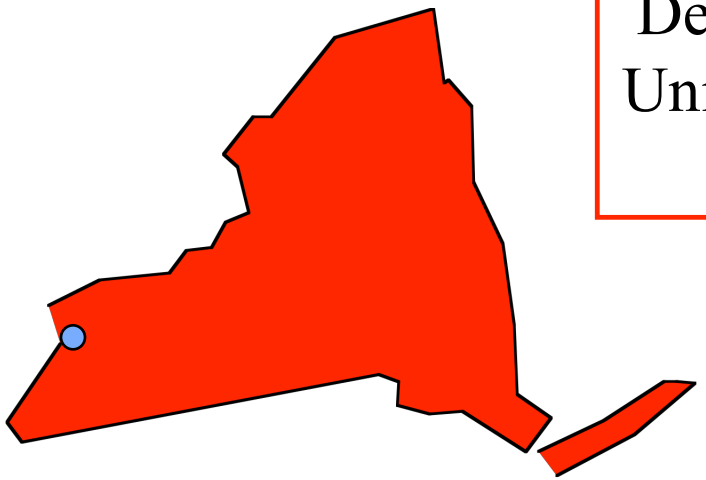
All About Potentials and Iterators

David A. Kofke

Department of Chemical Engineering
University at Buffalo, State University of New York



UNIVERSITY AT BUFFALO
State University of New York

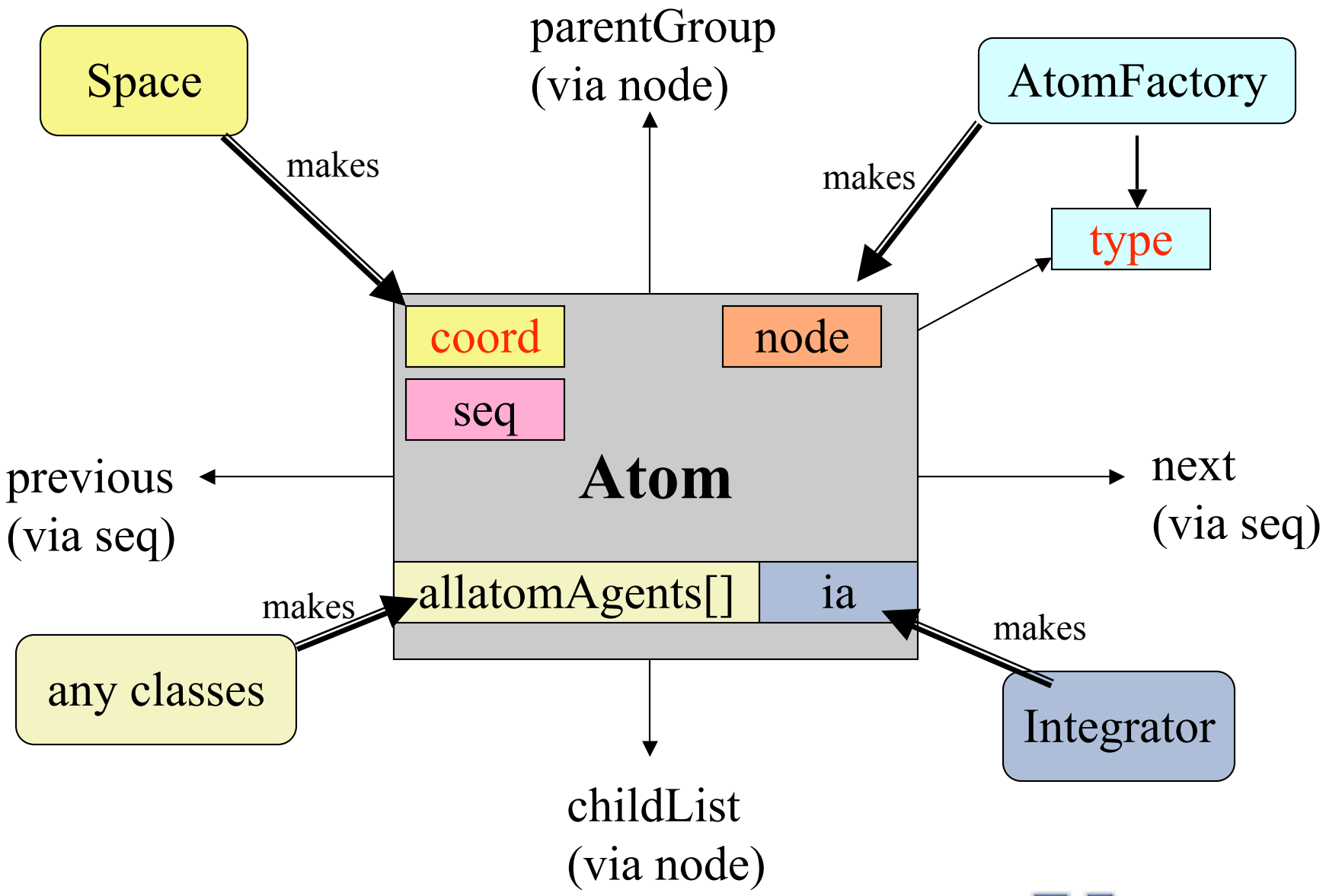
Potential

- Top-level simulation element
  - Species, Potential, Integrator, Controller, Meter, Device, Display
- Defines methods and holds fields that define interactions between atoms
- Potential is responsible for knowing the atoms to which it applies
  - Uses appropriate iterators for this purpose
- Hierarchical structure that parallels structure of species hierarchy

