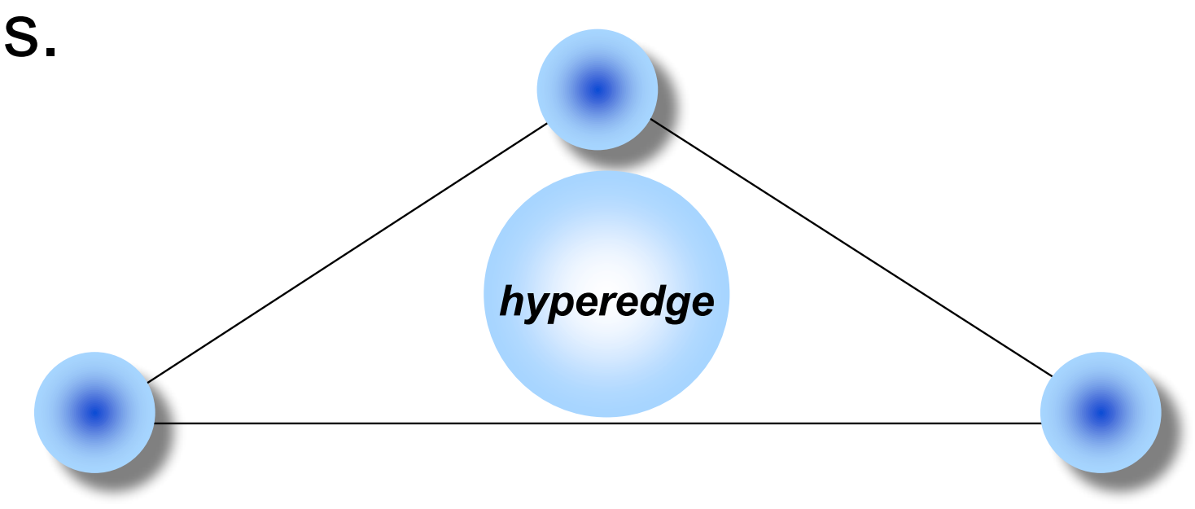
The above diagram could not be a data modelling statement, since the sequence entity appears more than once, whereas entities may only appear once in data model statement diagrams. Concept instance sketches convey the structure of the data model "by example" using pictures isomorphic to instances of objects and relations.

Inadequacy of graphs for concept instance sketching – hypergraphs.

Using a graph consisting of pairs of vertices connected by edges, we are restricted to isomorphic pictures of at most binary relationship instances. A hypergraph is a graph in which an "edge" may connect more than two vertices. This allows us to picture instances of ternary and higher relation instances.



