Cryptanalytic Bombe at Bletchley Park, UK.
Image credit: Antoine Taveneaux, en.wikipedia.org/wiki/Bombe

Can We Have HPC Data Science?

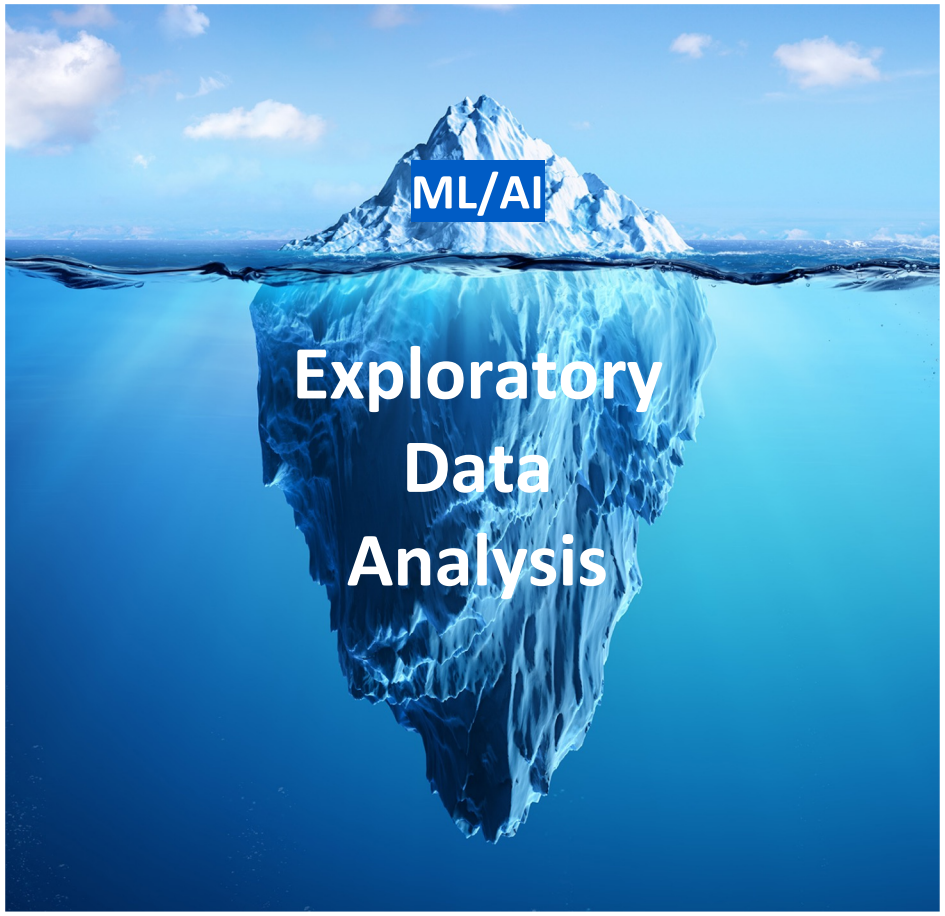
Dr. William Reus

US Department of Defense

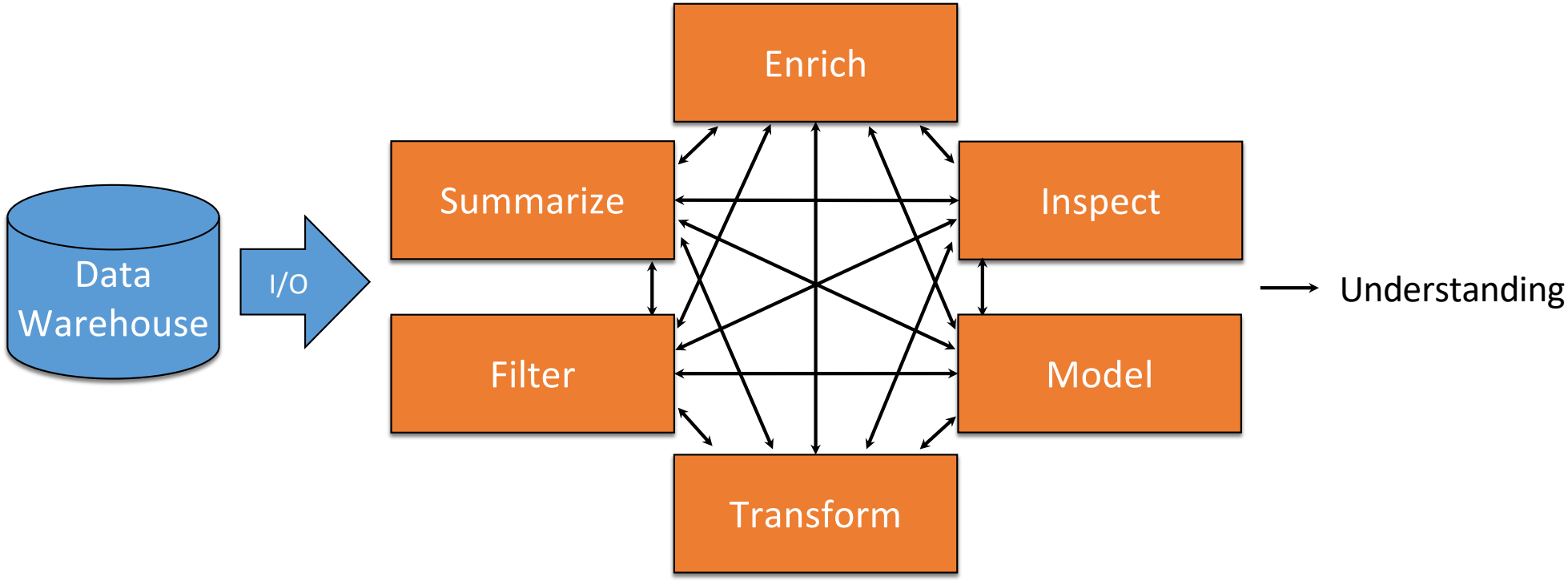
HPC → Data Science

1. HPC data science requires
   a. Interactivity (EDA)
   b. Data warehousing (ETL)
2. Challenges
   a. EDA is bursty
   b. HPC software is siloed
   c. Data breaks everything
3. A culture shift
   a. Think outside the slurm job
   b. Adopt an ecosystem mentality
   c. Honor data engineering