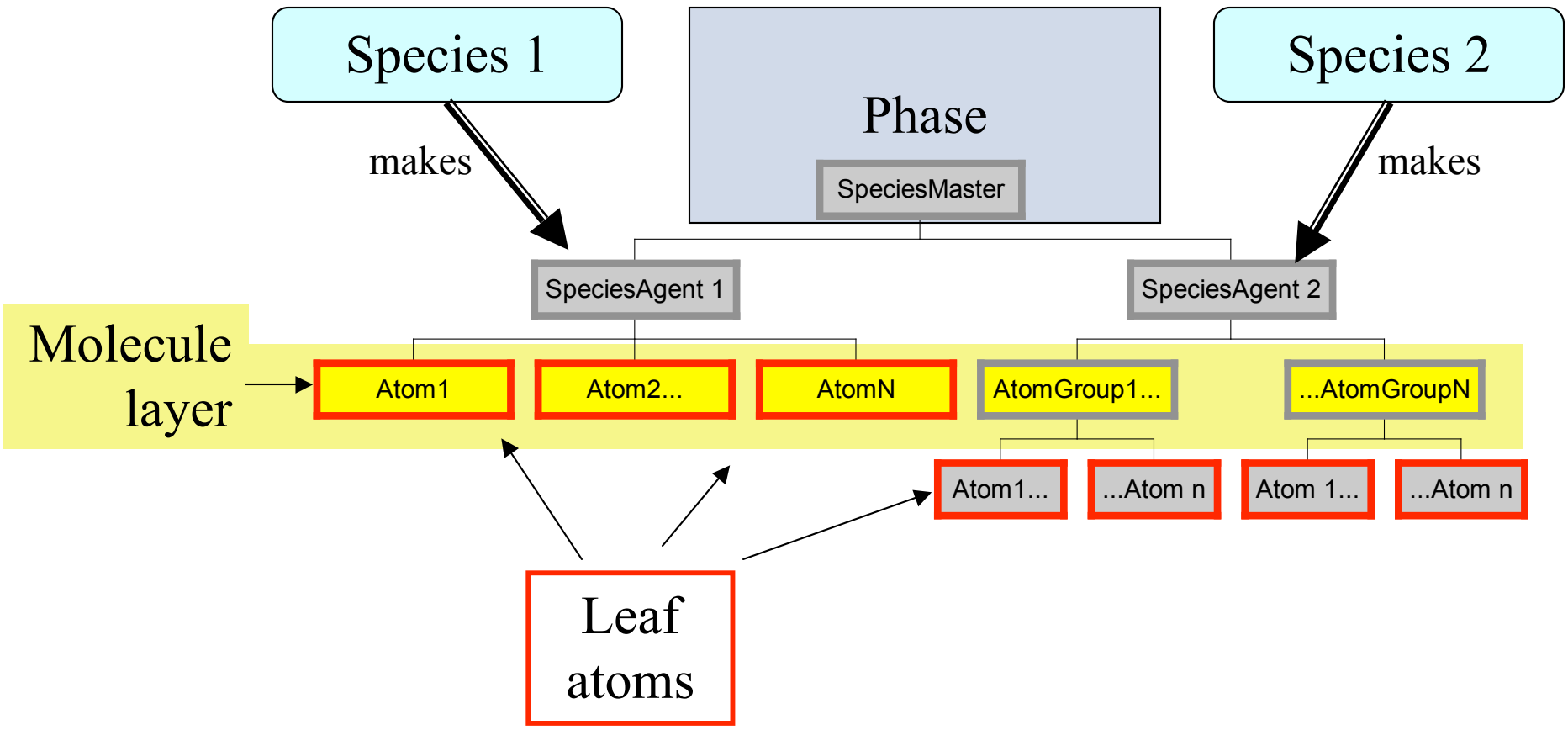
Atom

- Corresponds to a physical atom (in simplest case)
- Holds data relating to
  - Spatial coordinate (position and/or momentum)
  - Atom type
  - Position in atom hierarchy
  - Position in atom sequence
- Can also hold auxiliary data
  - Agents from other classes
  - Parameters specified by other classes (in type field)
  - Lists of other atoms for use by other classes

