Pattern & Pattern Discovery

10th Symposium and Summer School On Service-Oriented Computing,
June 27 – July 1, 2016 in Crete, Greece

University of Stuttgart
Universitätsstr. 38
70569 Stuttgart
Germany

Johanna Barzen, Michael Falkenthal,
Frank Leymann
Institute of Architecture of Application Systems
Leymann@iaas.uni-stuttgart.de

Phone +49-711-685 88470
Fax +49-711-685 88472

Agenda

Capturing Knowledge

Discovering Patterns

Patterns in the Humanities

From Patterns to Solutions

Summary

Agenda

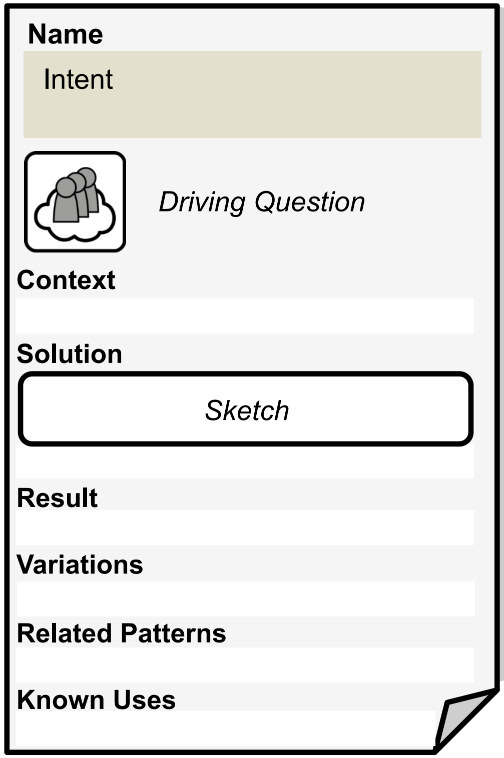
- Capturing Knowledge
- Discovering Patterns
- Patterns in the Humanities
- From Patterns to Solutions
- Summary

Pattern

- Concept introduced in 1977
  - By Ch. Alexander, a “real” architect
- Used to document techniques for solving a class of recurring problems in the abstract
- It is not a series of concrete instructions how to solve a problem

It’s a *nugget of advice*
about solving a recurring problem

Patterns in Many Domains, e.g. Cloud Computing



Name

Intent

Driving Question

Context

Solution

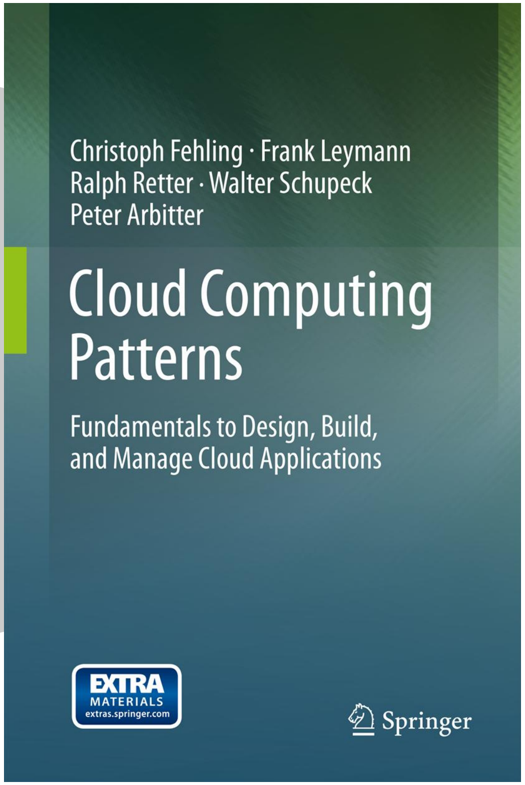
Sketch

Result

Variations

Related Patterns

Known Uses



Christoph Fehling · Frank Leymann
Ralph Retter · Walter Schupeck
Peter Arbitter

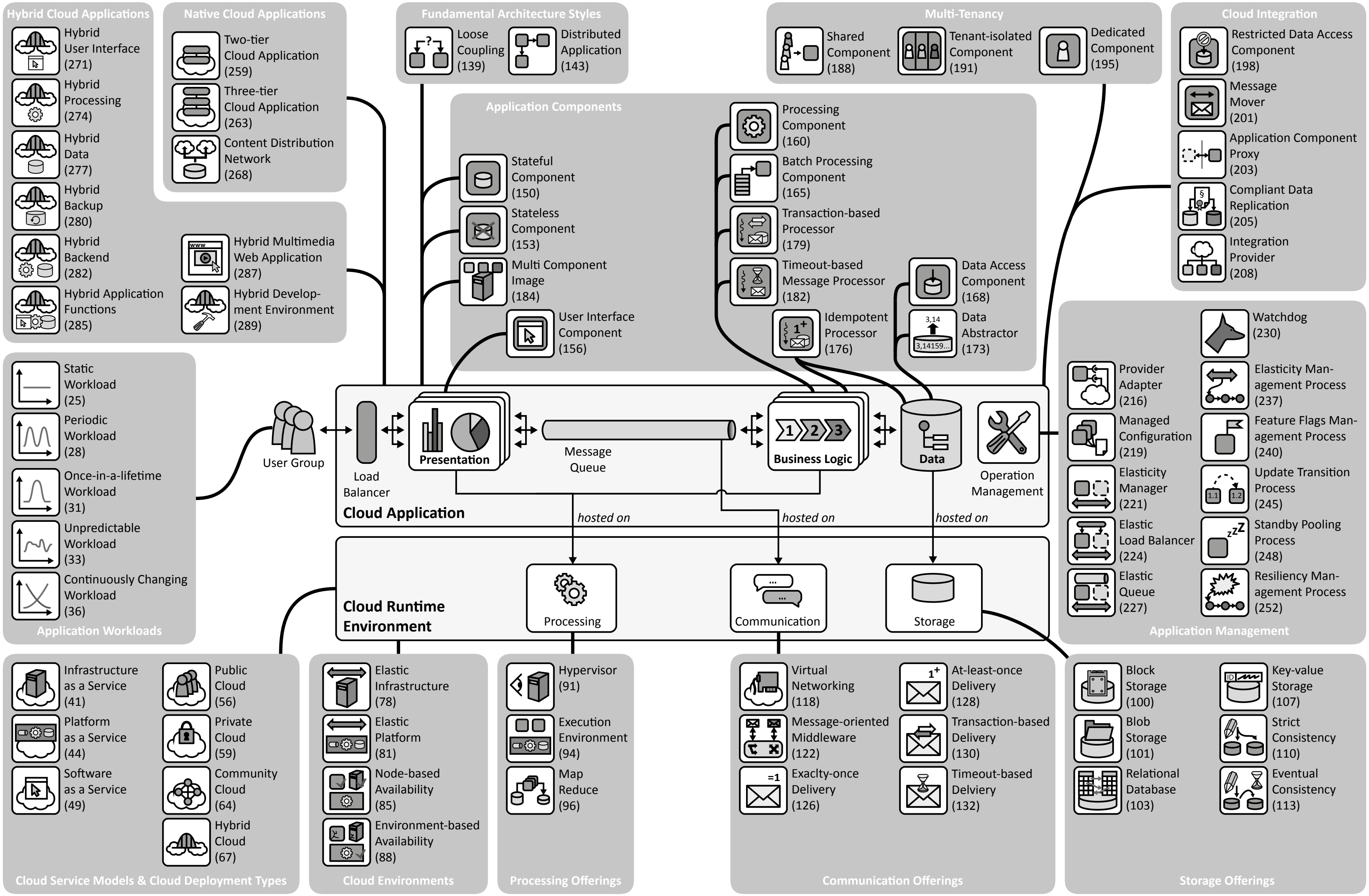
Cloud Computing Patterns

Fundamentals to Design, Build, and Manage Cloud Applications

EXTRA MATERIALS extras.springer.com

Springer

…It's a Pattern Language



Hybrid Cloud Applications

Hybrid User Interface (271)

Hybrid Processing (274)

Hybrid Data (277)

Hybrid Backup (280)

Hybrid Backend (282)

Hybrid Application Functions (285)

Native Cloud Applications

Two-tier Cloud Application (259)

Three-tier Cloud Application (263)

Content Distribution Network (268)

Hybrid Multimedia Web Application (287)

Hybrid Development Environment (289)

Fundamental Architecture Styles

Loose Coupling (139)

Distributed Application (143)

Multi-Tenancy

Shared Component (188)

Tenant-isolated Component (191)

Dedicated Component (195)

Cloud Integration

Restricted Data Access Component (198)

Message Mover (201)

Application Component Proxy (203)

Compliant Data Replication (205)

Integration Provider (208)

Application Components

Stateful Component (150)

Stateless Component (153)

Multi Component Image (184)

User Interface Component (156)

Processing Component (160)

Batch Processing Component (165)

Transaction-based Processor (179)

Timeout-based Message Processor (182)

Idempotent Processor (176)

Data Access Component (168)

Data Abstractor (173)

Application Workloads

Static Workload (25)

Periodic Workload (28)

Once-in-a-lifetime Workload (31)

Unpredictable Workload (33)

Continuously Changing Workload (36)

User Group

Cloud Application

Load Balancer

Presentation

Message Queue

Business Logic

Data

Operation Management

hosted on

hosted on

hosted on

Cloud Runtime Environment

Processing

Communication

Storage

Application Management

Watchdog (230)

Provider Adapter (216)

Elasticity Management Process (237)

Managed Configuration (219)

Feature Flags Management Process (240)

Elasticity Manager (221)

Update Transition Process (245)

Elastic Load Balancer (224)

Standby Pooling Process (248)

Elastic Queue (227)

Resiliency Management Process (252)

Cloud Service Models & Cloud Deployment Types

Infrastructure as a Service (41)

Platform as a Service (44)

Software as a Service (49)

Public Cloud (56)

Private Cloud (59)

Community Cloud (64)

Hybrid Cloud (67)

Cloud Environments

Elastic Infrastructure (78)

Elastic Platform (81)

Node-based Availability (85)

Environment-based Availability (88)

Processing Offerings

Hypervisor (91)

Execution Environment (94)

Map Reduce (96)

Communication Offerings

Virtual Networking (118)

Message-oriented Middleware (122)

Exaclty-once Delivery (126)

At-least-once Delivery (128)

Transaction-based Delivery (130)

Timeout-based Delivery (132)

Storage Offerings

Block Storage (100)

Blob Storage (101)

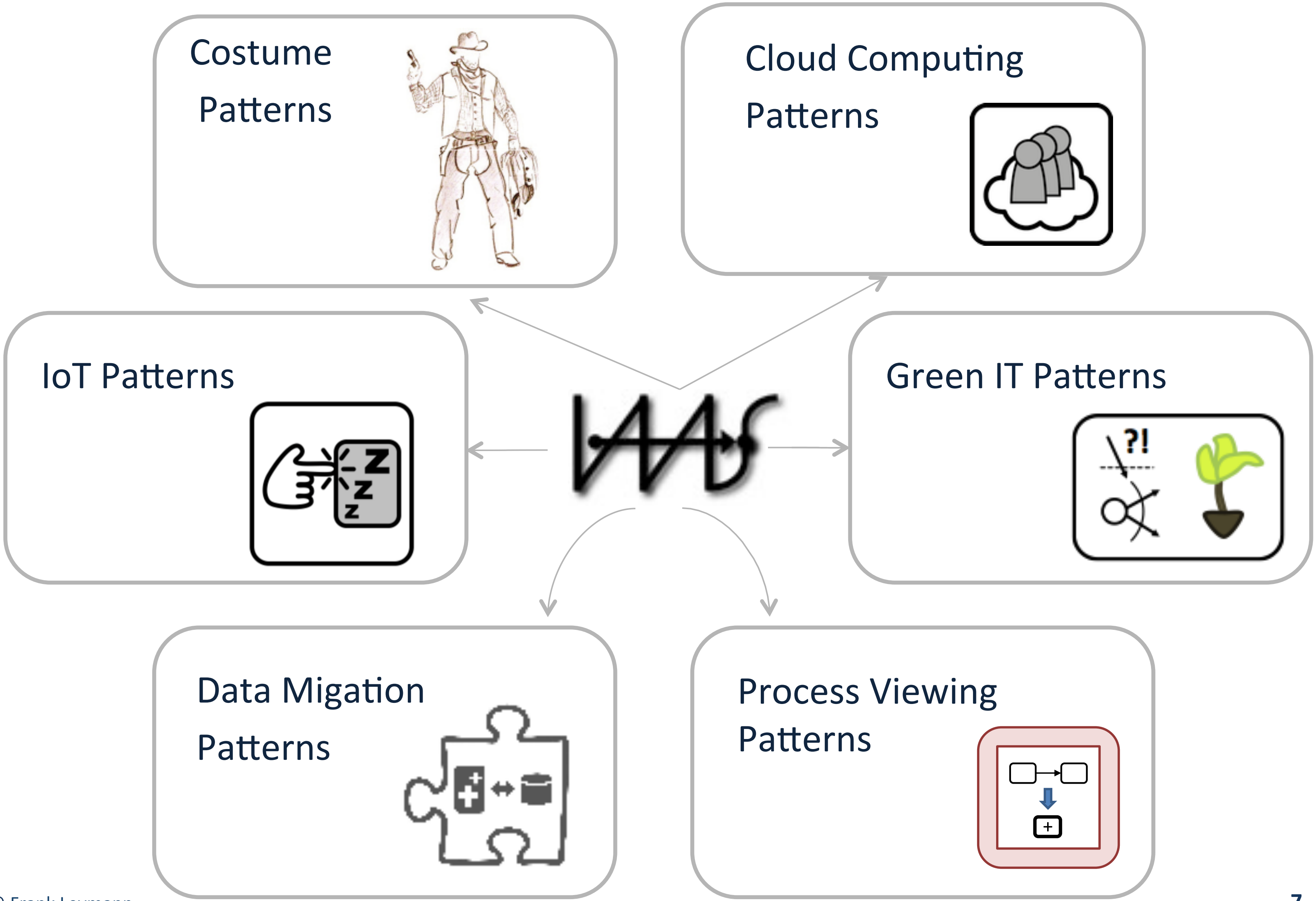
Relational Database (103)

Key-value Storage (107)

Strict Consistency (110)

Eventual Consistency (113)

Research on Patterns at IAAS



Costume Patterns

Cloud Computing Patterns

IoT Patterns

Green IT Patterns

Data Migation Patterns

Process Viewing Patterns

Agenda

Capturing Knowledge

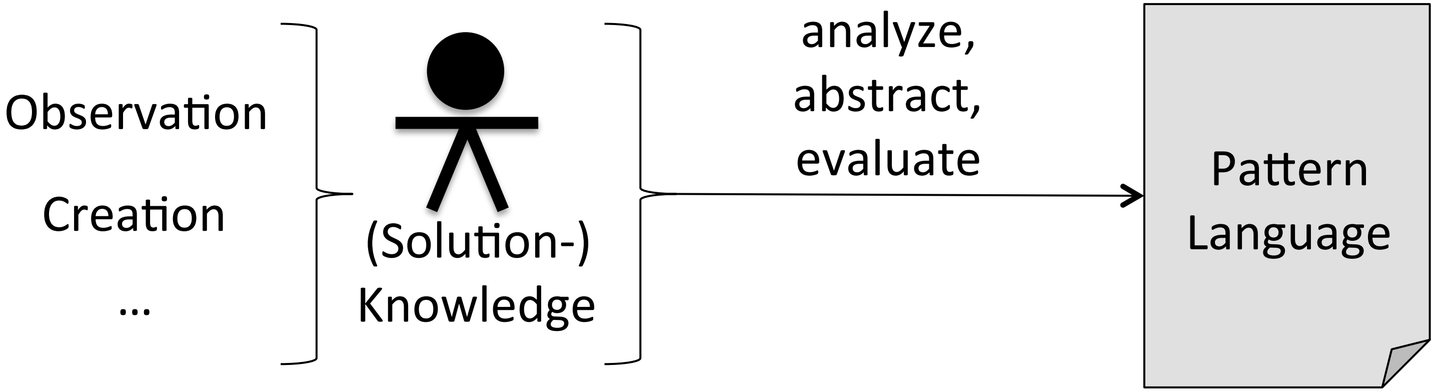
Discovering Patterns

Patterns in the Humanities

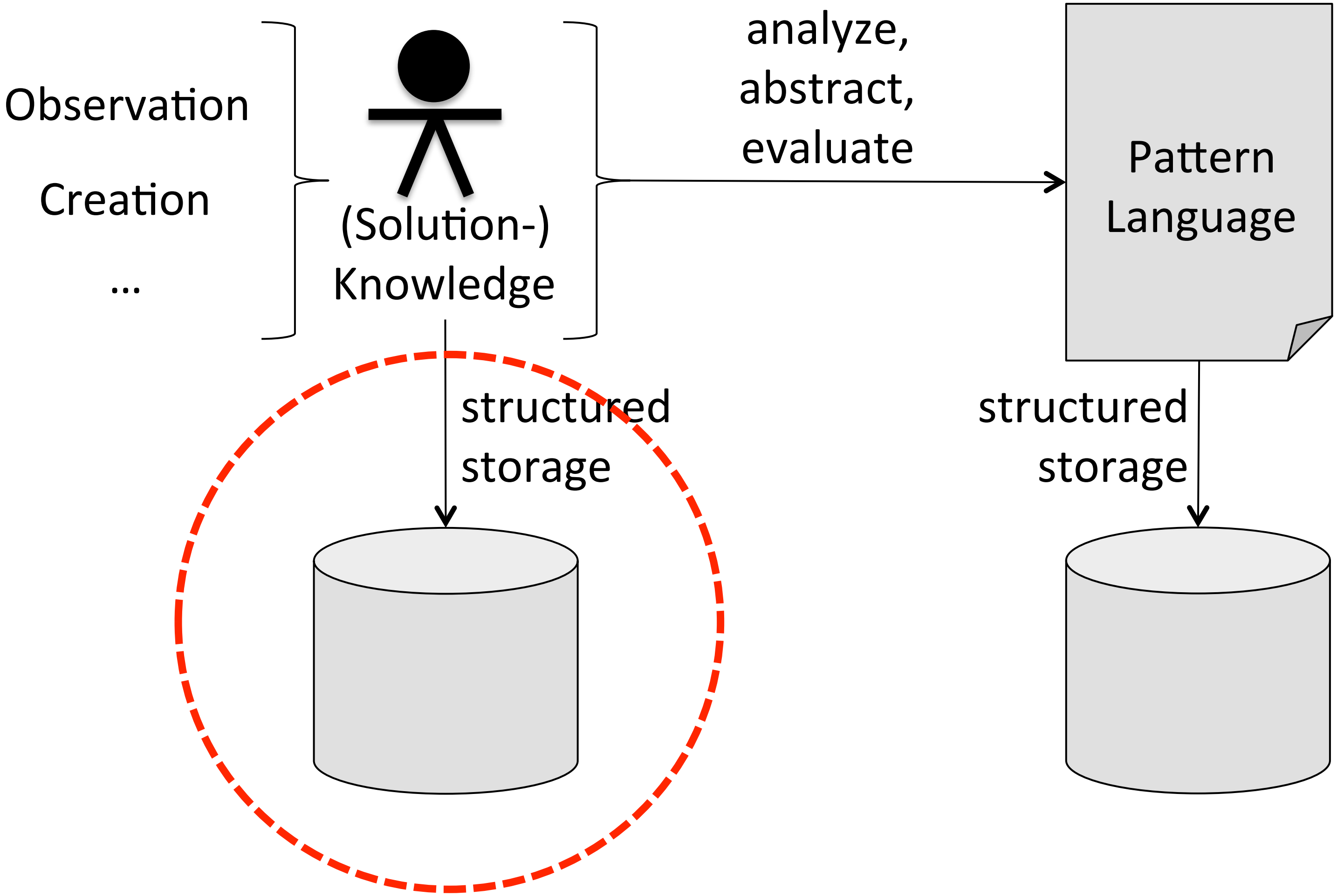
From Patterns to Solutions

Summary

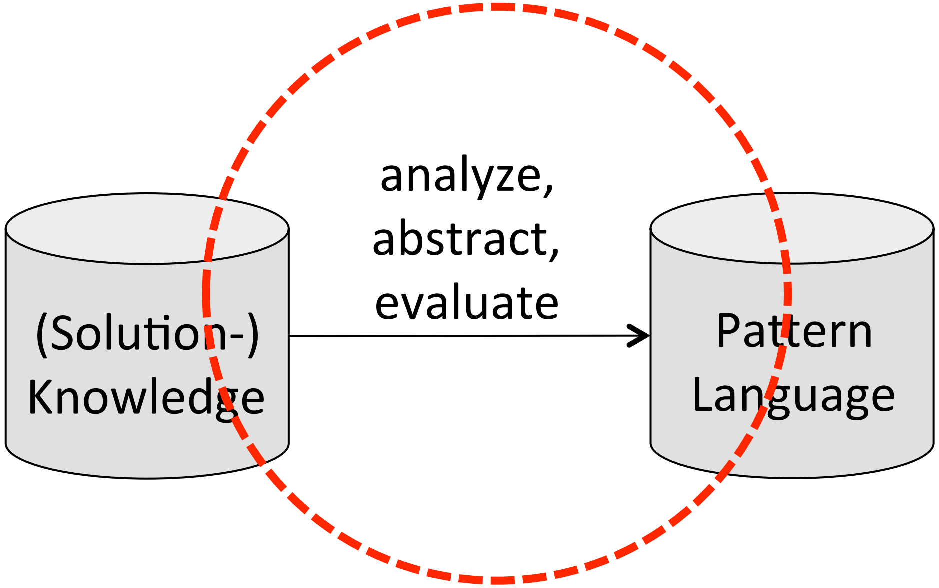
Development of Pattern Languages: Traditional



