Empowerment of Subject-Matter Experts

Why you should completely isolate subject matter from technical concerns – and how.

Markus Voelter

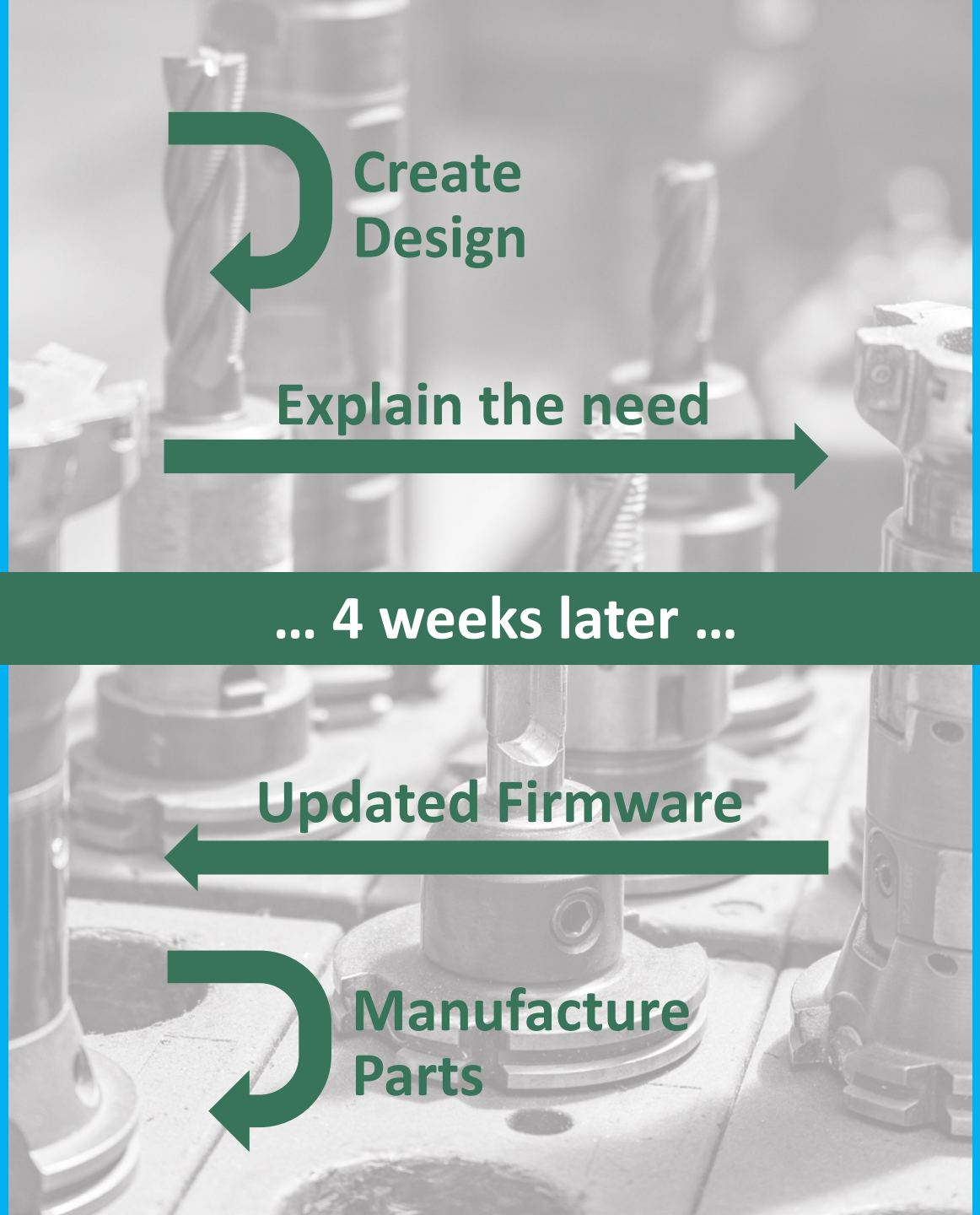
voelter@acm.org
http://voelter.de

Engineer at
XYZ Company

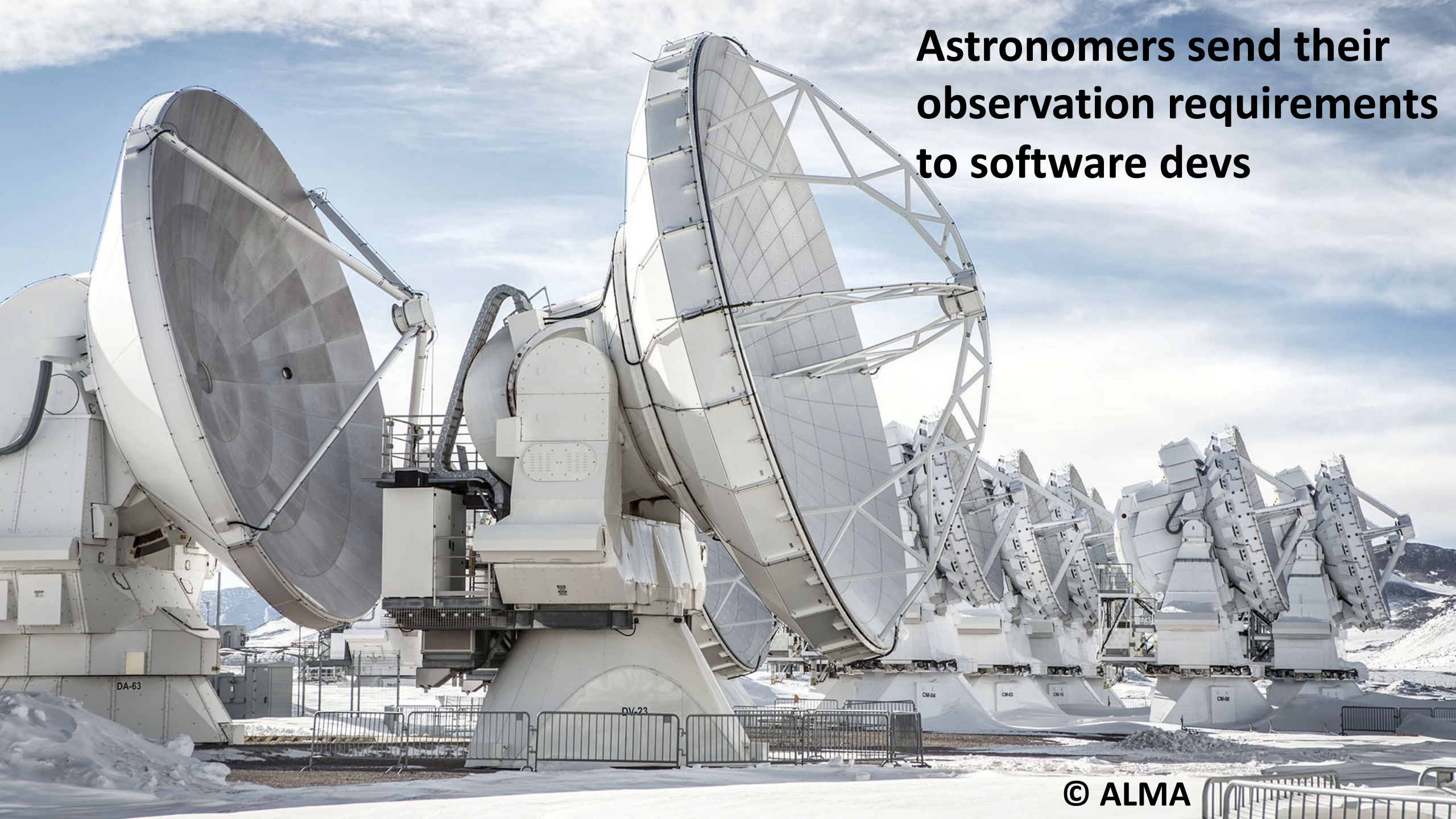
User

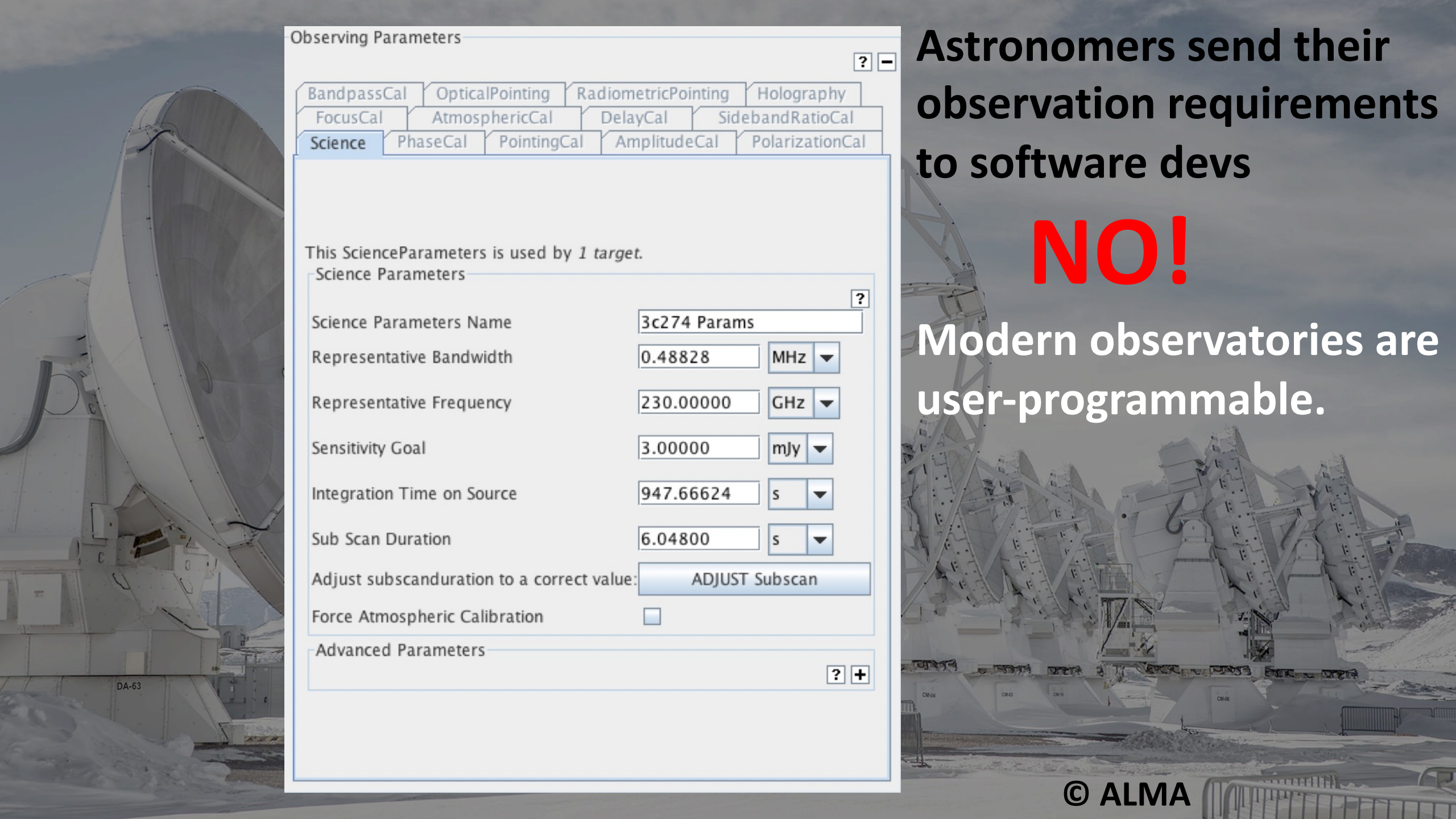
CNC Machine
Manufacturer

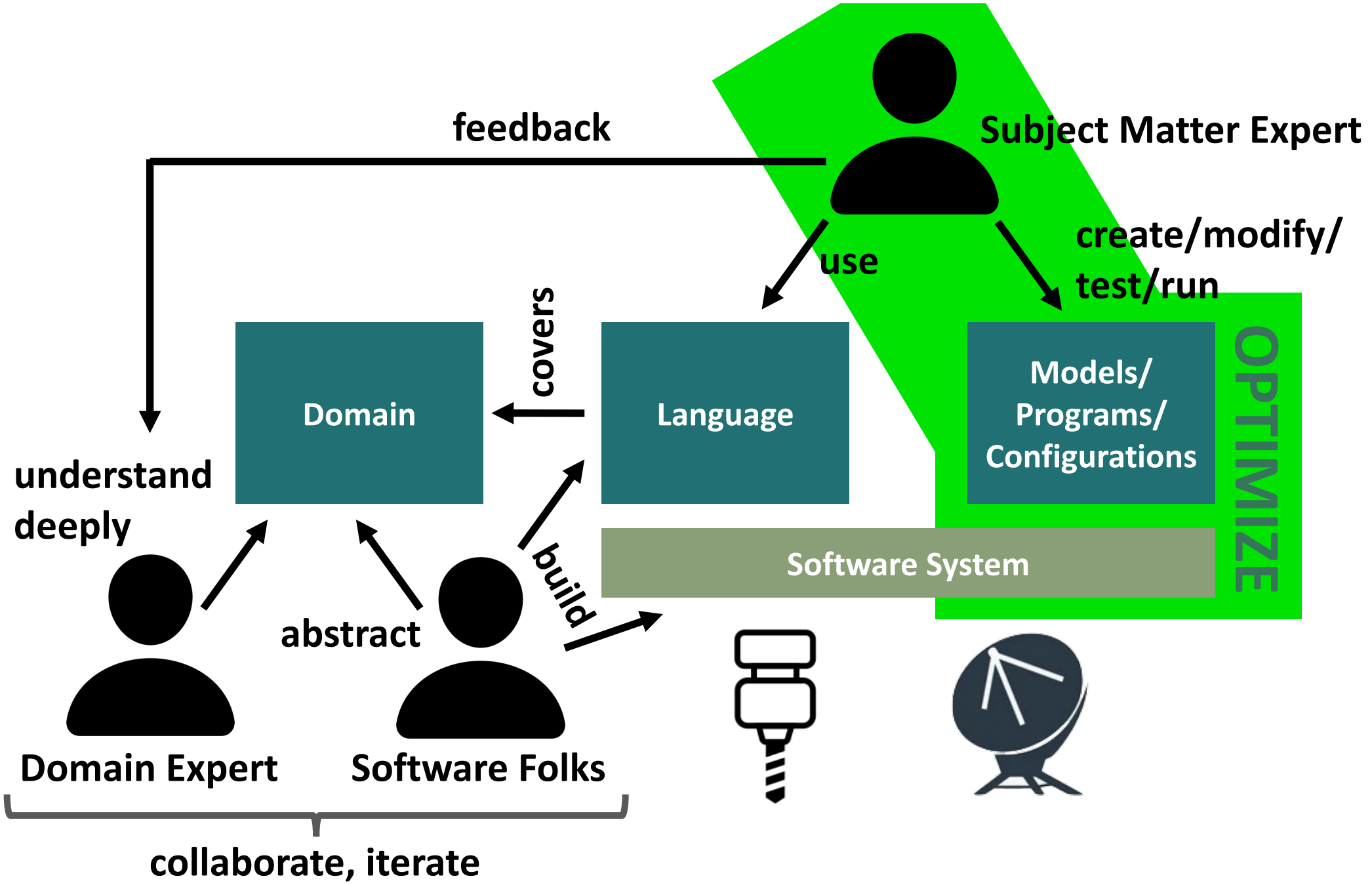
System Dev




NO!
CNC mills are user-programmable.

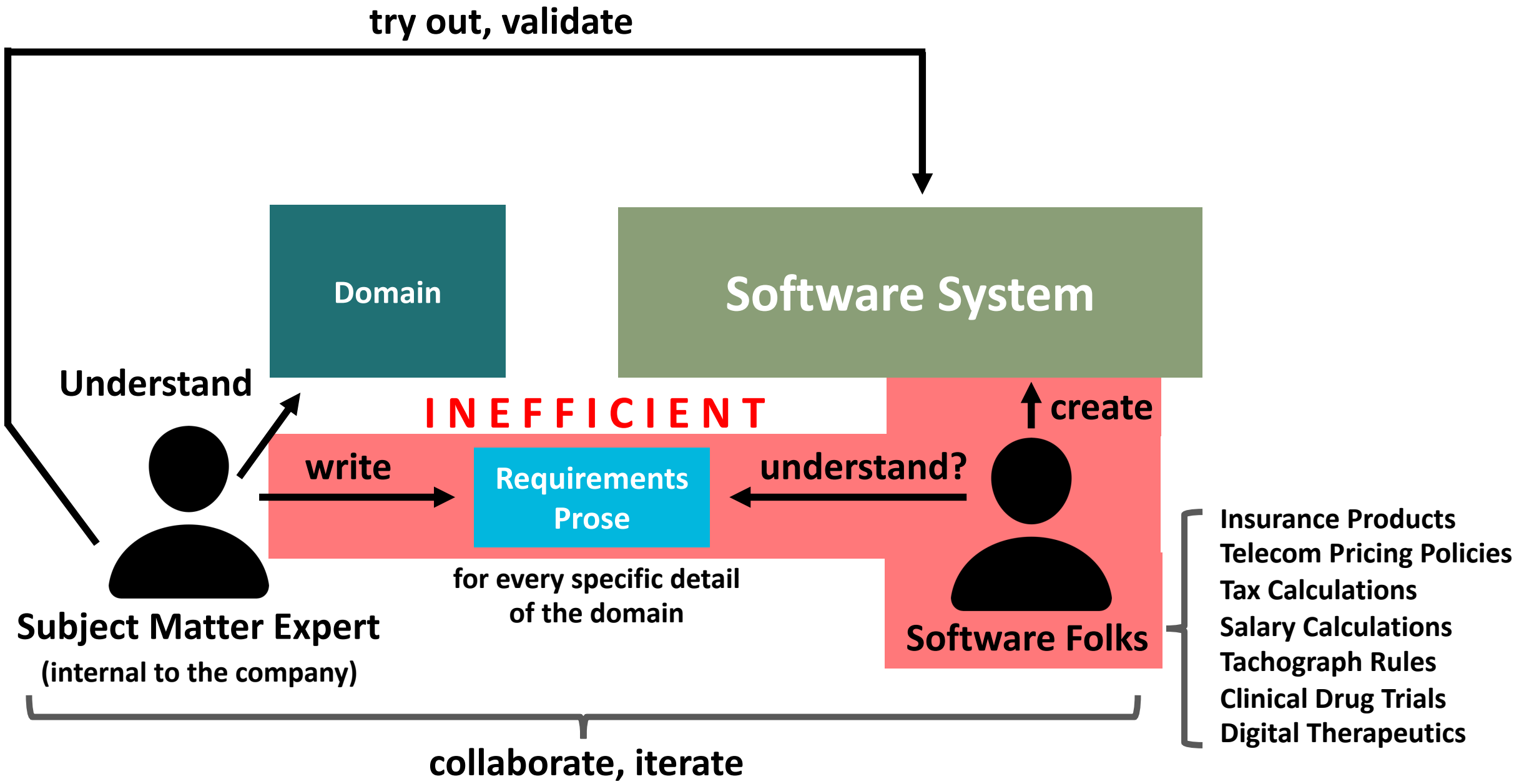

Astronomers send their observation requirements to software devs
DA-63
DV-23
© ALMA


Observing Parameters
BandpassCal
OpticalPointing
RadiometricPointing
Holography
FocusCal
AtmosphericCal
DelayCal
SidebandRatioCal
Science
PhaseCal
PointingCal
AmplitudeCal
PolarizationCal
This ScienceParameters is used by 1 target.
Science Parameters
Science Parameters Name
3c274 Params
Representative Bandwidth
0.48828
MHz
Representative Frequency
230.00000
GHz
Sensitivity Goal
3.00000
mJy
Integration Time on Source
947.66624
s
Sub Scan Duration
6.04800
s
Adjust subscanduration to a correct value:
ADJUST Subscan
Force Atmospheric Calibration
Advanced Parameters
Astronomers send their observation requirements to software devs
NO!
Modern observatories are user-programmable.
© ALMA


feedback
Subject Matter Expert
use
create/modify/
test/run
covers
Domain
Language
Models/
Programs/
Configurations
OPTIMIZE
understand
deeply
Software System
build
abstract
Domain Expert
Software Folks
collaborate, iterate

