17-708 SOFTWARE PRODUCT LINES: CONCEPTS AND IMPLEMENTATION

CHRISTIAN KAESTNER
CARNEGIE MELLON UNIVERSITY
INSTITUTE OF SOFTWARE RESEARCH
BUILD & PRICE YOUR 2016 IMPALA

2016 Impala LS

Price
$27,920* OR

Please see your dealer for payment information

View:
- Standard Equipment
- Package
- Mechanical
- Exterior
- Interior
- Safety

Package

- Protection Package $970
- All-Weather Mat Protection Package $200

Mechanical

- ECOTEC 2.5L DOHC 4-cylinder engine with direct injection (DI), Variable Valve Timing (VVT) and auto start/stop
- Engine block heater $75

Exterior

- No Selection $0
- 18" chrome finish aluminum wheels $2,500
- 18" aluminum wheels $1,500
- 19" aluminum wheels $3,530
- Rear spoiler $425
- Body-color surround grille $506

Some selected configurations, options, accessories, and/or color may not be shown on image.
MASS CUSTOMIZATION IN CAR PRODUCTION

30 years ago

- Little variability
- A single variant creating 40% of all profit

15 years ago

- $10^{20}$ configurations at Audi
- $10^{32}$ configurations at BMW
- Rarely two identical configurations produced
- 100 different undercarriages
- 50 different steering wheels
Früchte

Köstliche Bio-Trockenfrüchte, müsiggerecht aufbereitet. Du kannst eine Frucht auch mehrmals auswählen, um deren Anteil zu steigern.

Ananas
lecker, exotisch und wunderbar | 0,65€ (30g)
mehr Infos

Apfelstücke
Ohne Worte weil Klassiker | 0,45€ (25g)
mehr Infos

Aprikosen

©2011 mymuesli GmbH
Oko-Kontrollstelle DE-OST
Impressum

Nährwerte pro 100g

575g nur 4,70€
entspricht 8,17€/kg
inkl. MWSt., zzgl. Versandkosten

durchgemixt?
weiter

hoch ▲
runter ▼
VEGETARIAN

WHICH WICH WOULD YOU LIKE?

- TRIPLE CHEESE MELT
- ELVIS WICH (w/ Honey & Banana)
- TOMATO & AVOCADO
- BLACK BEAN PATTY
- HUMMUS & BELL PEPPERS

CHOOSE YOUR BREAD

- WHITE
- WHEAT

CHOOSE YOUR CHEESE (Optional)

- AMERICAN
- SWISS
- PROVOLONE
- CHEDDAR
- PEPPER JACK
- MOZZARELLA

How Would You Like Your WICH Worked?

MUSTARDS
- Yellow
- Dijon
- Honey
- Deli

MAYOS
- Regular
- Lite
- Horseradish
- Spicy

SPREADS & SAUCES
- BBQ
- Buffalo
- Marinara
- 1000 Island
- Ranch

ONION
- Red
- Grilled
- Crispy Strings

VEGGIES
- Lettuce
- Tomato
- Pickles
- Jalapenos
- Olive Salad
- Mushrooms
- Sauerkraut
- Coleslaw
- Bell Peppers

OILS & SPICES
- Oil
- Vinegar
- Salt
- Pepper
- Oregano
- Parmesan

EXTRAS (75¢ Each)
- Bacon
- Avocado
- Pickle (Whole)
- More Meat
- More Cheese
SOFTWARE BETWEEN STANDARDIZATION AND SPECIALIZATION
Printer Firmware
**LINUX KERNEL**

~6 000 000 Lines of C code

Highly configurable

> 10,000 configuration options!

(x86, 64bit, ...)

Most source code is “optional”
This cpufreq governor sets the frequency statically to the lowest available CPU frequency.

To compile this driver as a module, choose M here: the module will be called `cpufreq_powersave`.

If in doubt, say Y.
SOFTWARE PRODUCT LINES IN INDUSTRY

Boeing
Bosch Group
Cummins, Inc.
Ericsson
General Dynamics
General Motors
Hewlett Packard
Lockheed Martin
Lucent
NASA
Nokia
Philips
Siemens
...

...
Android Ecosystem

Devices

Infrastructural Components

Third Party Applications and Services

Applications and Services
BUT
Variability = Complexity
33 optional, independent features

a unique configuration for every person on this planet
320 optional, independent features

more configurations than estimated

atoms in the universe
Correctness?
Understanding?
MASS CUSTOMIZATION
IN CAR PRODUCTION

30 years ago

Little variability

A single variant creating 40% of all profit

15 years ago

10^{20} configurations at Audi
10^{32} configurations at BMW

Rarely two identical configurations produced

100 different undercarriages
50 different steering wheels
TOPICS

Process and Domain Engineering
Feature and decision models
Analysis of feature models
Product management, scoping
Variability in requirements
Version control systems, Parameters
Design patterns, architecture, components, frameworks, platforms
Build systems, preprocessors
Tradeoff and maintenance discussion (separation of concerns, modularity, traceability, ...)
Quality assurance: Testing and Analysis
Variability management
Team organization
Transition process, refactoring
ORGANIZATION
PREREQUISITES

Basic programming skills (any language, Java preferred)
Basic software engineering knowledge (e.g., process, requirements)
PROJECT

In the second half of the course
Connect course to your research / interests
Explore a topic in depth
   Case study, experiment, ...
Suggest a topic by Oct 14
Summarize results in a report (paper draft)
ASSIGNMENTS & 
READINGS

Weekly assignments

Small analysis, modeling, or implementation tasks
Investigating open source implementations
Due 11:59pm before class, expect in-class discussion

Reading assignments for most classes

Book chapters and papers
Background, context, case studies, …
Expect in-class discussions
see website for links
GRADING

30% assignments, graded on 3-point likert scale:
- exceeds expectations
- matches expectations
- below expectations

30% project

30% exams
- midterm Oct 14, final tbd.

10% participation
HOMEWORK 1 (DUE NEXT WEEK)

Select an experience report Chapter 8—17 of Software Product Lines in Action (first come first serve)


Prepare a 10 minute presentation

• Context
• Why a software product line?
• Development using domain/application engineering
• Goals, benefits, surprises, and challenges
READING ASSIGNMENT (WEDN.)


pages 3-31 (chapters 1 & 2 until 2.3.2)
LEARNING GOALS

explain the product-line development process consisting of domain engineering and application engineering (including how the different phases interact),

distinguish problem space and solution space,

explain the economic lever of product lines and understand the benefit of automation,
Domain Engineering

- domain knowledge

Domain Analysis

Domain Design

Domain Implementation

- domain-specific languages
- generators
- components

new requirements

Application Engineering

- customer needs

Requirements Analysis

Design Analysis

Custom Design

Integration and Test

- product configuration

features

new requirements

product
DOMAIN ENGINEERING

[...] is the activity of collecting, organizing, and storing past experience in building systems [...] in a particular domain in the form of reusable assets [...], as well as providing an adequate means for reusing these assets (i.e., retrieval, qualification, dissemination, adaptation, assembly, and so on) when building new systems.

K. Czarnecki and U. Eisenecker
SOFTWARE PRODUCT LINES

A software product line (SPL) is a set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.

Software Engineering Institute
Carnegie Mellon University
FURTHER READINGS

