

---

# **Biological Database Design**

## **Week 4**

---

Winter '04

Melanie Nelson, Ph.D.

---

# Gene Expression Data

- Gene expression can be measured many different ways
  - Libraries of Expressed Sequence Tags (ESTs)
  - Microarray experiments
  - Taqman (PCR-based method)
- Scientific output is similar
  - Which genes are expressed in X cell type under Y condition?
- Technical data content is quite different

---

# Gene Expression Arrays

- Spotted arrays
  - cDNAs are deposited, or spotted, onto slide
  - Common in custom arrays
- DNA chips
  - Oligonucleotides representing genes are synthesized onto slide
  - Oligonucleotides are referred to as “probes”
  - E.g., Affymetrix chips

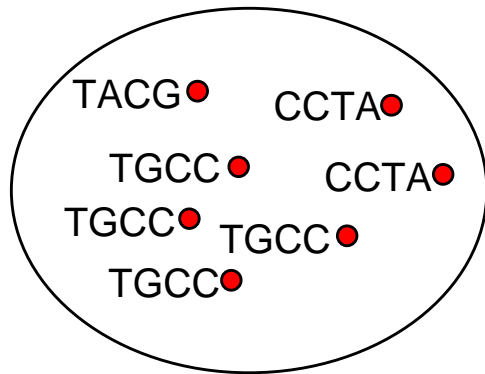
---

# Gene Expression Arrays

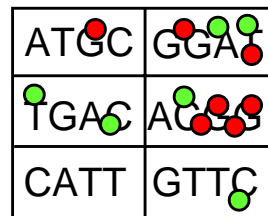
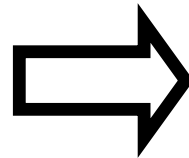
- There are technical limitations with reading absolute levels of RNA in the sample
- Arrays are most often used to compare two or more samples
  - Comparing relative expression in samples circumvents the technical limitations
  - Differences in gene expression are reported
    - Normal vs. Disease
    - Time course

# A Typical Gene Expression Array Experiment

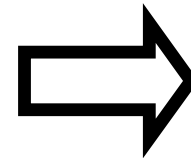
cDNA produced from the mRNA isolated from your normal sample



Process of binding your sample cDNA to the microarray is called *hybridization*

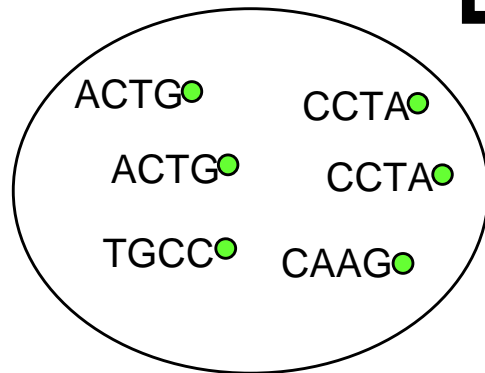


A grossly over-simplified array

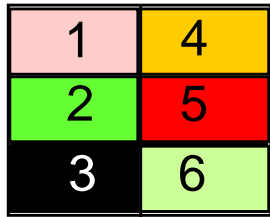


An image representing the relative abundances of the fluorophores at each spot

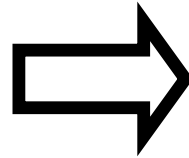
cDNA produced from the mRNA isolated from your disease state sample



# Expression Array Data Processing



1	4
2	5
3	6



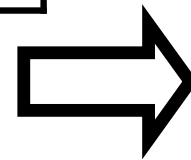
Array is scanned,  
producing a *raw  
image file*

	Mean green fluorescence intensity	Mean red fluorescence intensity	Median green fluorescence intensity	Median red fluorescence intensity
Spot 1	0.01	0.5	0.01	0.48
Spot 2	1.5	0.005	1.23	0.01
Spot 3	0.05	0.03	0.01	0.03
Spot 4	0.9	0.85	1.0	0.92
Spot 5	0.04	2.3	0.01	3.1
Spot 6	0.75	0.005	0.83	0.01

The raw image file is converted to numerical representations of fluorescence (*quantification matrix*) using image processing software.

# Expression Array Data Processing

	Mean green fluorescence intensity	Mean red fluorescence intensity	Median green fluorescence intensity	Median red fluorescence intensity
Spot 1	0.01	0.5	0.01	0.48
Spot 2	1.5	0.005	1.23	0.01
Spot 3	0.05	0.03	0.01	0.03
Spot 4	0.9	0.85	1.0	0.92
Spot 5	0.04	2.3	0.01	3.1
Spot 6	0.75	0.005	0.83	0.01



The data in the quantification matrix for each gene is combined and normalized, producing the **gene expression matrix**.

	Normal express.	Disease express.
Gene 1	1.2	0.03
Gene 2	0.03	4.6
Gene 3	0.01	0.05
Gene 4	1.5	1.1
Gene 5	5.1	0.0005
Gene 6	0.1	1.8

---

# Expression Array Data Analysis

- Will often run same experiment more than once, and combine results from multiple experiments
- There are a wide range of algorithms and software packages for image processing and data analysis
- Gene expression matrix is still “just data”
  - It is usually subjected to various statistical analyses and correlated with gene and pathway information to produce some novel biological insight



---

# Gene Expression Array Data Analysis

- The scanned image is subjected to extensive analysis
  - Normalization, noise reduction, etc.
  - Statistical analyses looking for patterns in data
- Data must be correlated with the genes represented by the probes on the array
  - Raw data just says that the gene represented by probe 145678\_at is expressed at a higher level in the disease state
- Scientist brings all of this together to answer the original biological question
  - What potential drug targets are upregulated in this type of cancer?

---

# Types of Data to Track

- Sample data
  - Cell line/tissue type
  - Experimental conditions
- Array design
  - Map probes to spots
- Gene data
  - Map genes to probes
  - Integrating additional data about gene adds value
- Expression data
  - The results of the experiment
  - Raw data, processed data, and processing steps

---

# Types of Data to Track

- Gene expression databases will almost always involve data integration
  - Scientists want to link to public data about gene function
- Not all gene expression databases will track all types of data
  - If using commercial arrays, details of array design may be proprietary
  - No consensus on need to store raw scanned image