AND NOW

Something completely different.


try out, validate
Domain
Software System
Understand
INEFFICIENT
create
write
Requirements Prose
understand?
for every specific detail of the domain
Subject Matter Expert
(internal to the company)
Software Folks
Insurance Products
Telecom Pricing Policies
Tax Calculations
Salary Calculations
Tachograph Rules
Clinical Drug Trials
Digital Therapeutics
collaborate, iterate

Example of concrete requirements

Einkommensteuergesetz (EStG)
§ 7b Sonderabschreibung für Mietwohnungsneubau

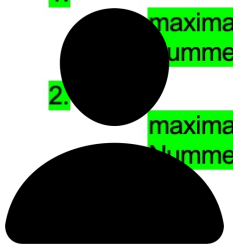
(1) [1]Für die Anschaffung oder Herstellung neuer Wohnungen, die in einem Mitgliedstaat der Europäischen Union belegen sind, können nach Maßgabe der nachfolgenden Absätze im Jahr der Anschaffung oder Herstellung und in den folgenden drei Jahren Sonderabschreibungen bis zu jährlich 5 Prozent der Bemessungsgrundlage neben der Absetzung für Abnutzung nach § 7 Absatz 4 in Anspruch genommen werden. [2]Im Fall der Anschaffung ist eine Wohnung neu, wenn sie bis zum Ende des Jahres der Fertigstellung angeschafft wird. [3]In diesem Fall können die Sonderabschreibungen nach Satz 1 nur vom Anschaffenden in Anspruch genommen werden. [4]Bei der Anwendung des Satzes 1 sind den Mitgliedstaaten der Europäischen Union Staaten gleichgestellt, die auf Grund vertraglicher Verpflichtung Amtshilfe entsprechend dem EU-Amtshilfegesetz in einem Umfang leisten, der für die Überprüfung der Voraussetzungen dieser Vorschrift erforderlich ist.

(2) [1]Die Sonderabs
1.
durch Bau … r dem 1. Januar 2022 oder nach dem … antrags oder einer in diesem Zeitraum … n hergestellt werden, die die Vorausse… erzu gehören auch die zu einer Woh…
2.
Wohnungen, die aufgrund eines nach dem 31. Dezember 2022 und vor dem 1. Januar 2027 gestellten Bauantrag… …ellt werden, in einem Gebäude liegen, da… …e erfüllt und dies durch Qualitätss…

(3) Bemessungsgr… …schaffungs- oder Herstellungskosten der nach Absatz 2 begünstigten Wohnung, jedoch
1.
maximal 2 000 Euro je Quadratmeter Wohnfläche für Wohnungen im Sinne des Absatzes 2 Satz 2 Nummer 1 und
2.
maximal 2 500 Euro je Quadratmeter Wohnfläche für Wohnungen im Sinne des Absatzes 2 Satz 2 Nummer 2.

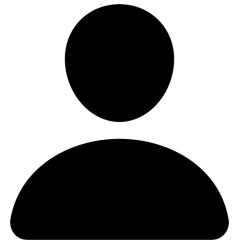
Express this
and all the other laws

Lots of it.
Changes all the time.



Subject Matter Expert

Example of the language needed to express such requirements (and similar ones in the domain):



Domain Expert
Software Folks

Currencies
Dates
Percentages

Arithmetics
Comparisons
Conditionals
(+wa…
Round…
Limiti…
Summa…
Tempo…
Year/M…

Data/Lookup Tables

Versioning (each year things change)

Testing

With this

Less of it.
Much more stable.

WHY DON'T WE LET THE EXPERTS "PROGRAM" SUBJECT MATTER DIRECTLY?

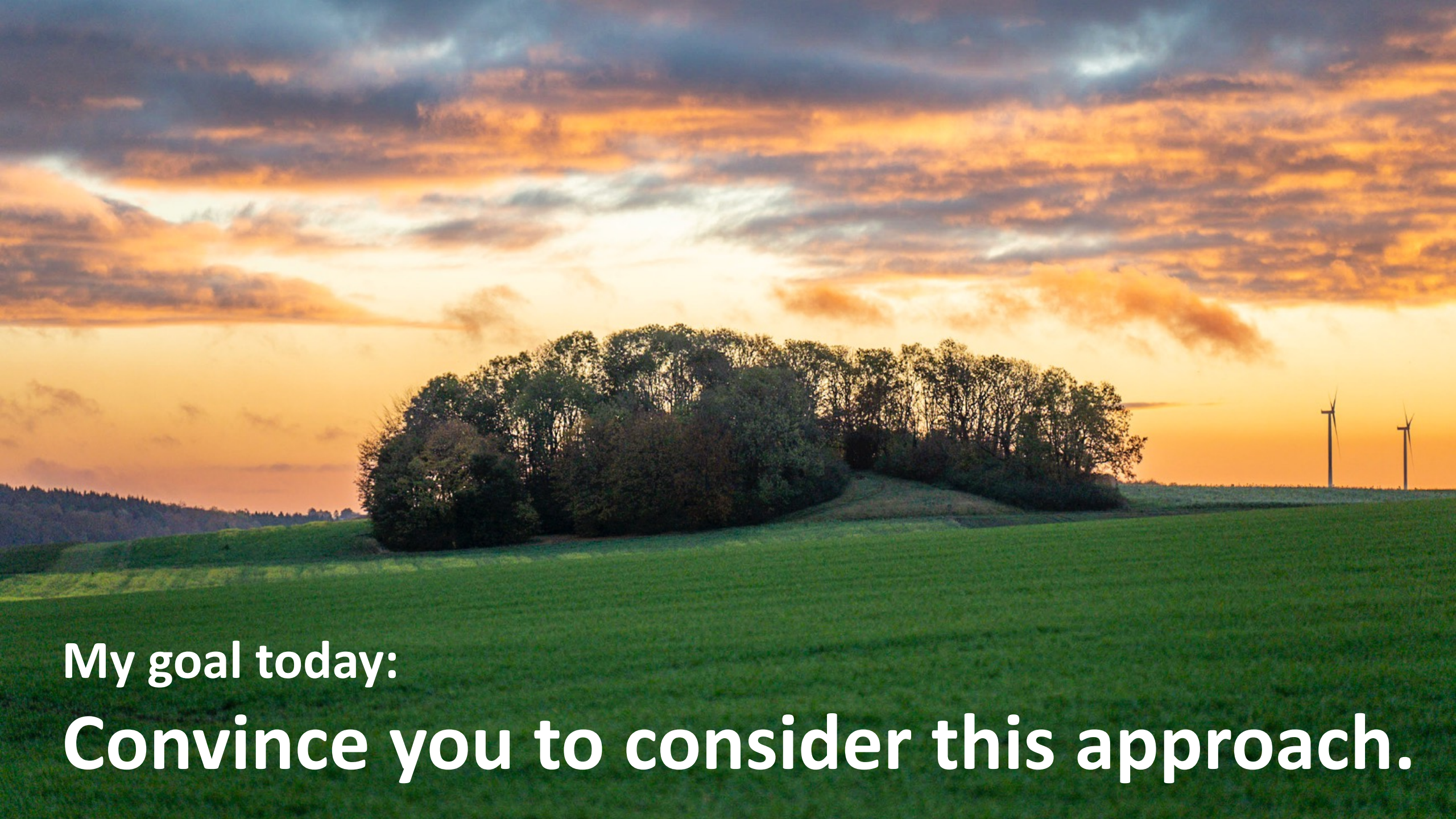
Programmability essential property of system

CNC machines
Radio Telescopes
Game Engines

vs.

Programmability seen as optional / „strange“
SHOULD YOU USE IT?

Insurance Products
Telecom Pricing Policies
Tax Calculations
Salary Calculations
Tachograph Rules
Clinical Drug Trials
Digital Therapeutics


My goal today:
Convince you to consider this approach.

WHEN SHOULD YOU?

There is sufficiently **complex** and **large** subject matter in the domain.

There's a role **subject matter expert** distinct from software developers.

There is sufficient **variability** or **change over time** within the domain.

There are **lots of instances** in the domain. { Different specific things that can be expressed in a similar way.

The domain as such is **long-lived.**

And the organisation that develops the language plans to work in that domain for a reasonable time.

Insurance Products
Telecom Pricing Policies
Tax Calculations
Salary Calculations
Tachograph Rules
Clinical Drug Trials
Digital Therapeutics

Why SHOULD YOU?

Subject Matter Experts are **empowered** – no longer 2nd class “behind” devs.

Devs can **focus** on technical concerns, don’t have to understand ”everything”.

Subject matter is **portable**, the legacy problem is much reduced.

Collaboration between SMEs and devs better because for “formal contract”.

The overall (subject matter) development process becomes **faster and more agile.**

Tradeoffs

Understand the domain (should do this anyway)
Build and evolve **language & tooling** → **WHEN**
Training + culture shift for SMEs and devs

Fachkräftemangel
Competitiveness

we have to
change somehow
to adress these!

Subject Matter First

SIGNATURES UPDATE LINK

Manifesto

We advocate for putting SMEs directly in control of “their” part of the software and letting developers focus on software engineering.