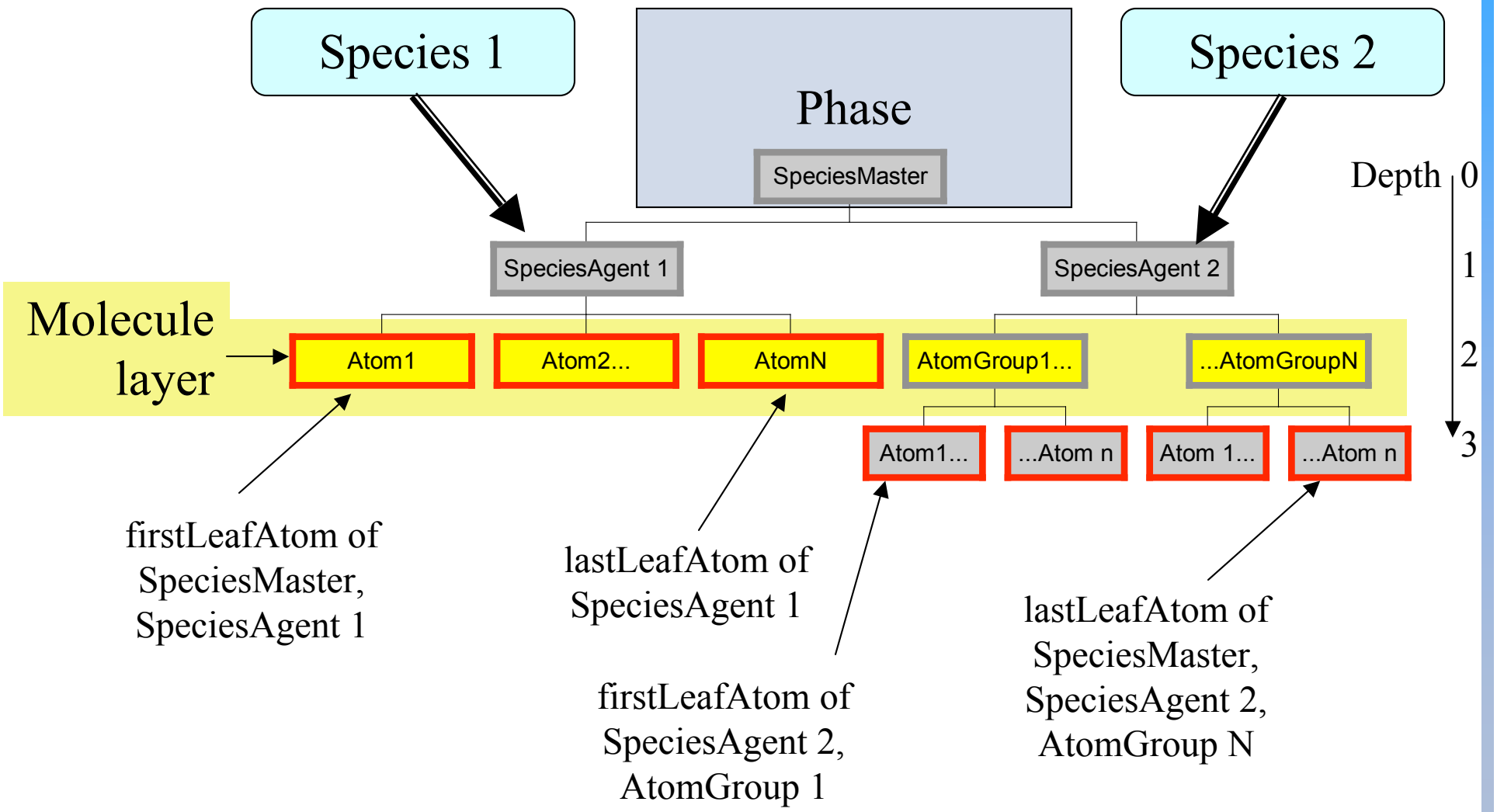
Atom Fields



Species Hierarchy

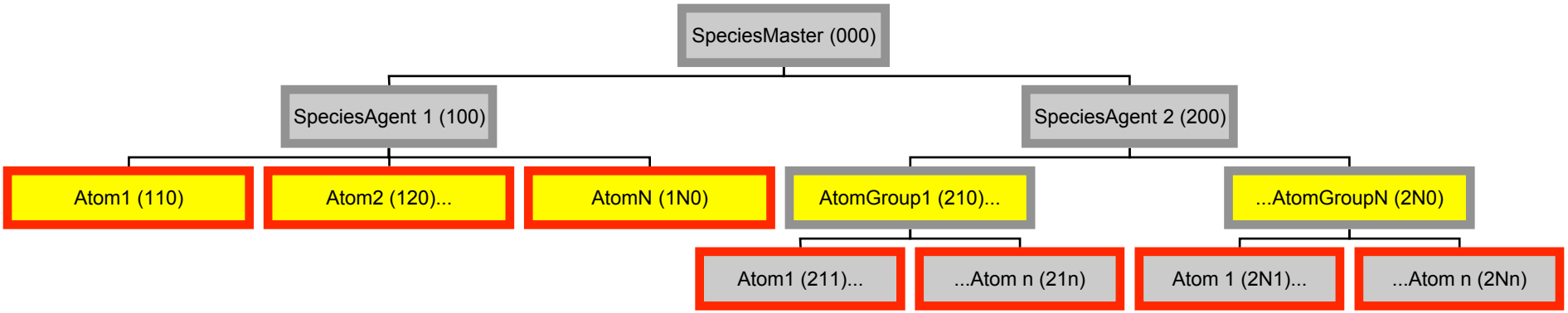


Species Hierarchy



Ordering

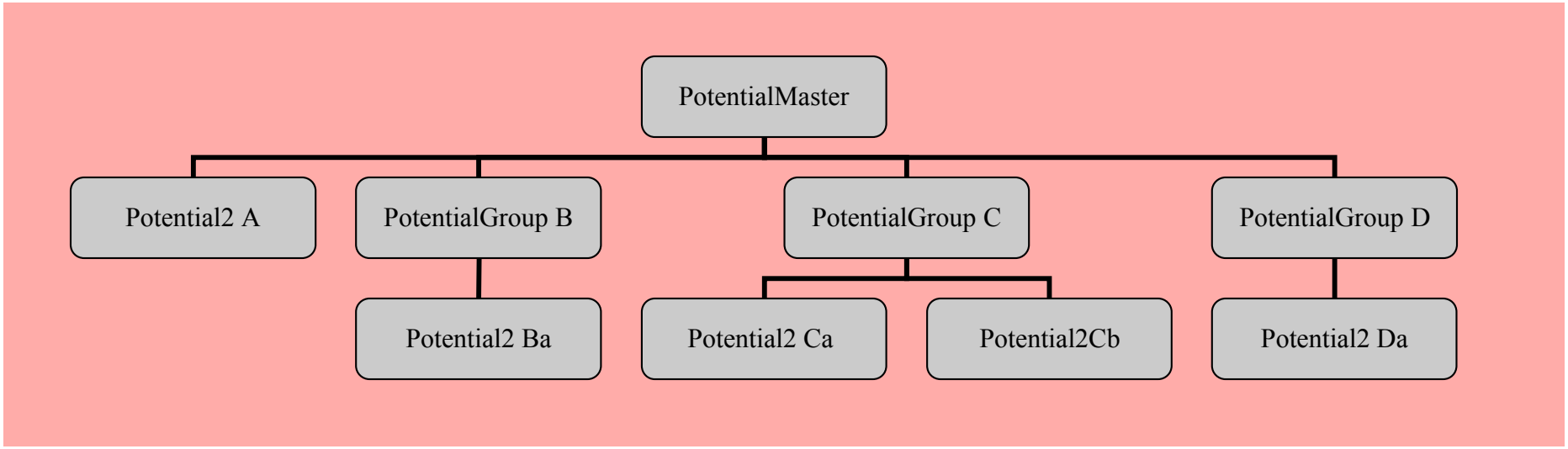
- Atoms are ordered to facilitate looping through them

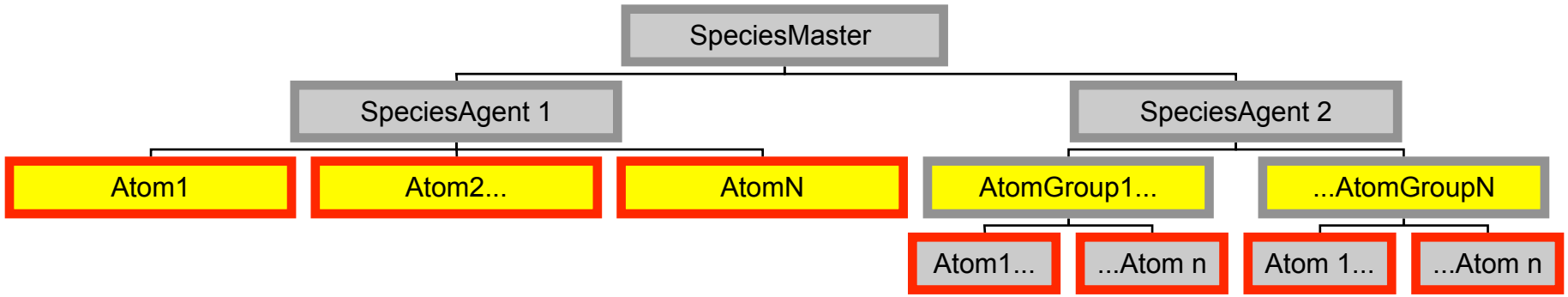


- 000 → 100 → 110 → 120 →…→ 1N0 → 200 → 210 → 211 →…→ 21n →…→ 2N0 → 2N1 → 2Nn
- Looping might be required to go up or down list, from any point, and at any level in hierarchy
- Ordering might be fixed, or might depend on spatial configuration (cell neighbor list)