Data Science

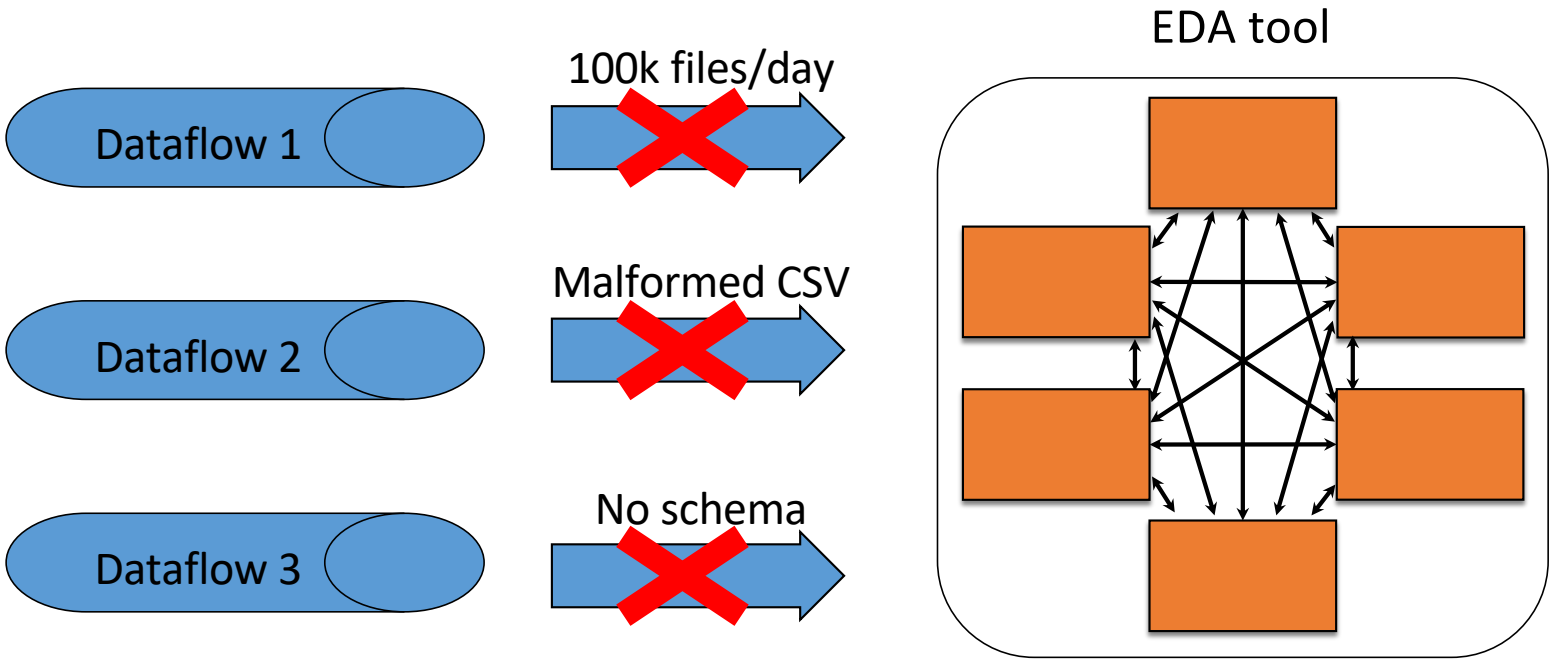


EDA: Learning the Language of a Dataset

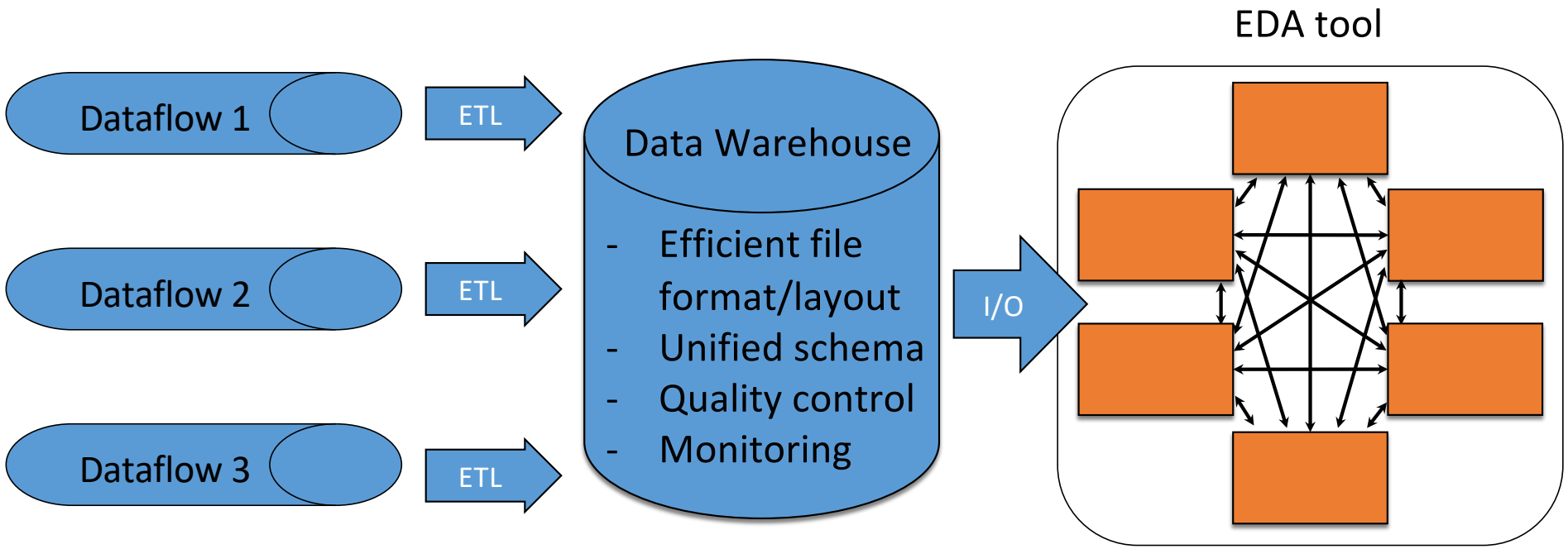


Synonyms: Data Cleaning, Data Wrangling, Data