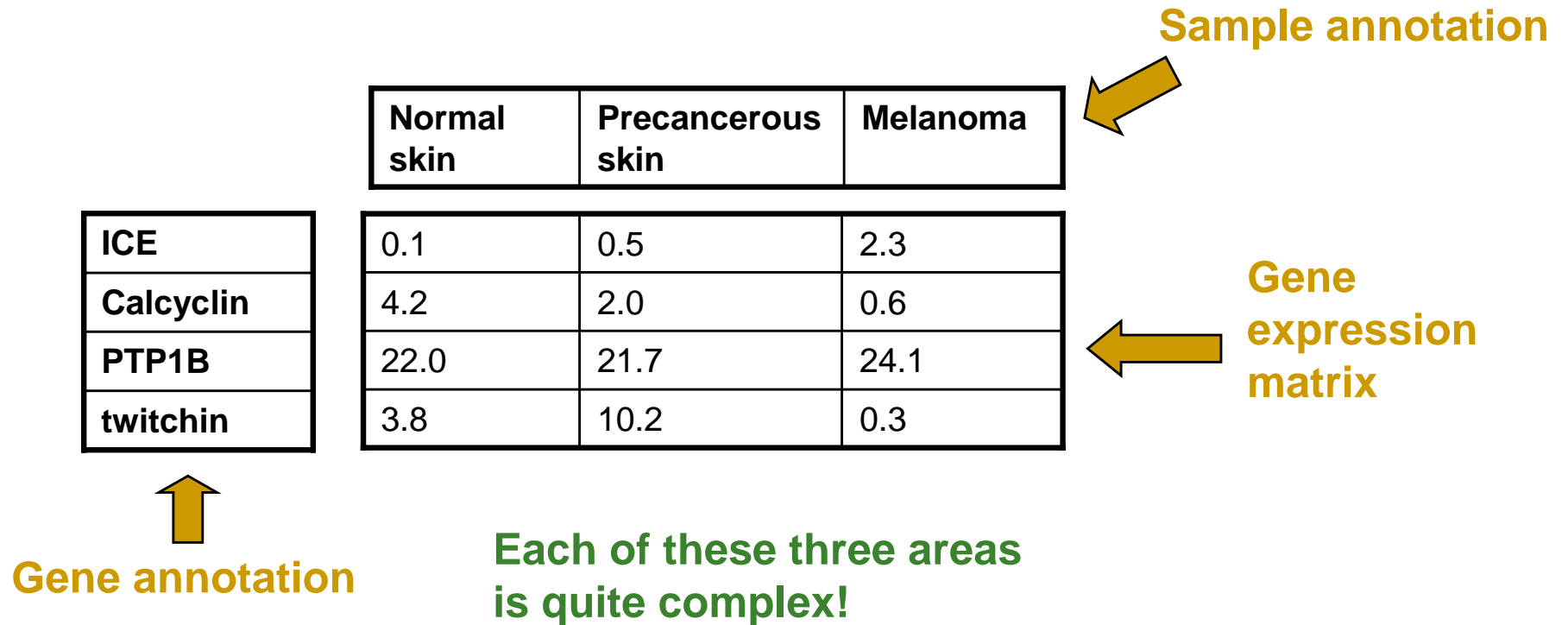
---

# MIAME

- **M**inimum **I**nformation **a**bout a **M**icroarray **E**xperiment
- Information requirements produced by the Microarray Gene Expression Database consortium (MGED; [www.mged.org](http://www.mged.org))
- Specified minimal information about a microarray experiment “required to ensure that microarray data can be easily interpreted and that results derived from its analysis can be independently verified”
- Excellent paper describing MIAME: Brazma, et al., Nature Genetics, 2001, 29:365-371.

# MIAME

- Divides microarray experiment data into three logical parts
  - Gene annotation
  - Sample annotation
  - Gene expression matrix



---

# Six Parts of MIAME

- Experimental Design **Sample Annotation**
  - Describes overall construction of experiment, which may include multiple arrays
- Array Design **Gene Annotation**
  - Describes the elements (spots) on an array
- Samples **Sample Annotation**
  - Describes the biological samples from which mRNA was isolated
- Hybridizations **Sample Annotation**
  - Describes the experimental protocol and parameters used to perform the hybridization between the array and the sample
- Measurements **Gene Expression Matrix**
  - The data from the hybridization
- Normalization Controls **Gene Expression Matrix**
  - Describes the normalization procedures used to process the data

---

# What MIAME Doesn't Cover

- Biological conclusions drawn from analyses of expression data
  - “Genes X, Y, and Z correlate with lymphoma, while genes A, B, and C correlate with leukemia”
  - “Genes D, E, and F are coregulated, and seem likely to be part of the same pathway”
- MIAME attempts to provide sufficient metadata to allow other researchers to analyze the gene expression array data and reach their own conclusions

---

# MIAME: Experimental Design

- Type of experiment
  - Time course, dose-response, normal vs. disease
  - Experiment = one or more hybridizations
  - Multiple hybridizations address a common biological question (e.g., “What genes are upregulated by the presence of my drug candidate?”)
- List of experimental variables
  - What conditions/parameters changed among different hybridizations?
- Basic quality control information
  - Usage of replicates, how non-specific hybridization is handled
- Experimental design: which arrays were hybridized to which samples
- Experiment title and free format description
- Housekeeping info: author, contact info, citations



---

# MIAME: Array Design

- The genes represented on the array
  - Associate specific DNA sequences with spots or cells on array
  - For commercial arrays, will usually just have the name of the gene represented (details are proprietary)
- The type of array
  - Type of surface: glass, membrane
  - Type of element: synthesized oligonucleotide, PCR-amplified cDNA
- Information is provided once for a particular type of array
- Chip providers supply info for their chips

---

# MIAME: Samples

- Source of sample
  - Species, cell type, etc.
- Biological treatments
  - Lab protocol information
- Technical details
  - How cDNA was extracted and labeled
  - Likely to be a standard protocol

---

# MIAME: Hybridizations

- Free text description of hybridization protocol
- Specific parameters
  - Hybridization solution
  - Blocking agent
  - Wash procedure
  - Quantity of target
  - Hybridization time
  - Volume
  - Temperature
- Description of hybridization instruments

---

# MIAME: Measurements

- Raw images of original scans of hybridized arrays
  - Raw images are primary data
  - Inclusion is controversial, due to large size of image files
- Microarray quantification matrices (results of image analysis)
  - One matrix per array: array elements vs. various quantifications
  - Include information about image processing software and description of image processing steps
- Final gene expression matrix (results of basic analyses such as normalization)
  - Summarized data: a matrix of expression level for each gene in each sample
  - Specify calculations used to produce expression levels
  - Inclusion of reliability indicators (e.g., standard deviations) is encouraged

---

# MIAME: Normalization Controls

- Normalization strategy (subset of genes used)
  - Total array
  - Limit to “housekeeping genes”, since these are assumed to have constant expression level
  - Use gene for which RNA is added (or “spiked”) into samples
- Normalization algorithm
  - Total intensity, ratio-based, linear or non-linear regression
- Identity and location of any array elements used as controls
  - Include purpose of controls
  - Specify how controls are included in samples

---

# MIAME Design Principles

- Searching for balance between support for deep queries provided by structured data and the need to support a still evolving field
- Use ontologies where possible
  - For example, taxonomy is from the NCBI
  - MGED Ontology Working Group to develop new ontologies
- Heavy use of “qualifier, value, source”
  - Similar to “entity, attribute, value” lists
  - Qualifier = entity
  - Attribute is replaced by source of value

---

# MIAME Design Principles

- Examples qualifier-value-source
  - Use of Gray's anatomy to define a cell type
    - Qualifier = cell type
    - Value = epithelial
    - Source = Gray's Anatomy (38<sup>th</sup> ed.)
  - Use of the Medical Subject Headings (MeSH) to define a disease state
    - Qualifier = disease state
    - Value = Diabetes Mellitus, Type II
    - Source = MeSH

---

# Gene Expression Database Design

- Gene expression databases can be separated into subsets
  - Gene annotation
  - Sample annotation and experimental details
  - Array design
  - Image processing and statistical analysis
  - Gene expression matrix
- Not all databases will need to store all types of data
  - Many companies/labs use only commercial arrays: commercial LIMS software will track array design



---

# Gene to Probe Relationship

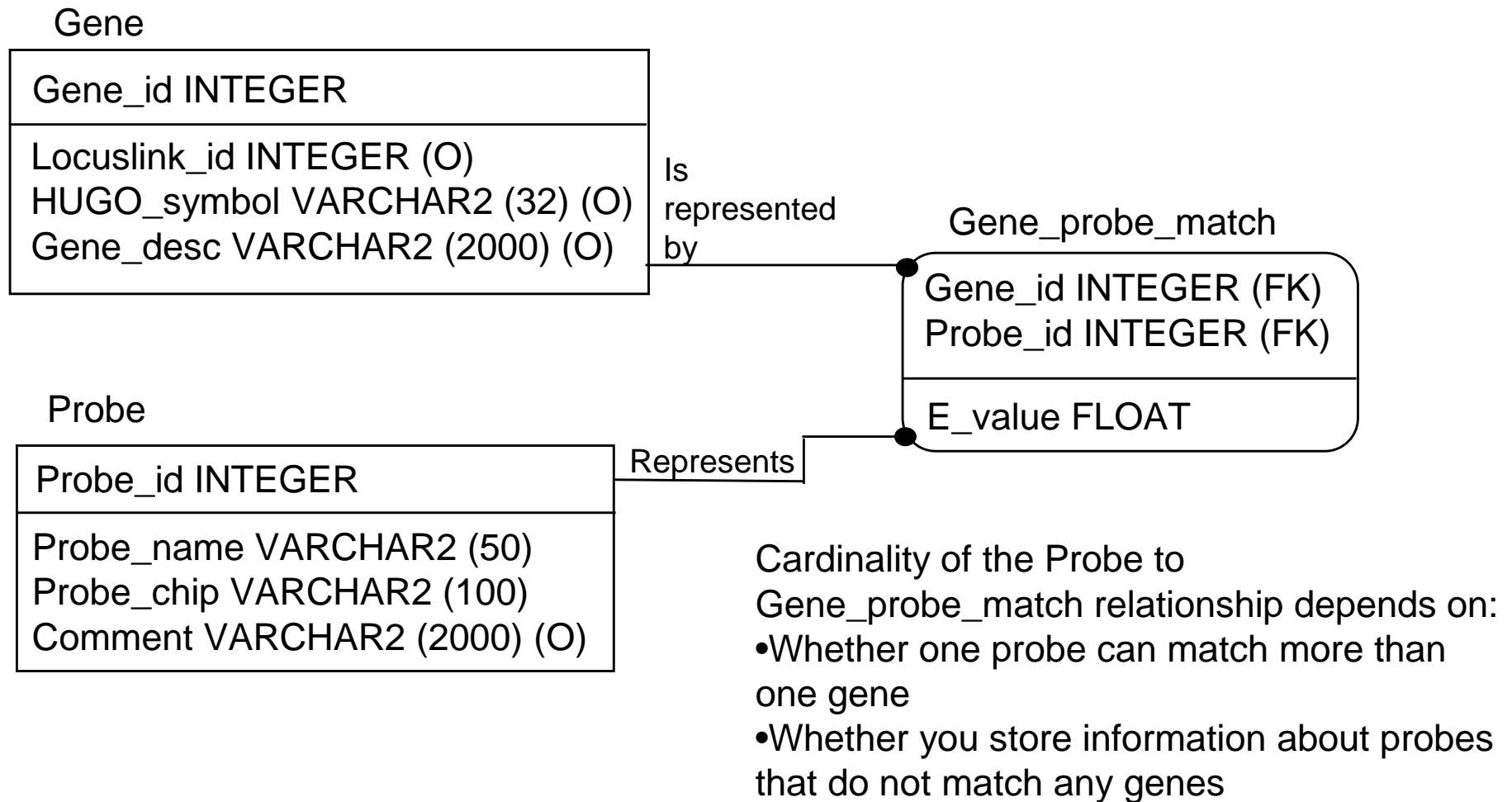
- Link between commercial probes and genes is an extremely common thing to store in a database
- Often part of a more general “gene data” database
  - Integrate available info about genes
  - Store Affymetrix probes that correspond to genes
- Forming the link between probes and genes requires scientist input
  - Usually done by sequence comparison
  - Scientists must decide how much sequence identity is required for a “match”

