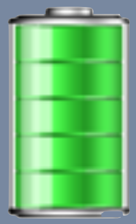
SEEDS: A SOFTWARE ENGINEER'S ENERGY OPTIMIZATION DECISION SUPPORT FRAMEWORK

Irene Manotas, **Lori Pollock**, and James Clause

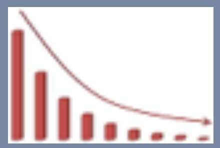
University of Delaware

SOFTWARE ENERGY USAGE: WHY CARE?

Reducing energy consumption can:



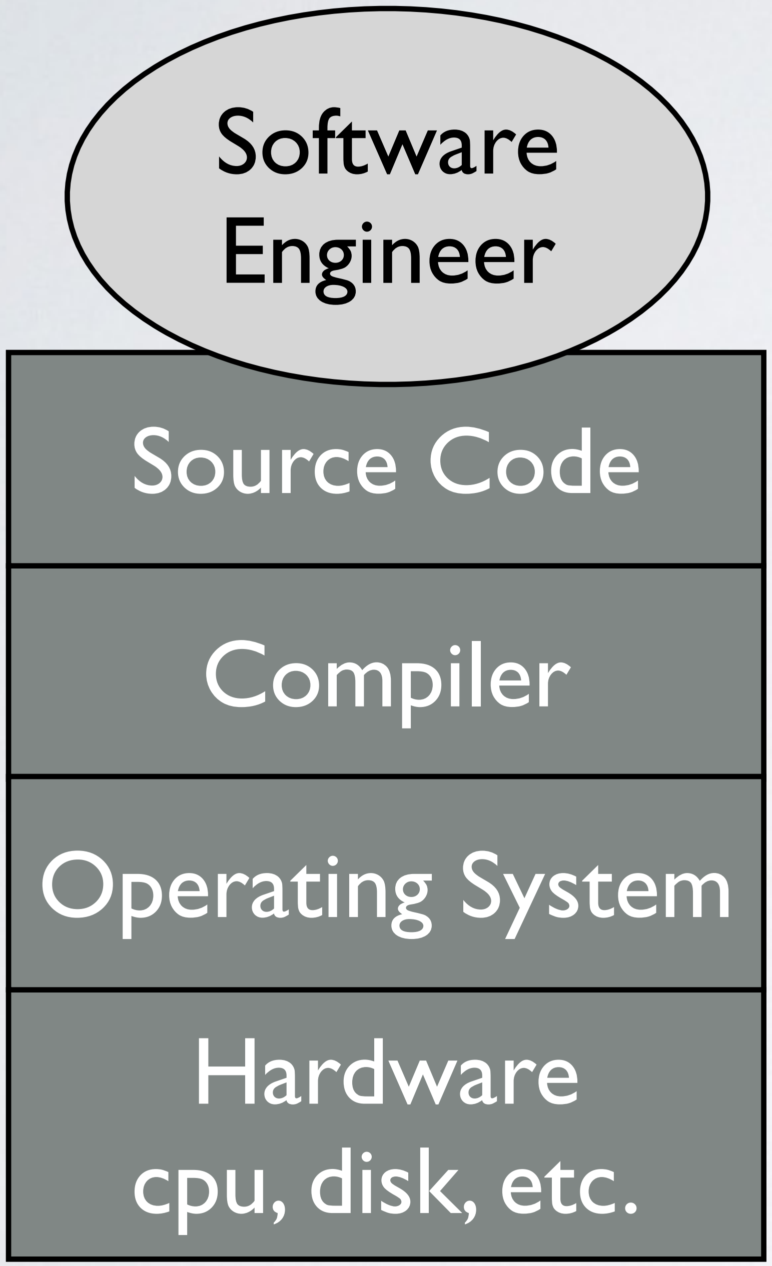
Extend battery life (mobile devices)



Reduce electricity bills and related costs (data centers)

Support sustainability

STATE OF THE ART



Can the software engineer help?

