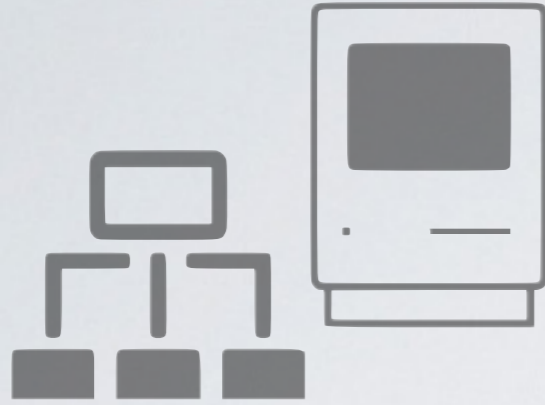




THE UNIVERSITY *of* EDINBURGH  
**informatics**

*This work was funded by Microsoft Research through  
their European PhD Scholarship Program.*

Microsoft®  
**Research**



Devices



Security

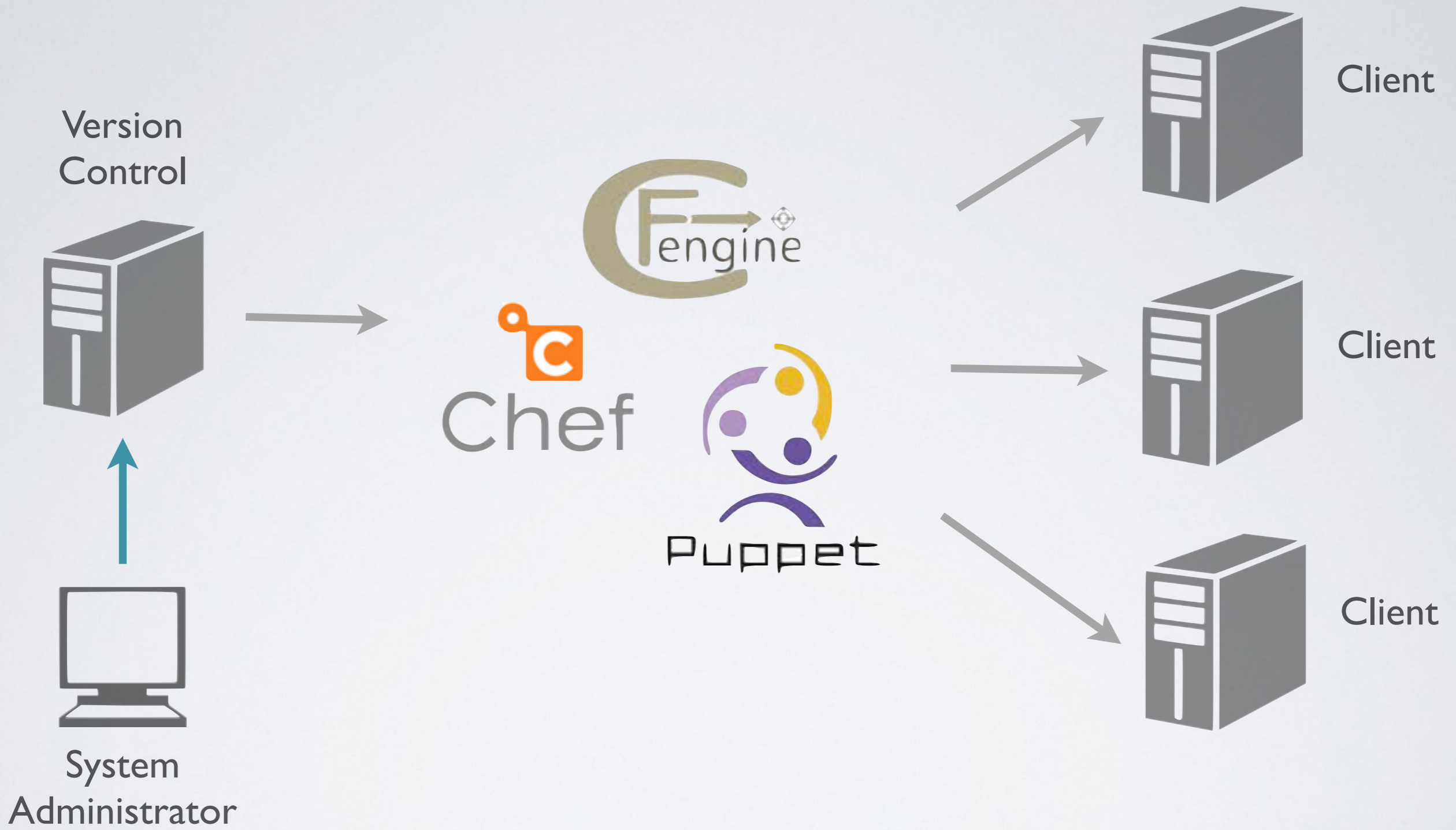
# System Configuration



Cloud



Enterprise



```
sudo apt-get -y install apache
```



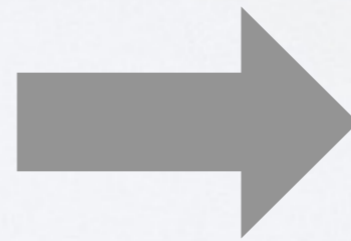
```
package {'apache':  
  ensure => installed  
}
```



