

# **Dynamic Isolation for Protection in the Movable Feast Machine**

**James Vickers**

**Department of Computer Science**

**University of New Mexico**

# The current software paradigm

Correctness and Efficiency Only (CEO):

- Software assumes reliable delivery platform (i.e. hardware)
- Born out of limited hardware resources
- Less important when hardware is more powerful than most applications need
- More risky as software gains responsibilities

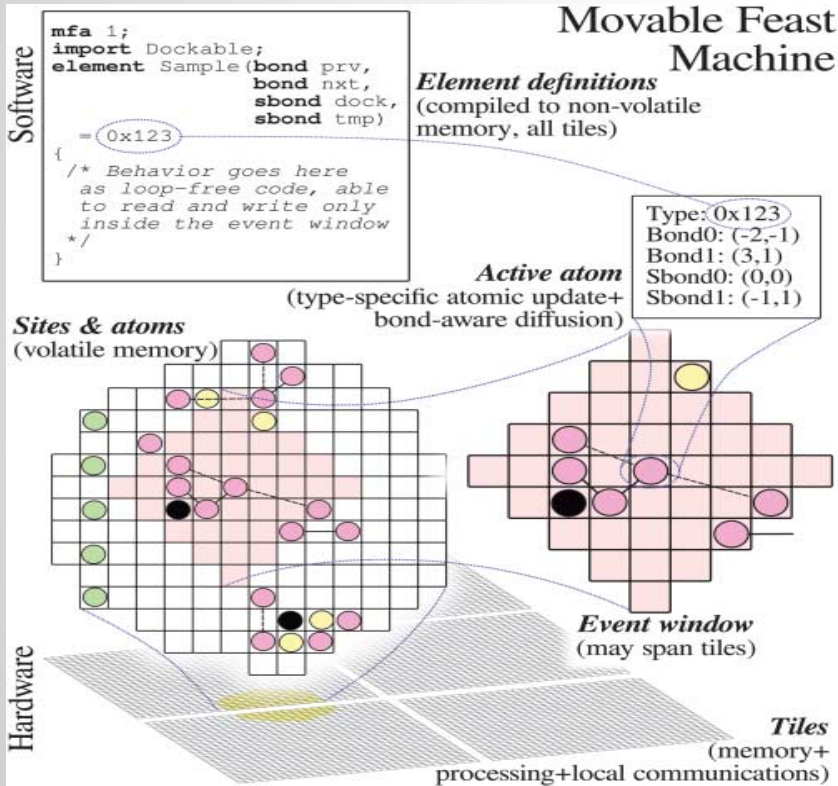
# Robust-first, scalable computing

- Robustness principles:
  - Redundancy versus efficiency
  - Local state instead of global state
  - Local interactions form into large-scale operations
  - Avoid state as much as possible - corruptible
- Scalability principles:
  - Systems made of small components
  - Local state instead of global state
  - De-centralized control flow

# The Movable Feast Machine (MFM)

- Indefinitely scalable, robust-first computing hardware architecture
- Homogenous hardware tiles with fixed-size volatile memory locations called *Sites*
- Programs called *Elements* read and write instances of *Atoms* to *Sites*
- Atom is an instance of an Element at a Site

# The Movable Feast Machine (MFM)



- Atoms called to act by MFM - *Event*
- Atom has access to Moore Neighborhood of Sites called *Event Window*
- Event Window has radius *Event Radius*

# Atom safety in the MFM

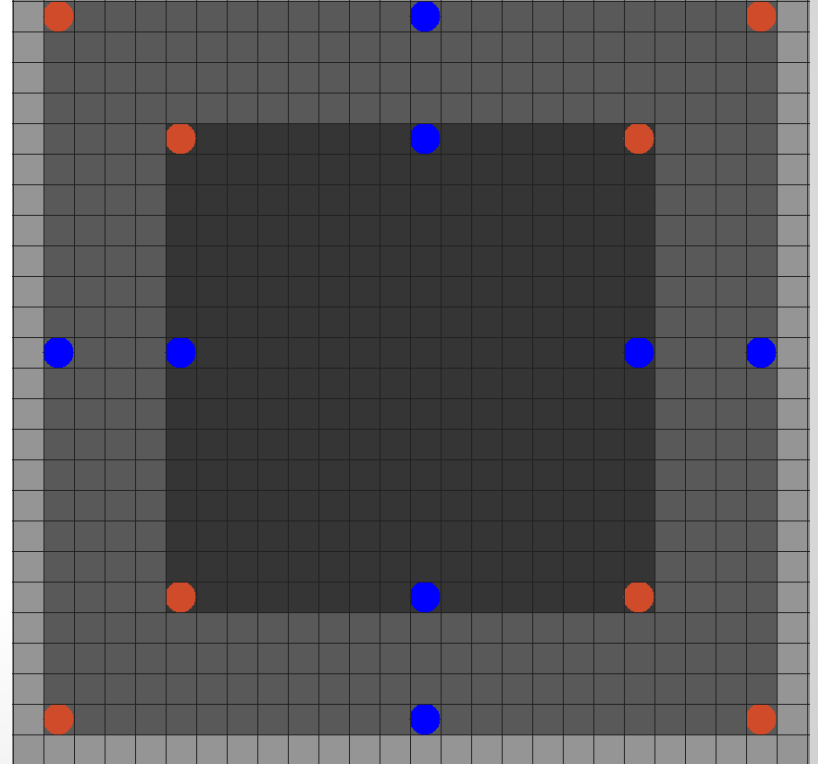
- During an Event, an Atom can read and write any Site in its Event Window
  - Thus, it can destroy/corrupt anything visible!
- Elements can be designed to tolerate this, but what if we want protection for Atoms?