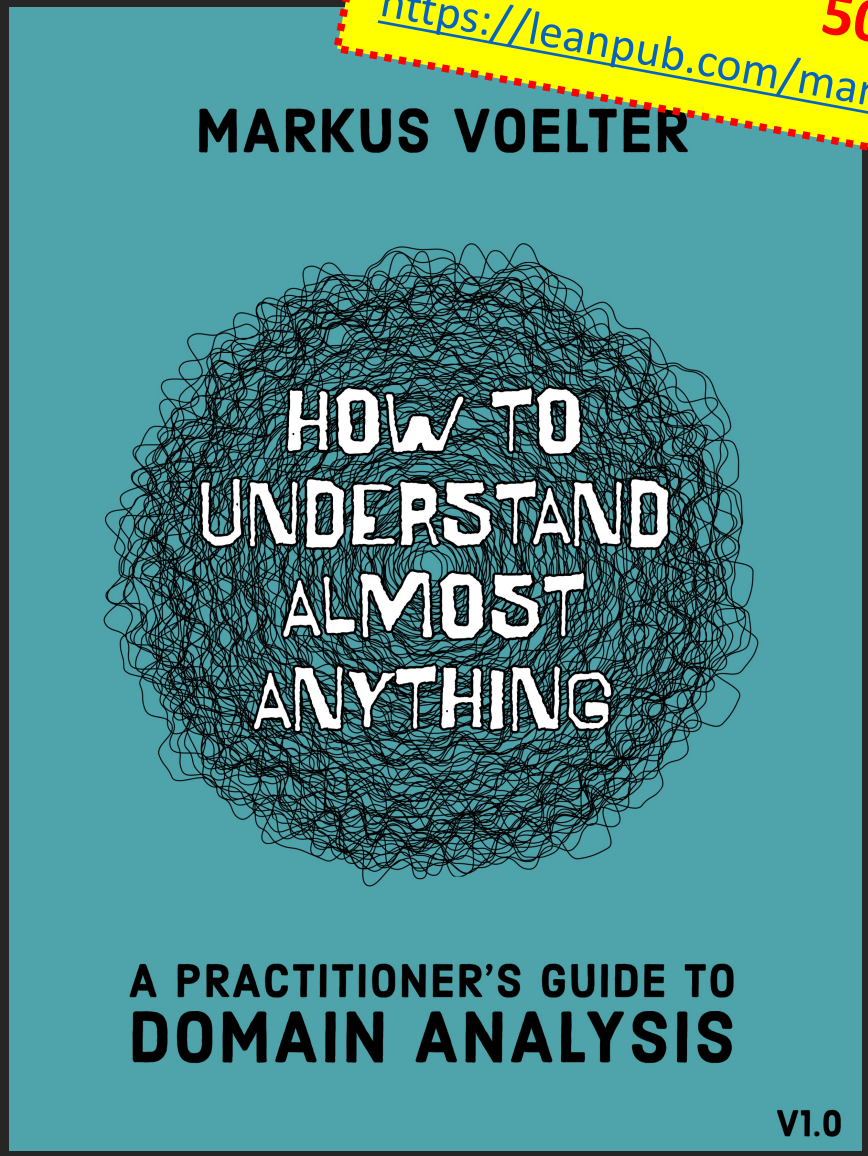
JOIN US

https://subjectmatterfirst.org


HOW
to know what goes into the language?
to do this, technically?

HOW do you know what goes into the language?

I wrote a whole book about that ☺



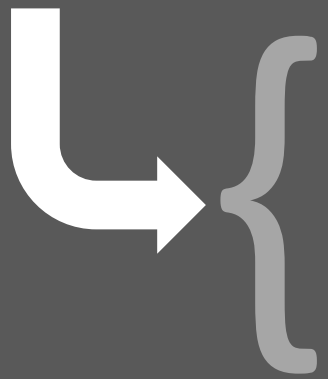
50% off
https://leanpub.com/markusvoelter-htuaa/c/wjax2023

http://voelter.de/htuaa

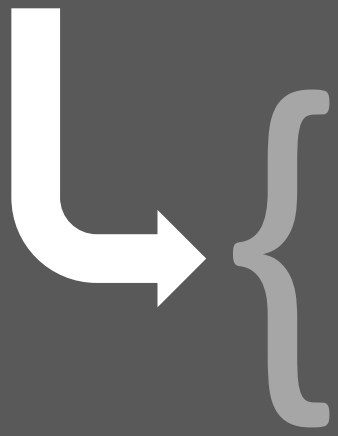
HOW do you know what goes into the language?

Domain analysis is about understanding the structures, rules, constraints and behaviors of a domain precisely and completely enough to **make it accessible to** people and **software tools.**

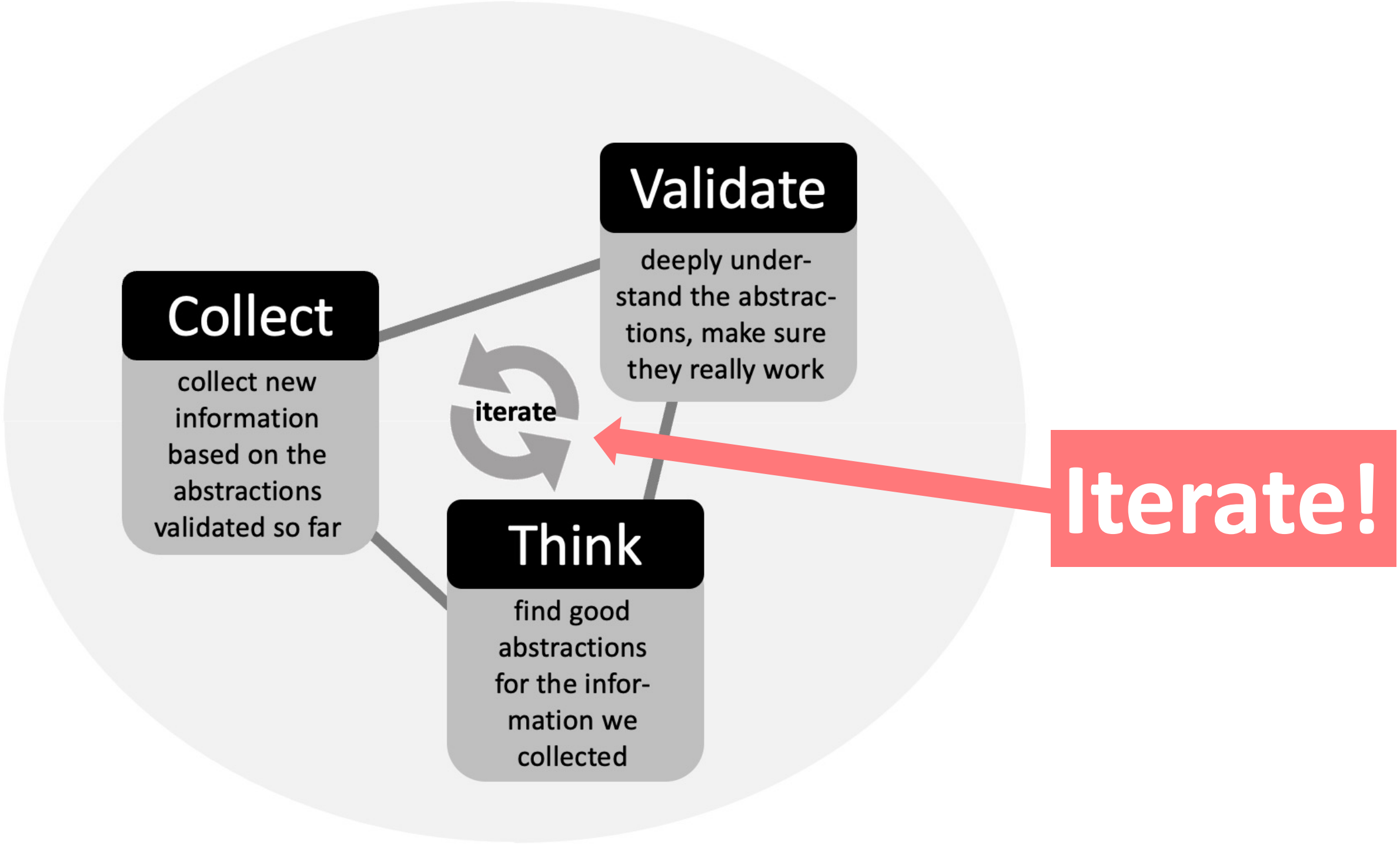
make it accessible to software tools.



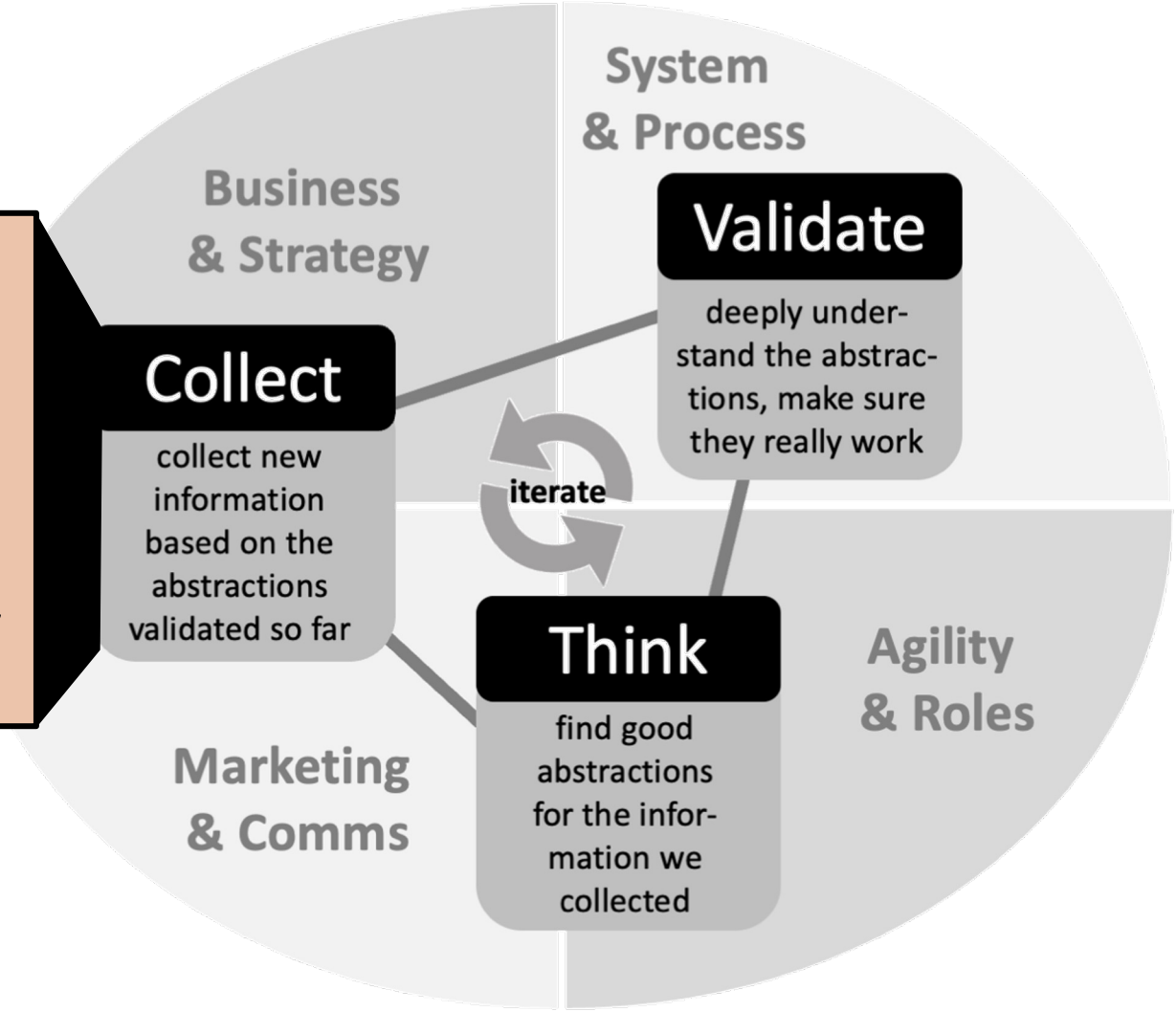
- data structures
- constraints & rules
- semantics
- models, languages

