Data Model Modifications

EMBL-EBI	Kim Henrick, Wim Vranken, Anne Pajon
University of Cambridge	Rasmus Fogh, Wayne Boucher, Tim Stevens
Daresbury Laboratory	Chris Morris
Diamond	Alun Ashton
University of York	Susy Griffiths
OPPF	Jon Diprose
CNRS Gif/Yvette	Anne Poupon, Eric Oeuillet
Dundee	Geoff Barton geoff@compbio.dundee.ac.uk
Manchester	Steve Prince steve.prince@umist.ac.uk
NRC Montreal	Mirek Cygler, Stéphane Raymond, Xueli Li
Weizmann Institute	Jaime Prilusky
EMBL Grenoble	*HTP Crystallisation:* Jose A. Marquez, Martin Roewer, Jacques Chung
	HTP Expression: Darren Hart, Franck Tarendeau, Benjamin Corbex

Table of Contents

UNITS

All values are floats and they are going to be stored in a fixed SI unit. Because the user is going to enter these values in different unit, a displayUnit will be added.

For concentration where three different types of value can be stored, three attributes are going to be created (concPerMol, concPerMass, concPerVolume) with fixed SI unit.
For amount where three different types of value can be stored, three attributes are going to be created (mass, volume, number) with fixed SI unit.

PACKAGE PROTOCOL

new class Protocol
- Add open enumeration: protocolType = {Pcr, DnaPurification, Digest,...}

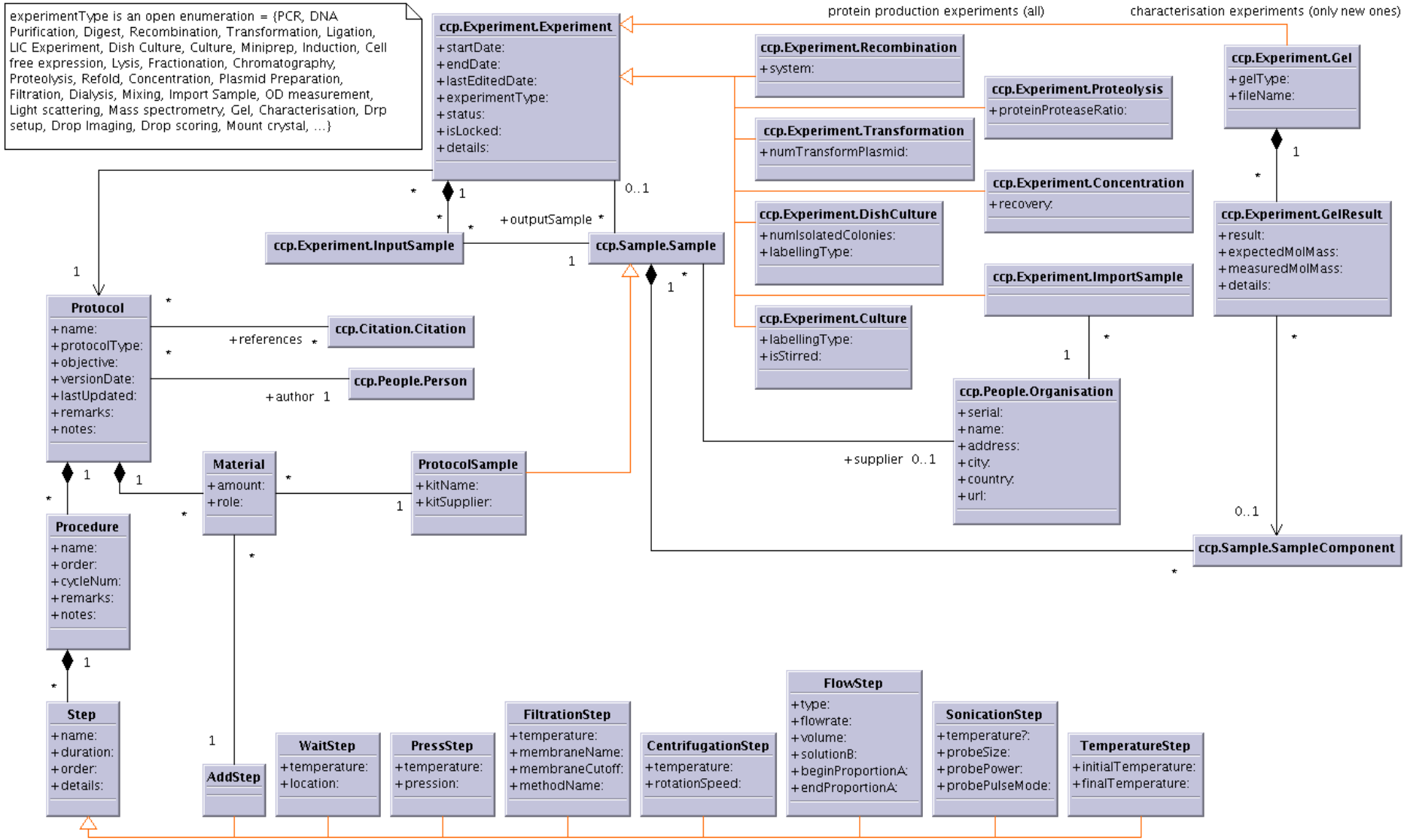
I'm not sure about the name, since a "protocol" to a scientist seems to mean a recipe for a whole series of planned experiments. Can it be ExperimentMethod?