# The Mindless Brute: *Eraser*

- Erases Atoms blindly, regardless of type
- Affected Sites are within configurable *Eraser Distance* (ED) in  $\{1,2,3,4\}$  of Eraser Atoms
- ED = 1: wipes directly adjacent Sites
- ED = 4: wipes entire Event Window (except itself)

# The *Data* Element

- *Data*: diffusing storage Element in MFM; doesn't reproduce or decay
- What happens to Data Atoms in the presence of Eraser's?





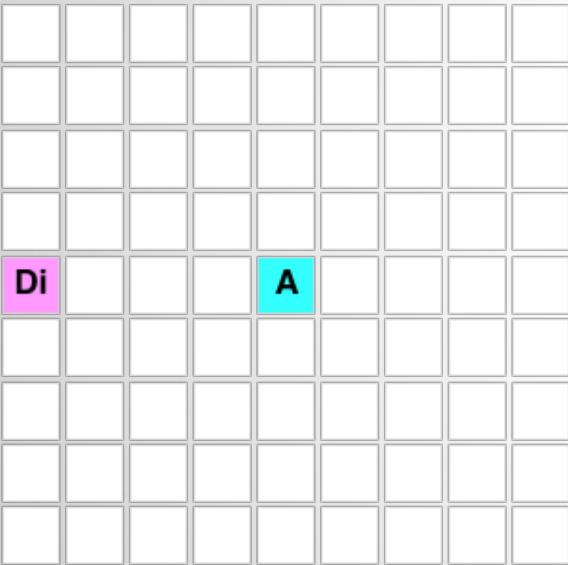


# Dynamic Isolator

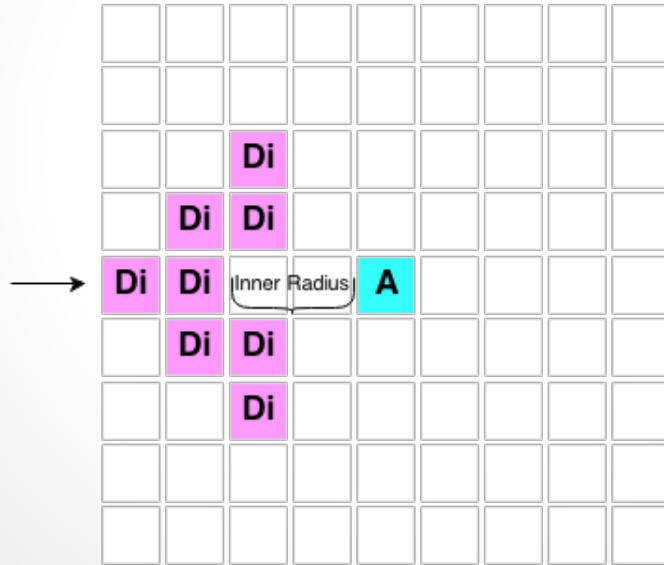
- Design goal: protect Atoms from each other by spatial separation
  - Wraps Atoms in *Isolator Bubble*
  - Keeps Isolator Bubbles from merging
- Difficulty: MFM is asynchronous, Atoms have full read-write access on Event Window

