Modelling and Learning Software Variability

Mathieu Acher (Associate Professor)

https://www.mathieuacher.com

https://teaching.variability.io

https://varyvary.github.io/

@acherm

DiverSE

Diversity-Centric Software Engineering

Inria INVENTEURS DU MONDE NUMÉRIQUE | UNIVERSITÉ DE RENNES 1 | INSA RENNES | cnrs

Disclaimer

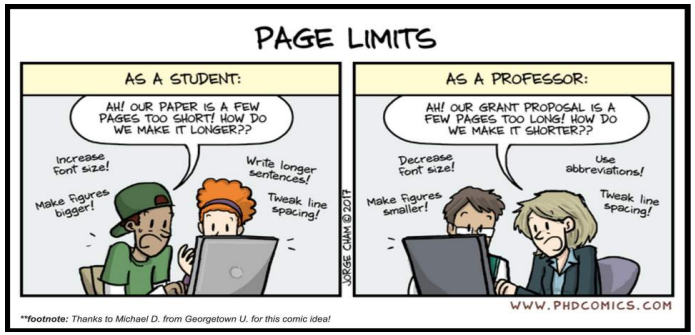
- Slides for the EJCP 2021 course
  - ~French summer school for PhD candidates in programming, verification, software engineering, etc.
- Abstract: *Most modern software systems are subject to variation or come in many variants. Web browsers like Firefox or Chrome are available on different operating systems, in different languages, while users can configure 2000+ preferences or install numerous 3rd parties extensions (or plugins). Web servers like Apache, operating systems like the Linux kernel, or a video encoder like x264 are other examples of software systems that are highly configurable at compile-time or at run-time for delivering the expected functionality andmeeting the various desires of users. Variability ("the ability of a software system or artifact to be efficiently extended, changed, customized or configured for use in a particular context") is therefore a crucial property of software systems. Organizations capable of mastering variability can deliver high-quality variants (or products) in a short amount of time and thus attract numerous customers, new use-cases or usage contexts. A hard problem for end-users or software developers is to master the combinatorial explosion induced by variability: Hundreds of configuration options can be combined, each potentially with distinct functionality and effects on execution time, memory footprint, quality of the result, etc. I will briefly introduce variability-intensive systems, their applications and challenges, in various software contexts with intuitive examples and real-world systems (like the Linux kernel or JHipster Web generator). Then I will show the relevance of artificial intelligence (AI) techniques for exploring and taming such enormous variability spaces. In particular, I will introduce how (1) satisfiability and constraint programming solvers can be used to properly model and reason about variability; (2) how machine learning can be used to discover constraints and predict the variability behavior of configurable systems or software product lines. We will start and end up the course with an exercise and challenge: fitting a LaTeX generator to only authorize PDF papers fitting formatting instructions of a conference!*
- I had 90 minutes (~ 3 hours)

Agenda

- VaryLaTeX (20’)
- Software Variability: An Overview (10’)
- AI1: Modeling and Reasoning about Variability (20’)
  - Jhipster case study
  - feature models: syntax, semantics, and logics
- AI2: Learning Variability (20’)
  - x264 case study
  - Sampling, measuring learning
- Back to VaryLaTeX (20’)

Challenge

- You have 10 minutes to fit a 5 pages paper into 4!
  - Written in LaTeX
  - Deadline
  - Meta-constraint: you can only edit "**values.tex**"
  - Multiple Overleaf links in the chat!
  - 6 max. per project (uniform random selection will collaboratively emerge, fingers crossed)



http://phdcomics.com/comics.php?f=1971

https://www.overleaf.com/6588449215bfnprspjxzsf

https://www.overleaf.com/7362463758wphfmszyfbcv

https://www.overleaf.com/9463927852znhjjqdjjzdk

https://www.overleaf.com/8125313914pbvssvdnscnn

https://www.overleaf.com/4586992838mcqwgtfpfvxy

https://www.overleaf.com/6262862787hcprdhwwftgd

https://www.overleaf.com/3911269559yqspcgdzhkkz

https://www.overleaf.com/9242511369bskpddyydhff

Or
git clone https://git.overleaf.com/60cc56fab0c4d8147ef166f0
pdflatex + bibtex + pdflatex + pdflatex

Variability

- "the **ability** of a software system or artifact to be efficiently extended, changed, customized or configured for use in a particular context" (Svahnberg et al. 2005)
  - software/**customization** perspective

- Terminology/applicability
  - Software product lines, configurable systems, variability-intensive systems, dynamic adaptive systems, generators
  - Options ~= features, flags, parameters, variation points, etc.

Software Variability

- Configurable system

VaryLaTeX

- Configuration options (aka software features)

template variables of a LaTeX file

- Variants

LaTeX source and PDF variants (papers)

- Large variability spaces

73,440 possible variants

Software Variability

- Configurable system

Linux operating system

- Configuration options (aka software features)

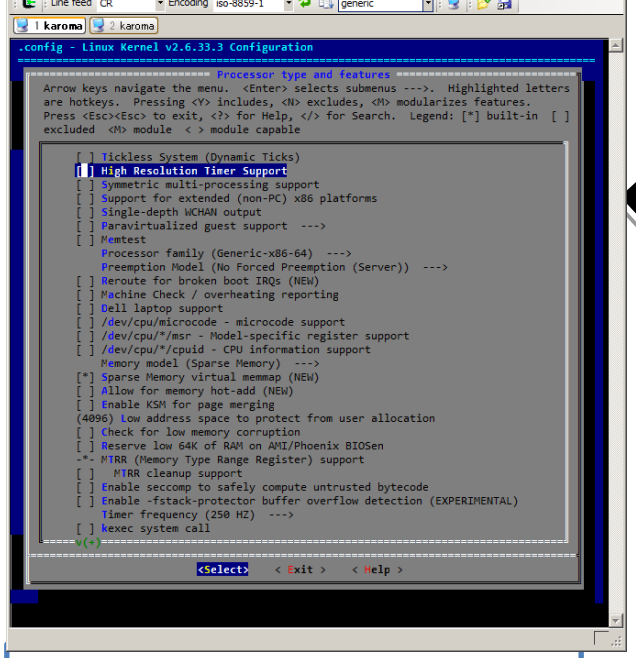
conditional compilation (#ifdef) in C files

- Variants

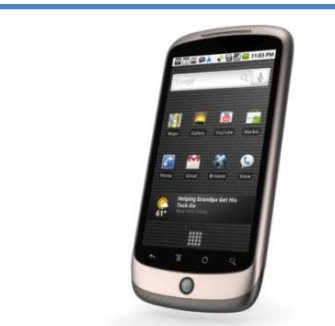
Linux kernel variants

- Large variability spaces

16,000 options (~“yes”, “no”, “module”)



Linux Kernel



Software Variability

- Configurable system

Firefox web browser

- Configuration options (aka software features)

feature flags (about:config)

- Variants

Firefox behavior (e.g., security)

- Large variability spaces

2000+ options (Boolean, categorical, numeric)

Software Variability

- Configurable system

Scikit

- Configuration options (aka software features)

Hyper-parameters

- Variants

Machine learning algorithm behavior

- Large variability spaces

Dozens of options (Boolean, categorical, numerical)

Software Variability

- Configurable system

x264 video encoder

- Configuration options (aka software features)

command line parameters

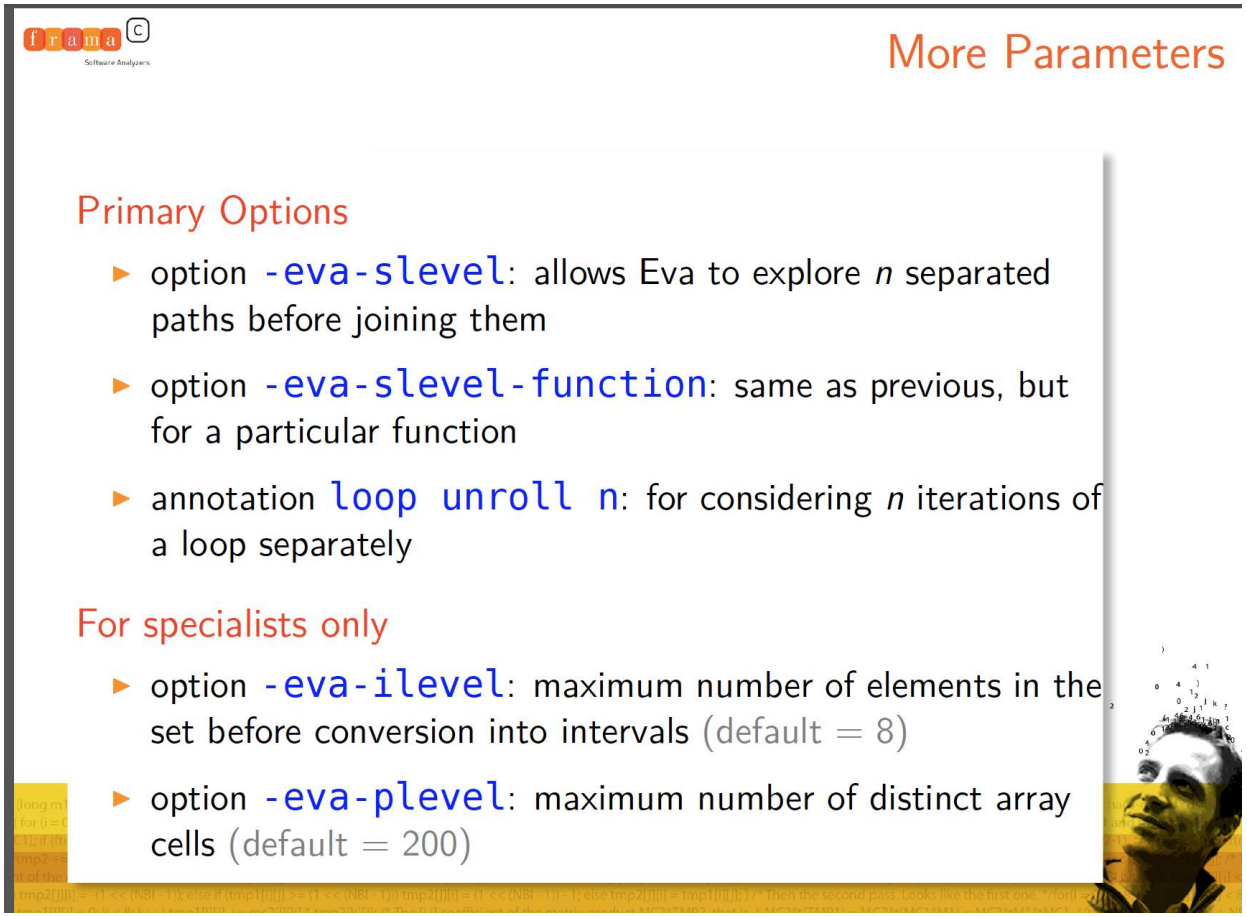
- Variants

x264 behavior (different outputs, execution time, etc.)

- Large variability spaces

Dozens of options (Boolean, categorical, numeric)

How to configure frama-c?



How to ensure that all Linux kernel configurations build?

Enormous configurations space eg Linux has 15K+ options, tri-state values {y, n, m}. A build takes 15 minutes on average on a recent machine

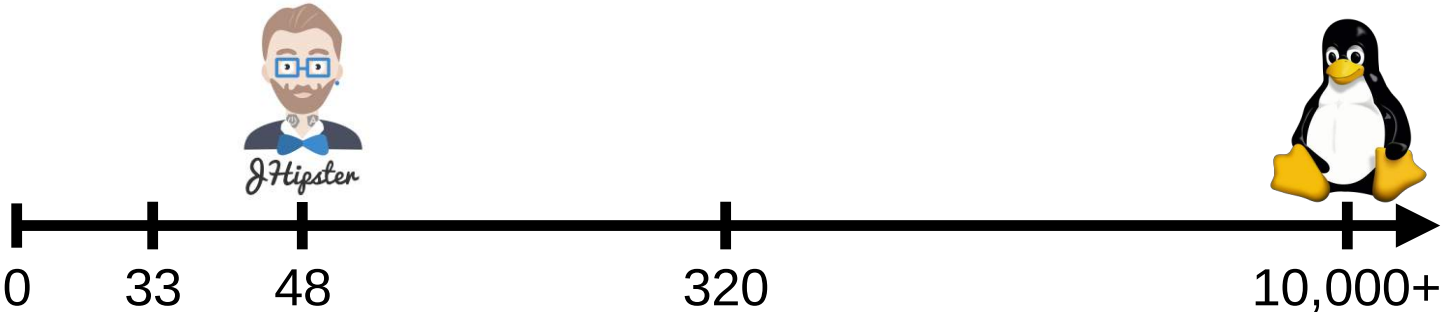
KConfig file

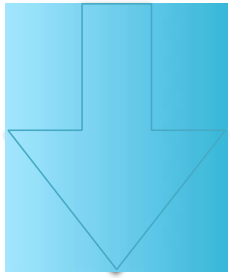
```
[...]

config PRINTK
	default y
	bool "Enable support for printk" if EXPERT
	select IRQ_WORK
	help
	  This option enables normal printk support. Removing it
	  eliminates most of the message strings from the kernel image
	  and makes the kernel more or less silent. As this makes it
	  very difficult to diagnose system problems, saying N here is
	  strongly discouraged.

config PRINTK_NMI
	def_bool y
	depends on PRINTK
	depends on HAVE_NMI

config BUG
	bool "BUG() support" if EXPERT
	default y
	help
	  Disabling this option eliminates support for BUG and WARN, reducing
	  the size of your kernel image and potentially quietly ignoring
	  numerous fatal conditions. You should only consider disabling this
	  option for embedded systems with no facilities for reporting errors.
	  Just say Y.

config ELF_CORE
	depends on COREDUMP
	default y
	bool "Enable ELF core dumps" if EXPERT
	help
	  Enable support for generating core dumps. Disabling saves about 4k.

[...]

config AIO
	bool "Enable AIO support" if EXPERT
	default y
	help
	  This option enables POSIX asynchronous I/O which may by used
	  by some high performance threaded applications. Disabling
	  this option saves about 7k.

[...]
```

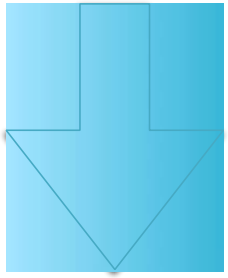
“According to several studies, configuration failures represent one of the most common types of software failures”

Mohammed Sayagh, Noureddine Kerzazi, Bram Adams, and Fabio Petrillo. 2018. Software Configuration Engineering in Practice: Interviews, Survey, and Systematic Literature Review. IEEE Transactions on Software Engineering (2018).

