AWS re:Invent

NETFLIX

PFC306

Performance Tuning EC2 Instances

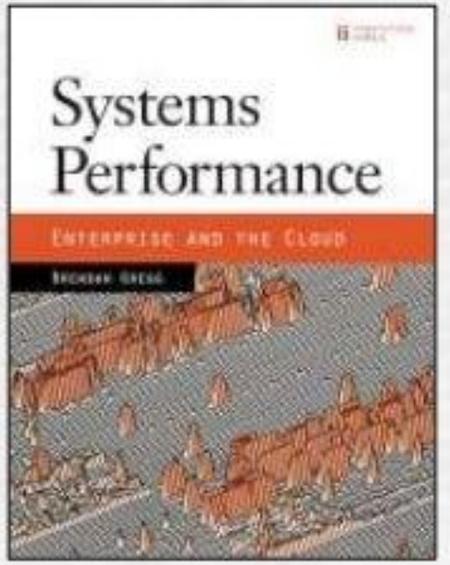
Brendan Gregg, Performance Engineering, Netflix

November 12, 2014 | Las Vegas, NV

amazon web services

Brendan Gregg

- Senior Performance Architect, Netflix
  - Linux and FreeBSD performance
  - On the Performance Engineering Team, led by Coburn Watson (and we're hiring!)
- Recent work:
  - Linux perf-tools, using ftrace & perf_events
- Previous work includes:
  - USE Method, flame graphs, heat maps, DTrace tools
  - Sysadmin, training, kernel engineering, performance



NETFLIX

- Massive Amazon EC2 Linux cloud
  - Tens of thousands of server instances
  - Auto scale by ~3k each day
  - CentOS and Ubuntu
- FreeBSD for content delivery
  - Approx. 33% of US Internet traffic at night
- Performance is critical
  - Customer satisfaction: now over 50M subscribers
  - $$$ price/performance
  - Develop tools for cloud-wide and instance analysis

Netflix Performance Engineering Team

- Evaluate technology
  - Instance types, Amazon EC2 options
- Recommendations & best practices
  - Instance kernel tuning, assist app tuning
- Develop performance tools
  - Develop tools for observability and analysis
- Project support
  - New database, programming language, software change
- Incident response
  - Performance issues, scalability issues

Agenda

1. Instance Selection
2. Amazon EC2 Features
3. Kernel Tuning
4. Observability

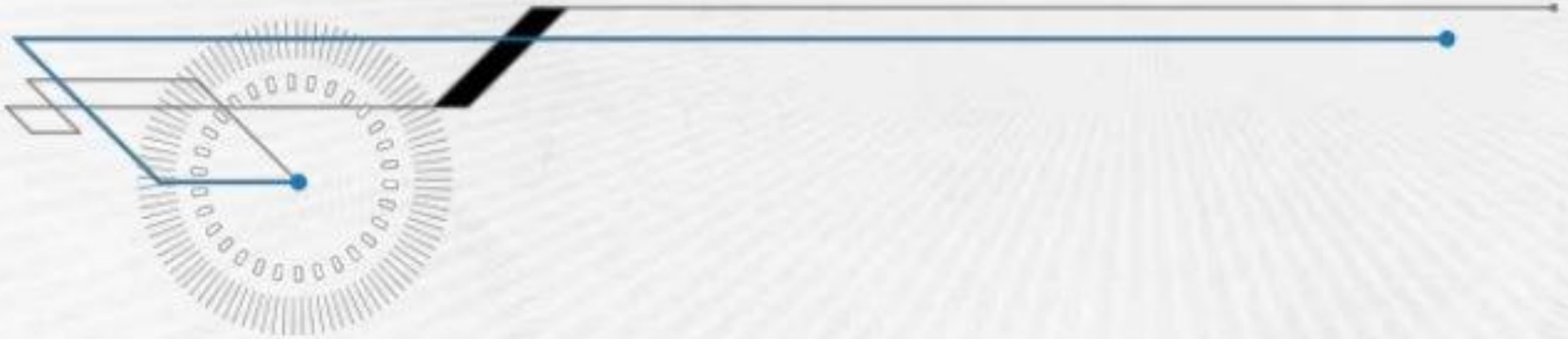
Performance Tuning on Amazon EC2

- In the Netflix cloud, everything is a tunable
  - Including instance type
- Performance wins have immediate benefits
  - Great place to do performance engineering!