# Goal: Surround Atom with Dynamic Isolator's

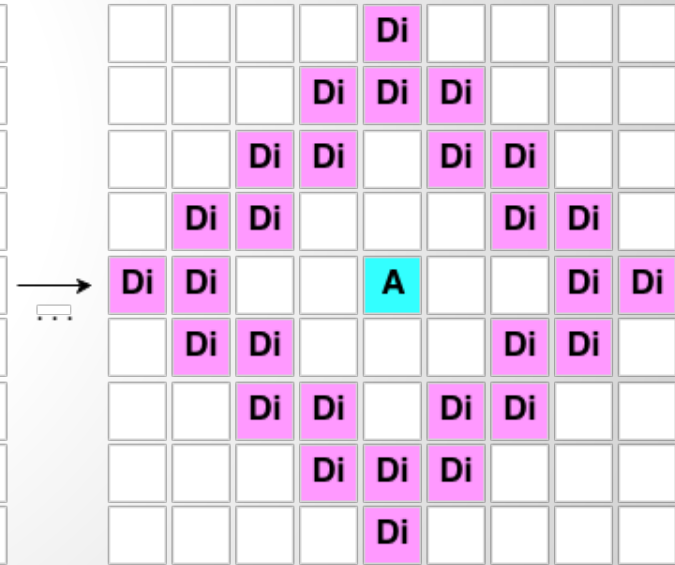
## Isolator Bubble Construction



Dynamic Isolator sees Atom



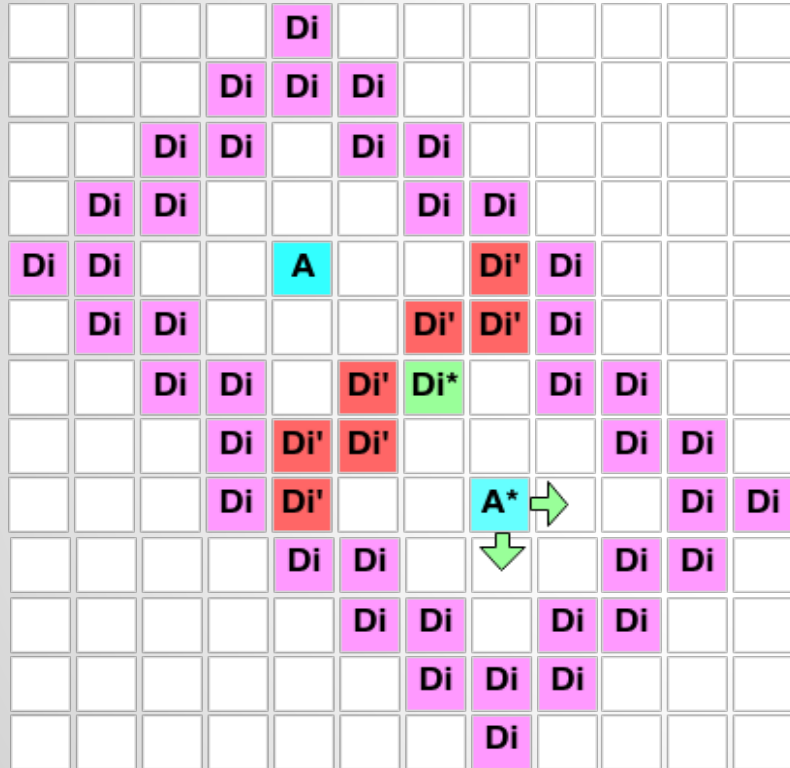
Isolator's written at qualifying Sites around A



Isolator Bubble adapts as A moves

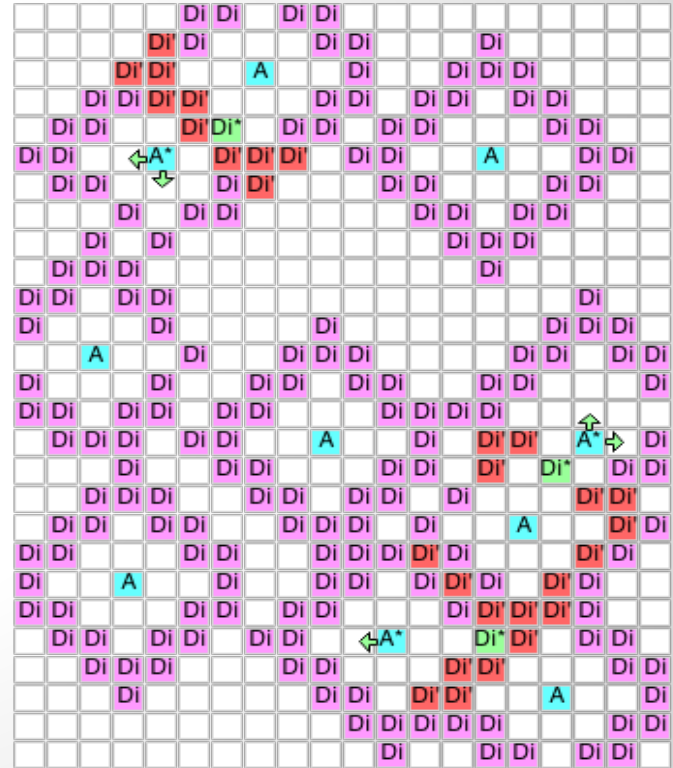
# Goal: Keep Isolator Bubble's separated

## Isolator Bubble Repulsion



Di\* will swap A\* away away randomly in one direction

## Isolated Atoms



Bubble's are imperfect briefly after A moves

# Swapping atom away

Basically: increase distance by one unit from the acting Dynamic Isolator.