new class FlowStep (subtype of Step)
- Add open enumeration: flowStepType = {fixed volume, one solution flow, 2 solutions gradient, ...}

How are solutionA and solutionB defined?

new class PressStep
- Add attr pressure not pression !!!

new class CentrifugationStep
- Add attr centrifugeModel ???
- Add attr distFromAxis ???
- Add attr numOfG ???


experimentType is an open enumeration = {PCR, DNA Purification, Digest, Recombination, Transformation, Ligation, LIC Experiment, Dish Culture, Culture, Miniprep, Induction, Cell free expression, Lysis, Fractionation, Chromatography, Proteolysis, Refold, Concentration, Plasmid Preparation, Filtration, Dialysis, Mixing, Import Sample, OD measurement, Light scattering, Mass spectrometry, Gel, Characterisation, Drp setup, Drop Imaging, Drop scoring, Mount crystal, ...}
protein production experiments (all)
characterisation experiments (only new ones)
ccp.Experiment.Experiment
+startDate:
+endDate:
+lastEditedDate:
+experimentType:
+status:
+isLocked:
+details:
ccp.Experiment.Recombination
+system:
ccp.Experiment.Proteolysis
+proteinProteaseRatio:
ccp.Experiment.Transformation
+numTransformPlasmid:
ccp.Experiment.Concentration
+recovery:
ccp.Experiment.DishCulture
+numIsolatedColonies:
+labellingType:
ccp.Experiment.Culture
+labellingType:
+isStirred:
ccp.Experiment.ImportSample
ccp.Experiment.Gel
+gelType:
+fileName:
ccp.Experiment.GelResult
+result:
+expectedMolMass:
+measuredMolMass:
+details:
ccp.Experiment.InputSample
+outputSample
ccp.Sample.Sample
ccp.Sample.SampleComponent
Protocol
+name:
+protocolType:
+objective:
+versionDate:
+lastUpdated:
+remarks:
+notes:
+references
ccp.Citation.Citation
ccp.People.Person
+author
ccp.People.Organisation
+serial:
+name:
+address:
+city:
+country:
+url:
+supplier
Material
+amount:
+role:
ProtocolSample
+kitName:
+kitSupplier:
Procedure
+name:
+order:
+cycleNum:
+remarks:
+notes:
Step
+name:
+duration:
+order:
+details:
AddStep
WaitStep
+temperature:
+location:
PressStep
+temperature:
+pression:
FiltrationStep
+temperature:
+membraneName:
+membraneCutoff:
+methodName:
CentrifugationStep
+temperature:
+rotationSpeed:
FlowStep
+type:
+flowrate:
+volume:
+solutionB:
+beginProportionA:
+endProportionA:
SonicationStep
+temperature?:
+probeSize:
+probePower:
+probePulseMode:
TemperatureStep
+initialTemperature:
+finalTemperature:

PACKAGE EXPERIMENT

class ccp.api.Experiment.DishCulture & ccp.api.Experiment.Culture
- Add attr labellingType = {SelenoMet, SelenoCys, C13, N15, C13-N15,...}

new class ccp.api.Experiment.Gel
- Add attr gelType = {agarose 1%, SDS-PAGE 14%, ...}