A Universe of Options







#Variants (independent Boolean options)

2^{33}

2^{320}

Exercice

- Give examples of real-world systems with (large|complex)+ variability spaces
- How variability is expressed?
- What’s the underlying problems for developers?
- What’s the underlying problems for users?

Modeling variability

Why? How?

Jhipster case study

Agenda

- VaryLaTeX (20’)
- Software Variability: An Overview (10’)
- **AI1: Modeling and Reasoning about Variability (20’)**
  - **Jhipster case study**
  - **feature models: syntax, semantics, and logics**
- AI2: Learning Variability (20’)
  - x264 case study
  - Sampling, measuring learning
- Back to VaryLaTeX (20’)

Case study: JHipster

- Web-apps generator
  - Spring-Boot
  - Bootstrap / AngularJS
  - 100 % Open Source
- Yeoman
  - Bower, npm
  - yo
- Used all over the world
  - Large companies (HBO, Google, Adobe)[1]
  - Independent developers
  - Our students
- GitHub
  - 6000+ stars
  - 118 releases (JHipster 3.6.1, 18 Aug 2016)
  - 300+ contributors

[1] https://jhipster.github.io/companies-using-jhipster/

```
macher-wifi:getting-started macher1$ yo jhipster


I'm all done. Running npm install & bower install for you to install the required dependencies.


HIPSTER STACK
     FOR
  JAVA DEVS


Welcome to the JHipster Generator v2.17.0

? (1/15) What is the base name of your application? jhipster
? (2/15) What is your default Java package name? com.mycompany.myapp
? (3/15) Do you want to use Java 8? Yes (use Java 8)
? (4/15) Which *type* of authentication would you like to use? (Use arrow keys)
❯ HTTP Session Authentication (stateful, default Spring Security mechanism)
  OAuth2 Authentication (stateless, with an OAuth2 server implementation)
  Token-based authentication (stateless, with a token)
```

jdubois 2 days ago Use Spring Boot's configuration meta-data

9 contributors 


```
1  package <%=packageName%>.config;
2  <% if (databaseType == 'sql') { %>
3  import <%=packageName%>.config.liquibase.AsyncSpringLiquibase;
4  import com.codahale.metrics.MetricRegistry;
5  import com.fasterxml.jackson.datatype.hibernate4.Hibernate4Module;
6  import com.zaxxer.hikari.HikariConfig;
7  import com.zaxxer.hikari.HikariDataSource;
8  import liquibase.integration.spring.SpringLiquibase;<% } %><% if (databaseType == 'mongodb' && authenticationType == 'oauth2') { %
9  import <%=packageName%>.config.oauth2.OAuth2AuthenticationReadConverter;<% } %><% if (databaseType == 'mongodb') { %>
10 import com.mongodb.Mongo;
11 import org.mongeez.Mongeez;<% } %>
12 import org.slf4j.Logger;
13 import org.slf4j.LoggerFactory;<% if (databaseType == 'sql') { %><% if (hibernateCache == 'hazelcast') { %>
14 import org.springframework.cache.CacheManager;<% } %>
15 import org.springframework.beans.factory.annotation.Autowired;
16 import org.springframework.boot.autoconfigure.condition.ConditionalOnExpression;<% } %><% if (databaseType == 'mongodb') { %>
17 import org.springframework.boot.autoconfigure.mongo.MongoAutoConfiguration;
18 import org.springframework.boot.autoconfigure.mongo.MongoProperties;<% } %><% if (databaseType == 'sql') { %>
19 import org.springframework.boot.autoconfigure.jdbc.DataSourceProperties;
20 import org.springframework.boot.autoconfigure.liquibase.LiquibaseProperties;
21 import org.springframework.context.ApplicationContextException;<% } %>
22 import org.springframework.context.annotation.Bean;
23 import org.springframework.context.annotation.Configuration;
24 import org.springframework.context.annotation.Profile;<% if (databaseType == 'mongodb') { %>
25 import org.springframework.context.annotation.Import;<% } %><% if (databaseType == 'sql') { %>
26 import org.springframework.core.env.Environment;<% } %><% if (databaseType == 'mongodb' && authenticationType == 'oauth2') { %>
27 import org.springframework.core.convert.converter.Converter;<% } %><% if (databaseType == 'mongodb') { %>
28 import org.springframework.core.io.ClassPathResource;<% } %><% if (searchEngine == 'elasticsearch') { %>
29 import org.springframework.data.elasticsearch.repository.config.EnableElasticsearchRepositories;<% } %><% if (databaseType == 'mon
30 import org.springframework.data.mongodb.config.AbstractMongoConfiguration;
31 import org.springframework.data.mongodb.config.EnableMongoAuditing;<% } %><% if (databaseType == 'mongodb' && authenticationType =
32 import org.springframework.data.mongodb.core.convert.CustomConversions;<% } %><% if (databaseType == 'mongodb') { %>
33 import org.springframework.data.mongodb.core.mapping.event.ValidatingMongoEventListener;
34 import org.springframework.data.mongodb.repository.config.EnableMongoRepositories;
35 import org.springframework.validation.beanvalidation.LocalValidatorFactoryBean;<% } %><% if (databaseType == 'sql') { %>
```

```
<%_ if (databaseType === 'mongodb' || databaseType === 'couchbase') { _%>
@Profile("!" + JHipsterConstants.SPRING_PROFILE_CLOUD)
<%_ } _%>
<%_ if (databaseType === 'mongodb') { _%>
    <%_ if (reactive) { _%>
@Import(value = {MongoAutoConfiguration.class, MongoReactiveAutoConfiguration.class})
    <%_ } else { _%>
@Import(value = MongoAutoConfiguration.class)
    <%_ } _%>
@EnableMongoAuditing(auditorAwareRef = "springSecurityAuditorAware")
<%_ } _%>
<%_ if (databaseType === 'couchbase') { _%>
@EnableCouchbaseRepositories(repositoryBaseClass = CustomN1qlCouchbaseRepository.class, basePackages = "<%=packageName%>.repository")
    <%_ if (reactive) { _%>
@EnableReactiveCouchbaseRepositories(repositoryBaseClass = CustomReactiveN1qlCouchbaseRepository.class, basePackages = "<%=packageName%>.re
    <%_ } _%>
@Import(value = CouchbaseAutoConfiguration.class)
@EnableCouchbaseAuditing(auditorAwareRef = "springSecurityAuditorAware")
<%_ } _%>
public class DatabaseConfiguration {

    private final Logger log = LoggerFactory.getLogger(DatabaseConfiguration.class);
<%_ if (databaseType === 'sql') { _%>
    <%_ if (devDatabaseType === 'h2Disk' || devDatabaseType === 'h2Memory') { _%>

    private final Environment env;

    public DatabaseConfiguration(Environment env) {
        this.env = env;
    }

```

```
<%_ if (applicationType === 'gateway' && authenticationType === 'uaa') { _%>
        <dependency>
            <groupId>org.apache.httpcomponents</groupId>
            <artifactId>httpclient</artifactId>
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'hazelcast') { _%>
        <dependency>
            <groupId>com.hazelcast</groupId>
            <artifactId>hazelcast</artifactId>
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'hazelcast' && enableHibernateCache) { _%>
        <dependency>
            <groupId>com.hazelcast</groupId>
            <artifactId>hazelcast-hibernate53</artifactId>
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'hazelcast') { _%>
        <dependency>
            <groupId>com.hazelcast</groupId>
            <artifactId>hazelcast-spring</artifactId>
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'infinispan') { _%>
        <dependency>
```

Variability crosscuts all artefacts

(40 languages are used in a contemporary Web app)

```
<%_ if (applicationType === 'gateway' && authenticationType === 'uaa') { _%>
        <dependency>
            <groupId>org.apache.httpcomponents</groupId>
            <artifactId>httpclient</artifactId>
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'hazelcast') { _%>
        <dependency>
            <groupId>com.hazelcast</groupId>
            <artifactId>hazelcast</artifactId>
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'hazelcast' && enableHibernateCache) { _%>
        <dependency>
            <groupId>com.hazelcast</groupId>
            <artifactId>hazelcast-hibernate53</artifactId>
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'hazelcast') { _%>
        <dependency>
            <groupId>com.hazelcast</grou
            <artifactId>hazelcast-spring
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'infinispan')
        <dependency>
```

Branch: master

generator-jhipster / app / templates / src / main / java / package / config / _DatabaseConfiguration.java

jdubois 2 days ago Use Spring Boot's configuration meta-data

9 contributors

184 lines (165 sloc) 9.69 KB Raw Blame History

```
package <%=packageName%>.config;
<% if (databaseType == 'sql') { %>
import <%=packageName%>.config.liquibase.AsyncSpringLiquibase;
import com.codahale.metrics.MetricRegistry;
import com.fasterxml.jackson.datatype.hibernate4.Hibernate4Module;
import com.zaxxer.hikari.HikariConfig;
import com.zaxxer.hikari.HikariDataSource;
import liquibase.integration.spring.SpringLiquibase;<% } %><% if (databaseType == 'mongodb' && authenticationType == 'oauth2') { %
import <%=packageName%>.config.oauth2.OAuth2AuthenticationReadConverter;<% } %><% if (databaseType == 'mongodb') { %>
import com.mongodb.Mongo;
import org.mongeez.Mongeez;<% } %>
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;<% if (databaseType == 'sql') { %><% if (hibernateCache == 'hazelcast') { %>
import org.springframework.cache.CacheManager;<% } %>
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.autoconfigure.condition.ConditionalOnExpression;<% } %><% if (databaseType == 'mongodb') { %>
import org.springframework.boot.autoconfigure.mongo.MongoAutoConfiguration;
import org.springframework.boot.autoconfigure.mongo.MongoProperties;<% } %><% if (databaseType == 'sql') { %>
import org.springframework.boot.autoconfigure.jdbc.DataSourceProperties;
import org.springframework.boot.autoconfigure.liquibase.LiquibaseProperties;
import org.springframework.context.ApplicationContextException;<% } %>
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.context.annotation.Profile;<% if (databaseType == 'mongodb') { %>
import org.springframework.context.annotation.Import;<% } %><% if (databaseType == 'sql') { %>
import org.springframework.core.env.Environment;<% } %><% if (databaseType == 'mongodb' && authenticationType == 'oauth2') { %>
import org.springframework.core.convert.converter.Converter;<% } %><% if (databaseType == 'mongodb') { %>
import org.springframework.core.io.ClassPathResource;<% } %><% if (searchEngine == 'elasticsearch') { %>
import org.springframework.data.elasticsearch.repository.config.EnableElasticsearchRepositories;<% } %><% if (databaseType == 'mon
import org.springframework.data.mongodb.config.AbstractMongoConfiguration;
import org.springframework.data.mongodb.config.EnableMongoAuditing;<% } %><% if (databaseType == 'mongodb' && authenticationType =
import org.springframework.data.mongodb.core.convert.CustomConversions;<% } %><% if (databaseType == 'mongodb') { %>
import org.springframework.data.mongodb.core.mapping.event.ValidatingMongoEventListener;
import org.springframework.data.mongodb.repository.config.EnableMongoRepositories;
import org.springframework.validation.beanvalidation.LocalValidatorFactoryBean;<% } %><% if (databaseType == 'sql') { %>
```

```
<%_ if (useKafka) { _%>
    zookeeper:
        extends:
            file: kafka.yml
            service: zookeeper
    kafka:
        extends:
            file: kafka.yml
            service: kafka
<%_ } _%>
<%_ if (serviceDiscoveryType === 'eureka') { _%>
    jhipster-registry:
        extends:
            file: jhipster-registry.yml
            service: jhipster-registry
<%_ } _%>
<%_ if (serviceDiscoveryType === 'consul') { _%>
    consul:
        extends:
            file: consul.yml
            service: consul
    consul-config-loader:
        extends:
            file: consul.yml
            service: consul-config-loader
<%_ } _%>
<%_ if (gatewayType === 'traefik') { _%>
    traefik:
        extends:
            file: traefik.yml
            service: traefik
<%_ } _%>
<%_ if (monitoring === 'elk') { _%>

    jhipster-elasticsearch:
        extends:
            file: jhipster-console.yml
            service: jhipster-elasticsearch
    jhipster-logstash:
```

configurator

```
{
    when: function (response) {
        return response.databaseType === 'sql';
    },
    type: 'list',
    name: 'hibernateCache',
    message: function (response) {
        return getNumberedQuestion('Do you want to use Hibernate 2nd level cache?', response.databaseType === 'sql');
    },
    choices: [
        {
            value: 'no',
            name: 'No'
        },
        {
            value: 'ehcache',
            name: 'Yes, with ehcache (local cache, for a single node)'
        },
        {
            value: 'hazelcast',
            name: 'Yes, with HazelCast (distributed cache, for multiple nodes)'
        }
    ],
    default: (applicationType === 'gateway' || applicationType === 'microservice' || applicationType === 'uaa') ? 2 : 1
},
```

```
? (7/15) Do you want to use Hibernate 2nd level cache?
  No
  Yes, with ehcache (local cache, for a single node)
❯ Yes, with HazelCast (distributed cache, for multiple nodes)
```