WARNINGS

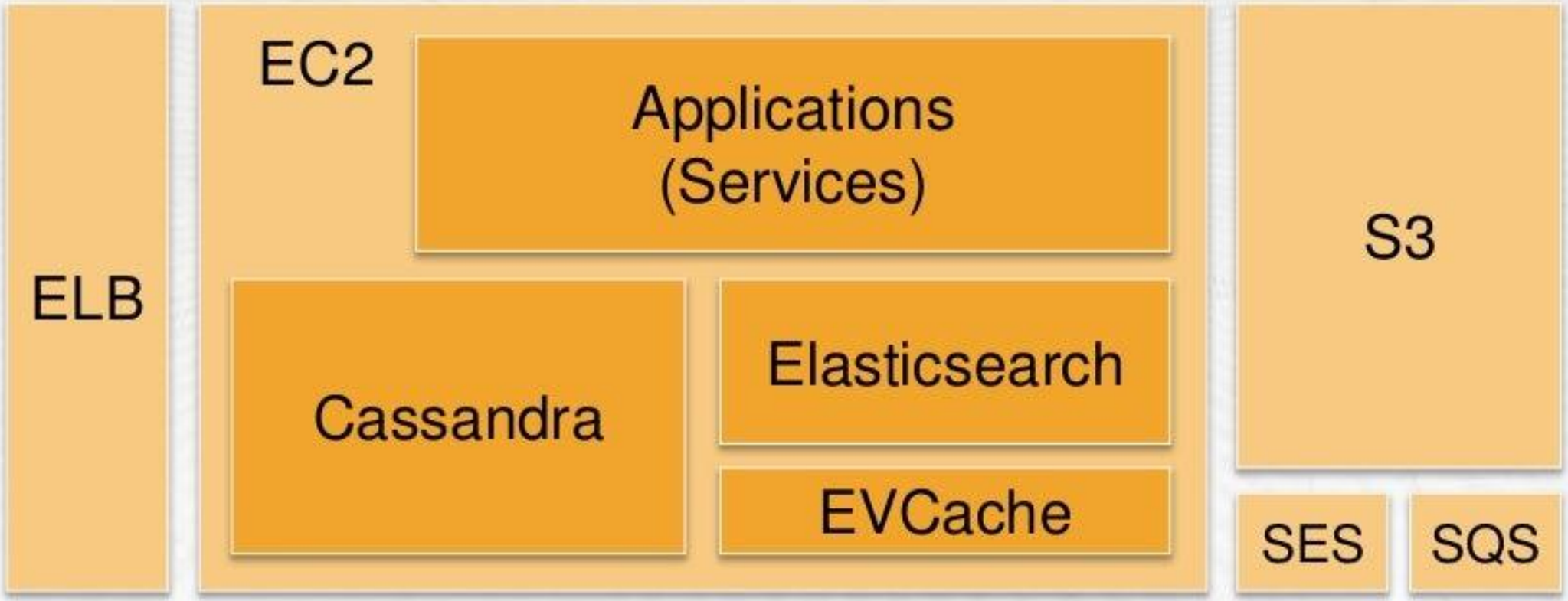
- This is what’s in our medicine cabinet
- Consider these “best before: 2015”
- Take only if prescribed by a performance engineer

1. Instance Selection



The Netflix Cloud

- Many different application workloads: compute, storage, caching…



Current Generation Instance Families

- i2: Storage-optimized
  - SSD large capacity storage
- r3: Memory optimized
  - Lowest cost/Gbyte
- c3: Compute-optimized
  - Latest CPUs, lowest price/compute perf
- m3: General purpose
  - Balanced
- Plus some others

