Pattern and Solution Repositories

University of Stuttgart
Universitätsstr. 38
70569 Stuttgart
Germany

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

Phone +49-711-685 88 487
Fax +49-711-685 88 472

Agenda

- Defining Patterns
- Approaching a Patterns Repository
- Patter Repository: PatternPedia
- Solution Repository: MUSE
- Summary

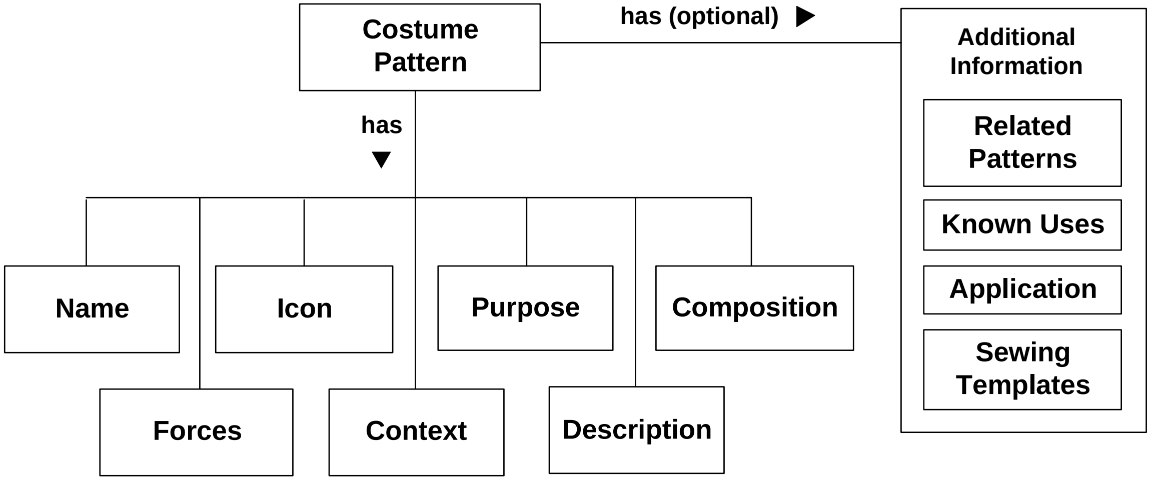
IAAS Research

Defining Patterns

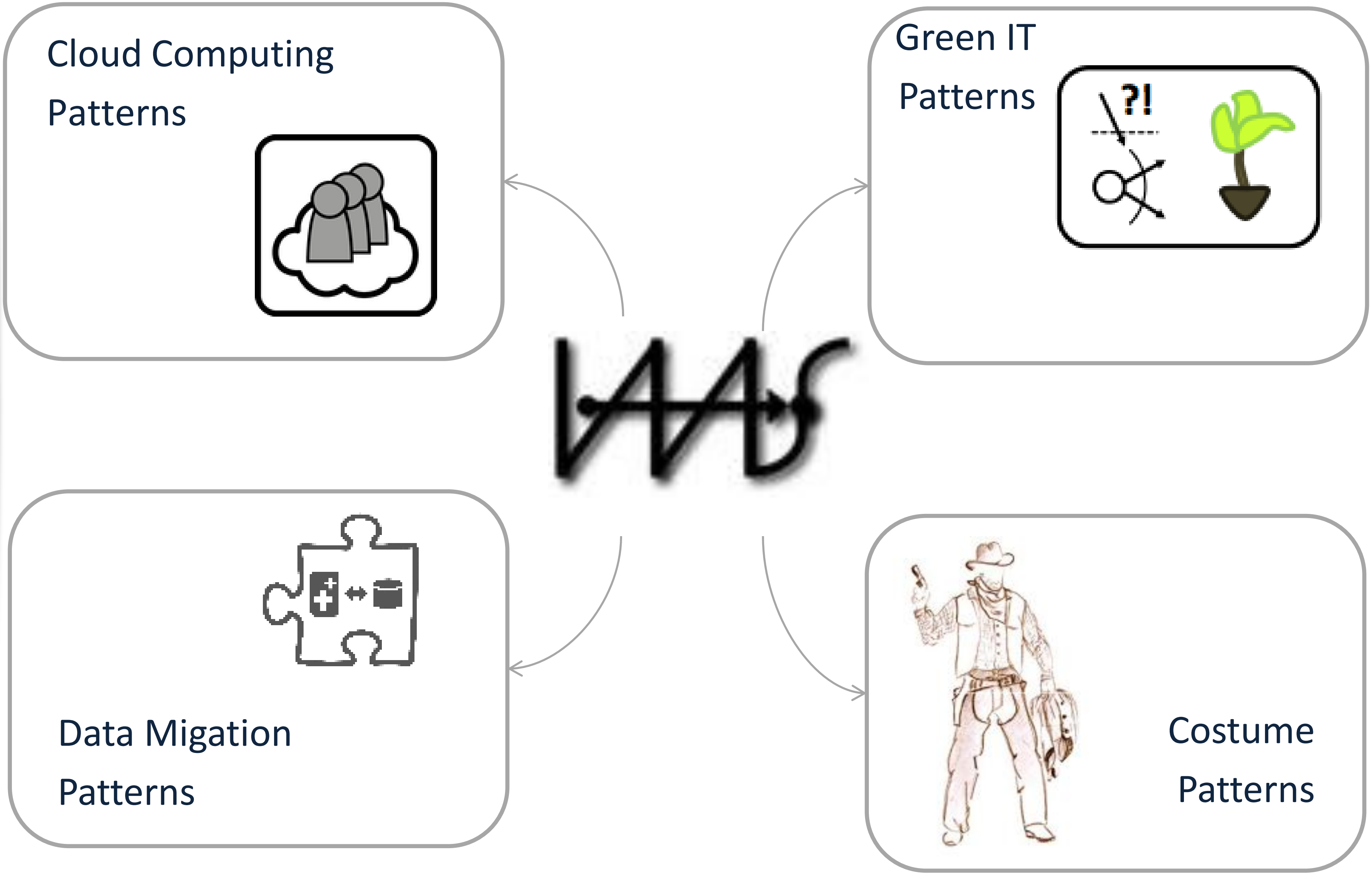
Research

Defining patterns

- A pattern is a proven solution to a re-occurring problem
- A pattern is a concept that aims to capture the best solution in an abstract way to make this knowledge reusable
- A pattern language is a set of patterns conforming to a particular pattern format as well as cross-references between these patterns



