

Flow Informatics and Computational Cytometry Society (FICCS) FCM Object Model Working Group Development Workshop

October 18-19, 2006

U.T. Southwestern Medical Center

6000 Harry Hines Blvd., Room NB7.128, Dallas TX 75390-9072

Meeting participants:

Keith Boyce (Immune Tolerance Network)

Jennifer Cai (Scheuermann group, UTSW)

Mark Dalphin (Amgen)

Megan Kong (Scheuermann group, UTSW)

Jamie Lee (Scheuermann group, UTSW)

Yu (Max) Qian (Scheuermann group, UTSW)

Richard Scheuermann (UTSW)

Josef Spidlen (Brinkman group, BCCC)

Peter Wilkinson (University of Montreal)

Meeting summary:

- Topics
 - Presentations given by
 - Josef Spidlen: Java, experiment and gating descriptions, UML object model, file formats (FCS, XML)
 - Jennifer Cai: The ImmPort experiment data model
 - Richard Scheuermann: the FuGE object model
 - Evaluate whether the FuGE core is compatible with the requirements of the FCM experiment metadata
 - Discuss ideas for how to extend FuGE for FCM derived data
 - Compare MIAFE-DB, MIFACE, ImmPort, and FuGE to determine how the minimum FCM information sets would be managed within the FuGE framework and to identify areas which may need to be extended in FuGE
- Conclusions
 - FuGE can be used as the core for the FCM object model
 - There are 3 main areas that need to be extended:
 - Protocol Application (to support data transformation such as compensation, assay types such as intracellular staining, and material transformation such as treatments and enrichments)
 - Data (to support FCM data)
 - Material (to support types of materials used in FCM experiments, especially reagents that may include reporters and detectors that play important unique roles)
 - Need to come up with use cases for FCM assays, material transformations and data transformations, including:
 - Biological sample isolation/enrichment
 - Standard cell surface staining assay

- Complex FCM assay involving mixed cell populations, treatments, intracellular staining and different categories of staining reagents
 - Compensation
 - Gaussian Fitting
 - Parameterization of Distribution
 - UserDirect
 - Other Data Transformations
 - Gating/Filtering
 - Scaling/Transformation
- Action items
 - Josef and Peter will work together on drafting FuGE FCM extensions
 - Each group will come up with an assay use case and a data analysis use case and determine how they map to FuGE and the FCM extensions
 - UTSW group will complete the MIAFE-DB/MIFACE/ImmPort/FuGE comparison
 - Peter will set up a mailing list for the FICCS Working Group
 - Deadline for action items is Friday, October 27 in order to allow time to prepare for the FuGE working group meeting on November 7

Detailed meeting notes:

Wednesday, October 18, 2006

AM session

- Meeting participants introduction and opening remarks
- Josef's presentation
 - Topics: Java, experimental and gating descriptions, UML object model, File formats (FCS, XML)
 - FCS 4.0 proposal
 - Focus on interoperability, focus on primary data only (list mode, uncompensated; no metadata), no user-defined keywords or segments
 - (Mark) Reason for choosing rdf for modeling metadata? (Josef) most versatile; libraries to help define relationships; compatible with ontology formats. The alternative is to use FuGE?
 - Gating-ML proposal
 - XML-based description of gates
 - Supported gate types: rectangular (n-D), polygon (2-D), polytope (n-D, convex only), ellipsoid (n-D), decision trees, Boolean collections of any of the types of gates
 - (Mark) – comment about gating strategy for each event, assign probability that it is a certain type of cell (rather than a B&W gate); non-rigid “gates” (not really called gate anymore because it's a probability)
 - (Rich) – how do you define the population? (Mark) “Semi-supervised clustering” – begin with a 2-D plot, interactively select a pop and put the population center there; or k-means to find the clusters, then give the clusters a name. Optimization occurs in all dimensions
 - (Rich) we are talking about 2 diff things – a Gate is a data filter. The probabilistic calculation is not – it's a classification
 - (Peter) – I intend to have clustering(?) fully automated
 - (Rich) – there should be a different standard to support this “classification” – right now we are dealing with standards for hard gating, which is what 99% of FCM users use and we must support.
 - (Josef) has also thought about “fuzzy gates” and had proposed this method, it is a straightforward extension of the current Gating-ML. However, it complicates the semantic, makes Gating-ML more difficult to implement, and we haven't identified enough use cases to put this in place.
 - (Rich) you have patient A and patient B – how do you describe their differences using FCM data? (Mark) we are working on that – try using distance metrics in n-D space, z-score of distance, predict the “state” of cells. Use weighted histograms where each item in the histogram is multiplied by the probability, generating another “weighted histogram” which can be presented to FCM users since it is a familiar format.