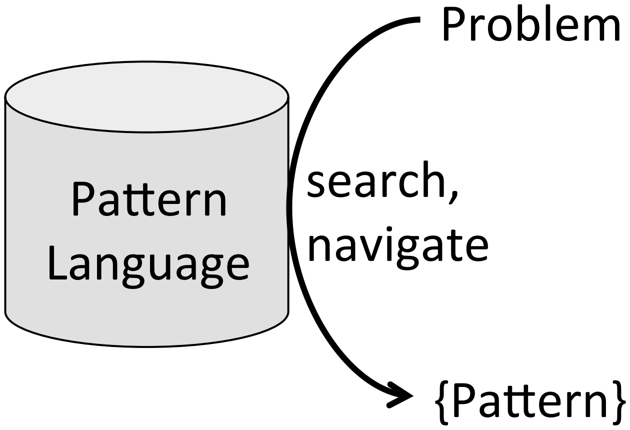
A Step Forward: Use of Repositories



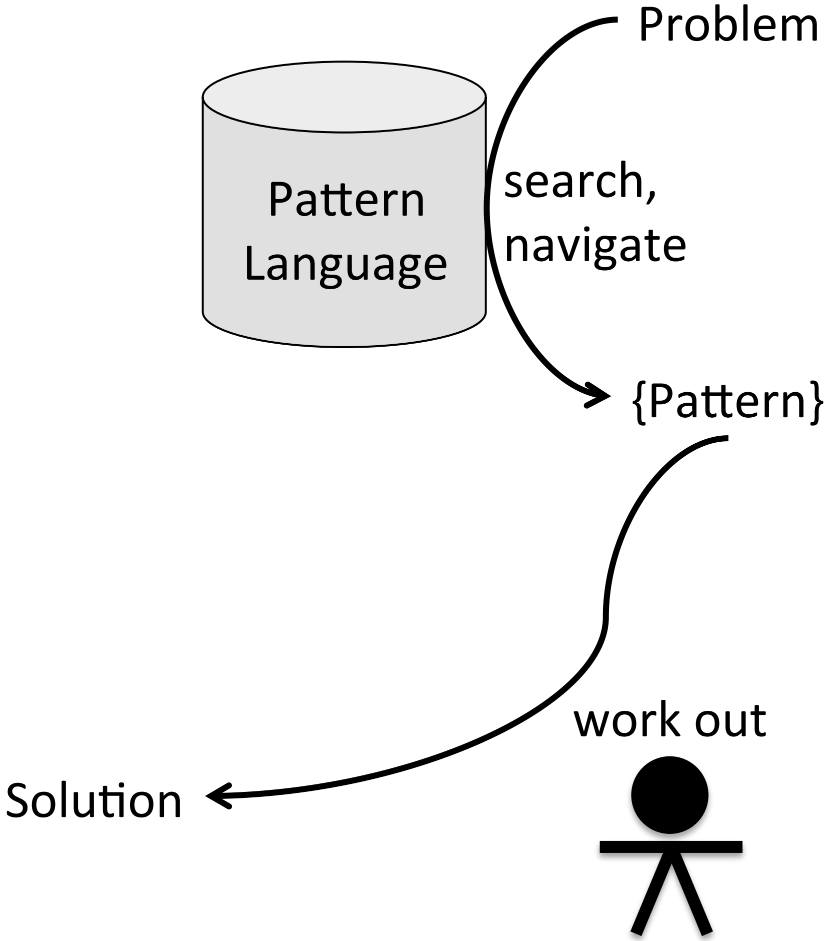
Deriving Patterns from Stored Solutions



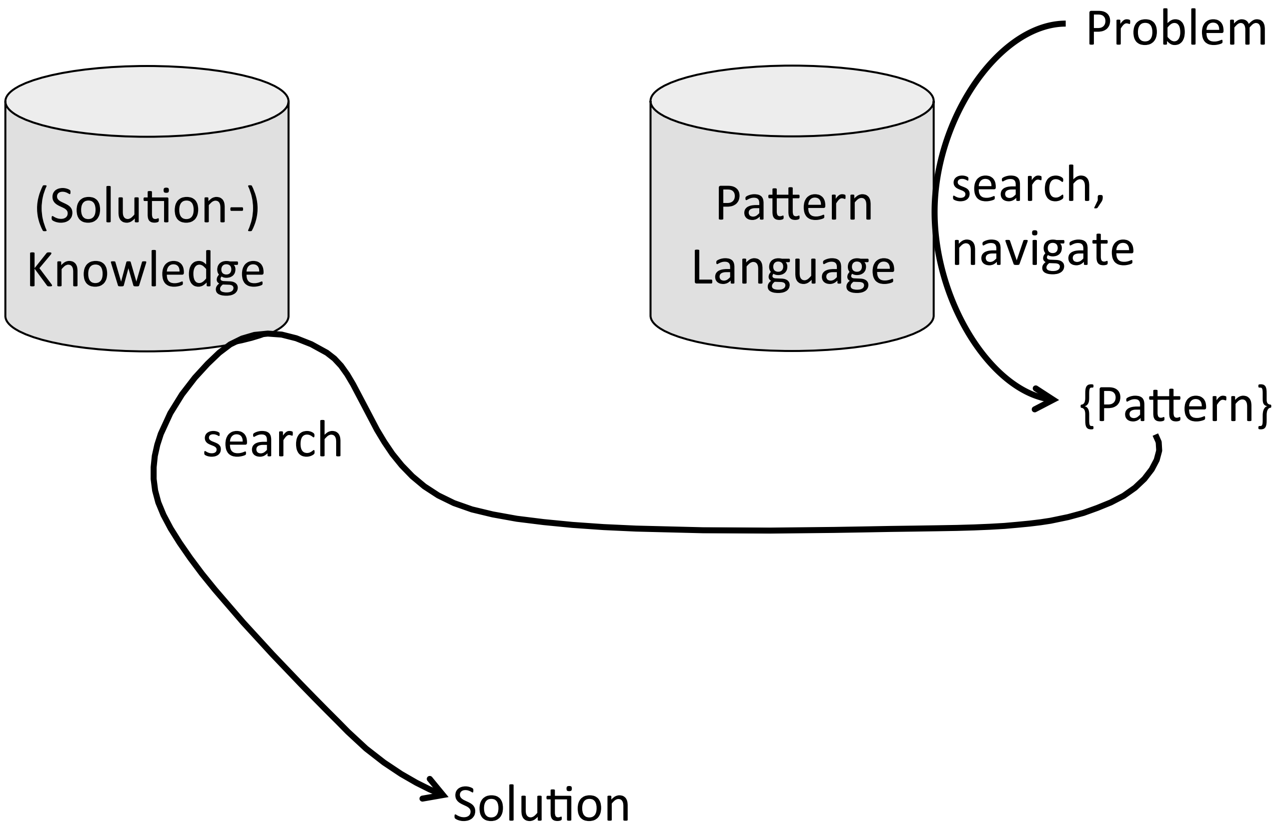
Solving Problems by Finding Patterns



From Patterns to Solutions: Today a Creative Act

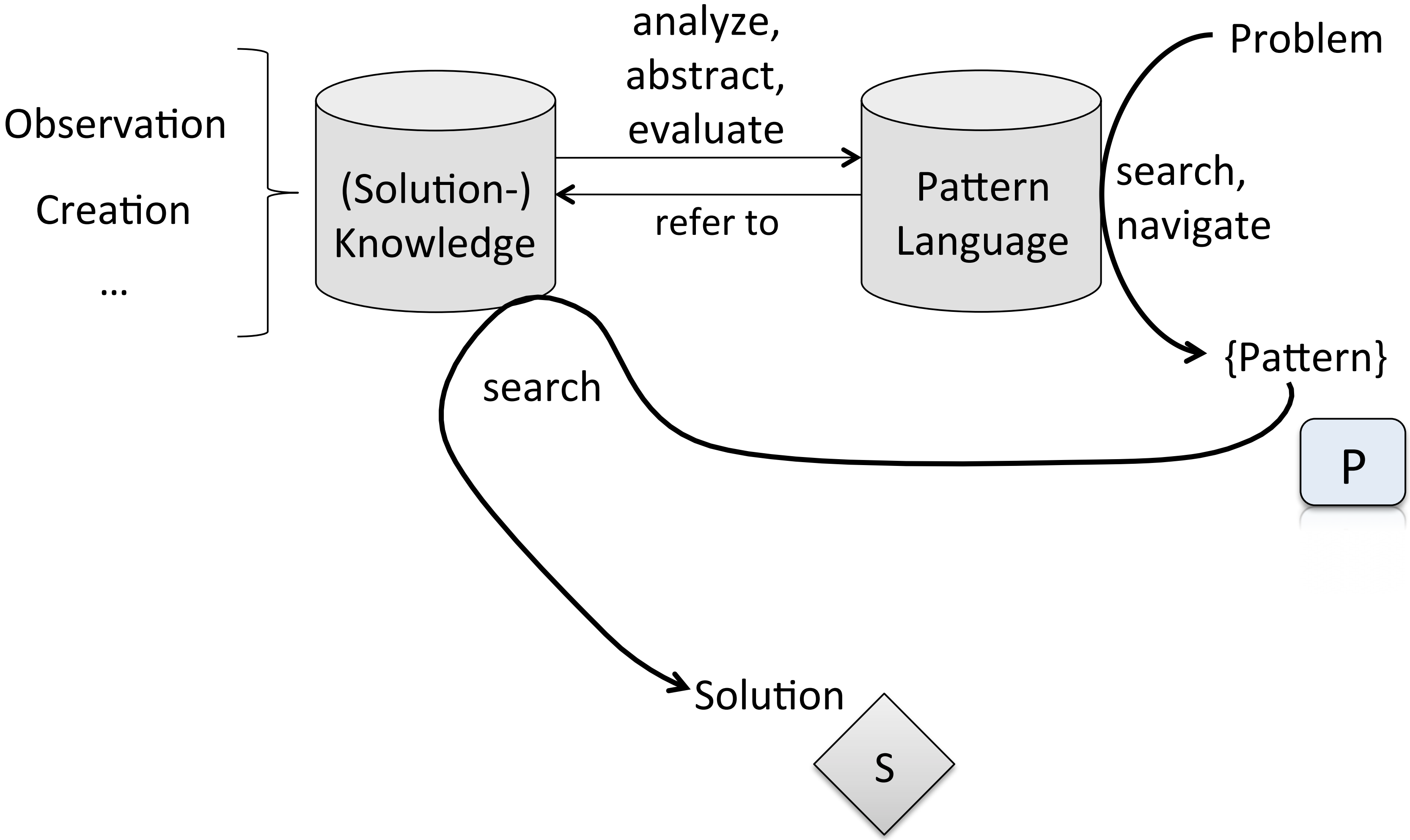


From Patterns to Solutions: Act of Recollection as Goal

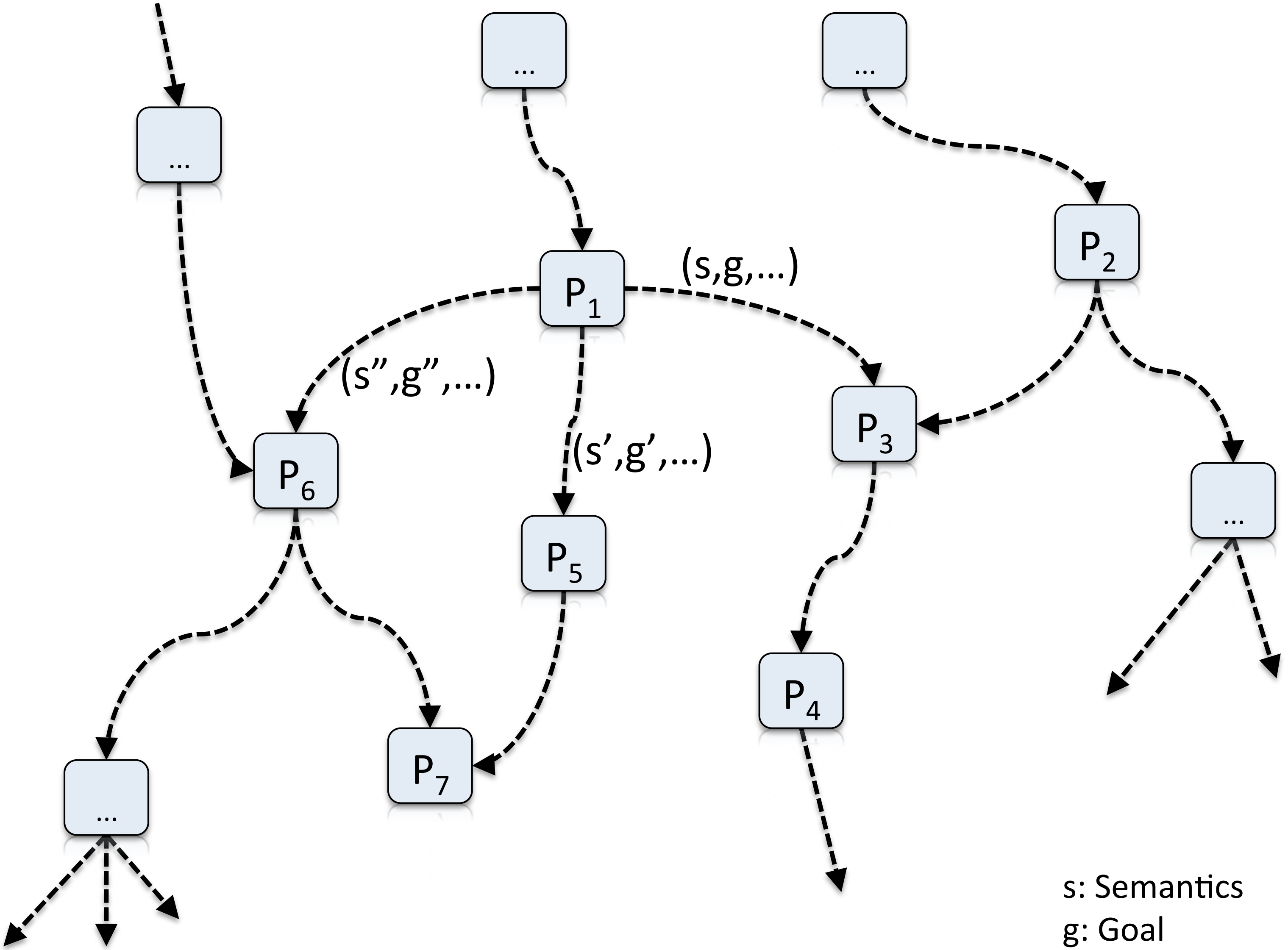


© Frank Leymann

Overall Goal

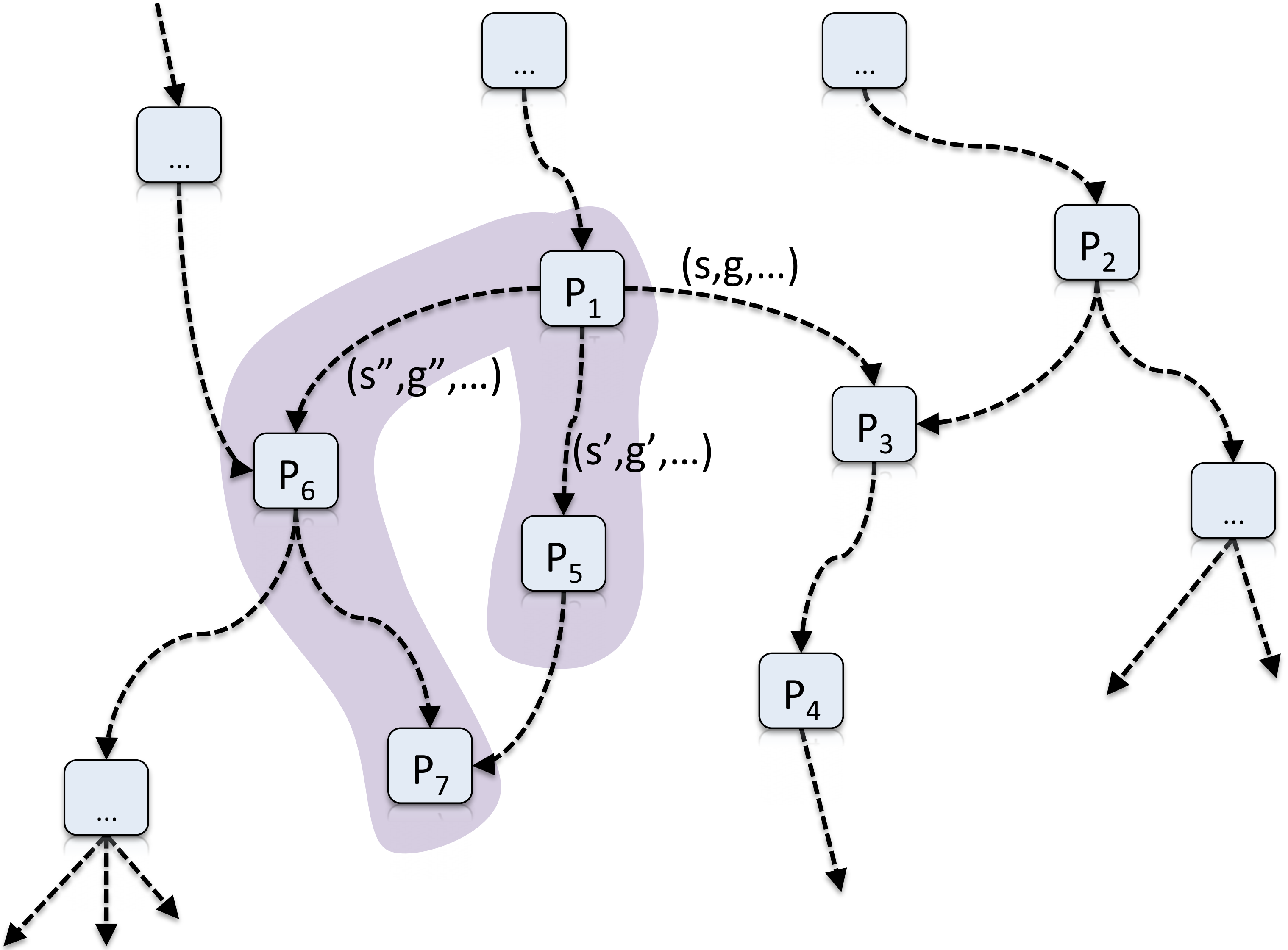


Reminder: Pattern Languages



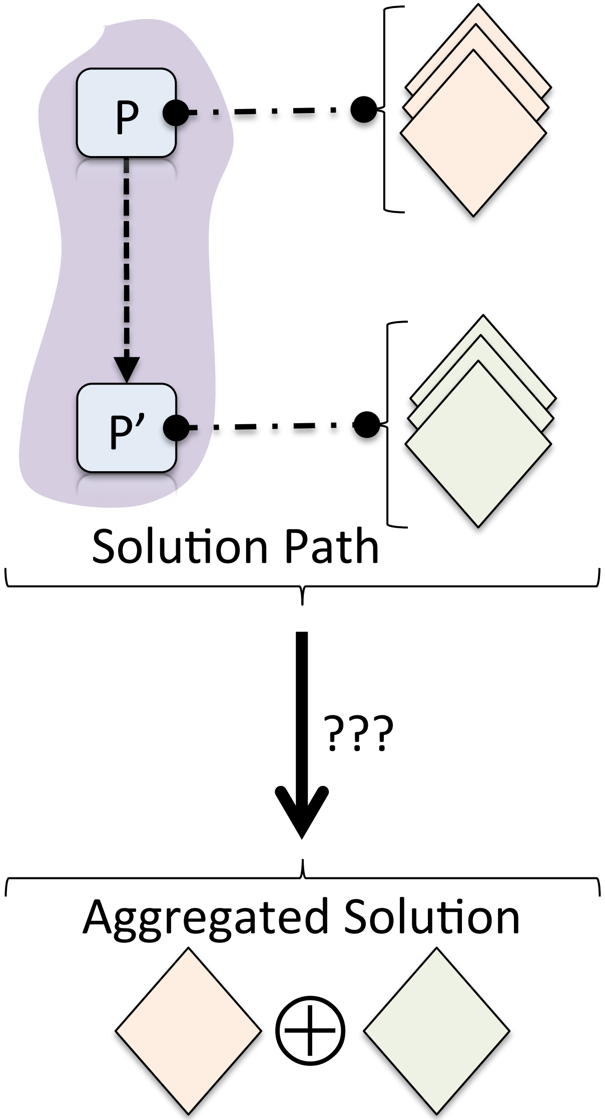
s: Semantics
g: Goal
... (further weights)

Solution Paths



© Frank Leymann

Solution Aggregation



Falkenthal, Michael; Barzen, Johanna; Breitenbücher, Uwe; Fehling, Christoph; Leymann, Frank**: From Pattern Languages to Solution Implementations.** In: Proceedings of the Sixth International Conferences on Pervasive Patterns and Applications (PATTERNS 2014) .

Agenda

Capturing Knowledge

Discovering Patterns

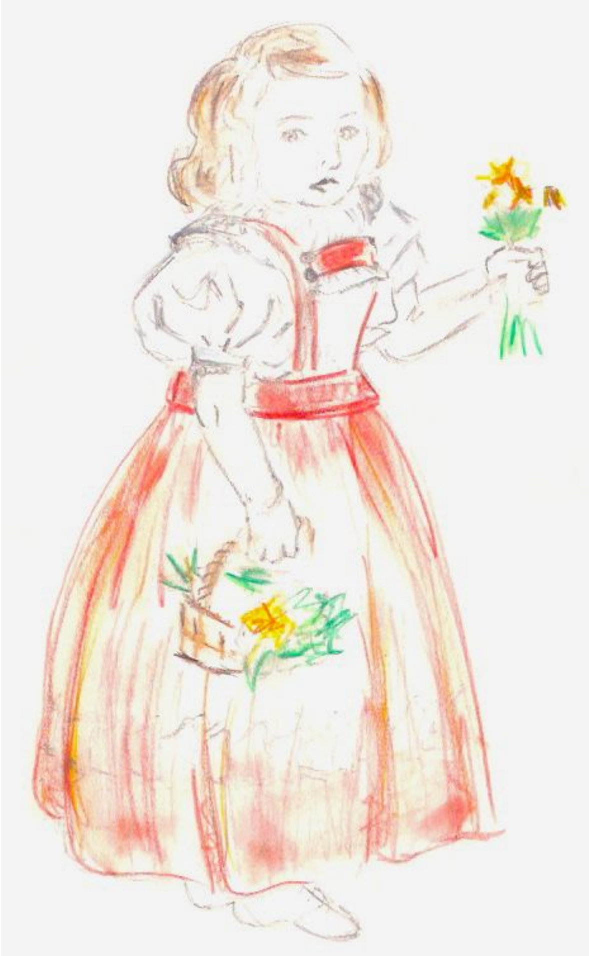
Patterns in the Humanities

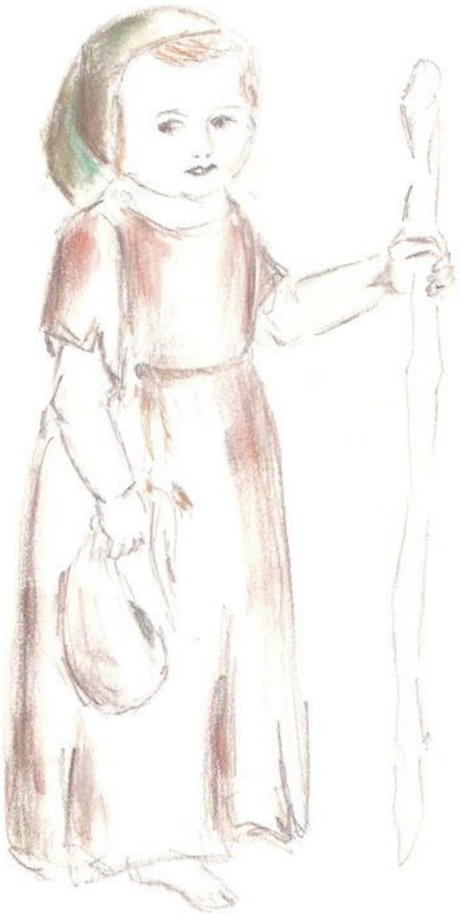
From Patterns to Solutions

Summary

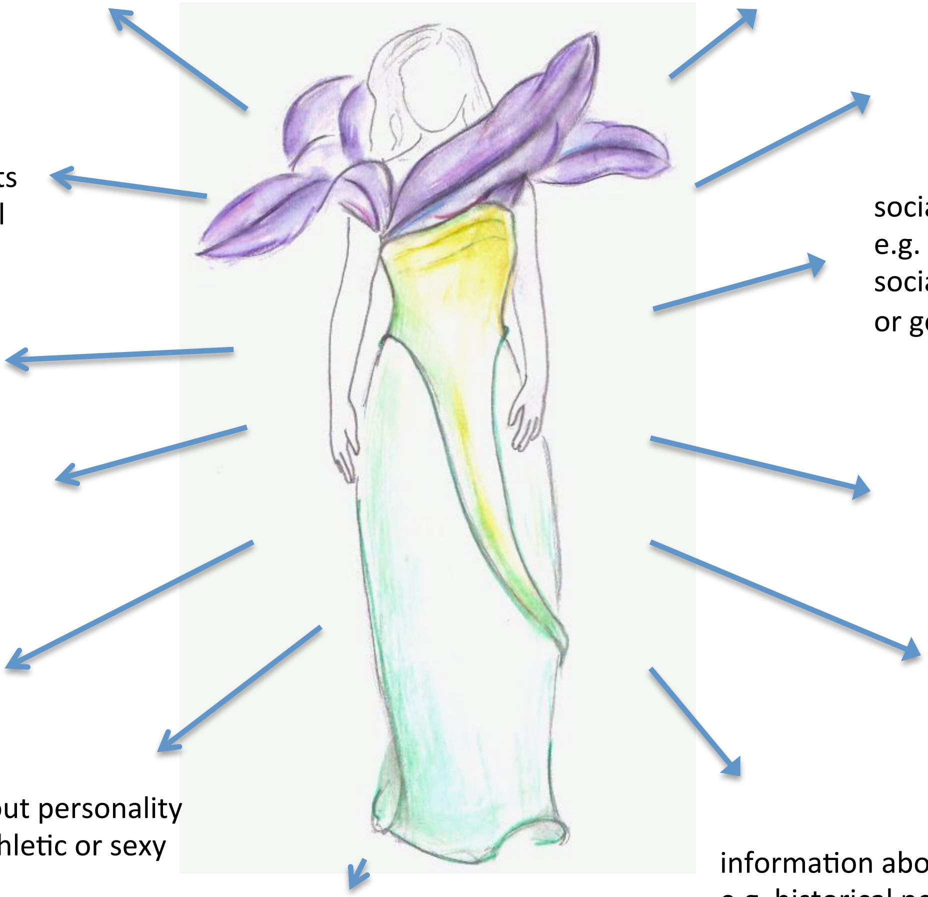
MUSE:
Towards a Pattern Language for Costumes

Vestimentary communication





Tasks and functions of costumes



protection
e.g. protection against: cold, heat,
rain, chemicals, germs, dirt

information about events
e.g. a wedding, a funeral

sign and symbol
e.g. on a uniform, on
football shirts

functional aspects
e.g. swimsuite

information about places
e.g. indoors, outdoor

indication about personality
e.g. to look athletic or sexy

signaling impact
e.g. reflective vest

producing personal identity

communication aspects
e.g. what do I want do say
about me?

social aspects
e.g. group membership,
social standing, religious
or gender classification

psychological aspects
e.g. own experience
wearing clothes

transformation
e.g. dress up, role
reversal

information about time
e.g. historical period, day
or night time

State of the Art

- Only very few scientific investigation focusing on costumes
- First serious theoretical investigation on costumes at the beginning of the 1980s
- There are three different approaches found in the literature about costumes:
  - Focusing on the aspect of costume communication based on symbolism
  - Focusing on the gender-aspect (Feministic film theory)
  - Focusing on the semiotic approach (costume as a sign)
- Current investigations are mainly based on a very small set of movies

⇨ It lacks a formal concept which enables getting general information on how costumes are used in movies and which conventions have been developed (costume language)

The term “costume language”

- The term “costume language” is usually used in a metaphorical way in the literature about costumes
- But lot of typical characters in movies can be recognized by stereotype costumes (e.g. the femme fatale or the girl next door)
- Especially when dressing the stock characters the costume designers have to use stereotype costumes to make sure these characters are well and fast understood

⇨ These ‘stereotype costumes’ can be described as patterns.
A costume pattern therefore is a proven solution to a re-occurring costume design problem.

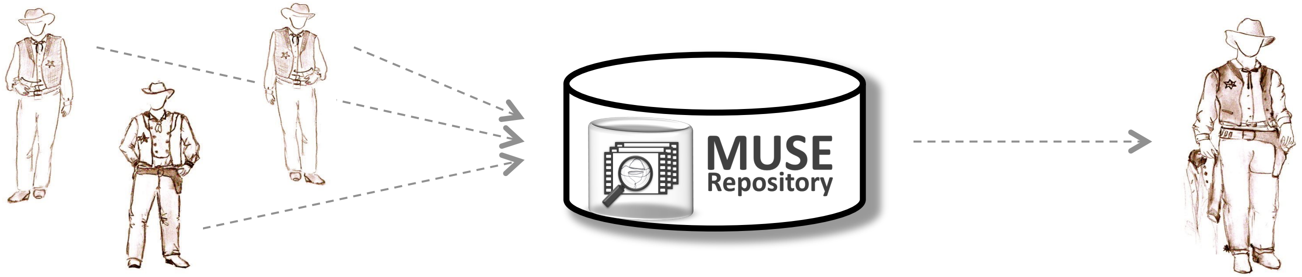
$$(\mathfrak{P}_N, \{V_\alpha\})$$

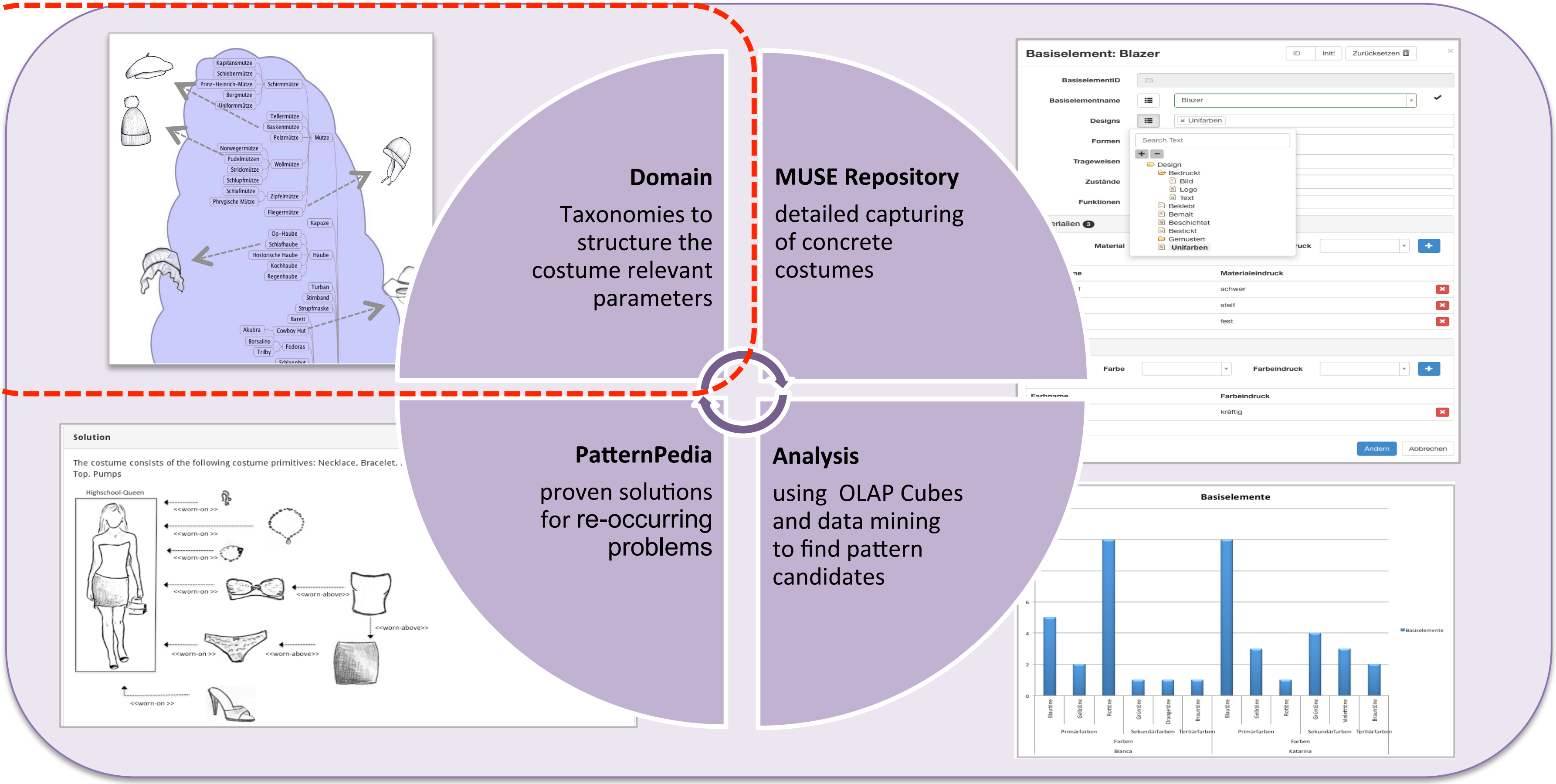
Defining patterns

- A pattern is a proven solution to a re-occurring problem
  - Concept introduced in 1977 by Ch. Alexander, a “real” architect
- A pattern is a concept that aims to capture the best solutions in an abstract way to make this knowledge reusable
  - It is not a series of concrete instructions how to solve a problem
- A pattern language is a set of patterns conforming to a particular pattern format as well as cross-references between these patterns

⇨ A costume pattern is a proven solution to a re-occurring costume design problem.