---

# Gene to Probe Relationship

- Depending on rules scientists define, relationship between genes and probes is one-to-many or many-to-many
  - One gene is often represented by more than one probe
  - Probe is usually intended to be specific for a single gene, but in practice may match more than one gene
- I favor using the many-to-many design
  - Some probes may not match any genes
    - Depends on your rules for storing gene info and for matching probes to genes
  - Match between gene and probe can have attributes
    - BLAST E-value, or other statistics
  - May not be appropriate if a one gene per probe rule exists
    - However, a trigger can be used to enforce this rule

# Gene to Probe Relationship



---

# Sample and Experiment Annotations

- As MIAME recognizes, it is not possible to define a fixed set of sample and experiment annotations
  - Field is relatively young, and rapidly evolving
  - Technique can be applied to many different types of questions
- A lab or company may work on only certain types of sample or perform only certain types of experiments
  - Be suspicious of this! Research directions change.
- I favor using entity-attribute-value type design
  - In final application, this will usually appear as user-extensible lists

# Sample and Experiment Annotation

Sample

Sample_id INTEGER
Species_id INTEGER (FK)
Sample_class_id INTEGER (FK)

This design allows only one class per sample. It may be necessary to instead allow a many-to-many relationship between Sample and Sample\_class

Sample\_class

Sample_class_id INTEGER
Sample_class_name VARCHAR2 (50)
Sample_class_desc VARCHAR2 (2000)
Sample_class_source VARCHAR2 (200)

Sample\_class\_source can be used to identify an ontology or controlled vocabulary from which the classification is taken. In some databases, it might be broken out into its own table, because additional information needs to be stored about the source.

# Sample and Experiment Annotation

## Hybridization

Hybridization_id INTEGER
Array_id INTEGER (FK) Sample_id INTEGER (FK) Hybridization_desc VARCHAR2 (2000)

## Parameter\_type

Parameter_type_id INTEGER
Parameter_type_name VARCHAR2 (200) Parameter_type_desc VARCHAR2 (2000)

In general, a hybridization should not be stored without parameters.

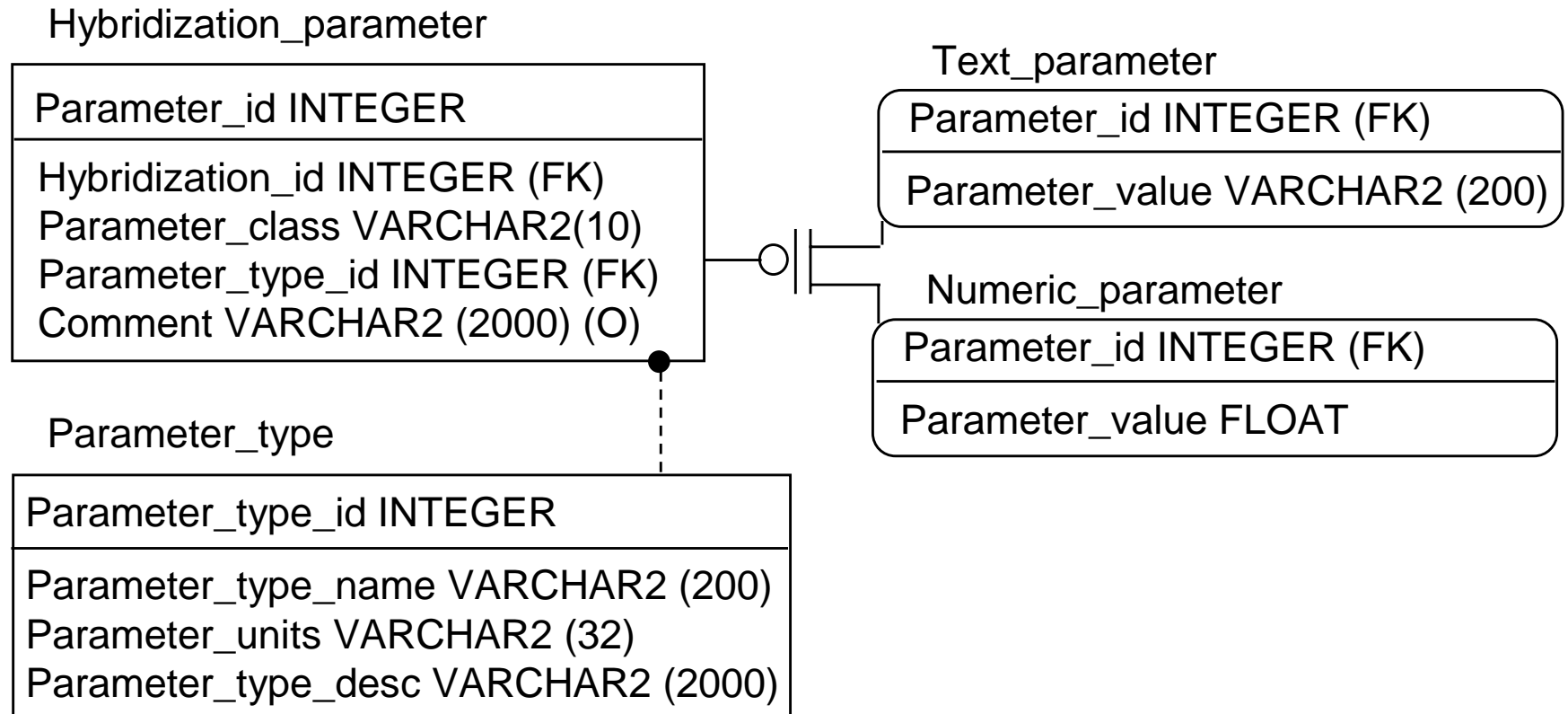
Hybridization_Parameter
Parameter_id INTEGER
Hybridization_id INTEGER (FK) Parameter_type_id INTEGER (FK) Parameter_value VARCHAR2 (200) Comment VARCHAR2 (2000) (O)

P

Let a parameter\_type exist without being used in a hybridization\_parameter: this allows you to “preload” some common types.