Modeling, Data Mining, etc.

Why Data Warehousing?

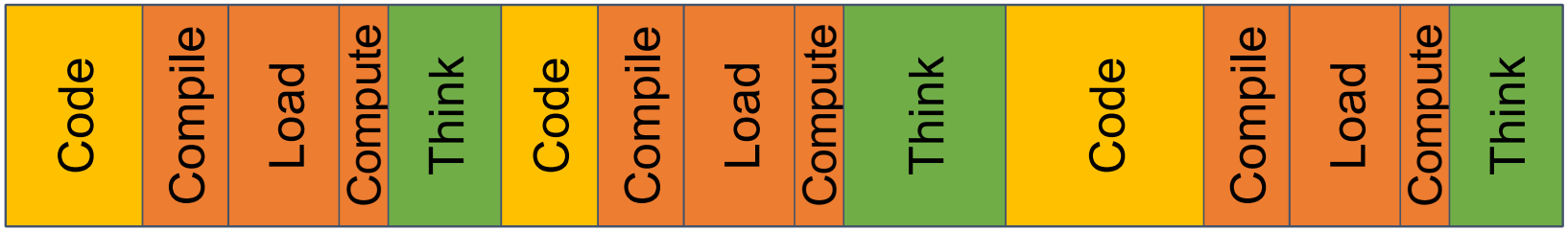


Why Data Warehousing?

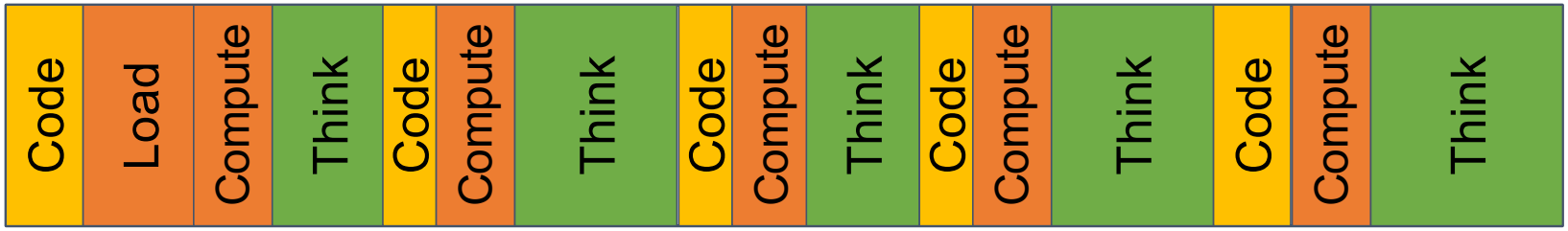


Why Interactive EDA?