Research on Patterns at IAAS



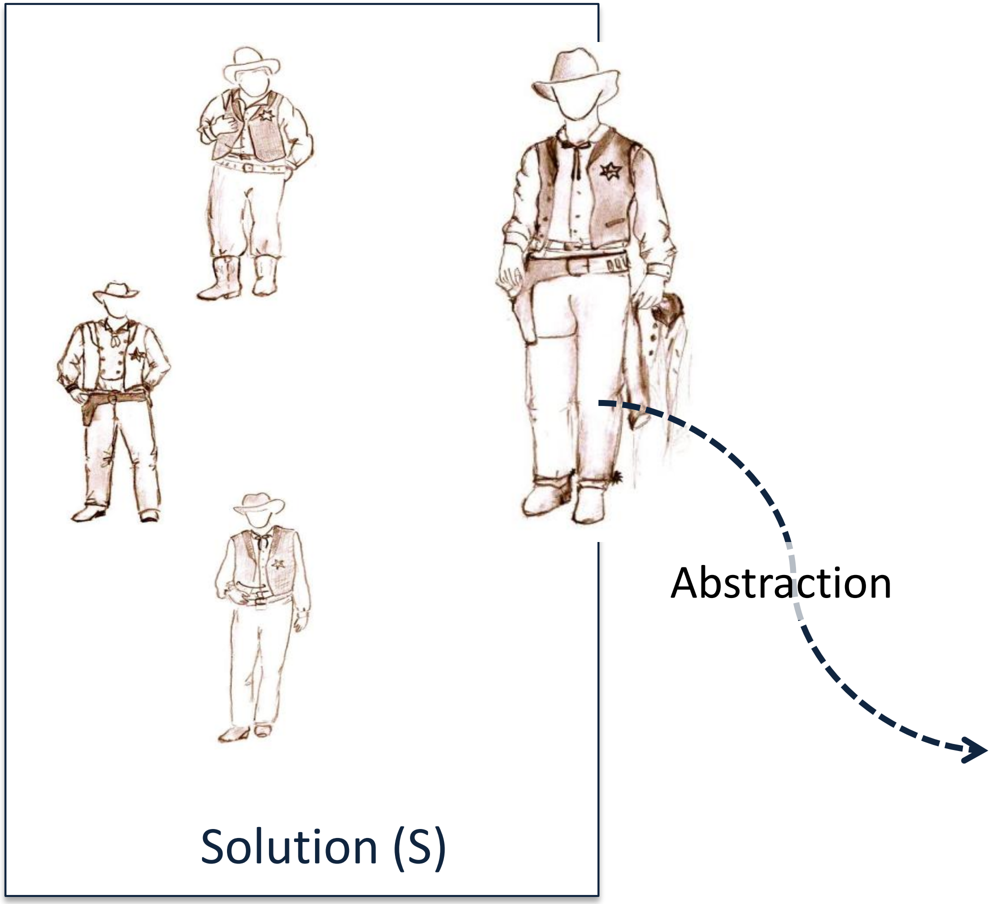
Cloud Computing Patterns

Green IT Patterns

Data Migation Patterns

Costume Patterns

Pattern Language – Patterns Abstraction

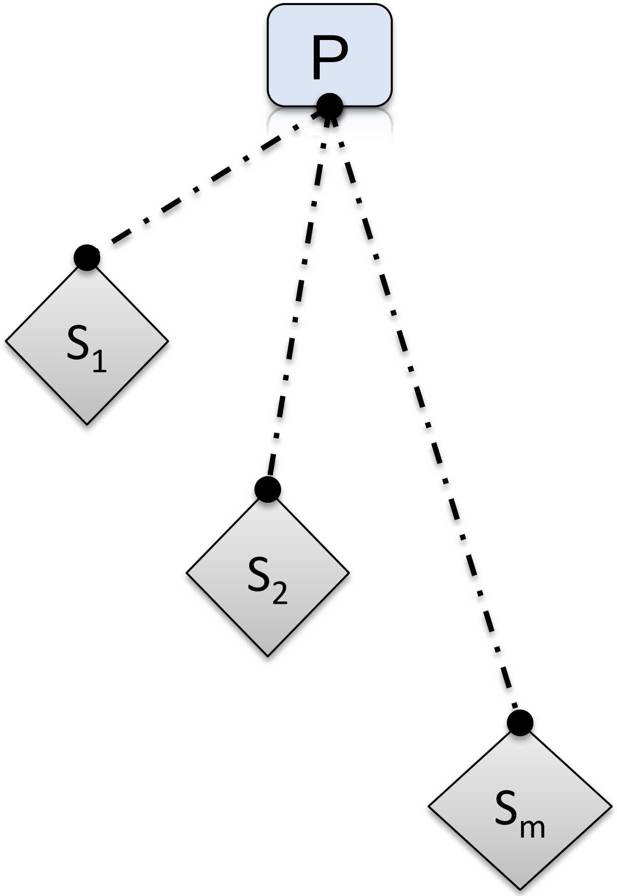


Solution (S)

Abstraction

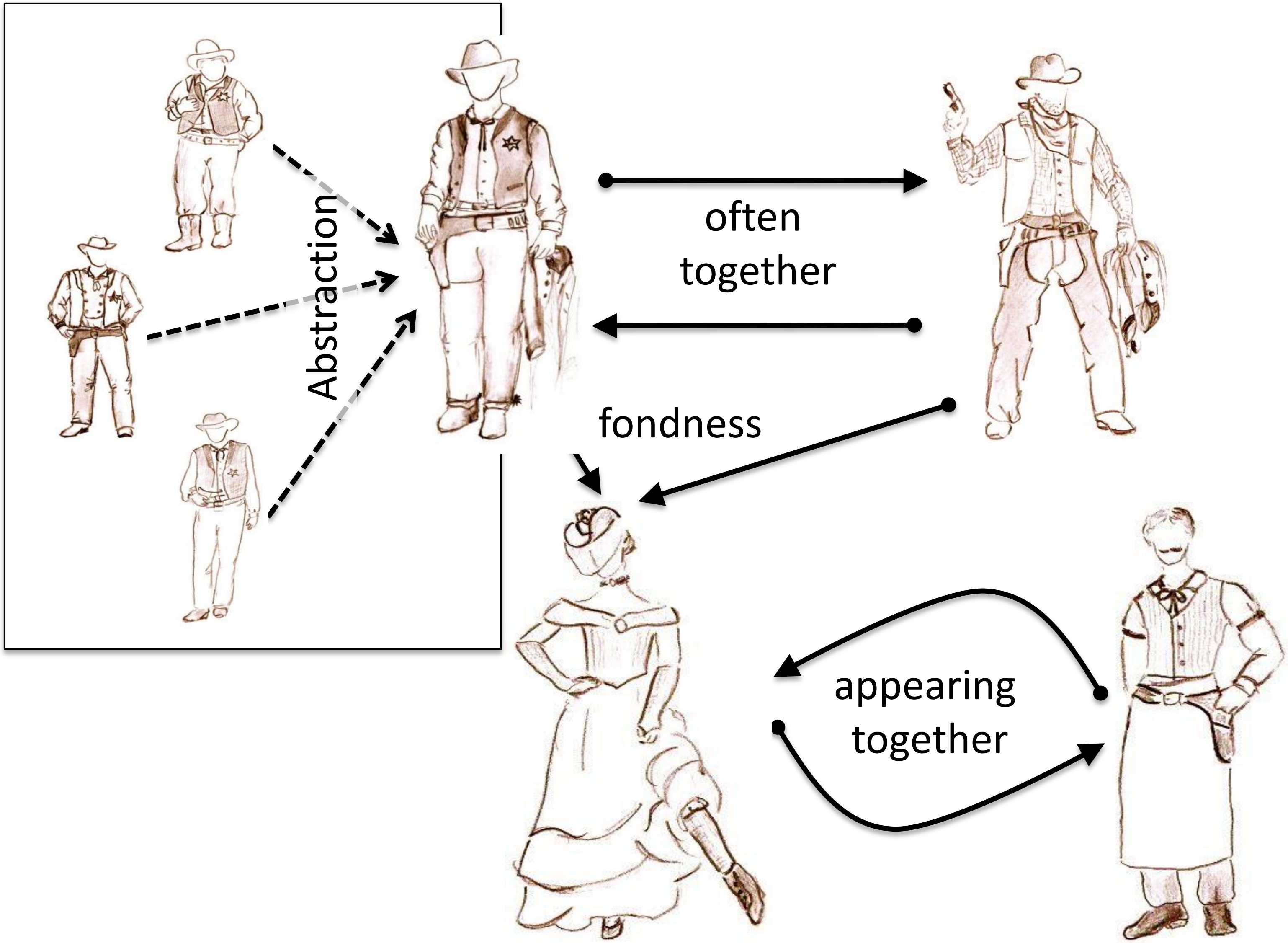
Pattern (P)

Patterns and Solutions

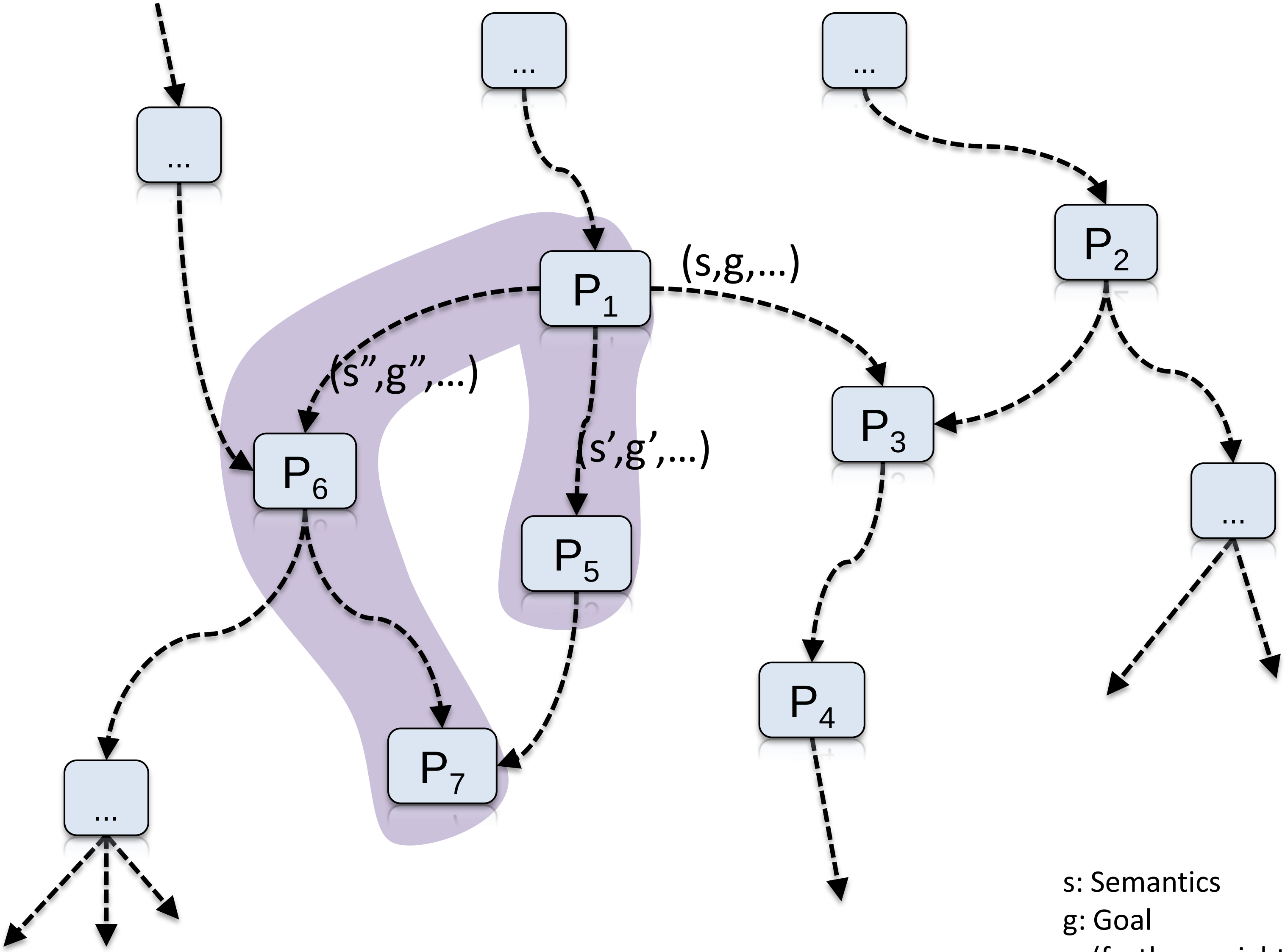


- A pattern {P} provides the „advice“ how to solve a recurring problem
  - abstract solution (i.e. a high-level description of how to solve the problem in principle)
  - independent of any concrete environment
  - a realization requires implementation effort
- Patterns are connected to concrete solutions $\{S_i\}$
  - this solutions S_i are concrete
  - tell exactly how to implement the abstract solution S^A in a concrete environment