Discovering Costume Languages





Domain

Taxonomies to structure the costume relevant parameters

MUSE Repository

detailed capturing of concrete costumes

PatternPedia

proven solutions for re-occurring problems

Analysis

using OLAP Cubes and data mining to find pattern candidates

Solution

The costume consists of the following costume primitives: Necklace, Bracelet, … Top, Pumps

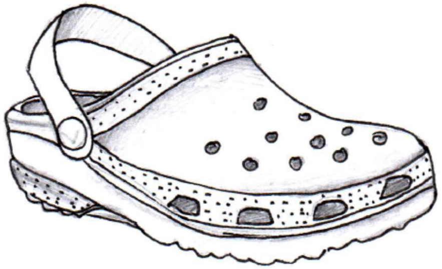
Basiselement: Blazer

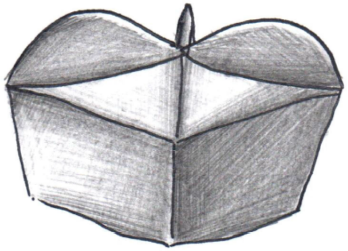
BasiselementID	23
Basiselementname	Blazer
Designs	Unifarben

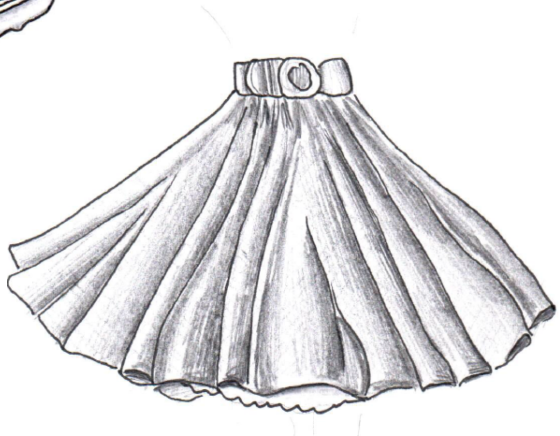
Materialeindruck
schwer
steif
fest

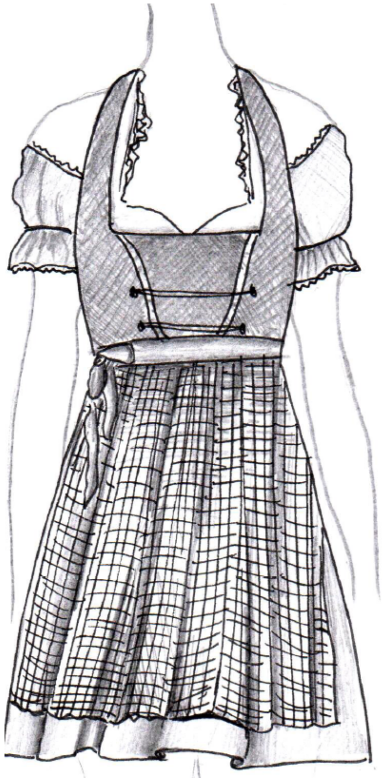
Farbname	Farbeindruck
	kräftig

Base Elements

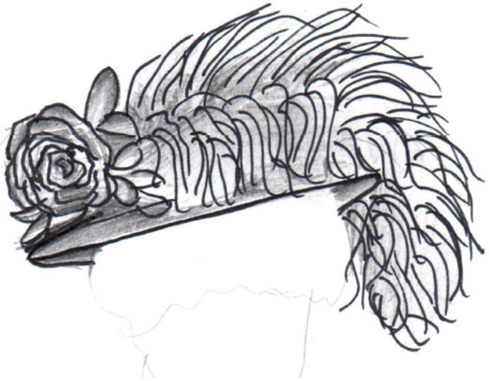








$\sum^B, L(\sum^B)$



Primitives

collar

button border

sleeve

front part

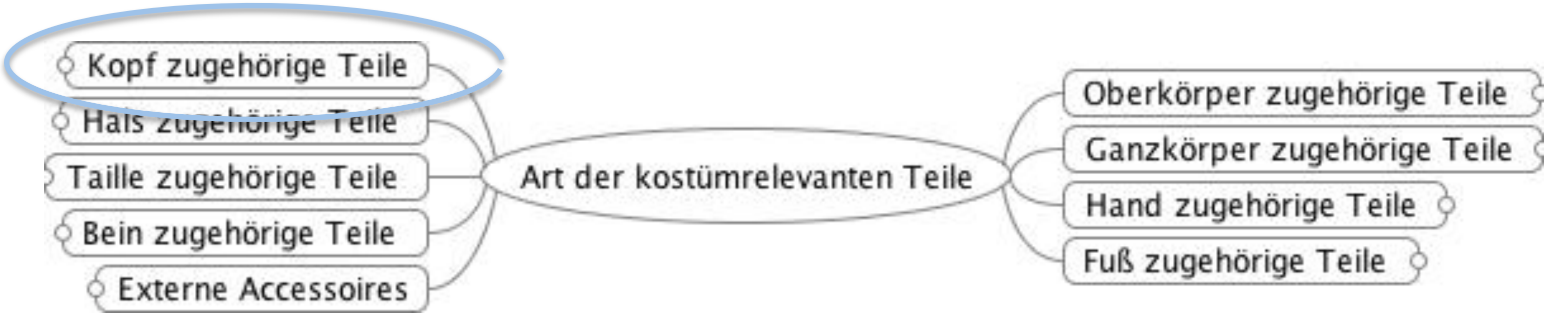
back part

$\Sigma^P, L(\Sigma^P)$

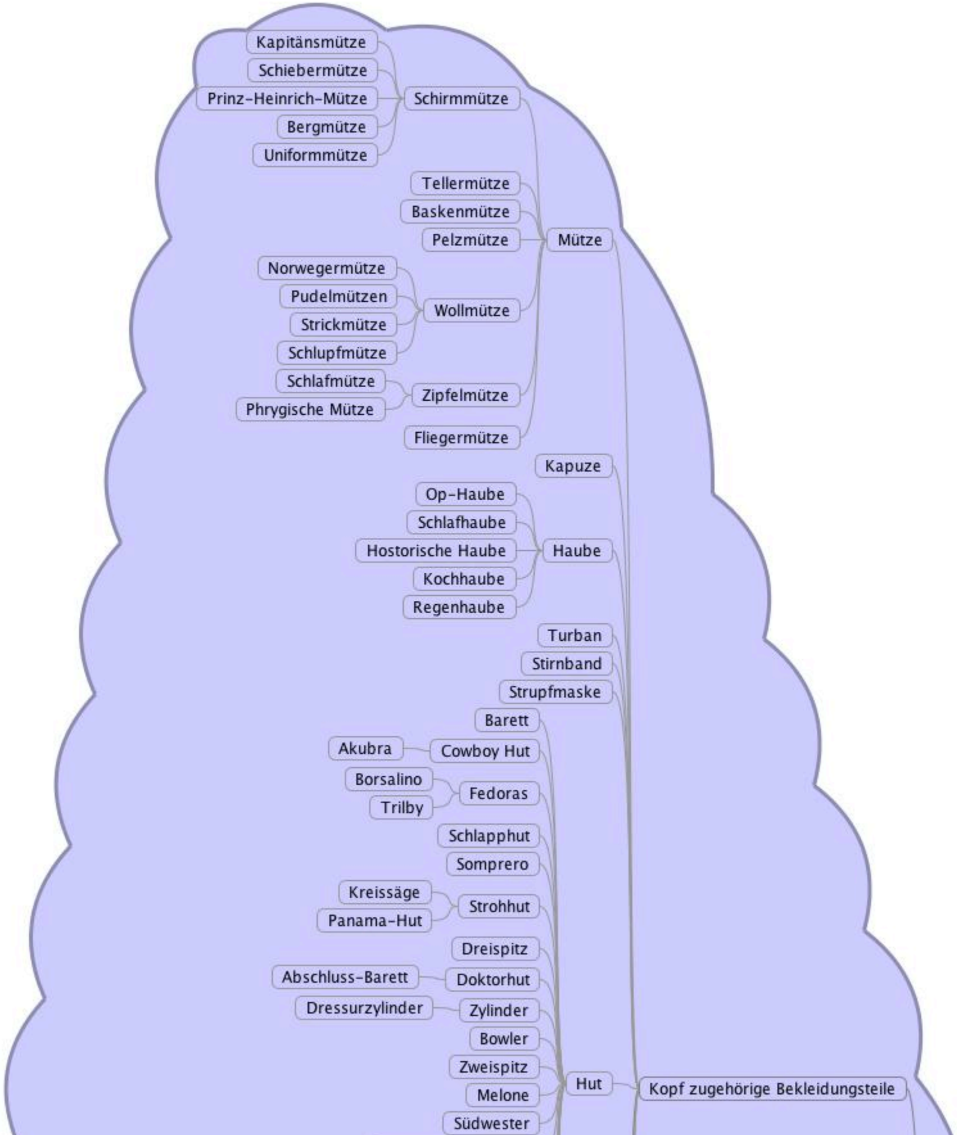
Costume Relevant Parameters

- Base Element
- Attributes
  - Primitives
  - Designs
  - Shapes
  - Material
  - Colour
- Specialties
  - Status
  - Ways of Wearing
- Function
- Body Modification

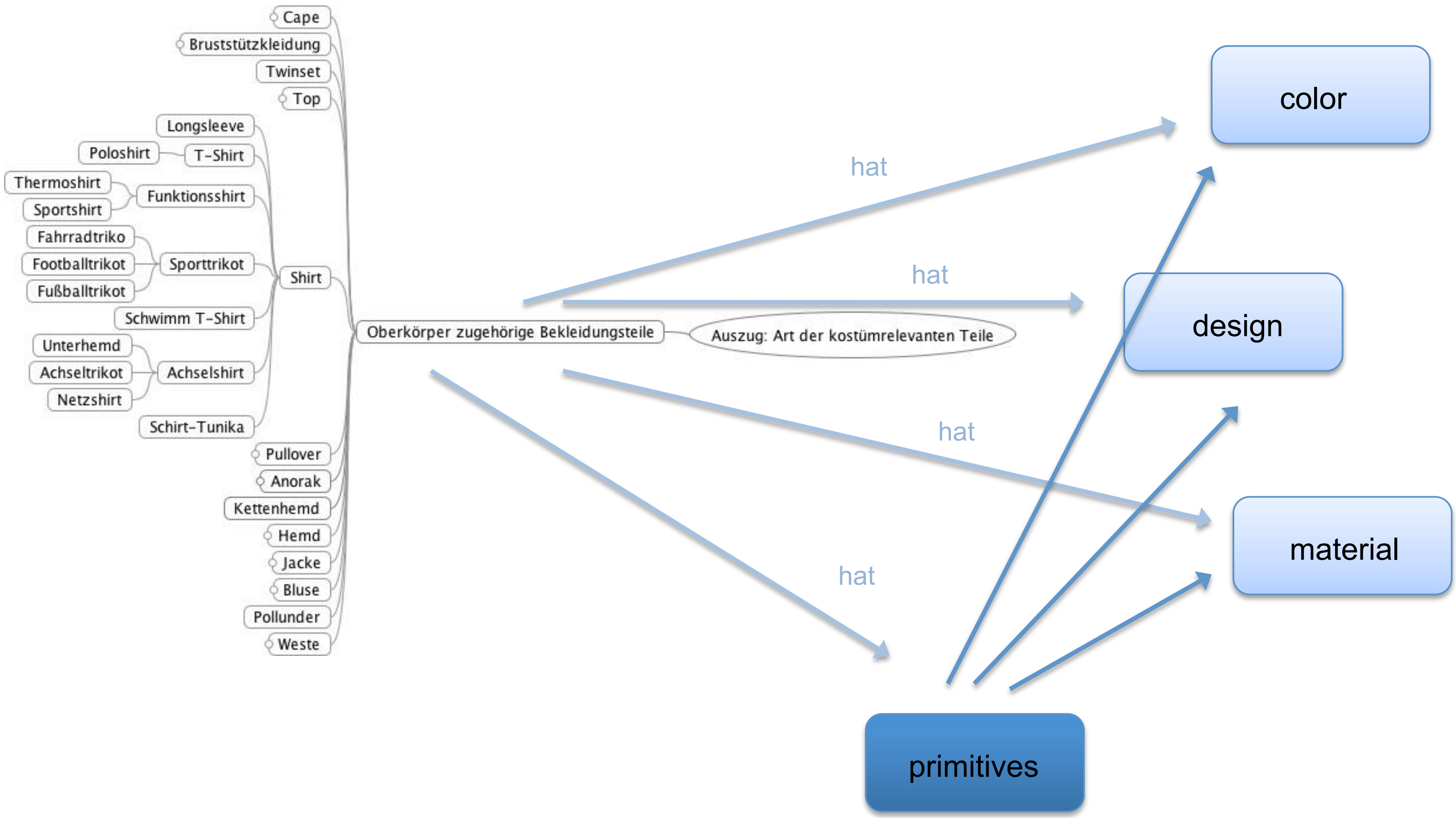
Taxonomy of the Base Elements (1)



Taxonomy of the Base Elements (2)

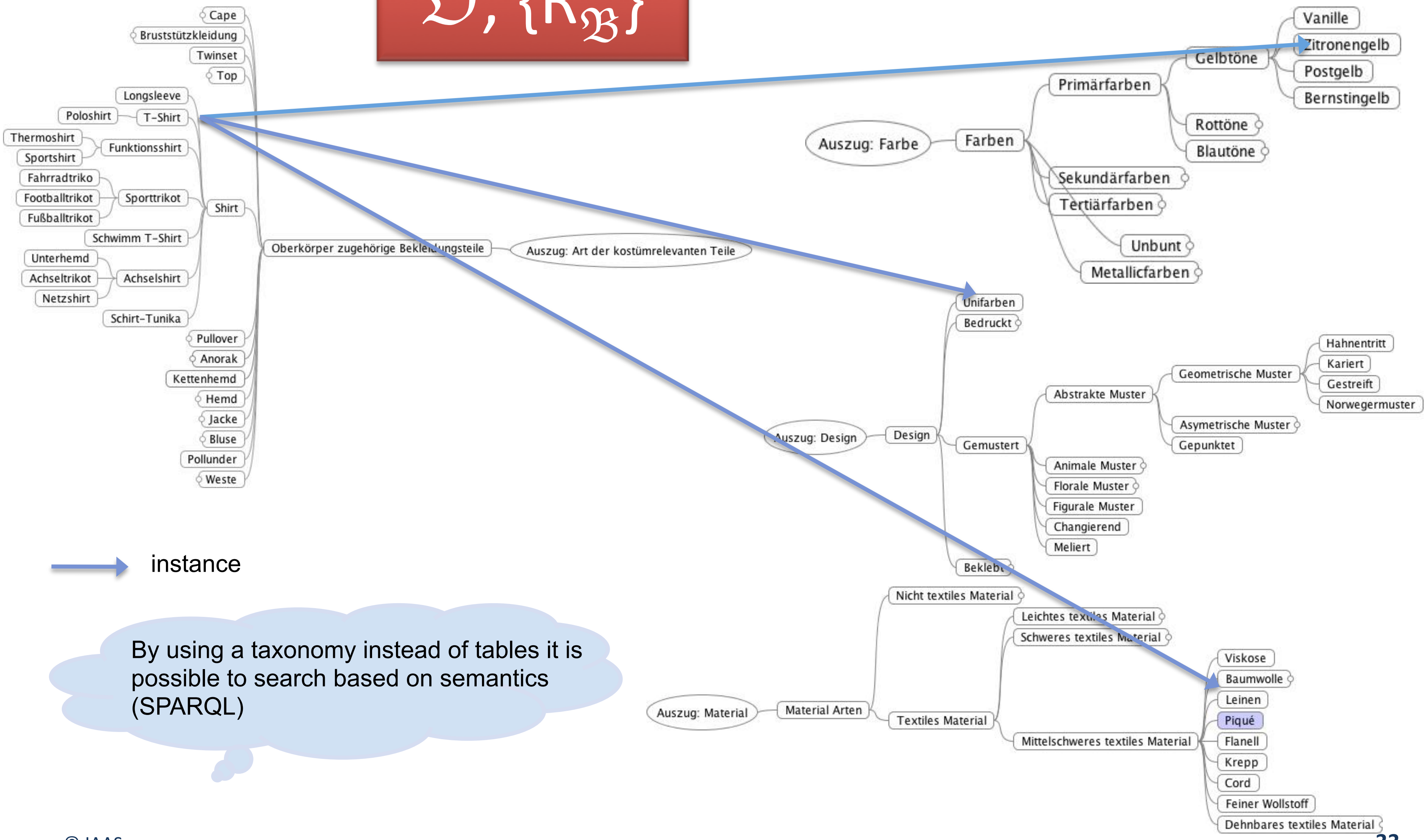


Approaching an ontology of clothes (1)



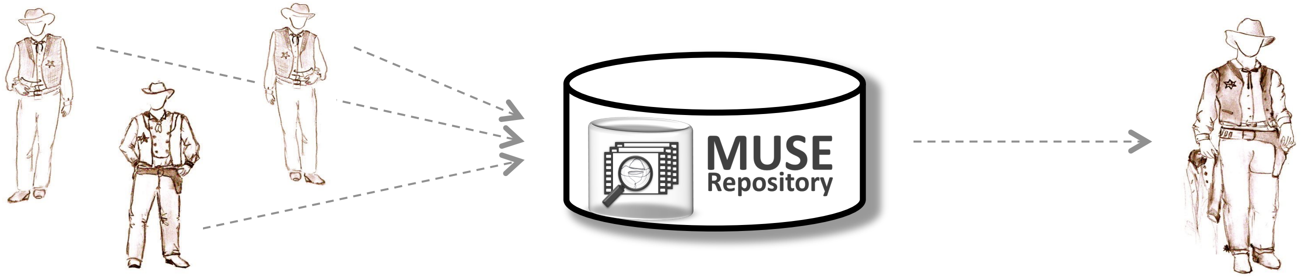
Approaching an ontology of clothes (2)

$\mathfrak{O}, \{R_{\mathfrak{B}}\}$



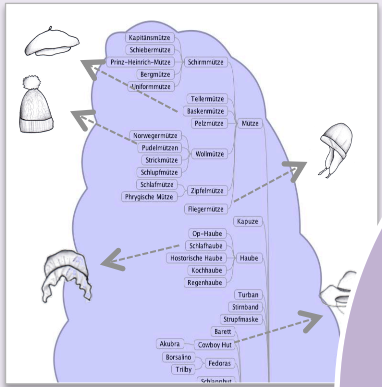
By using a taxonomy instead of tables it is possible to search based on semantics (SPARQL)

Detecting costume languages



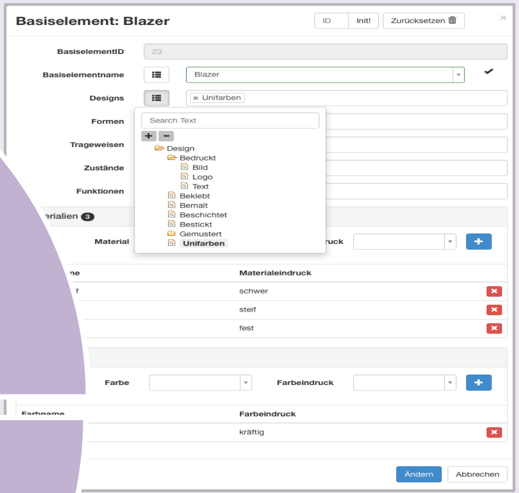
Domain

Taxonomies to structure the costume relevant parameters



MUSE Repository

detailed capturing of concrete costumes



Basiselement: Blazer

BasiselementID: 23

Basiselementname: Blazer ✓

Designs: × Unifarben

Formen

Trageweisen

Zustände

Funktionen

- Design
  - Bedruckt
    - Bild
    - Logo
    - Text
  - Beklebt
  - Bemalt
  - Beschichtet
  - Bestickt
  - Gemustert
  - **Unifarben**

erialien 3

Material | ruck | +

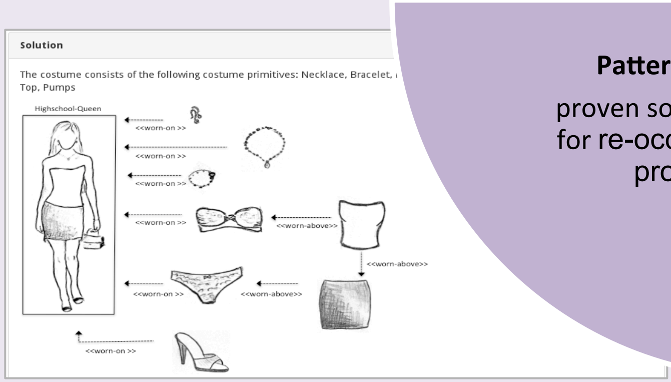
ne	Materialeindruck	
f	schwer	✖
	steif	✖
	fest	✖

Farbe | Farbeindruck | +

Farbname	Farbeindruck	
	kräftig	✖

PatternPedia

proven solutions for re-occurring problems

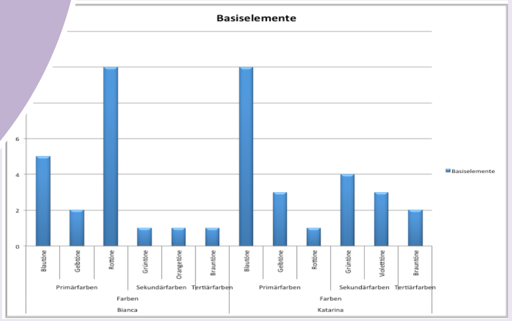


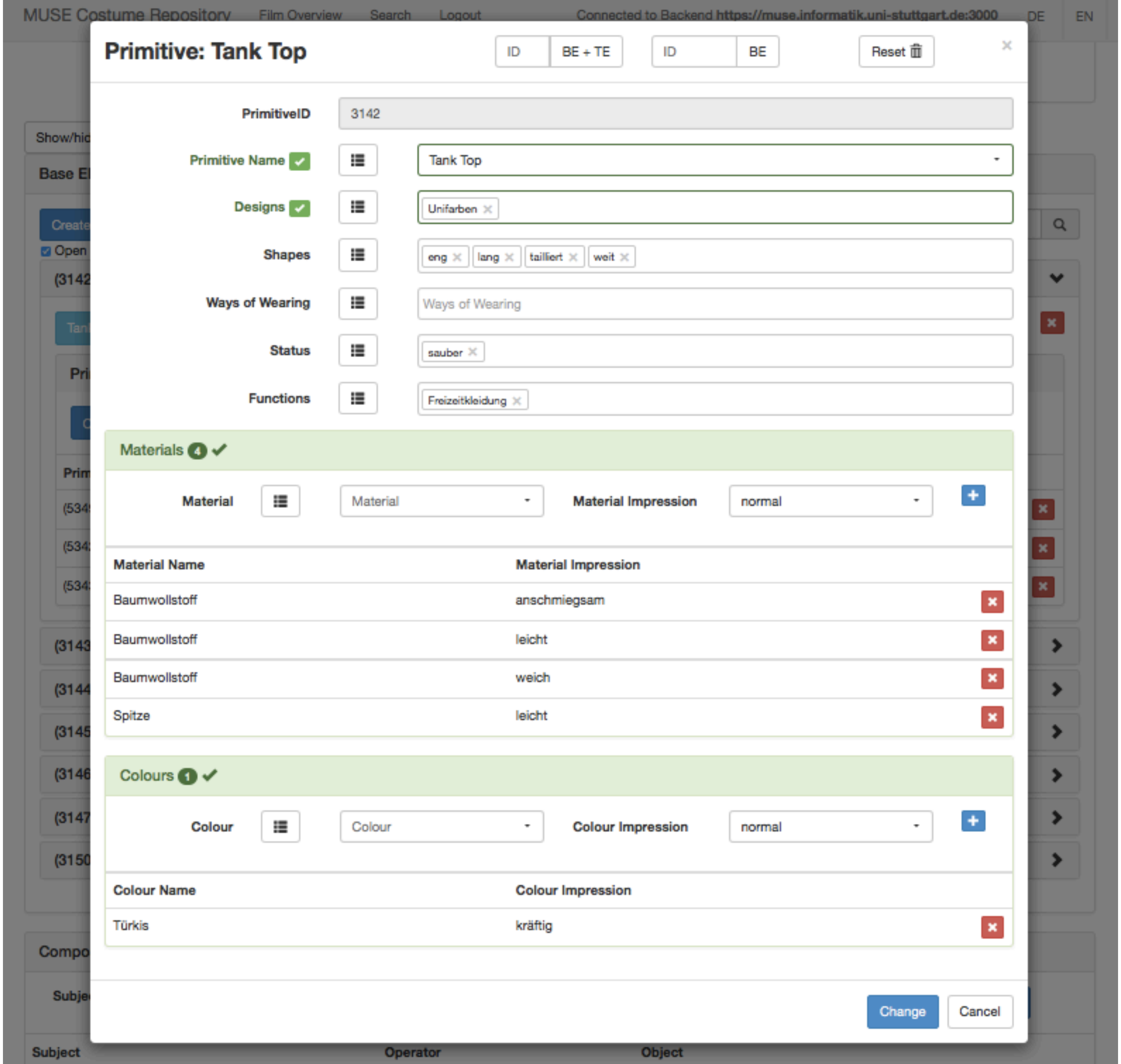
Solution

The costume consists of the following costume primitives: Necklace, Bracelet, Top, Pumps

Analysis

using OLAP Cubes and data mining to find pattern candidates





Primitive: Tank Top

PrimitiveID	3142
Primitive Name ✓	Tank Top
Designs ✓	Unifarben ×
Shapes	eng × lang × tailliert × weit ×
Ways of Wearing	Ways of Wearing
Status	sauber ×
Functions	Freizeitkleidung ×

Materials 4 ✓

Material: Material — Material Impression: normal +

Material Name	Material Impression
Baumwollstoff	anschmiegsam
Baumwollstoff	leicht
Baumwollstoff	weich
Spitze	leicht

Colours 1 ✓

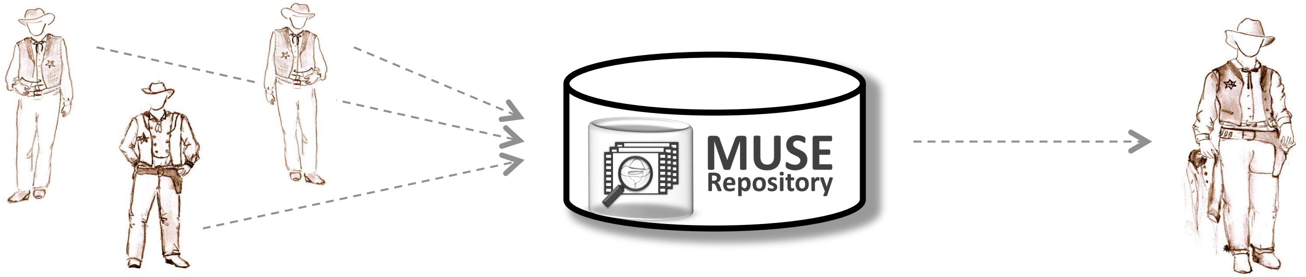
Colour: Colour — Colour Impression: normal +

Colour Name	Colour Impression
Türkis	kräftig

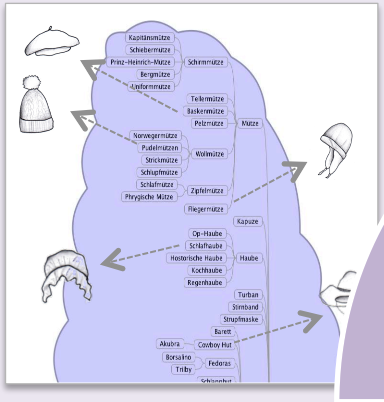
Current State (June 4th, 2016)

- Films: 23
- Costumes: 3.161
- Base Elements: 14.905
- Primitives: 29.087
- Assigned Colors: ca. 85.406
- Assigned Material: ca. 86.816

Detecting costume languages



MUSE Repository

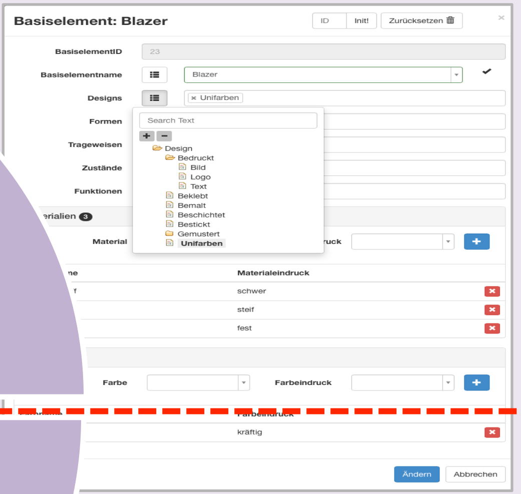


Domain

Taxonomies to structure the costume relevant parameters

MUSE Repository

detailed capturing of concrete costumes



Basiselement: Blazer

ID | Init! | Zurücksetzen

BasiselementID	23
Basiselementname	Blazer
Designs	× Unifarben
Formen	
Trageweisen	
Zustände	
Funktionen	

Search Text

- Design
  - Bedruckt
    - Bild
    - Logo
    - Text
  - Beklebt
  - Bemalt
  - Beschichtet
  - Bestickt
  - Gemustert
  - Unifarben

erialien 3

Material | ruck

ne	Materialeindruck
f	schwer
	steif
	fest

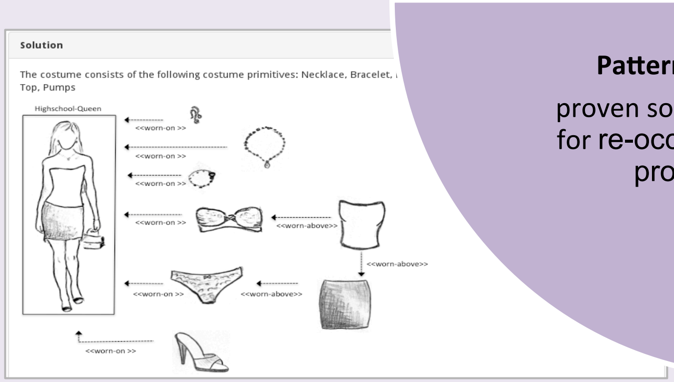
Farbe | Farbeindruck

	Farbeindruck
	kräftig

Ändern | Abbrechen

PatternPedia

proven solutions for re-occurring problems

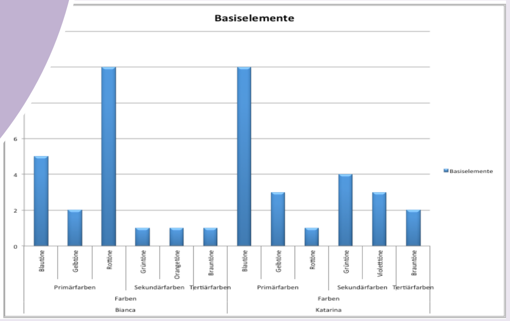


Solution

The costume consists of the following costume primitives: Necklace, Bracelet,
Top, Pumps

Analysis

using OLAP Cubes and data mining to find pattern candidates

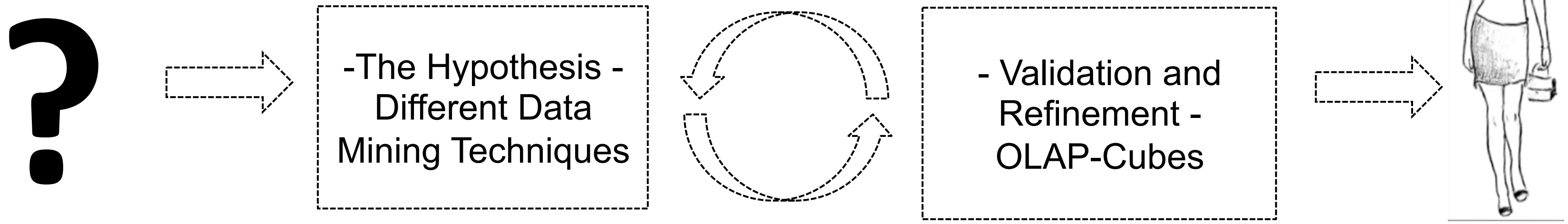


MUSE - Analytics

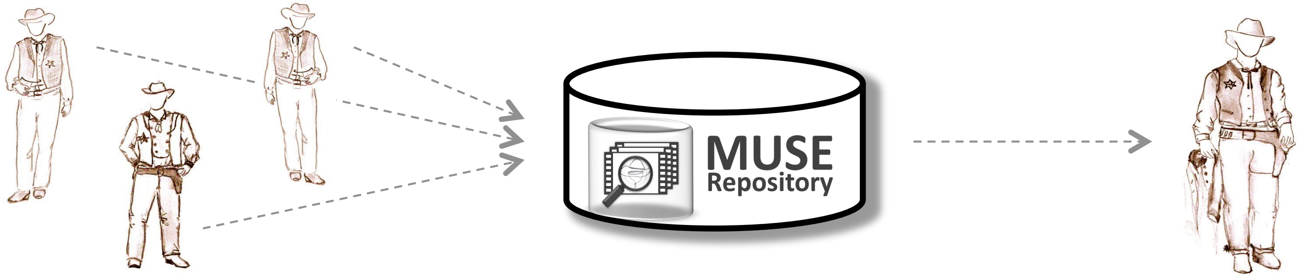
- Data Mining: support to reveal unsuspected relationships across all dimensions
- OLAP-Cubes: support multidimensional queries on the database

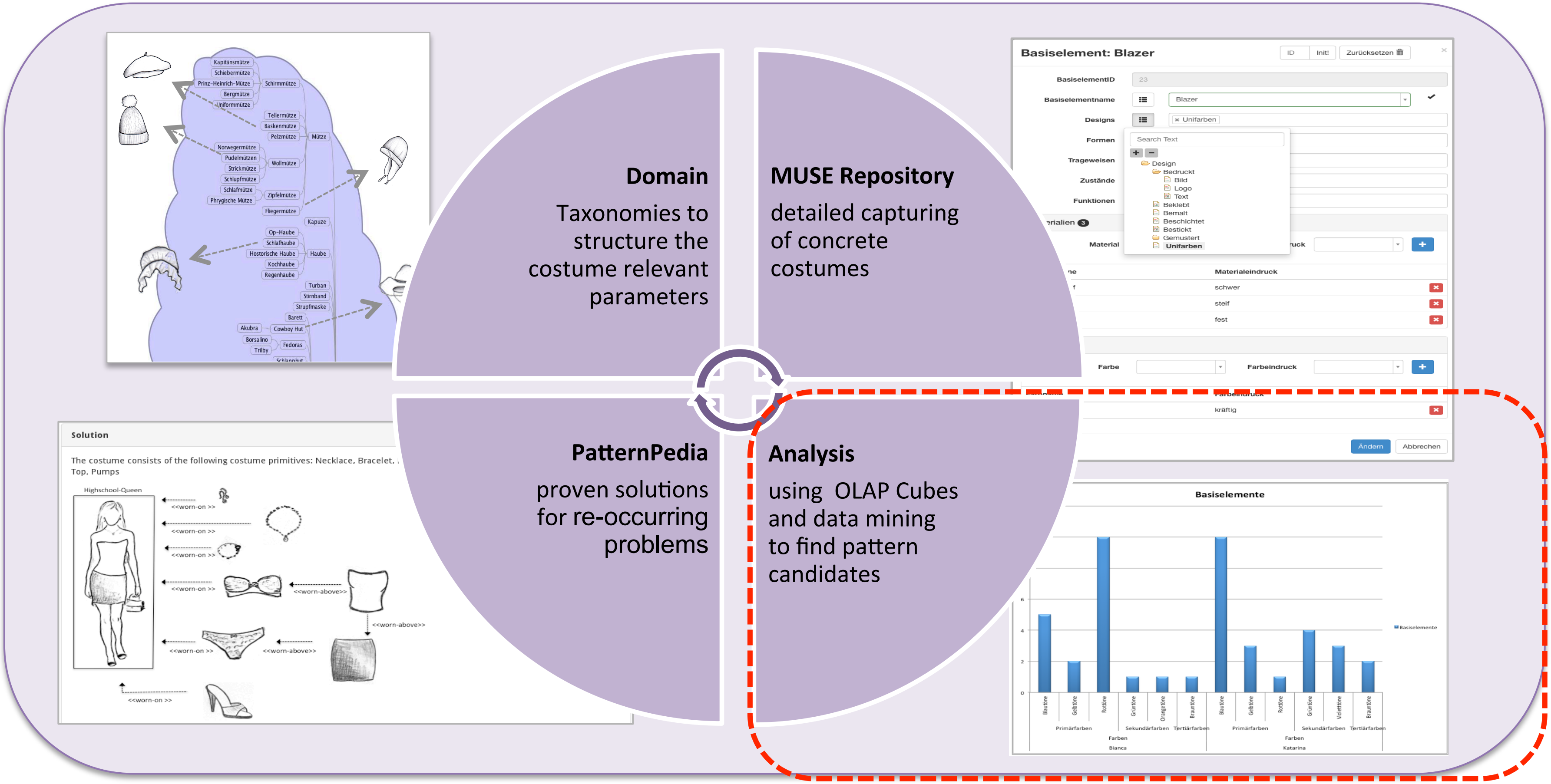
$$\mathfrak{P}_N=\mathfrak{C}/\approx$$

What base elements are typically worn by a high school queen?