Compiled workflow

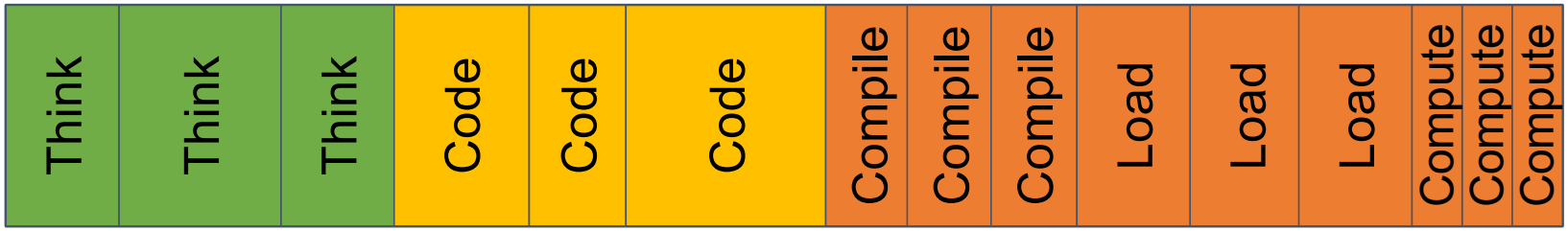


Interactive workflow

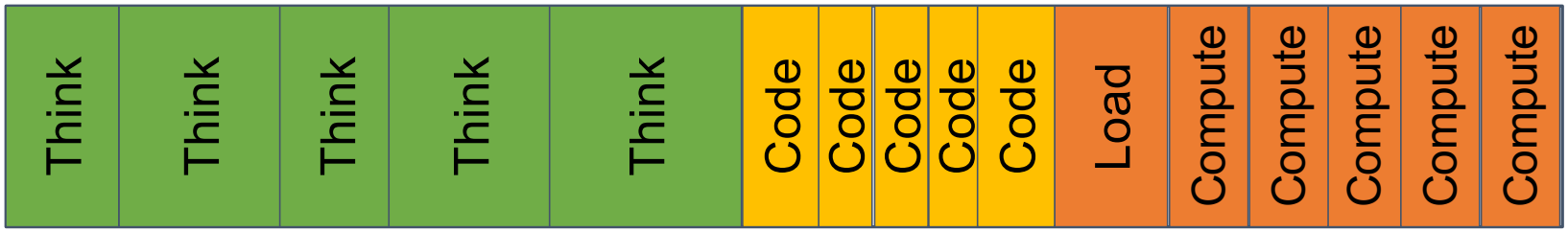


Difference 1: Overhead

Compiled workflow

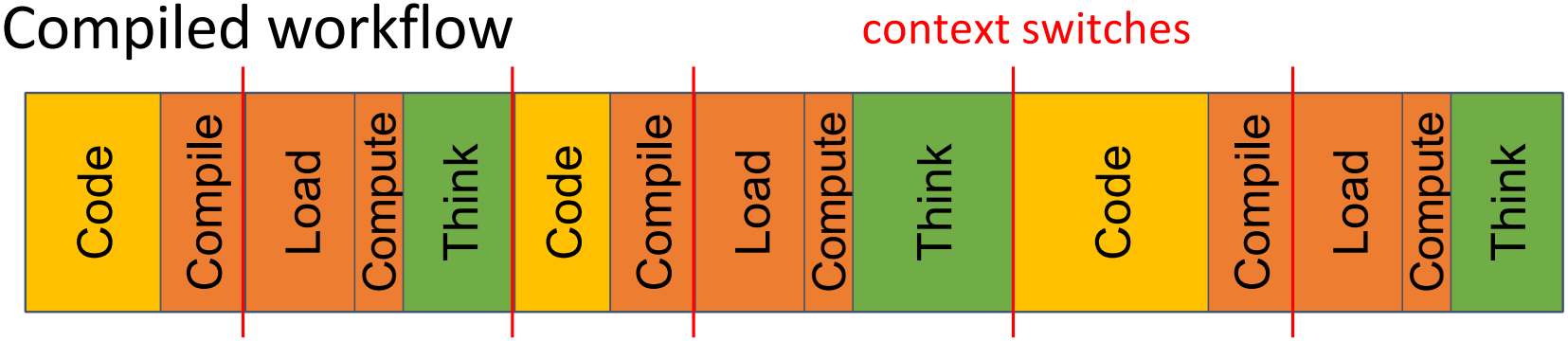


Interactive workflow

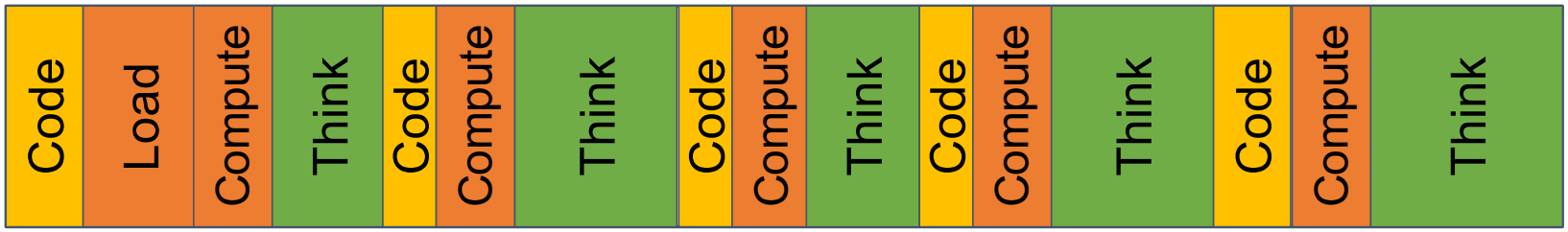


Difference 2: Context Switching

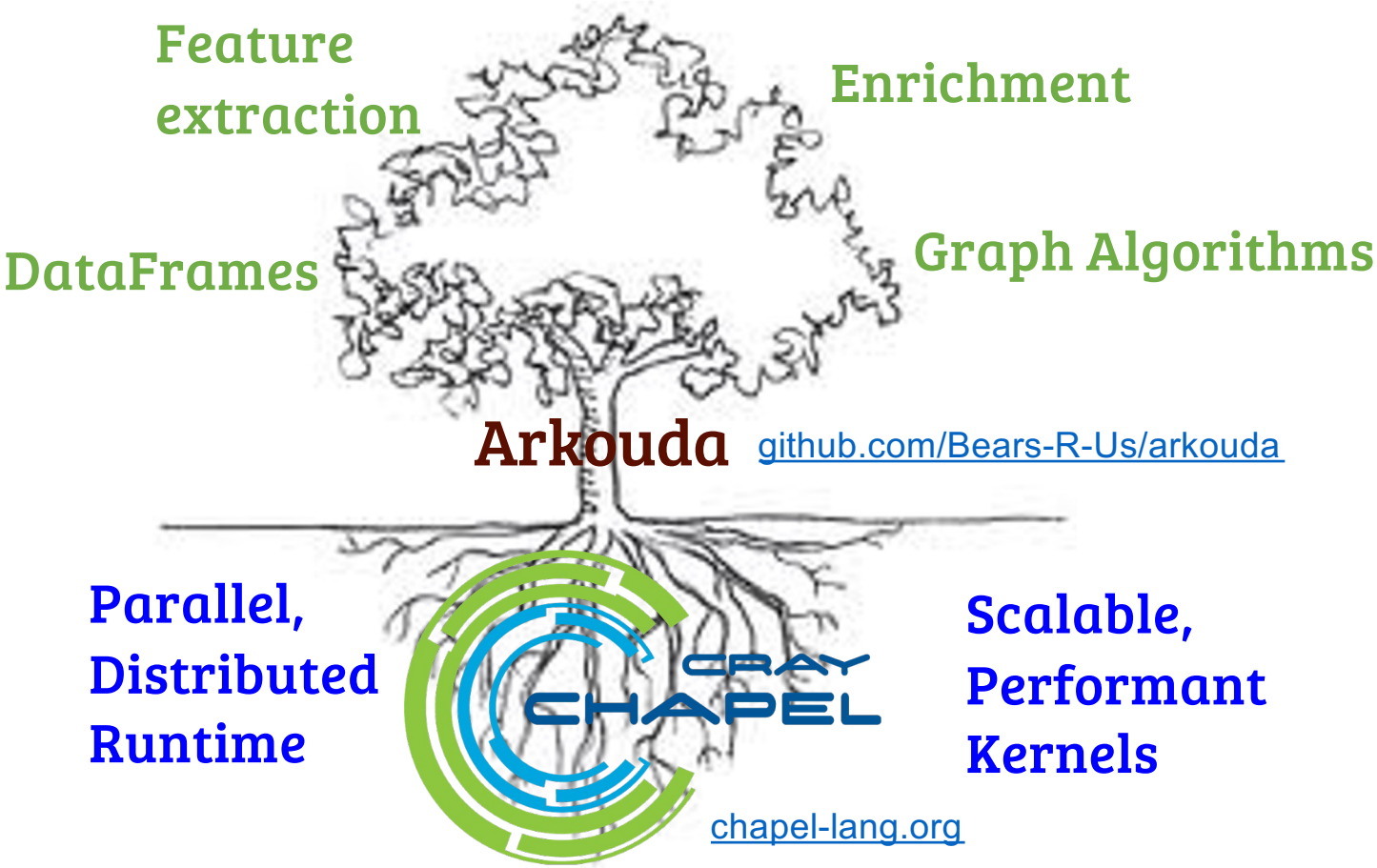
Compiled workflow



Interactive workflow



Arkouda: NumPy for HPC



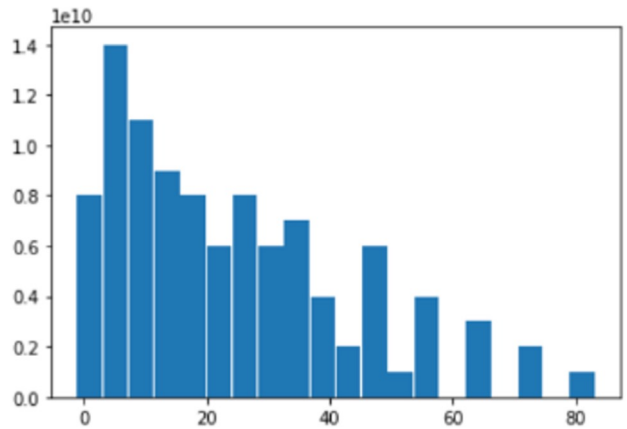
Interactive EDA

```
In [9]: A = ak.randint(0, 10, 10**11)