Energy Reducing Transformations

Manage Energy Usage

Dynamic Voltage Frequency Scaling

YES, THEY CAN

Many decisions made by developers impact energy usage:

- Algorithms:

  Increase or decrease energy usage up to 30%
  (Bunse et al., 2009; Zhuo and Chakrabarti, 2008)

- Design Patterns:

  Increase or decrease energy usage up to 700%
  (Sahin et al., 2012)

- Refactorings:

  Increase or decrease energy usage up to 7%
  (Sahin et al., 2014)

HOW CAN DEVELOPERS HELP?

```
Collection x = new ??
List x = new ??
```

Choose the most energy efficient implementation

LinkedHashSet
TreeSet
HashSet
EnumSet
CopyOnWriteArraySet
⋮
~ 40
JCF

UnifiedSet
UnifiedMap
SetAdapter
MultiReaderFastList
InmutableUnifiedMap
⋮
~ 40
Goldman Sachs

FastSet
FastBitSet
LocalMap
FastMap
FastList
⋮
~ 10
javolution

CharArraySet
ObjectArraySet
ObjectAVLTreeSet
OpenObjHashBigSet
ObjectRBTreeSet
⋮
~ 700
:::fastutil:

Hundreds of possibilities for each choice.

Hundreds of decision points!

SEEDS

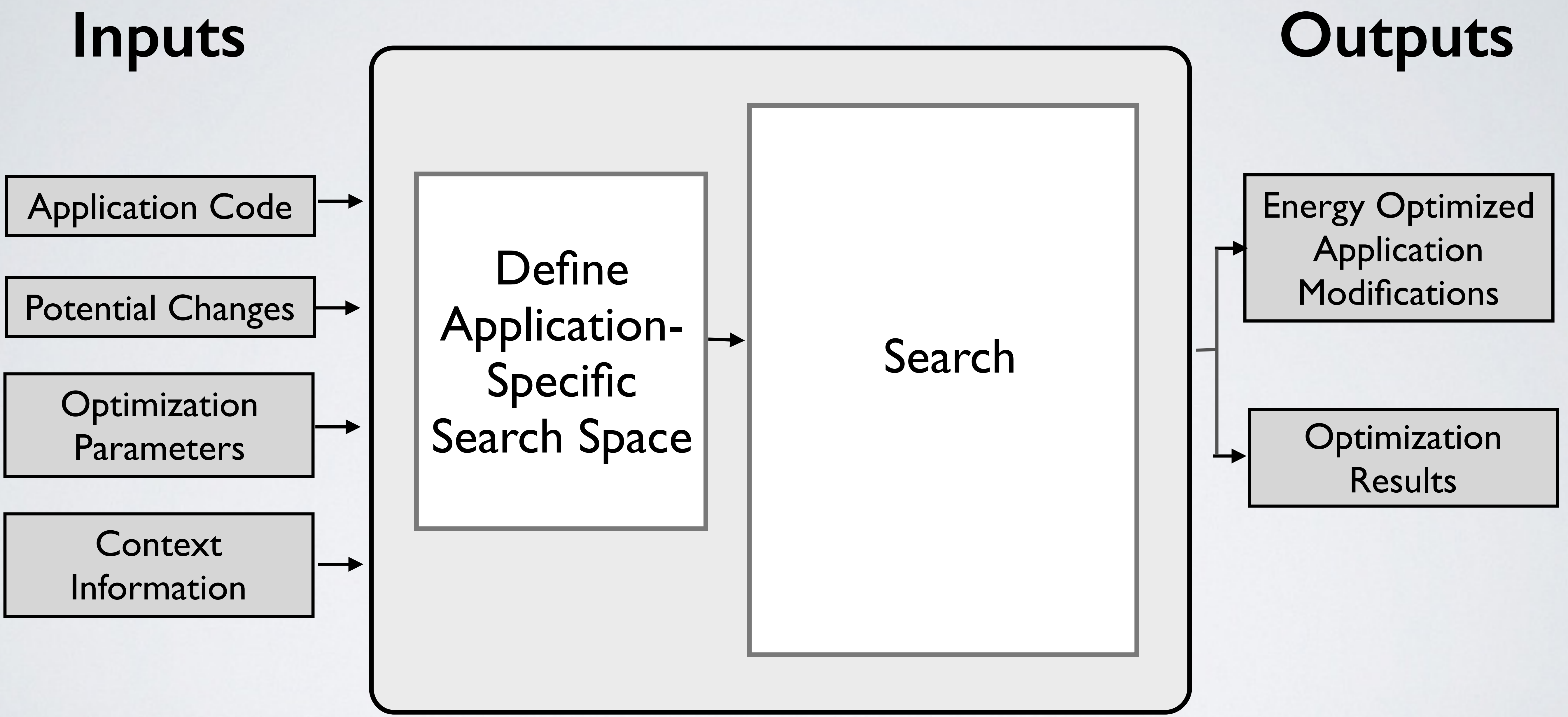
Software **E**ngineer's **E**nergy-optimization **D**esign Framework