- (Rich) Max has an algorithm form automatic population clustering; can be used to define centroids
 - Compensation-ML
 - XML-based description of parameter compensation
 - Based on spillover matrix specifications
 - Applicable on data on linear scales only
 - Non (directly) convertible to log scale after compensation (possibility of negative values)
 - Transformation-ML
 - XML-based description of parameter transformation
 - Gated using different scales
 - data visualization issues
 - Originally part of gating proposal, separation makes the usage more flexible
 - Predefined transformations
 - Lin, quadratic, log base e, 10 or other
 - Hyperlog, bi-exponential, logical, split-scale
 - Support for universal transformation description (Math-ML)
 - (Josef) Compensation could be expressed using transformation-ML (more complicated description, missing the “compensation” semantic); (Rich) compensation is a type of transformation. Separate the user interfaces from the data structure. (Josef) for comp, more efficient to store a matrix. (Rich) FuGE uses a matrix approach. Transformation is a data analysis markup language. Gating is one kind, transformation is another kind. In the short term this solution is fine
 - FACEJava
 - LGPL Java reference implementation for Gating-ML, transf-ML, comp-ML, and FlowRDF
 - Usable as
 - Set of libraries
 - Command line tool
 - Applies *-ML on input FCS files (FCS 2, 3) → outputs new FCS3 files with processed data
 - (Mark) – under impression that data only have compensated values (as output)? Or both raw + the compensation matrix
 - (Josef) – take raw data, apply gates using Gating-ML, then output as extra FCS3 file
 - FACEJava packages
 - Cfcs library used to work with fcs files (cfcs - FCS serialization library)
 - Gating, comp, transf implements the corresponding proposal
 - Utils: shared code for XML processing (JAXB)
 - Faceflow: connecting the other packages, implements relations between data and metadata
- Jennifer’s presentation
 - (Rich) – encourage everyone to think in a more generic context