- If aligned in X or Y: move in aligned direction
- Otherwise, randomly move away in X or Y

# Properties of Dynamic Isolator

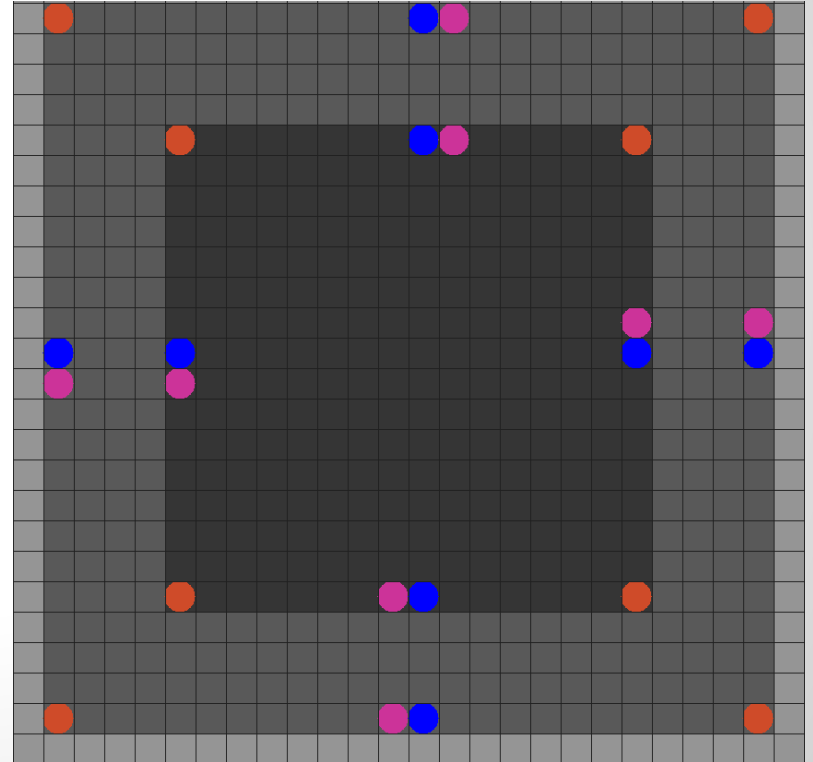
- Stateless: does not use any mutable state given to its instance Atoms.
- Type-agnostic: isolates any Atom that isn't of its own type or Empty.
- Limited-reproducing: only makes copies of itself around other Atoms
- Dies in total isolation: erases itself if it only sees own type and Empty during an Event