# Sample and Experiment Annotation

- Two broad classes of hybridization parameters:
  - Numeric (pH, temperature)
  - Text (hybridization buffer)



---

# Array Design

- Array design data model is quite complicated
  - Probes usually appear in more than one place
  - Different manufacturers have different chip designs
- Many people use database provided by chip manufacturer
  - Also provides data model for image processing and normalization steps, which are also dependent on the system in use
  - Data comes out into a more general DB at gene expression matrix stage
- Stanford Microarray Database (SMD) also has a data model for array design



---

# Gene Expression Data Models

- MAGE-ML
  - XML-based language for microarray gene expression data
  - Based on MIAME
  - Spellman, et al., Genome Biology, 2002, 3: research0046.1-0046.9
  - [www.mged.org/Workgroups/MAGE/mage-ml.html](http://www.mged.org/Workgroups/MAGE/mage-ml.html)
- Stanford Microarray Database (SMD)
  - Not yet MIAME compliant
  - Data model is online:
    - [genome-www5.stanford.edu/schema/Schema.html](http://genome-www5.stanford.edu/schema/Schema.html)
  - Longhorn Array Database is an open-source, MIAME compliant implementation of the SMD
    - [www.longhornarraydatabase.org](http://www.longhornarraydatabase.org)

---

# Gene Expression Data Models

- Affymetrix Analysis Data Model (AADM)
  - Data model for storing Affymetrix experimental results
  - Data model and detailed data dictionary are online
  - [www.affymetrix.com/support/developer/aadm/content.affx](http://www.affymetrix.com/support/developer/aadm/content.affx)
- NCBI's Gene Expression Omnibus (GEO)
  - MIAME compliant
  - NCBI Handbook chapter describes basic data model

---

# Laboratory Information Management Systems

- LIMS databases vary widely in topic and depth
  - Some are very detailed and specific to a certain subject area
    - Affymetrix's GCOS system for microarray data
  - Others are more general, and attempt to address many different types of laboratory samples and results
    - LabVantage Sapphire for Life Sciences

---

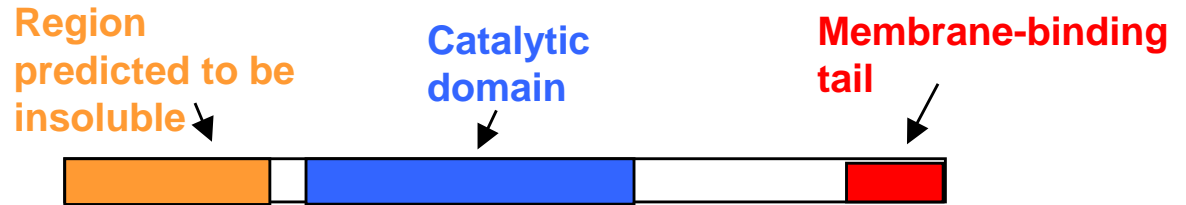
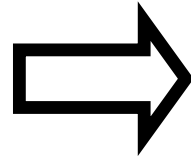
# Laboratory Information Management Systems

- **WARNING!** Laboratory processes and database designs shown here are for instructional purposes only!
  - Intended to demonstrate some of the issues to think about if designing a LIMS database
  - Designs are incomplete, and there is no guarantee that they would meet the requirements of the lab for which you are designing the LIMS

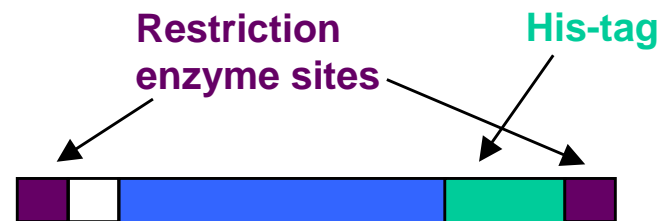
# Protein Expression Process



Scientist identifies sequence for gene that codes for the protein of interest



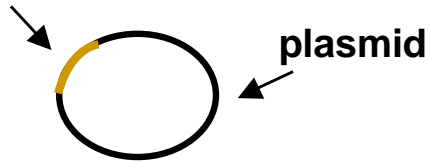
Sequence is inspected, and scientist uses several different criteria to select region to express



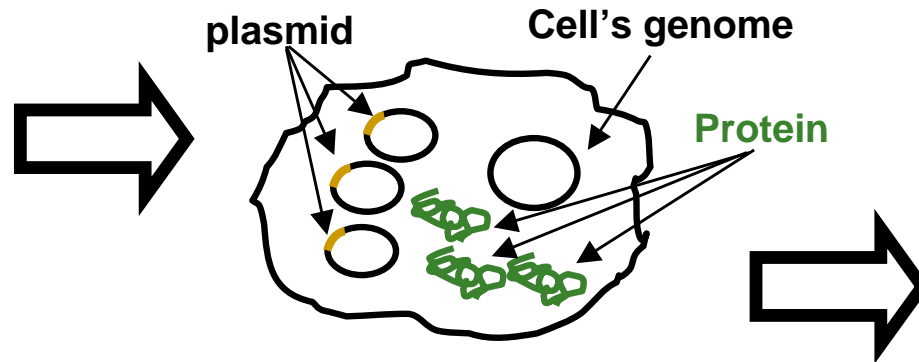
Other useful bits of sequence might be added, to aid in purification (e.g., a His-tag), or to allow the sequence to be inserted into a plasmid (restriction enzyme sites). The final sequence is called a **construct**.

# Protein Expression Process

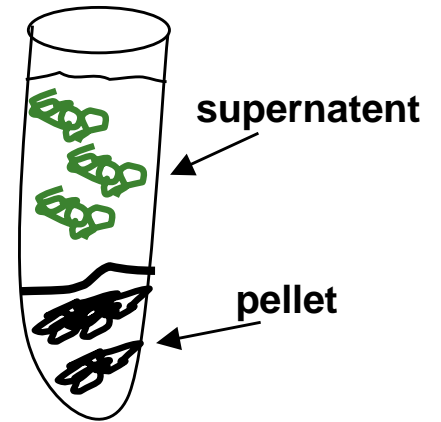
## Gene construct



Construct for gene representing protein of interest is cloned (copied) into a plasmid, or expression vector

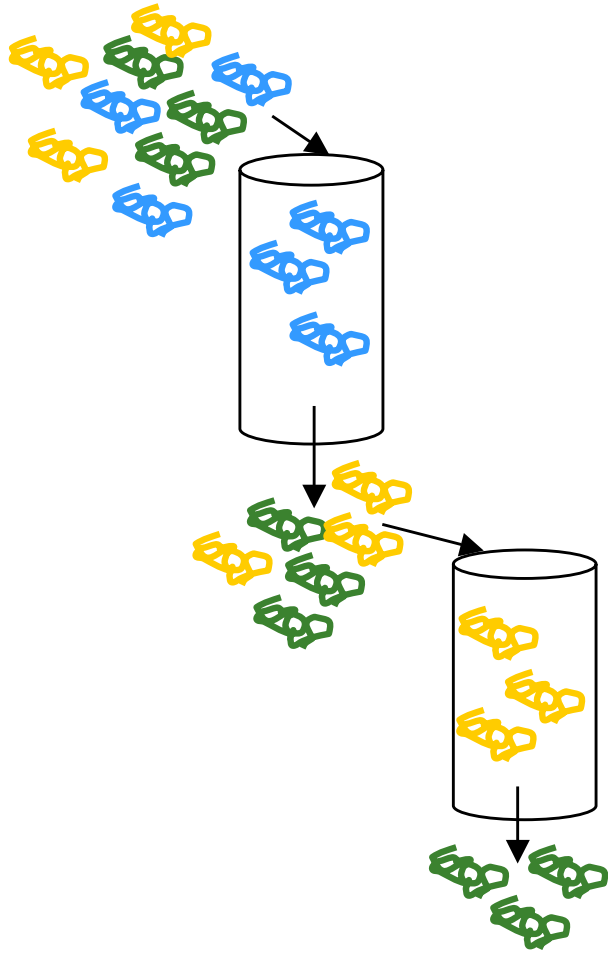


Plasmid is introduced into host cell. The host cell's protein production machinery translates the gene on the plasmid into a protein



Host cells are grown up to high density, then the cell membranes are disrupted, and the sample is put in a centrifuge. Membranes and other insoluble things are in the pellet. Your protein of interest is (hopefully!) in the supernatant.

# Protein Expression Process



The purified protein is stored in a stock solution or as lyophilized powder, or used directly in various types of experiments.