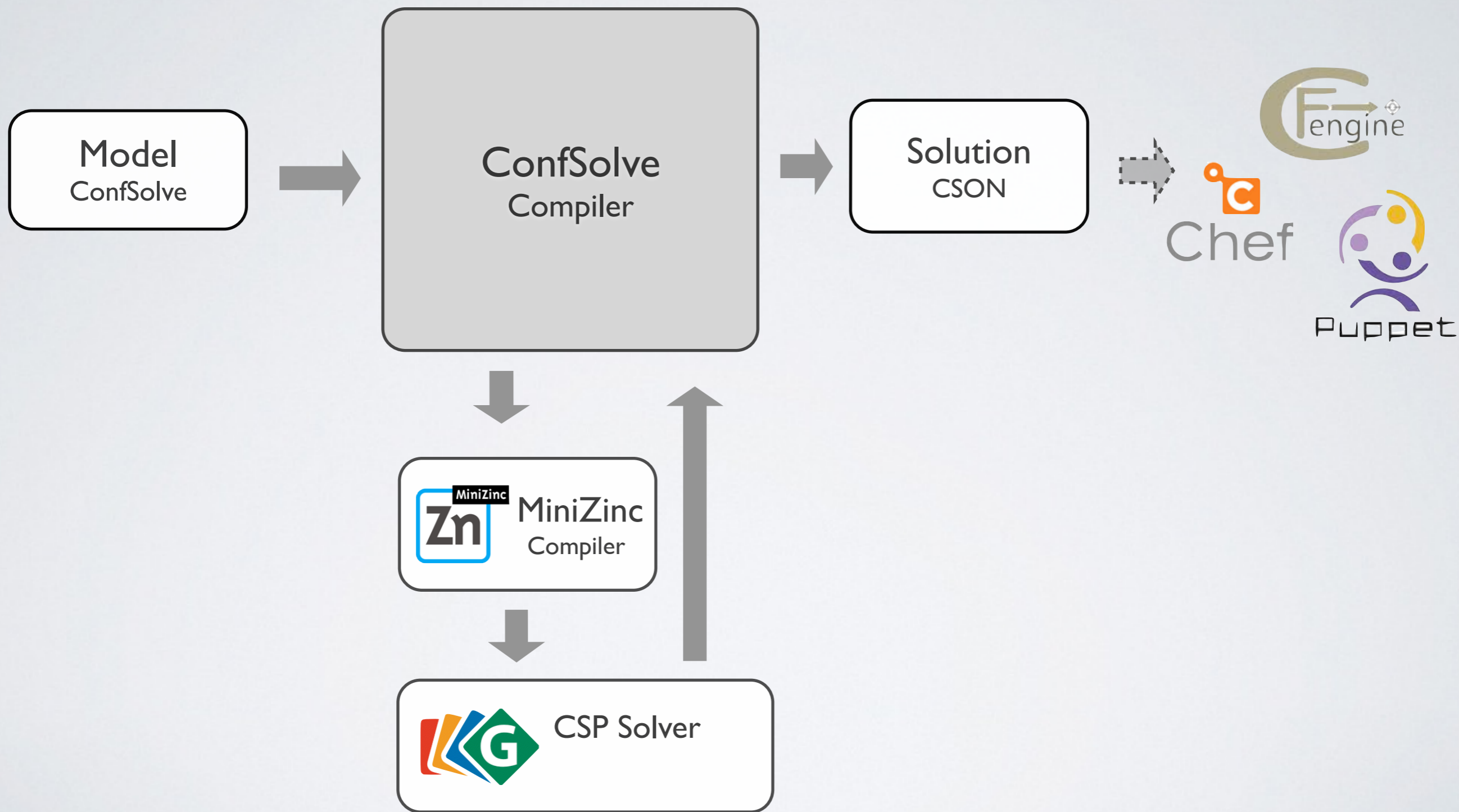
```
all_different(xs, ys)
```

```
contains(x, xs)
```