- (Peter) – we have to do the same thing. LIMS to record experiments, then a second database to store data in a generic fashion. (Rich) but at least you have a standard source of data (FCM) – we do not.
 - (Mark) – can biosamples refer to other biosamples as source? Yes
 - (Rich) – important points – formal separation of analyte from detector/reporter. FCS3 file – things that we want to capture may or may not be there so we need to be able to capture info whether or not they come from fcs file. 2 sources of the same info – use “secondary”. If there is an inconsistency, then the submission could be flagged for the user.
 - (Peter) – must have controlled vocabulary otherwise data needs a moderator. This is the end-goal.
 - (Mark) how do users get protocol into the system? (Rich) – submit free-form document. What we try to do is database only what is important. Other details in the protocol don’t need to be parsed. E.g. conditional variables, treatments, etc.
 - (Josef) – distinguish protocol and protocol application? (Rich) not yet but in evaluating FuGE, it is useful. But we haven’t made that distinction.
 - (Peter) – likes the use of the word “filter”. For us, it is just a more generic term
 - (Keith) – the important thing is that this is not an analyzable function right now. (Rich) There is not sufficient info to reproduce the data filters. (Mark) – ideally we need to be able to do this. Focus on what’s do-able, and what’s do-able NOW (e.g. FCS3 vs FCS4). (Rich) secondary fields are FCS3 standards, not FCS4 because FCS4 won’t be in machines until at least a few years later
 - (Rich) we are heavily focused on metadata, so that is our bias
 - (Mark) how useful has this been? Can people find the information they are looking for? (Rich) we first want the user to get comfortable and not have to fill in everything yet. (Keith) – it’s reasonable. People can do it.
 - (Peter) appoint someone to be centrally responsible for getting data into the database.
 - (Mark) my experiences have been disappointing over long term. If a field can be left empty, it will be left empty. Spelling mistakes, etc. (Rich) as we have the ontologies linked, then we will start making things required. Initially we will have a mess but we have to get buy-in from the user first. Query for research data works as long as there’s consistency in the way things are described (e.g. like IL2 won’t pull up IL-2)
 - (Peter) think about record entries with different status (e.g. new record entries that are unvalidated, then entries that are validated) Then you can start defining queries (e.g. query validated data only?)
 - (Rich) ontology will solve a lot of this. You will never solve misspellings, but it will solve synonyms. (Mark) the other problem is the greek characters – interferon gamma)
 - (Rich) need some automated validation. Maybe use web UI for entries and validate live.
 - (Peter) Joomla (Joomla.org) (originally Mambo) – content mgmt system; this is most popular; they are releasing 1.5. Turning into more of an application platform. Content mgmt system - Model is data-driven; uses portals to auto-generate pages. Joomla is simple, it’s built for performance. PHP, MySQL, Oracle implementations. Others content mgmt systems – IBM has one, WebSphere
- Richard’s presentation (FuGE)

- FuGE: a framework for developing standards for functional genomics (Angel Pizarro and Andrew Jones presentation modified by Richard)
- Ontology FuGO → OBI standardized nomenclature and relationships for objects described in FuGE object model
- Data standards for HT Genomics
 - Major challenge developing standards
 - Evolving technology
 - Heterogeneous data formats and types
 - Important info about starting sample is almost unlimited
 - Large quantities of metadata to validate results
 - But
 - The advantage is most problems are shared by microarrays, proteomics, metabolomics, etc
 - Our data model is an example of common features
- FuGE object model
 - Merges of MAGE and PEDRo models were attempted
 - Results where an even more complex model that still left other FG technologies untouched
 - Main motivation was reuse MAGE sample prep and ontology components
 - Idea abandoned
 - FuGE project created as an independent project from MGED and PSI
 - Model of common components across all FG experiments to enable synergy btwn standards
 - Sample desc, protocols, investigation structure
 - Handle unique aspects by
 - Referencing external data formats
 - Adding extensions to FuGE core (consider this to drill down for FCM data)
- Experiment workflow
 - Inputs and outputs of protocols
 - Instances of protocols
 - 3 kinds of protocols
 - biomaterial in and biomaterial out (e.g. purification, isolation of blood) – material transformation
 - biomaterial in and data out (main type)
 - data in and data out (data transformation); can be described with a protocol and reagents; similar to a wet lab investigation
- Architectural details
 - Represented in UML object model
 - Uses AndroMDA model-driven architecture approach to produce platform-specific models
 - Eventually we want to head in the same direction so our model merges with FuGE but timeline is uncertain
 - (Peter) FuGE has not addressed implementation into a db yet. There needs to be a layer between the model and the db