Potential Hierarchy

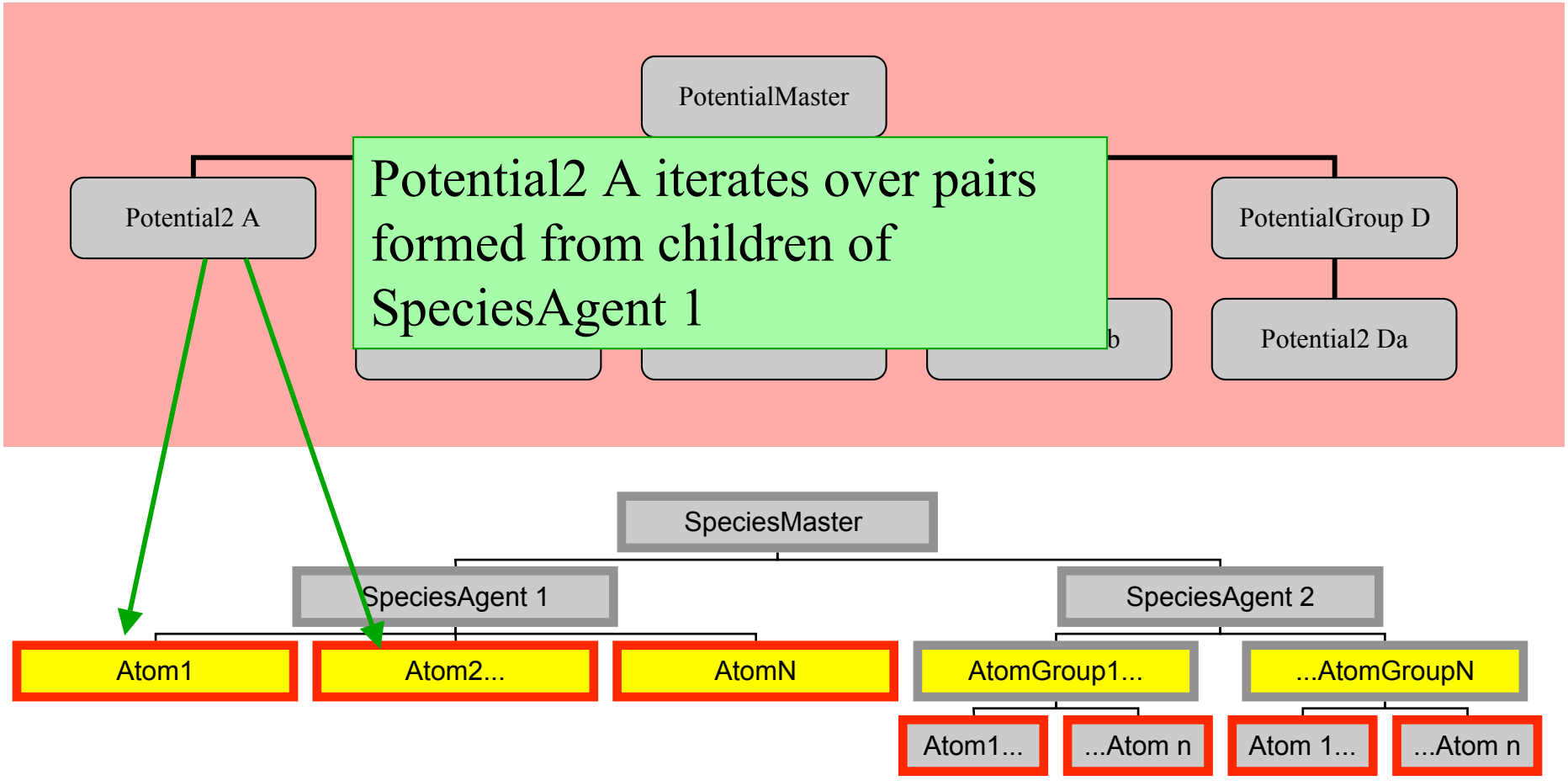
- Potential hierarchy parallels species hierarchy
  - Higher-level potentials define interactions between larger groups of atoms





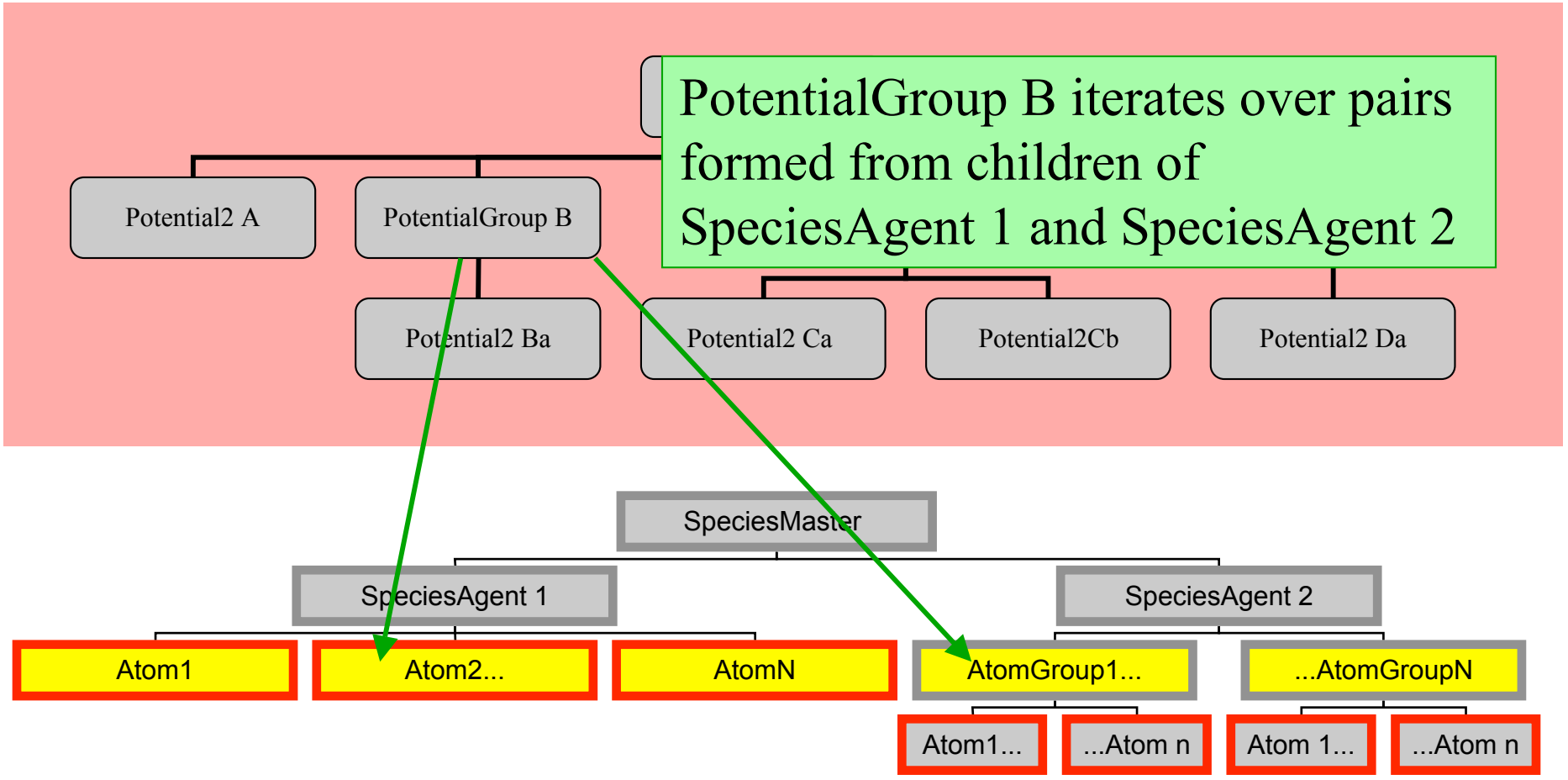
Potential Hierarchy

- Potential hierarchy parallels species hierarchy
  - Higher-level potentials define interactions between larger groups of atoms