i2.8xlarge

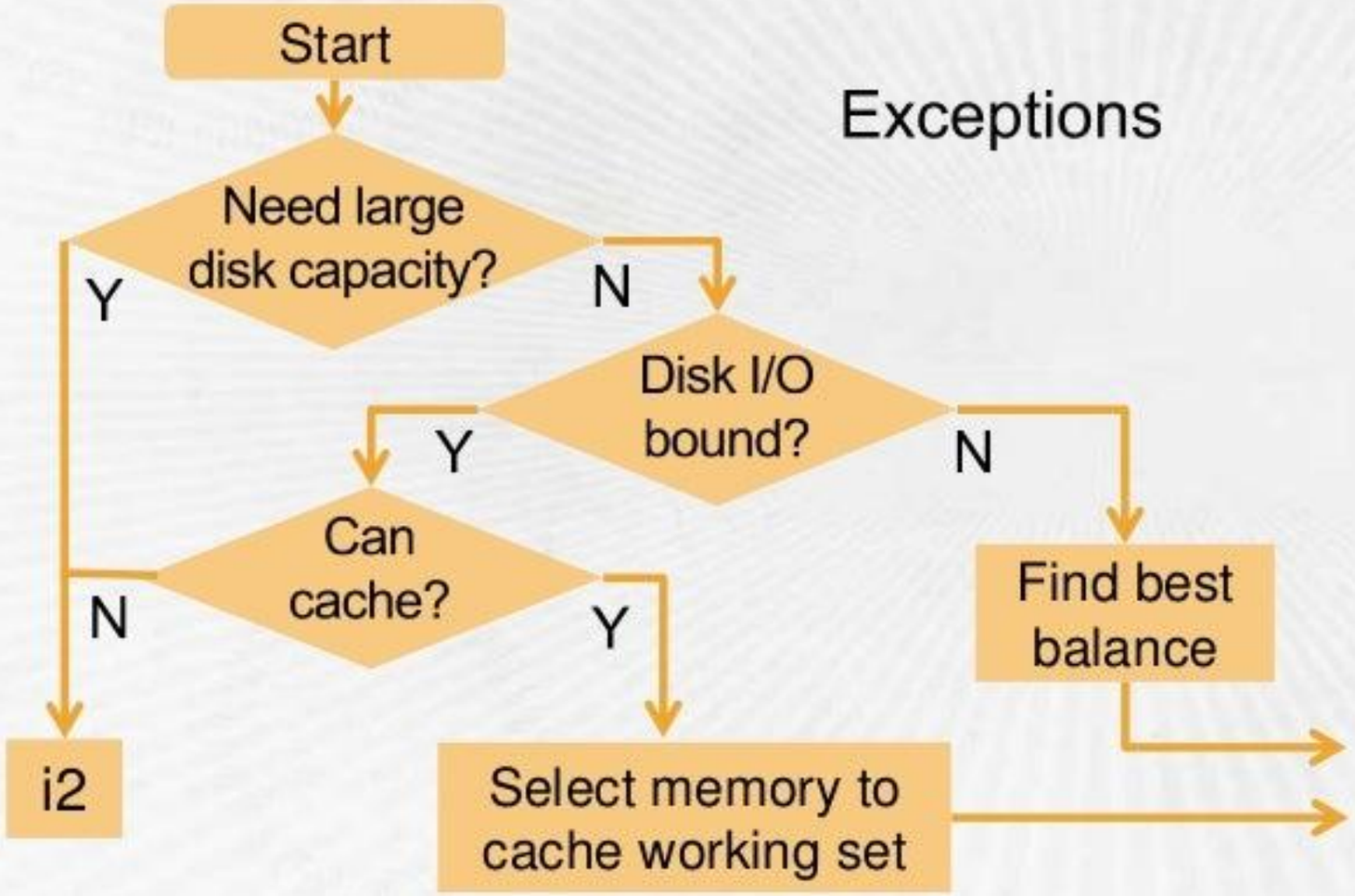
Instance Sizes

- Ranges from medium to 8xlarge, depending on type
- Netflix has over 30 instance types in use
- Traditional:
  - Tune the workload to match the server
- Cloud:
  - Find an ideal workload and instance type *combination*
    - Instead of: given A, optimize B; this is optimize A+B
  - Greater flexibility, best price/performance

Netflix Instance Type Selection

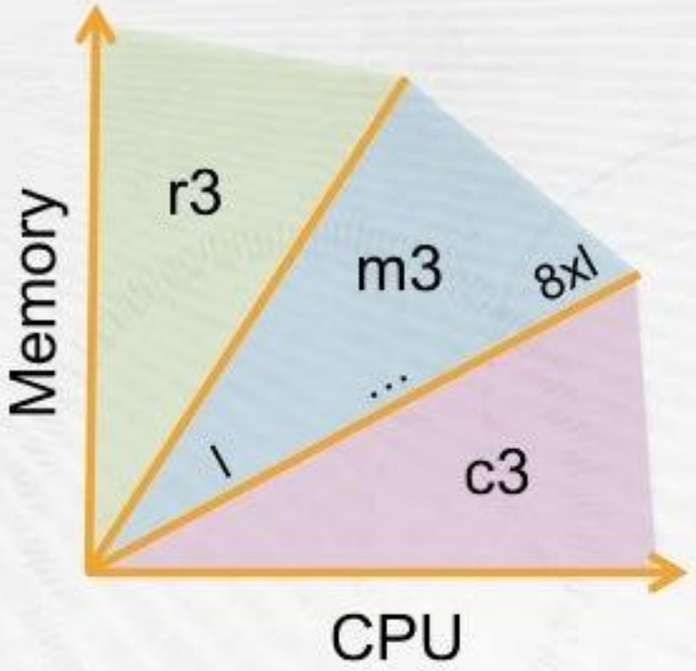
- Flow Chart
- By-Resource
- Brute Force