The supernatant is passed over a series of columns to purify the protein of interest away from the other soluble proteins

---

# Components of LIMS Databases

- Sample tracking
  - Links to notebooks (electronic or paper)
  - Storage details (location, sample state, etc.)
  - Track state of sample
    - Plasmid, raw cell pellet, purified protein
- Data capture
  - Incorporate data generated by lab machines
    - Automatic: data flows directly from machine
    - Semi-automatic: data file from machine is fed into database via scripts
  - Manual data entry may be appropriate for qualitative data



---

# Components of LIMS Databases

- Sample analysis
  - Capture results of analysis steps
  - Track which analyses have been performed on which samples
  - May need to allow qualitative ratings of data
    - Example: scientists view binding curve from which IC50 data is calculated and rate its quality
- Audit Trail
  - Some LIMS track who enters, modifies, and approves data, and when each event occurs
  - Required for validated systems

---

# Challenges in LIMS Design

- Research process tracking and standardization is implicit in all components
  - The change to research processes is often one of the most disruptive and difficult parts of a LIMS project
  - Crucial to involve lab scientists in design decisions
- In certain situations, LIMS must be validated, and adhere to 21 CFR Part 11
  - Greatly complicates database design: must be able to certify no unauthorized changes to the data and certify that all transformations are correct
  - Many commercial products are 21 CFR Part 11 compliant
  - I'm not going to cover validated systems

---

# Sample Tracking

- Can have generic “sample” or specific sample types
  - Decision depends on scope of database
- If specifying sample types, work with scientists to determine the appropriate
  - Name carefully, and document meanings
    - Has “raw pellet” been washed?
  - Pay attention to real experimental process
    - Be sure to allow flexibility where the science requires it
    - Perhaps some “raw pellets” must be washed, and other must not be

---

# Sample Tracking

- Example: database to track protein expression efforts
  - Taken from SPINE, and from proprietary databases
- Many different constructs may be used to express a protein
  - Try different lengths and termini in attempt to find soluble, active construct
  - SPINE even allows use of “same” protein from another species (orthologs)
- Measurements should attach to the construct
  - Subtle changes can have profound impact on biophysical measurements such as NMR spectra, solubility measurements, etc.
  - Biological measurements may be less sensitive to details of construct, but knowing which construct was used for a specific experiment can help track down contradictory results

# Identifying Biological Content of Sample

## Protein

Protein_id INTEGER
Protein_name VARCHAR2 (200)
Primary_researcher_id INTEGER (FK)
Therapeutic_area_code CHAR (FK)

## Construct

Construct_id INTEGER
Protein_id INTEGER (FK)
Plasmid_id INTEGER (FK)
Source_sequence CLOB
Construct_desc VARCHAR2 (2000)

Is represented by

Plasmid\_id may be a link to other tables in the LIMS DB, or an identifier that points to a record in plasmid management software such as Vector NTI

Source\_sequence may be replaced by a link to other tables in the LIMS DB, or by an external identifier (e.g., RefSeq ID). Be careful of sequence versioning if you're not storing the sequence directly.

---

# Sample Tracking

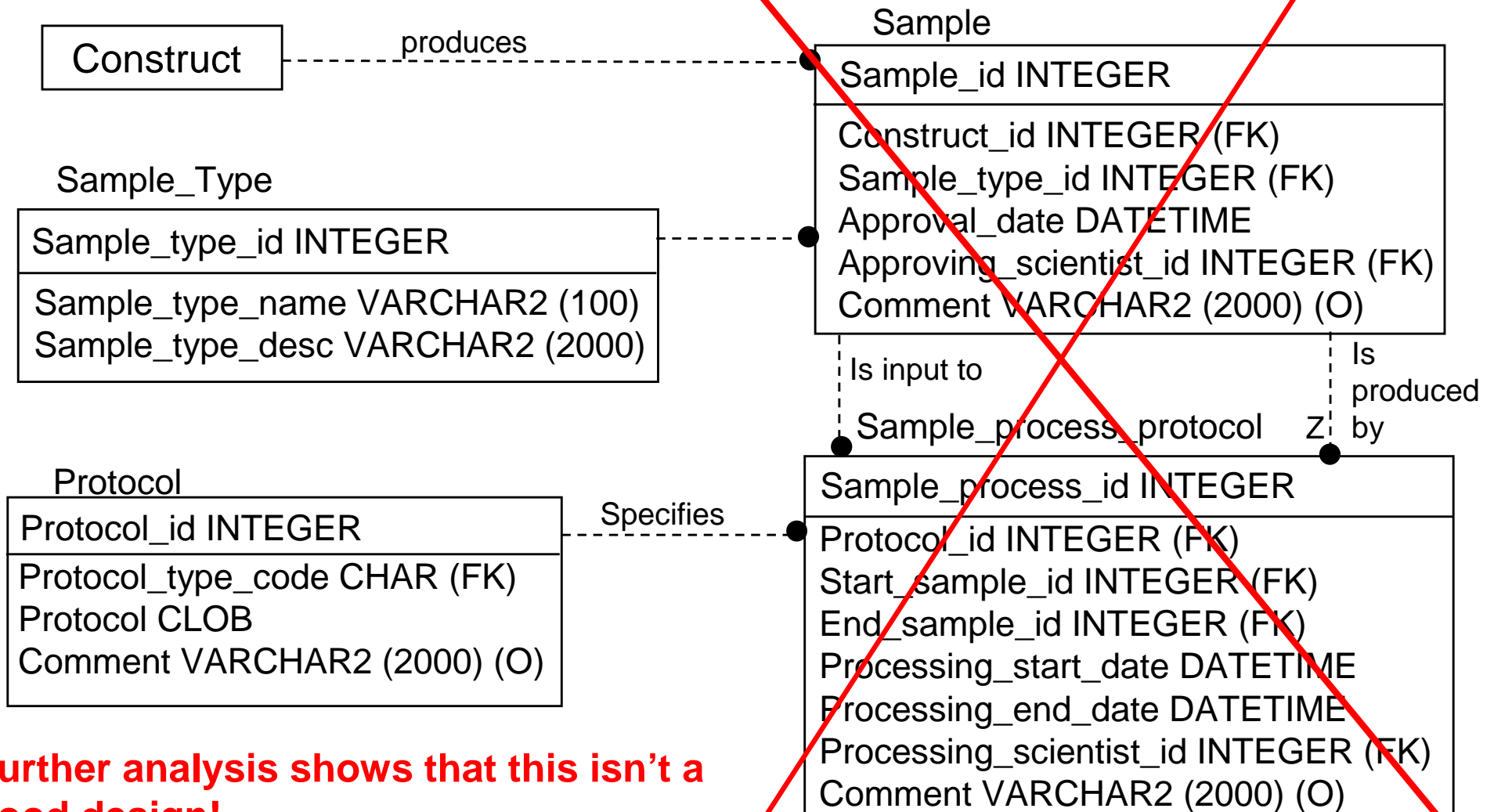
- **Constructs** are used to produce **samples**
- Examples of samples:
  - Plasmid preps (plasmid DNA, stored outside of any cell)
  - Cell lines (cells transformed with the plasmid)
  - Raw cell pellets or supernatants
  - Protein preps in various states of purification
  - Purified protein (as powder or stock solution)
  - I'm probably forgetting some....
- Each construct can produce multiple samples of a given type
  - Each new purification should be tracked separately
  - Allows scientists to trace problem data to a particular batch of protein

---

# Sample Tracking

- Database may track processing steps leading to each sample, or may simply associate each with the construct
  - Decision depends on lab process, how standardized it is (or should be!), and degree of searchability desired on protocols
- Intermediate option: associate a free form protocol with each sample
  - Can store standard protocols in database
  - Cannot easily include protocol steps in searches
    - “Find all proteins for which at least one construct has a purified protein sample that has been purified by ion exchange”