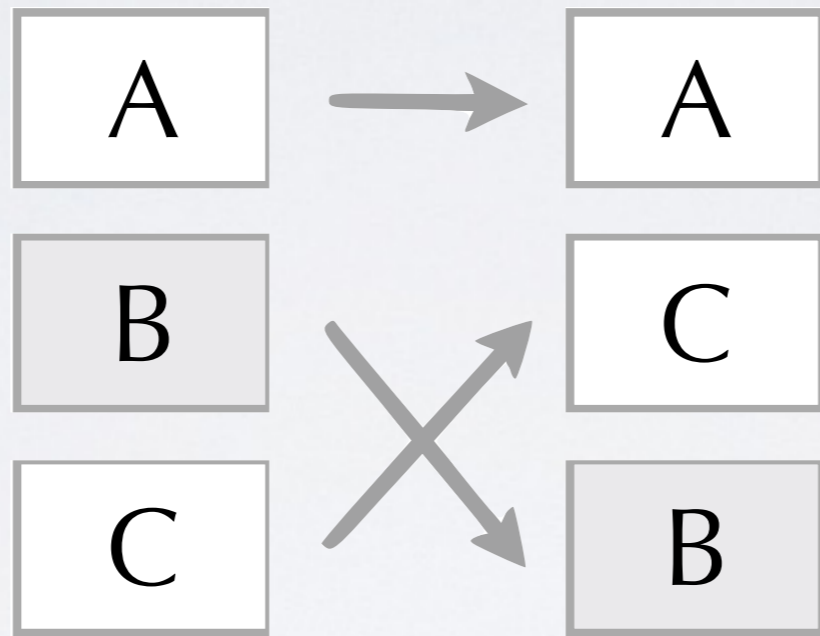
```
intersection(xs, ys)
```

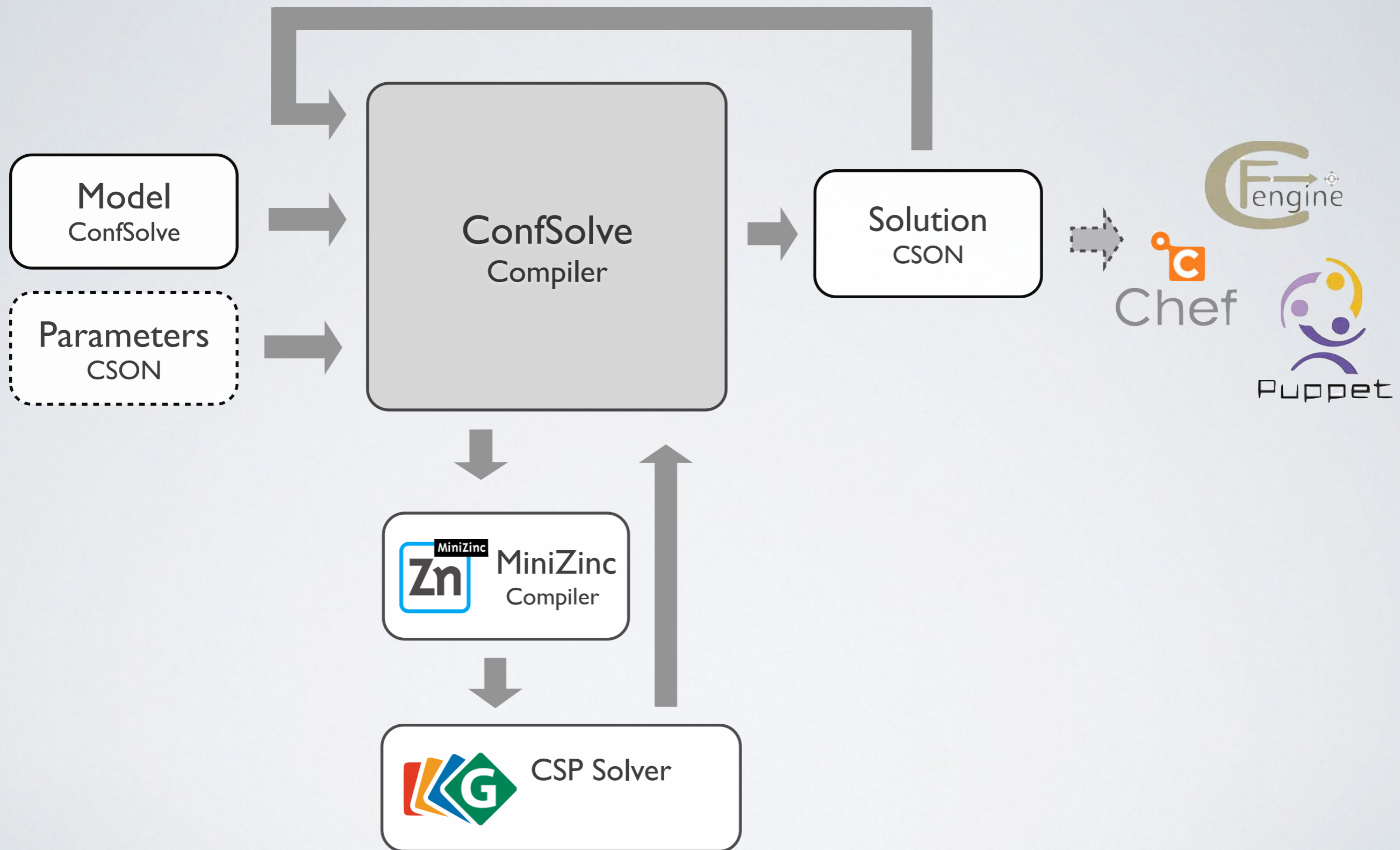






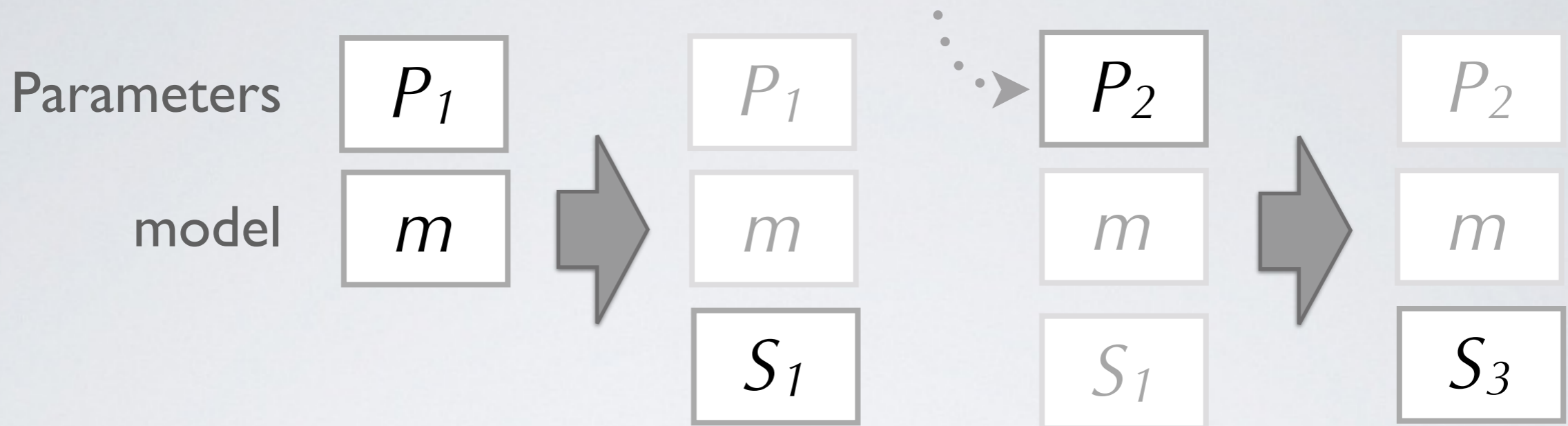
# But...