# Dynamic Isolator with stationary and diffusing Atoms

Demo

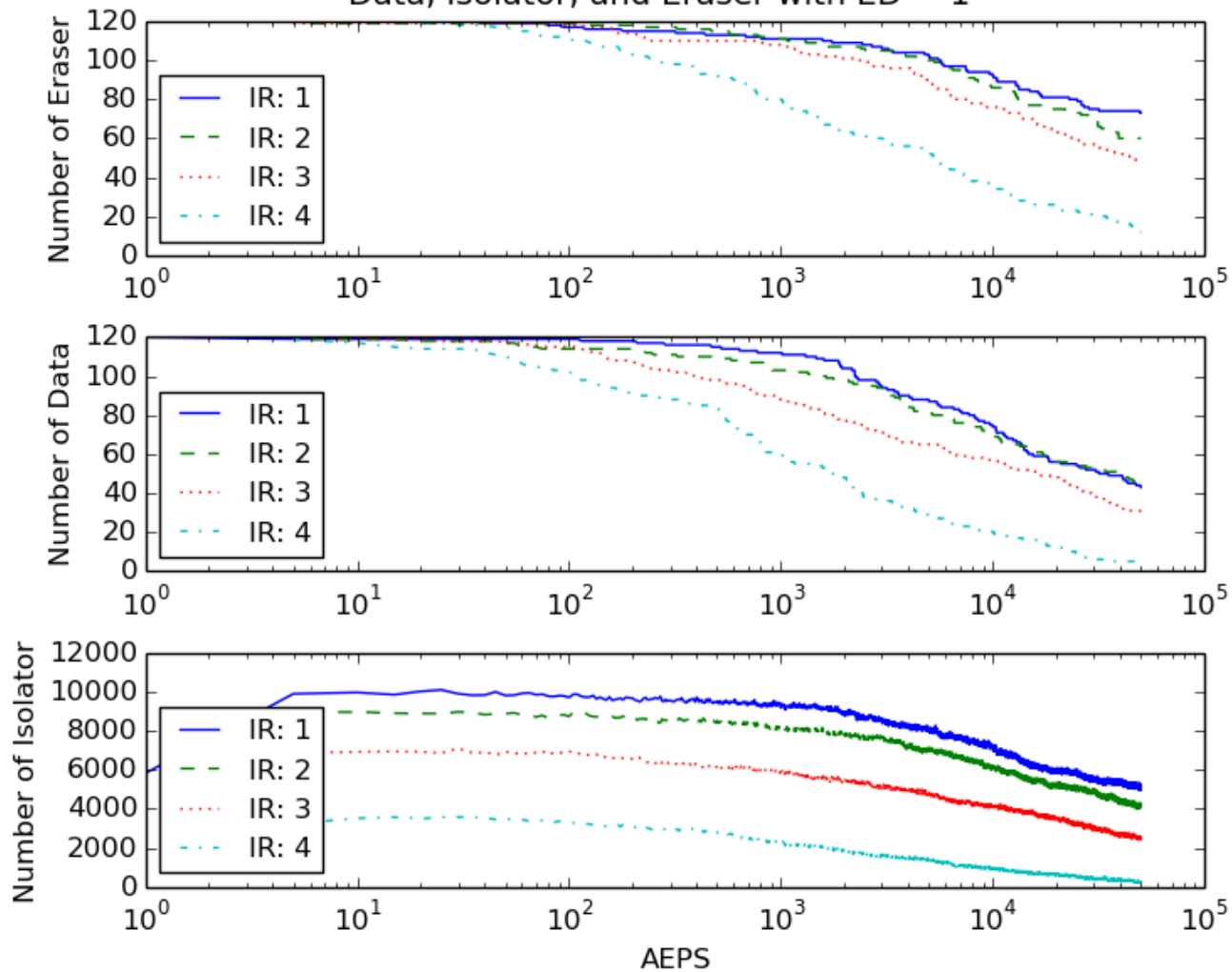
# Back to the Data-pocalypse

- Can Dynamic Isolator protect Data from Eraser?
- How do the values of Eraser Distance and Inner Radius affect population?