        B = ak.randint(0, 10, 10**11)
        C = A * B
        hist = ak.histogram(C, 20)
        Cmax = C.max()
        Cmin = C.min()
```

executed in 3.96s, finished 13:45:28 2019-09-12

```
In [10]: bins = np.linspace(Cmin, Cmax, 20)
         _ = plt.bar(bins, hist.to_ndarray(), width=(Cmax-Cmin)/20)
```

executed in 193ms, finished 13:45:28 2019-09-12

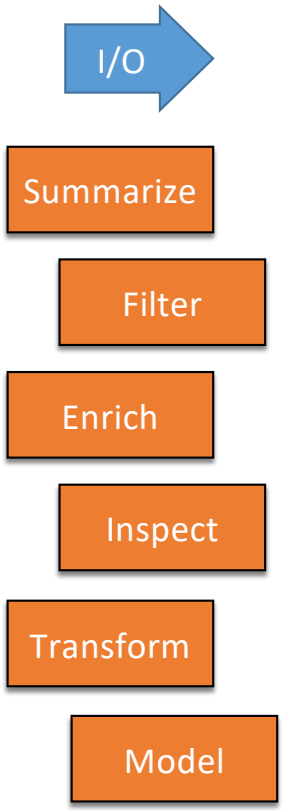


800 GB arrays that persist in distributed memory on compute nodes

Parallel computations