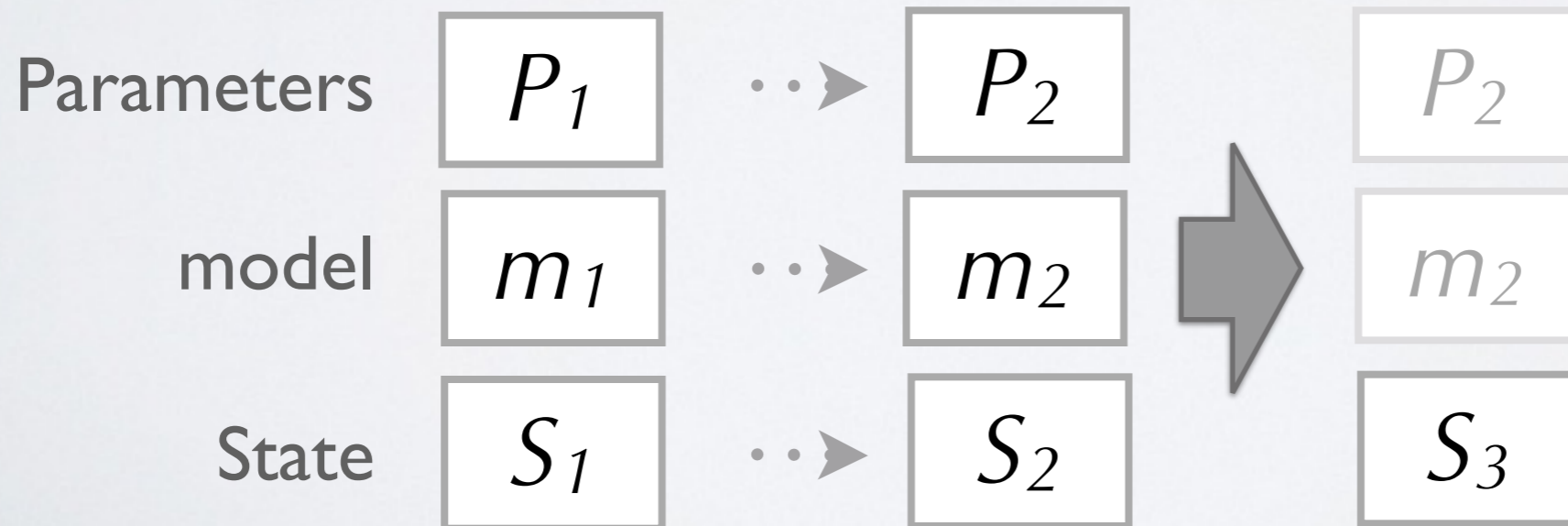




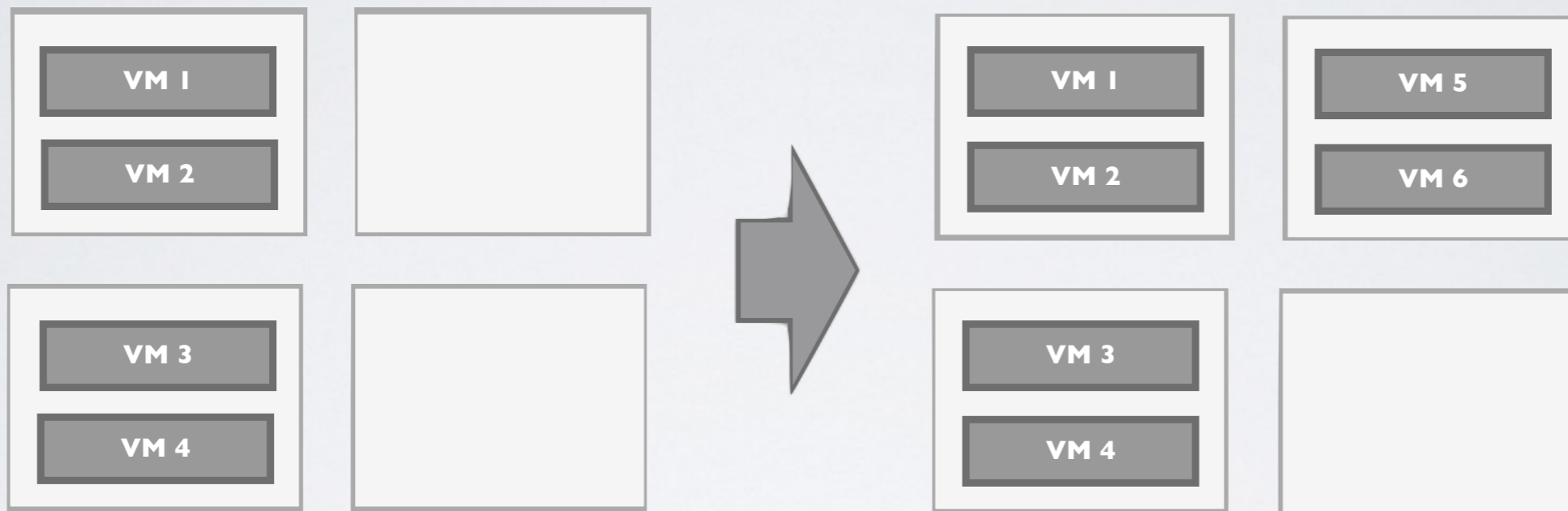
# Parameter Change



# Migration



# Migration: Adding Virtual Machines



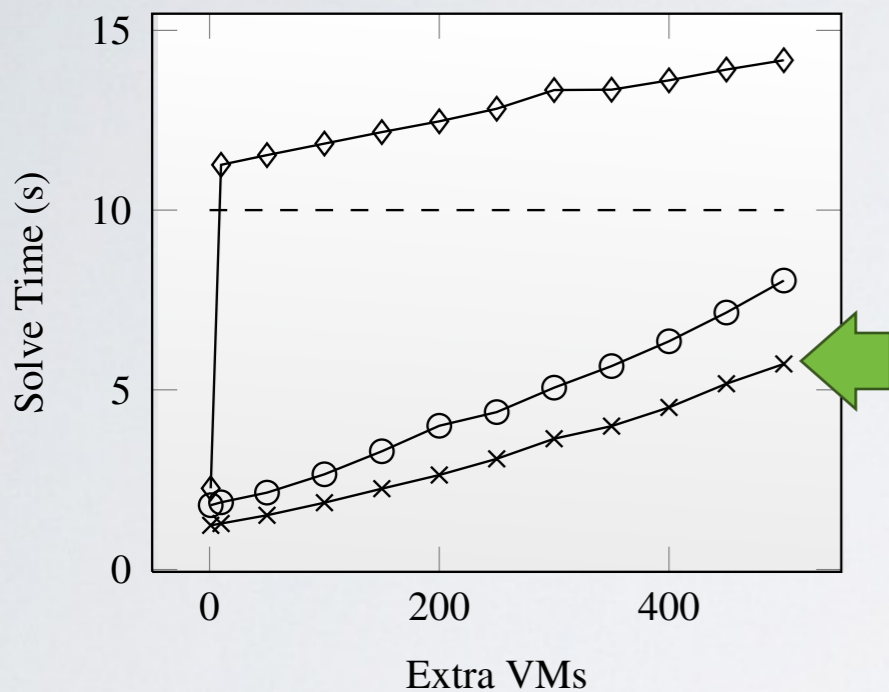
```
class Machine {  
  var cpu as int = 8;  
  var memory as int = 16384;  
  var disk as int = 2048;  
}
```

```
class VM {  
  var host as ref Machine;  
  var cpu as int = 1;  
  var memory as int = 768;  
  var disk as int = 20;  
}
```