Instance Selection Flow Chart



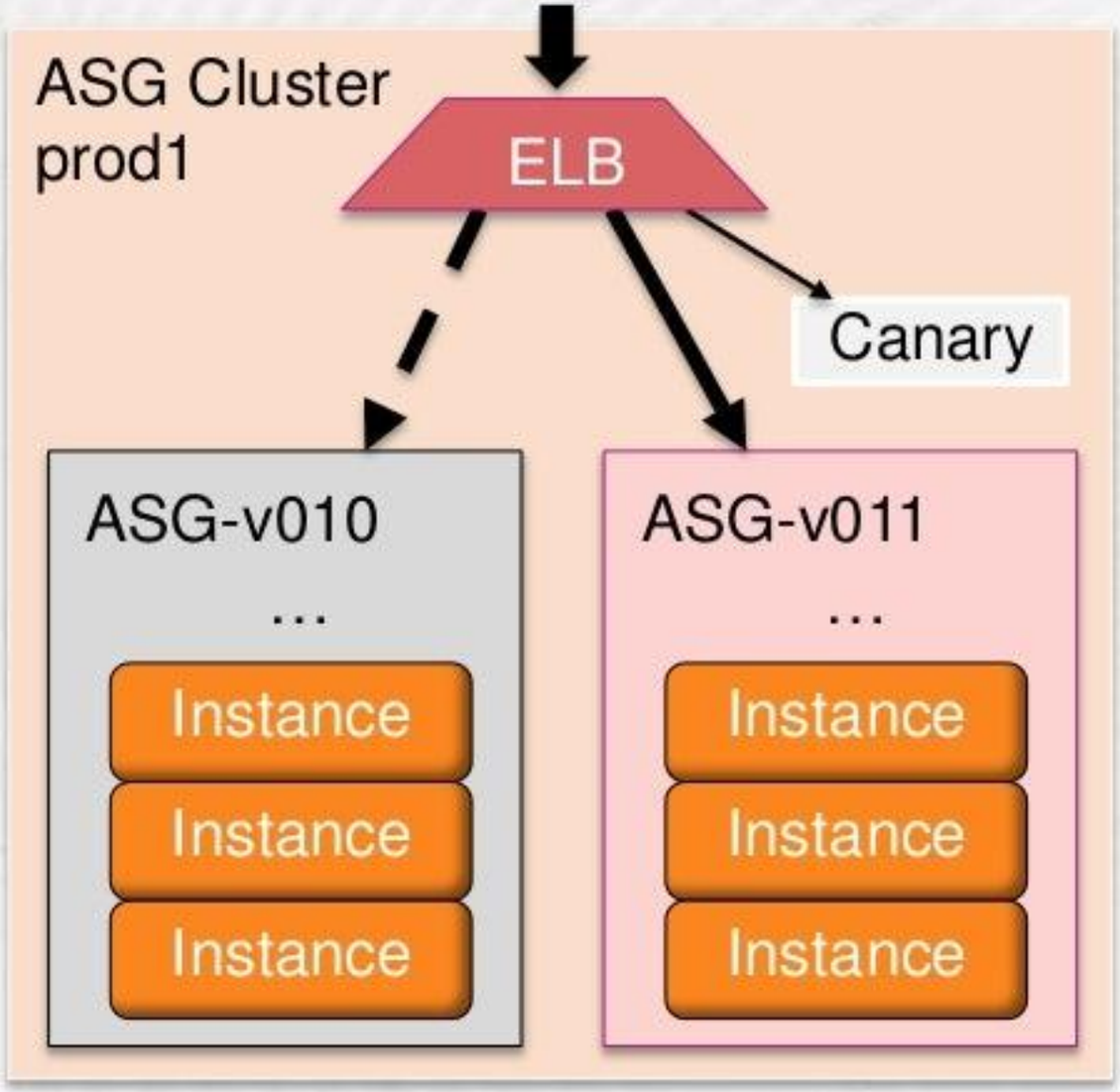
Exceptions

Trade-off



Netflix AWS Environment

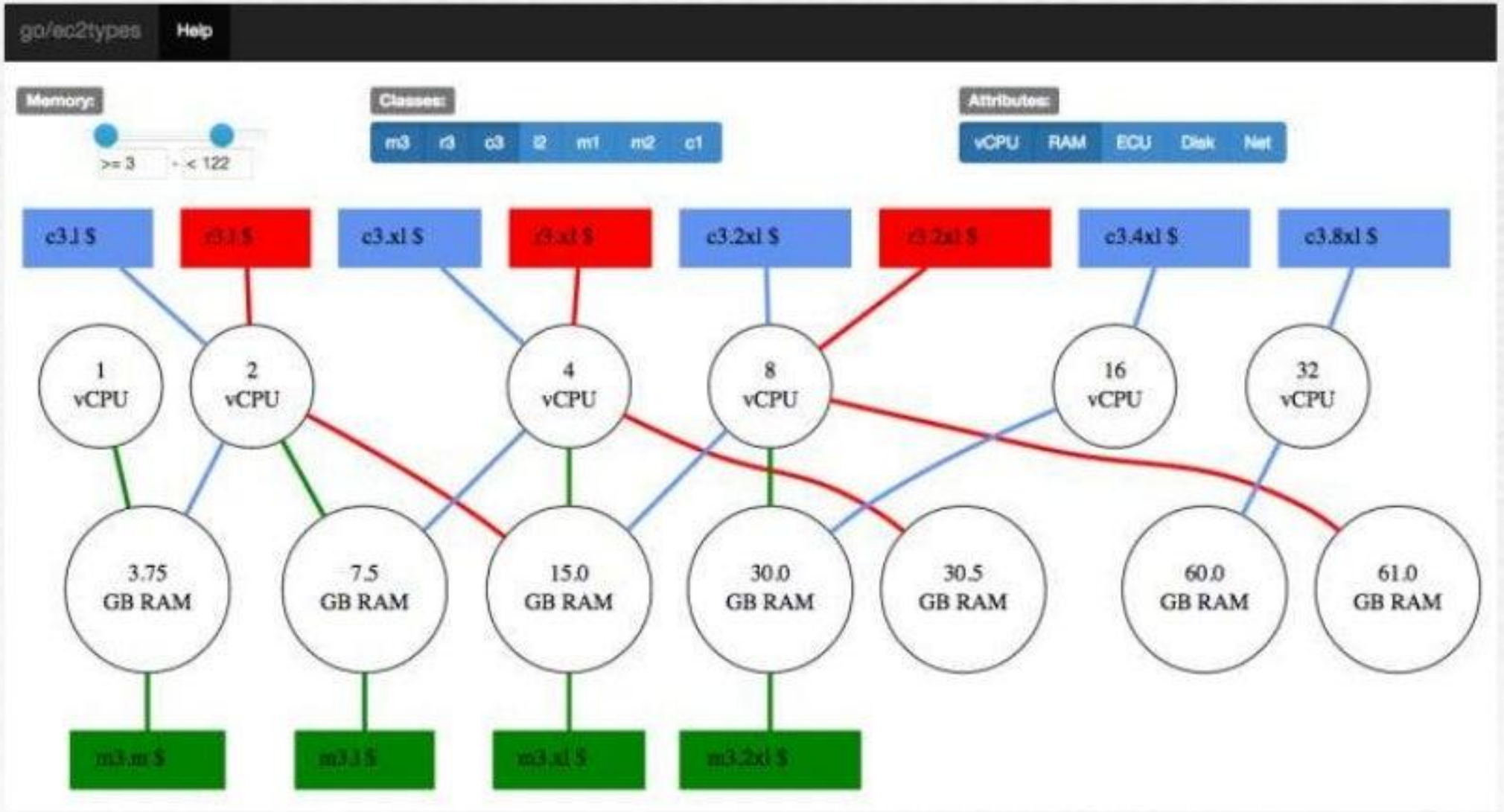
- Elastic Load Balancing allows instance types to be tested with real load
  1. Single instance canary, then,
  2. Auto Scaling Group
- Much better than micro-benchmarking alone, which is extremely error prone