Help developers improve energy usage of their applications by making decisions about which source-level changes to apply

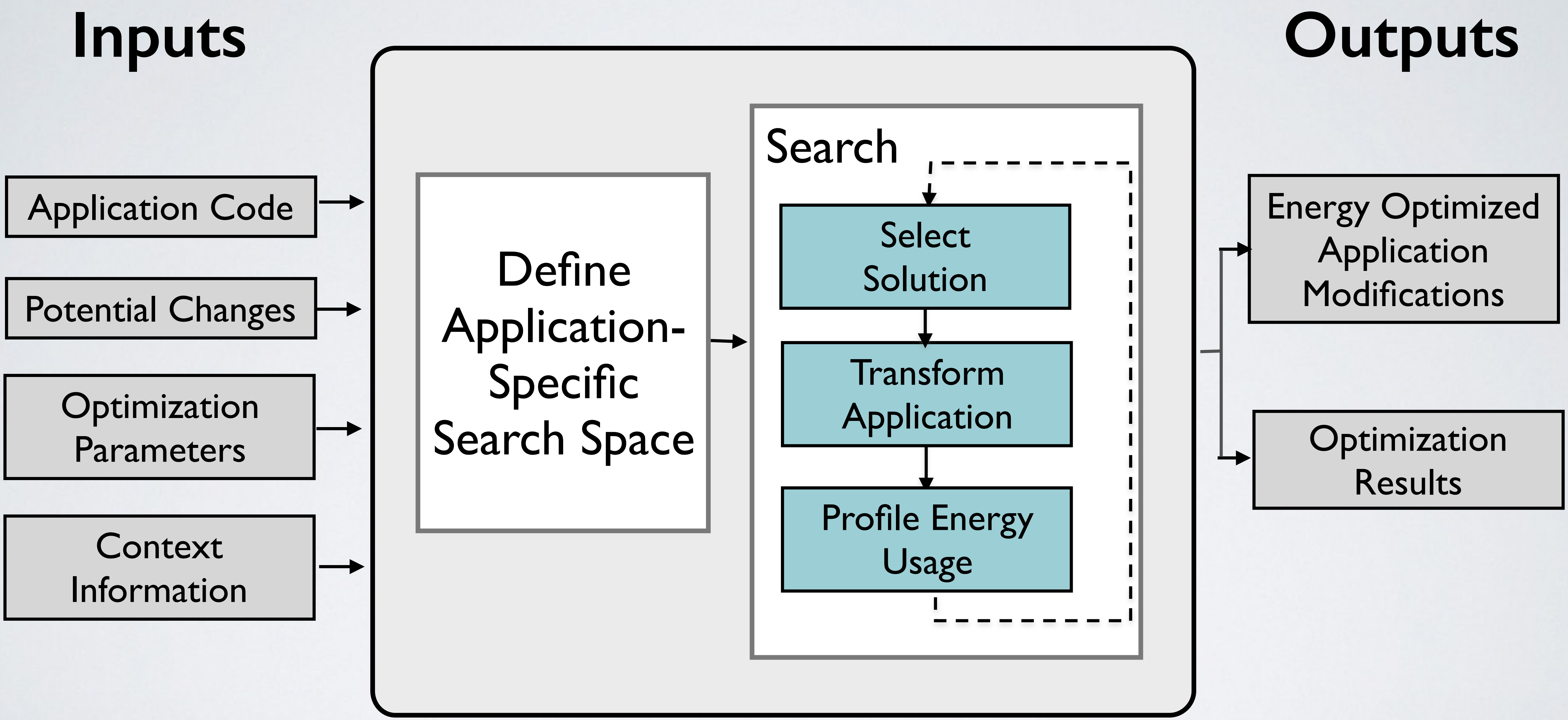
THE SEEDS FRAMEWORK

1. Automatically apply changes to explore decision space with regard to energy usage (many versions)
2. Abstract away tedious system level concerns:
   - create versions; monitor energy usage; select
3. Support different software engineering decisions