Software Variability

Software is working (sometimes)

- yes but perhaps for one specific configuration (the default one)
- is it working for **all configurations**?

```
? (3/15) Which *type* of authentication would you like to use? (Use arrow keys)
❯ HTTP Session Authentication (stateful, default Spring Security mechanism)
  HTTP Session Authentication with social login enabled (Google, Facebook, Twitter).
  OAuth2 Authentication (stateless, with an OAuth2 server implementation)
  Token-based authentication (stateless, with a token)
```

```
? (7/15) Do you want to use Hibernate 2nd level cache?
  No
  Yes, with ehcache (local cache, for a single node)
❯ Yes, with HazelCast (distributed cache, for multiple nodes)
```

Static analysis of variability across all artefacts is hard + combinatorial explosion

```
<%_ if (applicationType === 'gateway' && authenticationType === 'uaa') { _%>
        <dependency>
            <groupId>org.apache.httpcomponents</groupId>
            <artifactId>httpclient</artifactId>
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'hazelcast') { _%>
        <dependency>
            <groupId>com.hazelcast</groupId>
            <artifactId>hazelcast</artifactId>
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'hazelcast' && enableHibernateCache) { _%>
        <dependency>
            <groupId>com.hazelcast</groupId>
            <artifactId>hazelcast-hibernate53</artifactId>
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'hazelcast') { _%>
        <dependency>
            <groupId>com.hazelcast</grou
            <artifactId>hazelcast-spring
        </dependency>
<%_ } _%>
<%_ if (cacheProvider === 'infinispan')
        <dependency>
```

Branch: master

generator-jhipster / app / templates / src / main / java / package / config / _DatabaseConfiguration.java

jdubois 2 days ago Use Spring Boot's configuration meta-data

9 contributors

184 lines (165 sloc) 9.69 KB Raw Blame History

```
package <%=packageName%>.config;
<% if (databaseType == 'sql') { %>
import <%=packageName%>.config.liquibase.AsyncSpringLiquibase;
import com.codahale.metrics.MetricRegistry;
import com.fasterxml.jackson.datatype.hibernate4.Hibernate4Module;
import com.zaxxer.hikari.HikariConfig;
import com.zaxxer.hikari.HikariDataSource;
import liquibase.integration.spring.SpringLiquibase;<% } %><% if (databaseType == 'mongodb' && authenticationType == 'oauth2') { %
import <%=packageName%>.config.oauth2.OAuth2AuthenticationReadConverter;<% } %><% if (databaseType == 'mongodb') { %>
import com.mongodb.Mongo;
import org.mongeez.Mongeez;<% } %>
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;<% if (databaseType == 'sql') { %><% if (hibernateCache == 'hazelcast') { %>
import org.springframework.cache.CacheManager;<% } %>
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.autoconfigure.condition.ConditionalOnExpression;<% } %><% if (databaseType == 'mongodb') { %>
import org.springframework.boot.autoconfigure.mongo.MongoAutoConfiguration;
import org.springframework.boot.autoconfigure.mongo.MongoProperties;<% } %><% if (databaseType == 'sql') { %>
import org.springframework.boot.autoconfigure.jdbc.DataSourceProperties;
import org.springframework.boot.autoconfigure.liquibase.LiquibaseProperties;
import org.springframework.context.ApplicationContextException;<% } %>
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.context.annotation.Profile;<% if (databaseType == 'mongodb') { %>
import org.springframework.context.annotation.Import;<% } %><% if (databaseType == 'sql') { %>
import org.springframework.core.env.Environment;<% } %><% if (databaseType == 'mongodb' && authenticationType == 'oauth2') { %>
import org.springframework.core.convert.converter.Converter;<% } %><% if (databaseType == 'mongodb') { %>
import org.springframework.core.io.ClassPathResource;<% } %><% if (searchEngine == 'elasticsearch') { %>
import org.springframework.data.elasticsearch.repository.config.EnableElasticsearchRepositories;<% } %><% if (databaseType == 'mon
import org.springframework.data.mongodb.config.AbstractMongoConfiguration;
import org.springframework.data.mongodb.config.EnableMongoAuditing;<% } %><% if (databaseType == 'mongodb' && authenticationType =
import org.springframework.data.mongodb.core.convert.CustomConversions;<% } %><% if (databaseType == 'mongodb') { %>
import org.springframework.data.mongodb.core.mapping.event.ValidatingMongoEventListener;
import org.springframework.data.mongodb.repository.config.EnableMongoRepositories;
import org.springframework.validation.beanvalidation.LocalValidatorFactoryBean;<% } %><% if (databaseType == 'sql') { %>
```

```
<%_ if (useKafka) { _%>
    zookeeper:
        extends:
            file: kafka.yml
            service: zookeeper
    kafka:
        extends:
            file: kafka.yml
            service: kafka
<%_ } _%>
<%_ if (serviceDiscoveryType === 'eureka') { _%>
    jhipster-registry:
        extends:
            file: jhipster-registry.yml
            service: jhipster-registry
<%_ } _%>
<%_ if (serviceDiscoveryType === 'consul') { _%>
    consul:
        extends:
            file: consul.yml
            service: consul
    consul-config-loader:
        extends:
            file: consul.yml
            service: consul-config-loader
<%_ } _%>
<%_ if (gatewayType === 'traefik') { _%>
    traefik:
        extends:
            file: traefik.yml
            service: traefik
<%_ } _%>
<%_ if (monitoring === 'elk') { _%>

    jhipster-elasticsearch:
        extends:
            file: jhipster-console.yml
            service: jhipster-elasticsearch
    jhipster-logstash:
```

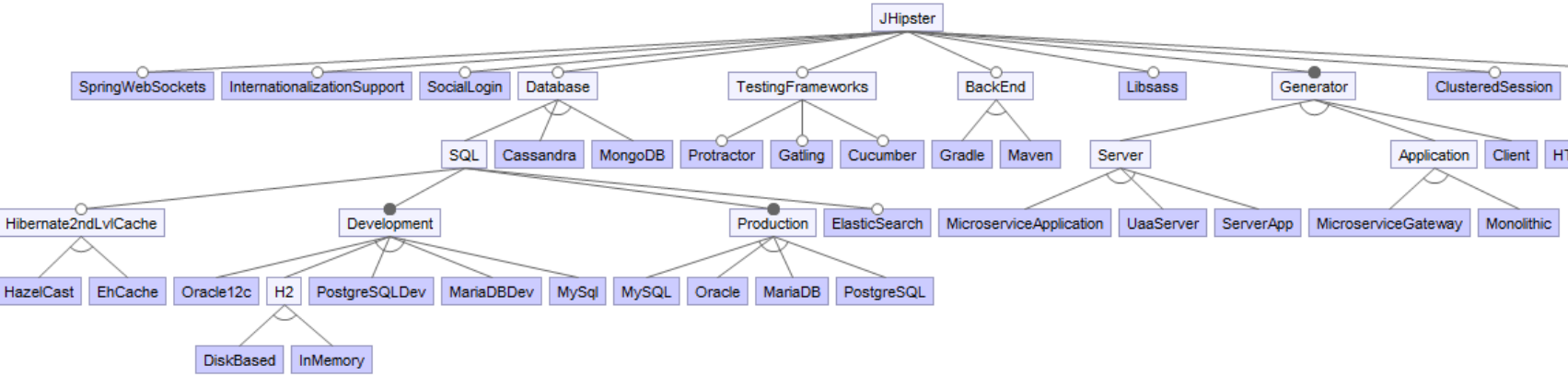
Testing?

Testing with the community

ALL

Sampling

Testing? Yes, but which configurations?



$$OAuth2 \wedge \neg SocialLogin \wedge \neg MicroserviceApplication \Rightarrow SQL \vee MongoDB$$
$$SocialLogin \Rightarrow (HTTPSession \vee JWT) \wedge (ServerApp \vee Monolithic) \wedge (SQL \vee MongoDB)$$
$$UaaServer \Rightarrow Uaa$$
$$Oracle \Rightarrow H2 \vee Oracle12c$$
$$\neg OAuth2 \wedge \neg SocialLogin \wedge \neg MicroserviceApplication \Rightarrow SQL \vee MongoDB \vee Cassandra$$
$$Server \Rightarrow \neg Protractor$$
$$MySQL \Rightarrow H2 \vee MySql$$
$$MicroserviceApplication \vee MicroserviceGateway \Rightarrow JWT \vee Uaa$$
$$Monolithic \Rightarrow JWT \vee HTTPSession \vee OAuth2$$
$$MariaDB \Rightarrow H2 \vee MariaDBDev$$
$$PostgreSQL \Rightarrow H2 \vee PostgreSQLDev$$
$$Server \vee Application \Rightarrow BackEnd \wedge Authentication$$
$$SpringWebSockets \vee ClusteredSession \Rightarrow Application$$
$$Client \Rightarrow \neg Gatling \wedge \neg Cucumber \wedge \neg BackEnd \wedge \neg Authentication$$
$$Libsass \Rightarrow Application \vee Client$$

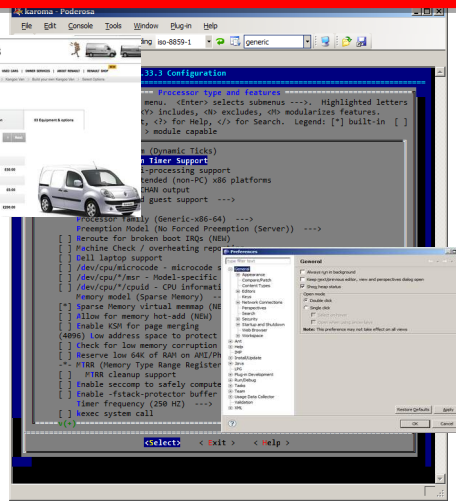
Legend:
- Mandatory
- Optional
- Alternative
- Abstract
- Concrete

Sampling

Modeling is needed!

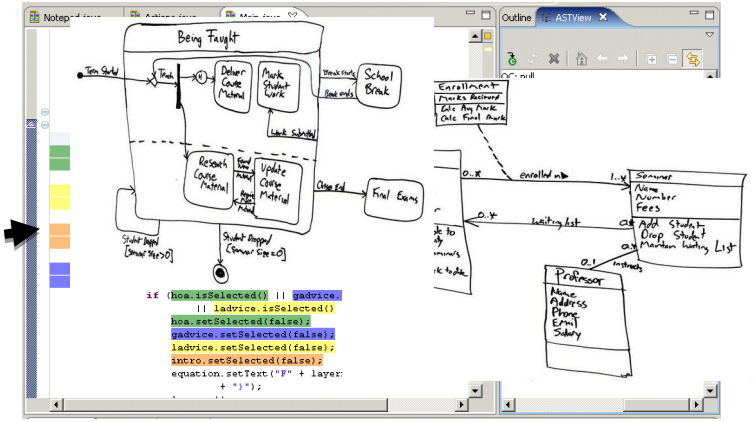
Modeling is needed
Feature model in action!
(demo)

```
{
    when: function (response) {
        return response.databaseType === 'sql';
    },
    type: 'list',
    name: 'hibernateCache',
    message: function (response) {
        return getNumberedQuestion('Do you want to use Hibernate 2nd level cache?', response.databaseType === 'sql');
    },
    choices: [
        {
            value: 'no',
            name: 'No'
        },
        {
            value: 'ehcache',
            name: 'Yes, with ehcache (local cache, for a single node)'
        },
        {
            value: 'hazelcast',
            name: 'Yes, with HazelCast (distributed cache, for multiple nodes)'
        }
    ],
    default: (applicationType === 'gateway' || applicationType === 'microservice' || applicationType === 'uaa') ? 2 : 1
},
```



Variability Model

mapping



Base Artefacts (e.g., models)

Configuration

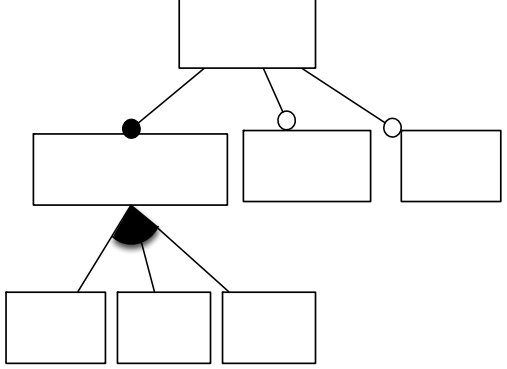
Software Generator

(derivation engine)


Unused flexibility

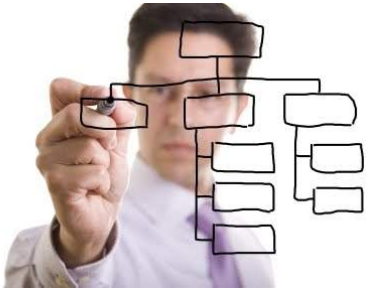

Illegal variant

Feature Model



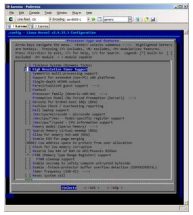
not, and, or, implies

Communicative

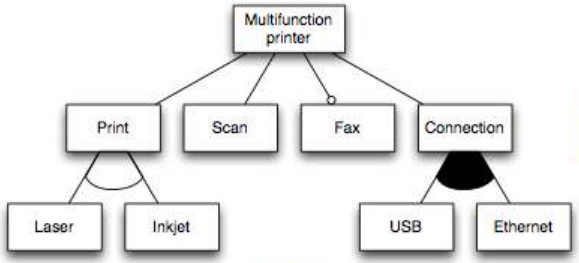


Analytic

Generative



Typical implementations



result

logics

solvers

- We have tested <u>all</u> configurations of an industrial-strength, open-source generator (Jhipster)
- 26K+ configurations, 4376 hours/machine, 8 man/month
- “Ground truth” allows us to *precisely* assess sampling