- FuGE structure
 - Common elements
 - General data format mgmt
 - Auditing
 - Reference external resources (e.g. ontology so that you don't have to build this into your object model)
 - Protocols
 - Bio
 - Investigation structure
 - Data
 - Materials (organisms, solutions, compounds)
 - Theoretical molecules e.g. sequences – a gene as a concept rather than an instance
- Status
 - Milestone 1 release – sept 2005
 - 2 – Dec 2005
 - Acceptance by PSI and MGED
 - 3 – Spring 2006
 - Version 1.0 – Fall 2006
- FuGE extensions
 - MAGE V2
 - Format for microarray data and annotations
 - GeIML
 - 2D gels
 - spML
 - liquid chromatography, electrophoresis, centrifugation
 - CPAS
 - Metabolomics community considering
 - FCM community interested
- FuGE base classes
 - 2 base classes – describable and identifiable
 - Describable
 - Every object is a subclass of describable
 - Allows merging of components in complex ways
 - Max flexibility
 - Can attach unique IDs to the object, give it a URI
 - You can describe it using text or ontology terms, others
 - Security and audit info is linked to describable
 - (Peter, Jennifer) – there are “Abstract classes” that you cannot form a database with. UML model (UML class diagrams) can't be directly translated to data model, which is represented by ERM diagram. Class entities describe the conceptual domain. DB will be designed based on this. E.g. person = conceptual, employee ID = DB table (Max – this is a model of a model – a meta meta model) (Mark) this model could be richer. (Rich) – are these concepts sufficient to describe a

FCM experiment and its data? (Mark – should use ORM (object relational mapping) which is a richer language than UML) (Max) if we use a object DB rather than relational DB, then we could translate FuGE directly into the DB (Mark) going from FuGE to DB is harder than going from ORM to DB. (Rich) – minimum information about an experiment – what do you WANT to capture. One possibility is to take minimum info and see where it fits in this framework to see if it is sufficient

- Identifiable
 - Subclass of describable
 - A describable which has an added referencing mechanism
 - Has been described elsewhere but you can link to its identifier from its external source and bring in all its info
- Ontology use
 - Ontology referenceing possible throughout the model
 - Ontology individual – an instance of ontology term??
 - Relationships between classes through ontologyProperty and objectProperty
 - DataProperty allows adding a value to a term
 - Example (4 hours)
 - ontologyIndividual – measurement
 - objectPropoerty – has_unit; Ontology Individual – hours
 - data Property – has_value; value = 4
- Abstract structure of protocol
 - Set of ordered actions
 - Protocol constructed by linking actions together
 - You can nest the protocols
 - Abstract classes, which can be extended for modular formats – particular types of protocols can be extended. E.g. we describe a generic staining protocol; it can be extended to domain-specific staining protocol
 - Non-abstract classes that can be used without extension e.g. generic protocol is at an instance level
- Generic protocol
 - Parameters, actions
 - Instantiable
 - Described by text or ordered set of actions
 - Action can be represented by text or ontology
 - Allows description of complex workflows
 - genericProtocol can have a set of defined parameters (instances)
- Protocols with default parameters
- Parameters
 - Can have default values and instance values
 - Values can be Boolean, atomic, range, complex
 - ParameterValue (actual instance) linked to Protocol Application (the instance level of protocol)
- Protocol application has equipment, software, protocol deviations, performers (who performed the protocol application)

- Extensions
 - Protocol can be extended with ImageAcquisition
 - Equipment with scanner
 - Action with scan (“scan action”)
 - Material
 - Protocol application can have inputMaterial and outputMaterial
 - Input goes thru protocol application and then out
 - Can be described by text or ontology
 - (Rich) FuGE is generic enough for all use which is the goal; implementation is another thing – it has to be more focused for performance issues (Mark) we need to be able to capture AND share the information (Rich) once data is fit into generic concept, then same concepts of different experiments will map together. (Mark) should we describe flow as a minimum set AND an extended set – you must have THIS and IF you want to add THAT, it has to be in THIS FORM. So we have to agree to the extension as well as the minimum amount. (Rich) e.g. probabilistic definitions – we should define a framework for extending the core model to be able to deal with this.
 - Material or data or input
 - Any combo. E.g. Input material → material AND data as output
 - Data Package
 - Data represented as a matrix of values (could be n-D)
 - Dimensions are reusable for multiple instances of the data
 - dataPartition for data subsets (data filtering could be a data Partition)
 - PartitionPair for linking input subset with output subset
 - (Peter) – a partition is a protocol application; why is it a separate table? A partition is a subset containing 100% of the data
 - (Rich) – we can give feedback to the FuGE group if we decide we want to use it
 - Investigation Package
 - High-level description of motivation
 - Experiment factors (variables, analytes, materials)
 - Which data applies to which factors
 - investigationComponent – a single technique used in an investigation
 - (Rich) Factor – describes the conditional variables, the constants, and the responding variables in the experiment. These are all different kinds of factors
 - Conceptual molecule package
 - (Rich) Concept IL-2 and then there’s the reagent IL-2 that we use to stimulate samples
 - (Peter) for flow – what happens when you change the clone, or change the fluorochrome. Anti-CD8 is really a particular version of anti-CD8