Pattern Language – Patterns and their cross-references



Solution Paths

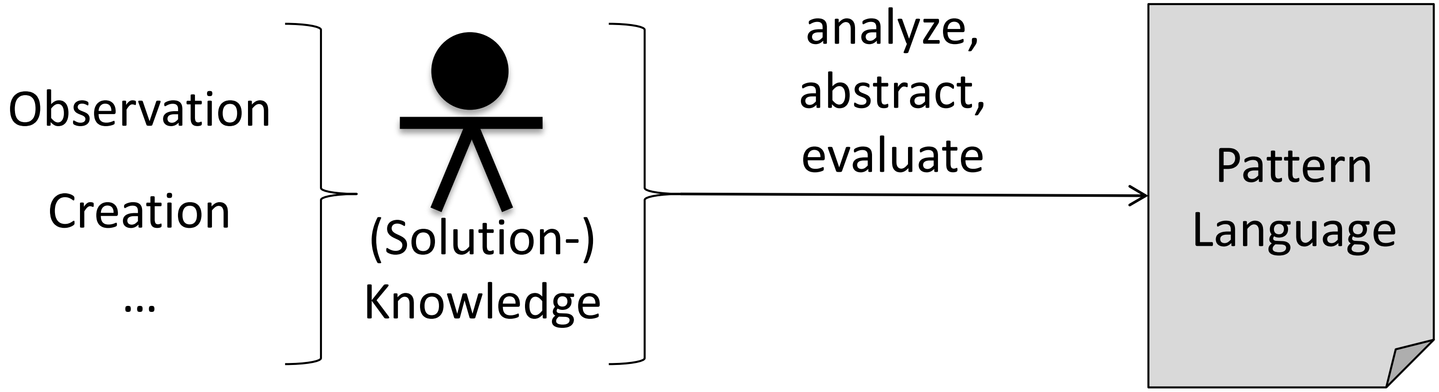


s: Semantics
g: Goal
… (further weights)