  36% failures explained by 6 feature interactions (faults)

- What is the most cost-effective sampling strategy?
  - T-wise or dissimilarity are very effective
  - **with “only” 126 configurations you can detect all 6 most important faults**

Axel Halin, Alexandre Nuttinck, Mathieu Acher, Xavier Devroey, Gilles Perrouin, Benoit Baudry.
Test them all, is it worth it? Assessing configuration sampling on the JHipster Web development stack (2018). In Empirical Software Engineering journal

Agenda

- VaryLaTeX (20’)
- Software Variability: An Overview (10’)
- AI1: Modeling and Reasoning about Variability (20’)
  - Jhipster case study
  - feature models: syntax, semantics, and logics
- **AI2: Learning Variability (20’)**
  - **x264 case study**
  - **Sampling, measuring learning**
- Back to VaryLaTeX (20’)

Software users:

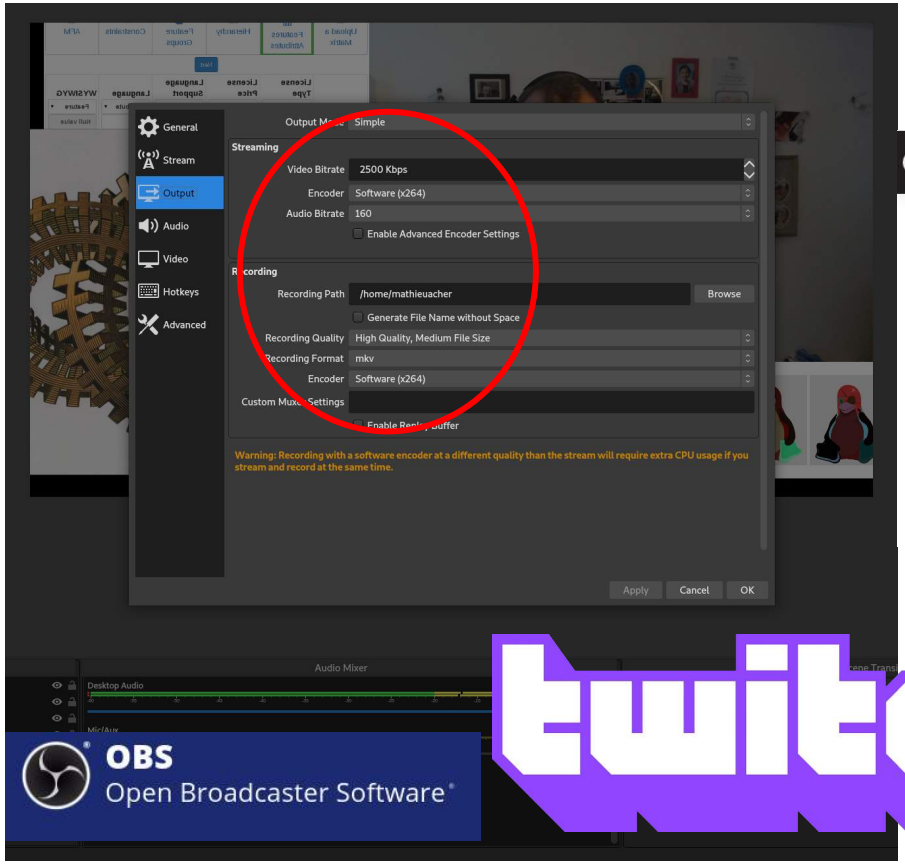
How to choose the configuration that fits my requirements?

```
    --psy-rd <float:float>  Strength of psychovisual optimization ["1.0:0.0"]
                                #1: RD (requires subme>=6)
                                #2: Trellis (requires trellis, experimental)
    --no-8x8dct             Disable adaptive spatial transform size
-t, --trellis <integer>     Trellis RD quantization. [1]
                                - 0: disabled
                                - 1: enabled only on the final encode of a MB
                                - 2: enabled on all mode decisions
    --nr <integer>          Noise reduction [0]
    --cqmfile <string>      Read custom quant matrices from a JM-compatible file

Input/Output:

-o, --output <string>       Specify output file
    --muxer <string>        Specify output container format ["auto"]
                                - auto, raw, mkv, flv
    --demuxer <string>      Specify input container format ["auto"]
                                - auto, raw, y4m, avs
    --input-fmt <string>    Specify input file format (requires lavf support)
    --input-csp <string>    Specify input colorspace format for raw input
    --output-csp <string>   Specify output colorspace ["i420"]
                                - i420, i422, i444, rgb
    --input-depth <integer> Specify input bit depth for raw input
    --input-range <string>  Specify input color range ["auto"]
                                - auto, tv, pc
    --input-res <intxint>   Specify input resolution (width x height)
    --index <string>        Filename for input index file
    --sar width:height      Specify Sample Aspect Ratio
    --fps <float|rational>  Specify framerate
    --seek <integer>        First frame to encode
    --frames <integer>      Maximum number of frames to encode
    --level <string>        Specify level (as defined by Annex A)
    --bluray-compat         Enable compatibility hacks for Blu-ray support
    --avcintra-class <integer> Use compatibility hacks for AVC-Intra class
                                - 50, 100, 200
    --stitchable            Don't optimize headers based on video content
                            Ensures ability to recombine a segmented encode
```

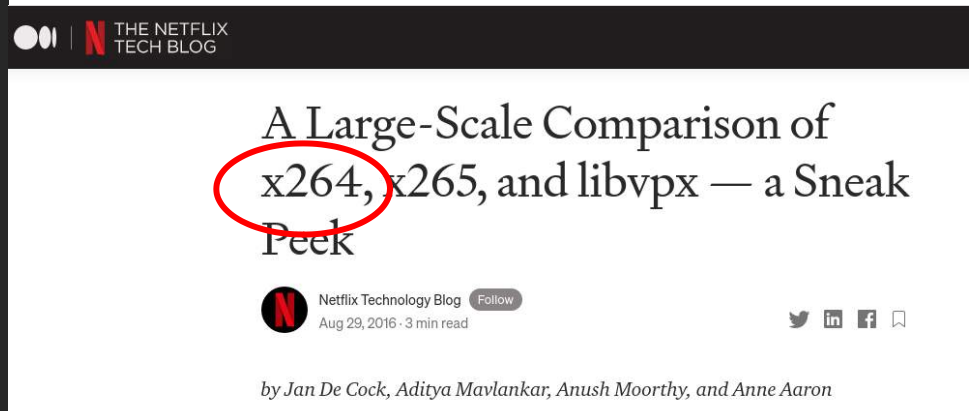
x264 --longhelp | wc -l

176

Claim: x264 is never optimally configured
causing incredible waste in CPU and energy consumption worldwide



twitch



NETFLIX

x264 video encoder

Open-source software library & a command-line utility for encoding video streams into H.264/MPEG-4

- Implementing a large number of features
- Highly configurable with dozens of command line options

```
x264 --bframes 2 --ref 3 --cabac "SourceVideo.mpg" 640x480
```

x264 video encoder

```
~ x264 DiverSE-meeting.mp4 -o meeting13.webm
encoded 32295 frames, 96.03 fps, 233.05 kb/s
~ ls -lh meeting13.webm
113M May 11 09:48 meeting13.webm
```

```
~ x264 --bframes 2 --ref 6 --no-cabac DiverSE-meeting.mp4 -o meeting13.webm
encoded 32295 frames, 80.94 fps, 266.82 kb/s
~ ls -lh meeting13.webm
129M May 11 09:56 meeting13.webm
```

```
~ x264 --bframes 1 --ref 3 --cabac DiverSE-meeting.mp4 -o meeting13.webm
```

?????

Configuration is hard: numerous options, informal knowledge

```
~ x264 --bframes 1 --ref 3 --cabac DiverSE-meeting.mp4 -o meeting13.webm
```

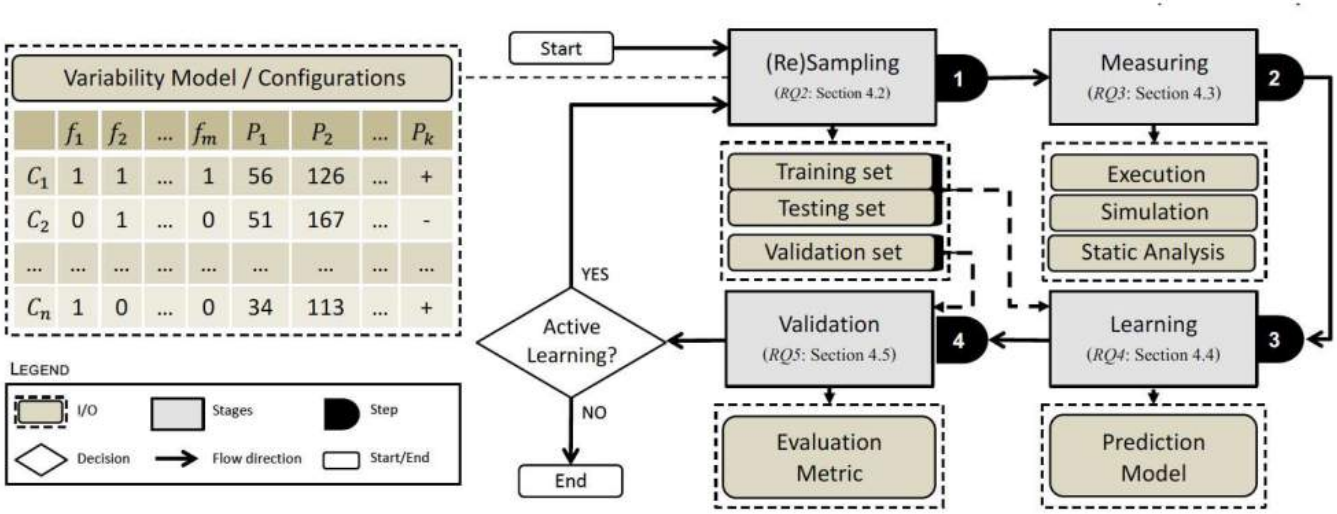
?????

```
mathieuacher localhost.localdomain ~ x264 --fullhelp | wc -l
480
```

```
      Lossless:
            x264 --qp 0 -o <output> <input>

      Maximum PSNR at the cost of speed and visual quality:
            x264 --preset placebo --tune psnr -o <output> <input>

      Constant bitrate at 1000kbps with a 2 second-buffer:
            x264 --vbv-bufsize 2000 --bitrate 1000 -o <output> <input>

Presets:

      --profile <string>      Force the limits of an H.264 profile
                                  Overrides all settings.
                                  - baseline, main, high, high10, high422, high4
      --preset <string>       Use a preset to select encoding settings [medium]
                                  Overridden by user settings.
                                  - ultrafast,superfast,veryfast,faster,fast
                                  - medium,slow,slower,veryslow,placebo
      --tune <string>         Tune the settings for a particular type of source
                              or situation
                                  Overridden by user settings.
                                  Multiple tunings are separated by commas.
                                  Only one psy tuning can be used at a time.
```

```
Frame-type options:

  -I, --keyint <integer or "infinite"> Maximum GOP size [250]
      --tff                    Enable interlaced mode (top field first)
      --bff                    Enable interlaced mode (bottom field first)
      --pulldown <string>      Use soft pulldown to change frame rate
                                   - none, 22, 32, 64, double, triple, euro (requires cfr input)

Ratecontrol:

  -B, --bitrate <integer>      Set bitrate (kbit/s)
      --crf <float>            Quality-based VBR (-12-51) [23.0]
      --vbv-maxrate <integer>  Max local bitrate (kbit/s) [0]
      --vbv-bufsize <integer>  Set size of the VBV buffer (kbit) [0]
  -p, --pass <integer>         Enable multipass ratecontrol
                                   - 1: First pass, creates stats file
                                   - 2: Last pass, does not overwrite stats file
```

Sampling, Measuring, Learning



Learning Software Configuration Spaces: A Systematic Literature Review
Juliana Alves Pereira, Hugo Martin, Mathieu Acher, Jean-Marc Jézéquel, Goetz Botterweck, Anthony Ventresque https://arxiv.org/abs/1906.03018

Performance Prediction

$f : \mathbb{C} \rightarrow \mathbb{R}^+$ the function affecting to any configuration $c \in \mathbb{C}$ its performance $f(c) \in \mathbb{R}^+$,

trailer_2k_480p24.y4m

no_8x8dct	no_asm	no_cabac	no_deblock	no_fast_pskip	no_mbtree	no_mixed_refs	no_weightb	rc_lookahead	ref	size	elapsedtime
True	False	False	True	True	False	True	True	20	9	1718492	3.444
True	False	True	False	True	False	False	True	40	9	1962957	4.744
True	False	False	True	False	True	True	False	40	1	3657562	2.427
True	False	True	False	True	True	True	False	40	9	3436410	3.447
False	False	False	True	False	False	True	False	60	5	2066645	2.957

Regression problem (linear regression, regression tree, random forest, gradient boosting, SVM, etc.)

Guo et al. ASE 2013, Apel et al. ASE'15, Czarnecki et al. SPLC'15, Siegmund et al. FSE'15, Kastner et al. ASE'17, Menzies et al. FSE'17, Batory et al. FSE'17

Performance prediction of configurable software systems

$f : \mathbb{C} \rightarrow \mathbb{R}^+$ the function affecting to any configuration $c \in \mathbb{C}$ its performance $f(c) \in \mathbb{R}^+$,

- Learning/**training** phase
  - **sampling** configurations
  - **measuring** each configuration
- Non-functional properties
  - Encoding time, FPS, CPU, Bitrate, encoding size…


```
x264 --bframes 1 --ref 2 --cabac "SourceVideo.mpg" 640x480
x264 --bframes 1 --ref 2 --no-cabac "SourceVideo.mpg" 640x480
x264 --bframes 1 --ref 2 --mbtree "SourceVideo.mpg" 640x480
x264 --bframes 1 --ref 2 --no-mbtree "SourceVideo.mpg" 640x480
…
```

- Then *predict* performances of a new configuration
- **Work remarkably well with statistical learning able to capture interactions among options**

But...

```
? x264 --bframes 1 --mbtree --cabac "SourceVideo.mpg" 640x480
-> x encoding time, y FPS, z CPU usage…
```

Demo!