SEEDS APPROACH

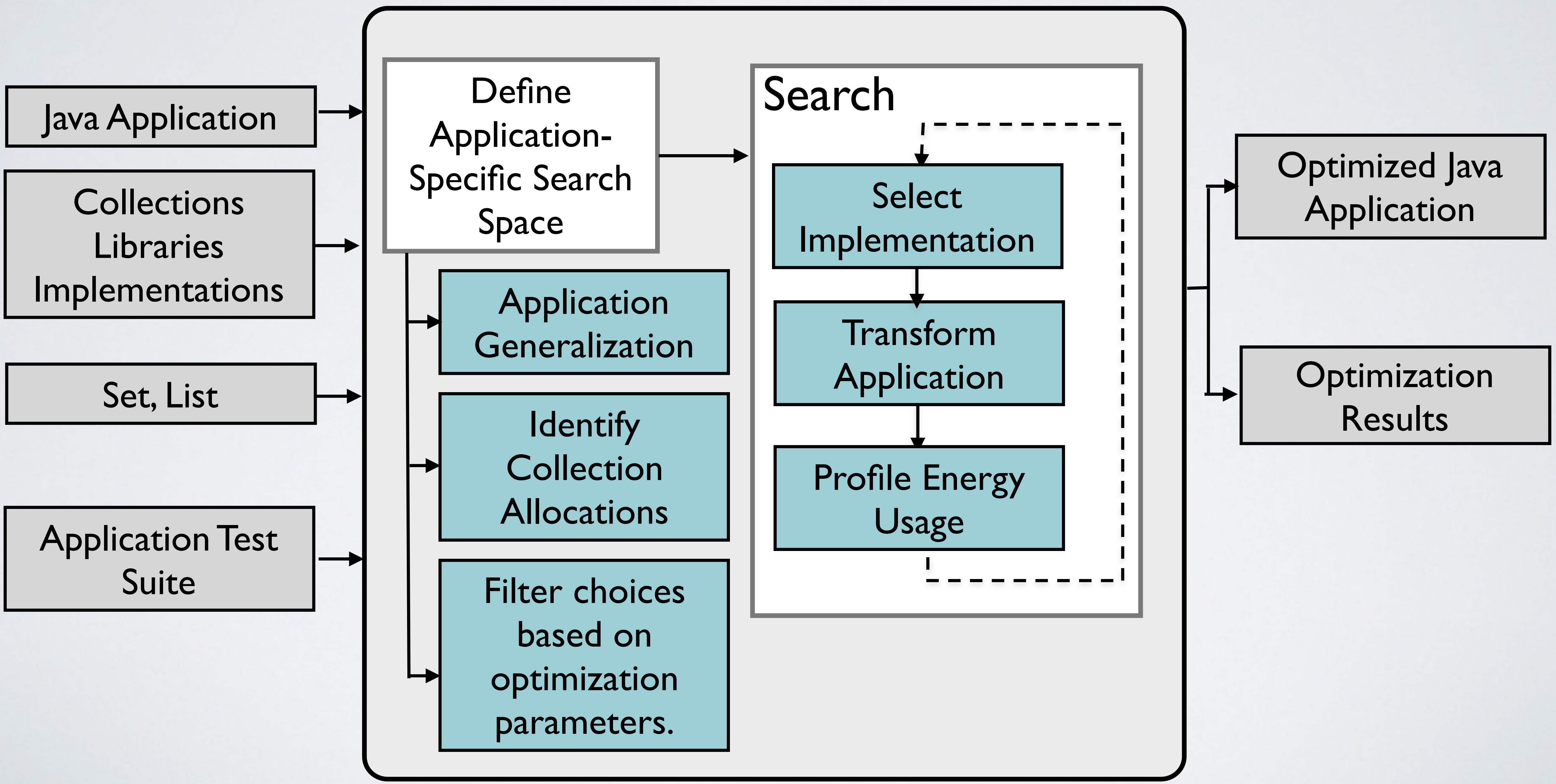


SEEDS FRAMEWORK COMPONENTS



AN INSTANTIATION: SEEDS$_{\text{API}}$

SEEDS$_{\text{api}}$ supports the selection of Collections implementations to optimize Java applications.



EVALUATION OF SEEDS

- RQ1: Effectiveness

  Can SEEDS improve the energy usage of applications?

- RQ2: Exploration Capability

  How does the size of the search space impact effectiveness?

  How often do developers choose the most energy efficient implementation?

  How often is an implementation the most energy efficient?

- RQ3: Cost

  What is the time required to find the best solution?

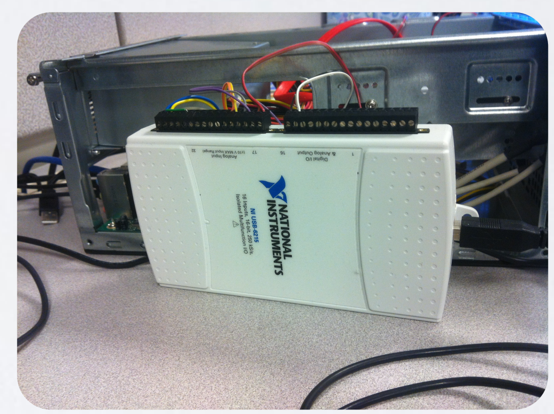
EVALUATION METHODOLOGY

- Independent Variable: Library Implementation

  157 implementations from 6 Collections Libraries

JCF

- Dependent Variable: Energy Usage (joule)



LEAP Machine

- Subjects

  Seven Java applications

  8k - 100k LoC

Barbecue, Jdepend, Apache-XML-Security, Jodatime, Commons-Lang, Commons-beanutils, Commons-cli

RQ1: EFFECTIVENESS

Application	% Improvement (JCF only)	% Improvement (all 6 libraries)
Barbecue	17∗	17∗
Jdepend	3∗	6∗
Apache-xml-security	5	5
JodaTime	8∗	9

Interactions between transformations cancel out expected benefits

SEEDS$_{api}$ was able to improve the energy efficiency of 6/7 applications.

RQ2: EXPLORATION CAPABILITY

How does the size of the search space impact effectiveness?

Application	% Improvement (JCF only)	% Improvement (all 6 libraries)	Increase
Barbecue	17∗	17∗	–
Jdepend	3∗	6∗	3
Apache-xml-security	5	5	–
JodaTime	8∗	9	1
Commons-lang	10	13	3
Commons-beanutils	–	–	–
Commons-cli	2∗	2∗	–

∗ only one change   † multiple changes

More choices can result in larger improvements.

RQ2: EXPLORATION CAPABILITY

How often do developers choose the most energy efficient implementation?

