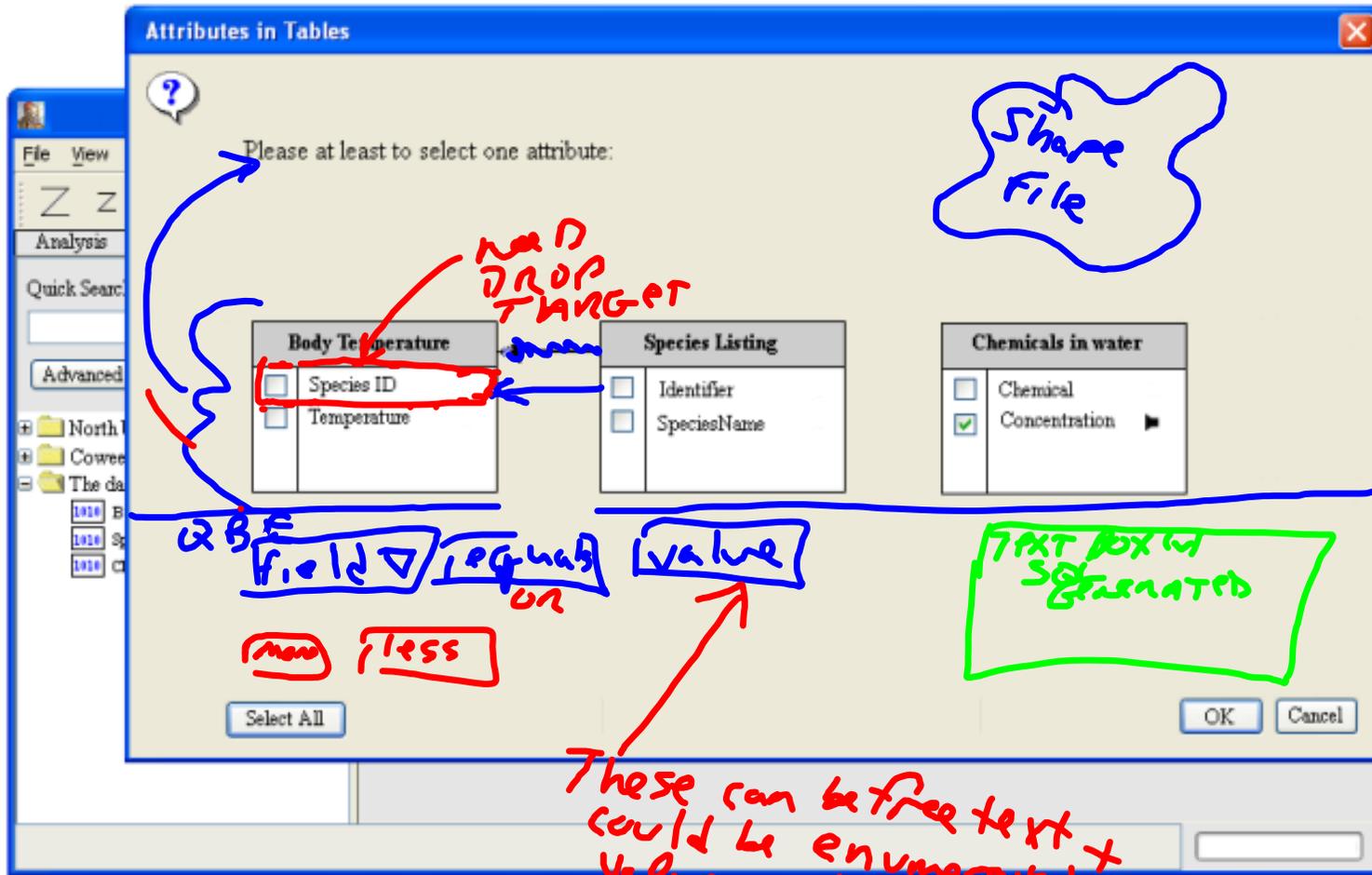
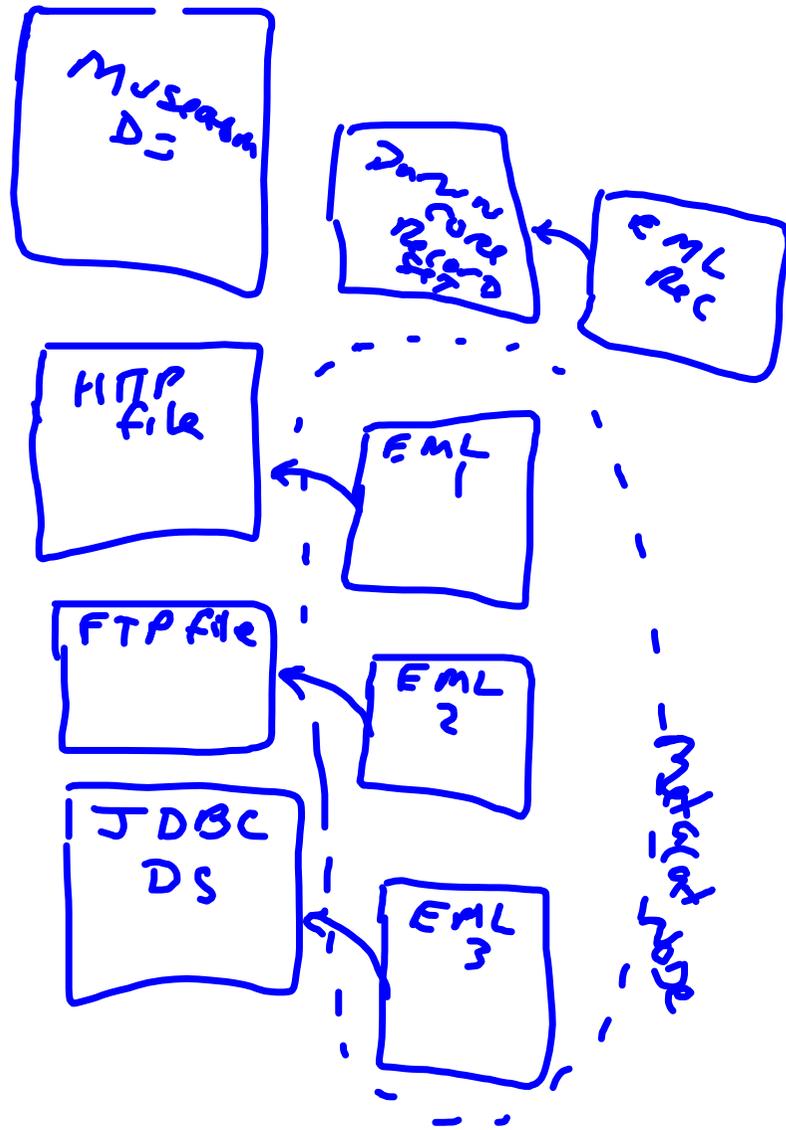