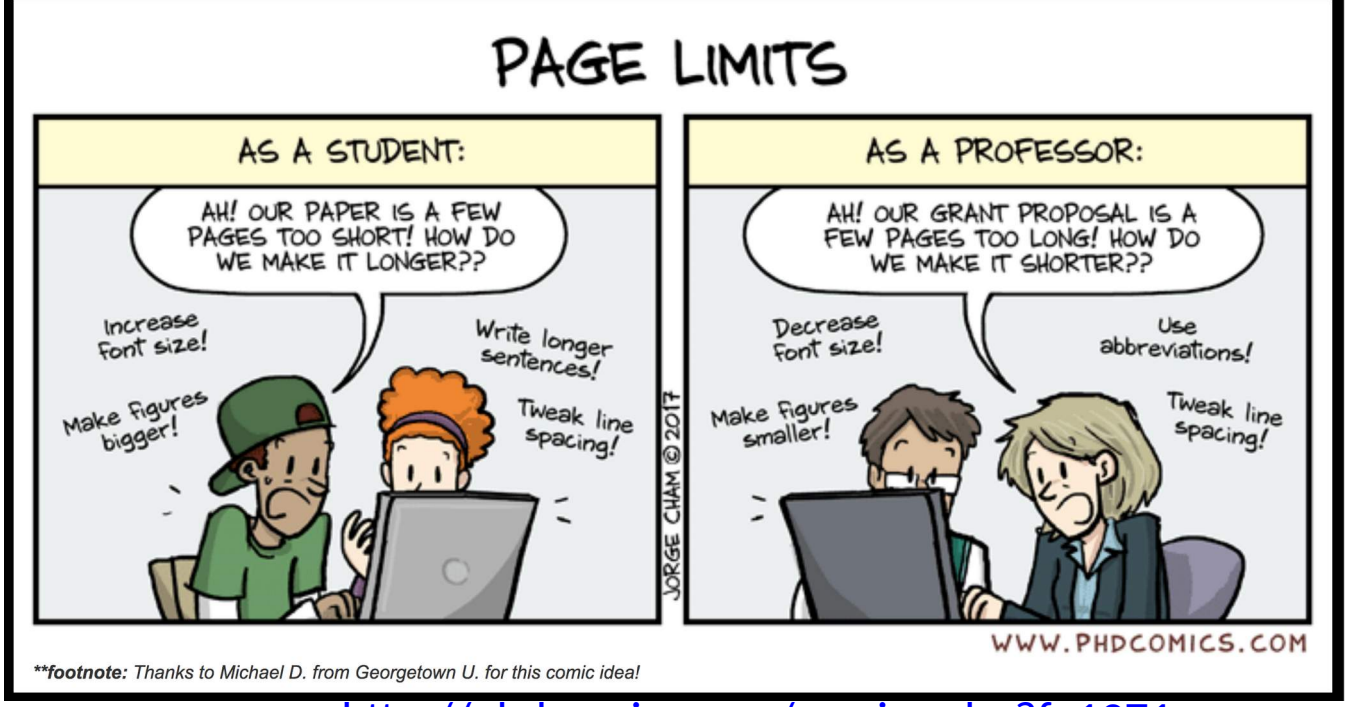
https://github.com/VaryVary/ML-configurable-SPLCTutorial/tree/master/x264

Agenda

- VaryLaTeX (20’)
- Software Variability: An Overview (10’)
- AI1: Modeling and Reasoning about Variability (20’)
  - Jhipster case study
  - feature models: syntax, semantics, and logics
- AI2: Learning Variability (20’)
  - x264 case study
  - Sampling, measuring learning
- **Back to VaryLaTeX (20’)**

Modeling Variability

- Very large variability spaces
- **AI#1 Abstraction/languages to formally and efficiently reason about configuration spaces**
  - with SAT/CSP/SMT solvers
  - Eg constrained sampling
- Variability Models
  - Elaborated by humans
  - Reverse engineered from existing artefacts/systems
  - Promise: sound and complete representation of the configuration space



http://phdcomics.com/comics.php?f=1971

VaryLaTeX: Learning Paper Variants That Meet Constraints

Mathieu Acher
Paul Temple
Jean-Marc Jézéquel
Univ Rennes, Inria, CNRS, IRISA
Rennes, France
mathieu.acher@irisa.fr

José A. Galindo
University of Sevilla
Sevilla, Spain
jagalindo@us.es

Jabier Martinez
Tewfik Ziadi
Sorbonne University UPMC
Paris, France
jabier.martinez@lip6.fr

Successfully submitted for VaMoS'18

(on time and meeting formatting instructions)

and then accepted

AI#1 Logic, satisfiability, constraints, reasoning, solving

① **Variability annotations and modeling**

LaTeX source files

```
{{#if ACK}}
{{#if BOLD_ACK}}\textbf{Acknowledgements.}{{/if}}
{{#if PARAGRAPH_ACK}}\paragraph{Acknowledgements}{{/if}} We thank anonymous re
{{#if LONG_ACK}} We thank Pierre Laperdrix for the newspaper example. {{/if}}
% project fundings also
{{/if}}
%
\scriptsize
%\vspace*{-2mm}
\vspace*{-{{{vspace_bib}}}mm}
\bibliographystyle{abbrv}
\bibliography{DEModularity15}
```

```
\begin{figure}
\centering
\includegraphics[width={{{bref_size}}}\linewidth]{figures/bref-generator.pdf}
\caption{\label{fig:generator}Video generator: modularity and variants}
\end{figure}
```

variability model

```
// Boolean options (features)
fmLaTeX = FM (VARY_LATEX : BREF BIB [PL_FOOTNOTE] [ACK] JS_STYLE
[LONG_AFFILIATION] ;
JS_STYLE : (JS_SCRIPTSIZE | JS_TINY | JS_FOOTNOTESIZE); // mutually exclusive
ACK : [LONG_ACK] (BOLD_ACK | PARAGRAPH_ACK); // LONG_ACK is optional
LONG_AFFILIATION : [EMAIL]; )
// numerical options (attributes)
real BIB.vspace_bib: [1.0..5.0] precision 1 // 1 decimal digit precision
real BREF.bref_size: [0.7..1.0] precision 1 // either 0.7 0.8 0.9 or 1.0
real cserver_size: [0.6..0.9] precision 1 // either 0.6 0.7 0.8 or 0.9
// specific constraints can be added a priori if needs be
…
```

Questions

List all options (features)

Feature model (live)

Hard constraints (part 1, for avoiding invalid PDF)

Hard constraints (part 2, for avoiding 5 pages)

https://github.com/diverse-project/varylatex/

Software Variability and Artificial Intelligence

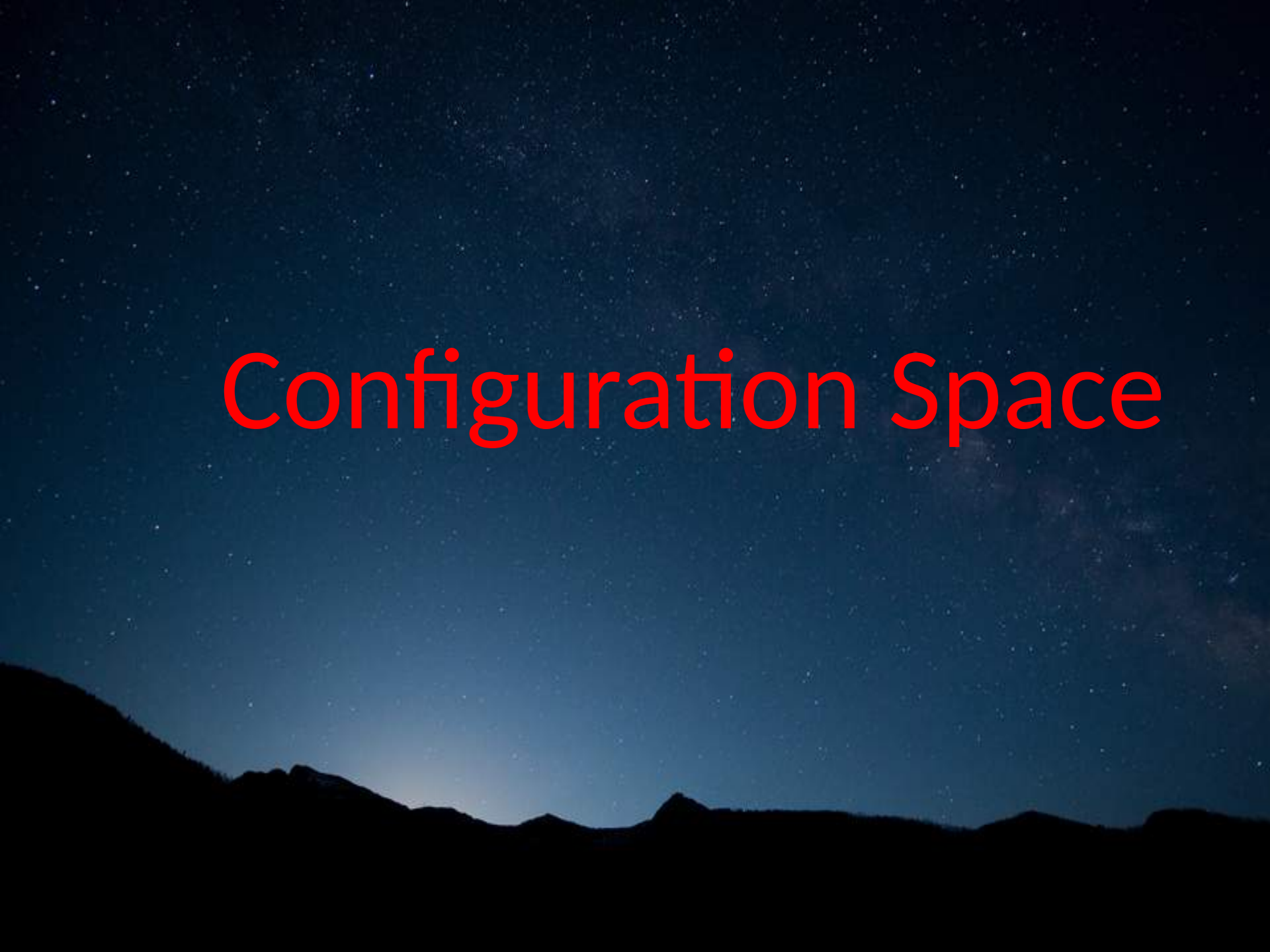
- Very large variability spaces
- **AI#2 Statistical, supervised machine learning** to (out of a sample):
  - Understand the configuration space
  - Find the best configuration
  - Specialize the configuration space (e.g., by capturing constraints)
  - In a cost-effective way

AI#2 Statistical, supervised machine learning (classification problem)

Paper variants building and measurements

ONG_ACK	LONG_AFFILIATION	PARAGRAPH_ACK	PL_FOOTNOTE	VARY_LATEX	bref_size	cserver_size	vspace_bib	nbPages	
rue	false	false	false	true	0.7	0.9	4.0	4	✅
alse	false	false	false	true	0.8	0.6	2.2	4	✅
alse	false	false	false	true	0.9	0.6	2.3	4	✅
rue	true	true	true	true	0.7	0.8	1.1	4	✅
alse	true	false	true	true	0.8	0.9	1.8	5	❌
alse	true	false	false	true	0.7	0.8	2.8	5	❌
alse	false	false	true	true	0.7	0.8	2.9	5	❌
alse	true	false	false	true	0.9	0.7	4.9	4	✅
rue	true	false	true	true	1.0	0.7	1.7	5	❌
alse	false	false	true	true	1.0	0.6	1.8	5	❌
alse	true	false	true	true	0.7	0.6	2.8	4	✅