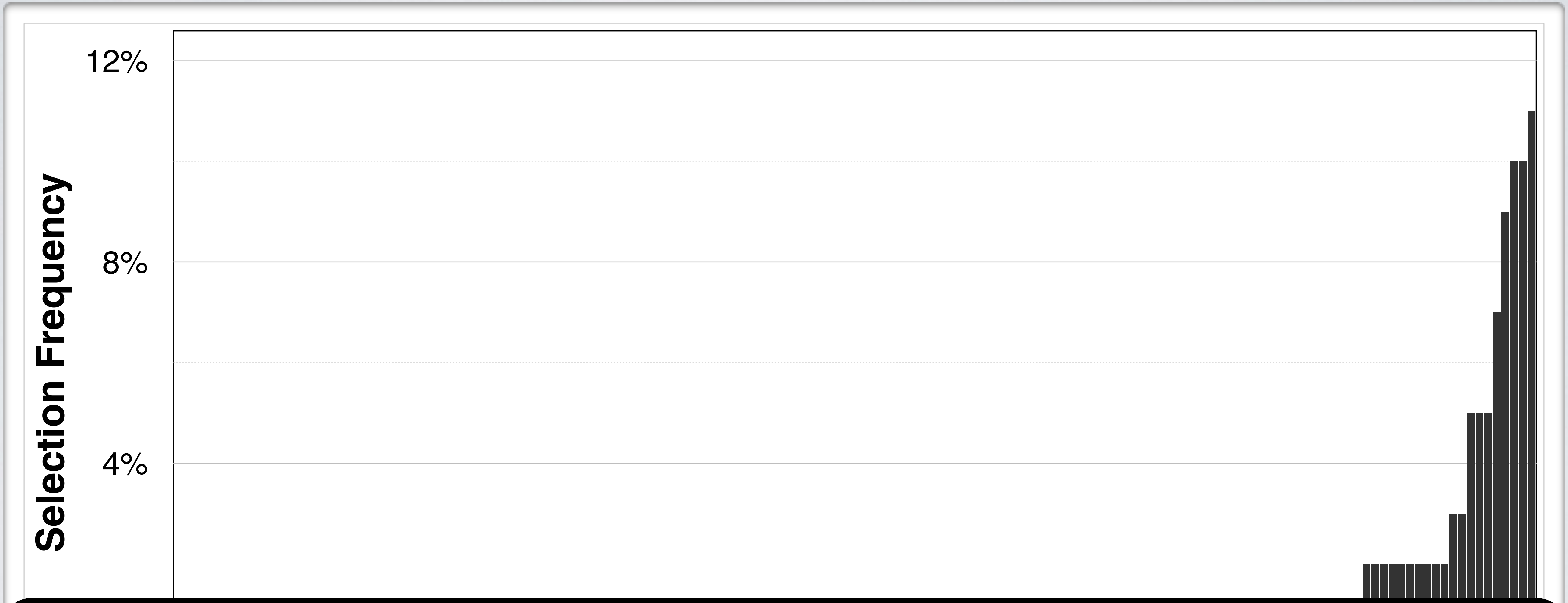
- SEEDS analyzed a total of **123** decision points
- Switching from the developer's original choice to a new choice improved the energy usage:

  JCF only: 56% (69 times)
  All Libraries: 72% (89 times)

Developers are unlikely to choose the most efficient implementation.