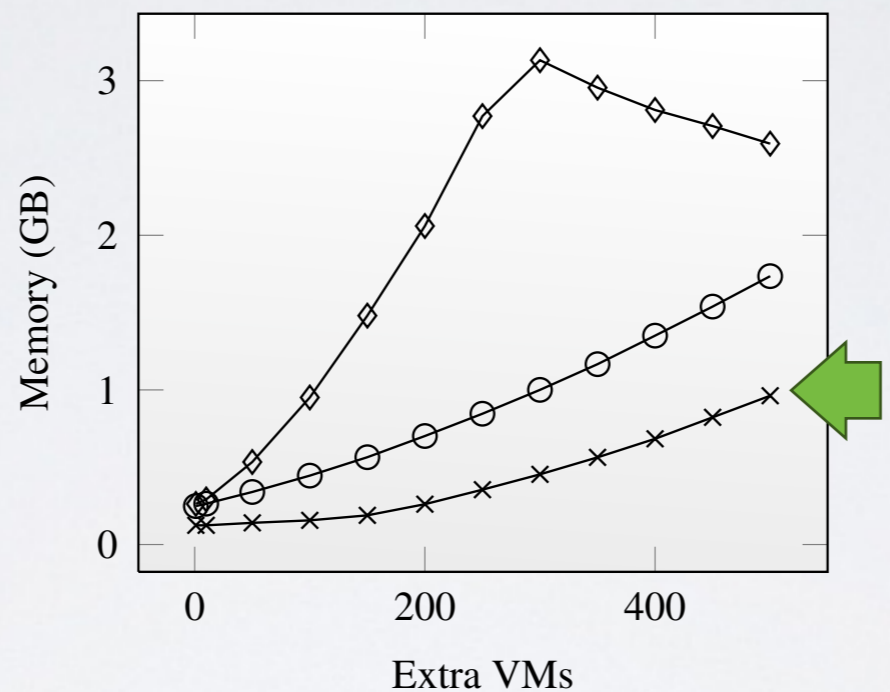
```
forall m in machines {  
  sum vm in vms where vm.host = m {  
    vm.cpu;  
  } <= m.cpu  
&& ...
```

```
change {  
  forall vm in vms { vm.host = ~vm.host; };  
}
```