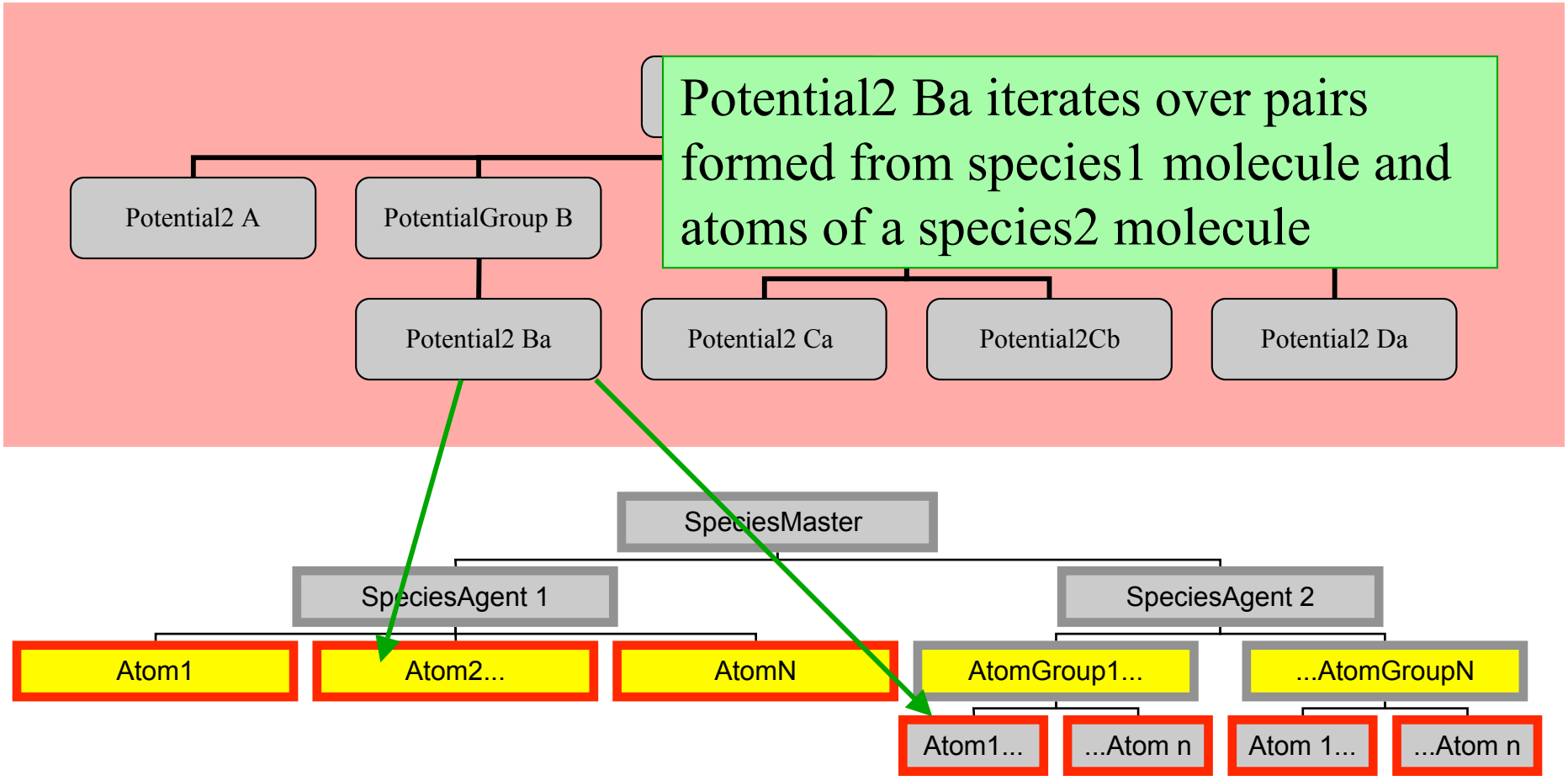
Potential Hierarchy

- Potential hierarchy parallels species hierarchy
  - Higher-level potentials define interactions between larger groups of atoms



Potential Hierarchy

- Potential hierarchy parallels species hierarchy
  - Higher-level potentials define interactions between larger groups of atoms

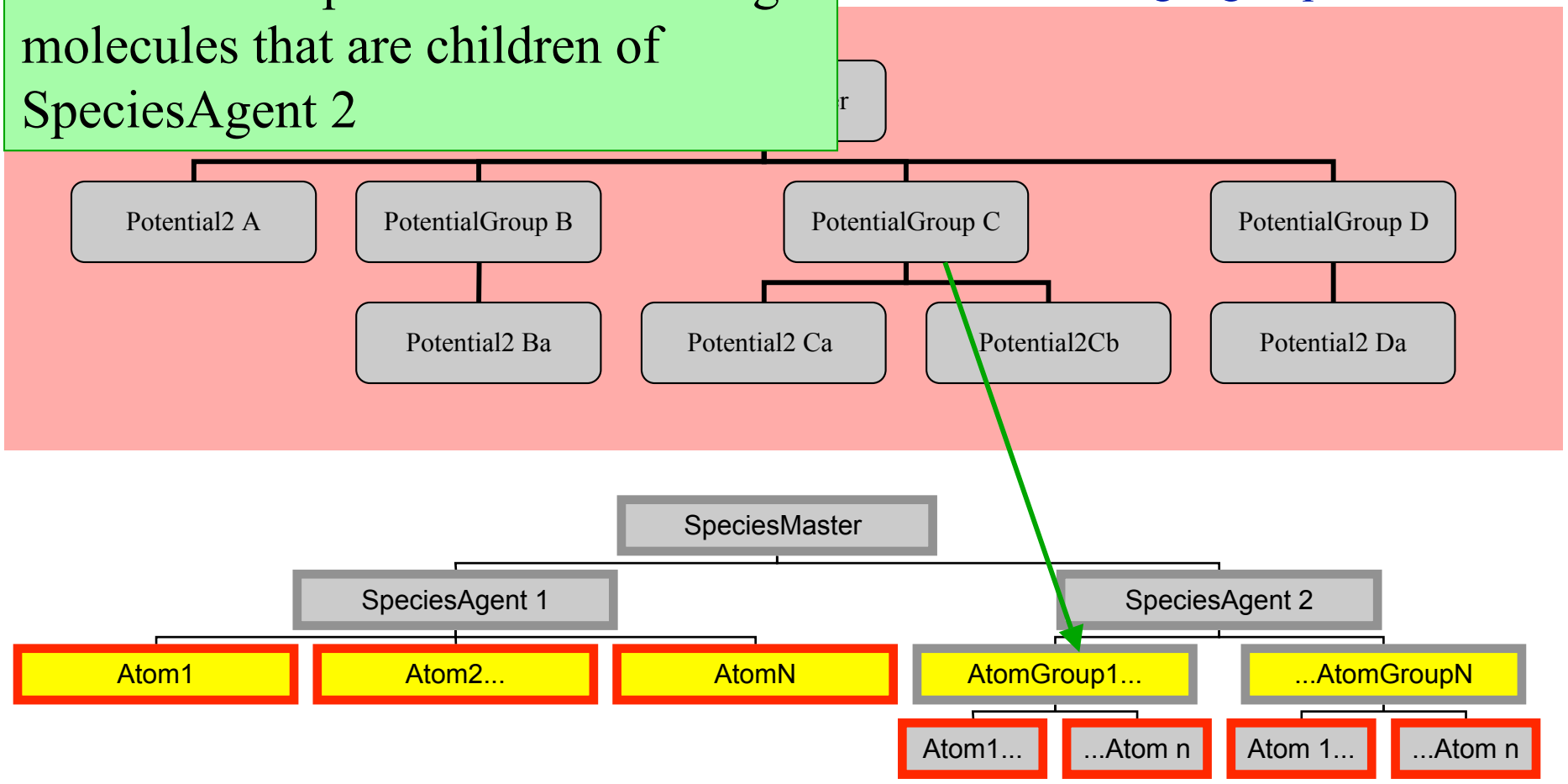


Potential Hierarchy

- Potential hierarchy parallels species hierarchy

PotentialGroup C iterates over single molecules that are children of SpeciesAgent 2