Ptolemy integration

1. Changes needed: efrat has a list -- will check into CVS PT4 MERGE branch as text file
 - issue: should we check into Kepler a snapshot of the Ptolemy CVS tree?
 - alternative 1 <-- matt will write proposal and send to edward
 - we have a version of ptolemy checked into the kepler cvs
 - we sync periodically (e.g., 1 week)
 - run unit and regression tests to be sure no subtle changes were introduced (need testing framework for this)
 - how do we get changes back into ptolemy cvs:
 - one or two kepler people get write access to PT tree (request efrat & chad to start, more if they are willing)
 - review by two other dev'ers before trying to submit to PT
 - submit our changes as quickly as possible to avoid bitrot
 - alternative 2: put kepler into their CVS tree
 - everyone on kepler would need write access to the PT tree, at least the kepler part
 - alternative 3: fork completely, don't worry about their base code
2. Nightly build and testing
 - alternatives:
 - 1) cron job running ant -> email
 - 2) gump <--- chad will install and evaluate
 - 3) tinderbox
 - recommendations for tests and implementaiton of framework ---> chad
 - 1) unit tests for all public methods
 - 2) test workflows that check if output is valid for given input, using ptexecute
 - best to test one actor at a time, unless there are required dependencies
 - write a single unit test that executes all of the moml in a test dir and compares output to known correct output
 - issue: how to record proper output for comparison
 - issue: how to deal with GUI actors? for now, don't include them in tests
 - 3) tests for coding conventions (see existing ptolemy tests for reference)





DIGIR

tuple

← query

resultset →

← data query (SQL)