Conclusion


Configuration Space

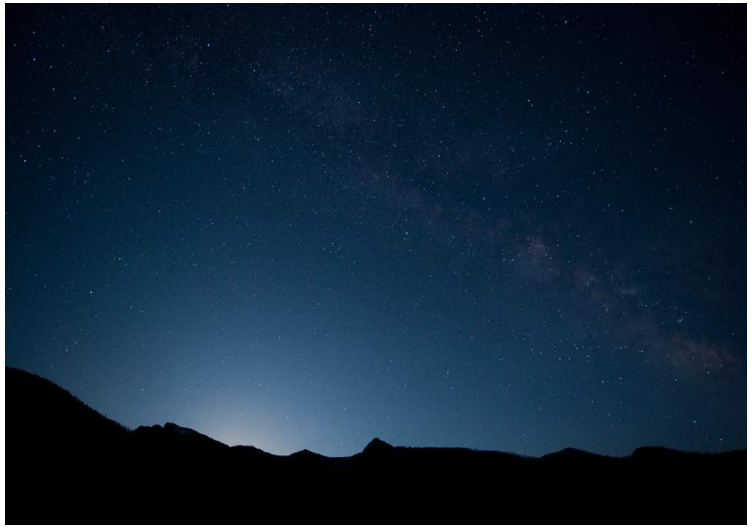
Humans

Machine

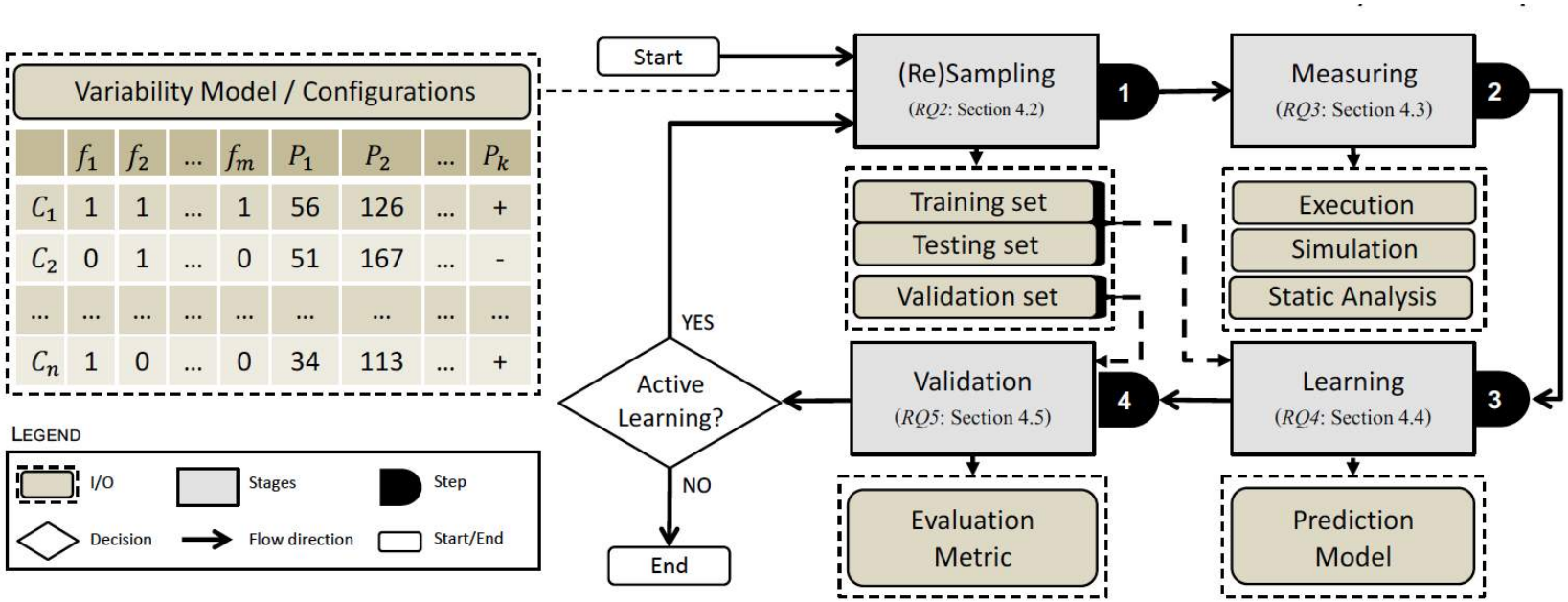
Modeling (SAT/CP)

Knowledge
Soundness
Completeness
Interpretability
Cost
Generalization

Sampling, measuring, learning

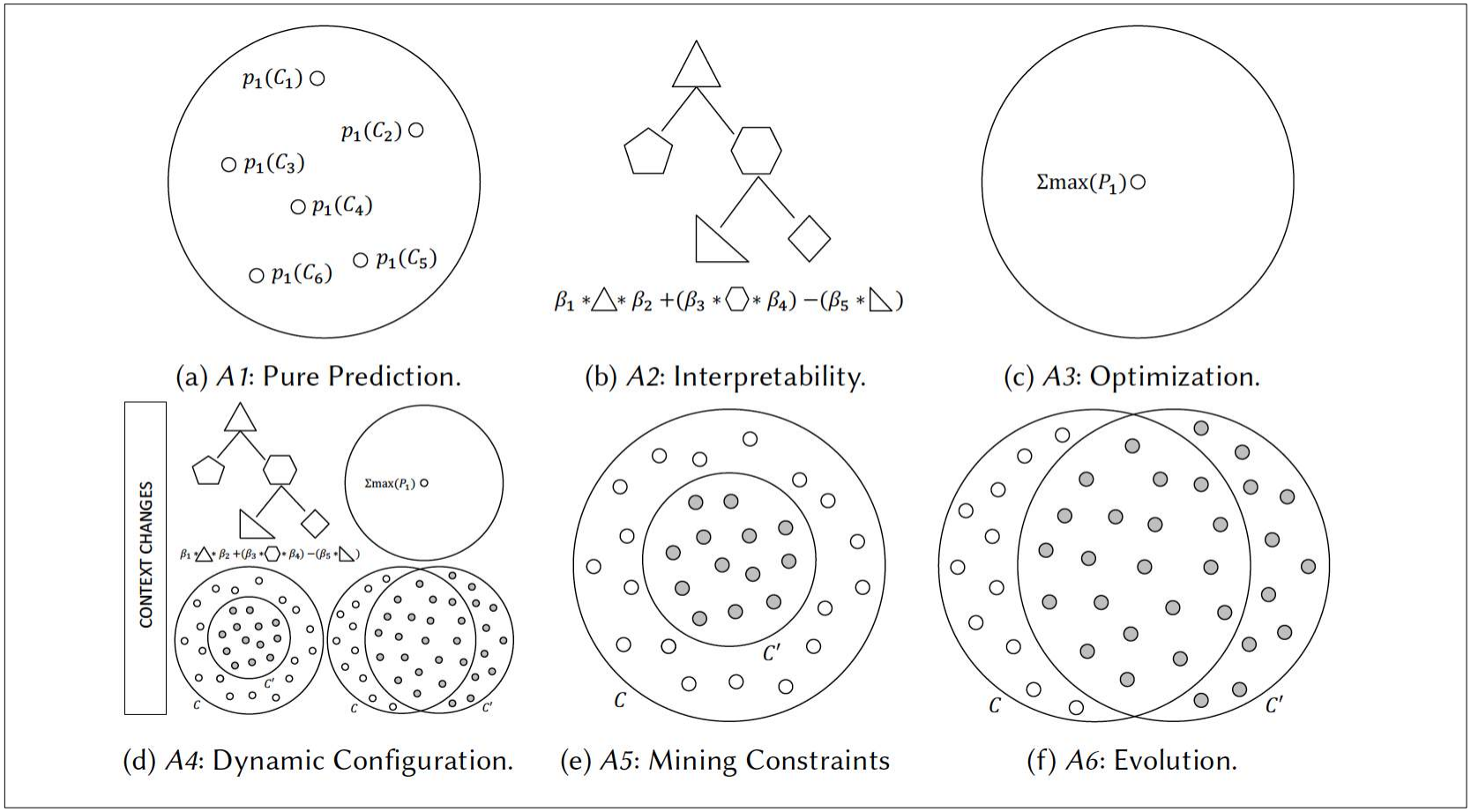


Sampling, Measuring, Learning



Learning Software Configuration Spaces: A Systematic Literature Review
Juliana Alves Pereira, Hugo Martin, Mathieu Acher, Jean-Marc Jézéquel, Goetz Botterweck, Anthony Ventresque https://arxiv.org/abs/1906.03018

Several scenarios



Learning Software Configuration Spaces: A Systematic Literature Review
Juliana Alves Pereira, Hugo Martin, Mathieu Acher, Jean-Marc Jézéquel, Goetz Botterweck, Anthony Ventresque https://arxiv.org/abs/1906.03018

Huge applicability!

Name	Domain	Non-Functional Properties
Thingiverse's	3D printer	defects
IBM WebSphere	Application server	throughput
Clasp	ASP solver	response time
SNW	Asset management	area and throughput
Binpacking	Binpacking algorithm	execution time and accuracy
XGBoost	Boosting algorithms	training time
SaaS system	Cloud computing	response time
Clustering	Clustering algorithm	execution time and accuracy
AJStats	Code analyzer	response time
SaC	Code analyzer	I/O time, response time
POLLY	Code optimizer	runtime
Libssh	Combinatorial model	defects
Telecom	Communication system	defects
LLVM	Compiler	memory footprint, performance, response time, code complexity, compilation time
Compressor SPL	Compression library	compression time, memory usage and compression ratio
7Z	Compression library	Compression time
LRZIP	Compression library	compressed size, compression time, compilation time
RAR	Compression library	code complexity
XZ	Compression library	compression time
ZipMe	Compression library	memory footprint, performance, code complexity, time
WordPress	Content management	CPU power consumption
LinkedList	Data structures	memory footprint, performance, maintainability, binary size
Curl	Data transfer	code complexity

Name	Domain	Non-Functional Properties
Wget	Data transfer	memory footprint, code complexity
Actian Vector	Database system	runtime
Apache Cassandra	Database system	latency
Berkeley DB	Database system	I/O time, memory footprint, performance, response time, code complexity, maintainability, binary size
FAME-DBMS	Database system	maintainability, binary size, performance
MySQL	Database system	defects, throughput, latency
Postgres	Database system	throughput, latency
Prevayler	Database system	memory footprint, performance
SQLite	Database system	memory footprint, performance, response time, code complexity, runtime
StockOnline	Database system	response time
Kafka	Distributed systems	throughput
DNN	DNNs algorithms	accuracy of predictions
Curriculum vitae	Document	number of pages
Paper	Document	number of pages
RUBiS	E-commerce application	response time
EMAIL	E-mail client	time
MBED TLS	Encryption library	response time
SAP ERP	Enterprise Application	response time
noc-CM-log	FPGA	CPU power consumption, runtime
sort-256	FPGA	area, throughput
E-Health System	Health	response time
$HIPA^{cc}$	Image processing	response time
Disparity SPL	Image processing	energy consumption
PKJab	Instant messenger	memory footprint, performance
IBM ILOG CPLEX	Integer solver	runtime
SPECjjbb2005	Java Server	response time, throughput
WEKA	Learning algorithm	accuracy of predictions
SVD	Linear algebra	execution time and accuracy
Trimesh	Mesh solver	iterations, response time
MBENCH	Micro benchmark	time
ACE+TAO system	Middleware software	defects
SensorNetwork	Network simulator	memory footprint, performance
Simonstrator	Network simulator	latency
NoC	Network-based system	energy and runtime
Helmholtz 3D	Numerical analysis	execution time and accuracy
Poisson 2D	Numerical analysis	execution time and accuracy
Linux kernel	Operating system	memory footprint, performance
DNN	Optimization algorithm	response time

Learning Software Configuration Spaces: A Systematic Literature Review
Juliana Alves Pereira, Hugo Martin, Mathieu Acher, Jean-Marc Jézéquel, Goetz Botterweck, Anthony Ventresque https://arxiv.org/abs/1906.03018

Linux

Machine
Learning

TUXML

Grid'5000

Software Engineering and Machine Learning

- Automated measurements of thousands of Linux variants
- Learning with a high precision, with a small sample

Conclusion

- Software variability everywhere for fitting users' requirements; combinatorial explosion
- **AI#1** Abstraction/languages to formally and efficiently reason about configuration spaces
  - with SAT/CSP/SMT solvers, Eg constrained sampling
- **AI#2** Statistical machine learning (out of a sample):
  - Understand the configuration space, find the best configuration, specialize the configuration space (e.g., by capturing constraints)
  - In a cost-effective way
- Artificial intelligence for fitting software variability
- Human/machines interplay

Possible impacts of software variability on your work

- When performing/reporting scientific experiments
  - (hyper-)parameters? heuristics? default configuration? versions?
  - many independent variables have variability
- When verifying a system
  - does it work on any configuration? on a very specific one? can you adapt your verification for many variants?
- More positively (forget problems!): variability can be a fantastic opportunity to explore and understand a space of possible solutions

Advanced topics
(backup slides)

Exercise: The case of Linux

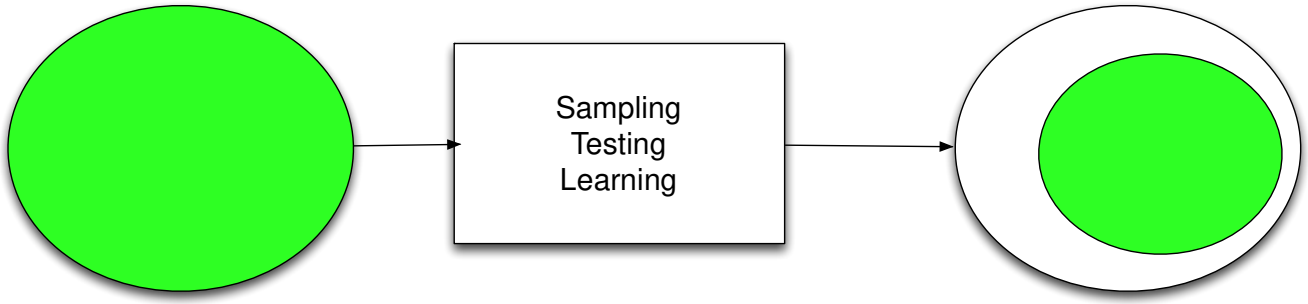
- Sampling
  - translating Kconfig files into SAT: incomplete/unsound
  - Uniform, random sampling Plazar et al. ICST'19
    - Either does not scale
    - Or is not really random
- Measuring
- Learning
- Applicability
- Research questions

configuration options: 9K+ for X86_64
100K+ configurations (!!)