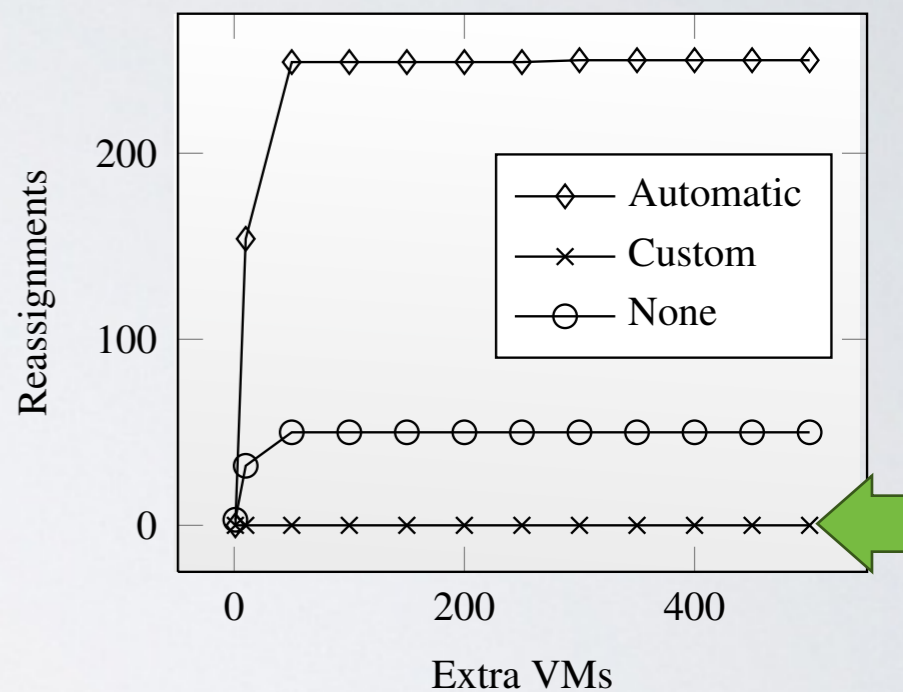
# Time



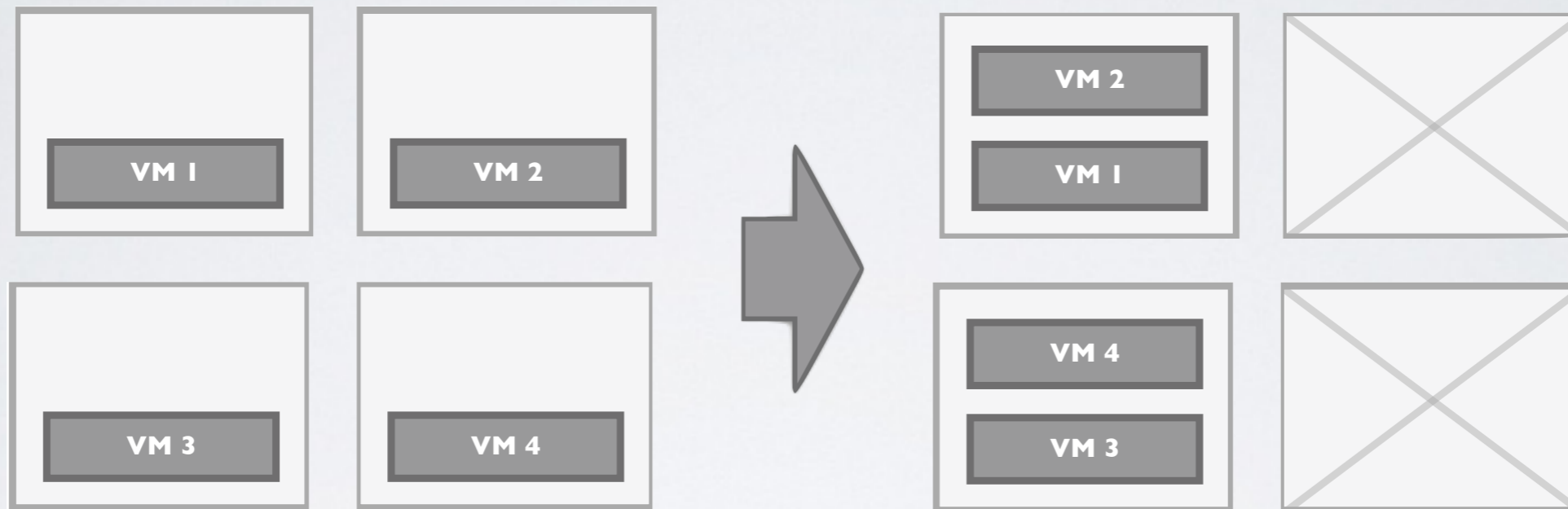
# Memory



# Reassignments



# Parameters: Virtual Server Failure





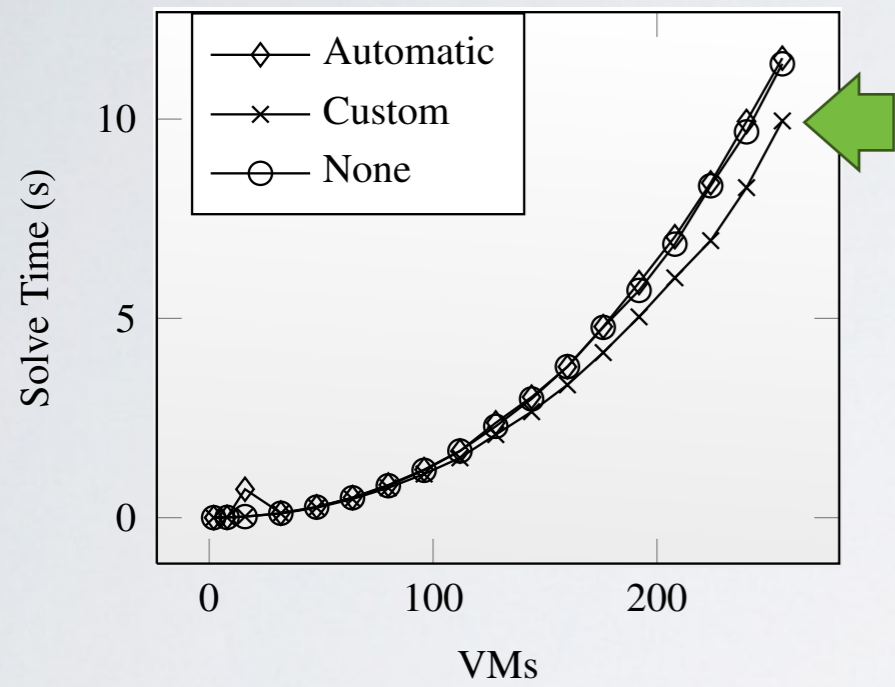
```
class Machine {
  param online as bool;
  ...
}

// VM host must be online
forall vm in vms {
  vm.host.online = true;
};

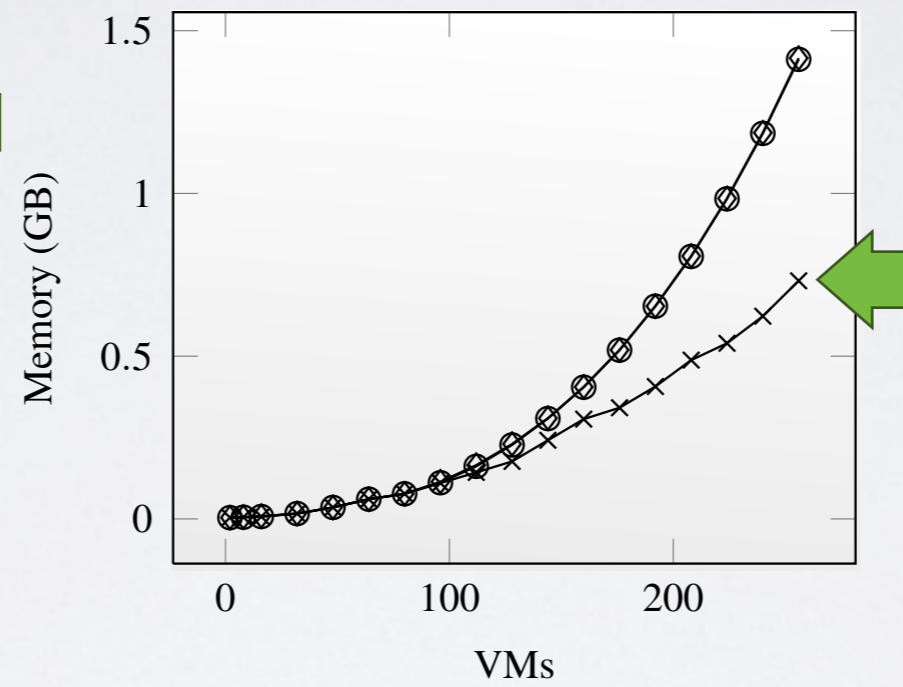
// distribute VMs across hosts
minimize sum vm1 in vms {
  count (vm2 in vms where vm1.host = vm2.host);
}
```

```
// keep VMs on online hosts
change {
  forall m in machines where m.online && ~m.online {
    forall vm in vms where ~vm.host = m {
      vm.host = ~vm.host;
    }
  }
}
```