?
-The Hypothesis - Different Data Mining Techniques
- Validation and Refinement - OLAP-Cubes

Detecting costume languages





Domain

Taxonomies to structure the costume relevant parameters

MUSE Repository

detailed capturing of concrete costumes

PatternPedia

proven solutions for re-occurring problems

Analysis

using OLAP Cubes and data mining to find pattern candidates

Basiselement: Blazer

ID | Init! | Zurücksetzen

BasiselementID: 23

Basiselementname: Blazer ✓

Designs: × Unifarben

Formen

Trageweisen

Zustände

Funktionen

Search Text

- Design
  - Bedruckt
    - Bild
    - Logo
    - Text
  - Beklebt
  - Bemalt
  - Beschichtet
  - Bestickt
  - Gemustert
  - Unifarben

erialien 3

Material | ruck

ne	Materialeindruck
f	schwer
	steif
	fest

Farbe | Farbeindruck

	Farbeindruck
	kräftig

Ändern | Abbrechen

Solution

The costume consists of the following costume primitives: Necklace, Bracelet, Top, Pumps

MUSE

Log in

Search Search

Costumes

High-School-Queen
Ugly Duckling
Nerd
Prom Queen
Prom King

High-School-Queen

The High-School-Queen costume stands for a beautiful girl who ´rules´ the high school.

The High-School-Queen costume stands for the girl who 'rules' the high school. She is beautiful on one side and mean on the other side. Therefore, everyone wants so be her friend.

References

Related Patterns

- Ugly Duckling
- Nerd

Consider Next

- Prom Queen
- Prom King

Known Uses

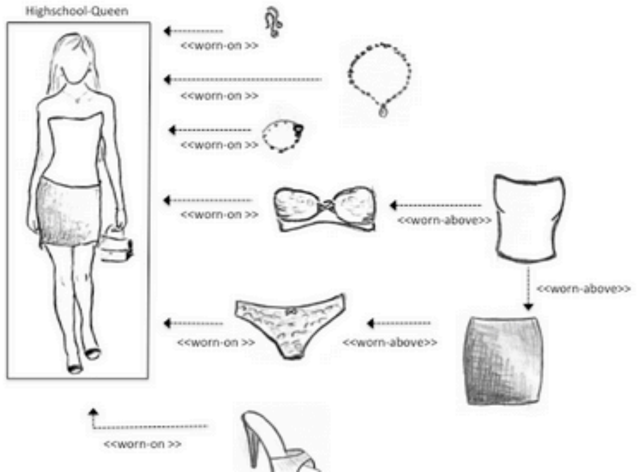
- Eine wie Keine - Taylor Vaughn - Schul-Outfit 1
- Eine wie Keine - Taylor Vaughn - Alltagsoutfit 4
- Ungeküsst - Kirstin 1 - Freizeitoutfit 2
- Ungeküsst - Kirstin 1 - Freizeitoutfit 3
- Ungeküsst - Kirstin 2 - Freizeitoutfit 3

Context

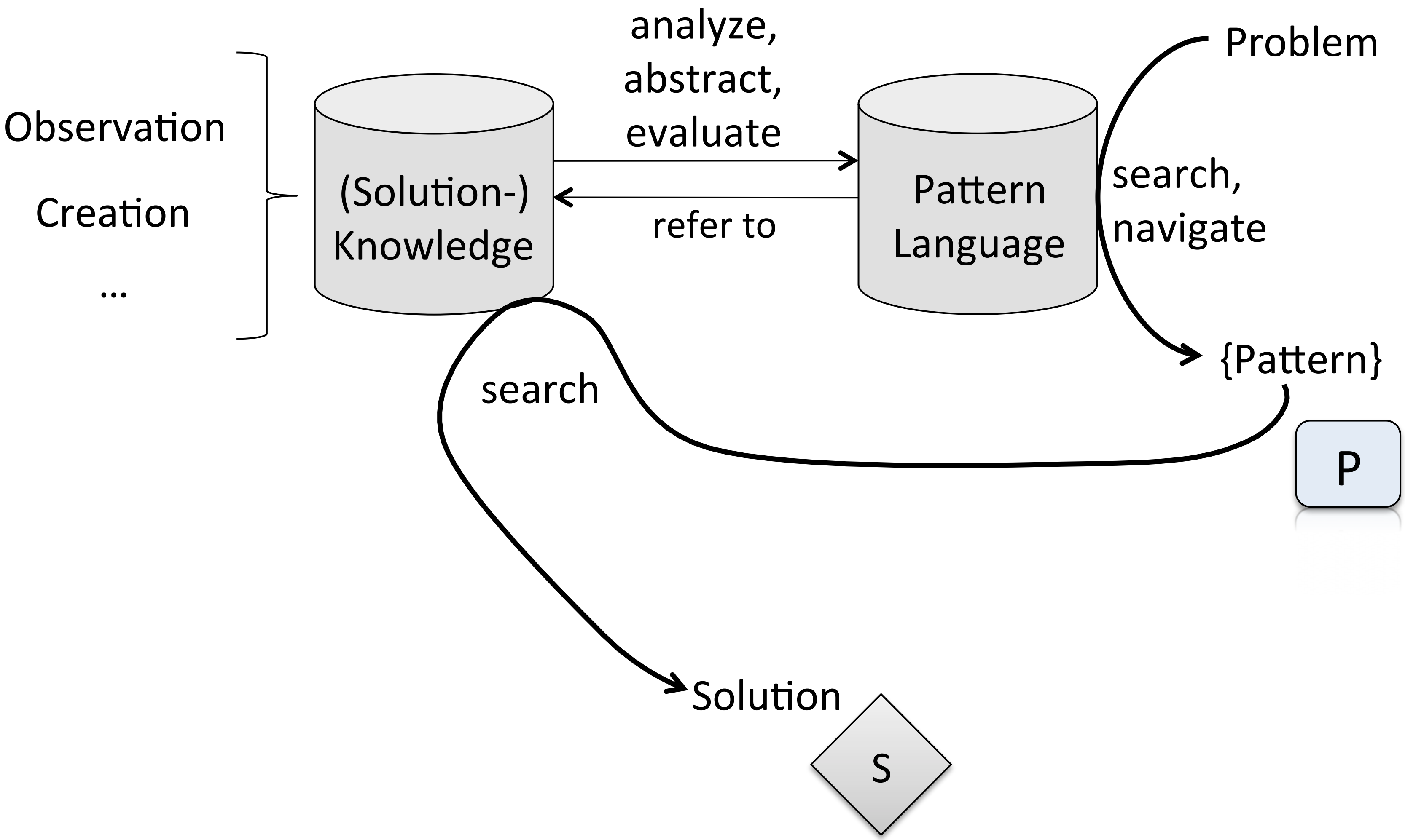
The High-School-Queen occurs in films sett at a high school; Genre: High School Comedy.

Solution

The costume consists of the following costume primitives: Necklace, Bracelet, Earrings, Underpants, Strapless Bra, Mini Skirt, tight Top, Pumps



Overall Goal

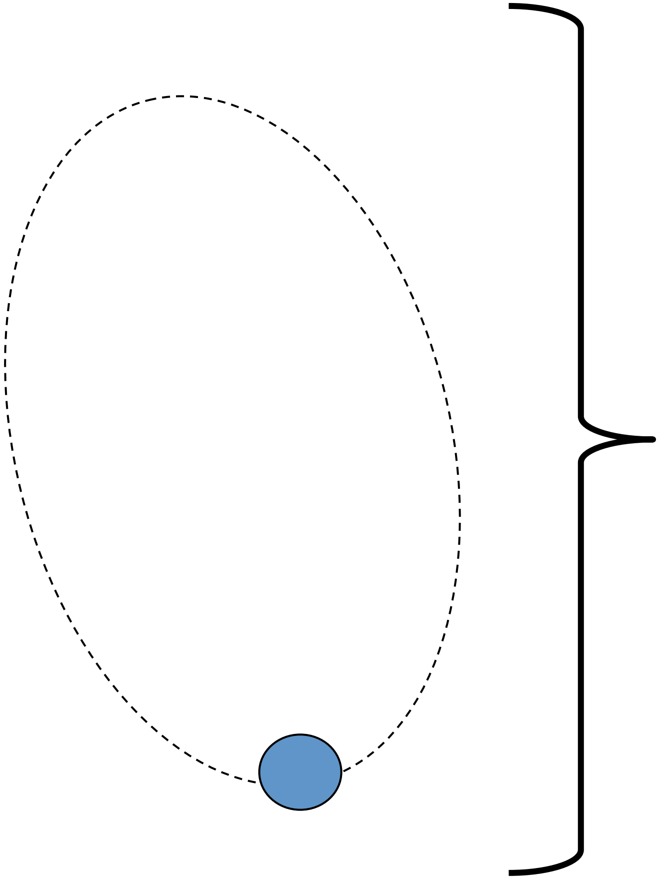


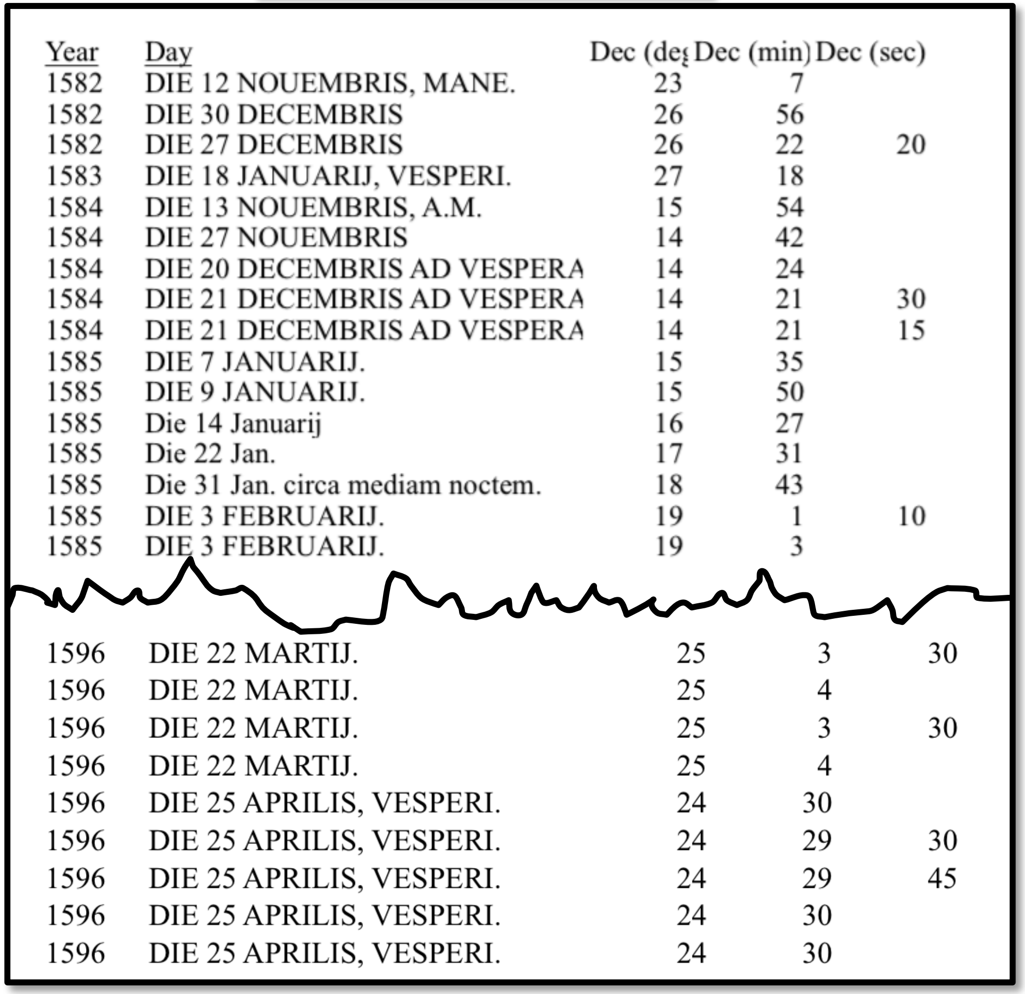
The MUSE Method
and the Scientific Paradigms

Tycho Brahe: Observation & Data Capture

Observation

Data Capturing



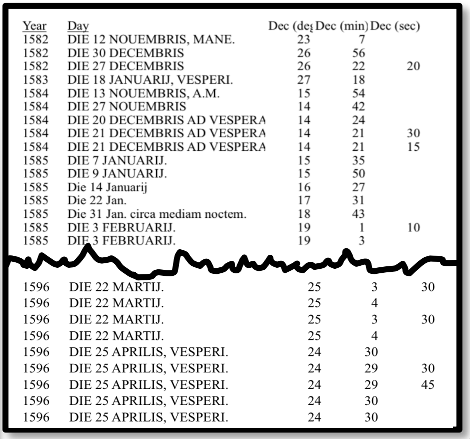


Year	Day	Dec (deg	Dec (min)	Dec (sec)
1582	DIE 12 NOUEMBRIS, MANE.	23	7	
1582	DIE 30 DECEMBRIS	26	56	
1582	DIE 27 DECEMBRIS	26	22	20
1583	DIE 18 JANUARIJ, VESPERI.	27	18	
1584	DIE 13 NOUEMBRIS, A.M.	15	54	
1584	DIE 27 NOUEMBRIS	14	42	
1584	DIE 20 DECEMBRIS AD VESPERA	14	24	
1584	DIE 21 DECEMBRIS AD VESPERA	14	21	30
1584	DIE 21 DECEMBRIS AD VESPERA	14	21	15
1585	DIE 7 JANUARIJ.	15	35	
1585	DIE 9 JANUARIJ.	15	50	
1585	Die 14 Januarij	16	27	
1585	Die 22 Jan.	17	31	
1585	Die 31 Jan. circa mediam noctem.	18	43	
1585	DIE 3 FEBRUARIJ.	19	1	10
1585	DIE 3 FEBRUARIJ.	19	3	
1596	DIE 22 MARTIJ.	25	3	30
1596	DIE 22 MARTIJ.	25	4	
1596	DIE 22 MARTIJ.	25	3	30
1596	DIE 22 MARTIJ.	25	4	
1596	DIE 25 APRILIS, VESPERI.	24	30	
1596	DIE 25 APRILIS, VESPERI.	24	29	30
1596	DIE 25 APRILIS, VESPERI.	24	29	45
1596	DIE 25 APRILIS, VESPERI.	24	30	
1596	DIE 25 APRILIS, VESPERI.	24	30	

(Table and pictures from Wikipedia)

Johannes Kepler: Analyzing Data & Formalization

Data Analysis



Year	Day	Dec (deg	Dec (min)	Dec (sec)
1582	DIE 12 NOUEMBRIS, MANE.	23	7	
1582	DIE 30 DECEMBRIS	26	56	
1582	DIE 27 DECEMBRIS	26	22	20
1583	DIE 18 JANUARIJ, VESPERI.	27	18	
1584	DIE 13 NOUEMBRIS, A.M.	15	54	
1584	DIE 27 NOUEMBRIS	14	42	
1584	DIE 20 DECEMBRIS AD VESPERA	14	24	
1584	DIE 21 DECEMBRIS AD VESPERA	14	21	30
1584	DIE 21 DECEMBRIS AD VESPERA	14	21	15
1585	DIE 7 JANUARIJ.	15	35	
1585	DIE 9 JANUARIJ.	15	50	
1585	Die 14 Januarij	16	27	
1585	Die 22 Jan.	17	31	
1585	Die 31 Jan. circa mediam noctem.	18	43	
1585	DIE 3 FEBRUARIJ.	19	1	10
1585	DIE 3 FEBRUARIJ.	19	3	
1596	DIE 22 MARTIJ.	25	3	30
1596	DIE 22 MARTIJ.	25	4	
1596	DIE 22 MARTIJ.	25	3	30
1596	DIE 22 MARTIJ.	25	4	
1596	DIE 25 APRILIS, VESPERI.	24	30	
1596	DIE 25 APRILIS, VESPERI.	24	29	30
1596	DIE 25 APRILIS, VESPERI.	24	29	45
1596	DIE 25 APRILIS, VESPERI.	24	30	
1596	DIE 25 APRILIS, VESPERI.	24	30	

1. The orbit of every planet is an ellipse with the sun in one of the two foci.
2. A line joining a planet and the Sun sweeps out equal areas during equal intervals of time
3. The square of the orbital period of a planet is directly proportional to the cube of the semi-major axis of its orbit

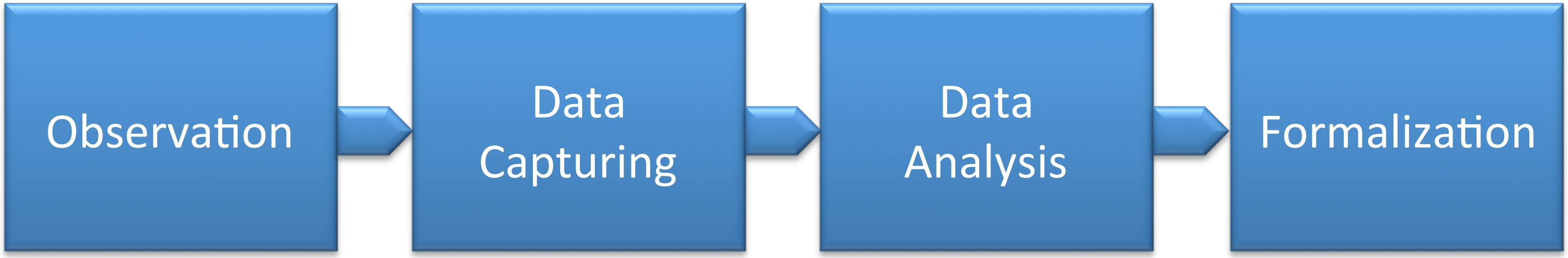
1. $r(\phi) = \dfrac{p}{1 + \varepsilon \cdot \cos\phi}$

2. $r^2\, d\theta = abn\, dt$

3. $\left(\dfrac{T_1}{T_2}\right)^2 = \left(\dfrac{a_1}{a_2}\right)^3$

Formalization

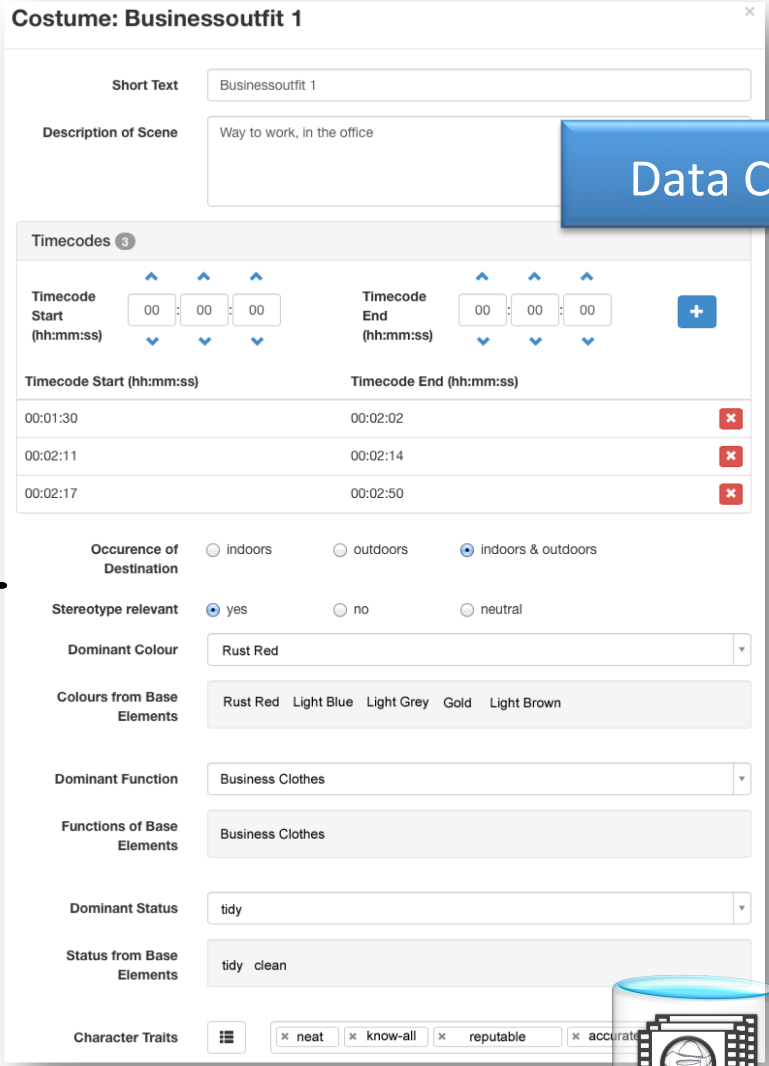
The Scientific Method



Capturing Movies

Observation

Data Capturing



Costume: Businessoutfit 1

Short Text: Businessoutfit 1

Description of Scene: Way to work, in the office

Timecodes 3

Timecode Start (hh:mm:ss) 00 : 00 : 00 **Timecode End (hh:mm:ss)** 00 : 00 : 00 +

Timecode Start (hh:mm:ss)	Timecode End (hh:mm:ss)	
00:01:30	00:02:02	✖
00:02:11	00:02:14	✖
00:02:17	00:02:50	✖

Occurence of Destination: ○ indoors ○ outdoors ● indoors & outdoors

Stereotype relevant: ● yes ○ no ○ neutral

Dominant Colour: Rust Red

Colours from Base Elements: Rust Red Light Blue Light Grey Gold Light Brown

Dominant Function: Business Clothes

Functions of Base Elements: Business Clothes

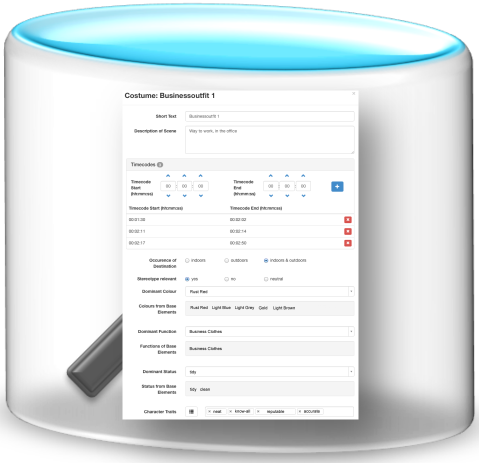
Dominant Status: tidy

Status from Base Elements: tidy clean

Character Traits: × neat × know-all × reputable × accurate

MUSE Repository

Analyzing Movies



Costume: Businessoutfit 1

Short Text	Businessoutfit 1
Description of Scene	Way to work, in the office

Timecodes 3

Timecode Start (hh:mm:ss) 00 : 00 : 00 Timecode End (hh:mm:ss) 00 : 00 : 00 +

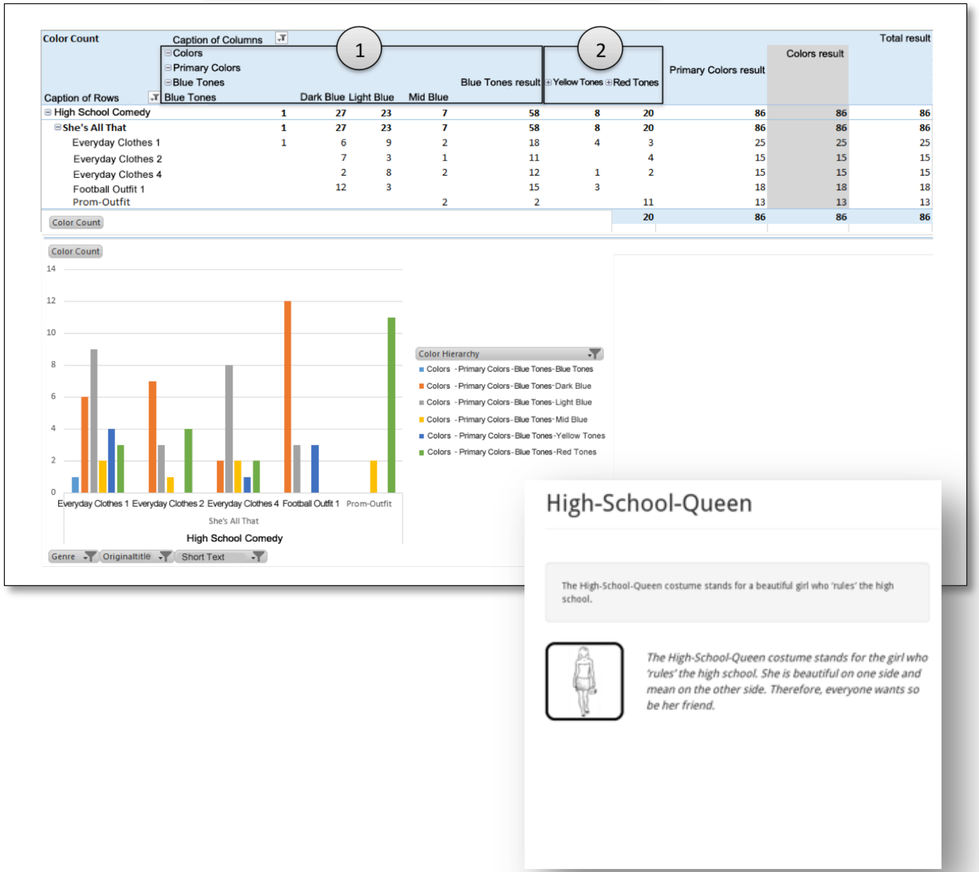
Timecode Start (hh:mm:ss)	Timecode End (hh:mm:ss)
00:01:30	00:02:02
00:02:11	00:02:14
00:02:17	00:02:50