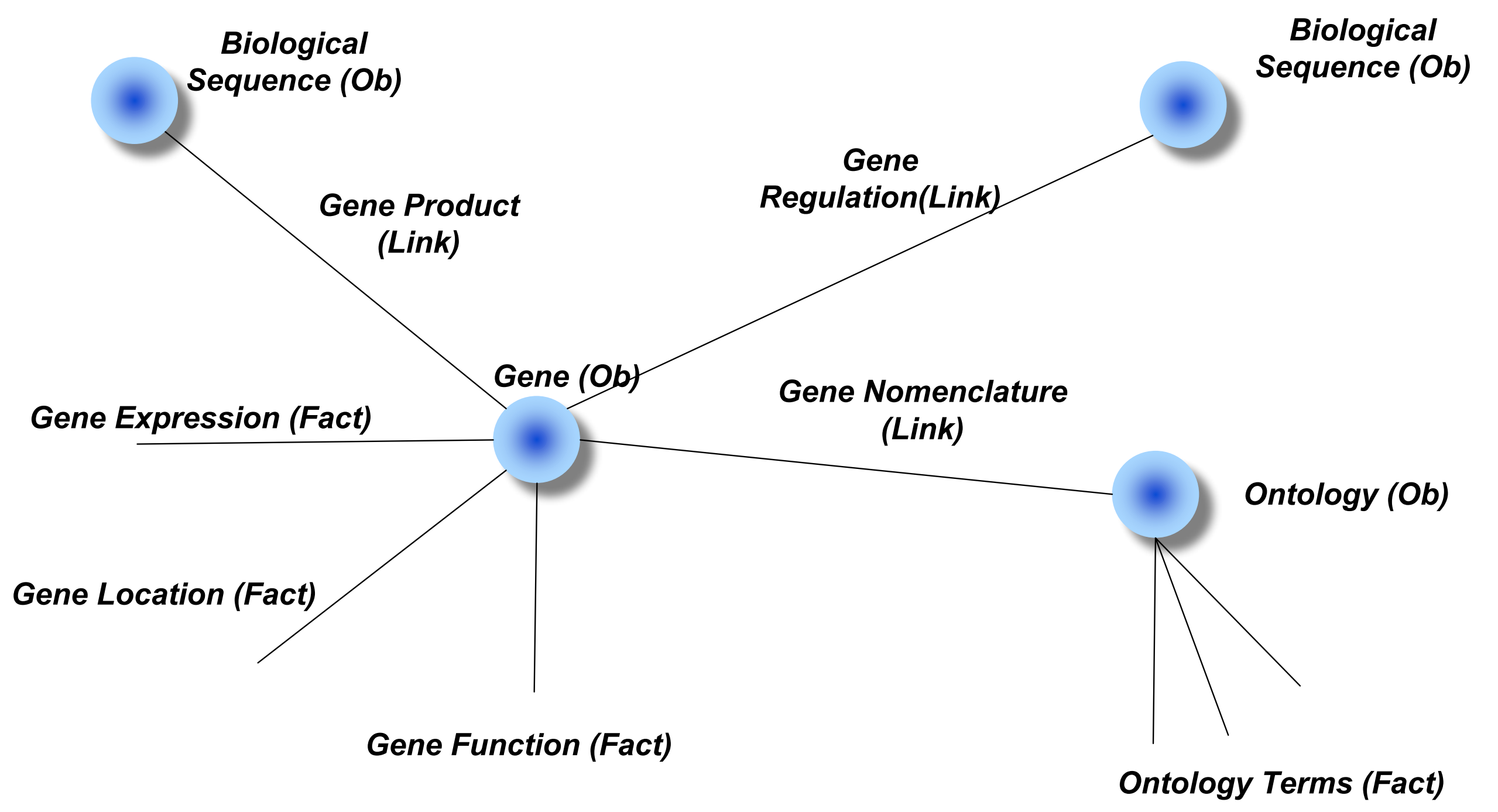
Three vertices connected by two (oriented, i.e. directed) edges in a standard graph



Three vertices connected by a single (unoriented) hyperedge in a hypergraph. We emphasize the existence of the hyperedge by drawing a circle inside the connected vertices.

Extended hypergraph notation – dangling edges for fact tables

We needed to depict instances of the typical data-warehouse star-schema design pattern, with a number of fact tables linked to a given entity. These are treated as unary relations, hence depicted by a dangling edge.