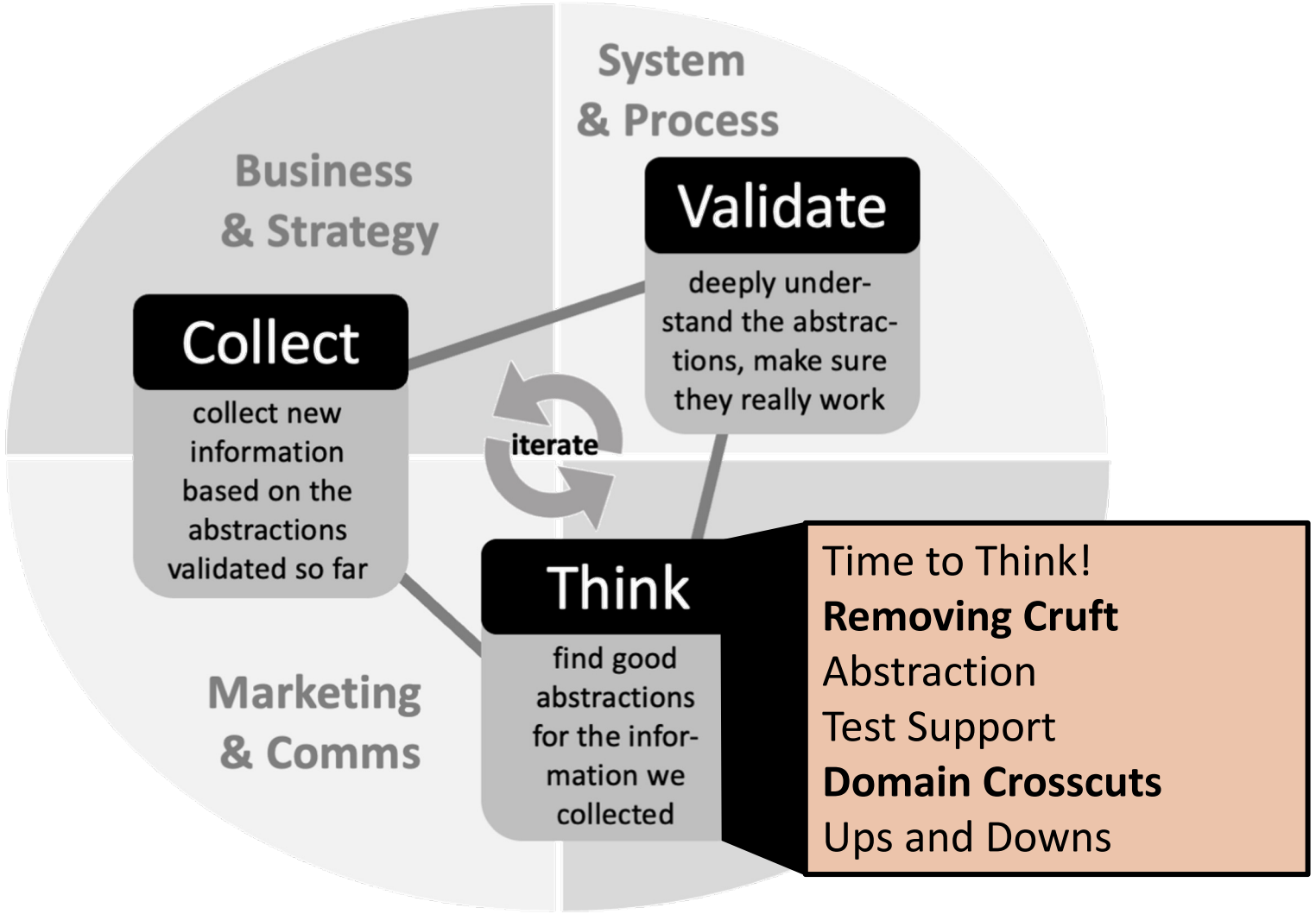


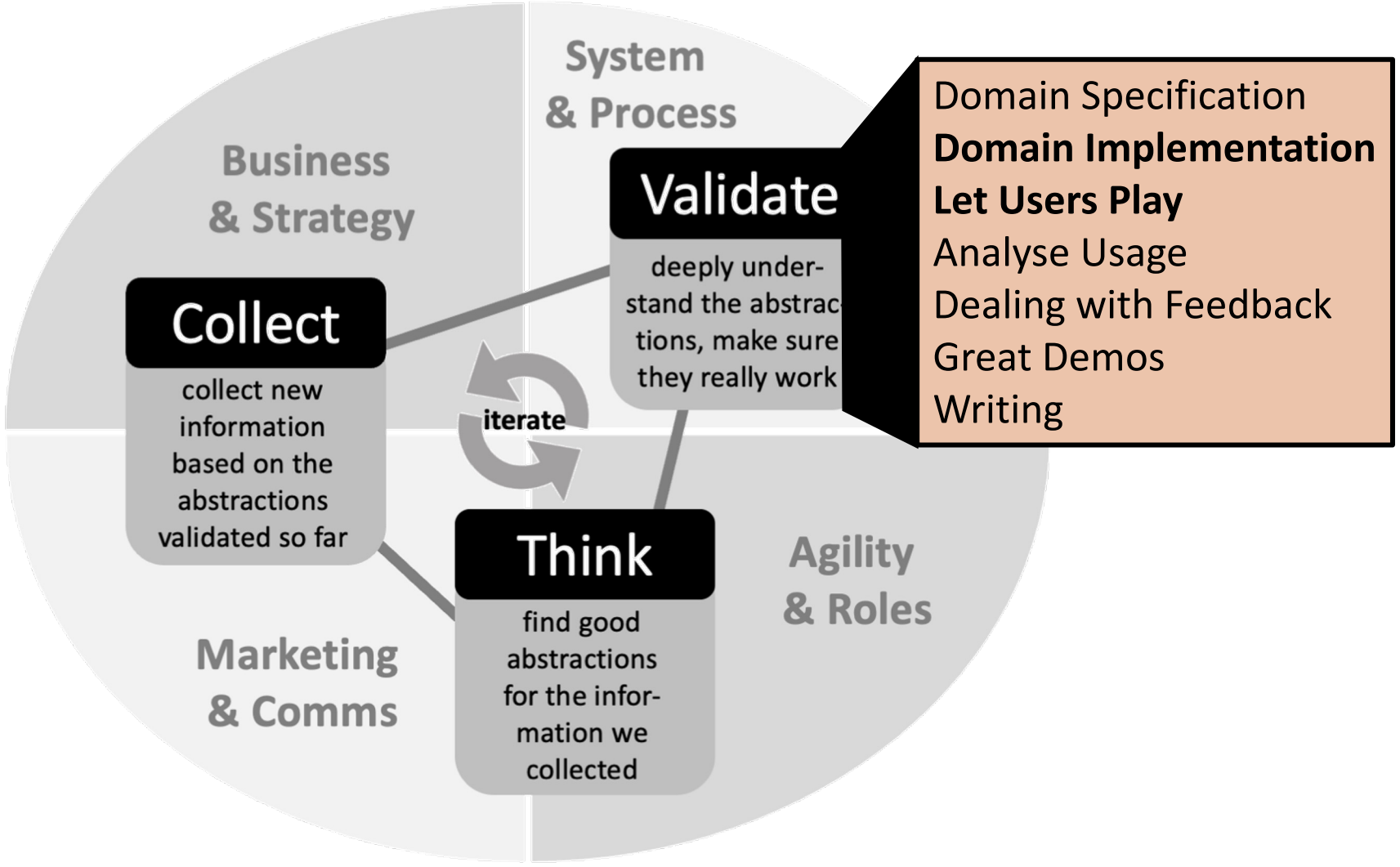
- APIs
- Software Structures
- Checkers, Compilers & Interpreters
- UIs & Simulators
- Transport Protocols


Validate
deeply under-
stand the abstrac-
tions, make sure
they really work
Collect
collect new
information
based on the
abstractions
validated so far
iterate
Think
find good
abstractions
for the infor-
mation we
collected
Iterate!