data results (get) →



Data access UI

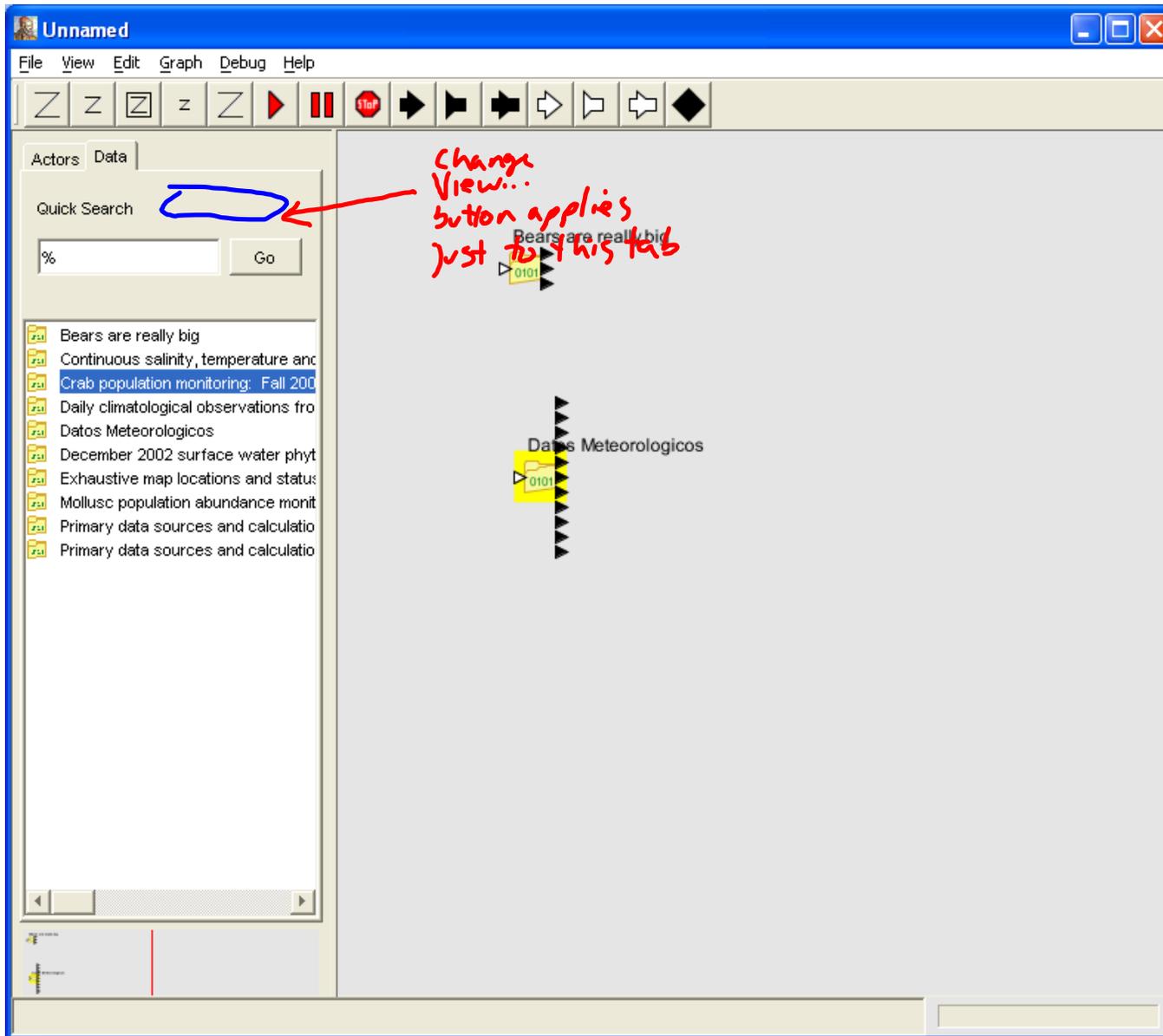
- Rod: 1) basically implement the MS Access QBE interface
- implement "getSql()" method for this interface
 - be sure data source is serializable in MoML
- 7) Implement DarwinCoreDataSource with Jing parallel with EML200DataSource

Data query implementation

- Jing: 2) EML parser that generates DDL for loading data into {SAS, DB, R} (usable in Kepler and in EcoGrid)
- 3) Query engine implementation in Kepler for one of the above to execute the query from the UI Rod produces
- 4) Define and implement EcoGrid "dataQuery" based on 2+3 above
- 5) view metadata at user request (possibly use morpho's data package display)
- 6) view data at user request (possibly use morpho's data package display)

Efrat: How can shape files and other spatial formats be integrated into data search?

- GEON searches using GeographyNetwork to locate shape files
- need to integrate ArcCatalog type of tools into EcoGrid API
- what metadata is used to describe the shape files? probably FGDC



The image shows a software application window titled "Unnamed" with a menu bar (File, View, Edit, Graph, Debug, Help) and a toolbar with various icons. On the left is a file explorer with a tree view containing folders like "utilities", "director library", "actor library", "sources", "kepler", "EML", "GARP", "GRASS Web Services", "geon", "GIS", "nmiworkflow", "resurgence", "spa", "Util", "web services", "sinks", "io", "math", "random", "flow control", "real time", "logic", "conversions", "array", "matrix", "signal processing", "domain specific", "dynamicActor", "more libraries", and "user library". The "actor library" folder is highlighted in blue. A blue circle around the "Data" tab in the top-left corner contains the handwritten text "Data...".