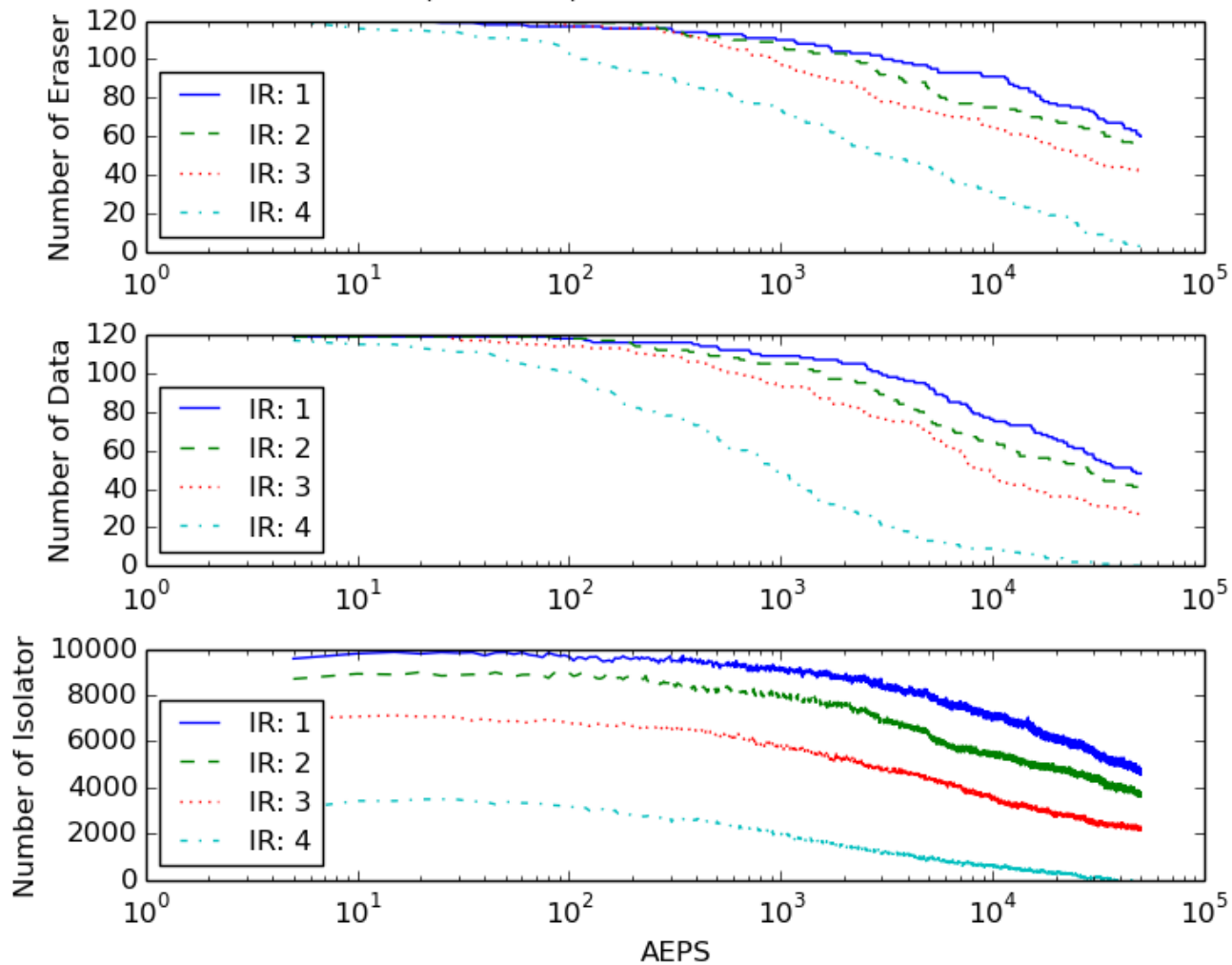




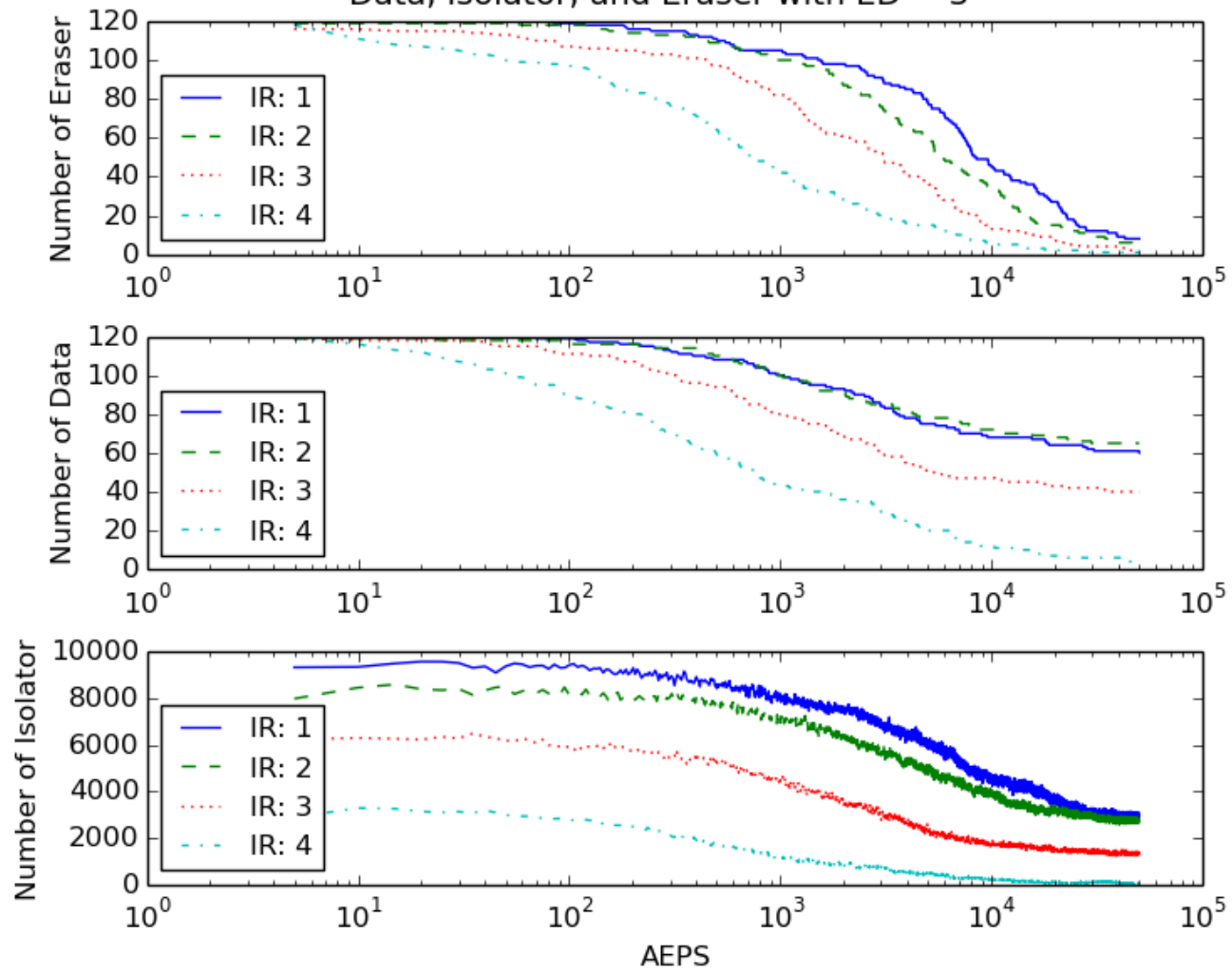
Data, Isolator, and Eraser with ED = 1



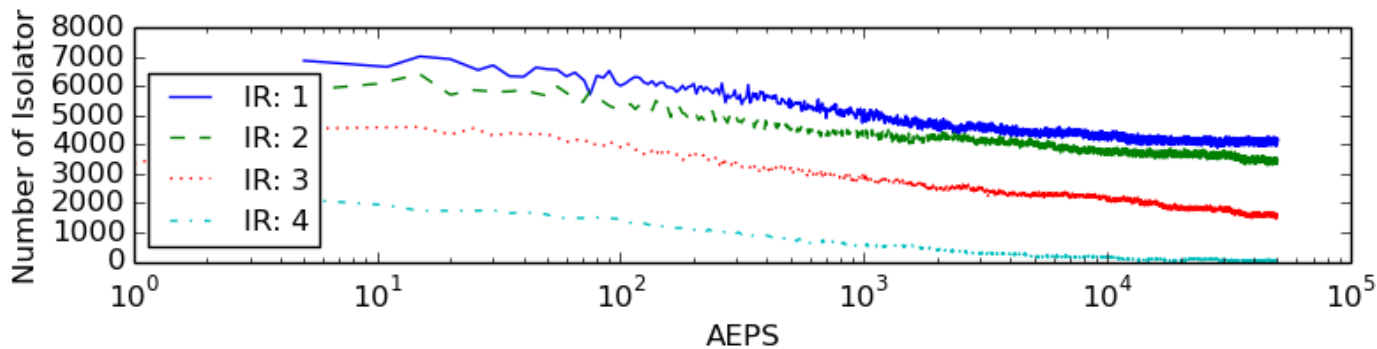
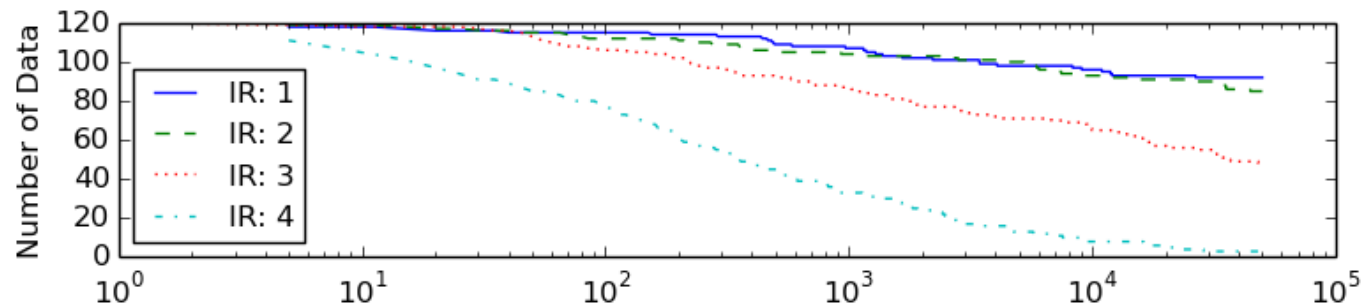
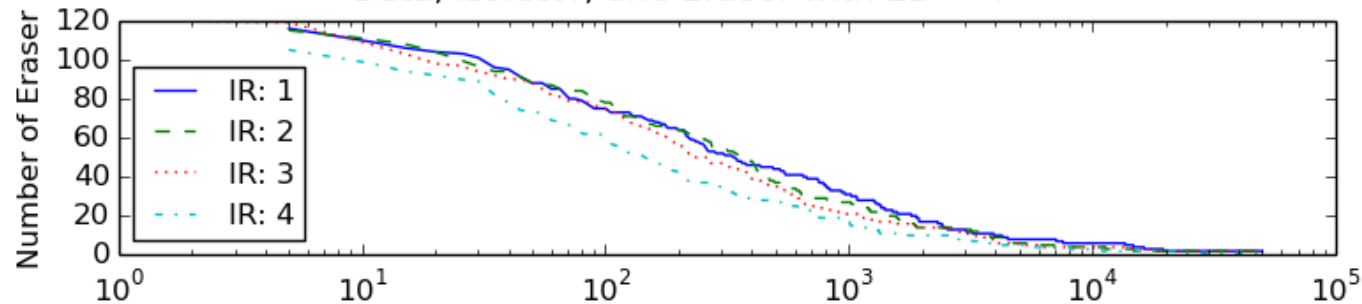
Data, Isolator, and Eraser with ED = 2



### Data, Isolator, and Eraser with ED = 3



Data, Isolator, and Eraser with ED = 4



Define *half-life* as the amount of AEPS for half of the population of an Element to be destroyed:

ED	No Isolator		IR = 1		IR = 2		IR = 3		IR = 4	
	Data	Eraser	Data	Eraser	Data	Eraser	Data	Eraser	Data	Eraser
1	90	245	14735	(F:73)	17155	39245	6705	23470	905	2575
2	60	145	26785	49235	12130	33735	6910	13810	635	1810
3	46	120	49825	7340	(F:65)	5010	2970	2025	495	545
4	35	105	(F:92)	220	(F:85)	245	16150	180	205	102