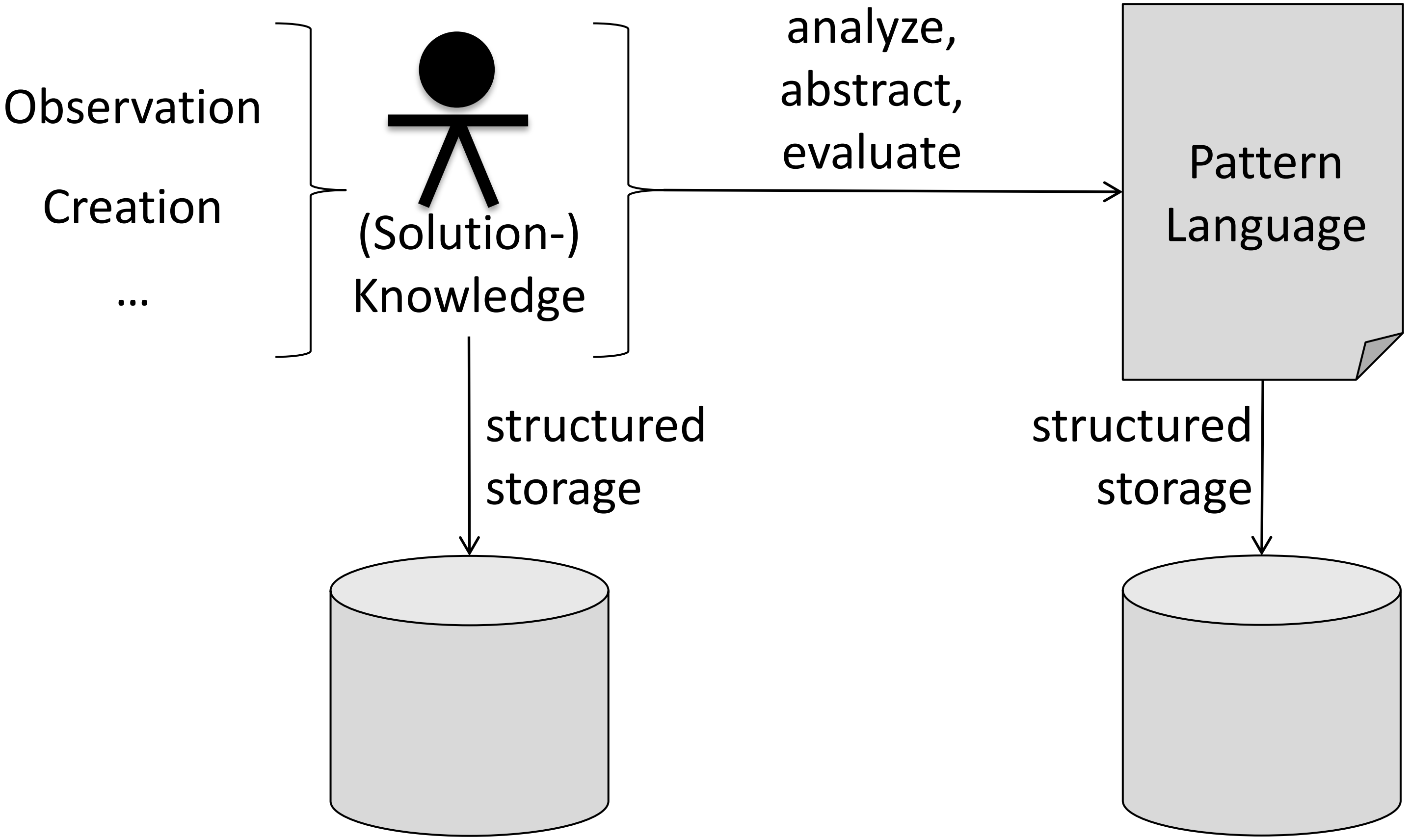
Approaching a Patterns Repository

IAAS Research

Development of Pattern Languages: Today



Knowledge and Pattern Languages: Use of Repositories



Observation

Creation

…

(Solution-) Knowledge

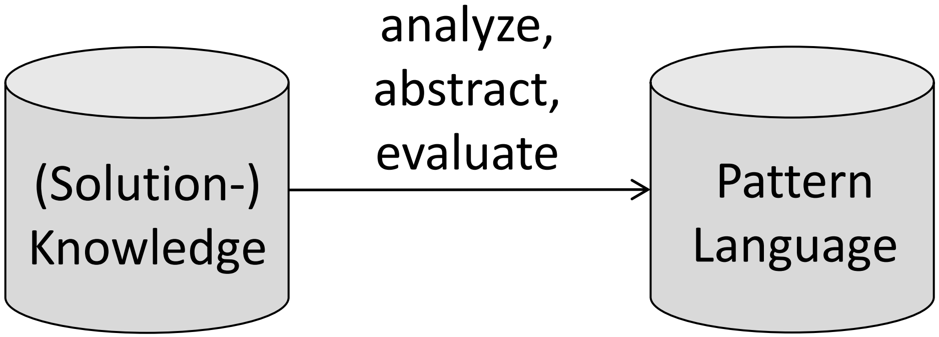
analyze, abstract, evaluate

Pattern Language

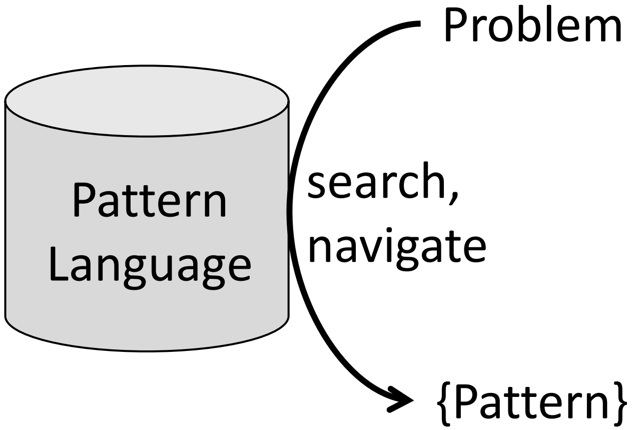
structured storage

structured storage

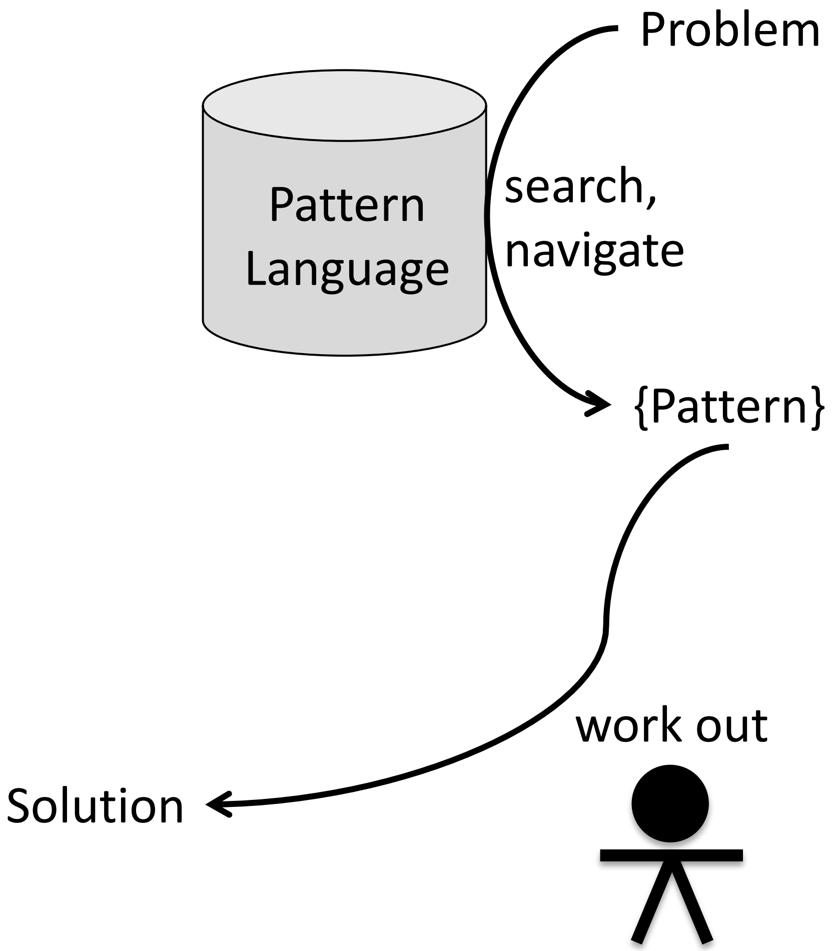
Deriving Patterns from Stored Knowledge



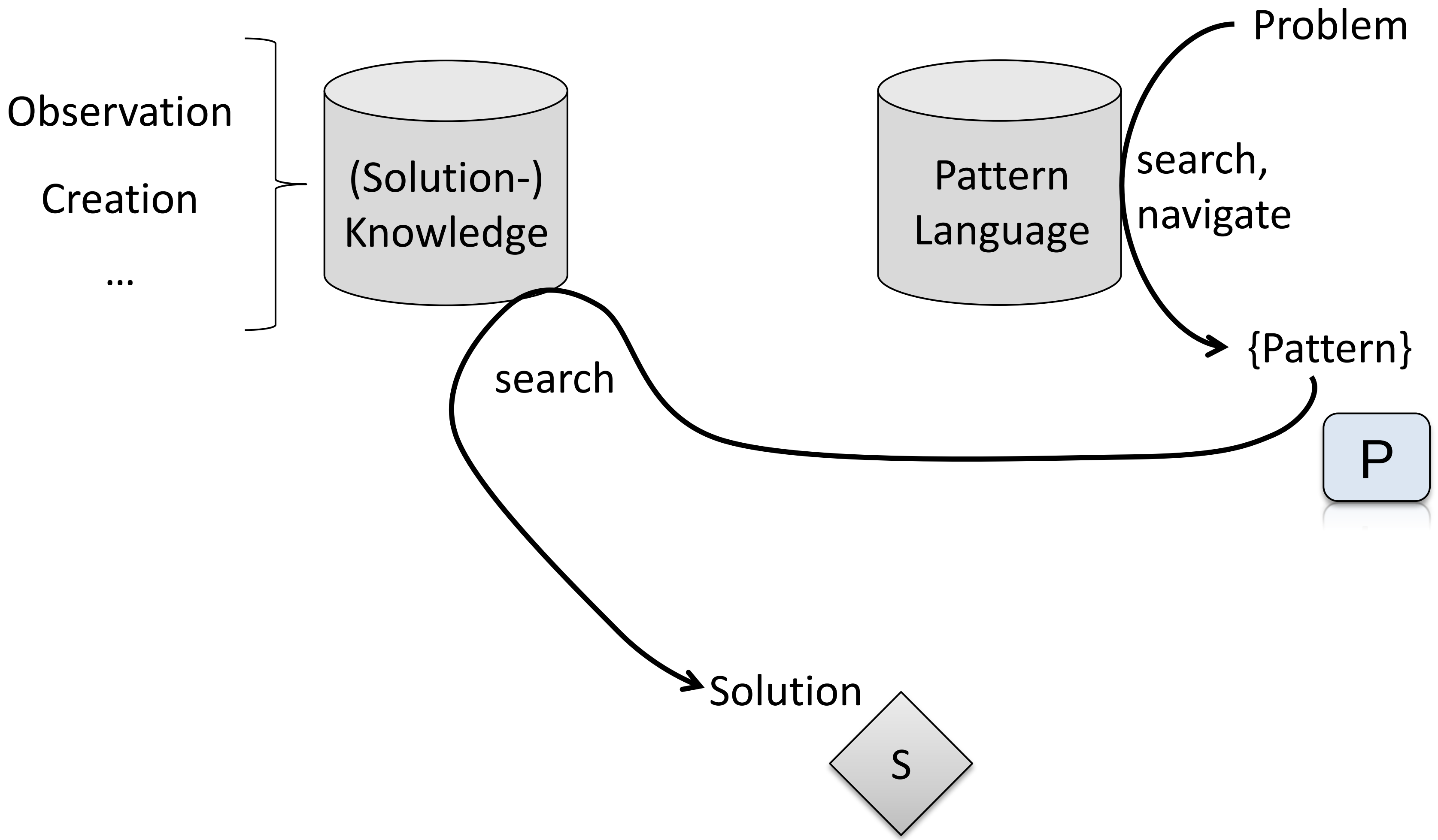
Using Stored Patterns



From Patterns to Solutions: A Creative Act



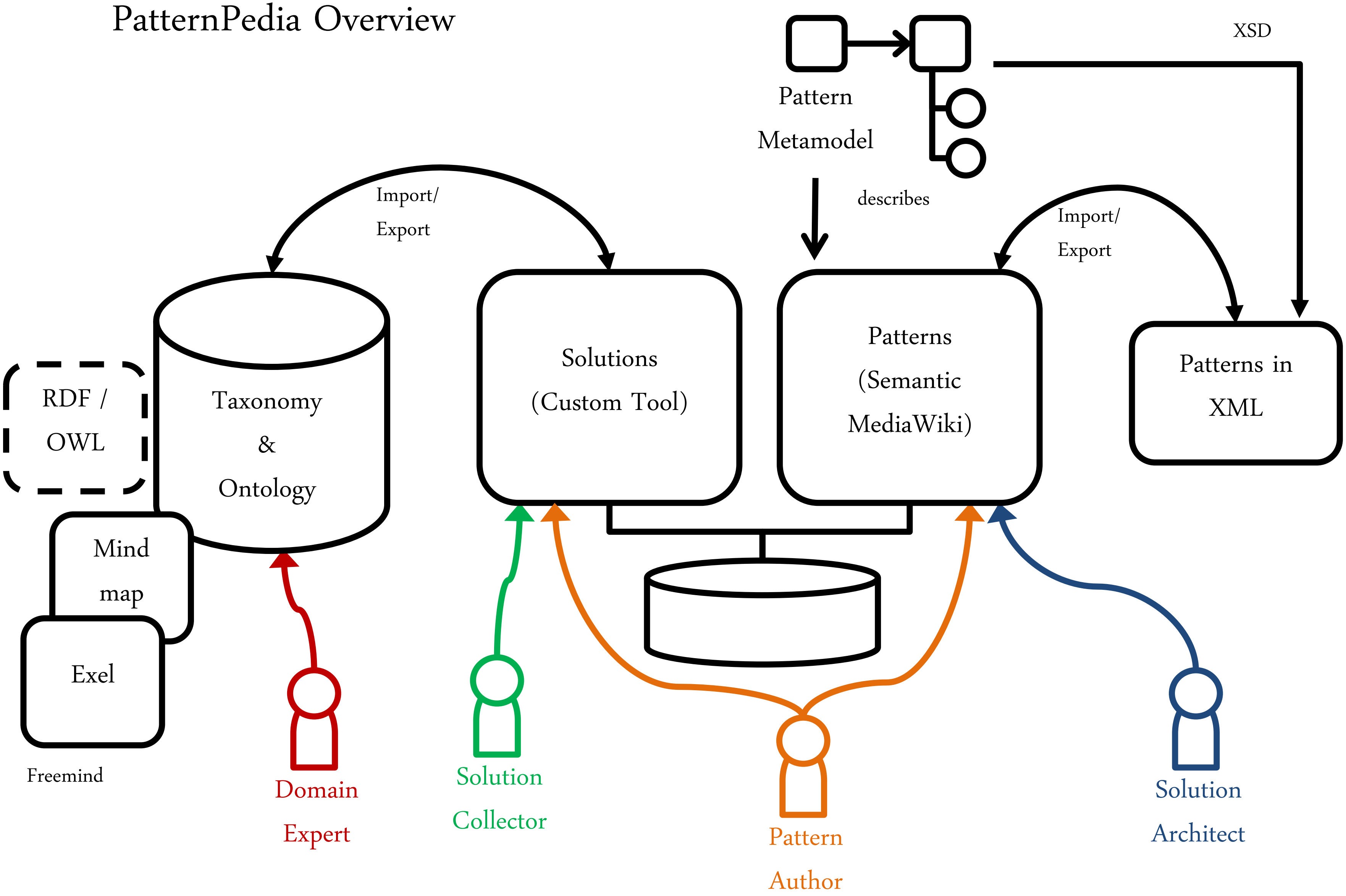
Final Goal


Observation
Creation
…
(Solution-) Knowledge
Pattern Language
Problem
search, navigate
{Pattern}
P
search
Solution
S