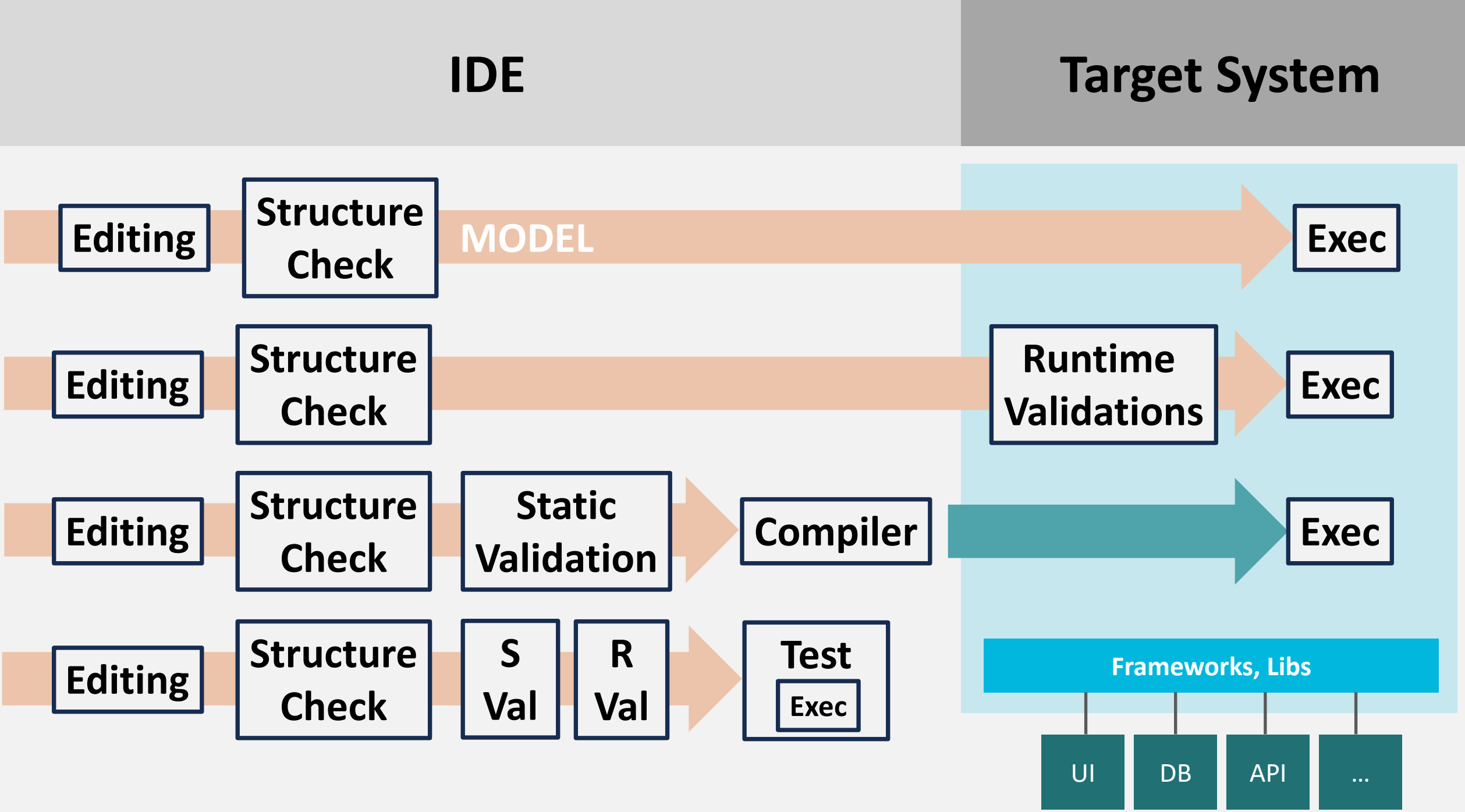
Written Material
Hidden Languages
Consistent Terminology
Working Sessions
Active Listening
Consistency vs. Change
Dealing with Uncertainty
Capture Results

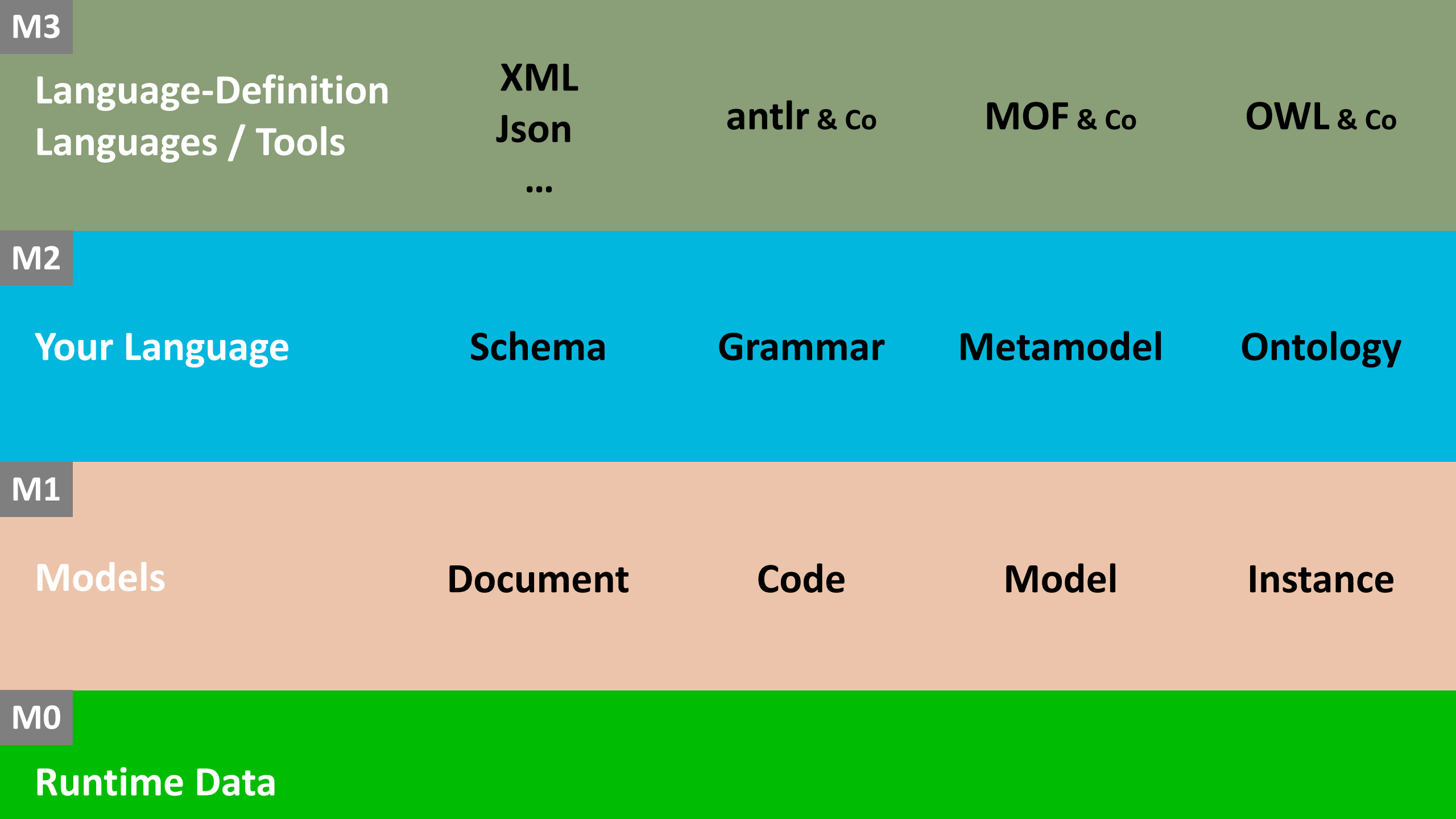



System
& Process
Business
& Strategy
Validate
deeply under-
stand the abstrac-
tions, make sure
they really work
Collect
collect new
information
based on the
abstractions
validated so far
iterate
Think
find good
abstractions
for the infor-
mation we
collected
Marketing
& Comms
Time to Think!
Removing Cruft
Abstraction
Test Support
Domain Crosscuts
Ups and Downs


System
& Process
Business
& Strategy
Validate
deeply under-
stand the abstrac-
tions, make sure
they really work
Collect
collect new
information
based on the
abstractions
validated so far
iterate
Think
find good
abstractions
for the infor-
mation we
collected
Agility
& Roles
Marketing
& Comms
Domain Specification
Domain Implementation
Let Users Play
Analyse Usage
Dealing with Feedback
Great Demos
Writing


HOW
to know what goes into the language?
to do this, technically?


IDE
Target System
Editing
Structure Check
MODEL
Exec
Editing
Structure Check
Runtime Validations
Exec
Editing
Structure Check
Static Validation
Compiler
Exec
Editing
Structure Check
S Val
R Val
Test
Exec
Frameworks, Libs
UI
DB
API
...


M3
Language-Definition Languages / Tools
XML
Json
...
antlr & Co
MOF & Co
OWL & Co
M2
Your Language
Schema
Grammar
Metamodel
Ontology
M1
Models
Document
Code
Model
Instance
M0
Runtime Data