Occurence of Destination	○ indoors ○ outdoors ● indoors & outdoors
Stereotype relevant	● yes ○ no ○ neutral
Dominant Colour	Rust Red
Colours from Base Elements	Rust Red Light Blue Light Grey Gold Light Brown
Dominant Function	Business Clothes
Functions of Base Elements	Business Clothes
Dominant Status	tidy
Status from Base Elements	tidy clean
Character Traits	× neat × know-all × reputable × accurate

MUSE

Repository

Data Analysis



Color Count — Caption of Columns: Colors / Primary Colors / Blue Tones

Caption of Rows	Blue Tones	Dark Blue	Light Blue	Mid Blue	Blue Tones result	Yellow Tones	Red Tones	Primary Colors result	Colors result	Total result
High School Comedy	**1**	**27**	**23**	**7**	**58**	**8**	**20**	**86**	**86**	**86**
She's All That	**1**	**27**	**23**	**7**	**58**	**8**	**20**	**86**	**86**	**86**
Everyday Clothes 1	1	6	9	2	18	4	3	25	25	25
Everyday Clothes 2		7	3	1	11		4	15	15	15
Everyday Clothes 4		2	8	2	12	1	2	15	15	15
Football Outfit 1		12	3		15	3		18	18	18
Prom-Outfit				2	2		11	13	13	13
Color Count							**20**	**86**	**86**	**86**

1
2
Color Count
14
12
10
8
6
4
2
0
Everyday Clothes 1
Everyday Clothes 2
Everyday Clothes 4
Football Outfit 1
Prom-Outfit
She's All That
High School Comedy
Genre
Originaltitle
Short Text
Color Hierarchy
Colors - Primary Colors - Blue Tones - Blue Tones
Colors - Primary Colors - Blue Tones - Dark Blue
Colors - Primary Colors - Blue Tones - Light Blue
Colors - Primary Colors - Blue Tones - Mid Blue
Colors - Primary Colors - Blue Tones - Yellow Tones
Colors - Primary Colors - Blue Tones - Red Tones

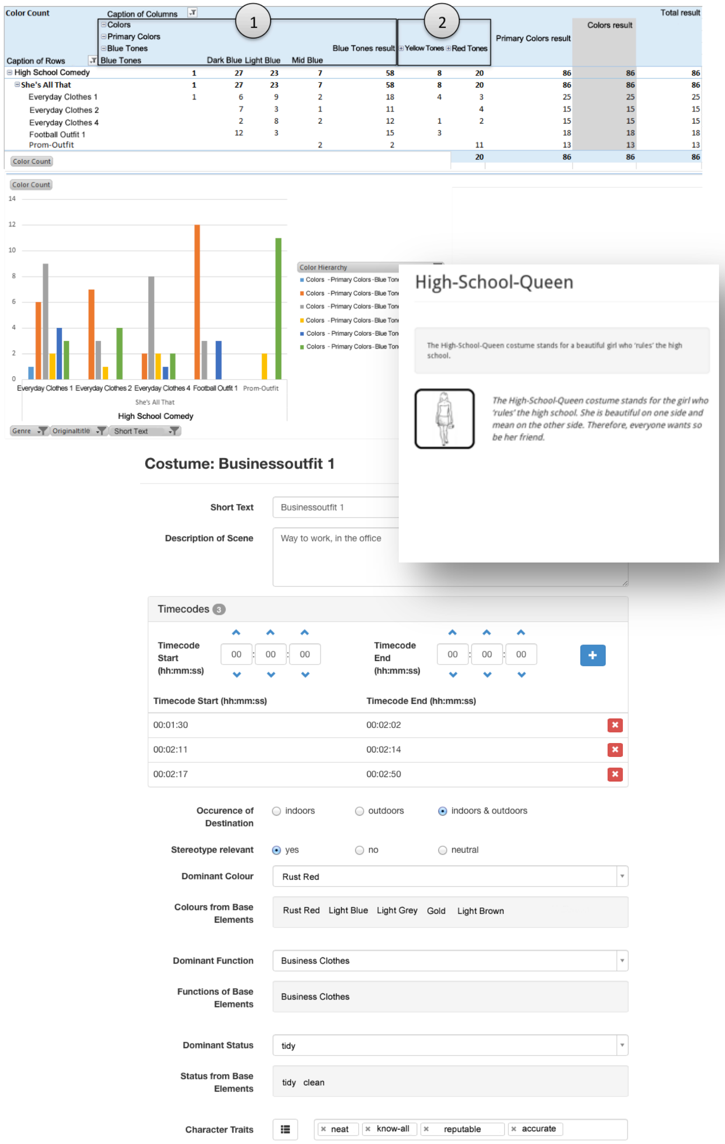
High-School-Queen

The High-School-Queen costume stands for a beautiful girl who 'rules' the high school.

The High-School-Queen costume stands for the girl who 'rules' the high school. She is beautiful on one side and mean on the other side. Therefore, everyone wants so be her friend.

Falkenthal, Michael; Barzen, Johanna; Dörner, Simon; Elkind, Vadym; Fauser, Jan; Leymann, Frank; Strehl, Tino: Datenanalyse in den Digital Humanities – Eine Annäherung an Kostümmuster mittels OLAP Cubes. In: BTW 2015.

Abstraction and Formalization



Color Count Caption of Rows	Caption of Columns ⊟ Colors ⊟ Primary Colors ⊟ Blue Tones Blue Tones	Dark Blue	Light Blue	Mid Blue	Blue Tones result	⊞ Yellow Tones	⊞ Red Tones	Primary Colors result	Colors result	Total result
⊟ High School Comedy	1	27	23	7	58	8	20	86	86	86
⊟ She's All That	1	27	23	7	58	8	20	86	86	86
Everyday Clothes 1	1	6	9	2	18	4	3	25	25	25
Everyday Clothes 2		7	3	1	11		4	15	15	15
Everyday Clothes 4		2	8	2	12	1	2	15	15	15
Football Outfit 1		12	3		15	3		18	18	18
Prom-Outfit				2	2		11	13	13	13
Color Count							20	86	86	86

1
2
Color Count
14
12
10
8
6
4
2
0
Everyday Clothes 1
Everyday Clothes 2
Everyday Clothes 4
Football Outfit 1
Prom-Outfit
She's All That
High School Comedy
Genre
Originaltitle
Short Text
Color Hierarchy
Colors - Primary Colors - Blue Ton
Colors - Primary Colors - Blue Ton
Colors - Primary Colors - Blue Ton
Colors - Primary Colors - Blue Ton
Colors - Primary Colors - Blue Ton
Colors - Primary Colors - Blue Ton

High-School-Queen

The High-School-Queen costume stands for a beautiful girl who 'rules' the high school.

The High-School-Queen costume stands for the girl who 'rules' the high school. She is beautiful on one side and mean on the other side. Therefore, everyone wants so be her friend.

Costume: Businessoutfit 1

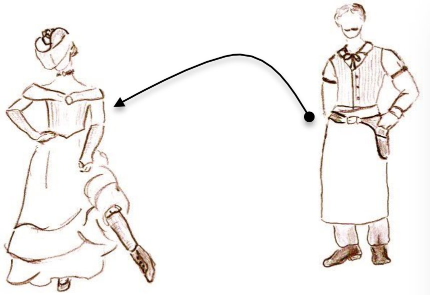
Short Text	Businessoutfit 1
Description of Scene	Way to work, in the office

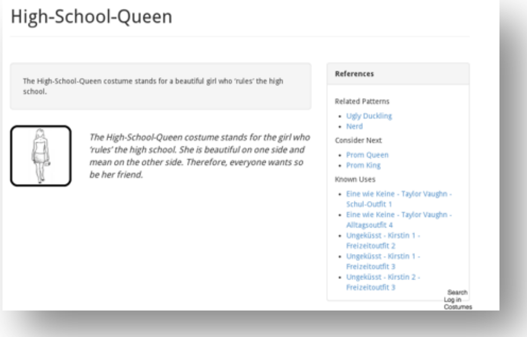
Timecodes 3

Timecode Start (hh:mm:ss) 00 : 00 : 00 Timecode End (hh:mm:ss) 00 : 00 : 00 +

Timecode Start (hh:mm:ss)	Timecode End (hh:mm:ss)	
00:01:30	00:02:02	✖
00:02:11	00:02:14	✖
00:02:17	00:02:50	✖

Occurence of Destination	○ indoors ○ outdoors ● indoors & outdoors
Stereotype relevant	● yes ○ no ○ neutral
Dominant Colour	Rust Red
Colours from Base Elements	Rust Red Light Blue Light Grey Gold Light Brown
Dominant Function	Business Clothes
Functions of Base Elements	Business Clothes
Dominant Status	tidy
Status from Base Elements	tidy clean
Character Traits	× neat × know-all × reputable × accurate





High-School-Queen

The High-School-Queen costume stands for a beautiful girl who 'rules' the high school.

The High-School-Queen costume stands for the girl who 'rules' the high school. She is beautiful on one side and mean on the other side. Therefore, everyone wants so be her friend.

References

Related Patterns

- Ugly Duckling
- Nerd

Consider Next

- Prom Queen
- Prom King

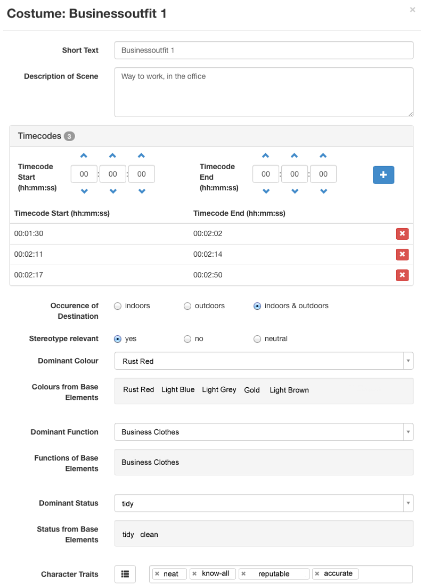
Known Uses

- Eine wie Keine - Taylor Vaughn - Schul-Outfit 1
- Eine wie Keine - Taylor Vaughn - Alltagsoutfit 4
- Ungeküsst - Kirstin 1 - Freizeitoutfit 2
- Ungeküsst - Kirstin 1 - Freizeitoutfit 3
- Ungeküsst - Kirstin 2 - Freizeitoutfit 3

$$\begin{tikzcd}
\mathfrak{D}, \{R_{\mathfrak{B}}\} & L(G^p) \arrow[l] & (\mathfrak{P}_N, \{V_\alpha\}) \\
& L(G^B) \arrow[u] \arrow[ul] \arrow[d, "\mathfrak{e}"] & \\
& \mathfrak{C} & \boxed{\mathfrak{P}_N = C/\approx \quad \{V_\alpha\}} \arrow[l] \arrow[uu]
\end{tikzcd}$$

Formalization

The MUSE Method



Costume: Businessoutfit 1

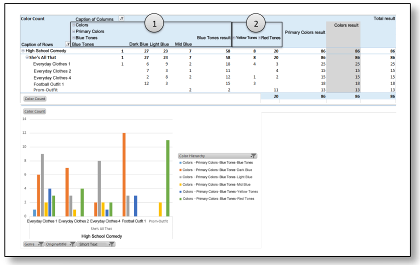
Short Text	Businessoutfit 1
Description of Scene	Way to work, in the office

Timecodes 3

Timecode Start (hh:mm:ss) 00 : 00 : 00 Timecode End (hh:mm:ss) 00 : 00 : 00 +

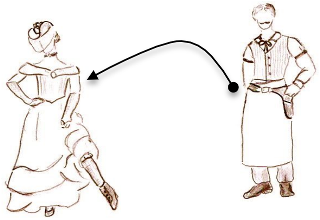
Timecode Start (hh:mm:ss)	Timecode End (hh:mm:ss)
00:01:30	00:02:02
00:02:11	00:02:14
00:02:17	00:02:50

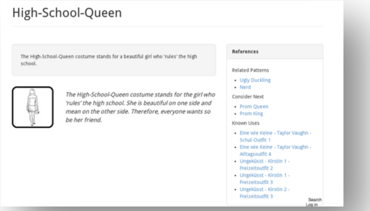
Occurence of Destination	○ indoors ○ outdoors ● indoors & outdoors
Stereotype relevant	● yes ○ no ○ neutral
Dominant Colour	Rust Red
Colours from Base Elements	Rust Red Light Blue Light Grey Gold Light Brown
Dominant Function	Business Clothes
Functions of Base Elements	Business Clothes
Dominant Status	tidy
Status from Base Elements	tidy clean
Character Traits	neat, know-all, reputable, accurate



Color Count Caption of Rows	Caption of Columns Colors Primary Colors Blue Tones Blue Tones	 Dark Blue	 Light Blue	 Mid Blue	 Blue Tones result	 Yellow Tones	 Red Tones	 Primary Colors result	 Colors result	Total result
High School Comedy	1	27	23	7	58	8	20	86	86	86
She's All That	1	27	23	7	58	8	20	86	86	86
Everyday Clothes 1	1	6	9	2	18	4	3	25	25	25
Everyday Clothes 2		7	3	1	11		4	15	15	15
Everyday Clothes 4		2	8	2	12	1	2	15	15	15
Football Outfit 1		12	3		15	3		18	18	18
Prom-Outfit				2	2		11	13	13	13
Color Count							20	86	86	86

1
2
Color Count
14
12
10
8
6
4
2
0
Everyday Clothes 1
Everyday Clothes 2
Everyday Clothes 4
Football Outfit 1
Prom-Outfit
She's All That
High School Comedy
Genre
Originaltitle
Short Text
Color Hierarchy
Colors - Primary Colors - Blue Tones - Blue Tones
Colors - Primary Colors - Blue Tones - Dark Blue
Colors - Primary Colors - Blue Tones - Light Blue
Colors - Primary Colors - Blue Tones - Mid Blue
Colors - Primary Colors - Blue Tones - Yellow Tones
Colors - Primary Colors - Blue Tones - Red Tones





High-School-Queen

The High-School-Queen costume stands for a beautiful girl who 'rules' the high school.

The High-School-Queen costume stands for the girl who 'rules' the high school. She is beautiful on one side and mean on the other side. Therefore, everyone wants so be her friend.

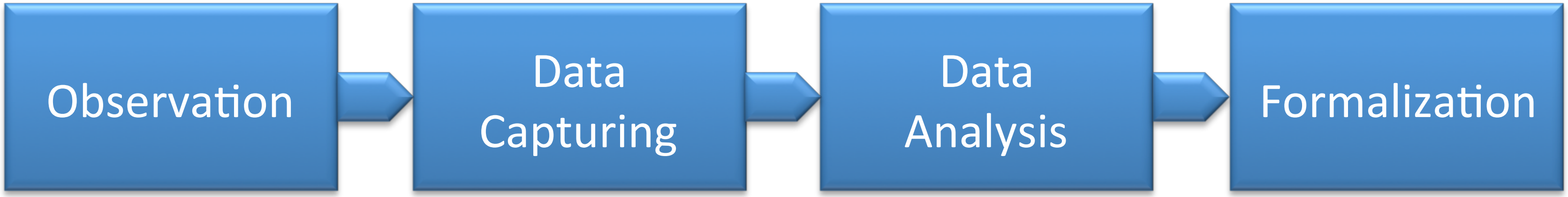
References

Related Patterns
- Ugly Duckling
- Nerd

Consider Next
- Prom Queen
- Prom King

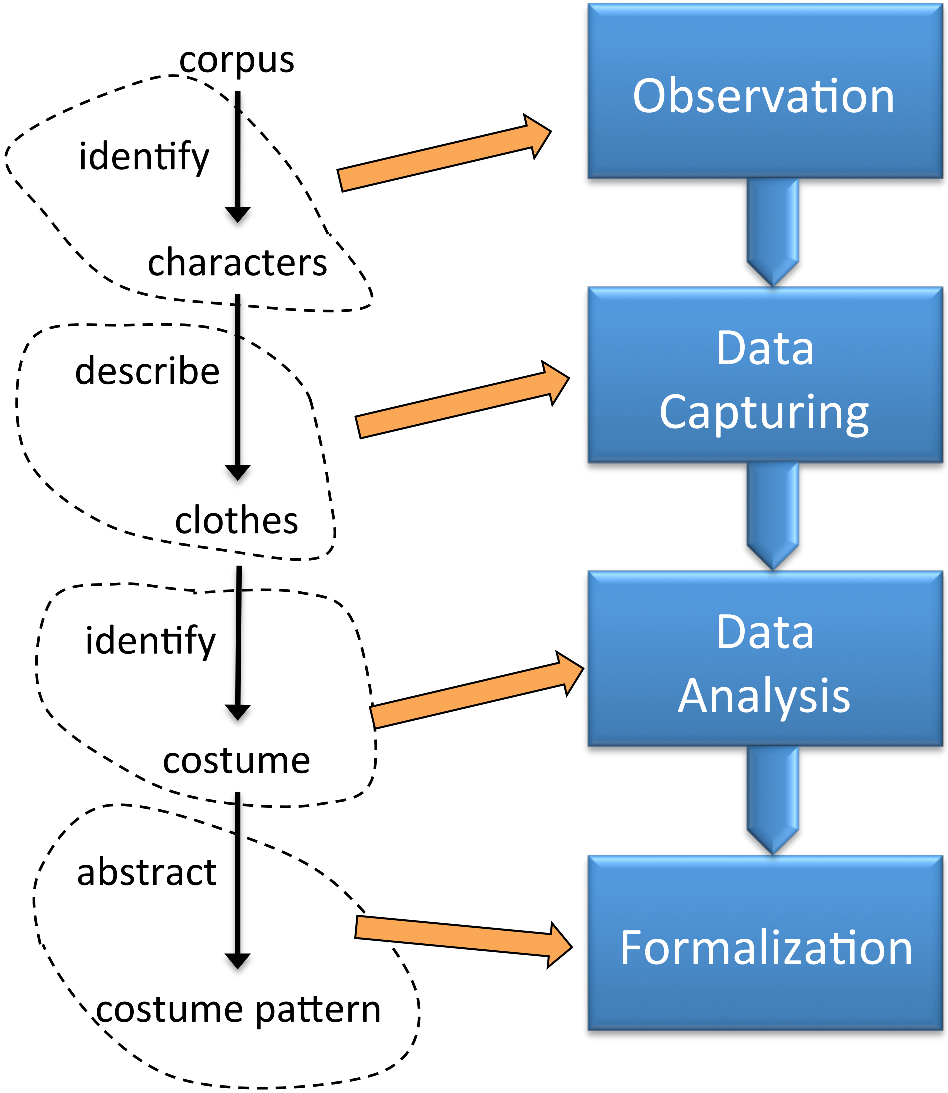
Known Uses
- Eine wie Keine - Taylor Vaughn - Schul-Outfit 1
- Eine wie Keine - Taylor Vaughn - Alltagsoutfit 4
- Ungeküsst - Kirstin 1 - Freizeitoutfit 2
- Ungeküsst - Kirstin 1 - Freizeitoutfit 3
- Ungeküsst - Kirstin 2 - Freizeitoutfit 3

$$\begin{tikzcd}
\mathfrak{D}, \{R_{\mathfrak{B}}\} & L(G^P) \arrow[l] & (\mathfrak{P}_N, \{V_\alpha\}) \\
& L(G^B) \arrow[u] \arrow[ul] \arrow[d, "\mathfrak{e}"] & \\
& \mathfrak{C} & \boxed{\mathfrak{P}_N = C/\approx \quad \{V_\alpha\}} \arrow[l] \arrow[uu]
\end{tikzcd}$$



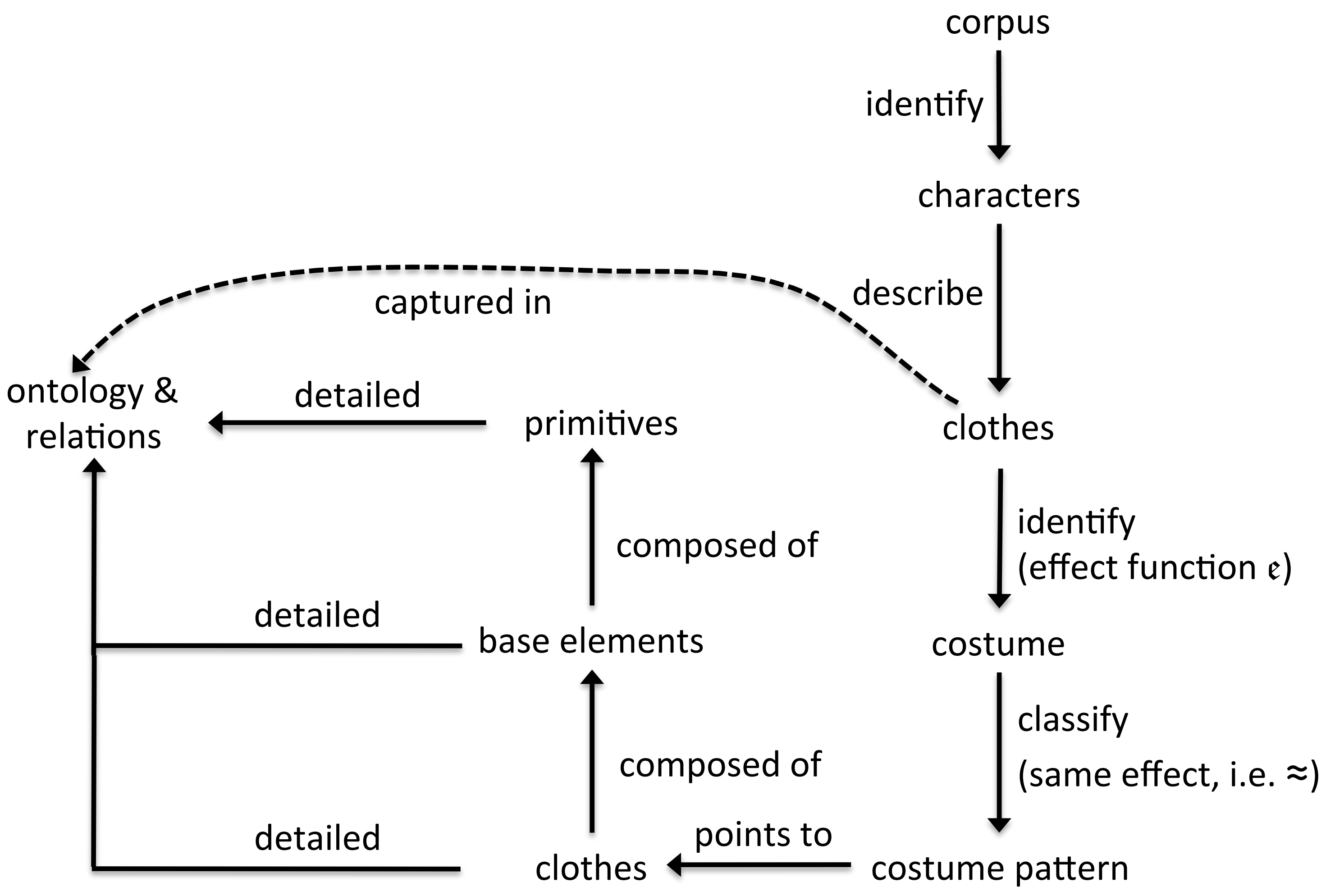
Formalization

Using the Scientific Method in the Humanities



Barzen, Johanna; Leymann, Frank: Patterns as Formulas: Applying the Scientific Method to the Humanities, Technischer Bericht Nr. 2016/01, Universität Stuttgart 2016.

Identification of Patterns: Costumes as Sample



corpus

identify

characters

describe

clothes

identify

(effect function $\mathfrak{e}$)

costume

classify

(same effect, i.e. ≈)

costume pattern

points to

clothes

composed of

base elements

composed of

primitives

detailed

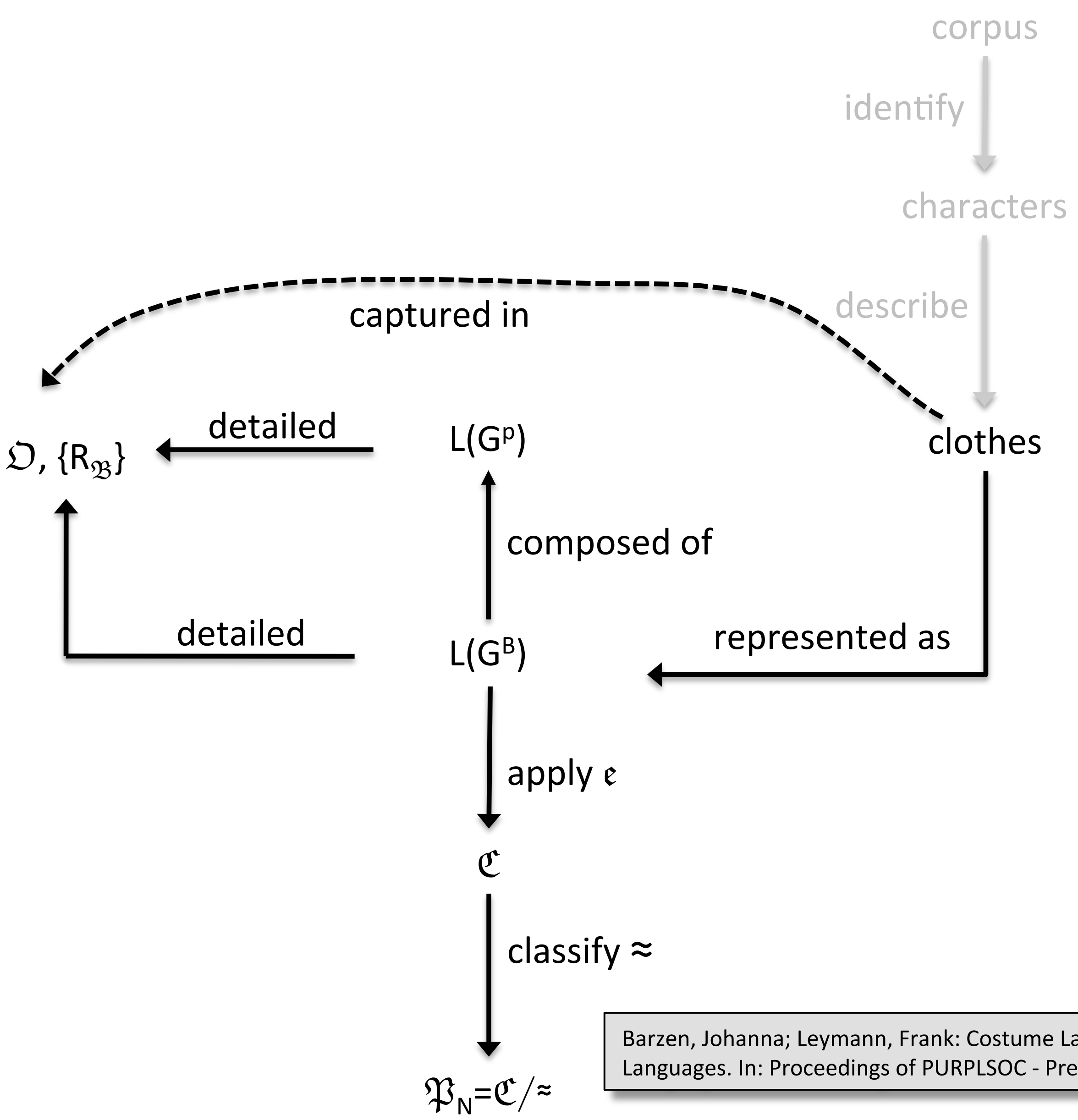
detailed

detailed

ontology & relations

captured in

Identification of Patterns: Formal Aspects



corpus

identify

characters

describe

clothes

captured in

$\mathfrak{O}$, $\{R_{\mathfrak{B}}\}$

detailed

$L(G^p)$

composed of

detailed

$L(G^B)$

represented as

apply $\mathfrak{e}$

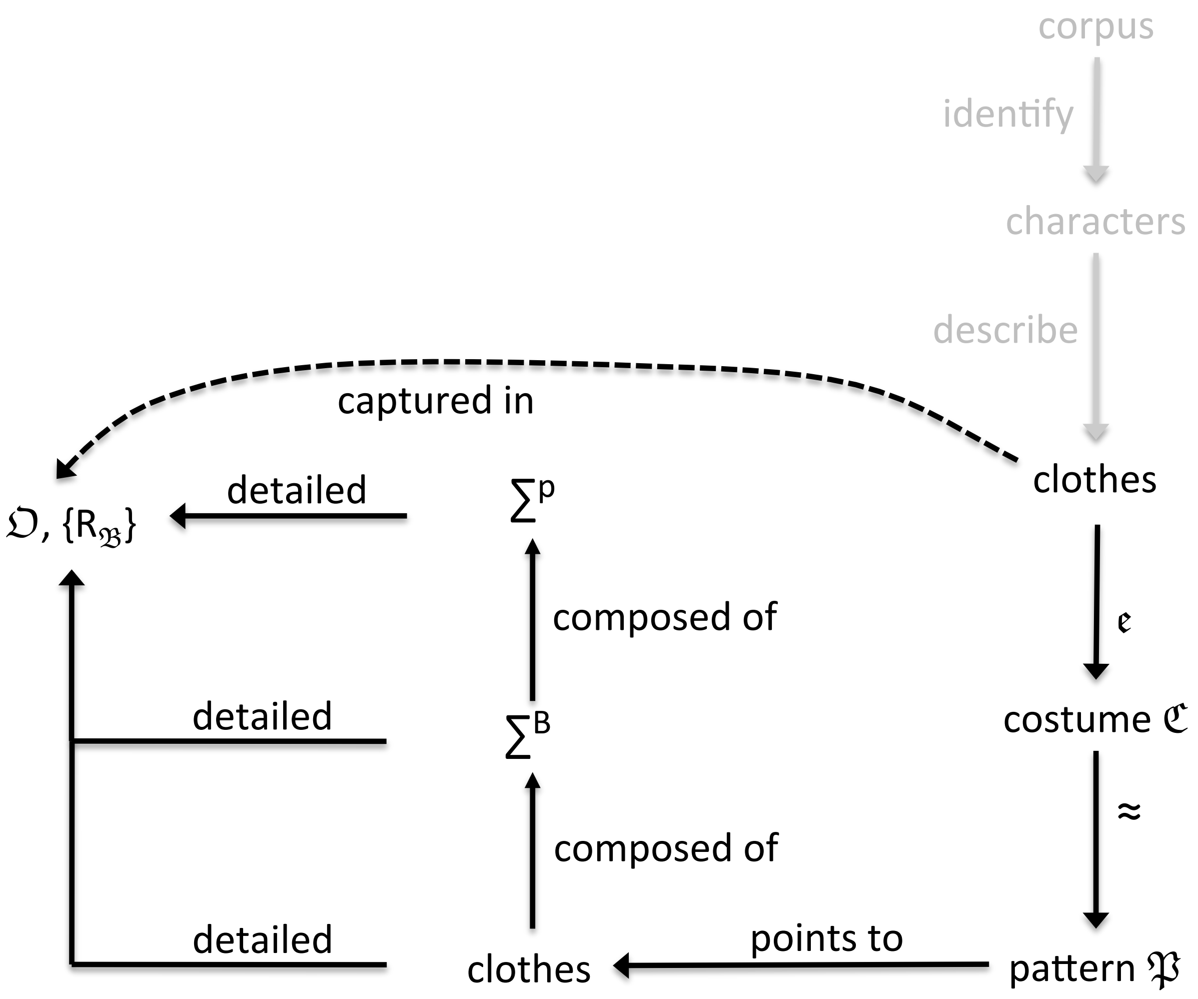
$\mathfrak{C}$

classify ≈

$\mathfrak{P}_N=\mathfrak{C}/\approx$

Barzen, Johanna; Leymann, Frank: Costume Languages As Pattern Languages. In: Proceedings of PURPLSOC - Preparatory Workshop 2014.