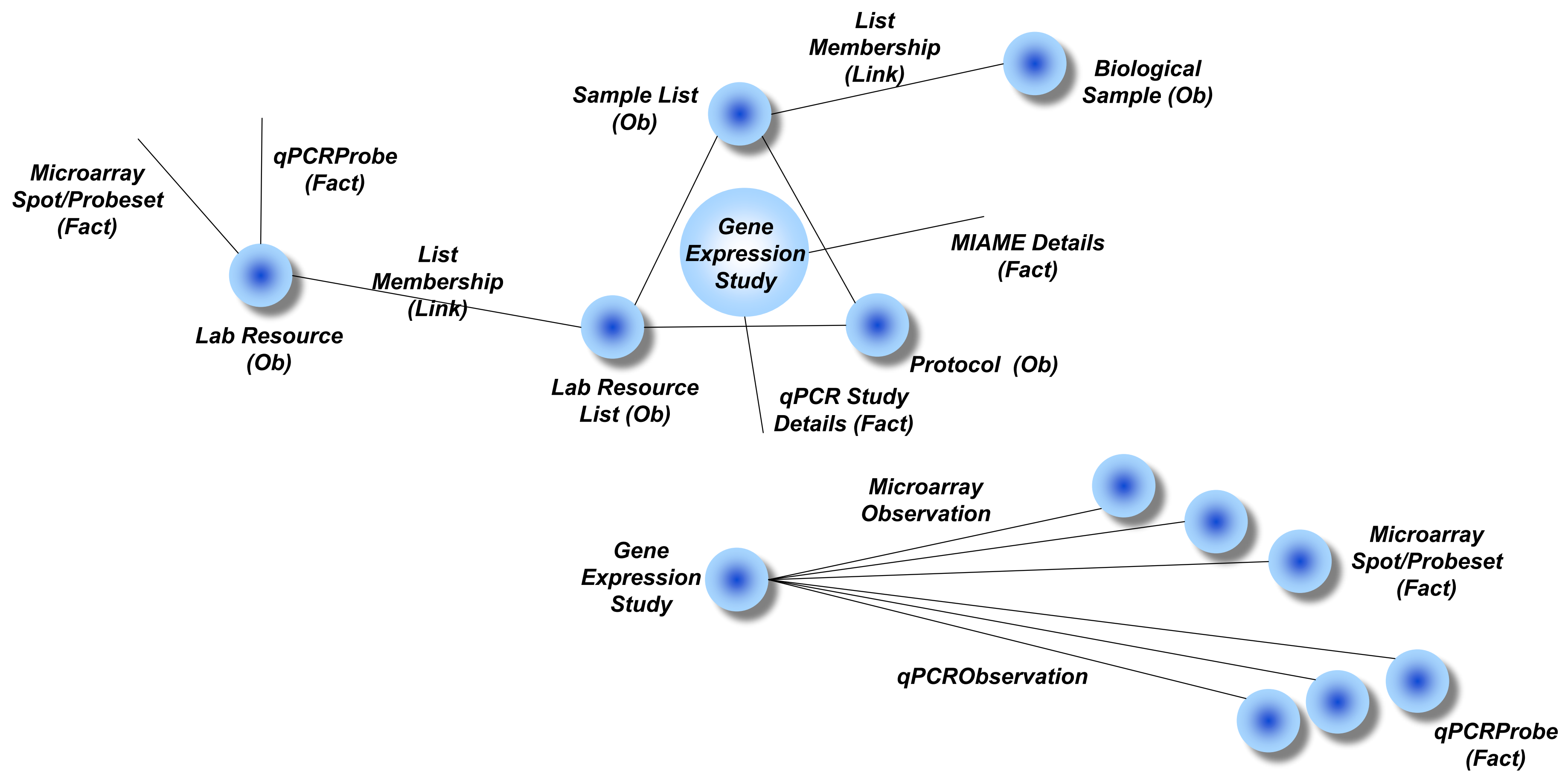
Results

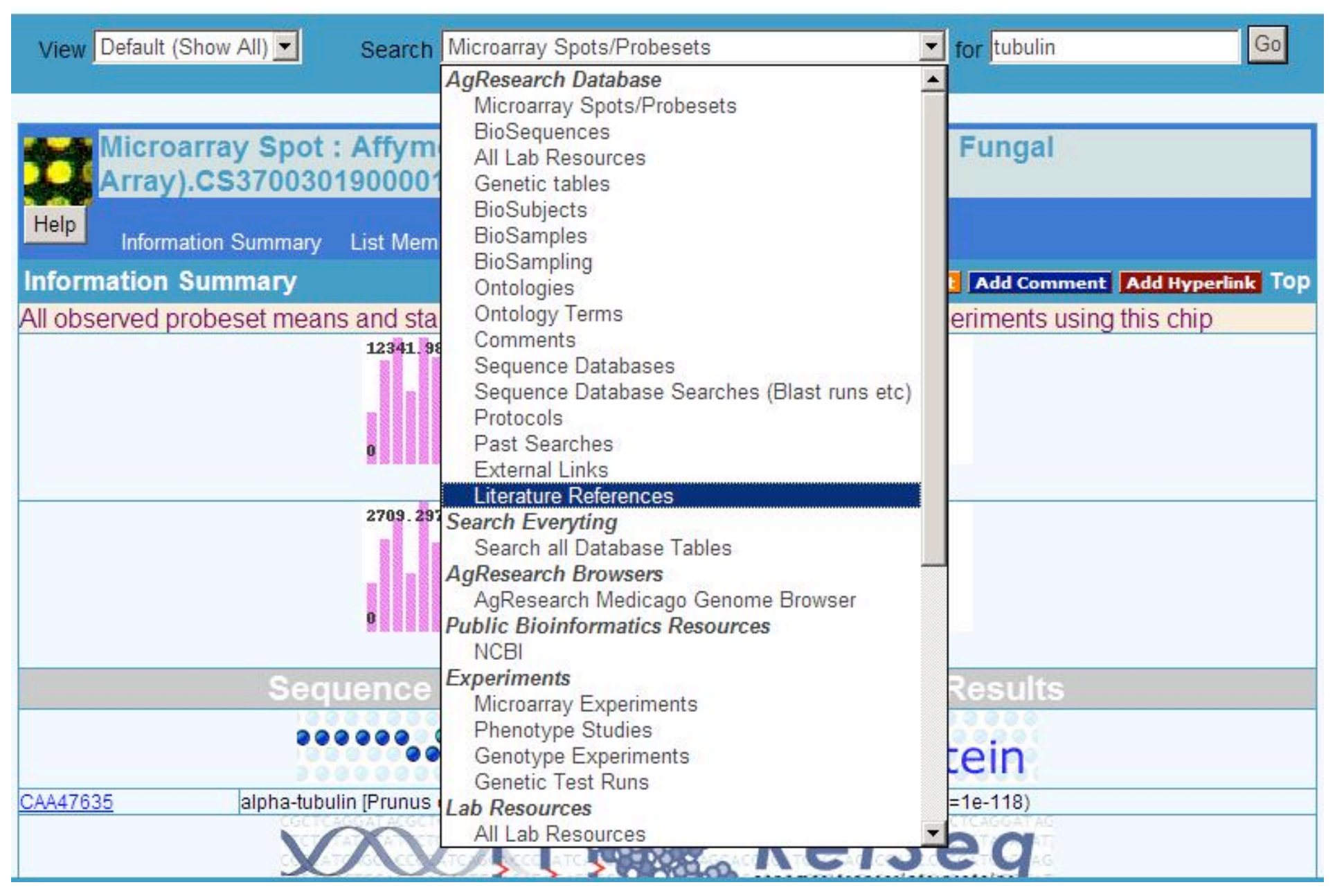
The results of concept instance sketching helped to classify the types of table to be supported by the framework by revealing common design patterns across different data domains. Any new tables added to the schema must be assignable to one of the resulting framework table classes, and table naming is determined by class. These framework rules help control and limit the complexity of the schema as it extends. The schema currently consists of some 85 tables, in the following classes

Fact tables – one data dimension per fact table. Additional fact tables added as needed	29
Link tables – associate 2 entities (examples : predicate link to record is-a and other subject-object relations ; list membership link to record membership of lists)	16
Study tables – examples include gene expression study, genotype study, in-silico study (blast run). Ternary or higher master relation between a protocol, and usually a sequence or sample or biological database, and a lab resource list.	5
Observation tables – an observation is a relation between a study and the experimental units observed. Gene expression observation (a study connected to a spot or probeset) ; genotype observation (a study connected to a genetic test) ; in-silico observation (a study connected to a blast-hit sequence)	6
Function Tables – associate 3 or more entities. For example, sequencing function associates a sequence, a sample and a list of lab resources	5
Ob tables – basic entities in the database – biological subjects, samples, sequences, protocols etc.	20
Framework – tables to support the API and runtime, including a data dictionary	4

Concept instance example – gene expression study & observations

Each distinct hyperedge, including dangling edges, is realised as a table in the schema. Thus the diagram below would be realised as 13 database tables. (Note that multiple instances only require a single table.)