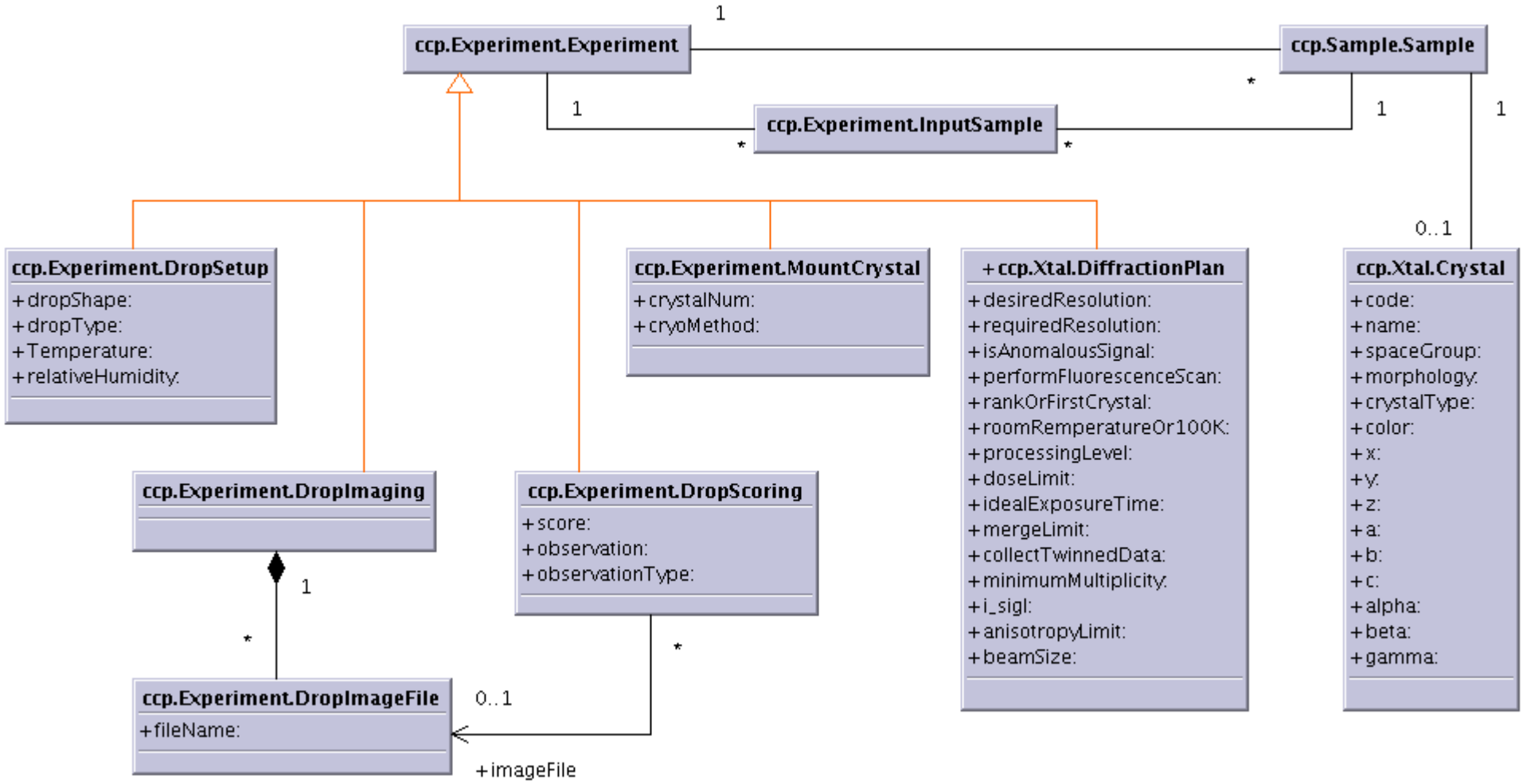
new class ccp.api.Experiment.GelResult
- Add attr result = {band at expected weight, band at unexpected weight, no band}

class ccp.api.Experiment.Experiment
- Add a link to the class Annotation
- Remove attr isModel (old isTemplate)
- Remove attr protocolDetails
- Remove link between Experiment and Instrument
- Add link between Experiment and Method
- Modify attr experimentType = {PCR, DNA purification, ...}. The open enumeration will carry the experiment type, and all experiment subtypes that can be fully described by the protocol class will be removed like PCR, DNA Purification, ...