Identification of Patterns: Symbols



corpus

identify

characters

describe

clothes

$\mathfrak{e}$

costume $\mathfrak{C}$

$\approx$

pattern $\mathfrak{P}$

points to

clothes

composed of

Σ^B

composed of

Σ^P

detailed

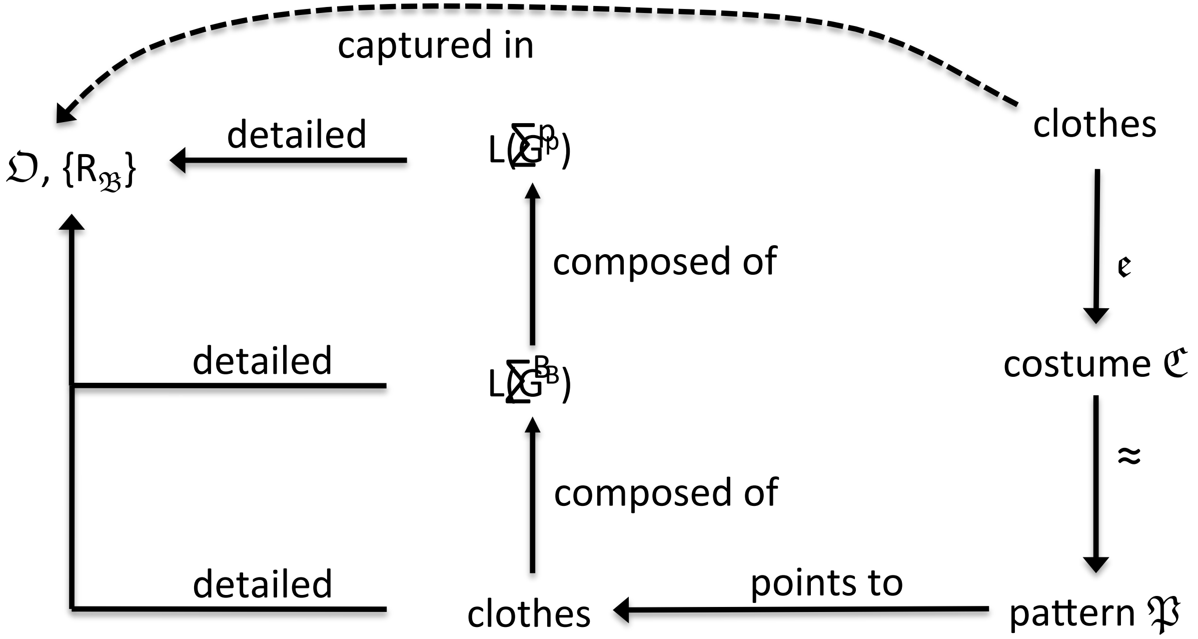
detailed

detailed

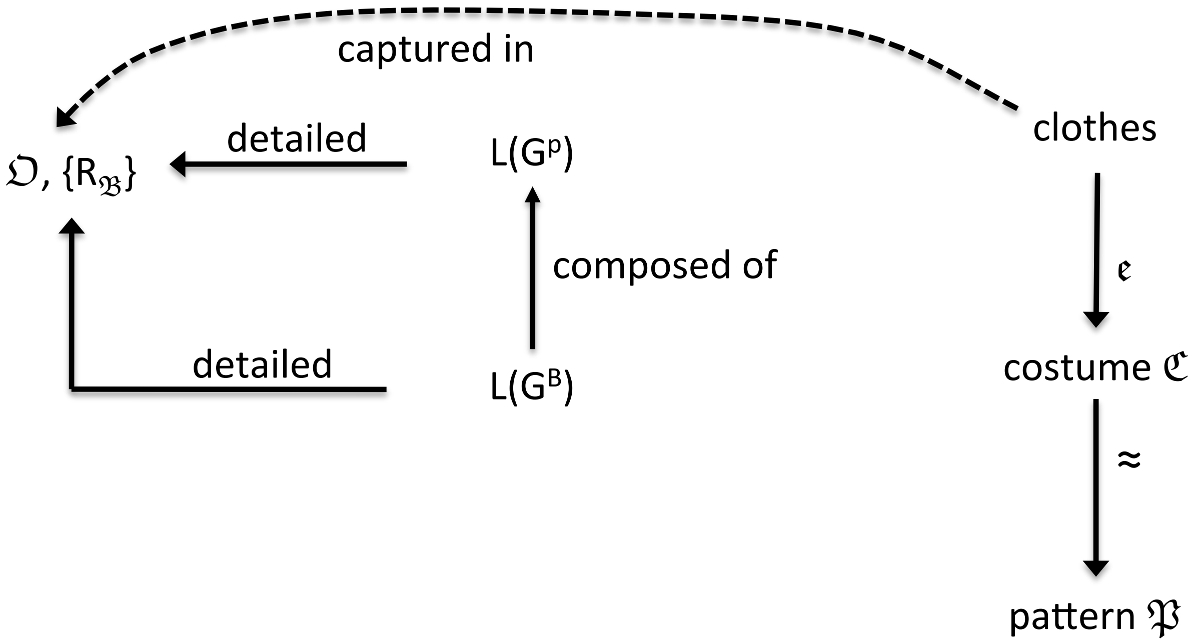
$\mathfrak{O}$, $\{R_{\mathfrak{B}}\}$

captured in

Identification of Patterns: Formal Languages



Identification of Patterns: Formal Languages



captured in

$\mathfrak{O}, \{R_{\mathfrak{B}}\}$

detailed

$L(G^p)$

clothes

composed of

$\mathfrak{e}$

detailed

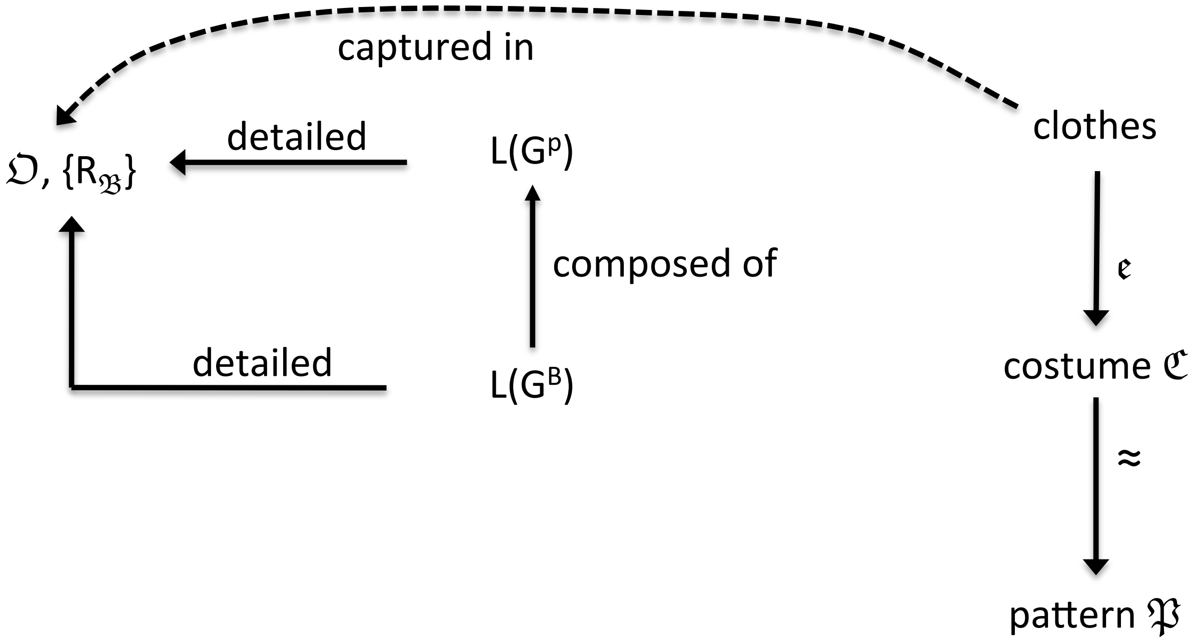
$L(G^B)$

costume $\mathfrak{C}$

$\approx$

pattern $\mathfrak{P}$

Identification of Patterns: Formal Aspects



captured in

clothes

$\mathfrak{O}$, $\{R_{\mathfrak{B}}\}$

detailed

$L(G^p)$

composed of

$\mathfrak{e}$

detailed

$L(G^B)$

costume $\mathfrak{C}$

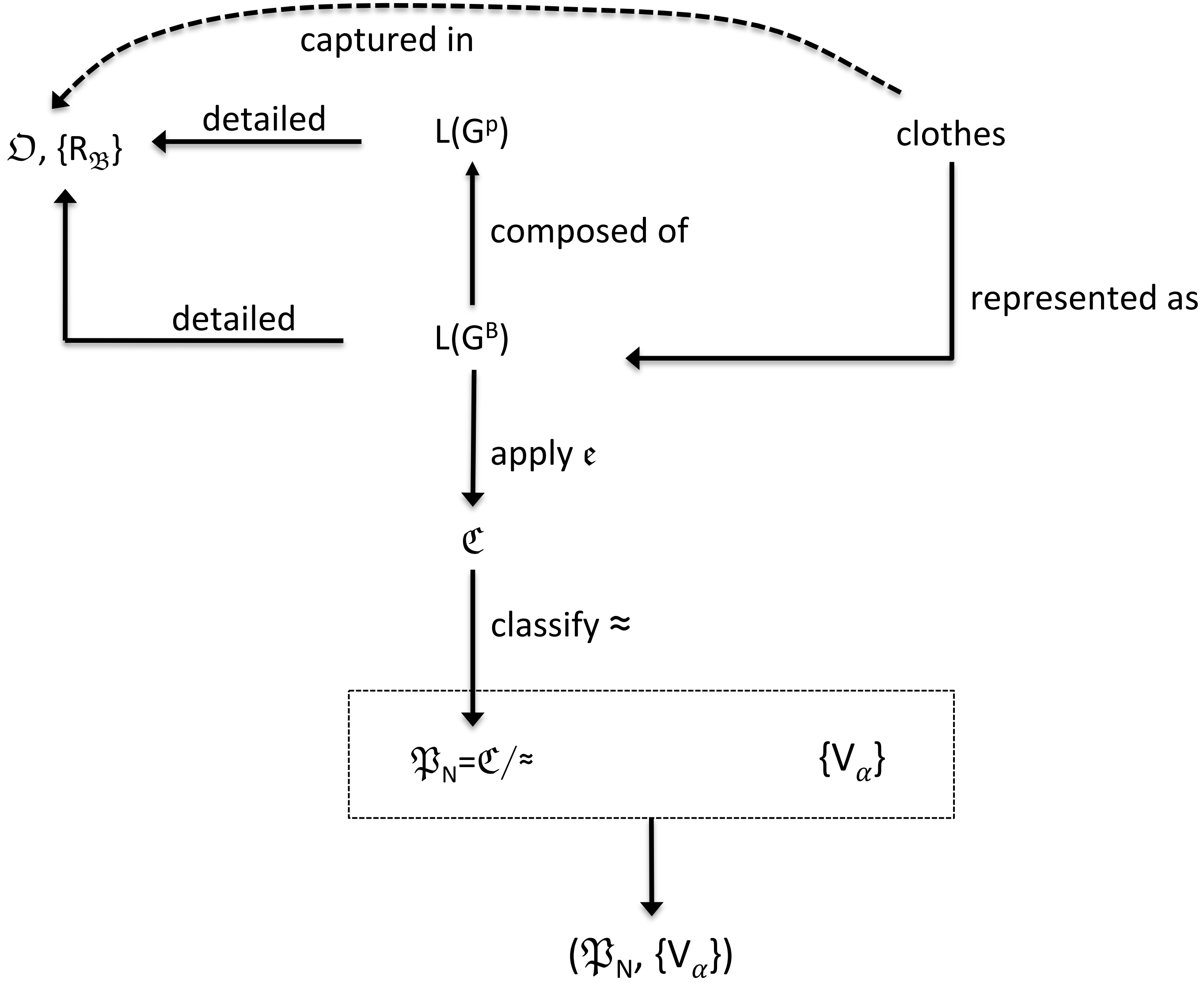
$\approx$

pattern $\mathfrak{P}$

Identification of Patterns: Formal Aspects

$$\begin{tikzcd}
\mathfrak{O}, \{R_{\mathfrak{B}}\} & L(G^p) \arrow[l, "\text{detailed}"'] & \text{clothes} \arrow[ll, dashed, bend right, "\text{captured in}"'] \arrow[dl, "\text{represented as}"] \\
 & L(G^B) \arrow[u, "\text{composed of}"'] \arrow[ul, "\text{detailed}"] & \\
 & \mathfrak{C} & \\
 & \mathfrak{P}_N = \mathfrak{C}/\approx &
\arrow[from=2-2, to=3-2, "\text{apply } \mathfrak{e}"]
\arrow[from=3-2, to=4-2, "\text{classify } \approx"]
\end{tikzcd}$$

Identification of Pattern Languages: Summary



captured in

$\mathfrak{O}$, $\{R_{\mathfrak{B}}\}$

detailed

$L(G^p)$

clothes

composed of

represented as

detailed

$L(G^B)$

apply $\mathfrak{e}$

$\mathfrak{C}$

classify ≈

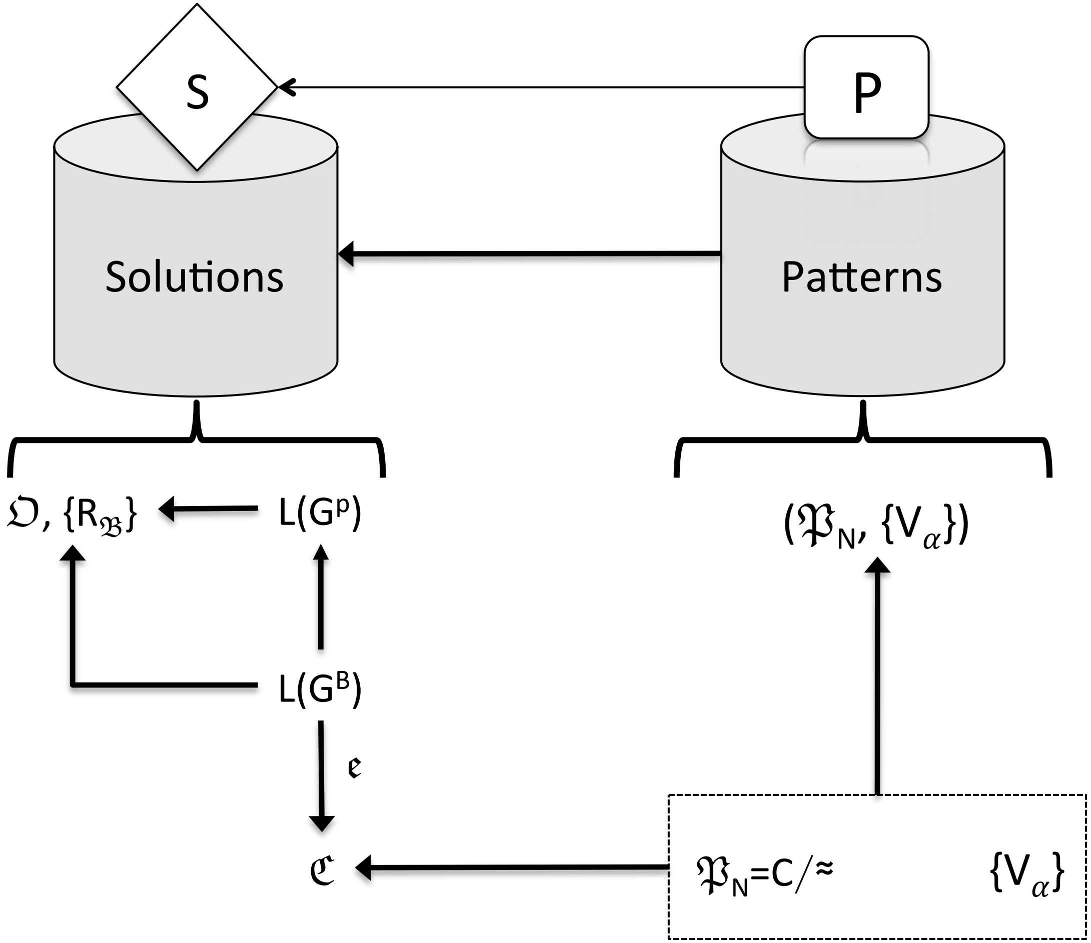
$\mathfrak{P}_N = \mathfrak{C}/\approx$

$\{V_\alpha\}$

$(\mathfrak{P}_N, \{V_\alpha\})$

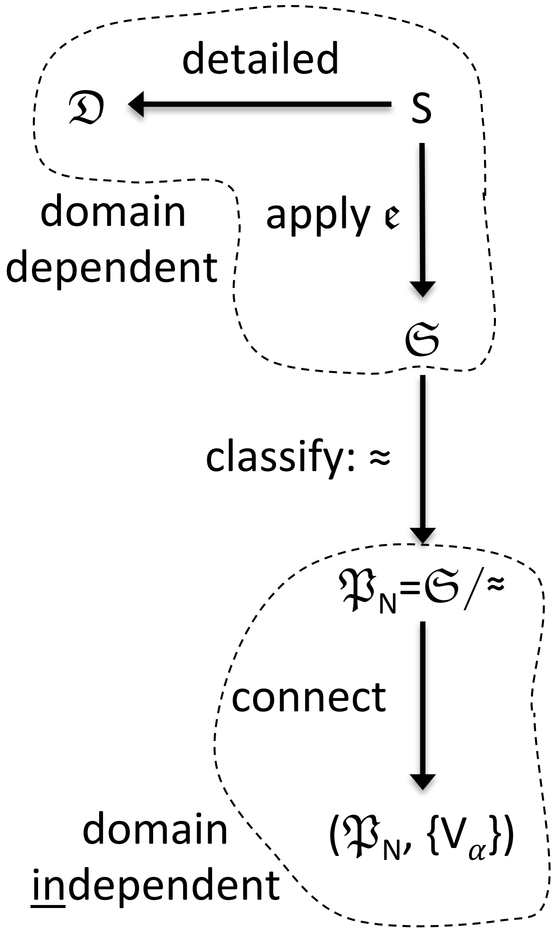
Barzen, Johanna; Leymann, Frank: Costume Languages As Pattern Languages. In: Proceedings of PURPLSOC - Preparatory Workshop 2014.

Repositories


S
P
Solutions
Patterns
$\mathfrak{D}, \{R_{\mathfrak{B}}\}$
$L(G^p)$
$(\mathfrak{P}_N, \{V_\alpha\})$
$L(G^B)$
$\mathfrak{e}$
$\mathfrak{C}$
$\mathfrak{P}_N=C/\approx$
$\{V_\alpha\}$

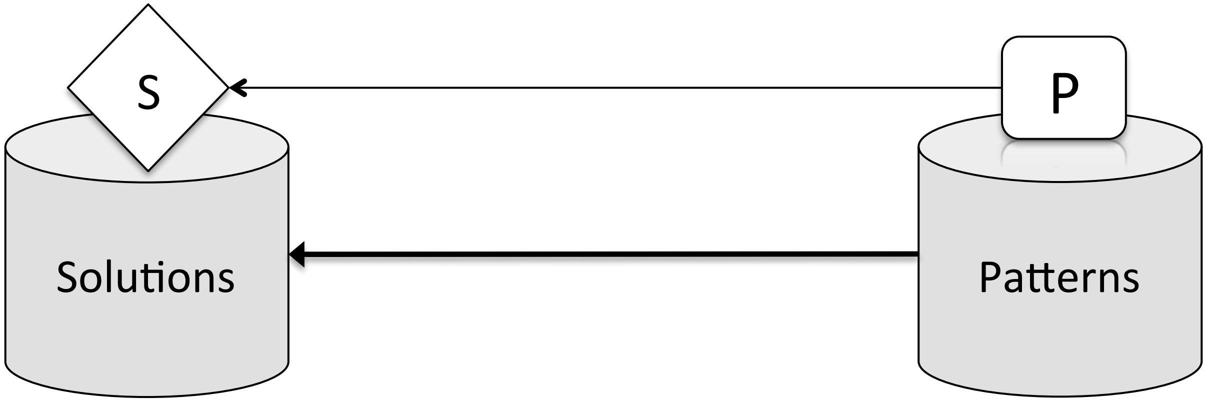
Fehling, Christoph; Barzen, Johanna; Falkenthal, Michael; Leymann, Frank: PatternPedia - Collaborative Pattern Identification and Authoring. In: Proceedings of PURPLSOC - Preparatory Workshop 2014.

Generalization of Formalization



$\mathfrak{D}$: Domain ontology
S: Solution language
$\mathfrak{S}$: Solutions
$\mathfrak{P}$: Patterns (proven solutions)

Generalization: Repositories



$$\overbrace{\mathfrak{D} \xleftarrow{\text{detailed}} S \xrightarrow{\text{apply } \mathfrak{e}} \mathfrak{S}} \xrightarrow{\text{classify: } \approx} \overbrace{\mathfrak{P}_N = \mathfrak{S}/\approx \xrightarrow{\text{connect}} (\mathfrak{P}_N, \{V_\alpha\})}$$

Agenda

Capturing Knowledge

Discovering Patterns

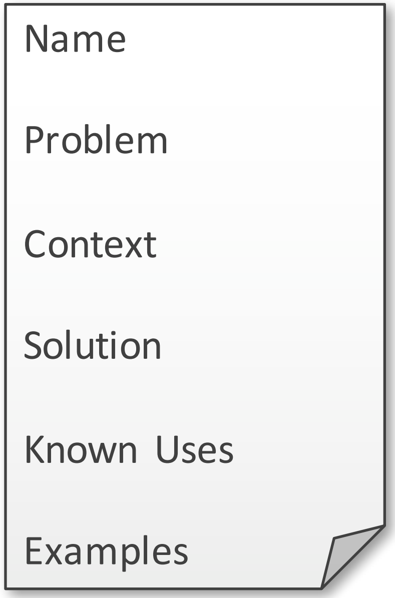
Patterns in the Humanities

From Patterns to Solutions

Summary

Let’s recap: State of the Art in Pattern Usage

- Patterns are used to present **proven solution knowledge to recurring problems in a human readable way**



Name

Problem

Context

Solution

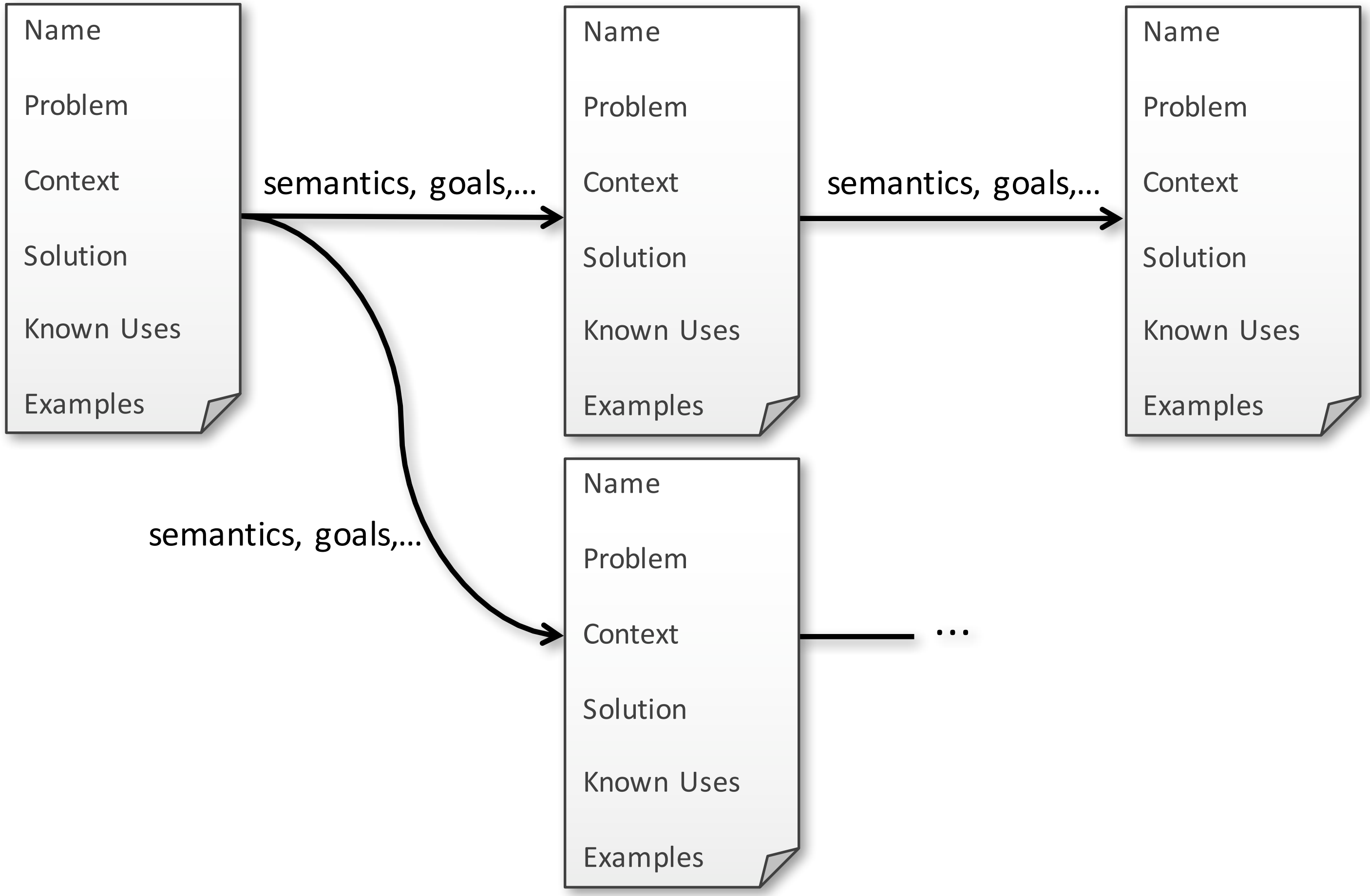
Known Uses

Examples

- Patterns aim at **generalization and abstraction** of solution knowledge

Let’s recap: State of the Art in Pattern Usage

- Patterns are organized into pattern languages to **solve problems in combination**