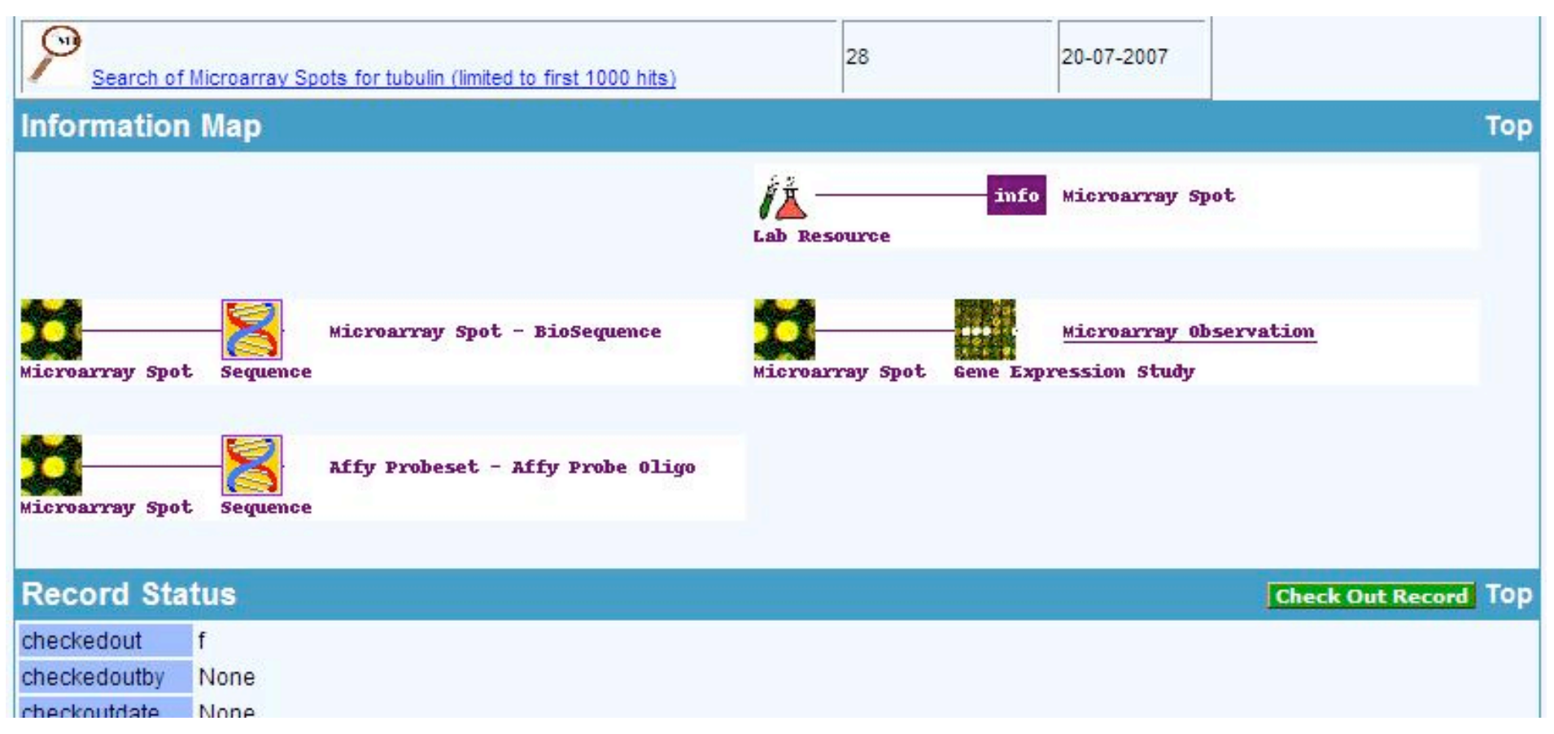
View Default (Show All) Search Microarray Spots/Probesets for tubulin Go

Microarray Spot : Affym… Fungal Array).CS3700301900001…

Information Summary

All observed probeset means and sta… …eriments using this chip

CAA47635 alpha-tubulin [Prunus … =1e-118)