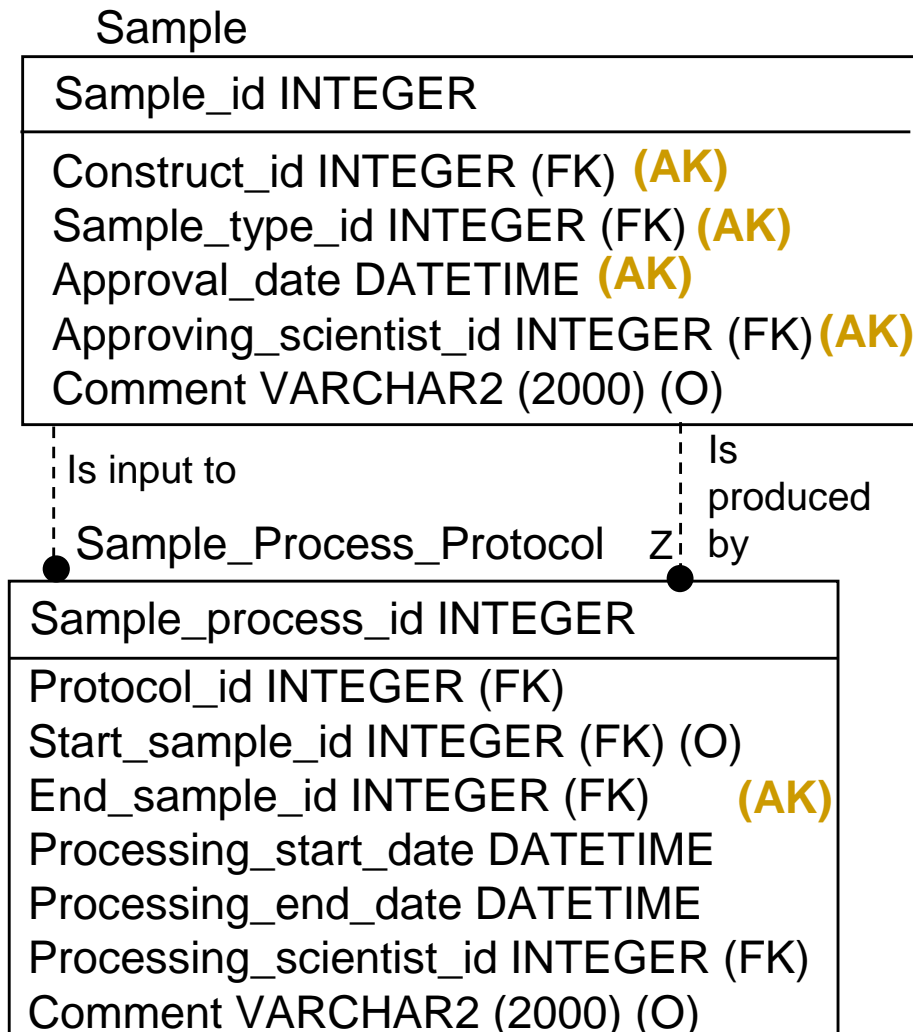
# Sample Tracking



**Further analysis shows that this isn't a good design!**



# Sample Tracking Issues



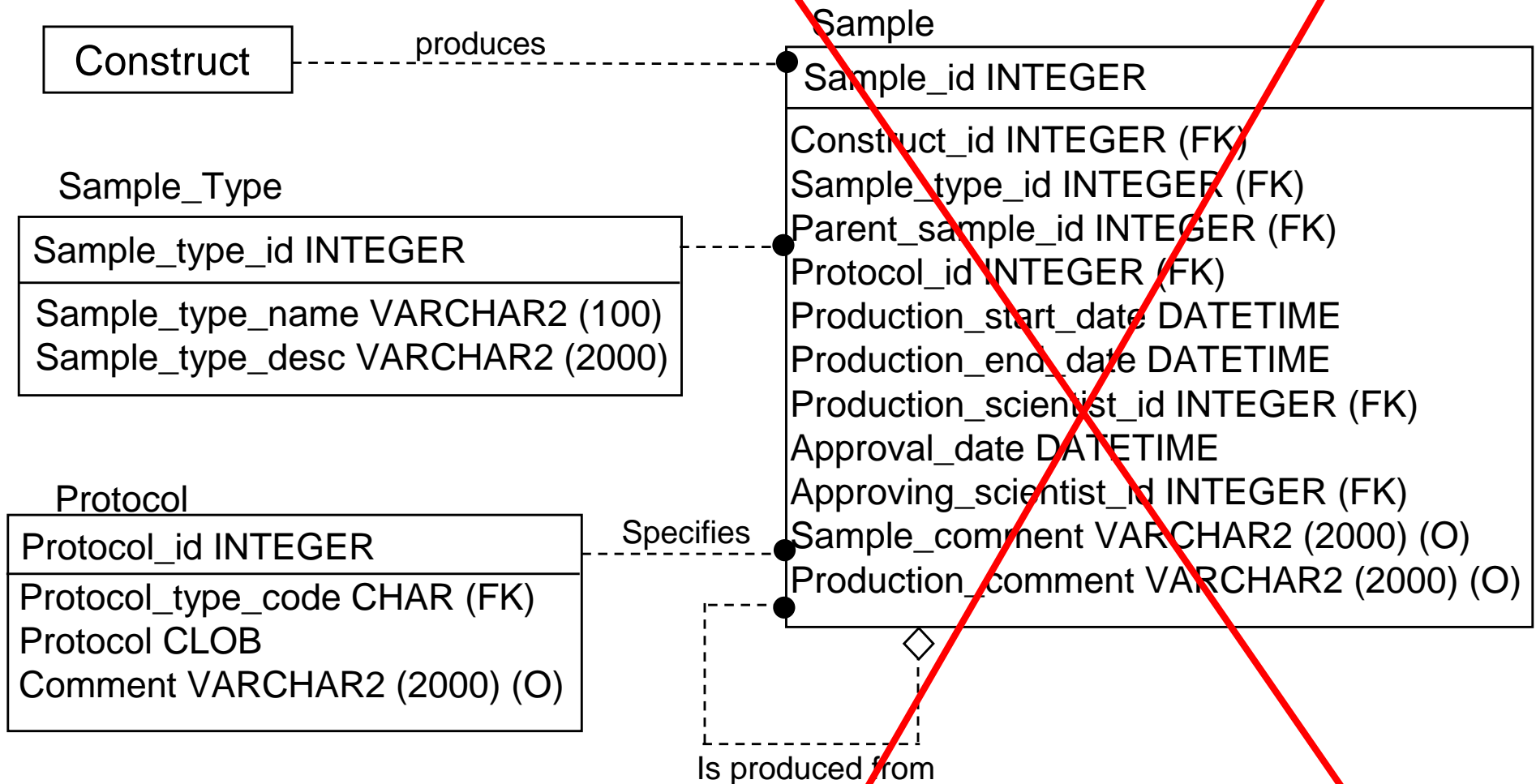
What is the alternate key for Sample?

This demonstrates why its so common to use system-generated primary keys!

What is the alternate key for Sample\_process\_protocol?

Demonstrates the value of thinking about natural keys, even when using a system generated key. Why am I using a system generated key when the natural alternate key has only one column?

# Sample Tracking Issues



**Also a problematic design!**

# Sample Tracking Issues

Sample

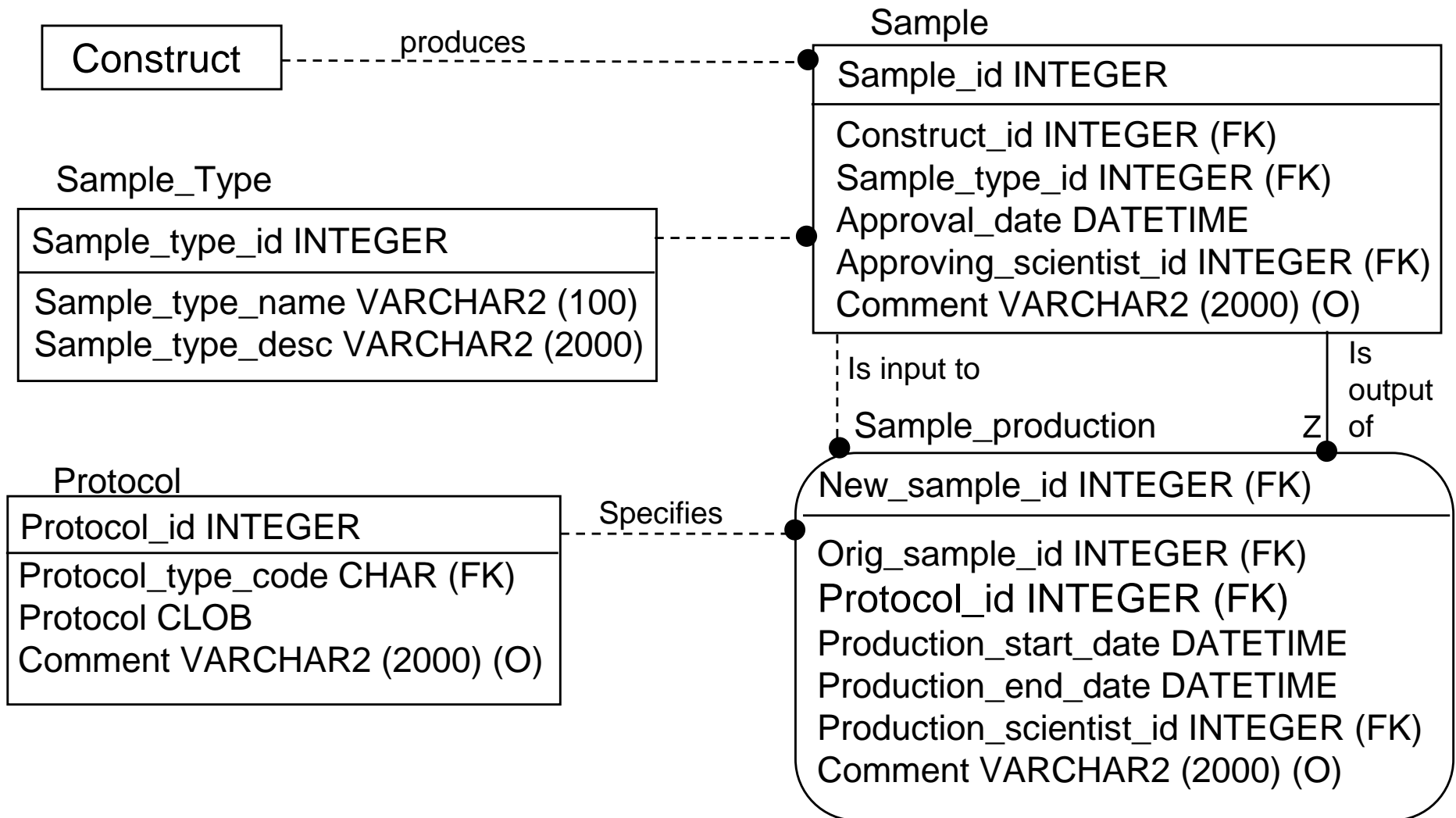
Sample_id INTEGER
Construct_id INTEGER (FK)
Sample_type_id INTEGER (FK)
Parent_sample_id INTEGER (FK) (O)
Protocol_id INTEGER (FK) (O)
Production_start_date DATETIME (O)
Production_end_date DATETIME (O)
Production_scientist_id INTEGER (FK) (O)
Approval_date DATETIME
Approving_scientist_id INTEGER (FK)
Sample_comment VARCHAR2 (2000) (O)
Production_comment VARCHAR2 (2000) (O)