M3

Language-Definition
Languages / Tools

M2

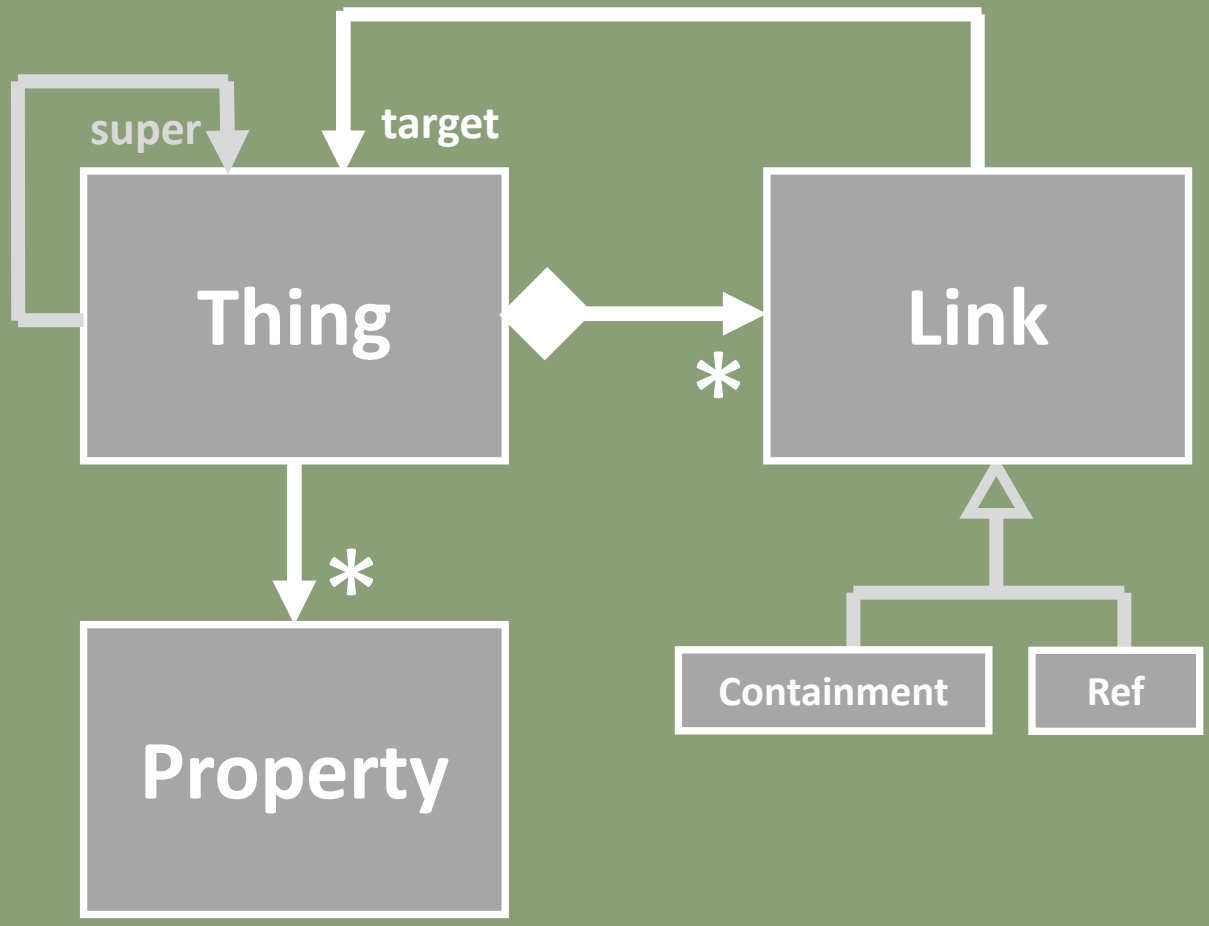
Your Language

M1

Models

M0

Runtime Data



M3

Language-Definition
Languages / Tools

M2

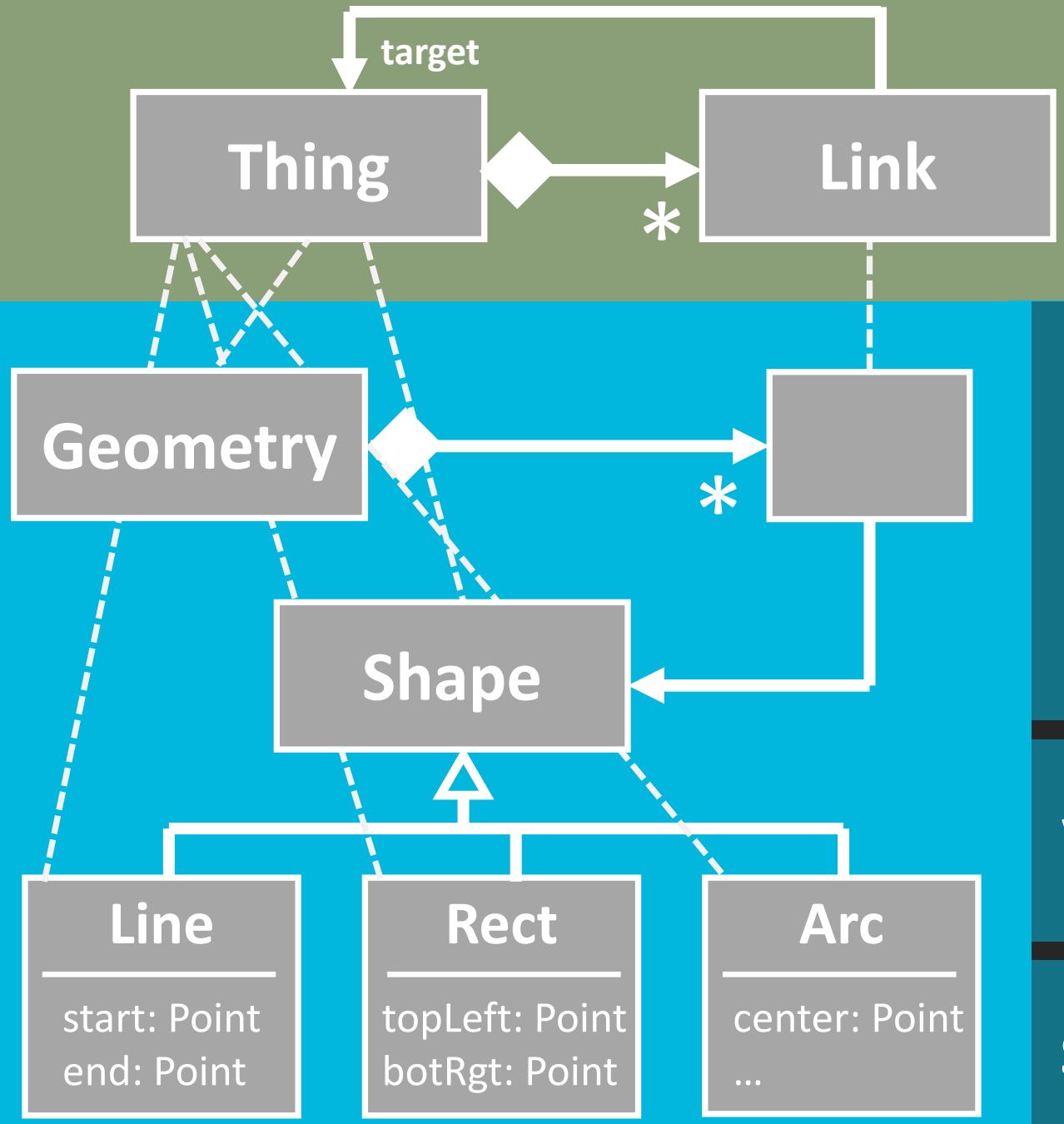
Your Language

M1

Models

M0

Runtime Data

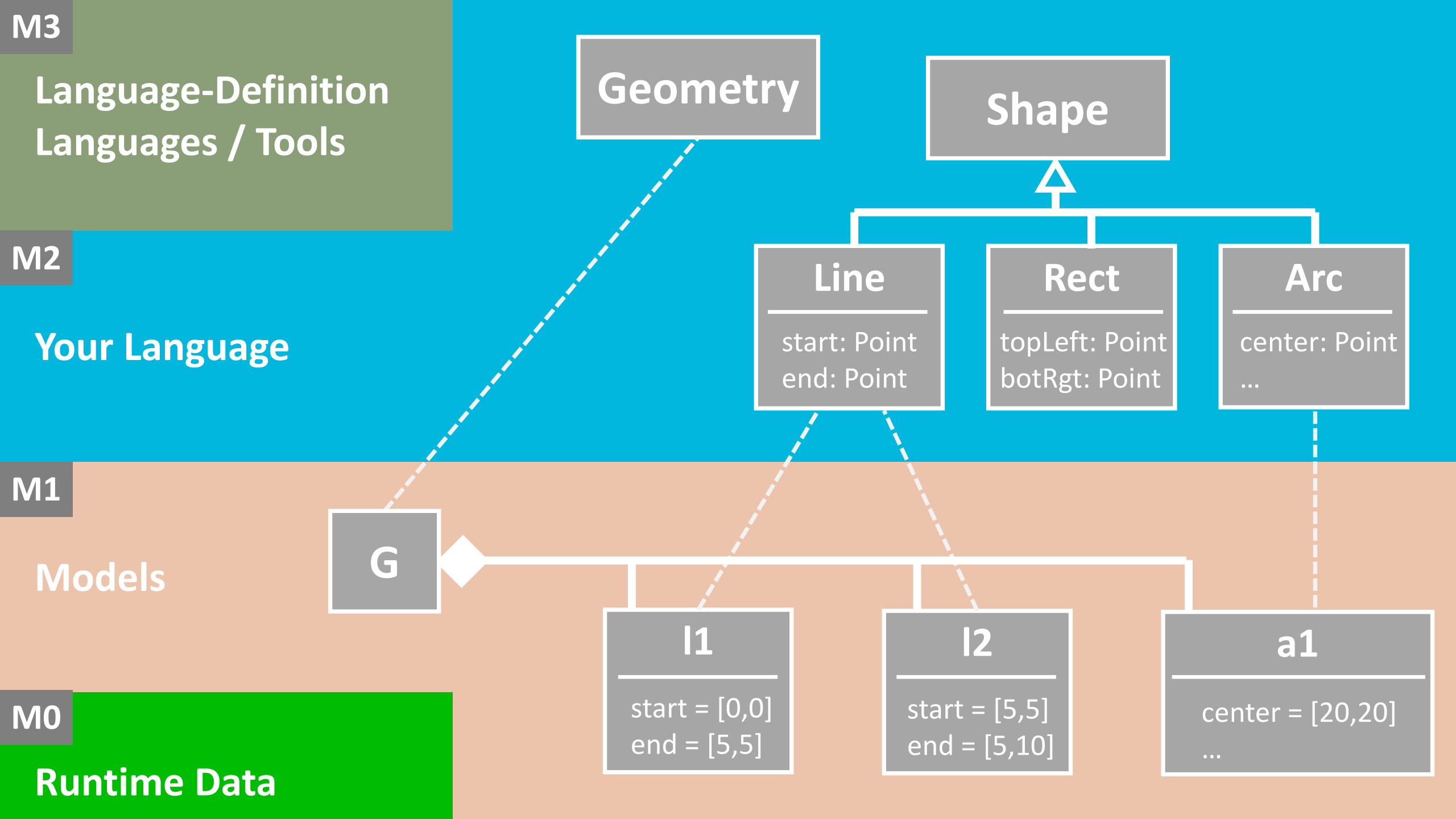


Syntax

Text
Tables
Diagrams
GUIs

Validation

Semantics


M3
Language-Definition
Languages / Tools
M2
Your Language
M1
Models
M0
Runtime Data
Geometry
Shape
Line
start: Point
end: Point
Rect
topLeft: Point
botRgt: Point
Arc
center: Point
…
G
l1
start = [0,0]
end = [5,5]
l2
start = [5,5]
end = [5,10]
a1
center = [20,20]
…

TOOLS

Language-Definition Languages / Tools	XML Json …	antlr & Co	MOF & Co	OWL & Co
Your Language	Schema	Grammar	Metamodel	Ontology
Models	Document	Code	Model	Instance
Runtime Data				

MPS Meta Programming System

Create your own domain-specific language

DOWNLOAD WATCH VIDEO

WHY MPS?

Monaco - The Editor of the Web

The Monaco Editor is the code editor that powers VS Code. A good page describing the code editor's features is here. It is licensed under the MIT License and supports Edge, Chrome, Firefox, Safari and Opera. The Monaco editor is not supported in mobile browsers or mobile web frameworks. Find more information at the Monaco Editor repo.