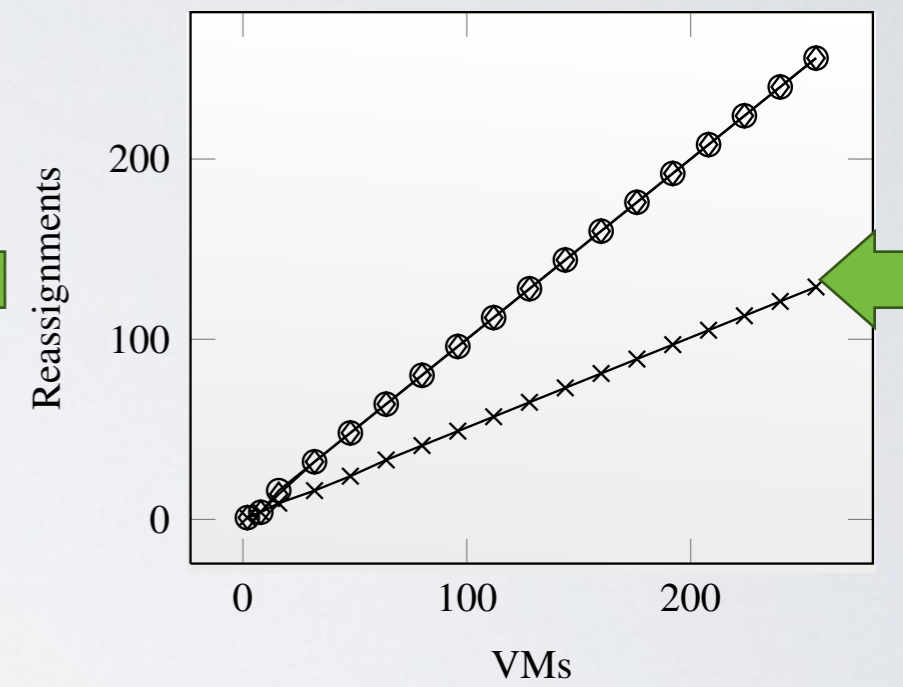
# Time



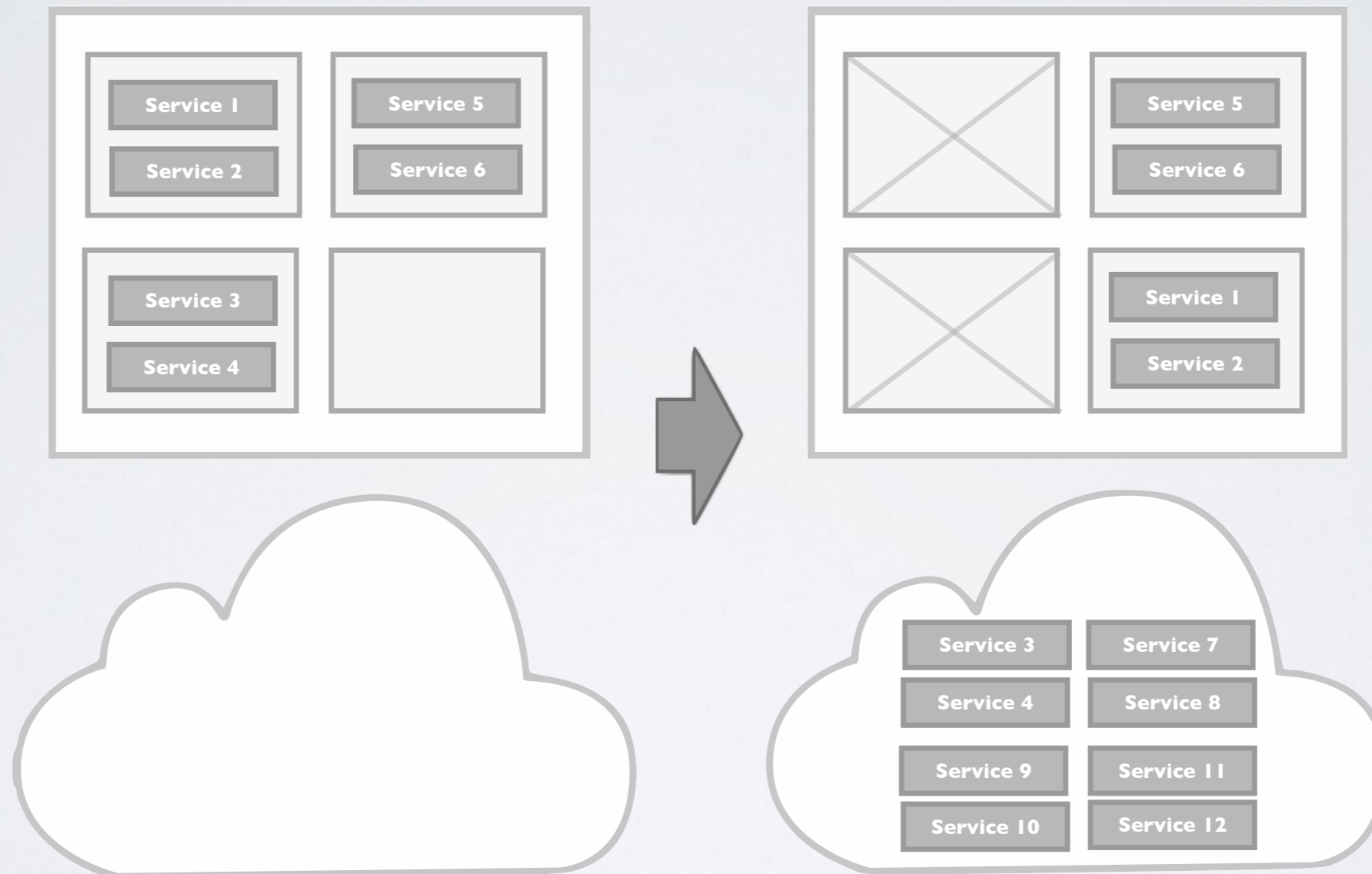
# Memory



# Reassignments



# Migration with Parameters: Cloudbursting



```
abstract class Host {}
```

```
graph TD; Machine[Machine] --> Host[Host]; Cloud[Cloud] --> Host; Web[Web] --> Service[Service]; Worker[Worker] --> Service; Database[Database] --> Service;
```

```
class Machine extends Host {  
    param online as bool;  
    var cpu as int = 4;  
    var memory as int = 4096;  
}
```

```
class Cloud extends Host {}
```

```
abstract class Service {  
    var host as ref Host;  
    var cpu as int;  
    var memory as int;  
}
```

```
class Web extends Service {  
    cpu = 2;  
    memory = 2048;  
}
```

```
class Worker extends Service {  
    cpu = 2;  
    memory = 2048;  
}
```

```
class Database extends Service {  
    cpu = 4;  
    memory = 4096;  
}
```