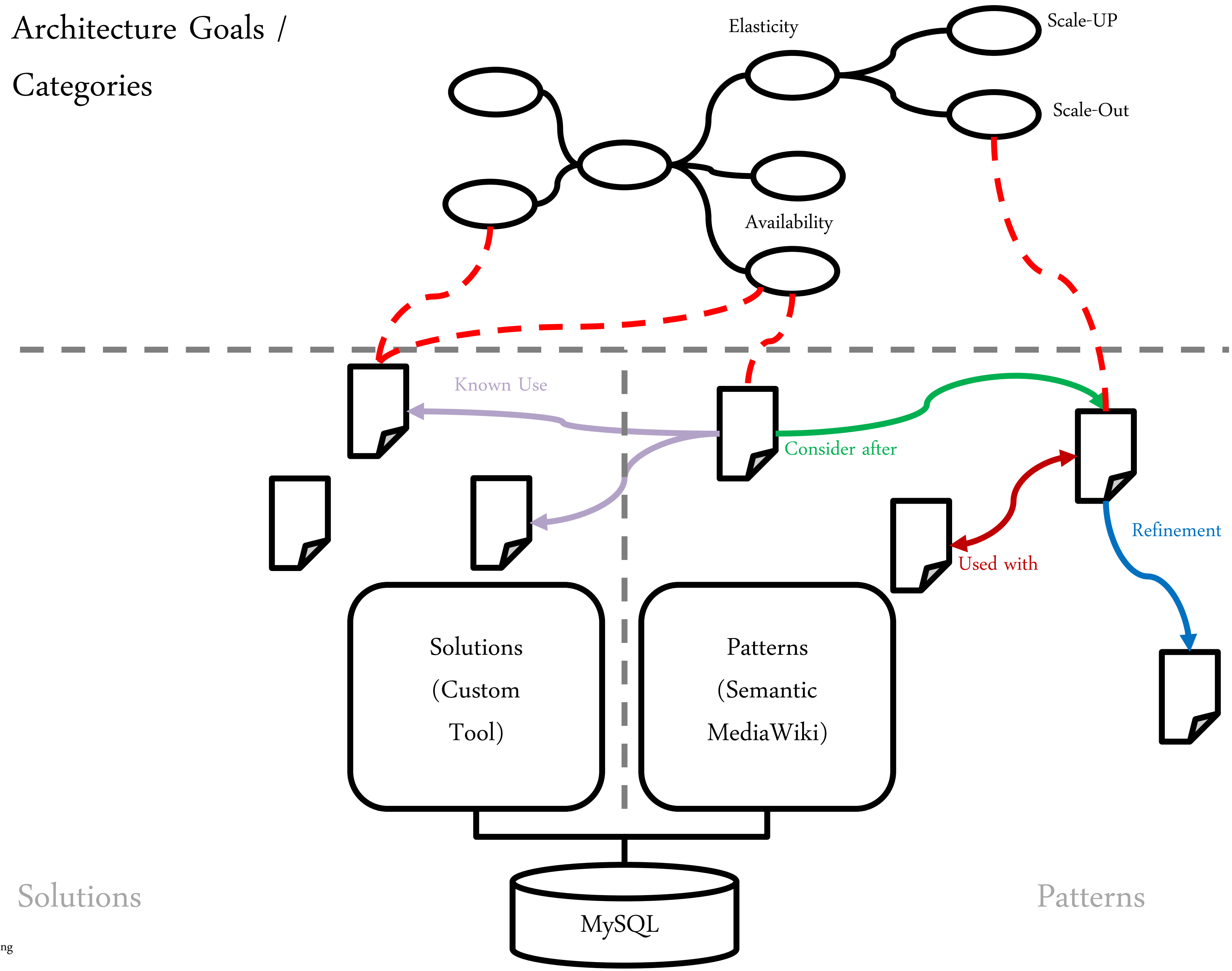
Pattern Repository: PatternPedia

IAAS Research

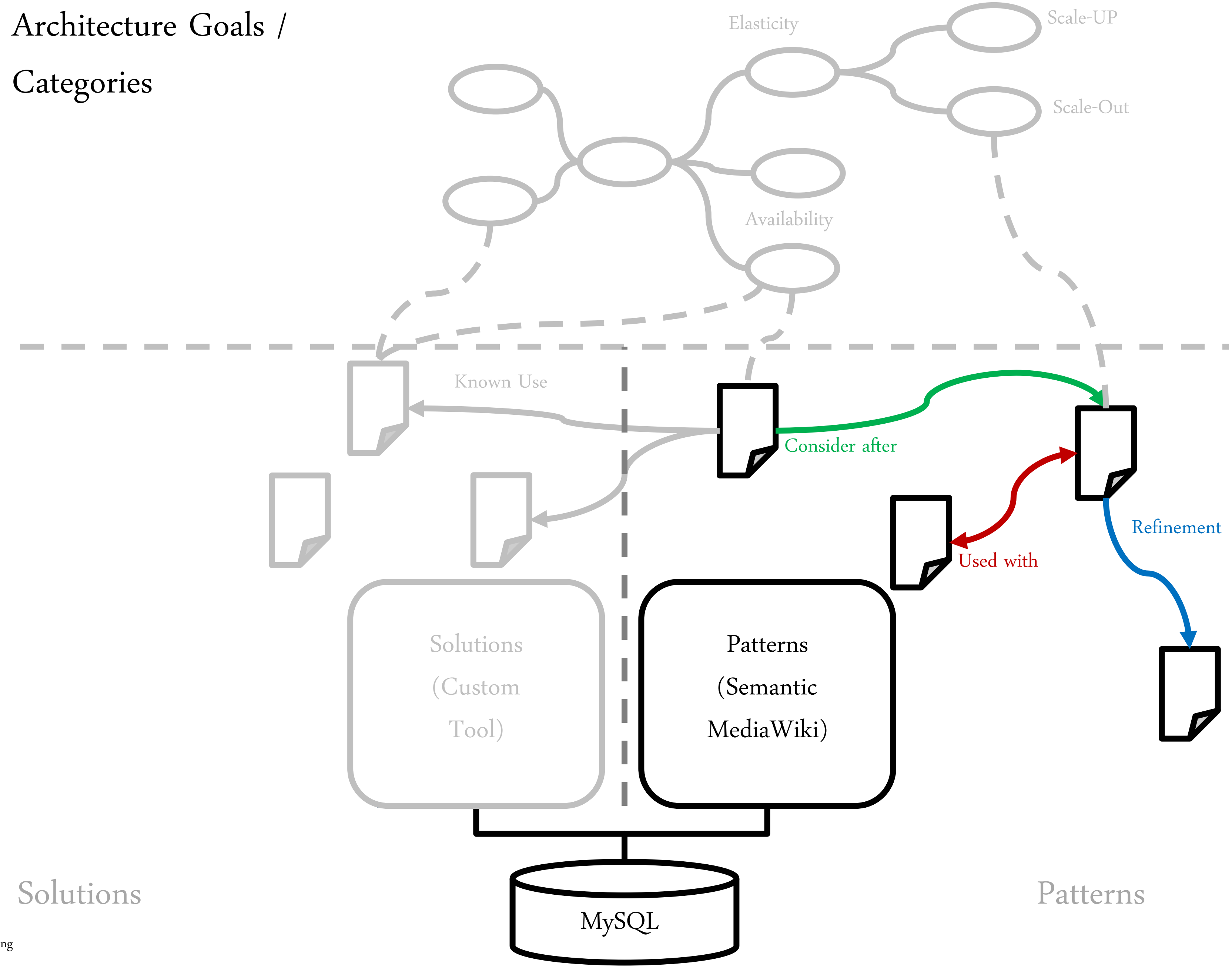
PatternPedia Overview



© Fehling

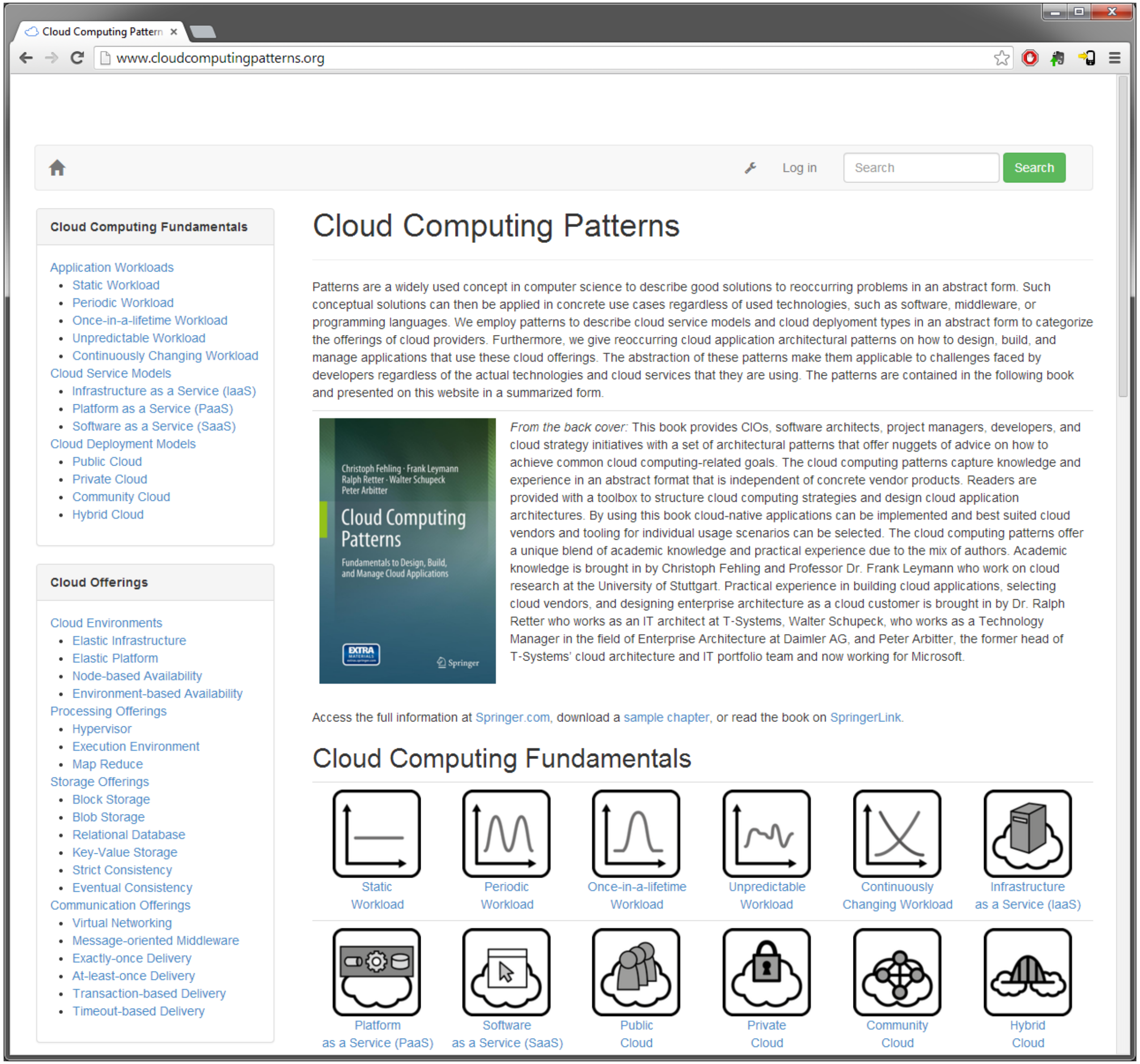

Architecture Goals /
Categories
Elasticity
Scale-UP
Scale-Out
Availability
Known Use
Consider after
Used with
Refinement
Solutions
(Custom
Tool)
Patterns
(Semantic
MediaWiki)
Solutions
Patterns
MySQL

Architecture Goals / Categories



© Fehling

PatternPedia: http://www.cloudcomputingpatterns.org



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
- Processing Offerings
  - Hypervisor
  - Execution Environment
  - Map Reduce
- Storage Offerings
  - Block Storage
  - Blob Storage
  - Relational Database
  - Key-Value Storage
  - Strict Consistency
  - Eventual Consistency
- Communication Offerings
  - Virtual Networking
  - Message-oriented Middleware
  - Exactly-once Delivery
  - At-least-once Delivery
  - Transaction-based Delivery
  - Timeout-based Delivery

Cloud Computing Patterns

Patterns are a widely used concept in computer science to describe good solutions to reoccurring problems in an abstract form. Such conceptual solutions can then be applied in concrete use cases regardless of used technologies, such as software, middleware, or programming languages. We employ patterns to describe cloud service models and cloud deplyoment types in an abstract form to categorize the offerings of cloud providers. Furthermore, we give reoccurring cloud application architectural patterns on how to design, build, and manage applications that use these cloud offerings. The abstraction of these patterns make them applicable to challenges faced by developers regardless of the actual technologies and cloud services that they are using. The patterns are contained in the following book and presented on this website in a summarized form.

Christoph Fehling · Frank Leymann
Ralph Retter · Walter Schupeck