(Rich) come up with a game plan for the rest of the workshop

- What are our goals? What do we want to do? What do we want to accomplish? How? What will be the product of the workshop? Short term vs long term

- (Josef) Ryan wants me to take FuGE and start experimenting with it and how it can be extended for FCM – need to get some ideas on how to do that. Unless there are good reasons to not extend from FuGE. If this is the case, we should identify the reasons for not using FuGE
- (Rich) I would like us to extend FuGE unless there's a good reason not to. People like the idea of a generic structure and it is a benefit
- (Peter) model FCM, describe common elements; is there something missing for FuGE
- (Rich) take minimum info set and see how it fits into conceptual description. Do the relationships make sense?
- (Josef) FuGE needs to be extended for analysis but not for shared things like experiment metadata
- (Rich) we really need to evaluate the data processing associated with FCM, how does it fit into this framework? FuGE seems to deal with experiment metadata fine. It's analysis that will need the extension. What about describing material as a reporter and detector? (peter) we have to model it and see if it fits into the framework. How do we extend it?
- (Rich) this PM – start with the idea that we will try to extend FuGE to bring in domain-specific knowledge. If we can, then we should propose to adopt the FuGE framework. Do we do that by building out FuGE or modeling FCM and then pasting it back on top
- (Max) try to connect MIAFE-db and FuGE first

PM working discussion

- (Mark) – come up with object model that captures what was done in the experiments but real interest is working with analysis of the stored data
- (Peter) – interested in generic data storage and putting into db whatever needed to select raw data for analysis
- List of things to extend:
 - Experiment
 - Analysis
 - Primary raw data →(protocol (preprocessing), including QC/QA) → derived data
 - Raw data
 - QA/QC
 - Auditing
 - Security
- Metadata – 3 types
 - Metadata about raw data
 - Metadata about derived data
 - Metadata about the experiment
- Raw data
 - Inputs
 - Fluorescence intensity values
 - What is the type of the data? Is it uncompensated? Is it linear? Is it transformed? Primary data is what you get from the machine. Raw, unprocessed data is the linear uncompensated values.

- Raw data (primary data) is uncompensated linear data; secondary data is data that has been transformed = derived data. Transformation is a data analysis protocol application.
 - Primary data has n parameters (columns)
 - Primary data has p events (rows) that are not ordered; with each row associated with a time (event + time)
 - Derived data has m rows
 - Remember to distinguish from implementation
 - All we need in FuGE is the Data and ProtocolApplication. Our “derived data” is really data (primary) that has undergone a protocol application to give data (derived)
 - Bottom line – FuGE is fine for primary data
 - Use empty objects for non-existent raw data
- Protocols (enumerated types)
 - Scaling (log, etc)
 - Compensation
 - Gating/filtering
 - Transformation
 - Josef’s proposed extension for compensation protocol
 - (Rich) is the compensation matrix a complex parameter value?
- (Rich) FuGE – linkage between analyte, detector, and reporter? It’s not there
- You extend Protocol, not Generic protocols. So a compensation protocol is a child of Protocol, and compensation protocol is a sibling of generic protocol.
- (Peter) ORM will allow you to derive the UML data model. In his opinion, ORM is the way to go.
- Protocol extension
 - Compensation
 - Scaling/transforming
 - Filtering/Gating/Subset
 - Combination of the above
 - Parameterization of distribution (e.g. MFI, range, mean, median, mode)
 - Staining
- QA/QC what is it? A protocol application, an investigation? Rich thinks it’s an investigation.
 - QA is an investigation – it addresses a question – it relies of protocols having been run
 - QC is a protocol (done in advance)
- Mark’s use case – receptor occupancy has several different fluorescence intensities (how much drug is on receptor when it becomes effective?)
 - (Rich) Determining receptor occupancy is an investigation (comparing drug under condition zero versus drug under condition x); calculating MFI is a data transformation
 - (Mark) Maybe an investigation is one where you apply experiment metadata (experiment metadata-drive) in order to manipulate the data. In contrast to data transformation where you only need the data (data-driven only), and metadata about the experiment is not applied