class ccp.api.Experiment.InputSample
- Remove attr sampleIoType
- Replace attr amount with three attrs:
  - mass
  - volume
  - number
- Add attr displayUnit

A generic way of handling any type of parameters for experiment will be to add two new classes:
new class ccp.api.Experiment.Parameter attrs [name | type | unit | description]
new class ccp.api.Experiment.ExperimentHasParameter attrs [value]
This will give the possibility to store any kind of experiments.

new class ccp.api.Experiment.LightScatResult attrs [intensity | expectedMolMass | measuredMolMass]
class ccp.api.Experiment.LightScaterring attrs [proteinTreatement | radiusSize | polydispersity]



PACKAGE ANNOTATION

class ccp.api.Annotation.MolFeature

- Add derived attr position (derived from MolResidue.seqId)
- Add method getPosition()

new class ccp.api.Annotation.Annotation (Mole bug 263)

- Add attr description
- Add attr fileName
- Add attr url
- Add attr date
- Add attr details
- Add a link to Person (+author)

PACKAGE EXPBLUEPRINT

class aap.api.ExpBlueprint.ExpBlueprint

- Add a link to ccp.api.People.Person called owner

PACKAGE REFCONTAINER

class ccp.api.RefContainer.RefHolder

- Add attr numColumn e.g. 12
- Add attr columnName e.g. 8
- Add attr columnSpacing
- Add attr numRow e.g. 1,2,3,4,5,6,7,8,9,10,11,12
- Add attr rowName e.g. A,B,C,D,E,F,G,H
- Add attr rowSpacing
- Change attr numSubunits -> capacity
- Remove attr vendorName
- Add link to ccp.api.People.Organisation called supplier
- Add attr catalogNum

class ccp.api.RefContainer.RefContainer

- Add attr length
- Add attr wireWidth
- Remove attr vendorName
- Add link to ccp.api.People.Organisation called supplier
- Add attr catalogNum

PACKAGE REFSAMPLECOMPONENT

class ccp.api.RefSampleComponent.AbstractComponent
- componentType enum: lower case as molType in Molecule
- Remove attr vendorName
- Add link to ccp.api.People.Organisation called supplier
- Add attr catalogNum

class ccp.api.RefSampleComponent.Substance (Mole bug 266)
- Add attr pH. This would mean having different Substance objects for phosphate buffers at different pHs.
- componentType enum: add chemShiftRef (should be an open enum)

The substance class is used to store not only mixtures but also components for which you do not want to create a molecule for.
Detergent (NonPolyMolComp or Substance?): Critical Micelle Concentration in pure water is an important parameter

class ccp.api.RefSampleComponent.RestrictionEnzyme
- Change class name RestrictionEnzyme to Enzyme

class ccp.api.RefSampleComponent.Construct
- Add attr constructType = {empty, with target, bac, cosmid, ...}

class ccp.api.RefSampleComponent.MolComponent
- Add attr empiricalFormula
- Add attr expectedMolMass (not derived if descriptor not frozen)
- Add attr measuredMolMass
- Remove attr molecularMass

class ccp.api.RefSampleComponent.MolCompFeature
- Add derived attr length
- Add method getLength = endSeqId – starSeqId
- Add attr status: {cut, uncut, ...}

PACKAGE SAMPLE

class ccp.api.Sample.Sample

- Add link to ccp.api.People.Organisation called supplier or used the experiment import with a link to Organisation ???
- Attr derived attr creator from Experiment.creator
- Add attr catalogNum
- Replace attr initialAmount with three attrs:
  - initialMass
  - initialVolume
  - initialNumber
- Replace attr currentAmount with three attrs:
  - currentMass
  - currentVolume
  - currentNumber
- Add attr displayUnit

class ccp.api.Sample.SampleComponent

- Move attr isotopicLabelling to class SampleComponent.PolyMolComp
- Change attr isotopicLabelling to labellingType, and add an open enumeration {SelenoMethionine, SelenoCysteine, N15, C13, N15+C13, ...} (look at BioMagResBank labelling)
- Replace attr concentration with three attrs:
  - concPerMol
  - concPerMass
  - concPerVolume
- Remove attr concentrationUnit
- Add attr displayUnit

class ccp.api.Sample.AnomalousScatterer

- Add assessmentMethod as link name to ccp.api.Method.Method
- change attr count -> numberOfInstances

class ccp.api.Sample.SampleLocation/HolderLocation

- Check location validity for samples when they are part of an holder and this holder has a particular location

PACKAGE LOCATION

class ccp.api.Location.Location

- Add attr room
- Add link to Laboratory or Organisation ?