Peter Arbitter
Cloud Computing Patterns
Fundamentals to Design, Build, and Manage Cloud Applications
EXTRA MATERIALS
Springer

From the back cover: This book provides CIOs, software architects, project managers, developers, and cloud strategy initiatives with a set of architectural patterns that offer nuggets of advice on how to achieve common cloud computing-related goals. The cloud computing patterns capture knowledge and experience in an abstract format that is independent of concrete vendor products. Readers are provided with a toolbox to structure cloud computing strategies and design cloud application architectures. By using this book cloud-native applications can be implemented and best suited cloud vendors and tooling for individual usage scenarios can be selected. The cloud computing patterns offer a unique blend of academic knowledge and practical experience due to the mix of authors. Academic knowledge is brought in by Christoph Fehling and Professor Dr. Frank Leymann who work on cloud research at the University of Stuttgart. Practical experience in building cloud applications, selecting cloud vendors, and designing enterprise architecture as a cloud customer is brought in by Dr. Ralph Retter who works as an IT architect at T-Systems, Walter Schupeck, who works as a Technology Manager in the field of Enterprise Architecture at Daimler AG, and Peter Arbitter, the former head of T-Systems' cloud architecture and IT portfolio team and now working for Microsoft.

Access the full information at Springer.com, download a sample chapter, or read the book on SpringerLink.

Cloud Computing Fundamentals

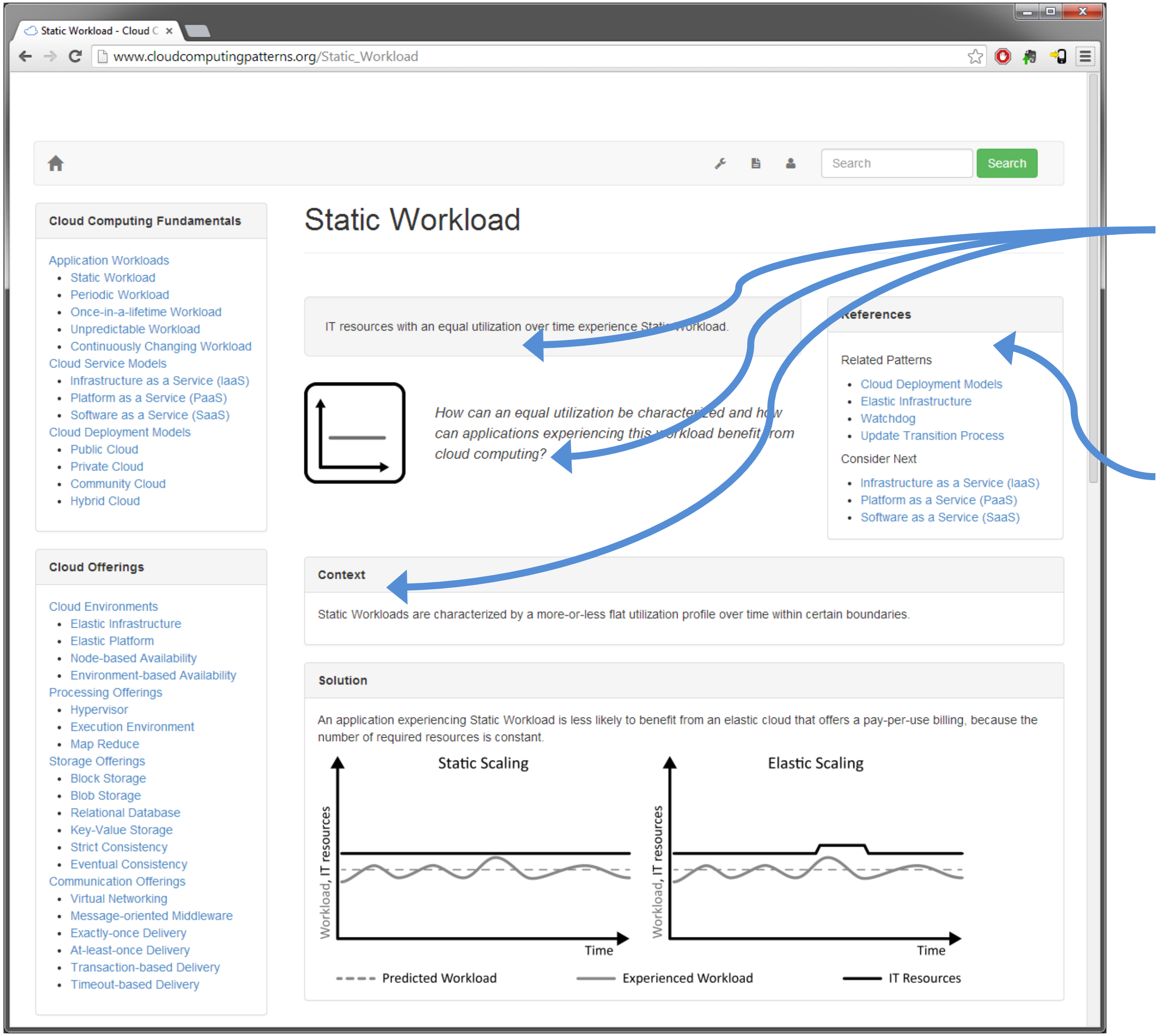
Static Workload
Periodic Workload
Once-in-a-lifetime Workload
Unpredictable Workload
Continuously Changing Workload
Infrastructure as a Service (IaaS)
Platform as a Service (PaaS)
Software as a Service (SaaS)
Public Cloud
Private Cloud
Community Cloud
Hybrid Cloud