Inline visualization

Interactive EDA



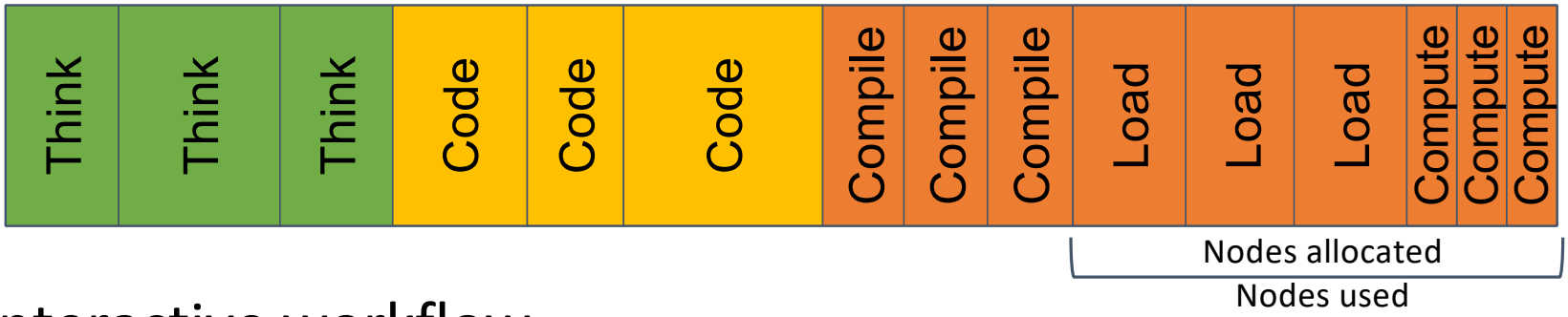
Operation	Example	Approx. Time (seconds)
Read from disk	A = ak.read_hdf()	30-60
Scalar Reduction	A.sum()	< 1
Histogram	ak.histogram(A)	< 1
Vector Ops	A + B, A == B, A & B	< 1
Logical Indexing	A[B == val]	1 - 10
Set Membership	ak.in1d(A, set)	1
Gather	B = Table[A]	4 - 120
Get Item	print(A[42])	< 1
Sort Indices by Value	I = ak.argsort(A)	15
Group by Key	G = ak.GroupBy(A)	30
Aggregate per Key	G.aggregate(B, 'sum')	10