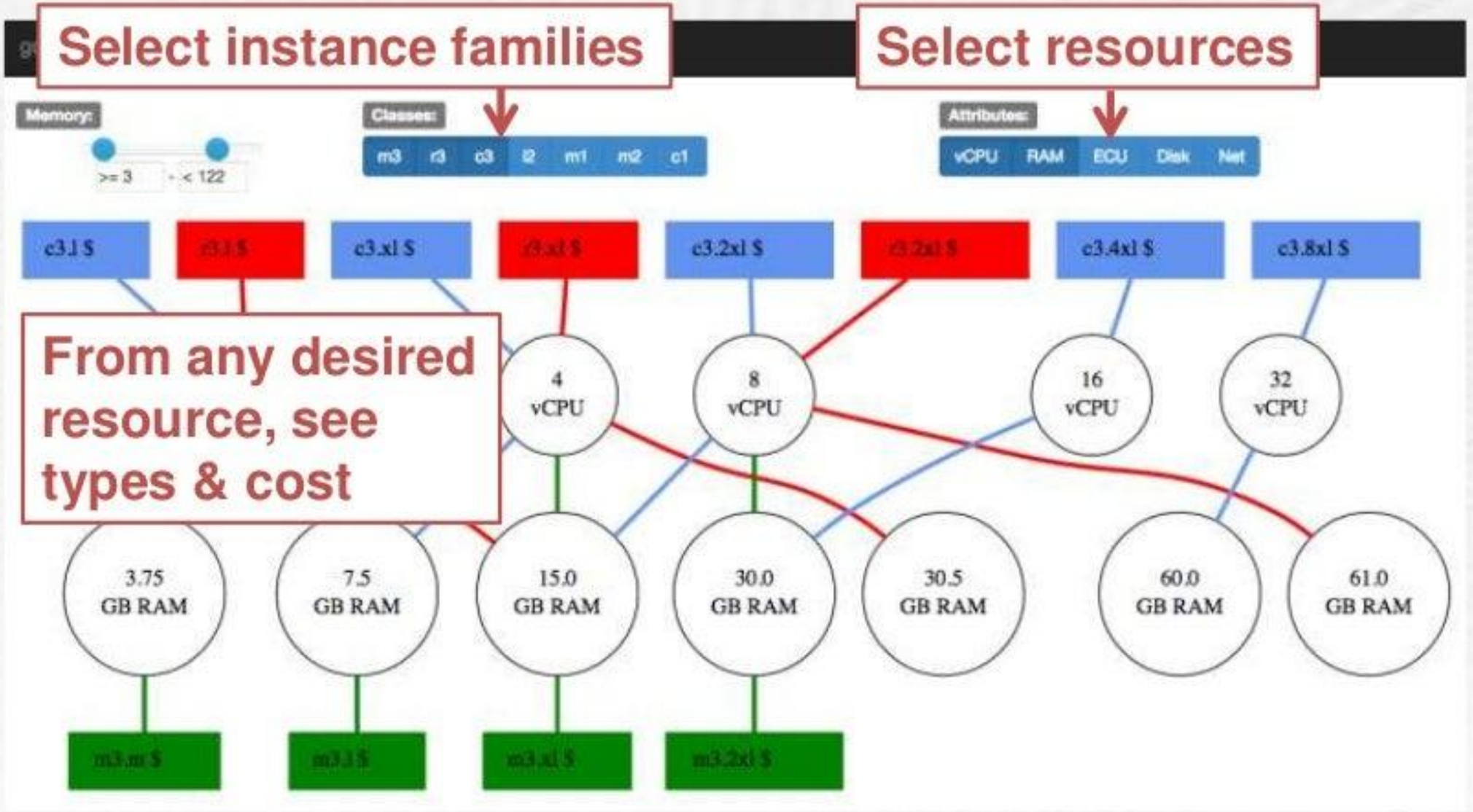
By-Resource Approach

1. Determine bounding resource
   - Eg: CPU, disk I/O, or network I/O
   - Found using:
     - Estimation (expertise)
     - Resource observability with an existing real workload
     - Resource observability with a benchmark or load test (experimentation)
2. Choose instance type for the bounding resource
   - If disk I/O, consider caching, and a memory-optimized type
   - We have tools to aid this choice: Nomogram Visualization