The main workspace contains several elements:

- A yellow folder icon labeled "0101" with the text "Bears are really big" above it.
- A vertical stack of yellow folder icons labeled "0101" with the text "Data Meteorologicos" to its right.
- Handwritten blue text "Modeling" with two arrows pointing to "Population Model" and "Physiology Model".
- Handwritten blue text "Statistical Method" with three arrows pointing to "Linear model", "neural net", and "genetic algorithm".
- A blue circle around the "Data Meteorologicos" folder icon with a handwritten "7" below it.

View

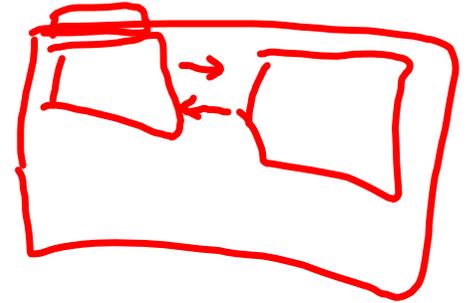
Ecology	X
Modeling Techniques	
Statistical Methods	
Owner	
Discipline	1

Community is top level ontology



	C	D	∅
A	AC.	AP.	
B	BC.	BD.	
∅			

Floating tool?



Ontology-based Browsing of data and actors tasks

1. Generate simple browsing ontologies for use in Kepler (Rich, assistance from Chad, Mark, Matt)
 - examples: habitats, spatial regions, top level taxa, (see the KNB web page)
2. a) Design UI via screenshots and collaboration diagrams for browsing with ontologies (Chad, Matt)
 - includes changes to actor/data tree and dialogs for choosing views
 - for now, punt on the user-configurable views, but keep it in mind for designb) Prototype and implement (Chad, Matt)
3. Design and prototype interface to execute sparrow in kepler (Shawn, Chad, Efrat, Bertram)
4. Tagging data and actors
 - a) UI -- kepler/morpho/standalone/web interface? (Chad) (lowest priority of things on this page)
 - b) generate proposal on how to extend MOML, EML, etc, feedback and implement (Shawn, Chad)
 - c) need mechanism for submission of registrations that are not from original owner (Chad)
 - store registrations in either or **both** of a) the EML/MOML/etc document, or
 - b) as external document that references the originals
 - d) Chad will arrange conference call with shawn, chad, matt to generate concrete proposal for this stuff
 - e) tag representative set of data and actors using the ontologies from (1) (Mark, Rich, Deana)

Goal: kepler release by Oct 15, 2004 (as specified in Edinburgh)

Everyone: enter summary of their tasks from these 9 in bugzilla

Actors for Sampling

-
- Dan demonstrated sequence and random sampling actors in Kepler
- discussion of using R for sampling rather than Kepler -- advantages of using R's pre-canned and extensive routines
- Dan has R (almost) working using the interactive commandline actor
 - initial step: R uses STDIN, STDOUT, and commandline params only, no typing
 - more sophisticated: type R script port requirements and expose in Kepler

General User Profile

- * Somewhat savvy user
- * good with rudimentary digital data
 - o Text, Excel, maybe Access
- * no assumptions about general programming knowledge
- * Probably does not know SQL
- * Probably does not know relational DB theory or practice
- * Some (10%) maybe analytical experts with a lot of experience in creating workflows

General Profile of their Data

- * Maybe multiple tables
 - o But probably not relational (not normalized)
 - o more than likely one table per year or per species etc.
- * Low probability of 100s of tables
- * Large tables
 - o Many columns (attributes)
 - o Many rows (records)

DB Capabilities

Text	A lot	1
Excel	Most	1
Access	Some	2
RDBMS	A few	3

General Profile of Their Analytical Abilities

- * Import heterogeneous data
- * Understand statistical based algorithms
- * Good with statistical problems
- * Algorithm knowledge very narrow (depth, not breadth)
- * Understand known (or generic) functions of systems

Analytical

Matlab	A lot	1
SAS	A lot	1
S+/R	A lot	1
General Programmers	A few	3

Evolution of User

- * Duplicate an existing problem with their local data
 - o Easily importing their existing data
 - o Importing Algorithms from: Matlab, SAS, R, (GRASS?)

Value Add

- * New Data (out on the Grid)
- * Data Prep
- * Graphics (Plotting)
 - o More like for SAS users than Matlab
- * Workflow Sharing?
- * Workflow is documentation for the resultset (information loss as to how the data was created)
- * Providence
- * Ease of use for doing analysis
- * Visually document the process/analysis
- * Generate a Web document?