- (Mark) What about data-driven investigations (such as data mining)? Did FuGE miss it? (Rich) we need to feed back to FuGE. There should be a data investigation versus a bio investigation
- (Rich) how do you re-use extensions that you make to FuGE? How do you compare microarray data to FCM data?
 - (Mark) and in the end, you want to be able to do systems biology. Convince FuGE that we need a mathematical transformation (from we extend our compensation protocol); then it becomes re-usable
 - (Josef) how to do this? How to you add this to FuGE? How do you re-use common extensions?
 - (Rich) so everyone who wants to do a dose response investigation has to make their own extension in the current FuGE
 - (Peter) need a common mechanism for microarray experiments, flow experiments, etc so that the extensions (i.e. data transformation) can be re-usable
 - Suggest these things to FuGE?
- We will move forward to create a FCM extension at least for the short term
- (Rich) summary of what we just discussed:
 - Today we have banged away at FuGE, for the most part it is compatible with FCM data
 - (Mark) we should write up specific use cases to feed to the FuGE group; hard to talk about thinks in abstract terms
 - Peter would like to create the extension at its simplest form (e.g. compensation)
 - (Rich) we should at least describe the staining protocol application (parameters/dimensions, fluorochromes) just to get something done with experiment metadata. Tricky part is to connect the parameters with the antibody-fluorochromes.
- Need to connect:
 - Dimensions in the fcs file
 - Time
 - FSC
 - SSC
 - FL1, 2, etc
 - Conceptual molecule (analytes)
 - CD8
 - CD3
 - p-65
 - size, granularity (cell concept)
 - What connects them is the Reagent
 - Detector (e.g. anti-CD3) – interacts with an instance of the conceptual molecule
 - Reporter (e.g. FITC) – generates the value for that all dimensions (if uncompensated)
 - For FSC, SSC, you conceptual “molecule” is cell; there is no reagent
 - 2 general classes of material
 - Biosample

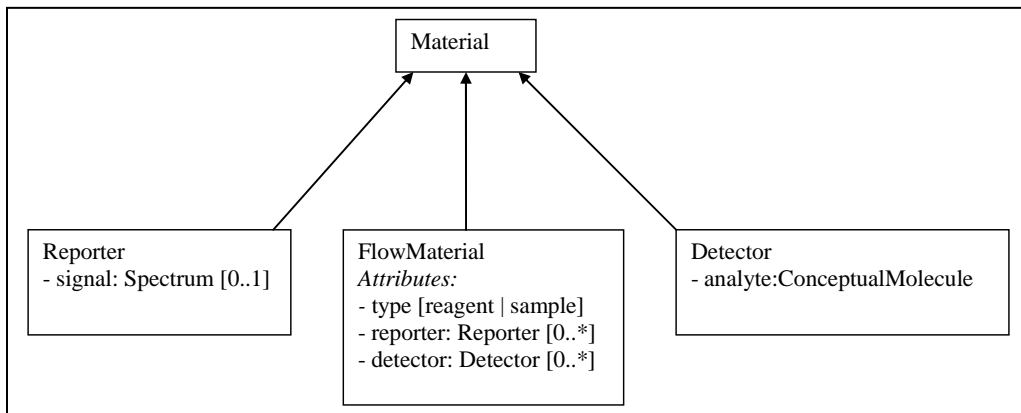
- Reagent
 - How to deal with reagent? Critical because it connects conceptual molecule with dimension.
 - We can associate the event with the material (reagent), but not the dimension to the material (reagent) and linking primary event data with reagent involves a protocol application (compensation)