Is produced from

The problem with this design is that it includes several columns that will be NULL for the first sample in a chain.

The reason for the NULLs is that those attributes actually belong to the relationship between the sample and its parent sample.

# Sample Tracking Issues



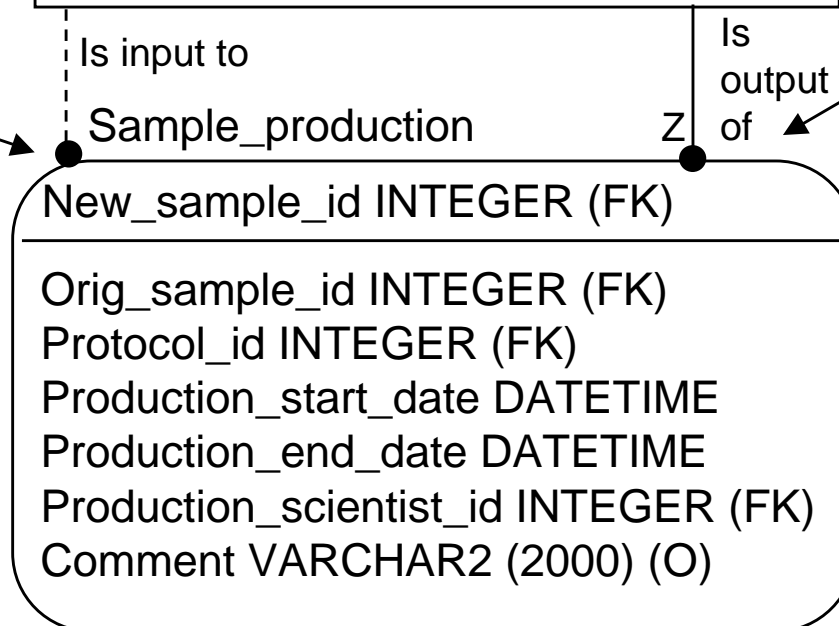
# Sample Tracking Issues

A sample may be processed into more than one “child” sample, but may also be the “end of the chain”: i.e., the final sample.

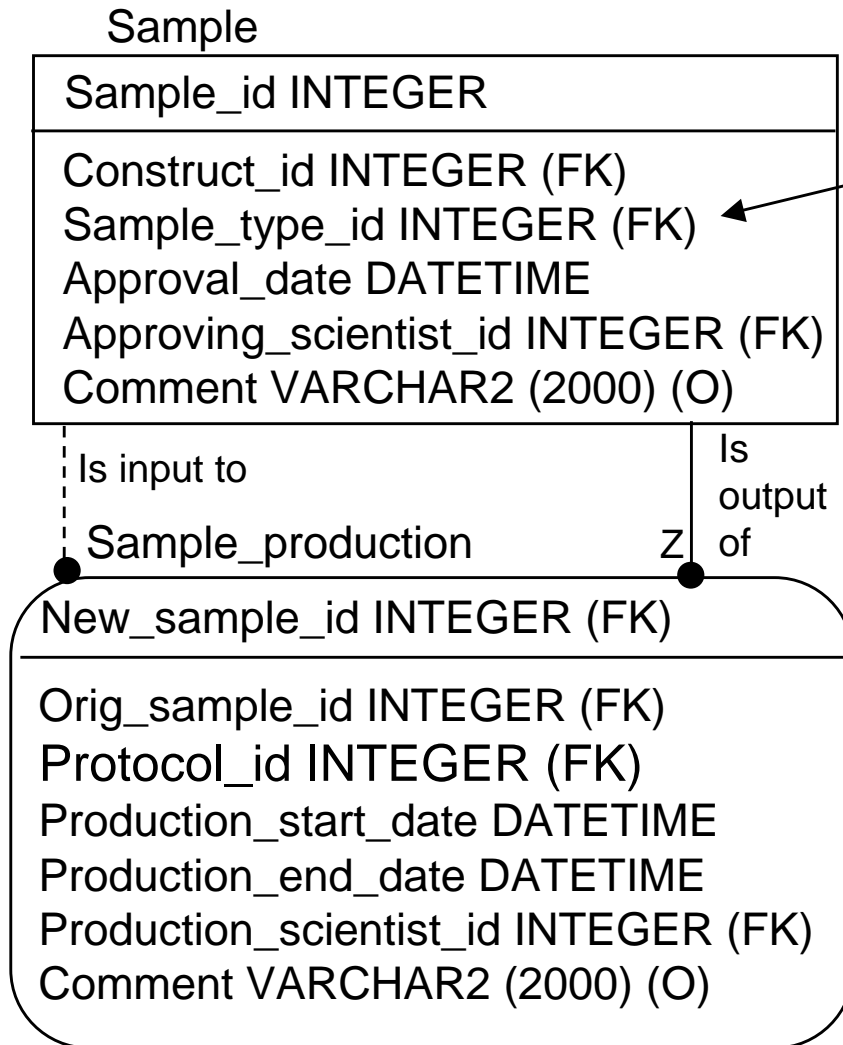
Sample

Sample_id INTEGER
Construct_id INTEGER (FK)
Sample_type_id INTEGER (FK)
Approval_date DATETIME
Approving_scientist_id INTEGER (FK)
Comment VARCHAR2 (2000) (O)

There must be a first sample! The first sample in a chain will not appear as the “new\_sample\_id” in the Sample\_production table.