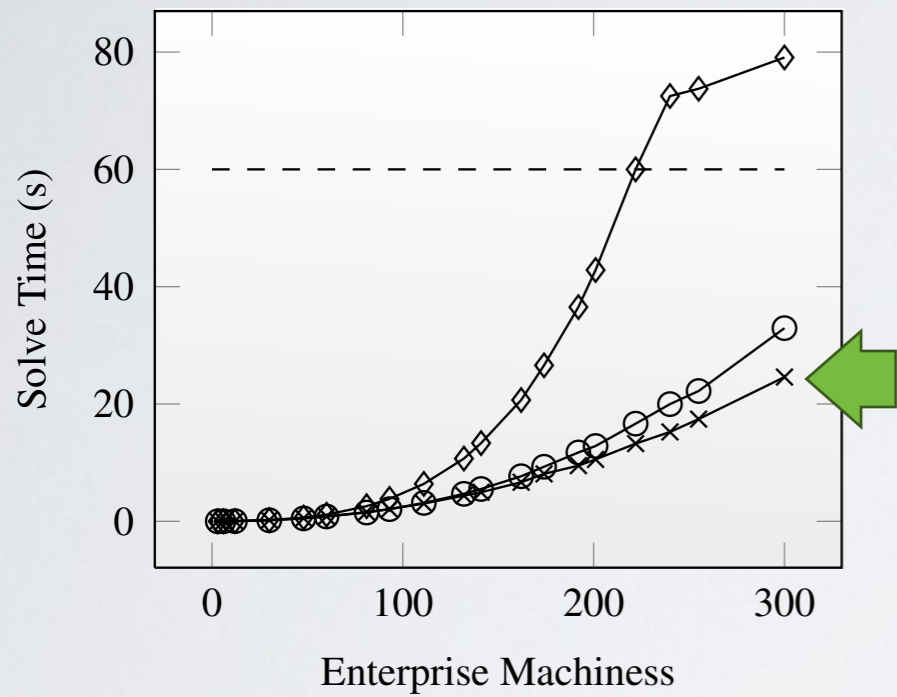
```
// all service hosts online ...  
// bin-pack in the enterprise ...
```

```
// fill the enterprise before using the cloud  
count (s in services where s.host = cloud) > 0 ->  
  count (s in services where s.host in enterprise) =  
    count (m in enterprise where m.online);
```

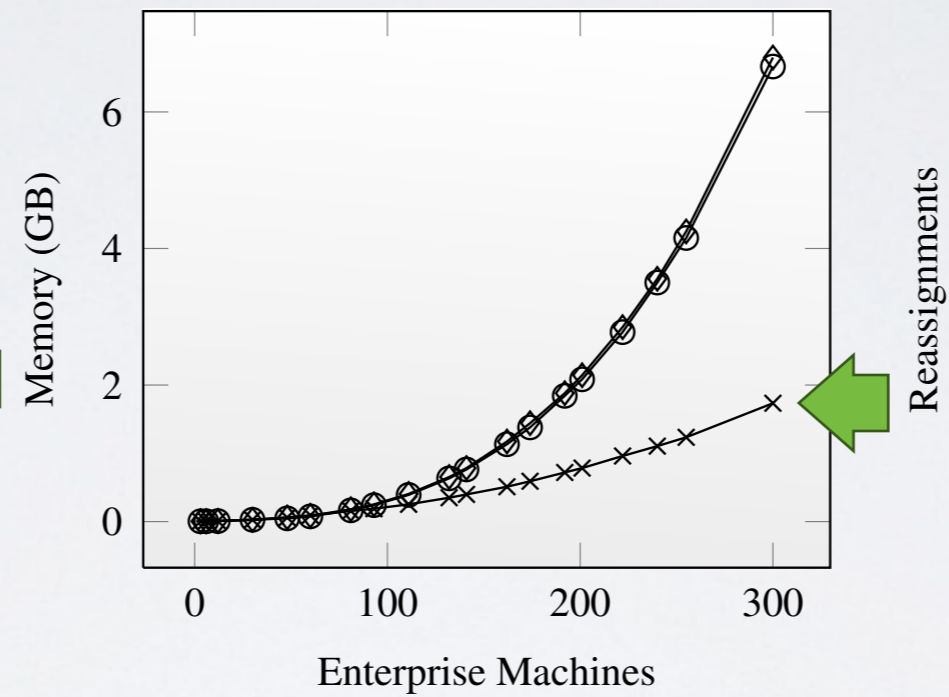
```
// keep VMs on online hosts  
change {  
  forall m in enterprise where m.online && ~m.online {  
    forall s in services where ~s.host = m {  
      vm.host = ~s.host;  
    };  
  };  
}
```



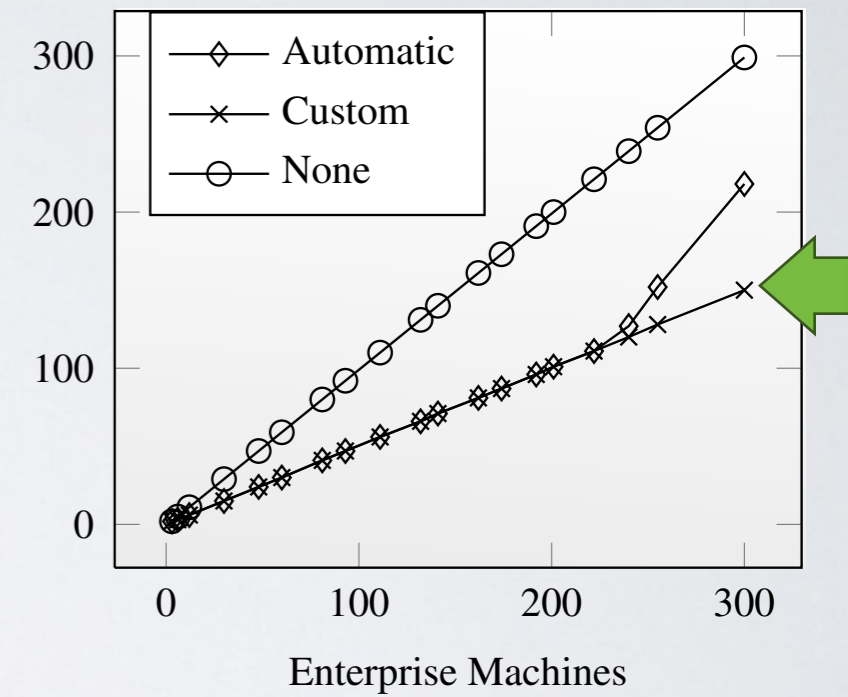
# Time



# Memory



# Reassignments





[john.hewson@ed.ac.uk](mailto:john.hewson@ed.ac.uk)



THE UNIVERSITY of EDINBURGH  
**informatics**