Name
Problem
Context
Solution
Known Uses
Examples

semantics, goals,…

Name
Problem
Context
Solution
Known Uses
Examples

semantics, goals,…

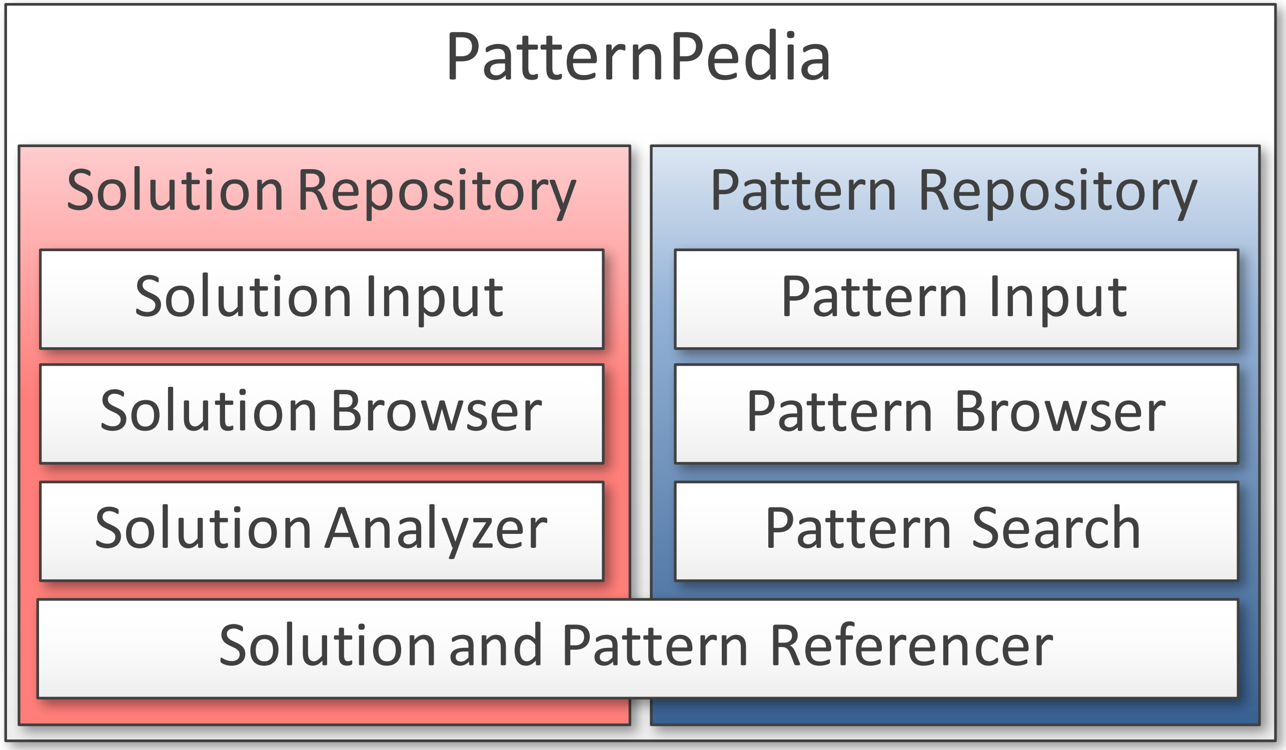
Name
Problem
Context
Solution
Known Uses
Examples

semantics, goals,…

Name
Problem
Context
Solution
Known Uses
Examples

…

PatternPedia
A Tool Chain to Manage Patterns and Solutions


PatternPedia

Solution Repository

Solution Input

Solution Browser

Solution Analyzer

Pattern Repository

Pattern Input

Pattern Browser

Pattern Search

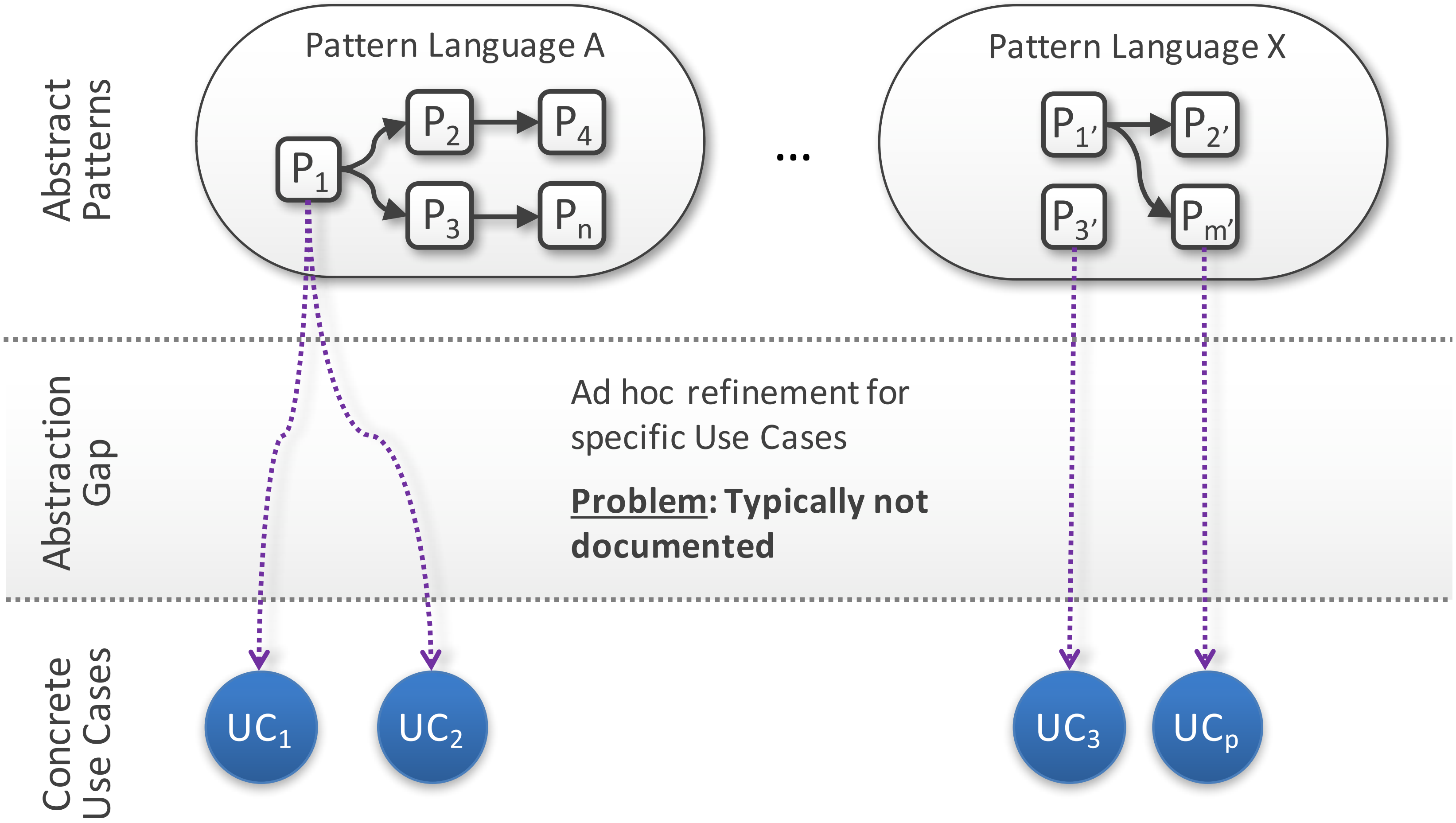
Solution and Pattern Referencer

Fehling et al.: PatternPedia – Collaborative Pattern Identification and Authoring.
In PURPLSOC Preparatory Workshop 2014.

Motivation

Imagine, you are faced with

- a **Problem**,
- **several Pattern Languages** that provide solution knowledge
- and you have **no clue on how to implement** the patterns for your use case



Ad hoc refinement for specific Use Cases

<u>Problem</u>: Typically not documented

Shortcomings

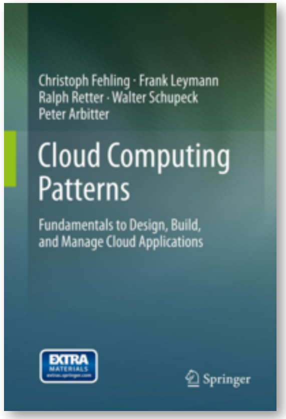
- While patterns aim to provide **abstract** and general solution knowledge it is **time consuming** to apply them for concrete and individual use cases
- The Gap of Abstraction between patterns and concrete implementations leads to **huge efforts** because of **ad hoc refinements and implementations**
- Concrete solution artifacts are **lost during** the pattern authoring process and **cannot be reused**

How to systematically support, guide and ease applying abstract patterns to concrete use cases?

Findings from the Domain of Cloud Computing

Fehling et al.
www.cloudcomputingpatterns.org

+
Level of Abstraction
-

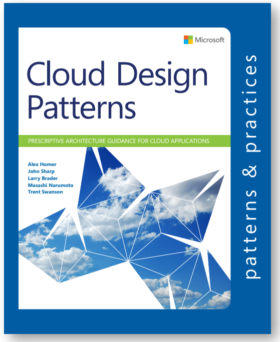

Christoph Fehling · Frank Leymann
Ralph Retter · Walter Schupeck
Peter Arbitter

Cloud Computing Patterns

Fundamentals to Design, Build, and Manage Cloud Applications

EXTRA MATERIALS

Springer


Microsoft

Cloud Design Patterns

PRESCRIPTIVE ARCHITECTURE GUIDANCE FOR CLOUD APPLICATIONS

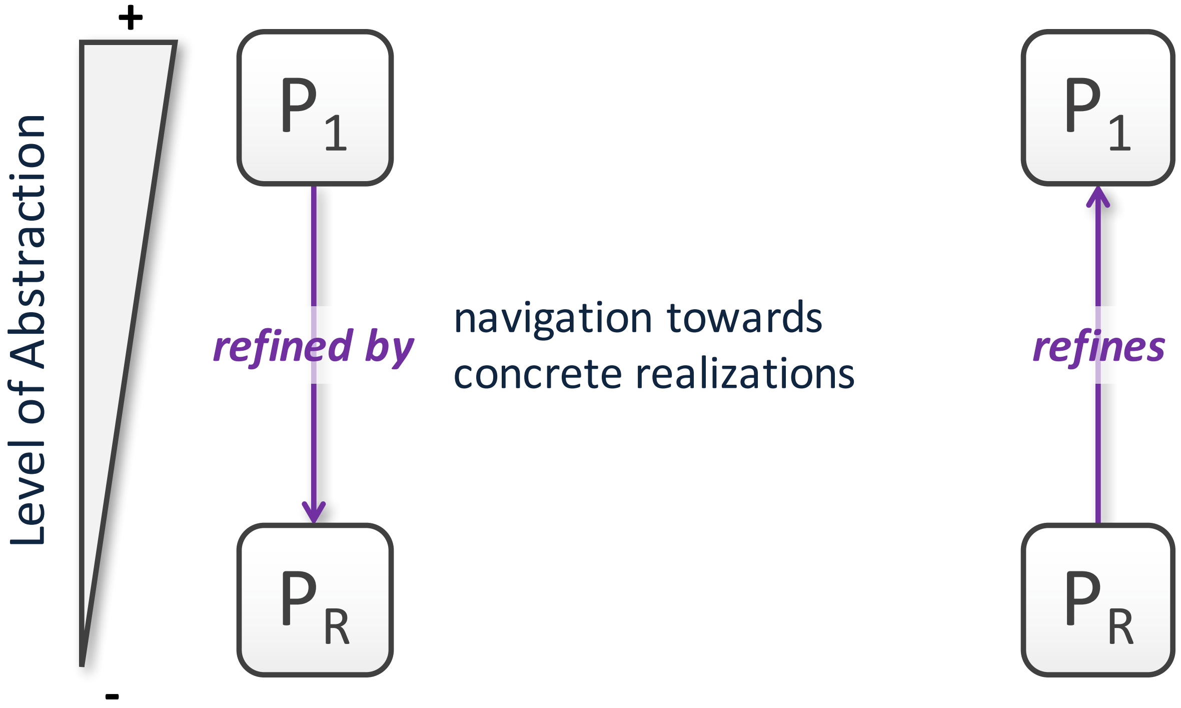
Alex Homer
John Sharp
Larry Brader
Masashi Narumoto
Trent Swanson

patterns & practices

Ninja of Three (Amazon)
http://en.clouddesignpattern.org/index.php/Main_Page

Homer et al. (Microsoft)
https://msdn.microsoft.com/en-us/library/dn600223.aspx

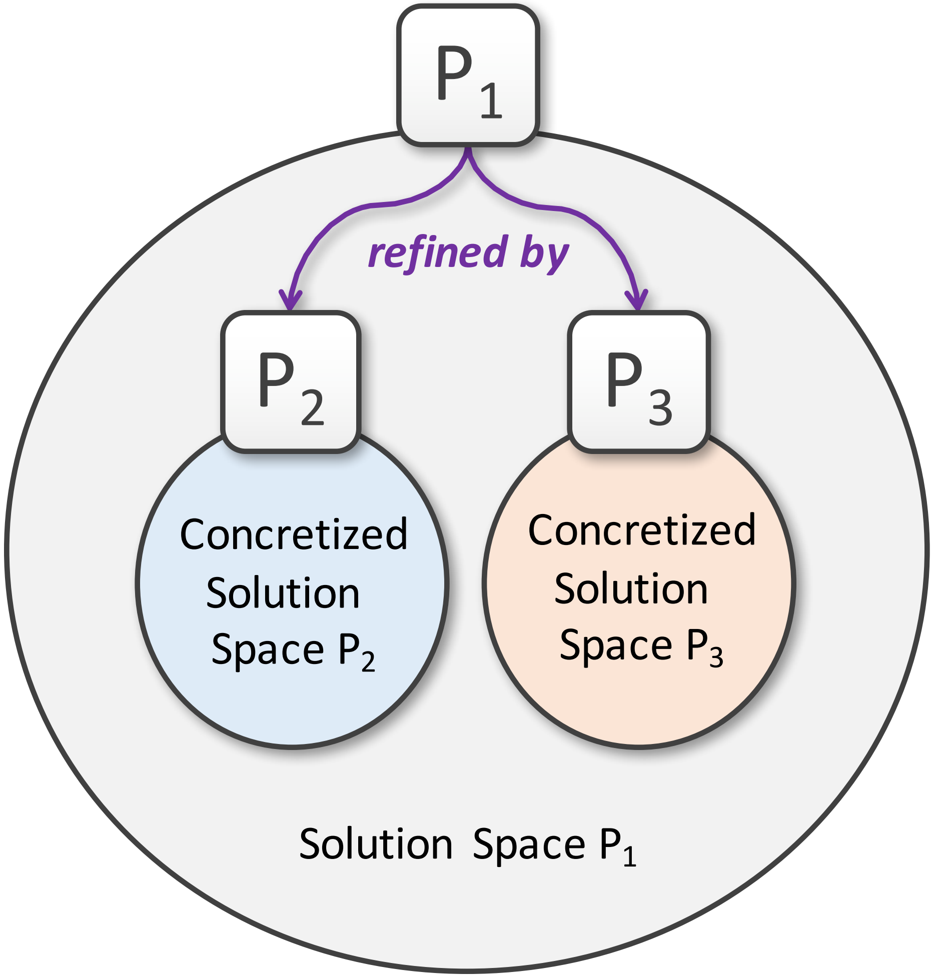
Connecting Patterns via Refinement Links



navigation towards concrete realizations

navigation towards general problem and solution concepts

Pattern Refinement and Solution Spaces



P_1
refined by
P_2
P_3
Concretized Solution Space P_2
Concretized Solution Space P_3
Solution Space P_1

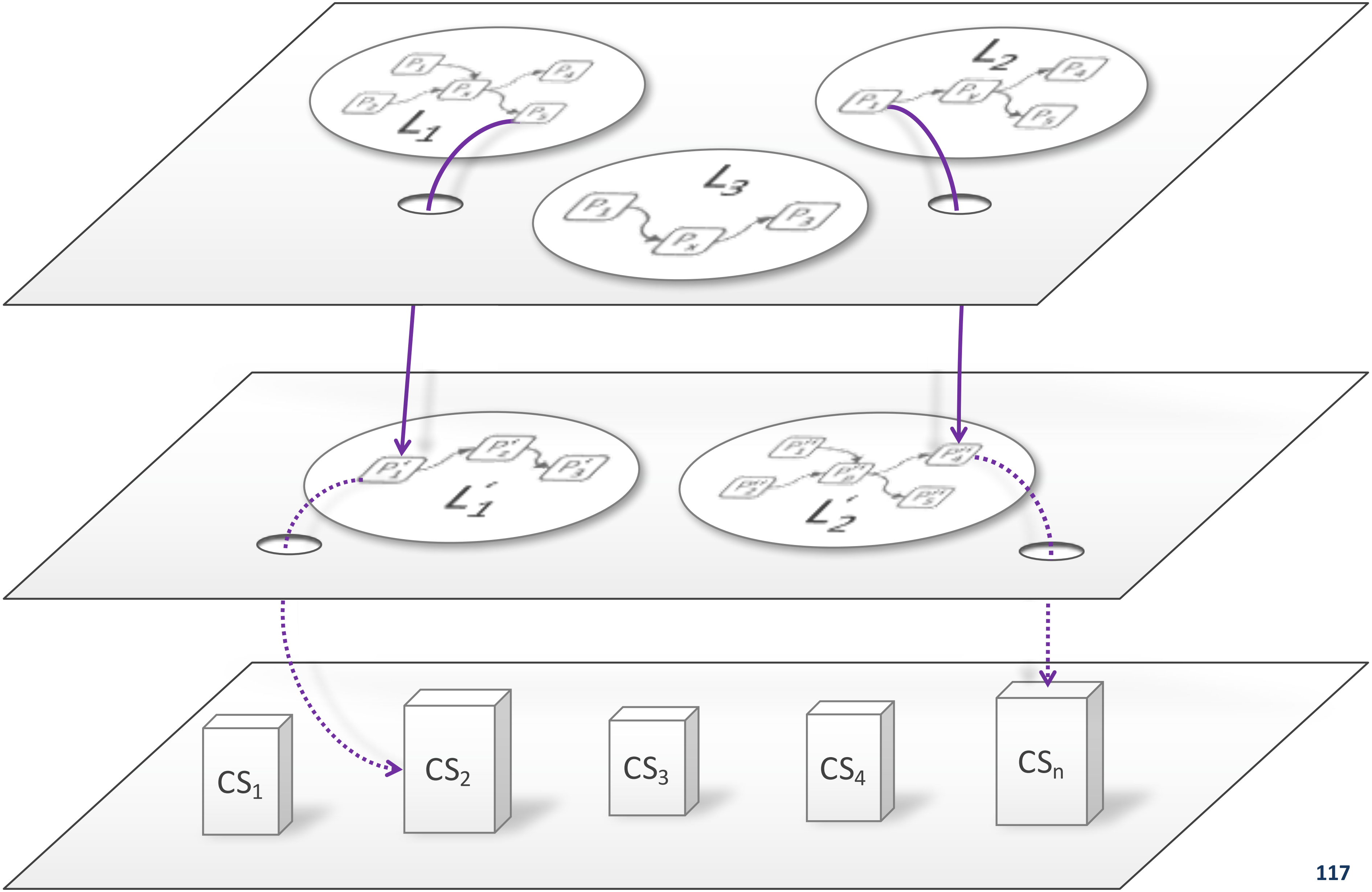
More general applicable

- Vendor agnostic
- Technology agnostic
- Genre agnostic
- …

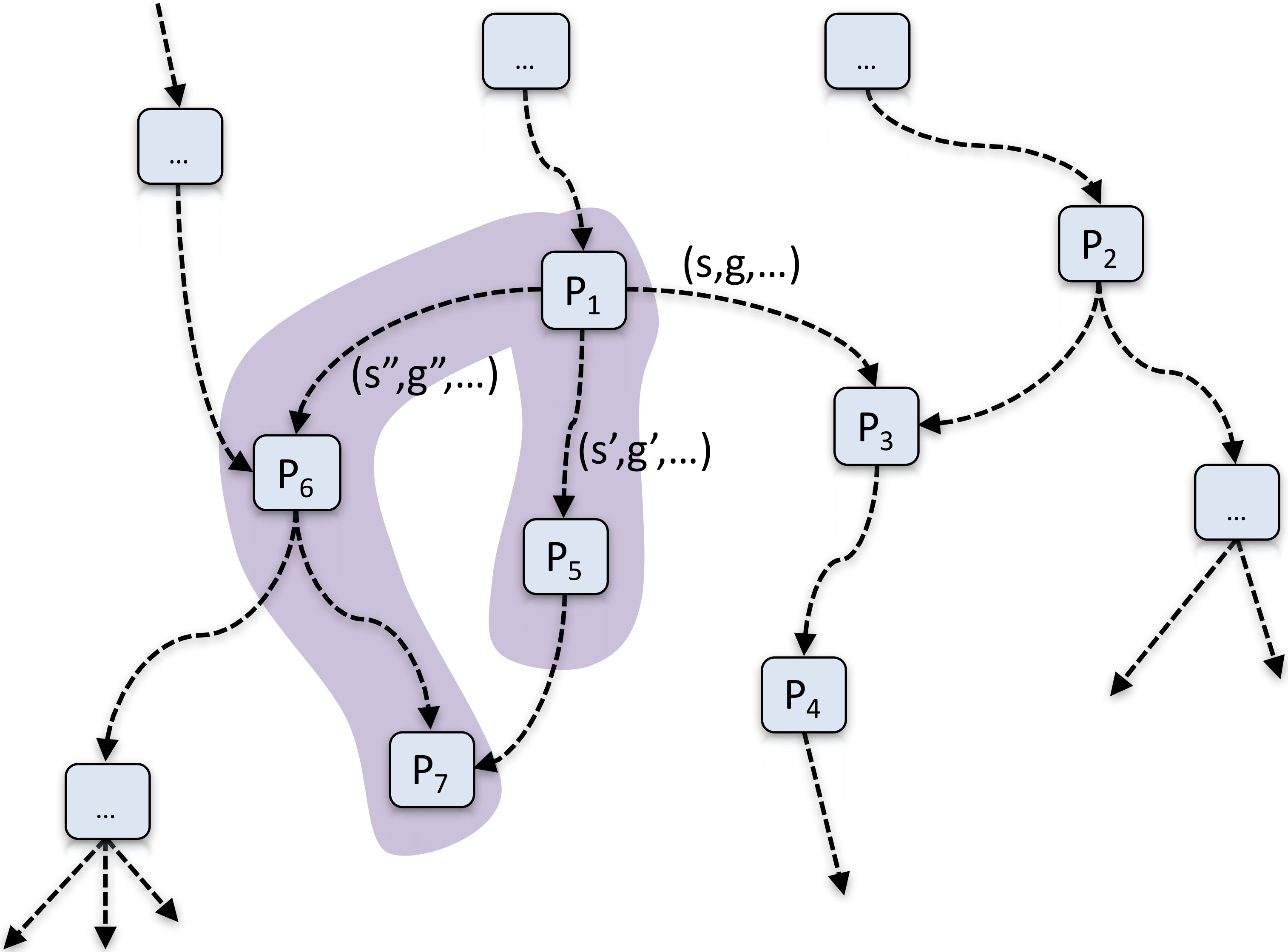
More specific and more guidance

- Vendor specific
- Technology specific
- Genre specific
- …

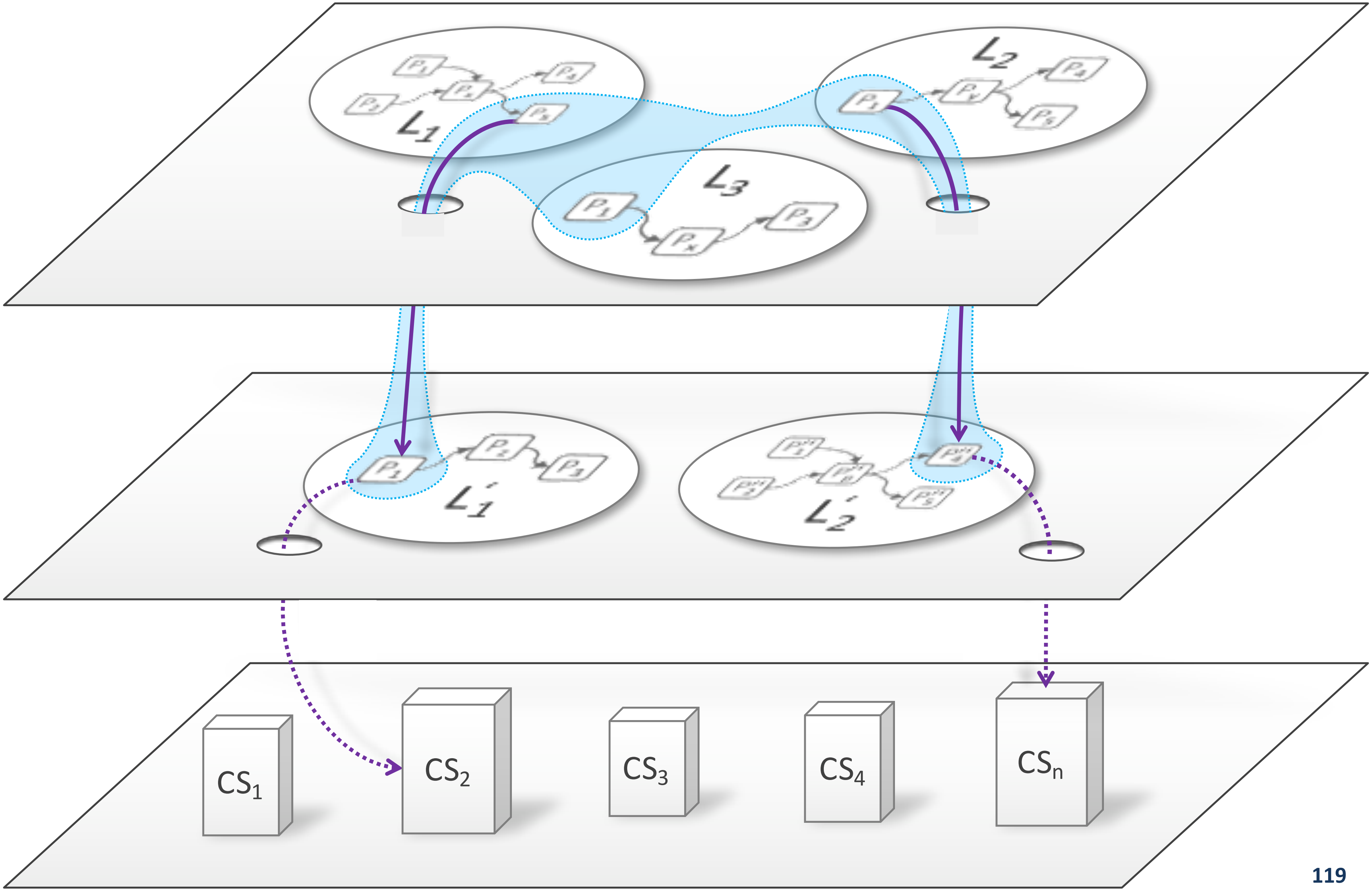
Refinement Links Applied



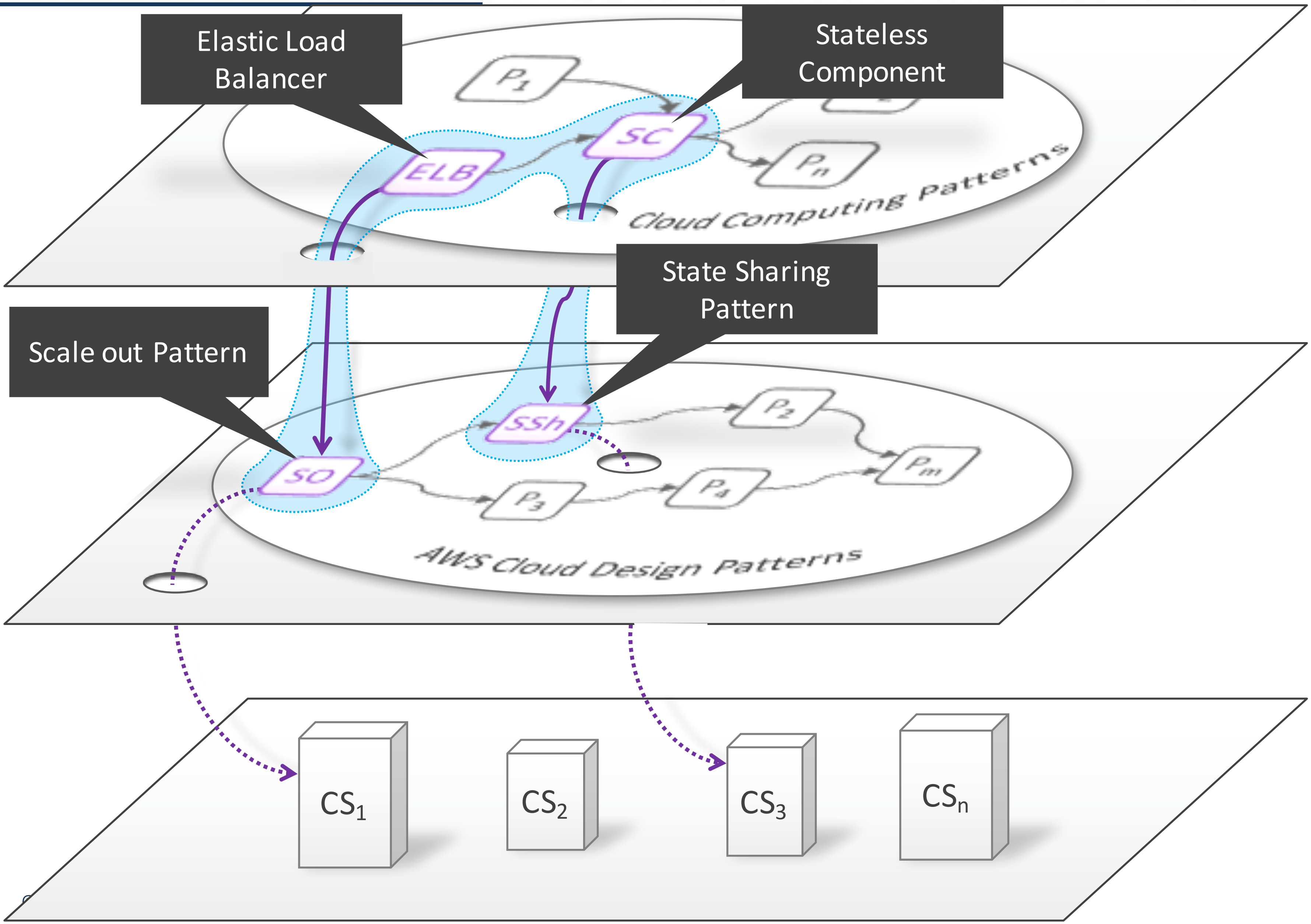
Remember: Solution Paths



Solution Path spanning multiple Pattern Languages on different levels of Abstraction