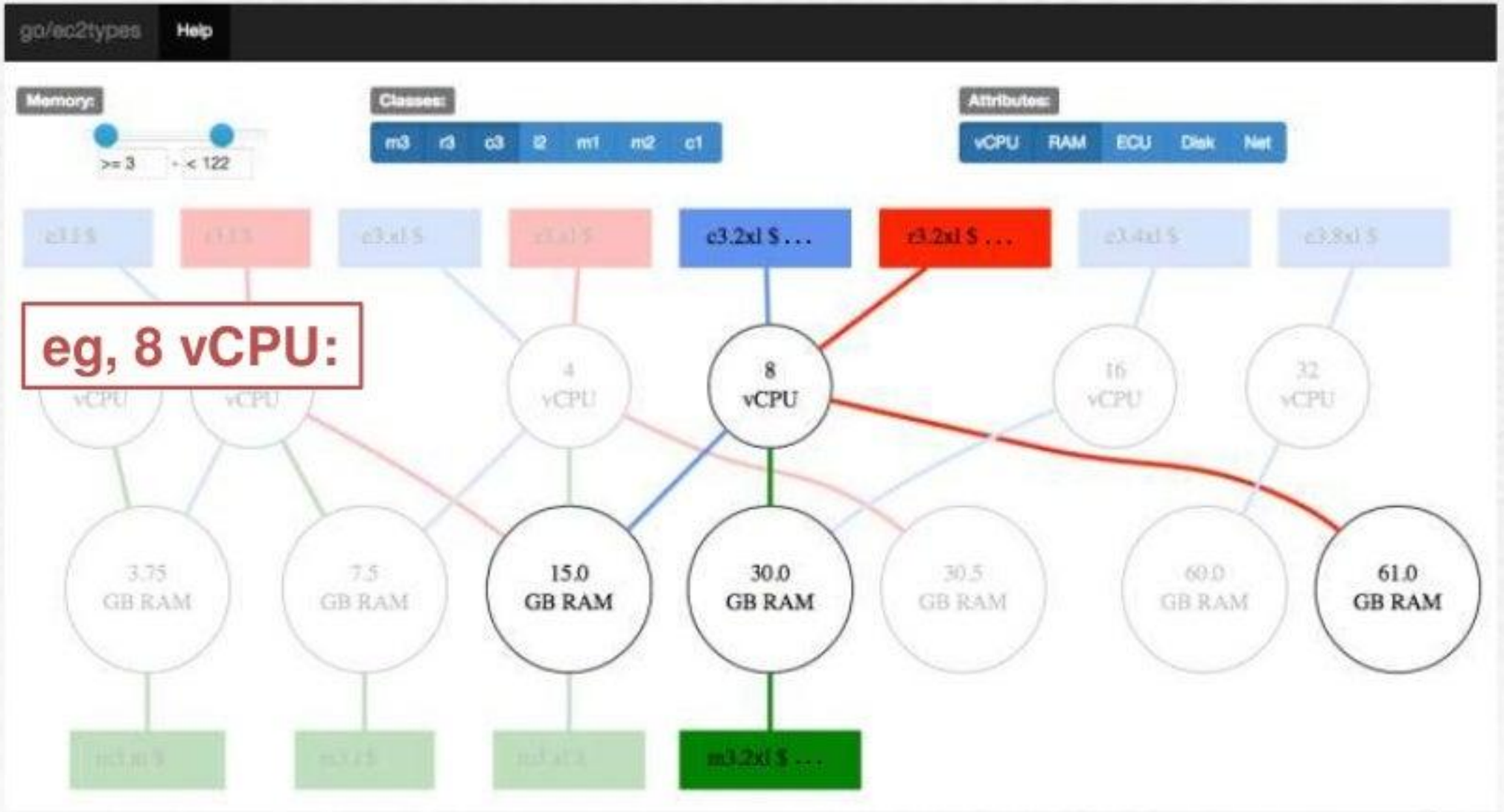
Nomogram Visualization Tool



Nomogram Visualization Tool



Nomogram Visualization Tool



By-Resource Approach, cont.

- This focuses on optimizing a given workload
- More efficiency can be found by adjusting the workload to suit different instance types

Brute Force Choice

1. Run load test on ALL instance types
   - Optionally different workload configurations as well
2. Measure throughput
   - And check for acceptable latency
3. Calculate price/performance for all types
4. Choose most efficient type