PACKAGE PEOPLE

new class ccp.api.People.ContactData

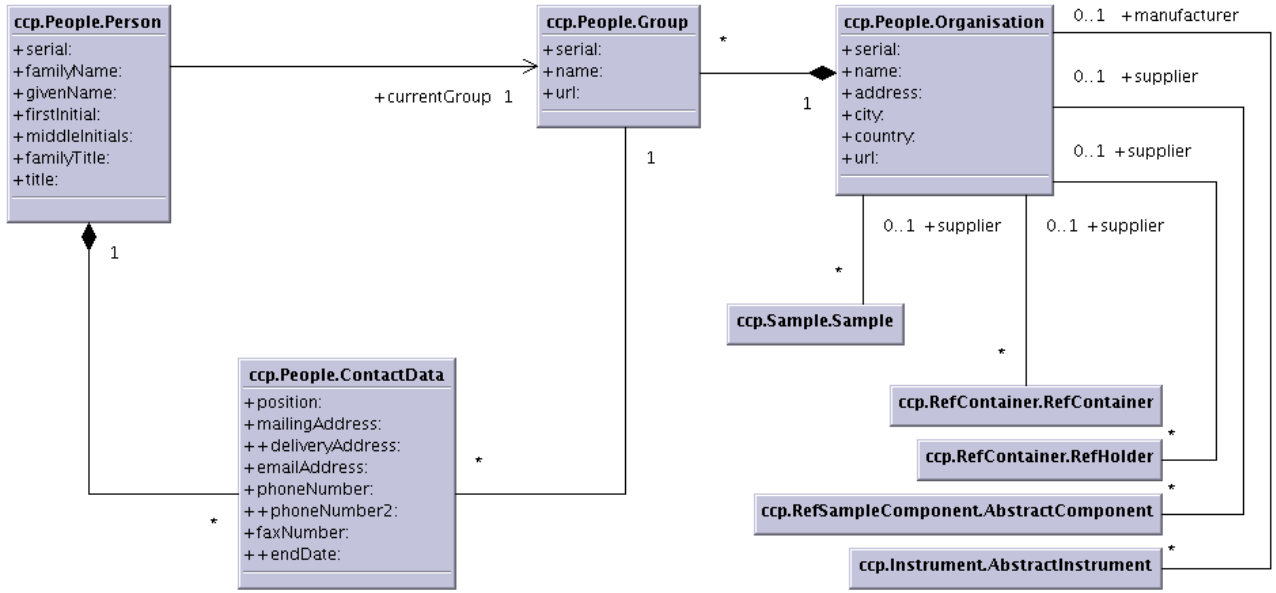
- Add attr deliveryAddress
- Add attr phoneNumber2

class ccp.api.People.Person

- Remove attrs position, addresses and phone/fax numbers because they depend on the laboratory which this person is affilated with.