ctions between larger groups of atoms

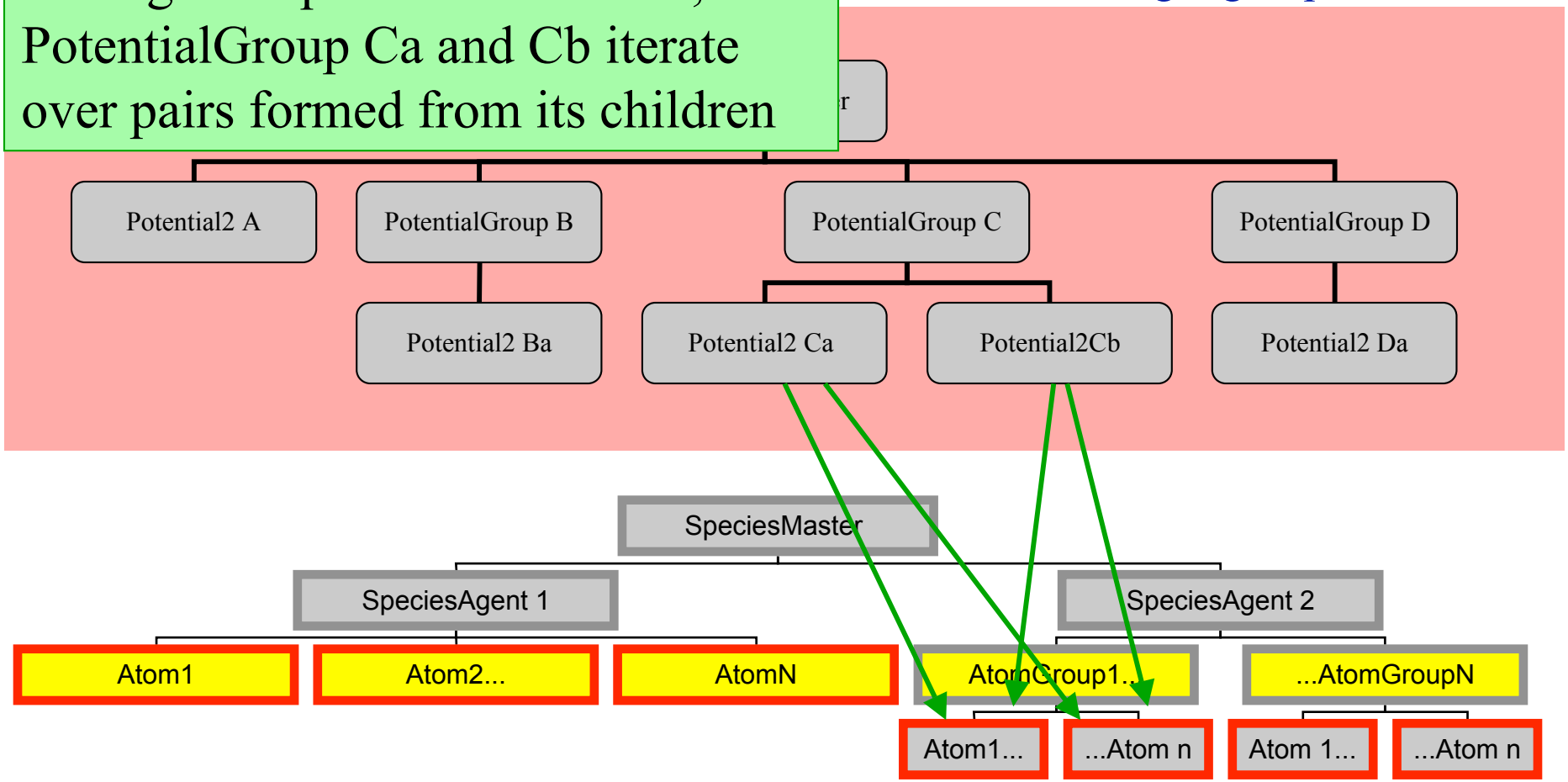


Potential Hierarchy

- Potential hierarchy parallels species hierarchy

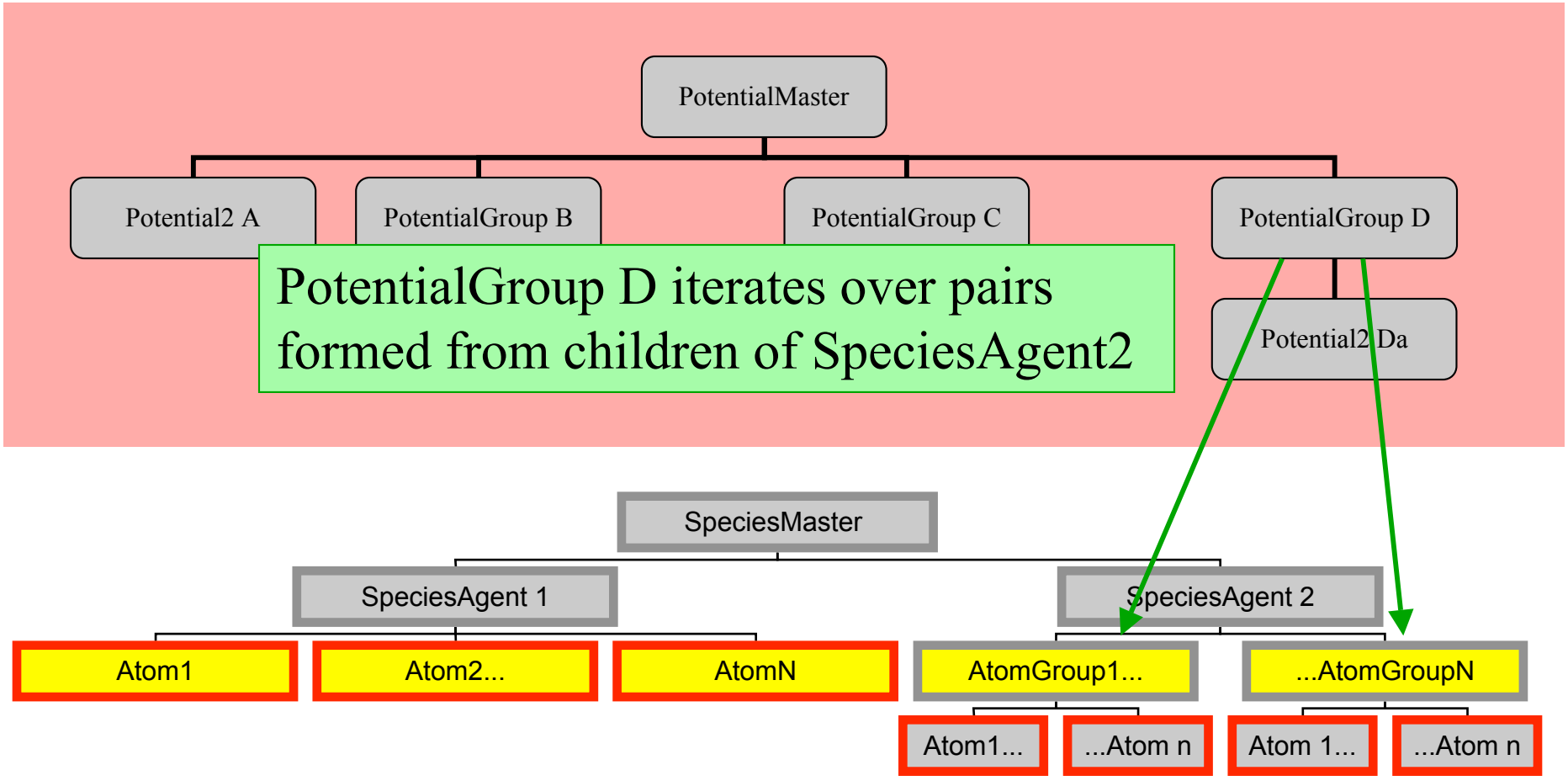
For a given species-2 molecule, PotentialGroup Ca and Cb iterate over pairs formed from its children