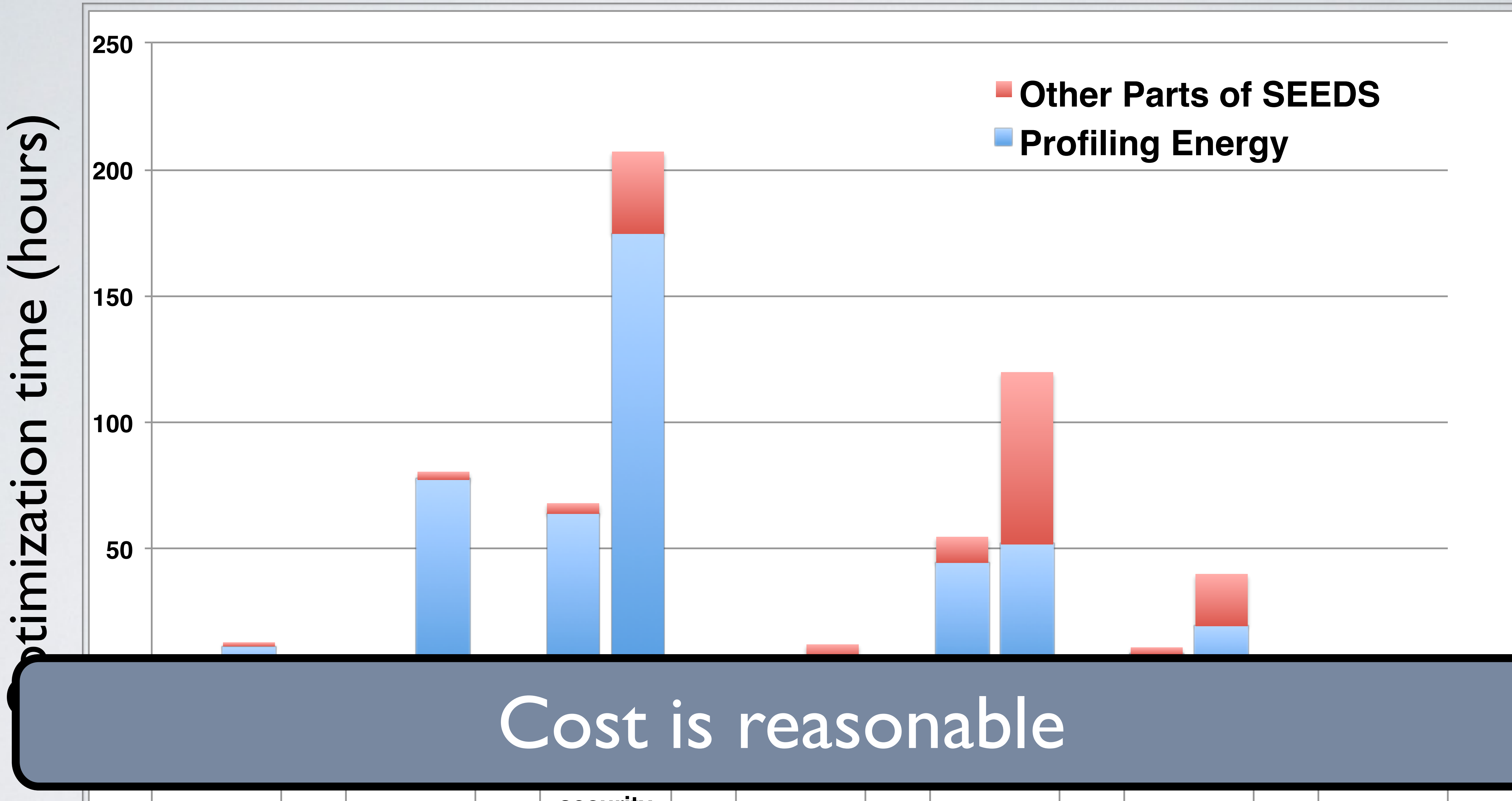
RQ2: EXPLORATION CAPABILITY

How often is a given implementation the most energy efficient?



No single implementation is the most energy efficient.