Case Study: Vendor-agnostic and Vendor-specific Cloud Patterns



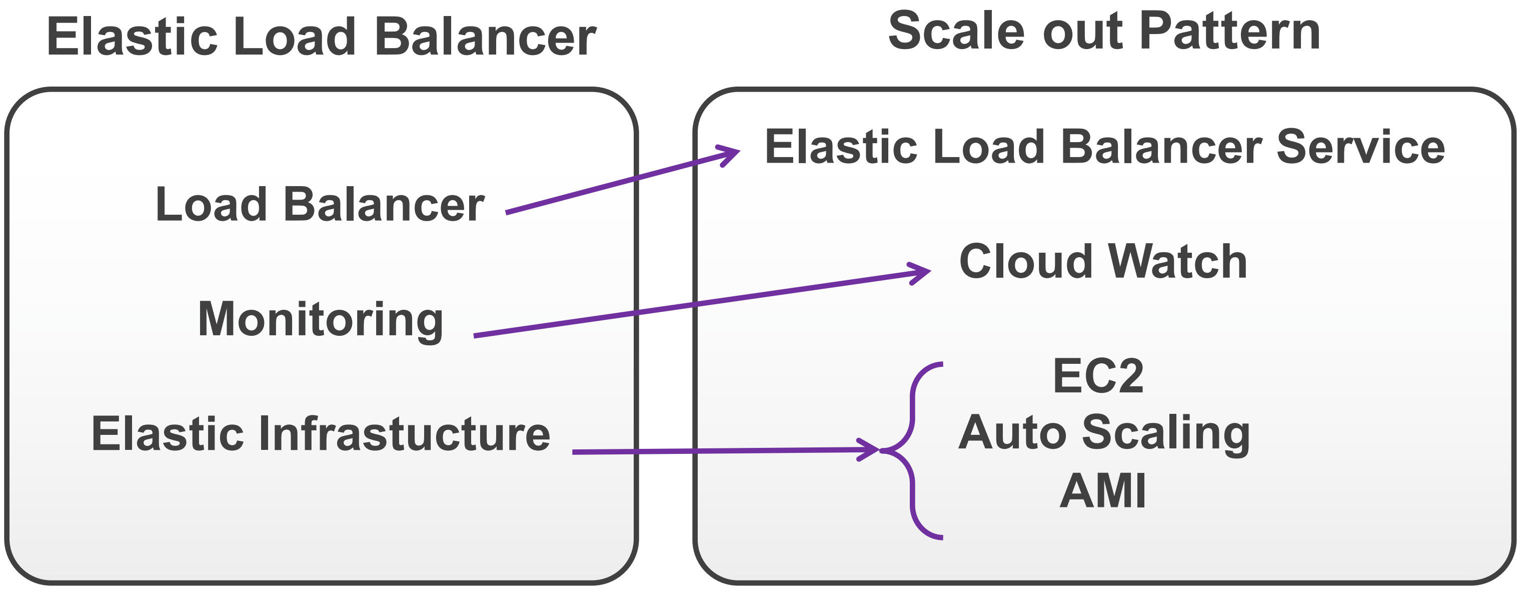
Elastic Load Balancer

Stateless Component

State Sharing Pattern

Scale out Pattern

Case Study: Mapping of Concepts to Technologies



Elastic Load Balancer

Load Balancer

Monitoring

Elastic Infrastucture

Scale out Pattern

Elastic Load Balancer Service

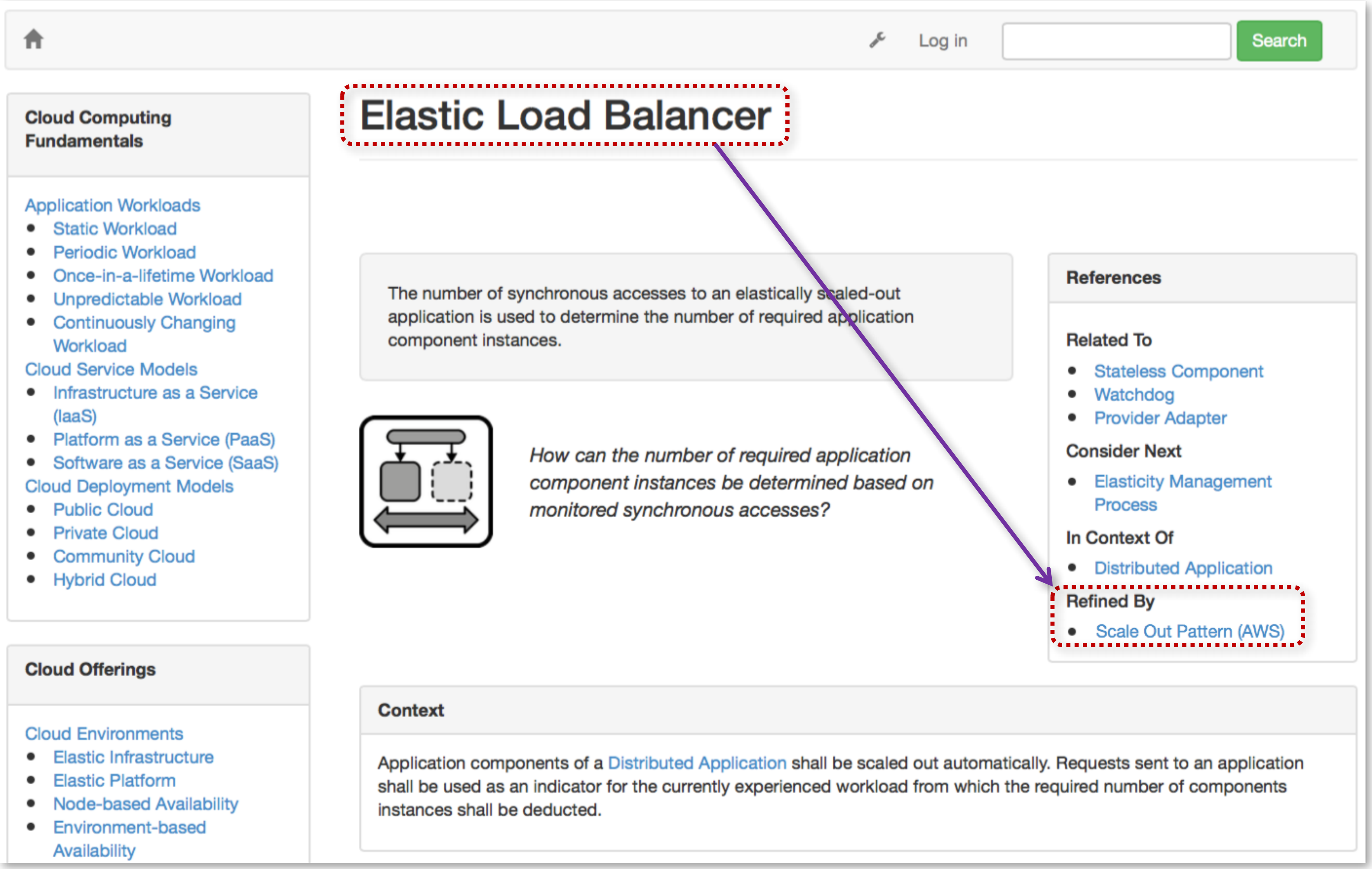
Cloud Watch

EC2
Auto Scaling
AMI

Concepts

Vendor specific Services

Implementation in PatternPedia



Cloud Computing Fundamentals

Application Workloads
- Static Workload
- Periodic Workload
- Once-in-a-lifetime Workload
- Unpredictable Workload
- Continuously Changing Workload

Cloud Service Models
- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

Cloud Deployment Models
- Public Cloud
- Private Cloud
- Community Cloud
- Hybrid Cloud

Cloud Offerings

Cloud Environments
- Elastic Infrastructure
- Elastic Platform
- Node-based Availability
- Environment-based Availability

Elastic Load Balancer

The number of synchronous accesses to an elastically scaled-out application is used to determine the number of required application component instances.

How can the number of required application component instances be determined based on monitored synchronous accesses?

References

Related To
- Stateless Component
- Watchdog
- Provider Adapter

Consider Next
- Elasticity Management Process

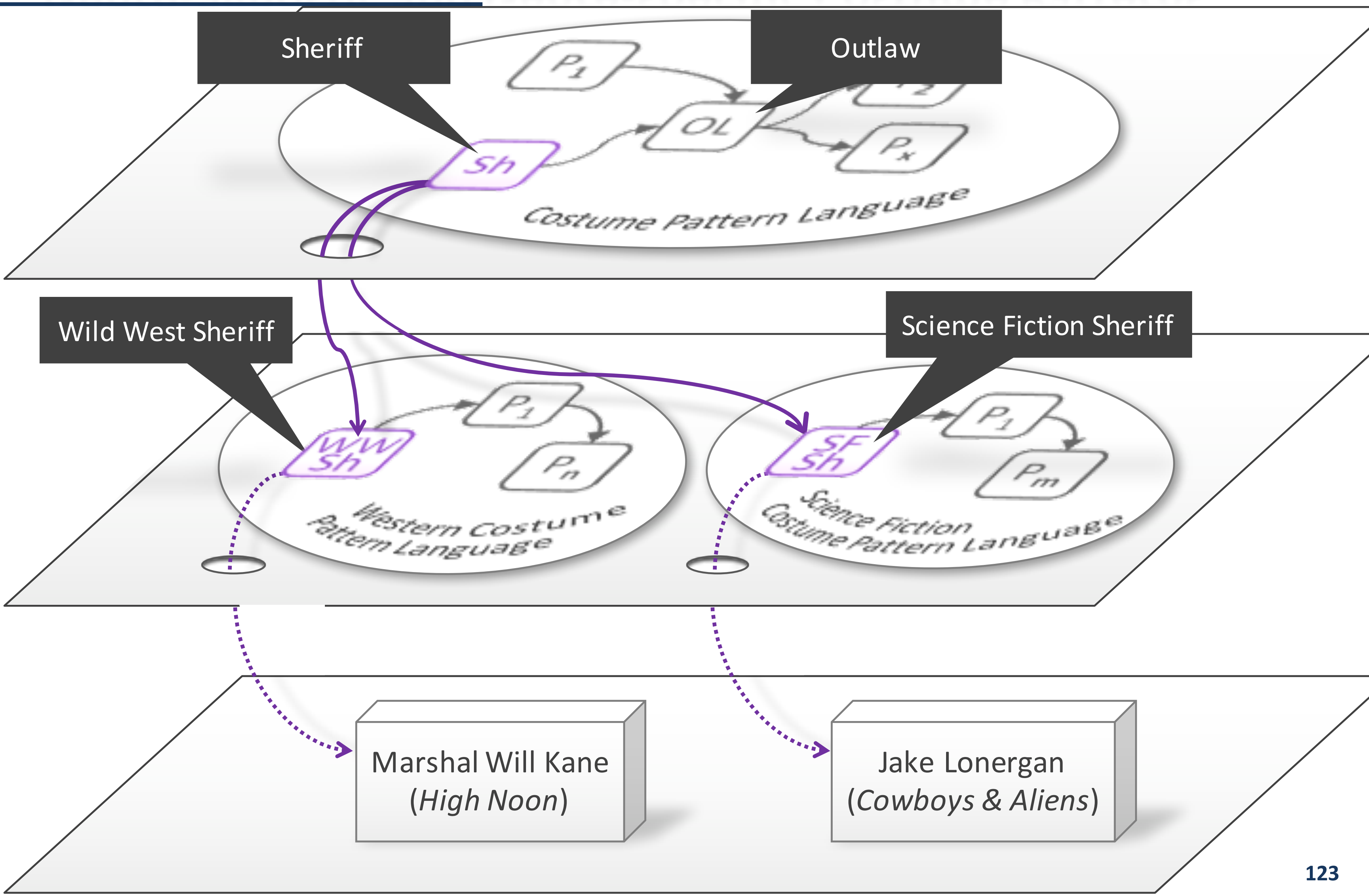
In Context Of
- Distributed Application

Refined By
- Scale Out Pattern (AWS)

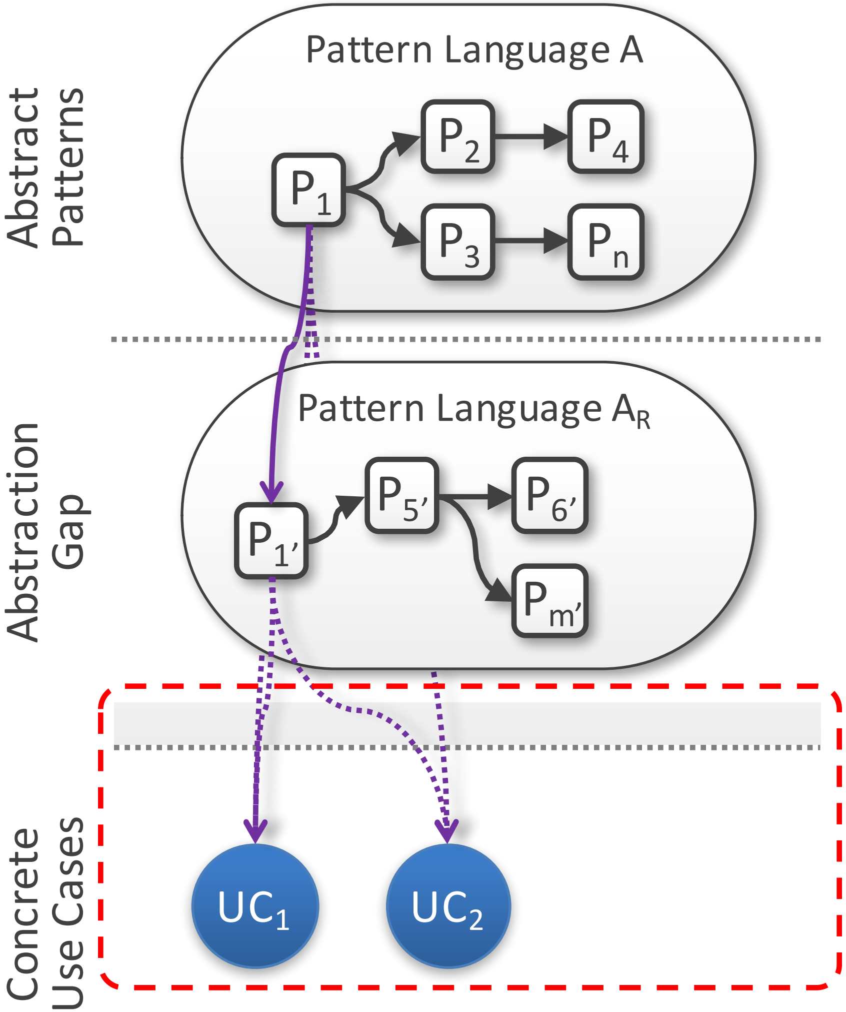
Context

Application components of a Distributed Application shall be scaled out automatically. Requests sent to an application shall be used as an indicator for the currently experienced workload from which the required number of components instances shall be deducted.

Case Study:
Genre-agnostic and Genre-specific Costume Patterns



Guided Solution Refinement



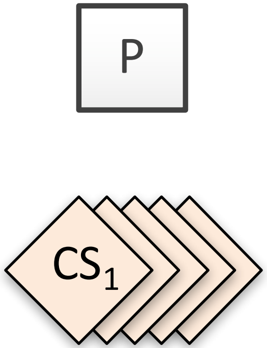
From ad hoc refinement for specific Use Cases

To guided solution refinement towards concrete solutions

How to connect and reuse existing Concrete Solutions?

Approach: Connecting Concrete Solutions

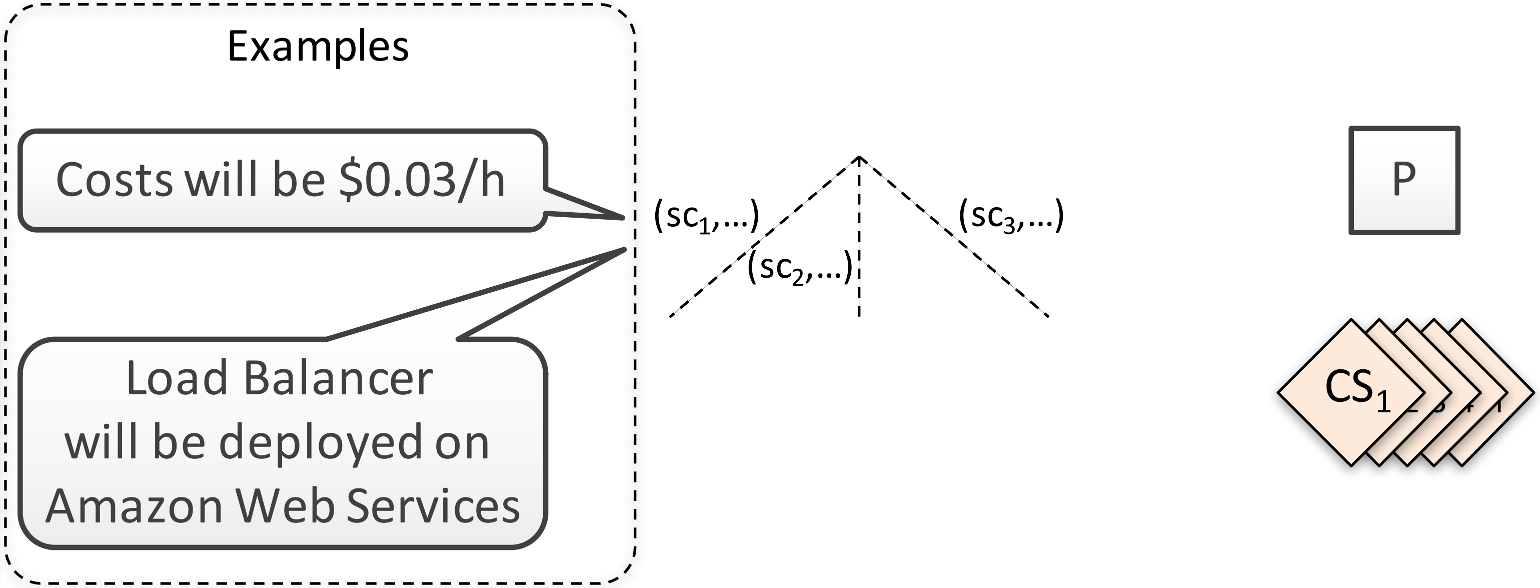
- Concrete Solutions are **concrete instances** of a pattern's solution
  - Concrete Solutions could be, e.g.,
    - Concrete programming code
    - Configuration files
    - AWS cloud formation templates
    - Workflow snippets
    - Costumes
    - …



- Aim: Change of pattern usage → From an act of design and construction of solutions to **find and reuse the right solutions**

Approach: Selection Criteria

- Human and machine readable *Selection Criteria* (sc) support selection of proper Concrete Solutions



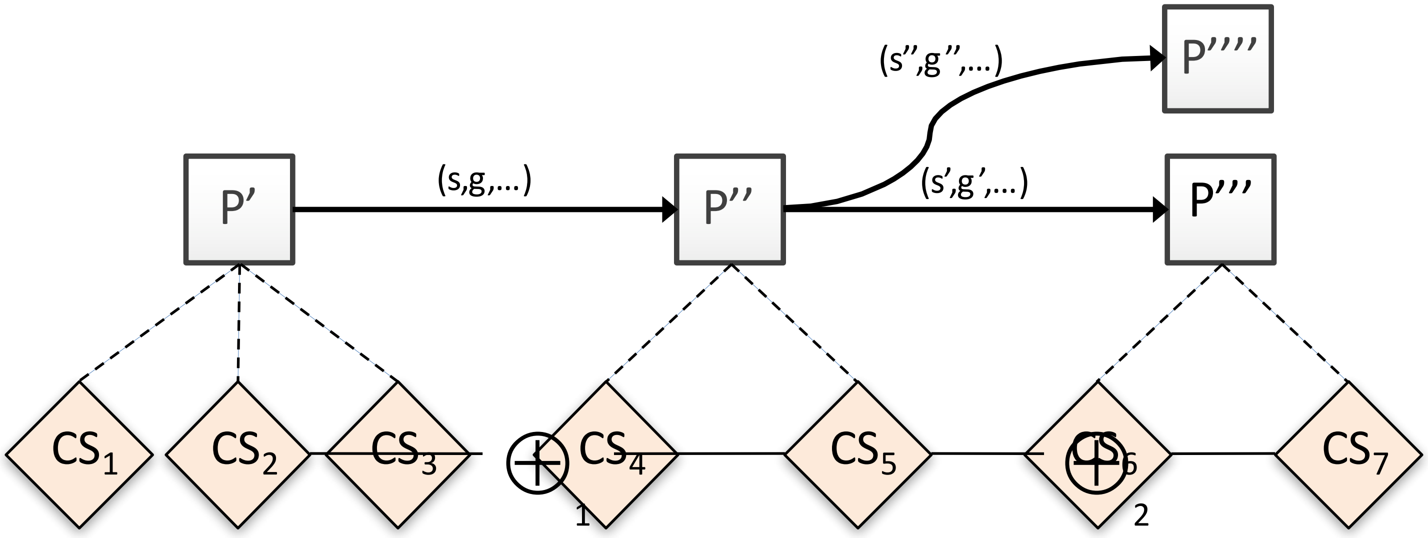
Examples

Costs will be $0.03/h

Load Balancer will be deployed on Amazon Web Services

Approach: Aggregating Concrete Solutions

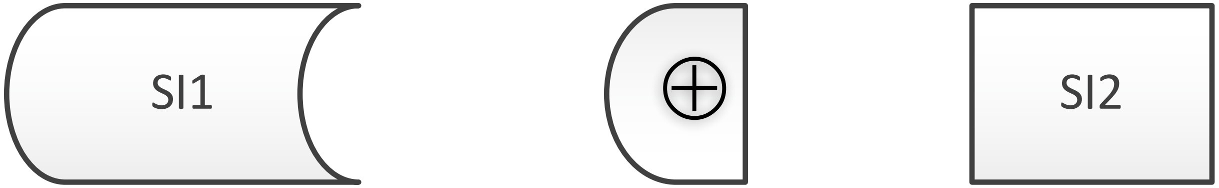
- Since patterns typically solve problems in combination, **also Concrete Solutions have to be combined**



- The means to combine Concrete Solutions is an *Aggregation Operator*

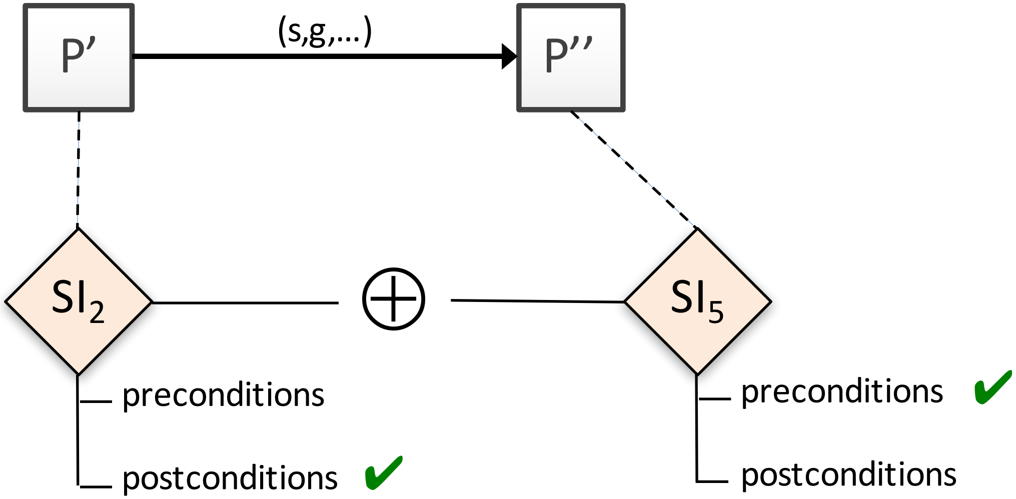
Approach: Aggregation Operator

- An Aggregation Operator **combines solutions on the level of Concrete Solutions**
- Each pattern language has **specific Aggregation Operators**
  - Pattern Language of a specific domain
    → Aggregation Operators **combine solution artifacts** of this specific domain
  - Example
    - An Aggregation Operator of a pattern language dealing with software design patterns may simply combine programming code



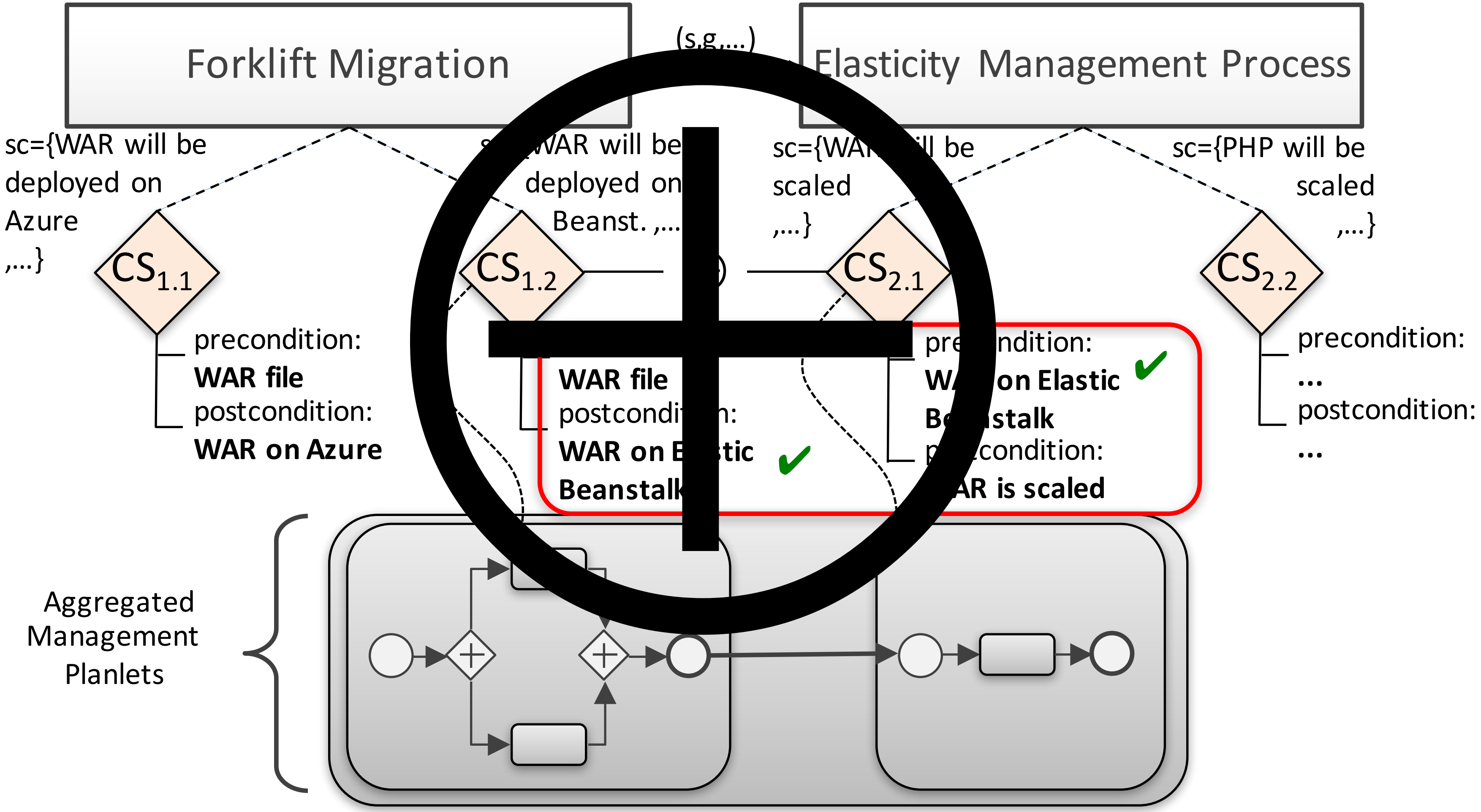
Approach: Preconditions and Postconditions

- Aggregation depends on *Preconditions* and *Postconditions* of Concrete Solutions



- Preconditions and Postconditions may define **functional and non-functional dependencies** between Concrete Solutions

Scenario: Management Planlets



(s,g,...)

Forklift Migration

Elasticity Management Process

sc={WAR will be deployed on Azure ,...}

s WAR will be deployed on Beanst. ,...

sc={WA ill be scaled ,...}

sc={PHP will be scaled ,...}

$CS_{1.1}$

$CS_{1.2}$

$CS_{2.1}$

$CS_{2.2}$

precondition: **WAR file**
postcondition: **WAR on Azure**

WAR file
postcondit n:
WAR on E tic Beanstalk ✓

pre ndition: **W on Elastic Be stalk** ✓
p condition: **AR is scaled**

precondition: **...**
postcondition: **...**

Aggregated Management Planlets

Agenda

Capturing Knowledge

Discovering Patterns

Patterns in the Humanities

From Patterns to Solutions

Summary

Conclusion

- Methodology
  - New method on how to capture patterns by storing and analyzing concrete solutions
- Patterns in the Humanities
  - By storing the concrete solutions a pattern gets verifiable (pattern provenance)
  - Opens the possibility to use analysis tools on the stored solutions to identify patterns (more on this in the afternoon session)
- MUSE Method & Scientific Paradigm
  - Mental model from natural sciences applied in humanities
- Formalization & Pattern Application
  - Improve working with patterns by supporting search of patterns and navigation through pattern languages
  - Improve the application of patterns by pattern refinement links and connecting concrete solutions and patterns

…and to be done

- Finding entries in pattern languages
- Finding optima
  - Optimal paths – ongoing research
  - Optimal solutions - ongoing research
- Formalizing aggregation operators
  - An algebra for solution aggregation
- Method and toolchain to leverage pattern refinement and concrete solutions
  - Guidance from architecture via refinements to concrete implementations