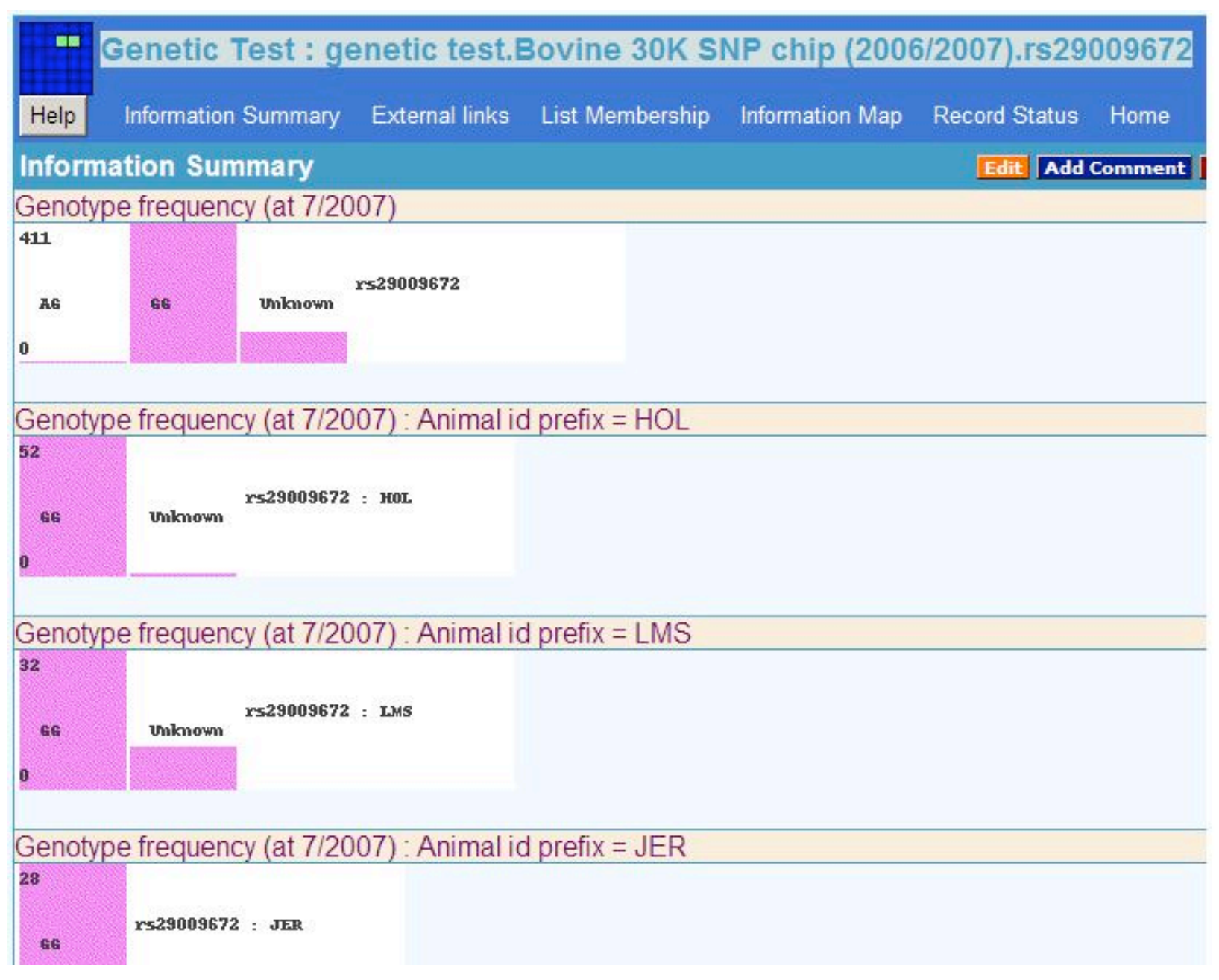


Search of Microarray Spots for tubulin (limited to first 1000 hits) | 28 | 20-07-2007