Tooling is based on MediaWiki (used for Wikipedia)

+

Semantic Extensions

PatternPedia: Pattern Format



Static Workload - Cloud C ×
www.cloudcomputingpatterns.org/Static_Workload
Search
Search

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

Processing Offerings
- Hypervisor
- Execution Environment
- Map Reduce

Storage Offerings
- Block Storage
- Blob Storage
- Relational Database
- Key-Value Storage
- Strict Consistency
- Eventual Consistency

Communication Offerings
- Virtual Networking
- Message-oriented Middleware
- Exactly-once Delivery
- At-least-once Delivery
- Transaction-based Delivery
- Timeout-based Delivery

Static Workload

IT resources with an equal utilization over time experience Static Workload.

How can an equal utilization be characterized and how can applications experiencing this workload benefit from cloud computing?

References

Related Patterns
- Cloud Deployment Models
- Elastic Infrastructure
- Watchdog
- Update Transition Process

Consider Next
- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

Context

Static Workloads are characterized by a more-or-less flat utilization profile over time within certain boundaries.

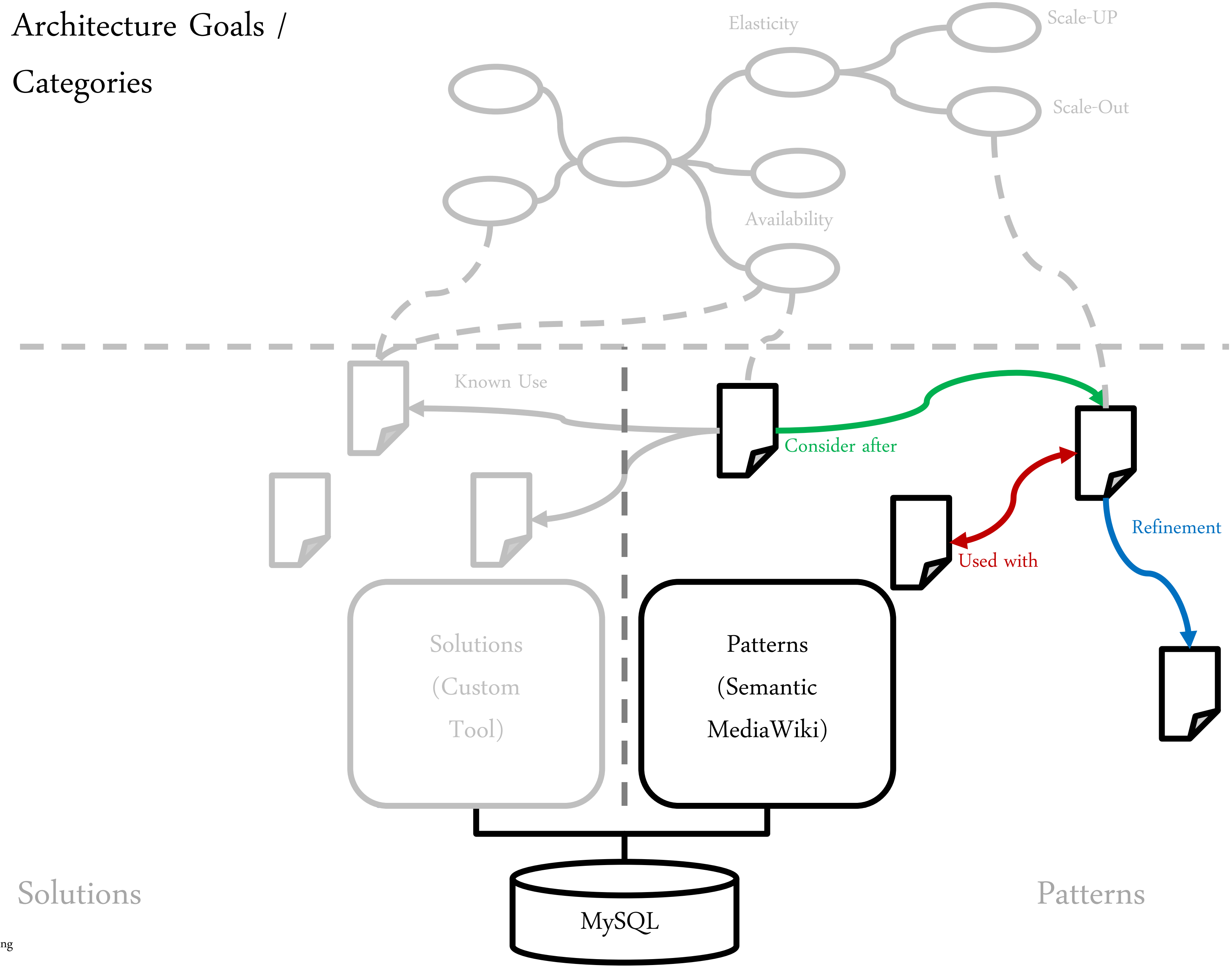
Solution

An application experiencing Static Workload is less likely to benefit from an elastic cloud that offers a pay-per-use billing, because the number of required resources is constant.

Pattern Format is enforced

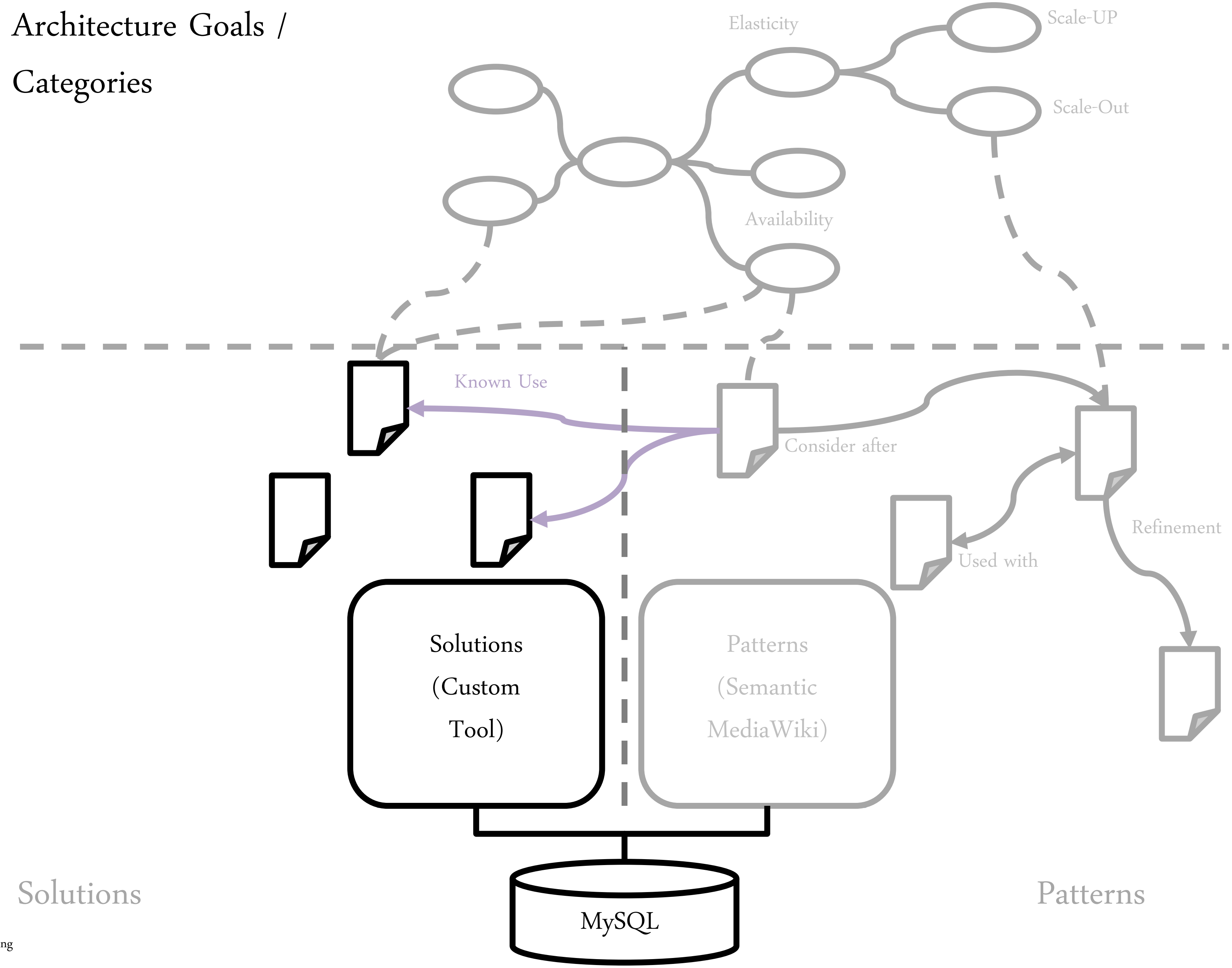
Semantic Links are summarized in Info Boxes