Performance charts (updated nightly): https://chapel-lang.org/perf/arkouda/

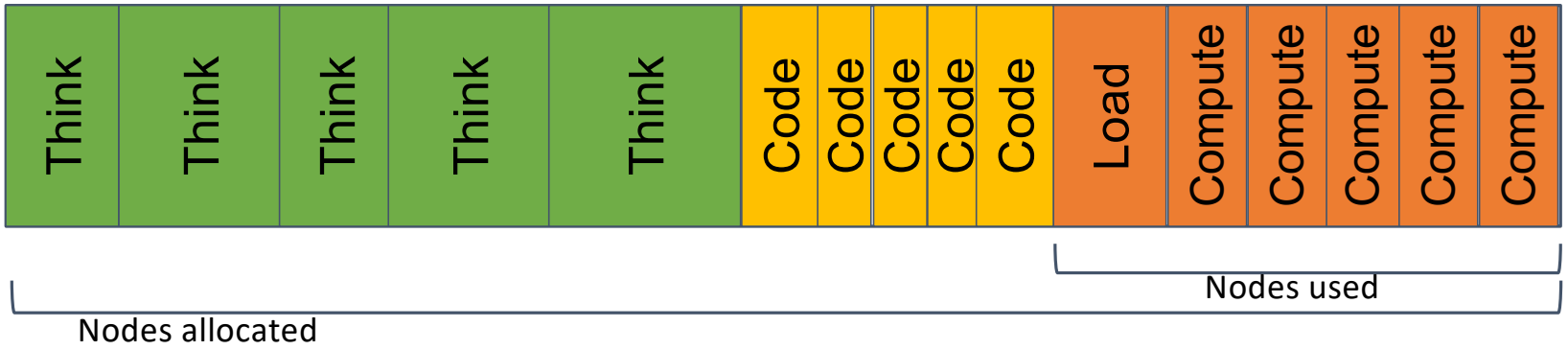
- A, B are 50 billion-element arrays of 32-bit values
- Timings measured on real data
- Hardware: Cray XC40
  - 96 nodes
  - 3072 cores
  - 24 TB
  - Lustre filesystem