ctions between larger groups of atoms



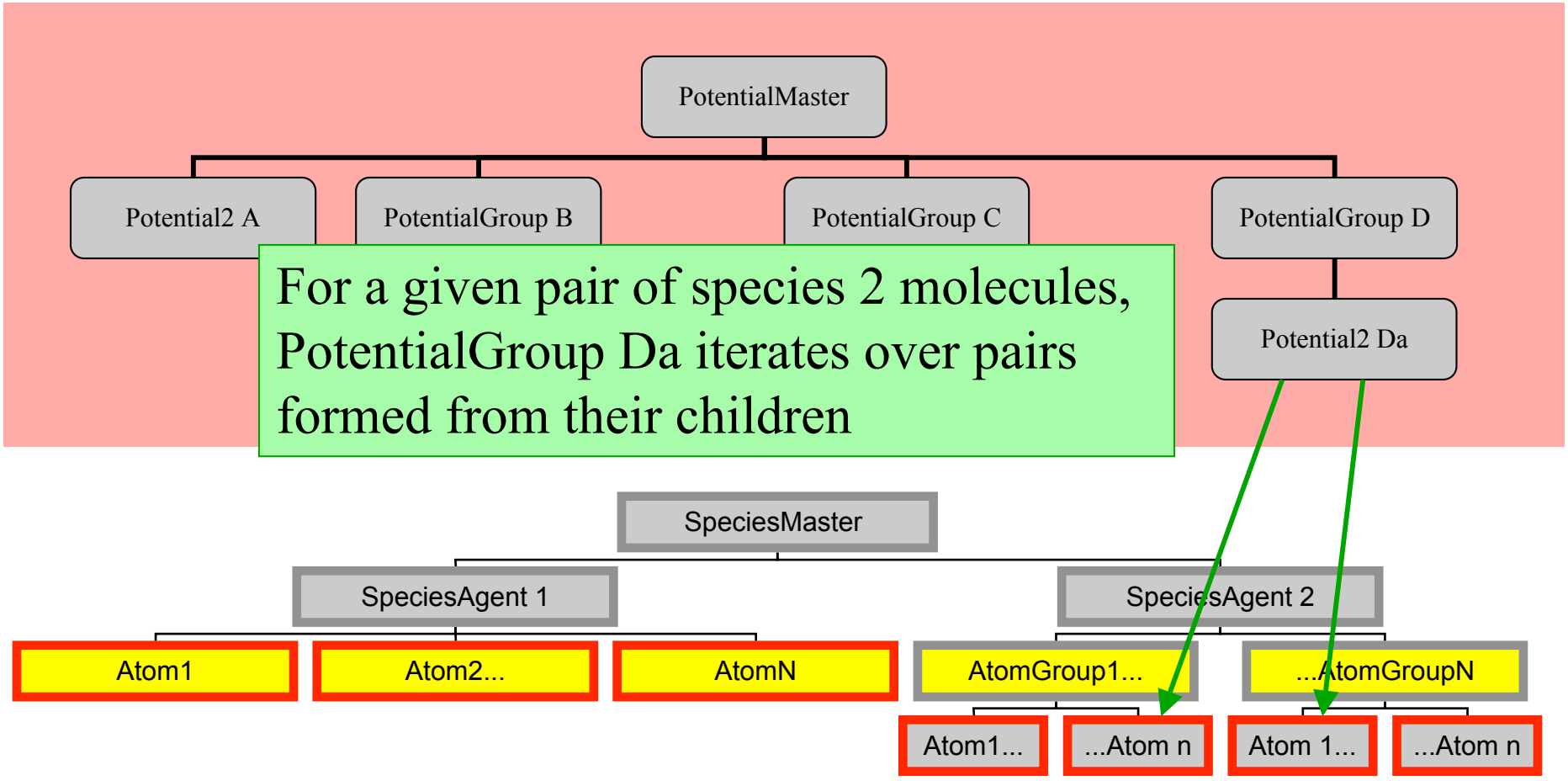
Potential Hierarchy

- Potential hierarchy parallels species hierarchy
  - Higher-level potentials define interactions between larger groups of atoms