Information Map

Record Status

checkedout f
checkedoutby None
checkoutdate None



Genetic Test : genetic test.Bovine 30K SNP chip (2006/2007).rs29009672

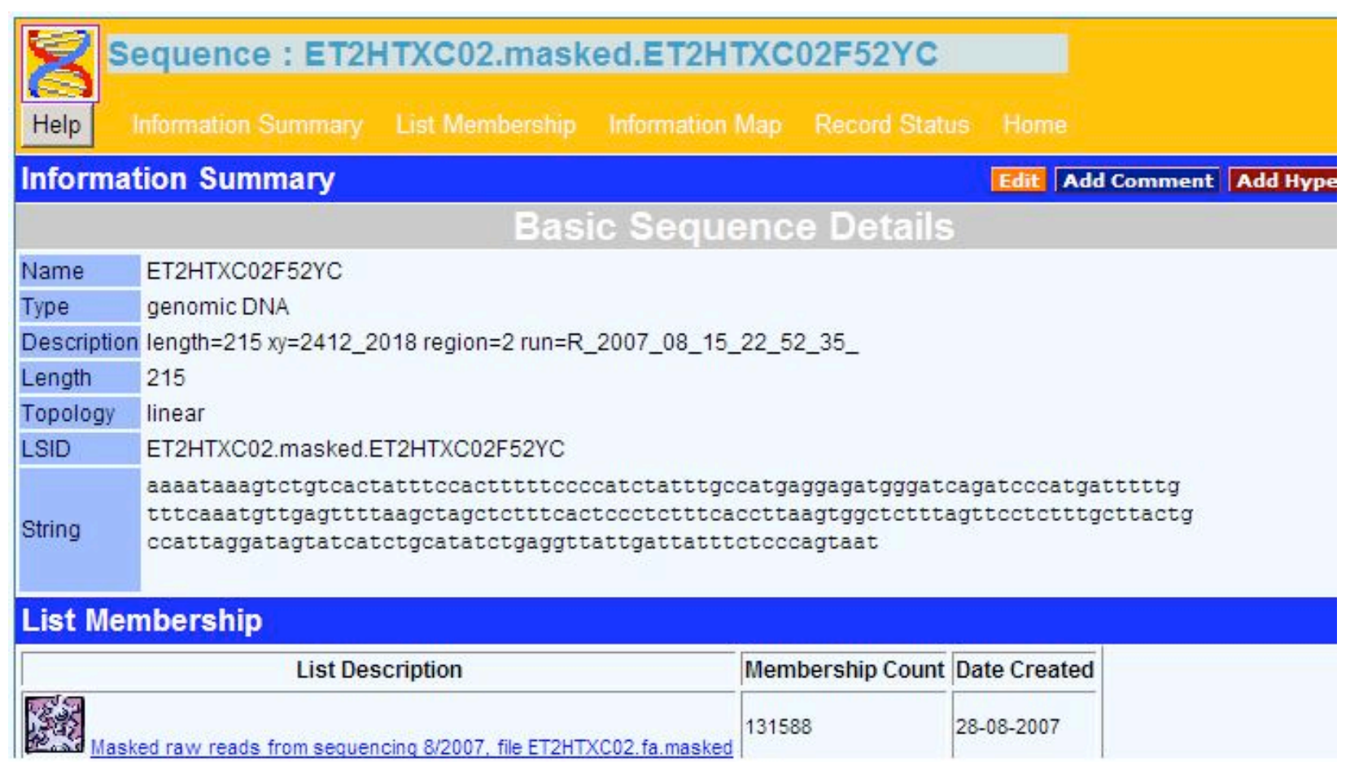
Information Summary

Genotype frequency (at 7/2007)

Genotype frequency (at 7/2007) : Animal id prefix = HOL

Genotype frequency (at 7/2007) : Animal id prefix = LMS

Genotype frequency (at 7/2007) : Animal id prefix = JER



Sequence : ET2HTXC02.masked.ET2HTXC02F52YC

Information Summary

Basic Sequence Details

Name	ET2HTXC02F52YC
Type	genomic DNA
Description	length=215 xy=2412_2018 region=2 run=R_2007_08_15_22_52_35_
Length	215
Topology	linear
LSID	ET2HTXC02.masked.ET2HTXC02F52YC
String	aaaataaagtctgtcactatttccacttttccccatctatttgccatgaggagatgggatcagatcccatgatttttg tttcaaatgttgagttttaagctagctctttcactccctctttcaccttaagtggctctttagttcctctttgcttactg ccattaggatagtatcatctgcatatctgaggttattgattatttctcccagtaat

List Membership

List Description	Membership Count	Date Created
Masked raw reads from sequencing 8/2007, file ET2HTXC02.fa.masked	131588	28-08-2007

Summary and future work

We have developed a simple concept instance sketching technique which we have found very useful in solving a difficult data modelling problem. Because our diagrams are not statements in a modelling language, the opportunity for formalism is probably limited. Nevertheless it would be desirable to define the technique more formally than we have done.

Our main focus for future software development is in further development of the database browser user interface, mainly in the direction of providing single integrated views of data and images; and in developing additional data extract reports and extending current reports.