Thursday, October 19, 2006

AM session

- Protocol
 - Assay
 - Treatments
 - Samples
 - Computational
- Material
 - Subclasses of material?
- Schedule for today
 - (Rich) 2 things
 - Come up with some deliverable for data analysis part that we worked on yesterday
 - Then come back to the assay part to accomplish as much as possible
- Data analysis
 - What do we need to define?
 - Extensions that we want to make to FuGE
 - Specific packages (groups of extensions) for FCM
 - Assays (biochemical protocol application)
 - Protocols
 - Computational protocols (computational protocol application)
 - Treatments (this gets into the metadata, we probably won't solve it today; this might be handled by protocol)
 - Material
 - Samples
 - Reagents
 - Reporter
 - Detector
 - Event ?? (vector of time plus several measurements). A set of events gives rise to the results
 - The problem with making the connection between Reporter and Dimension
 - 3 use cases (Mark)
 - Raw data compensated giving different data and column headings (conceptual molecule)
 - Previously compensated data and transformation is just changing column names

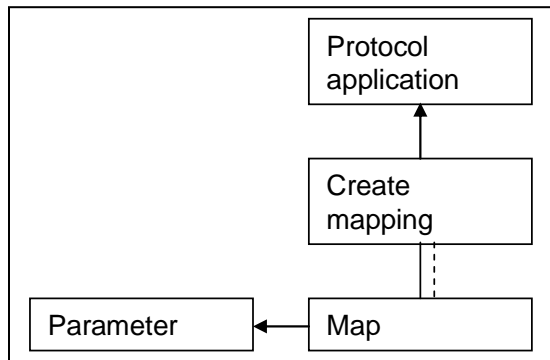
- Assign these things differently (fit distributions and get probability tables); values of columns and headings change
- Summary
 - Extending classes (subclassing from FuGE)
 - InternalFlowData inherits from InternalData (add columns)
 - ExternalFlowData inherits from ExternalData (add columns)
 - 2 classes: internal and external
 - Extend *ProtocolApplication* to *FlowProtocolApplication* (with attribute “Type”)
 - FlowDataTransformation
 - Type (e.g. compensation, log transformation, etc)
 - FlowMaterialTransformation
 - Type (e.g. treatment, enrichment)
 - FlowMaterial_DataTransformation/FlowAssay (Assay)
 - Type (e.g. extracellular staining, intracellular staining, combined staining, QC (but not QA!), calcium flux assay)
 - Use Cases
 - Compensation
 - Gaussian Fitting (Mark – receptor occupancy)
 - Parameterization of Distribution
 - UserDirect
 - Other Transformation
 - Gating/Filtering
 - Scaling/Transformation
 - Data transformation
 - 1 dataset → 1 dataset
 - 1...n dataset → 1 dataset
 - 1 dataset → 1...n dataset
 - 1...n dataset → 1...n dataset



(above) – two proposed ways to extend Material

- (Rich) We must model a connection between the result and what it’s measuring

- Cannot map CD3 to FL1 without compensation matrix (e.g. FL should only be changed to FITC after compensation)
- Gating-ML is a Parameter



- “Deadline” – November 7
- Meeting Summary
 - Evaluation of FuGE and decided that it can be used as the core for FCM object model
 - Areas that need to be extended; those include
 - Protocol Application – Josef, Peter
 - Data – Josef, Peter
 - Material –Josef, Peter
 - Decided to develop FCM use cases – each group should develop an experiment (assay) use case and an analysis use case
 - Assay use cases will help to determine whether metadata is sufficiently modeled in FuGE
 - Indicate how use cases map to FuGE and extended FuGE
 - Timeline: Draft extensions and use cases by next Friday
 - UTSW/Keith– come up with assay use case and computational use case
 - UTSW – finish the FuGE/MIAFE comparison table
 - Peter – create FICCS working group mailing list (@xbioinformatics.org)