Challenge: Trapped Capacity

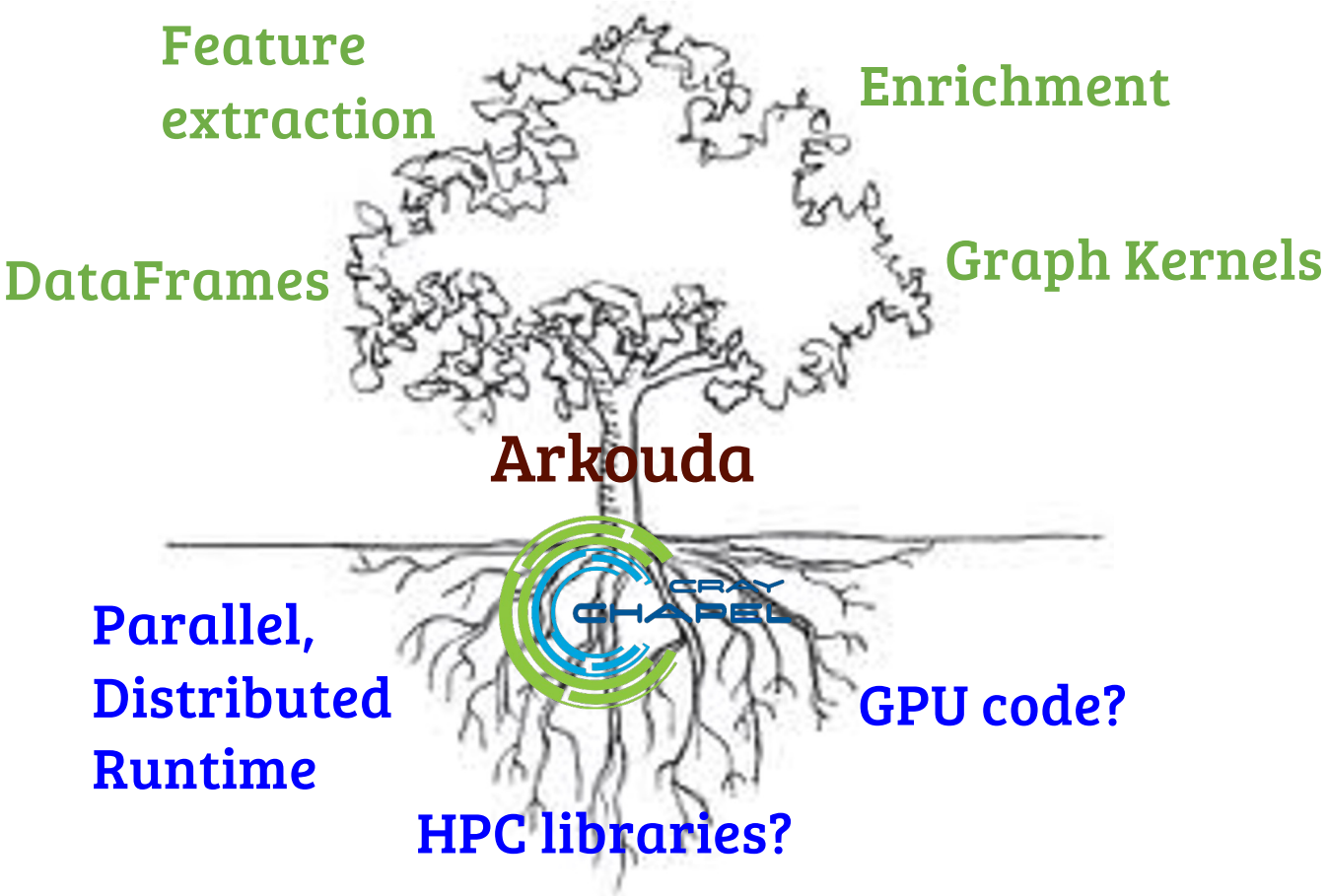
Compiled workflow



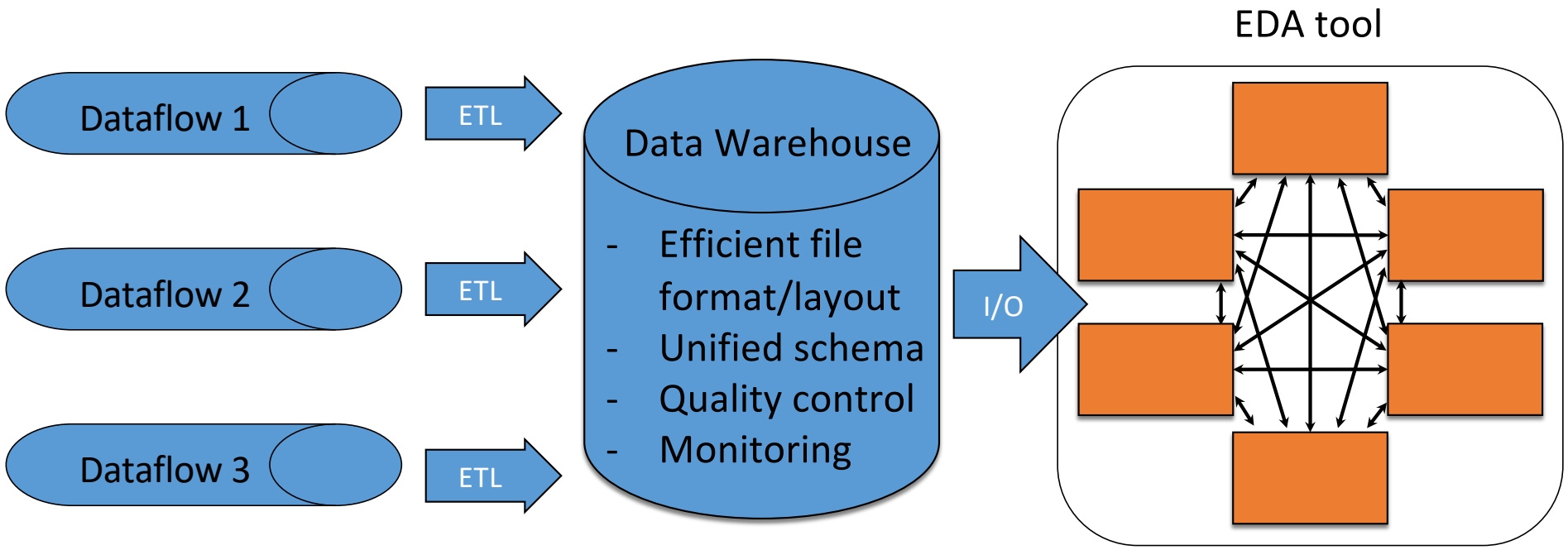
Interactive workflow



Challenge: Tool Integration



Challenge: HPC Data Engineering



HPC → Data Science

1. Think outside the slurm job
   a. Vision: persist variables to fast FAM/FAS and release nodes between computations
   b. Vision: interact with data already loaded by teammate
   c. Example: Hierarchical object storage
2. Develop for the ecosystem
   a. Vision: load and clean a graph in Arkouda, then call out to [HPC graph library], all within Jupyter notebook (no context switching!)
   b. Example: require standard data interfaces/layouts
3. Honor HPC data engineering
   a. Vision: data warehouse with cloud-like reliability and HPC scalability
   b. Example: professionalize and incentivize HPC data engineering like we do HPC software and hardware engineering

Acknowledgements

- Arkouda developers and users
- Chapel team at HPE
- Partners in academia and national labs

chapel-lang.org

github.com/Bears-R-Us/arkouda