Architecture Goals / Categories



© Fehling

Architecture Goals / Categories

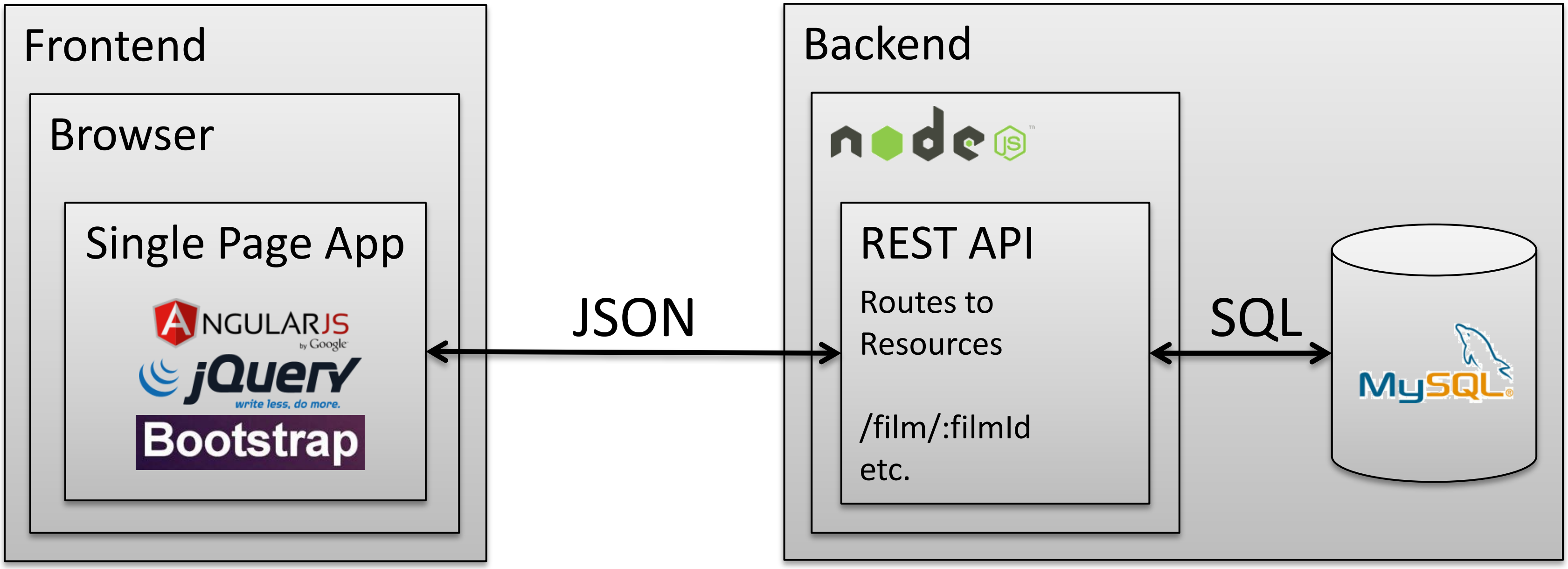


© Fehling

Solution Repository:
MUSE-Tool

IAAS Research

Solution Architecture



Frontend

Browser

Single Page App

JSON

Backend

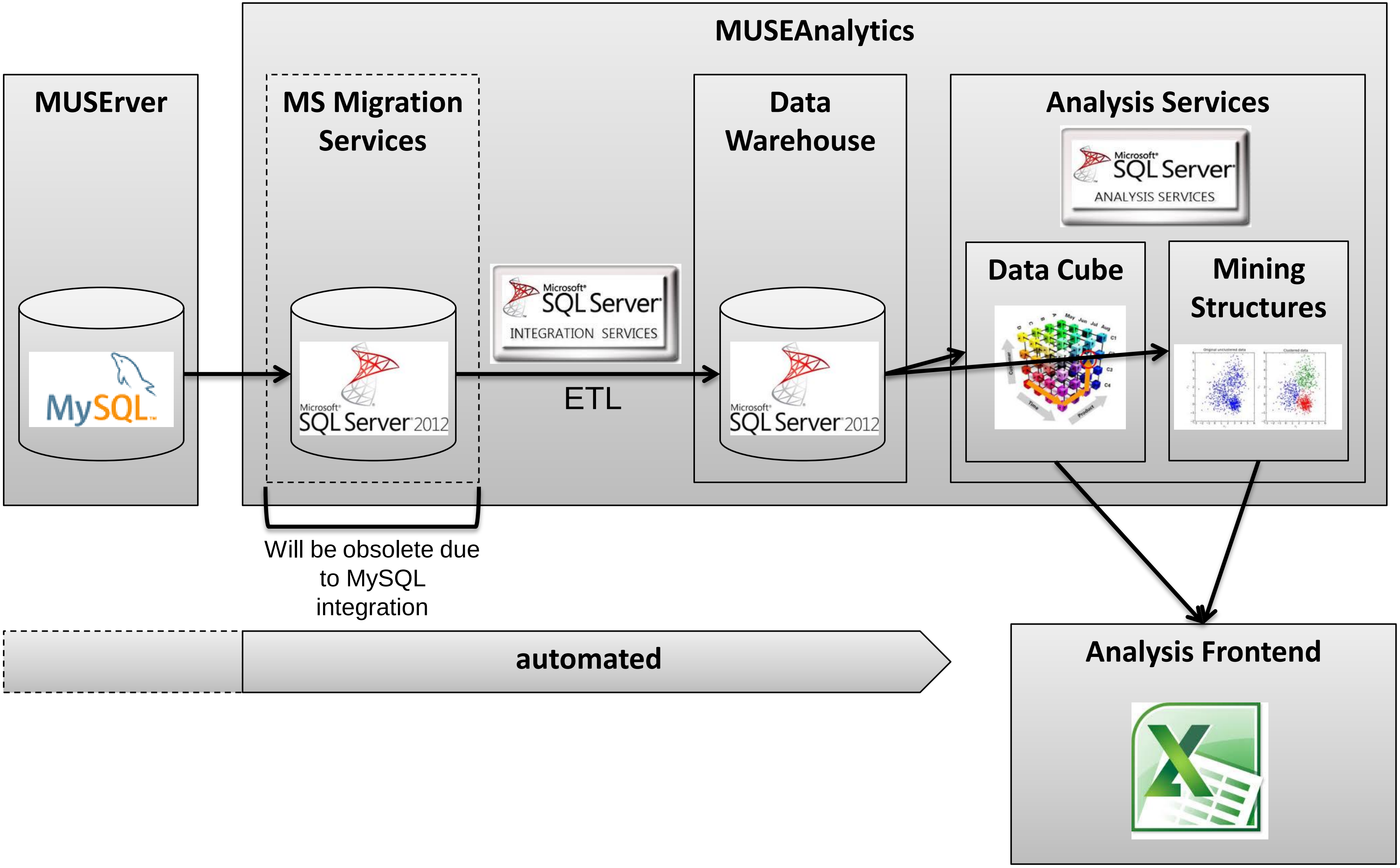
REST API

Routes to Resources

/film/:filmId
etc.

SQL

MUSE – Queries



MUSEAnalytics

MUSErver

MS Migration Services

Data Warehouse

Analysis Services

Data Cube

Mining Structures

ETL

Will be obsolete due to MySQL integration

automated

Analysis Frontend

MySQL
Microsoft SQL Server 2012
Microsoft SQL Server INTEGRATION SERVICES
Microsoft SQL Server 2012
Microsoft SQL Server ANALYSIS SERVICES

Summary

Research

Summary

- Methodology:
  - New method on how to capture patterns by also storing concrete solutions
- Content:
  - By storing the concrete solutions a pattern gets verifiable (pattern provenance)
  - Opens the possibility to use analysis tools on the stored solutions to find patterns
- Practical work:
  - Improve working with patterns by supporting search of patterns and navigation through pattern languages
  - Improve the application of patterns by connecting concrete solutions and patterns

Thanks for your attention!

VAS Research