# Eraser, Data, Dynamic Isolator

## Demo

# The *Dreg* and *Res* Elements

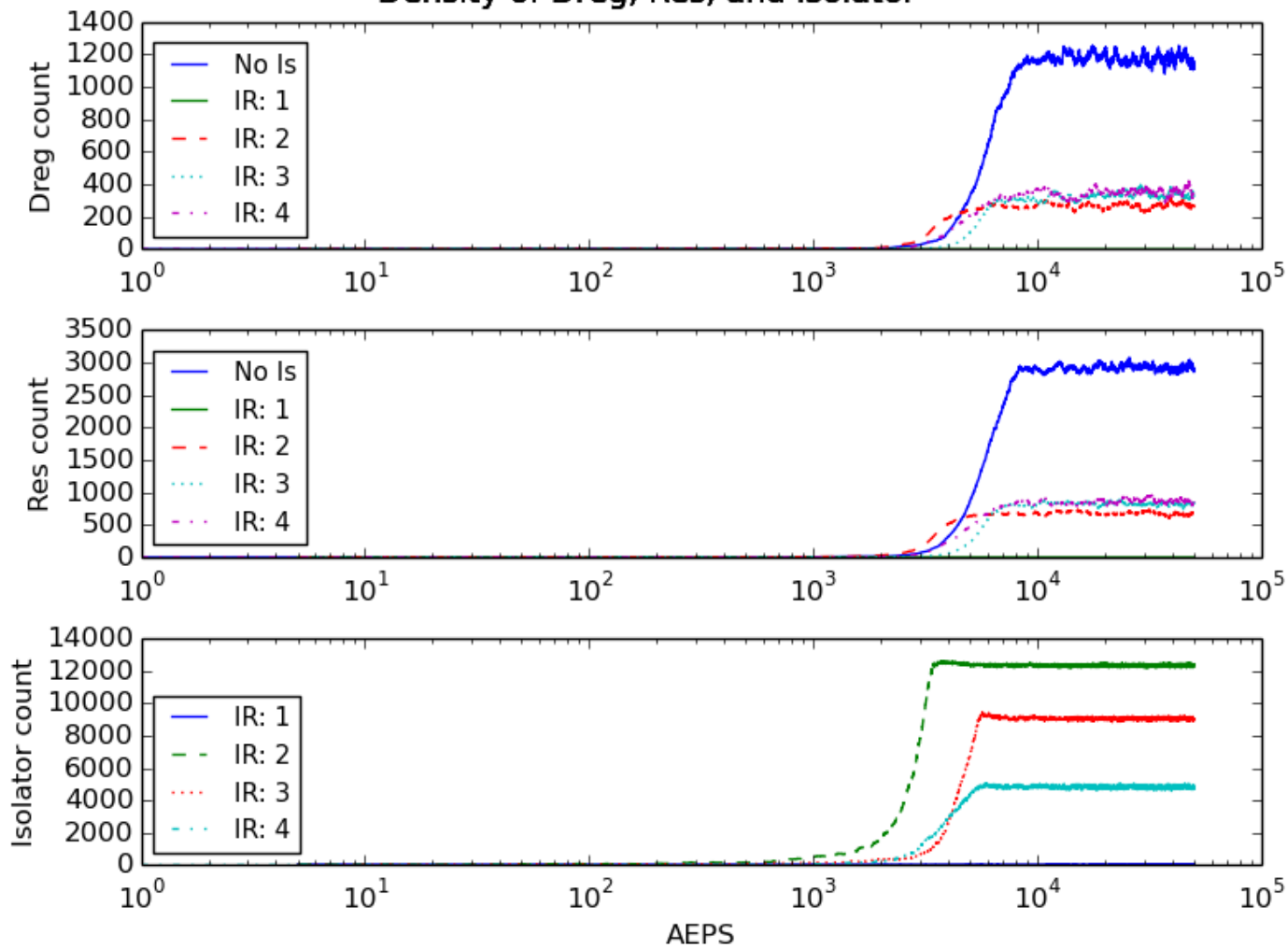
- Resource (*Res*) holds no state, diffuses, does not reproduce
- Dynamic Regulator (*Dreg*) fills space in MFM with *Res* and itself
- During Event, *Dreg* look at adjacent sites:
  - Delete Atom, delete other *Dreg*, create *Res*, create *Res* - four separate probabilities

# Experiment

- Dreg tries to fill space in MFM, but Dynamic Isolator wants to separate all Atoms
- What happens when these two are placed in an environment together?



### Density of Dreg, Res, and Isolator



# Future work ideas

- *Dynamic Isolator with claws*: could try to detect destructive Elements, destroy them
- *Dynamic Aggregator*: surround and separate Atoms of different types, but push together Atoms of the same type
  - Spatial sorting by Element type

# Conclusion

- The Movable Feast Machine is an indefinitely scalable computer architecture
- ...lacks any means of protecting Atoms in it
- Dynamic Isolator is presented as a means of protecting Atoms and as a general-purpose spatial separation method