https://github.com/TuxML/ProjetIrma/

cid	compilation_date ▾ 1	compilation_time	config_file	stdlog_file	errlog_file	output_file	core_size	dependencies	gcc_version	libc_version	core_used	incremental_mod	tuxn
1464	2018-04-19 15:23:19	204.414	[BLOB - 22,7 Kio]	[BLOB - 33,2 Kio]	[BLOB - 339 o]	[BLOB - 3,3 Kio]	36313640		6.3.0-18+deb9u1	2.24-11+deb9u3	16	0	pre-a
1463	2018-04-19 15:19:23	122.739	[BLOB - 18,6 Kio]	[BLOB - 25,1 Kio]	[BLOB - 265 o]	[BLOB - 2,9 Kio]	17455904		6.3.0-18+deb9u1	2.24-11+deb9u3	16	0	pre-a
1462	2018-04-19 15:16:51	82.1942	[BLOB - 17 Kio]	[BLOB - 18,8 Kio]	[BLOB - 286 o]	[BLOB - 3 Kio]	30085248		6.3.0-18+deb9u1	2.24-11+deb9u3	16	0	pre-a
1461	2018-04-19 15:14:59	108.779	[BLOB - 19,7 Kio]	[BLOB - 19,1 Kio]	[BLOB - 132 o]	[BLOB - 3,3 Kio]	24138304		6.3.0-18+deb9u1	2.24-11+deb9u3	16	0	pre-a
1460	2018-04-19 15:12:37	168.36	[BLOB - 20,1 Kio]	[BLOB - 26,5 Kio]	[BLOB - 2,9 Kio]	[BLOB - 3,3 Kio]	62716560		6.3.0-18+deb9u1	2.24-11+deb9u3	16	0	pre-a
1459	2018-04-19 15:09:17	204.448	[BLOB - 26,9	[BLOB - 30,7	[BLOB - 14	[BLOB - 2,9	108303064		6.3.0-	2.24-	16	0	pre-a

o1 : {true, false}
o2 : {true, false}
o3 : [0..10]



o1 = false
o2 = {true, false}
o3 : [2..8]
o3 > 6 => o2

Enormous configurations space eg Linux has thousands of options, tri-state values {y, n, m}

KConfig file

```
[...]

config PRINTK
	default y
	bool "Enable support for printk" if EXPERT
	select IRQ_WORK
	help
	  This option enables normal printk support. Removing it
	  eliminates most of the message strings from the kernel image
	  and makes the kernel more or less silent. As this makes it
	  very difficult to diagnose system problems, saying N here is
	  strongly discouraged.

config PRINTK_NMI
	def_bool y
	depends on PRINTK
	depends on HAVE_NMI

config BUG
	bool "BUG() support" if EXPERT
	default y
	help
	  Disabling this option eliminates support for BUG and WARN, reducing
	  the size of your kernel image and potentially quietly ignoring
	  numerous fatal conditions. You should only consider disabling this
	  option for embedded systems with no facilities for reporting errors.
	  Just say Y.

config ELF_CORE
	depends on COREDUMP
	default y
	bool "Enable ELF core dumps" if EXPERT
	help
	  Enable support for generating core dumps. Disabling saves about 4k.

[...]

config AIO
	bool "Enable AIO support" if EXPERT
	default y
	help
	  This option enables POSIX asynchronous I/O which may by used
	  by some high performance threaded applications. Disabling
	  this option saves about 7k.

[...]
```

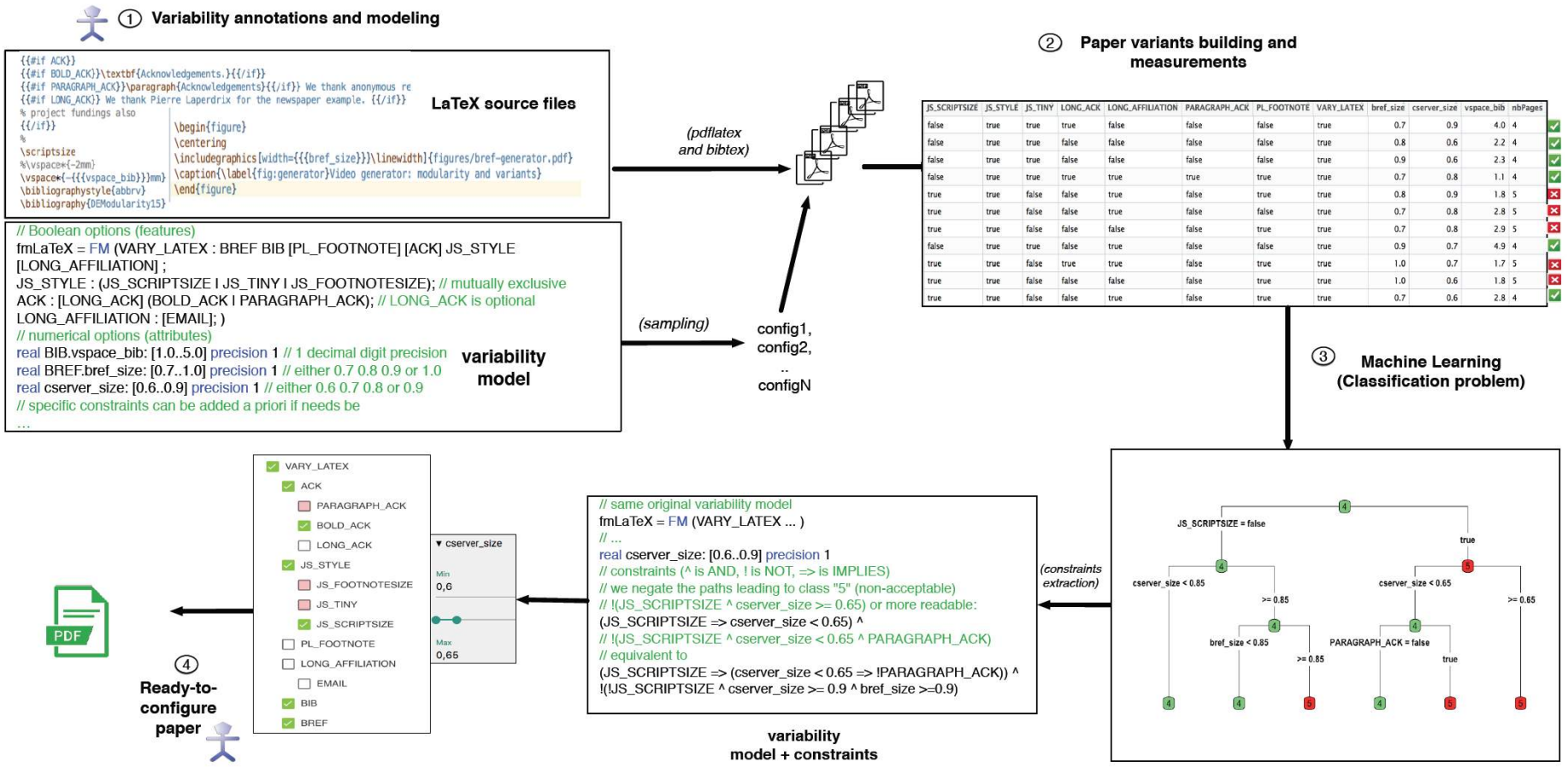
Configurator

(live demonstration)

Two case studies

- FSE paper (see demonstration)
  - Page limit: 4
  - Accuracy: ~85% with 40 papers in the training set (there are 73,440 valid configurations)
- Curiculum vitae generation
  - 18 pages limit; 5 Boolean options; full generation, only 32 papers (not need to learn here)

Process


① Variability annotations and modeling
LaTeX source files
{{#if ACK}}
{{#if BOLD_ACK}}\textbf{Acknowledgements.}{{/if}}
{{#if PARAGRAPH_ACK}}\paragraph{Acknowledgements}{{/if}} We thank anonymous re
{{#if LONG_ACK}} We thank Pierre Laperdrix for the newspaper example. {{/if}}
% project fundings also
{{/if}}
%
\scriptsize
%\vspace*{-2mm}
\vspace*{-{{{vspace_bib}}}mm}
\bibliographystyle{abbrv}
\bibliography{DEModularity15}
\begin{figure}
\centering
\includegraphics[width={{{bref_size}}}\linewidth]{figures/bref-generator.pdf}
\caption{\label{fig:generator}Video generator: modularity and variants}
\end{figure}
(pdflatex and bibtex)
② Paper variants building and measurements
JS_SCRIPTSIZE | JS_STYLE | JS_TINY | LONG_ACK | LONG_AFFILIATION | PARAGRAPH_ACK | PL_FOOTNOTE | VARY_LATEX | bref_size | cserver_size | vspace_bib | nbPages
false | true | true | true | false | false | false | true | 0.7 | 0.9 | 4.0 | 4
false | true | true | false | false | false | false | true | 0.8 | 0.6 | 2.2 | 4
false | true | true | false | false | false | false | true | 0.9 | 0.6 | 2.3 | 4
false | true | true | true | true | true | true | true | 0.7 | 0.8 | 1.1 | 4
true | true | false | false | true | false | true | true | 0.8 | 0.9 | 1.8 | 5
true | true | false | false | true | false | false | true | 0.7 | 0.8 | 2.8 | 5
true | true | false | false | false | false | true | true | 0.7 | 0.8 | 2.9 | 5
false | true | true | false | true | false | false | true | 0.9 | 0.7 | 4.9 | 4
true | true | false | true | true | false | true | true | 1.0 | 0.7 | 1.7 | 5
true | true | false | false | false | false | true | true | 1.0 | 0.6 | 1.8 | 5
true | true | false | false | true | false | true | true | 0.7 | 0.6 | 2.8 | 4
// Boolean options (features)
fmLaTeX = FM (VARY_LATEX : BREF BIB [PL_FOOTNOTE] [ACK] JS_STYLE [LONG_AFFILIATION] ;
JS_STYLE : (JS_SCRIPTSIZE I JS_TINY I JS_FOOTNOTESIZE); // mutually exclusive
ACK : [LONG_ACK] (BOLD_ACK I PARAGRAPH_ACK); // LONG_ACK is optional
LONG_AFFILIATION : [EMAIL];)
// numerical options (attributes)
real BIB.vspace_bib: [1.0..5.0] precision 1 // 1 decimal digit precision
real BREF.bref_size: [0.7..1.0] precision 1 // either 0.7 0.8 0.9 or 1.0
real cserver_size: [0.6..0.9] precision 1 // either 0.6 0.7 0.8 or 0.9
// specific constraints can be added a priori if needs be
...
variability model
(sampling)
config1,
config2,
..
configN
③ Machine Learning (Classification problem)
JS_SCRIPTSIZE = false
true
cserver_size < 0.85
>= 0.85
cserver_size < 0.65
>= 0.65
bref_size < 0.85
>= 0.85
PARAGRAPH_ACK = false
true
(constraints extraction)
// same original variability model
fmLaTeX = FM (VARY_LATEX ...)
// ...
real cserver_size: [0.6..0.9] precision 1
// constraints (^ is AND, ! is NOT, => is IMPLIES)
// we negate the paths leading to class "5" (non-acceptable)
// !(JS_SCRIPTSIZE ^ cserver_size >= 0.65) or more readable:
(JS_SCRIPTSIZE => cserver_size < 0.65) ^
// !(JS_SCRIPTSIZE ^ cserver_size < 0.65 ^ PARAGRAPH_ACK)
// equivalent to
(JS_SCRIPTSIZE => (cserver_size < 0.65 => !PARAGRAPH_ACK)) ^
!(!JS_SCRIPTSIZE ^ cserver_size >= 0.9 ^ bref_size >=0.9)
variability model + constraints
VARY_LATEX
ACK
PARAGRAPH_ACK
BOLD_ACK
LONG_ACK
JS_STYLE
JS_FOOTNOTESIZE
JS_TINY
JS_SCRIPTSIZE
PL_FOOTNOTE
LONG_AFFILIATION
EMAIL
BIB
BREF
▼ cserver_size
Min
0,6
Max
0,65
PDF
④ Ready-to-configure paper

AI#1 Logic, satisfiability, constraints, reasoning, solving

① **Variability annotations and modeling**

LaTeX source files

```
{{#if ACK}}
{{#if BOLD_ACK}}\textbf{Acknowledgements.}{{/if}}
{{#if PARAGRAPH_ACK}}\paragraph{Acknowledgements}{{/if}} We thank anonymous re
{{#if LONG_ACK}} We thank Pierre Laperdrix for the newspaper example. {{/if}}
% project fundings also
{{/if}}
%
\scriptsize
%\vspace*{-2mm}
\vspace*{-{{{vspace_bib}}}mm}
\bibliographystyle{abbrv}
\bibliography{DEModularity15}
```

```
\begin{figure}
\centering
\includegraphics[width={{{bref_size}}}\linewidth]{figures/bref-generator.pdf}
\caption{\label{fig:generator}Video generator: modularity and variants}
\end{figure}
```

variability model

```
// Boolean options (features)
fmLaTeX = FM (VARY_LATEX : BREF BIB [PL_FOOTNOTE] [ACK] JS_STYLE
[LONG_AFFILIATION] ;
JS_STYLE : (JS_SCRIPTSIZE | JS_TINY | JS_FOOTNOTESIZE); // mutually exclusive
ACK : [LONG_ACK] (BOLD_ACK | PARAGRAPH_ACK); // LONG_ACK is optional
LONG_AFFILIATION : [EMAIL]; )
// numerical options (attributes)
real BIB.vspace_bib: [1.0..5.0] precision 1 // 1 decimal digit precision
real BREF.bref_size: [0.7..1.0] precision 1 // either 0.7 0.8 0.9 or 1.0
real cserver_size: [0.6..0.9] precision 1 // either 0.6 0.7 0.8 or 0.9
// specific constraints can be added a priori if needs be
…
```