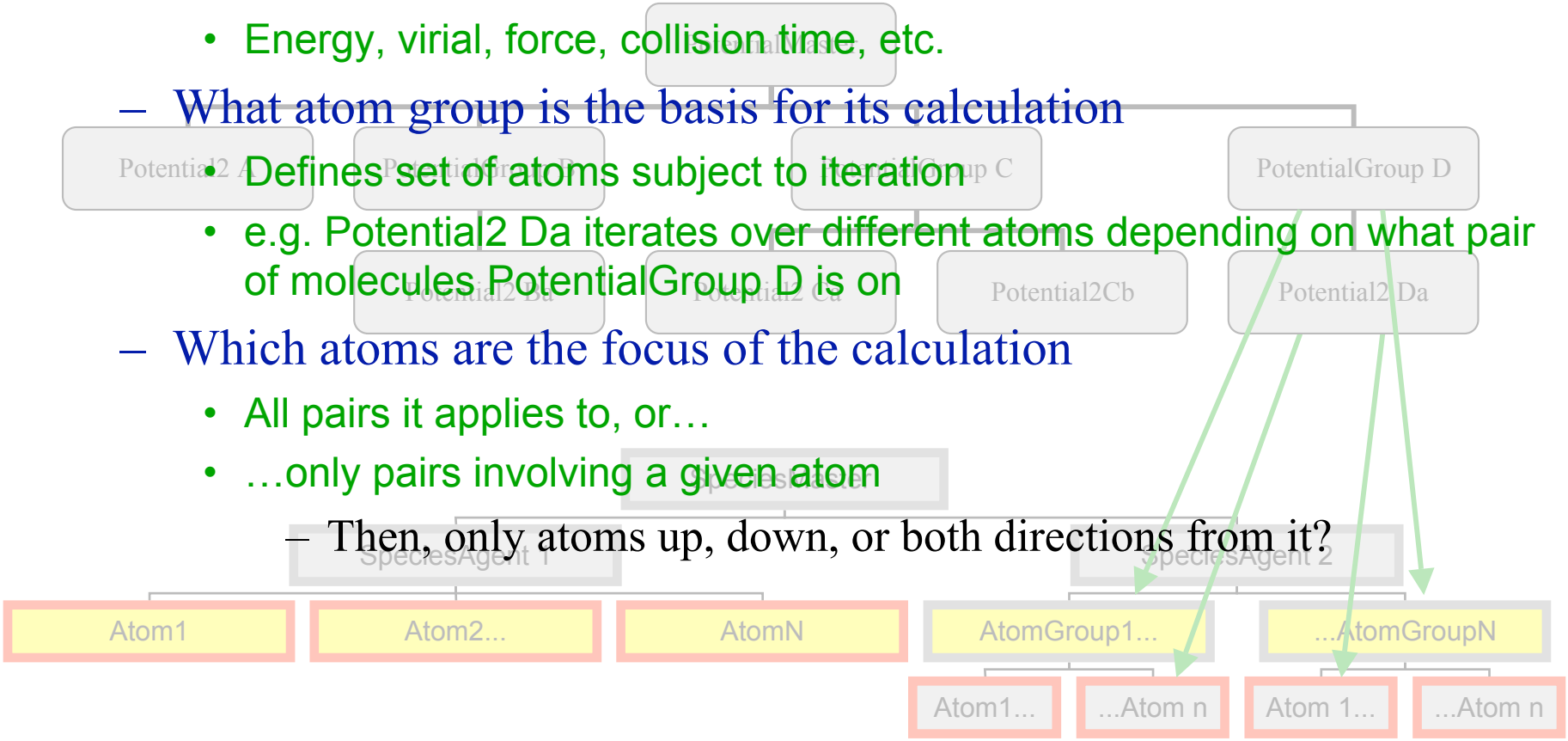
Potential Hierarchy

- Potential hierarchy parallels species hierarchy
  - Higher-level potentials define interactions between larger groups of atoms



Using the Potentials

- To get information from potential, we need to tell it
  - What calculation is to be done
    - Energy, virial, force, collision time, etc.
  - What atom group is the basis for its calculation
    - Defines set of atoms subject to iteration
    - e.g. Potential2 Da iterates over different atoms depending on what pair of molecules PotentialGroup D is on
  - Which atoms are the focus of the calculation
    - All pairs it applies to, or…
    - …only pairs involving a given atom
      - Then, only atoms up, down, or both directions from it?



Using the Potentials

- Basic format (current)

Current form

```
potential.set(basis).calculate(iteratorDirective,
                                        energyCalculation);
```

New form

```
potential.calculate(basis, iteratorDirective,
                                        energyCalculation);
```

- Arguments
  - **basis**: Defines set of atoms subject to iteration
  - **iteratorDirective**: specifies type of iteration desired
  - **energyCalculation**: contains method to perform desired calculation
- Process (examine code, but currently under revision)
  - calculate method sets up potential's iterator for looping, and instructions in energyCalculation are performed

Iterator Directive

- IteratorDirective holds fields that direct iteration
  - Key attributes (among a few others)
    - direction
      - UP, DOWN, BOTH, NEITHER
    - atom
      - iteration performed in reference to atom, if not null
- Basis defines the source of atoms for the iterator
  - Within a given basis, iterator may loop over atoms in different ways
    - Depends on iteratorDirective
    - Depends on design of iterator
      - All children
      - Children two (e.g.) steps deeper
      - “neighbors” only
      - Only ones bonded/not bonded to an atom
      - Only first or last child
      - Etc.

Using an Iterator

- Perform actions on iterates (pass atoms to action)

```
iterator.reset();
while(iterator.hasNext()) {
    Atom atom = iterator.next();
    //do something with atom
}
```

- Pass action to iterator

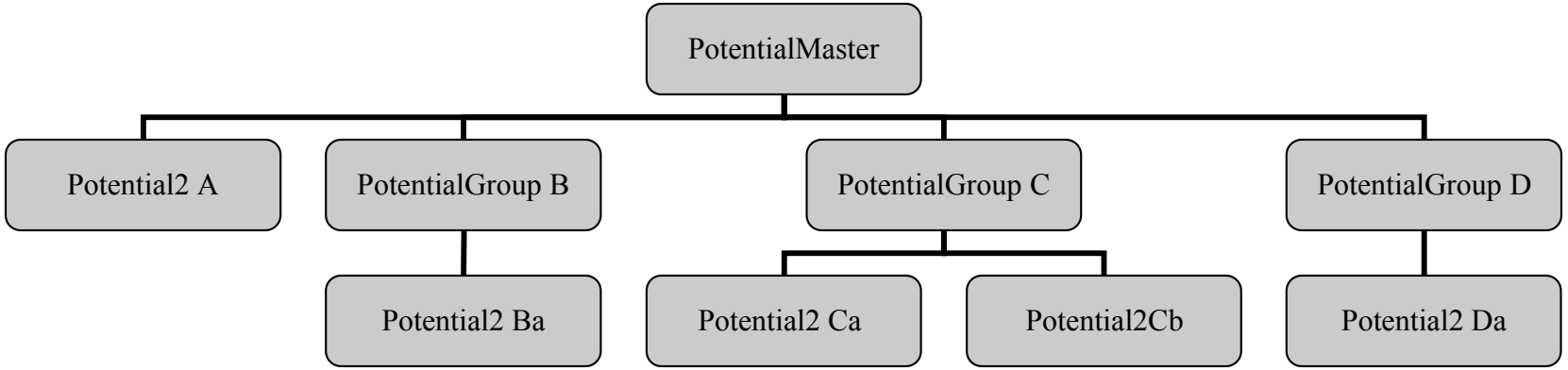
```
AtomAction action = ...//define action
iterator.all(basis, directive, action);
     //performs action on directed atoms
     //in basis
```

Threading in Java

- Independent process of calculation
  - In single-processor platforms, control moves from one thread to another in unpredictable ways
  - In multi-processor platforms, each processor can be running a different thread
- Different threads may be using the same data, or the same instances of objects
  - Must be careful that threads do not interfere or corrupt each other's activities
- Possible paradigm for conducting parallelized simulations
  - Different parallel-tempering phases running on individual threads
  - Simulations on a single phase might use different threads to advance simulation in different regions of space

Problem with Multithreading

- Only one potential hierarchy is instantiated
- Threads running in different phases operate on hierarchy simultaneously
  - E.g. Thread 1 has Potential2 A iterating over atoms in its phase
  - In the middle of this, Thread 2 comes in and resets iteration for atoms in another phase



Solutions to Multithreading Problem

- Construct identical potential hierarchies, each devoted to a single thread
  - Complicated by need to ensure that potentials always remain identically defined across all instances
  - Modifications (change in parameters or addition/removal of a potential) must be transmitted to all instances
  - Difficulty can be alleviated by defining hierarchy of potential agents
    - Agents have individual iterators
    - Agents refer to single, common hierarchy of potentials for their definition
- Use stateless iteration (demonstration)
  - Pass calculation to iterator
  - Much easier, less error-prone to program, understand, and maintain
  - Performance issues?
  - Current modifications are now following this path