class ccp.api.People.Laboratory

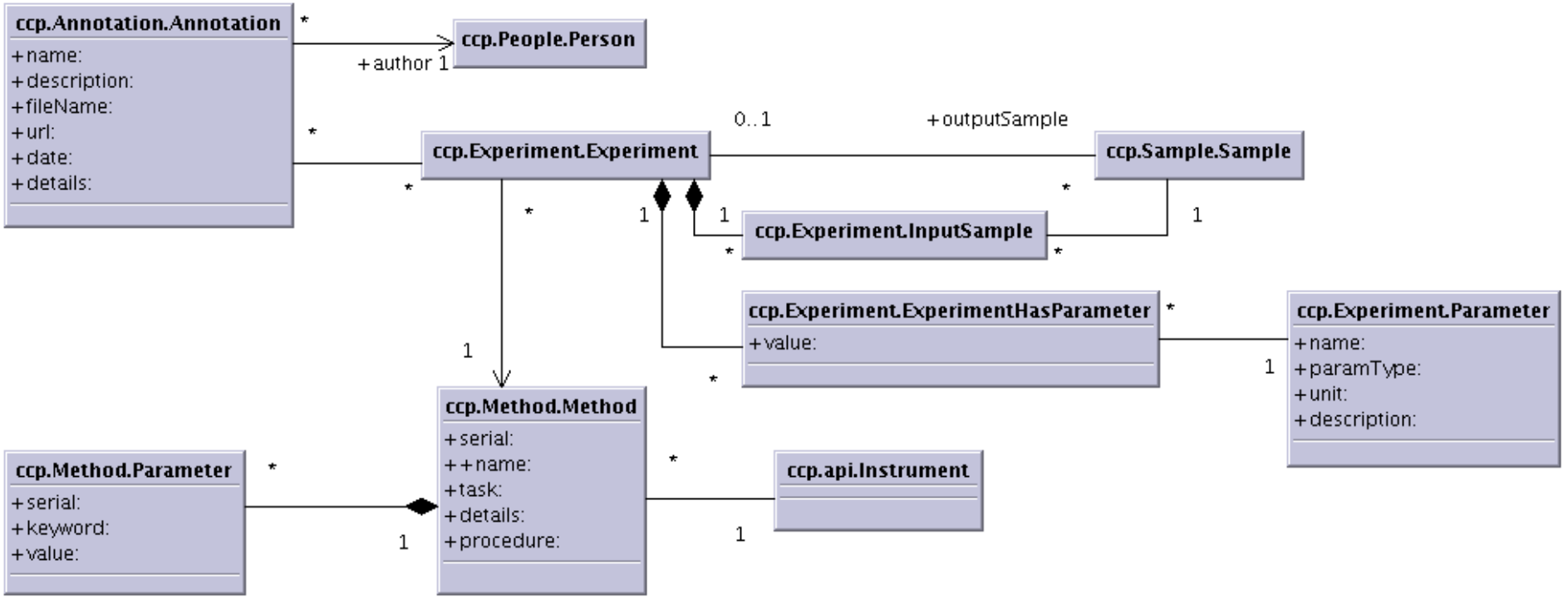
- Change class name Laboratory to Group



PACKAGE METHOD

class ccp.api.Method.Method
- Add attr name

class ccp.api.Method.Parameter
- Add attr unit ???



PACKAGE INSTRUMENT

class ccp.api.Instrument.AbstractIntrument

- Remove attr manufacturer
- Add link to ccp.api.People.Organisation called manufacturer

class ccp.api.Instrument.Intrument

- Add link to Method

new class ccp.api.Instrument.Column (subtype of Instrument)

- Add attr columnType
- Add attr volume
- Add link to ccp.api.Method.Method called loadingMethod

PACKAGE ACCESSCONTROL

Rasmus Questions:

Q1: Are we going to make the access control rigidly database-centric? If not we should maybe define permissions in terms of classes rather than tables, and call the operations read, create, modify and delete?