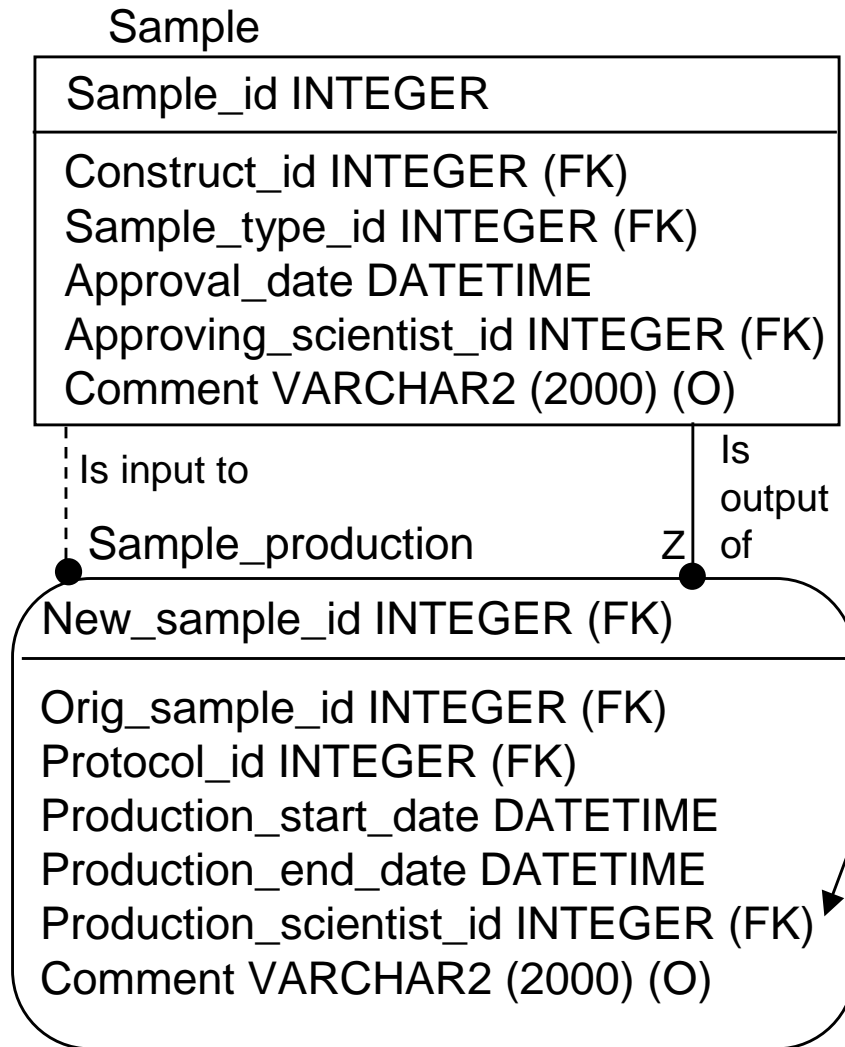


# Sample Tracking Issues



Using “lookup table” to control allowed values for sample type. If your DBMS supports user-defined data types, may want to use those instead.

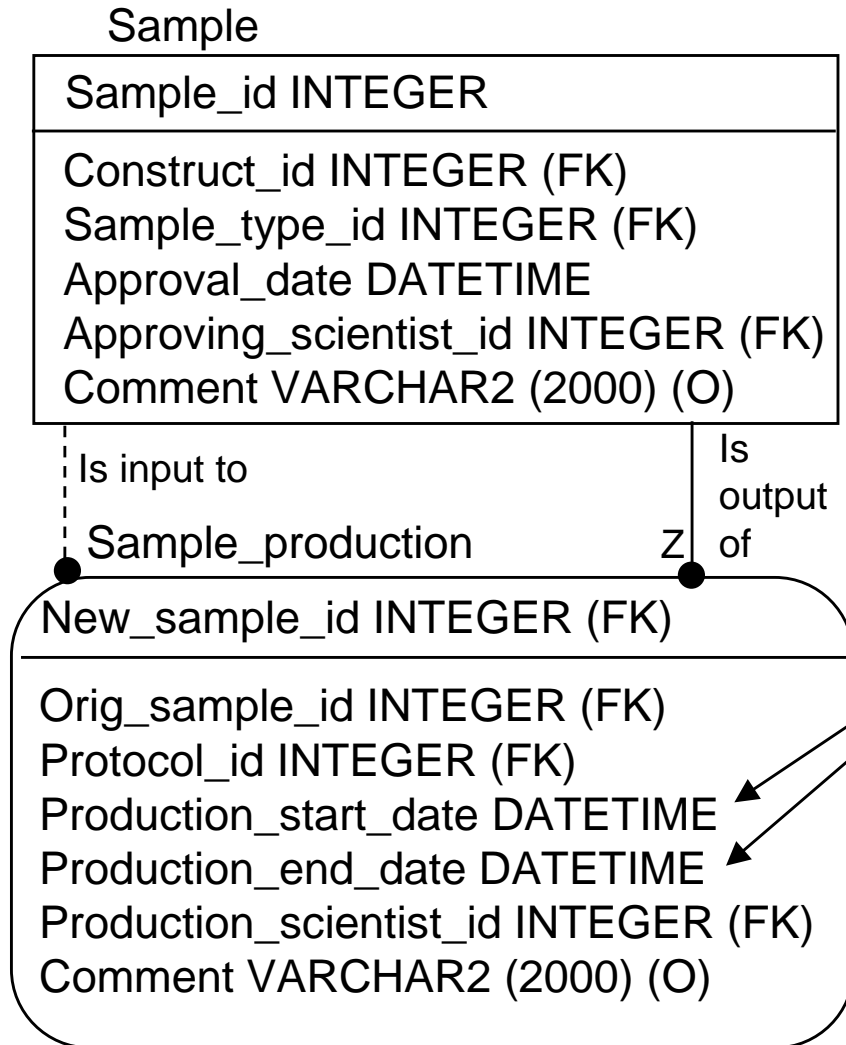
# Sample Tracking Issues



This design assumes that one scientist (a technician) performs sample processing. Another scientist (a supervisor) “approves” the sample: accepts that it has been produced properly, and can now be used for experiments.

In some labs, more than one technician may work on the producing one sample from another. In this case, would need to create a new table “Production\_scientist” and set up a one to many relationship between Sample\_production and Production\_scientist

# Sample Tracking Issues



Often, the process of converting one type of sample to another will require more than one day.

“Dates” are actually “date-times”, and by default often are specified to the second. However, I generally wouldn’t bother to put a start and end date if they are both on the same day.



# Sample Tracking Issues

Protocol	
Protocol_id	INTEGER
Protocol_type_code	CHAR (FK)
Protocol	CLOB
Comment	VARCHAR2 (2000) (O)

Even if you're storing the protocol as a CLOB, it may be a good idea to classify the protocol. For instance, the scientists may want to be able to find all protein purification protocols using His-tags.

Careful definition of protocol types can alleviate one problem with this design: the inability to search on specific protocol steps.

---

# Example LIMS Databases

- SPINE: Structural Proteomics in the NorthEast
  - Medium throughput protein structure determination, by X-ray and NMR
  - Bertone, et al., Nucleic Acids Research, 2001, 29: 2884-2898
  - Not necessarily a great database design: hard to tell from the information in the paper
  - NESG website provides MySQL generation scripts ([spine.nesg.org/download](http://spine.nesg.org/download))
- Gene expression databases often include LIMS portions
  - Stanford Microarray Database (SMD): [genome-www5.stanford.edu](http://genome-www5.stanford.edu)
- LabBase
  - Object-Oriented
  - Steve Rosen, Lincoln Stein, Nathan Goodman
  - [www.broad.mit.edu/genome\\_software/labbase/lbback/lbback.html](http://www.broad.mit.edu/genome_software/labbase/lbback/lbback.html)

---

# Other Types of BioDBs

## ■ Proteomics

- ❑ PEDRo: an attempt to provide a standard data model and suggest data requirements for proteomic experiments
- ❑ Roughly analogous to MIAME
- ❑ [pedro.man.ac.uk](http://pedro.man.ac.uk)
- ❑ Taylor, et al, Nature Biotechnology, 2003, 21: 247-54

## ■ Protein structure

- ❑ Protein Databank (PDB), now managed by the RCSB
- ❑ [www.rcsb.org/pdb](http://www.rcsb.org/pdb)
- ❑ Berman, et al., Nucleic Acids Research, 2000, 28: 235-242

---

# Other Types of BioDBs

- Protein-Protein Interactions
  - Database of Interacting Proteins (DIP)
    - [dip.doe-mbi.ucla.edu](http://dip.doe-mbi.ucla.edu)
  - Biomolecular Interaction Network Database (BIND)
    - [www.blueprint.org/bind/bind.php](http://www.blueprint.org/bind/bind.php)

## Pathways

- Rapidly expanding field
- Peter Karp's EcoCyc and MetaCyc provide a good overview of some of the issues.
- Many more!

---

# Homework

- Homework: design a set of tables to store information in MIAME section 2
- Reading for this week's class
  - Paper discussing GeneLogic's approach to managing gene expression data
  - Implementing LIMS: A "How To" Guide
- Optional reading for this week's class
  - Nature Genetics paper on MIAME (strongly recommended, but will require a trip to the library)
  - A computer scientist's explanation of microarrays (strongly recommended for those not familiar with the technique)
  - MAGE-ML paper
- Reading for next week's class: my BIOSILICO paper (reprints will be handed out in class)

---

# Homework

- Design a set of tables to store information about samples used in a microarray experiment
  - Details of data are in MIAME, section 2 (“Samples used, extract preparation and labeling”).
    - [www.mged.org/Workgroups/MIAME/miame\\_1.1.html](http://www.mged.org/Workgroups/MIAME/miame_1.1.html)
  - Only include subsections (1) (“Bio-source properties”) and (2) (“Biomaterial manipulations”)
    - Don’t worry about connecting your tables with sample information to other tables with information about the rest of the microarray experiment
  - Remember principles of MIAME:
    - MIAME explicitly states that it is not always possible for them to list all relevant attributes. Be sure to accommodate this in your design.