RQ3: COST OF SEEDS



Cost is reasonable

Cost could be improved by using different energy measurement techniques

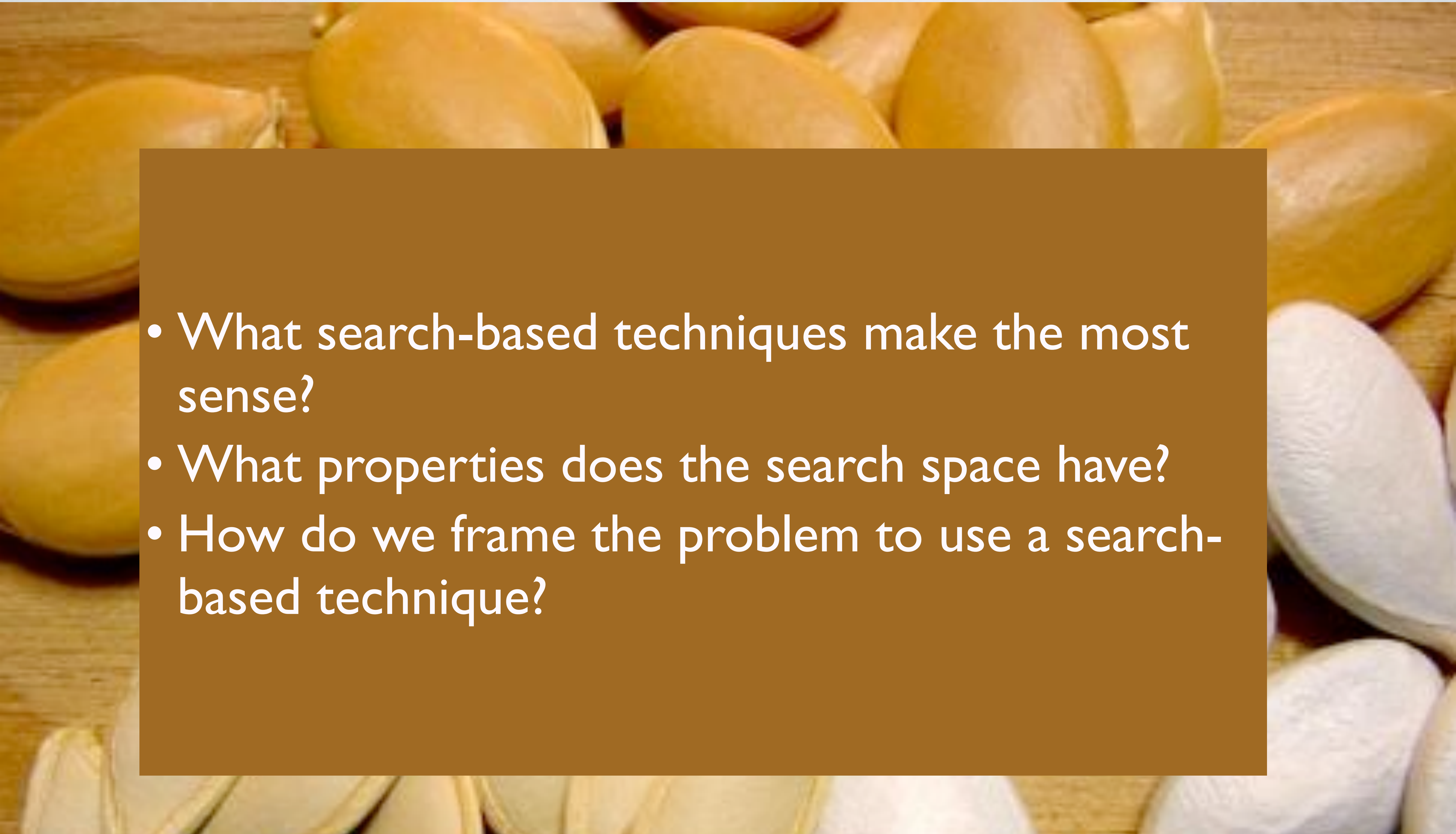
CONCLUSIONS

- Designed SEEDS, a general framework for supporting decisions to improve applications' energy usage.
- Demonstrated feasibility/usefulness:
  - $\text{SEEDS}_{\text{api}}$, an instantiation of SEEDS for API implementation selection.
- Evaluated SEEDS with 7 Applications:
  - effective at improving the energy usage of Java applications
  - supports asking and answering energy-usage related questions

FUTURE WORK

- Improve SEEDS search strategy
  - many decisions
  - many decision points in code
- Explore more instantiations
  - how general

IMPROVING SEEDS SEARCH



- What search-based techniques make the most sense?
- What properties does the search space have?
- How do we frame the problem to use a search-based technique?

ADDITIONAL RELATED WORK

- **Measure the Energy Consumption of Applications**

  Provide energy usage of application by different approaches (HW instrumentation, simulation, estimation) (e.g., Singh, 2010; Hao, 2013).

  Does not provide information about which changes could be made to reduce the energy.

- **Language Support**

  Implement energy management policies according to device status, tasks being executed, events or features (e.g., Cohen, 2012; Malakuti, 2014; Siegmund, 2010).

  Need to know which policies to implement and where and when to apply them manually.