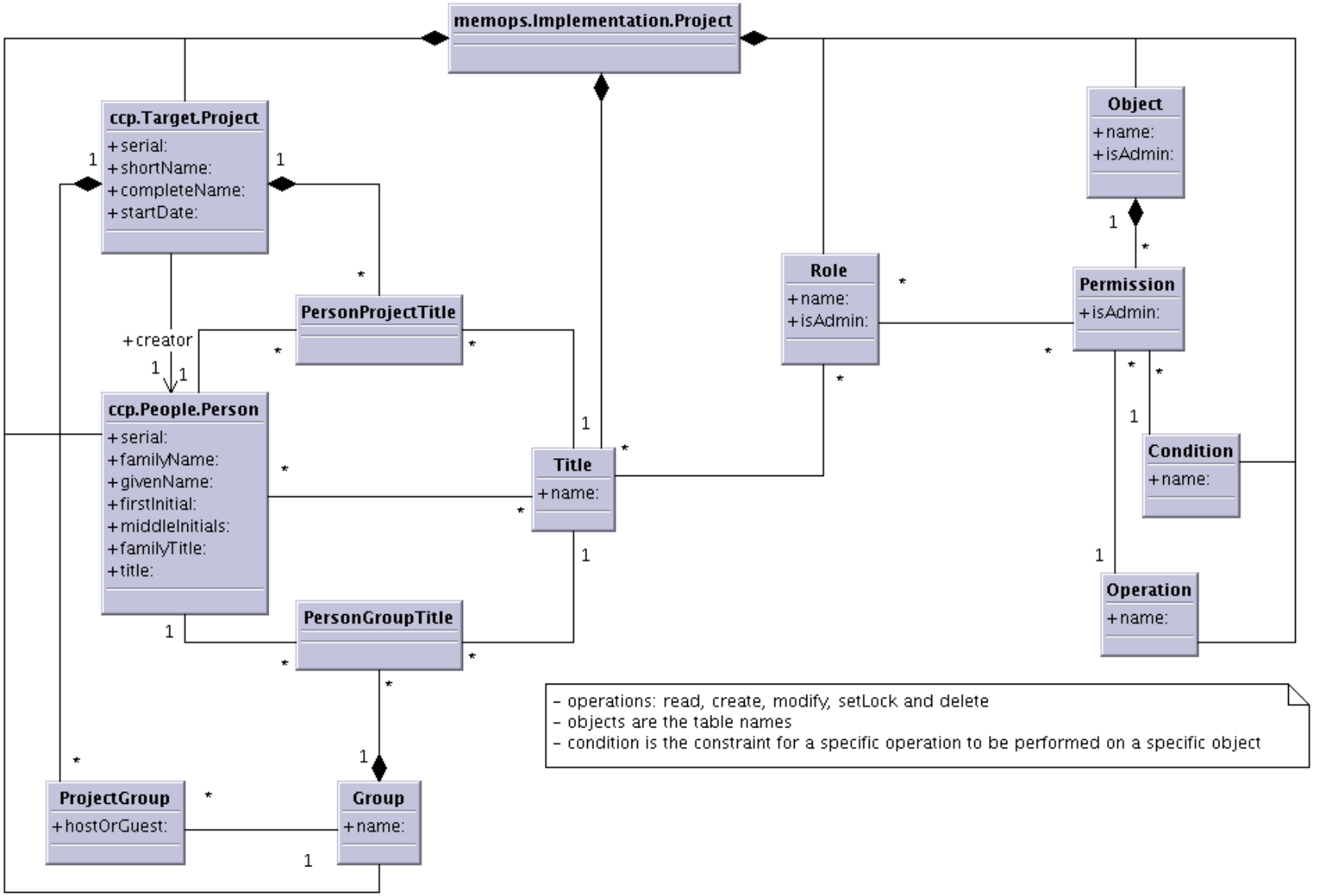
Q2: Can't the regular and admin distinction be handled by a set of parallel subclasses? Or even by a single set of classes with an isAdmin Boolean?

Q3: If there is a limited number of operations (it sounds like operation.name is a closed enum with only four values) why not get rid of the class and have an operationType string attribute in Permission?

Q4: As long as there are only a handful of 'operations', every API function will know what permissions to check when. But how is an implementation going to actually check that a given person is allowed to do what the wants to do? Working off the classes as they stand it would seem to require a lot of navigation - will there be a separate data structure optimised for speed?

Q5: How are the Conditions supposed to be entered, and how is an aplication supposed to parse them, evaluate them, and find out whether the current situation satisfies the condition?

Q6: What information may be needed to evaluate the conditions? If we need creators and owners for every object they have to be put in MemopsBaseClass, how do we do that? How do we find out which Access.Project owns a given object? What will that require of the API/Model?


memops.Implementation.Project
ccp.Target.Project
+serial:
+shortName:
+completeName:
+startDate:
Object
+name:
+isAdmin:
Role
+name:
+isAdmin:
Permission
+isAdmin:
PersonProjectTitle
+creator
ccp.People.Person
+serial:
+familyName:
+givenName:
+firstInitial:
+middleInitials:
+familyTitle:
+title:
Title
+name:
Condition
+name:
Operation
+name:
PersonGroupTitle
ProjectGroup
+hostOrGuest:
Group
+name:
- operations: read, create, modify, setLock and delete
- objects are the table names
- condition is the constraint for a specific operation to be performed on a specific object