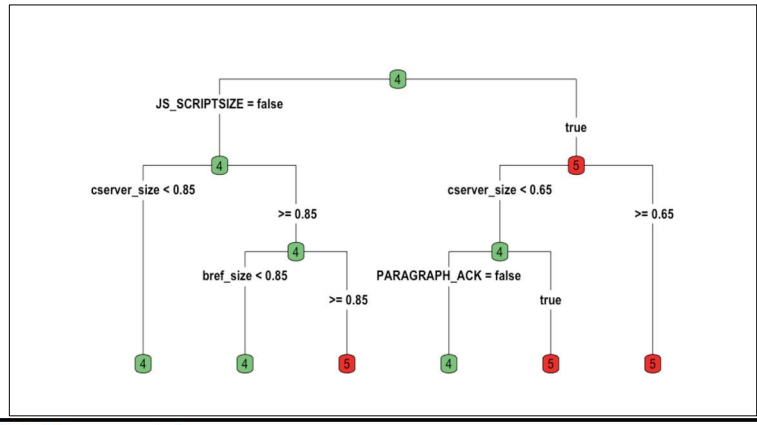
AI#2 Statistical, supervised machine learning (classification problem)

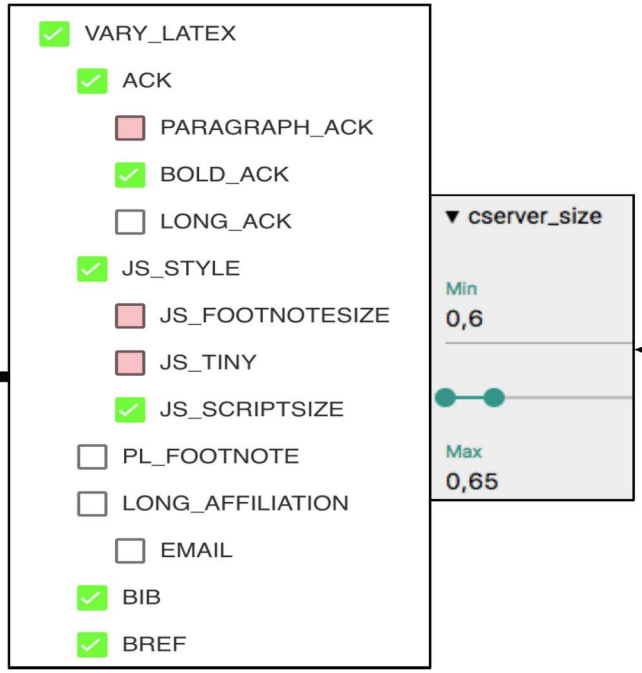
Paper variants building and measurements

ONG_ACK	LONG_AFFILIATION	PARAGRAPH_ACK	PL_FOOTNOTE	VARY_LATEX	bref_size	cserver_size	vspace_bib	nbPages	
rue	false	false	false	true	0.7	0.9	4.0	4	✅
alse	false	false	false	true	0.8	0.6	2.2	4	✅
alse	false	false	false	true	0.9	0.6	2.3	4	✅
rue	true	true	true	true	0.7	0.8	1.1	4	✅
alse	true	false	true	true	0.8	0.9	1.8	5	❌
alse	true	false	false	true	0.7	0.8	2.8	5	❌
alse	false	false	true	true	0.7	0.8	2.9	5	❌
alse	true	false	false	true	0.9	0.7	4.9	4	✅
rue	true	false	true	true	1.0	0.7	1.7	5	❌
alse	false	false	true	true	1.0	0.6	1.8	5	❌
alse	true	false	true	true	0.7	0.6	2.8	4	✅

#AI1 + #AI2

Specialization of the variability model






```
// same original variability model
fmLaTeX = FM (VARY_LATEX ... )
// ...
real cserver_size: [0.6..0.9] precision 1
// constraints (^ is AND, ! is NOT, => is IMPLIES)
// we negate the paths leading to class "5" (non-acceptable)
// !(JS_SCRIPTSIZE ^ cserver_size >= 0.65) or more readable:
(JS_SCRIPTSIZE => cserver_size < 0.65) ^
// !(JS_SCRIPTSIZE ^ cserver_size < 0.65 ^ PARAGRAPH_ACK)
// equivalent to
(JS_SCRIPTSIZE => (cserver_size < 0.65 => !PARAGRAPH_ACK)) ^
!(!JS_SCRIPTSIZE ^ cserver_size >= 0.9 ^ bref_size >=0.9)
```

variability

https://github.com/FAMILIAR-project/varylatex/

```
{{#if ACK}}
{{#if BOLD_ACK}}\textbf{Acknowledgements.}{{/if}}
{{#if PARAGRAPH_ACK}}\paragraph{Acknowledgements}{{/if}} We thank anonymous re
{{#if LONG_ACK}} We thank Pierre Laperdrix for the newspaper example. {{/if}}
% project fundings also
{{/if}}
%
\scriptsize
%\vspace*{-2mm}
\vspace*{-{{{vspace_bib}}}mm}
\bibliographystyle{abbrv}
\bibliography{DEModularity15}
```

Acknowledgements.
We thank anonymous reviewers for their valuable feedbacks. We thank Pierre Laperdrix for the newspaper example.
4. REFERENCES

Acknowledgements. We thank anonymous reviewers for their valuable feedbacks.

4. REFERENCES

Acknowledgements. We thank anonymous reviewers for their valuable feedbacks.
4. REFERENCES

...

Acknowledgements.
We thank anonymous reviewers for their valuable feedbacks.
4. REFERENCES

Variability and LaTeX source files

Paper variants (PDF)

(a) Variability annotations and excerpt of some possible paper variants

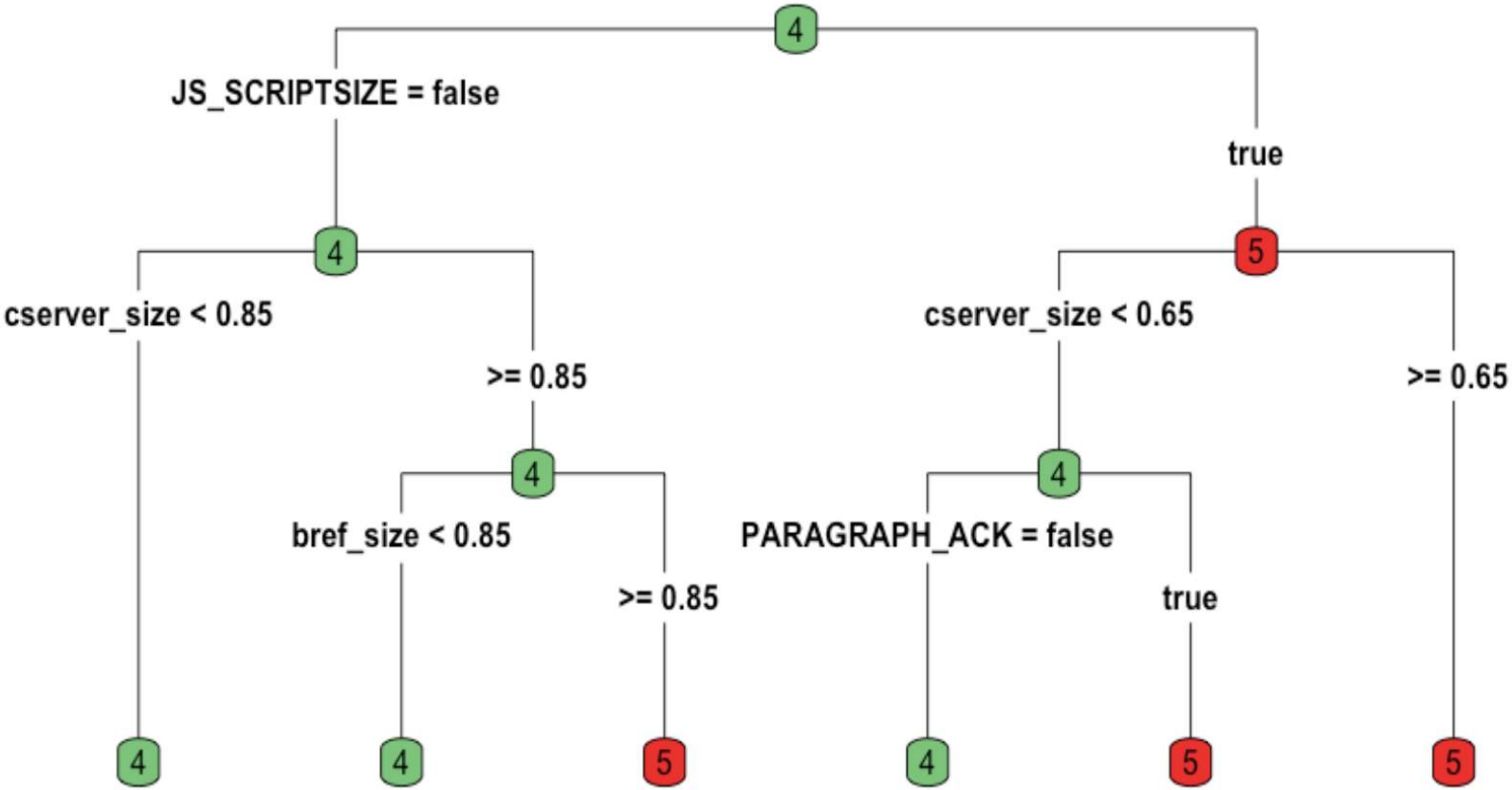
```
\lstdefinelanguage{JavaScript}{
  keywords={typeof, new, true, false, catch, function, return, null, catch, switch, var, if, in, while, do, else, case, break},
  keywordstyle=\color{blue}\bfseries,
  basicstyle=\ttfamily{{#if JS_SCRIPTSIZE}}\scriptsize{{/if}}{{#if JS_TINY}}\tiny{{/if}}{{#if JS_FOOTNOTESIZE}}\footnotesize{{/if}},
```

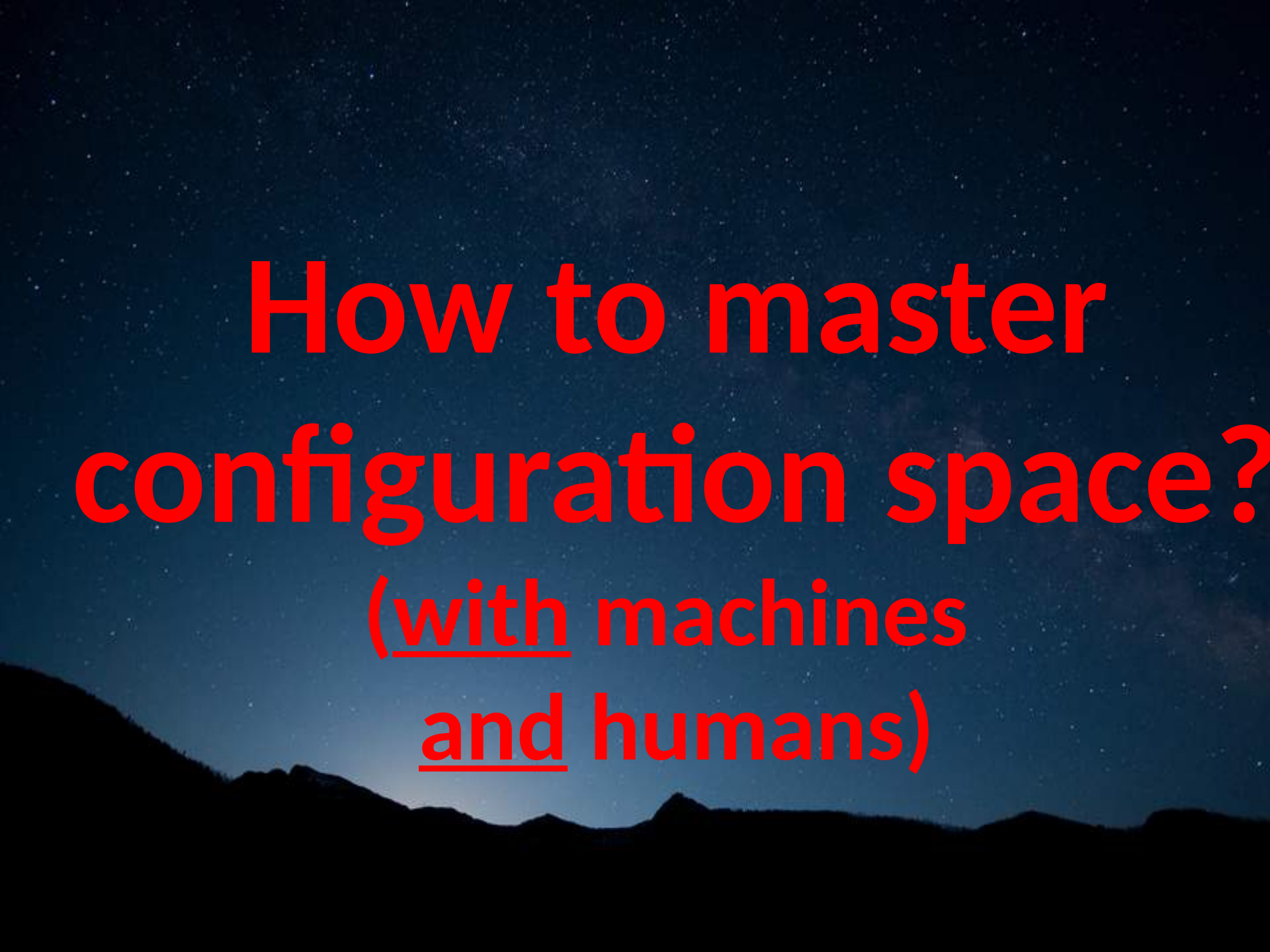
```
{{#if PL_FOOTNOTE}}\footnote{We are considering "product lines" in a broad sense,
```

```
\begin{figure}
\centering
\includegraphics[width={{{bref_size}}}\linewidth]{figures/bref-generator.pdf}
\caption{\label{fig:generator}Video generator: modularity and variants}
\end{figure}
```

(b) Users can vary the font size of a code snippet, activate a footnote, vary the font size of a figure, *etc.*

Classification tree




How to master configuration space?
(with machines and humans)


How to master configuration space?
(for machines and humans)


How to master configuration space?
AI (here) =
variability modeling / automated reasonin
+ statistical machine learning