CodeMirror Examples Documentation Try Discuss GitHub Version 5

Extensible Code Editor

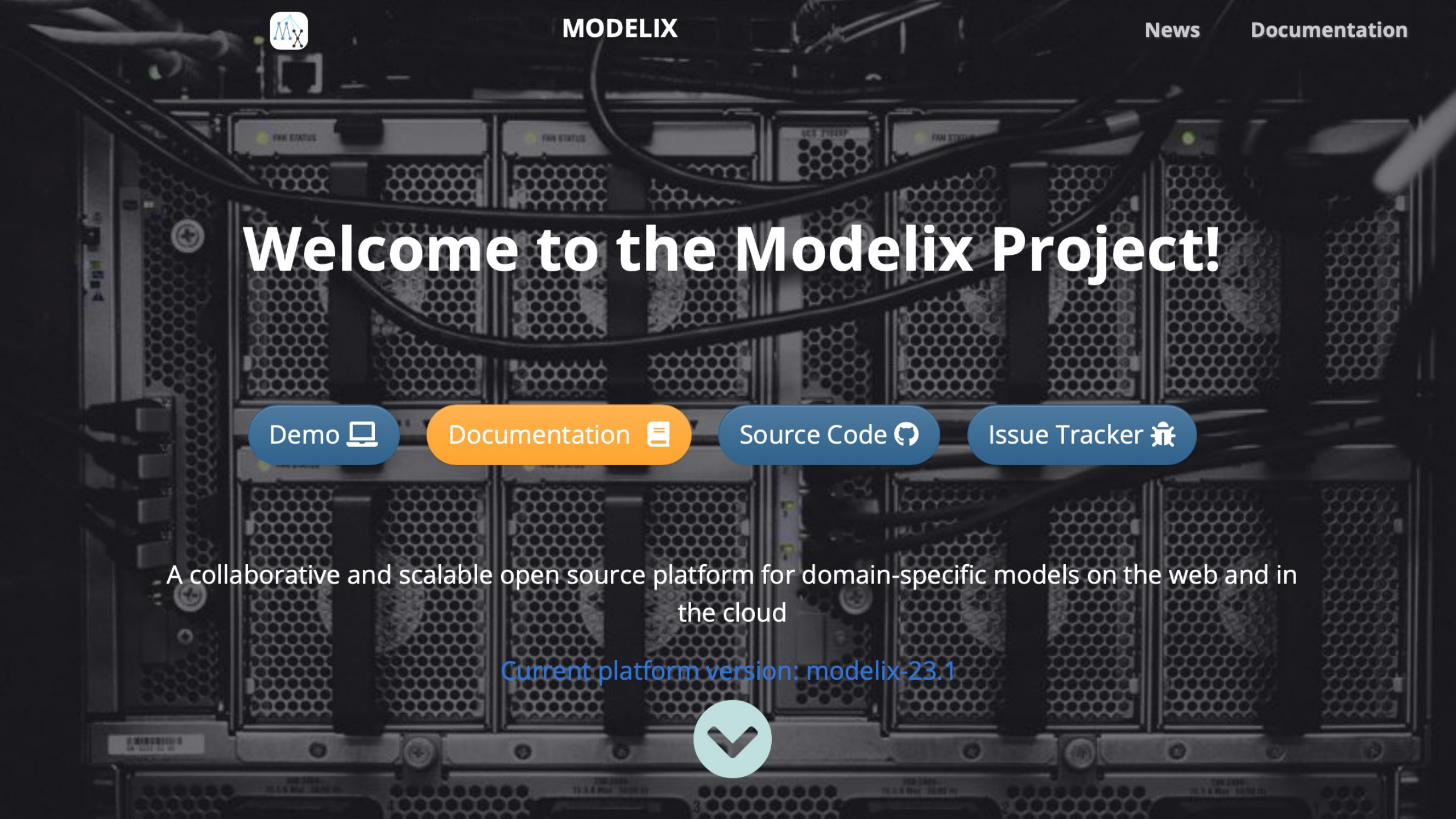
CodeMirror is a code editor component for the web. It can be used in websites to implement a text input field with support for many editing features, and has a rich programming interface to allow further extension.

Built to bring language engineering to the next level_

Langium is an open source language engineering tool with first-class support for the Language Server Protocol, written in TypeScript and running in Node.js.

This future-proof technology stack enables domain-specific languages in VS Code, Eclipse Theia, web applications, and more.

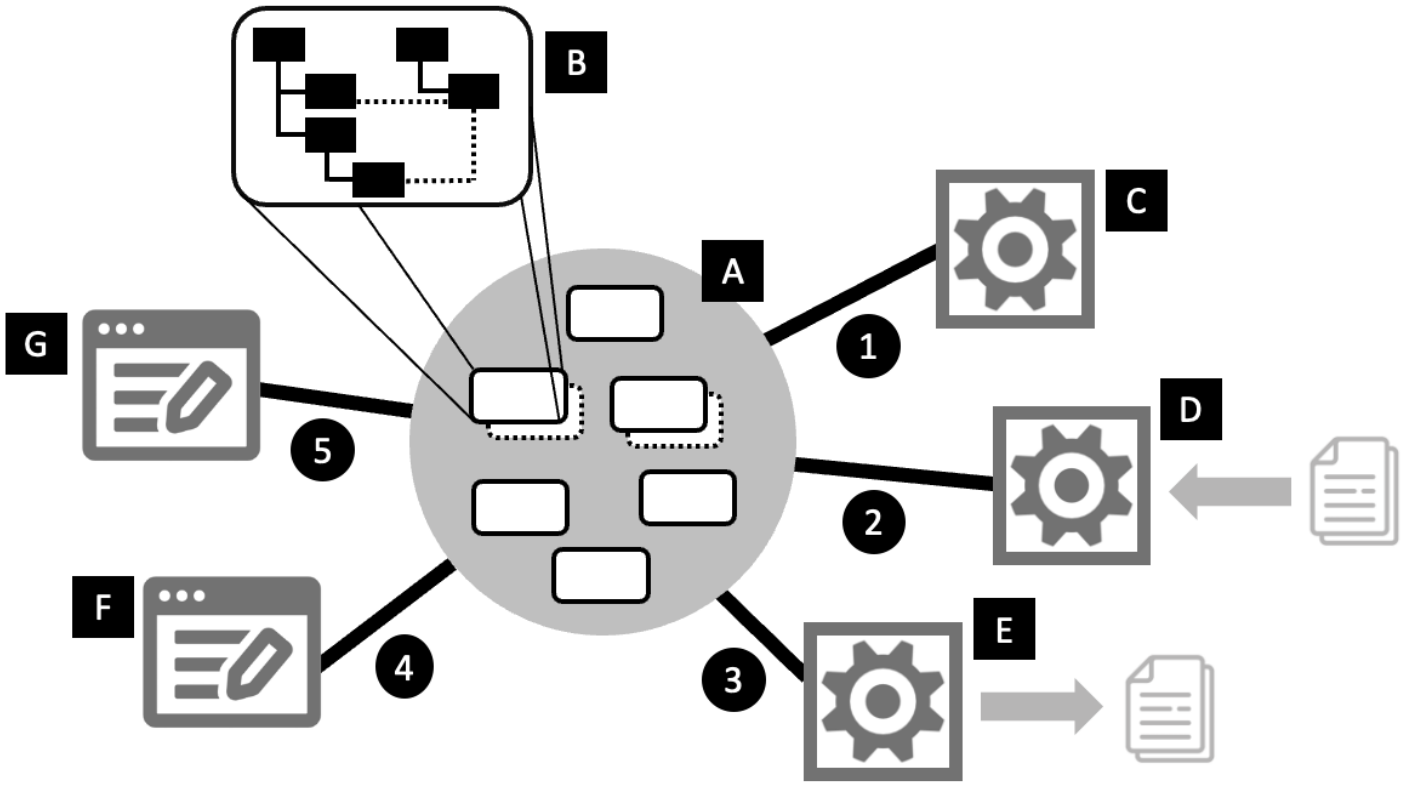
Try it! Getting Started


MODELIX
News
Documentation
Welcome to the Modelix Project!
Demo
Documentation
Source Code
Issue Tracker
A collaborative and scalable open source platform for domain-specific models on the web and in the cloud
Current platform version: modelix-23.1

LionWeb Language Interfaces on the Web

Follow

Overview | Repositories 7 | Projects 1 | Packages | Teams 1 | People 9 | Settings



The LIonWeb initiative aims to facilitate the community-based development of language engineering and modeling tools on the web.

1. Protocols for communication between participating software components
2. Meta-meta model as well as a reference architecture
3. APIs to access models and metamodels and to encapsulate the protocols
4. Hub for the developers of such components and to empower other software developers to develop web-based modeling solutions.

http://lionweb.io

Roll your own – what do you need?

A robust M3

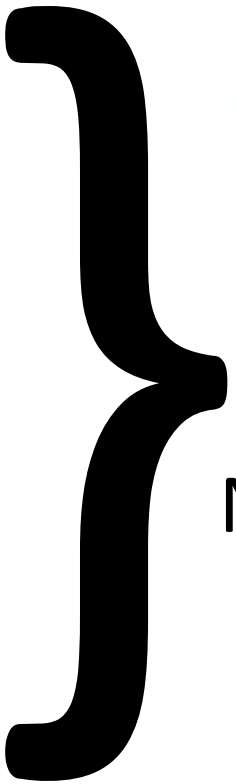
- A formalism to define meta models
- Data structures to represent models in memory,
- An API to read, traverse and modify models,
- Formats to persist them somehow using a metamodel-specific serialization format (not a syntax),
- A rudimentary but generic way of editing them.

Model API

```
modelelement.getChildren(role: string)

modelelement.getReferences(role: string)

modelelement.getPropertyValue(propName: string)
```

Meta Model Def

```
Machine.getStates() : list<State> {
 (list<State>)this.getChildren("states");
}

State.getName() : string {
  (string)this.getProperty("name")
}

Transition.getTarget() : State {
  (State)this.getReference("target").deref()
}
```



Store
Query
Notification

Find more details at:
https://medium.com/@markusvoelter/the-minimum-infrastructure-for-running-languages-and-models-da922aa3b4b4

Closing Thoughts

How is this related?

Domain-Driven DESIGN

Definition (Wikipedia)

- place the primary focus on the core domain and domain logic;
- base complex designs on a model of the domain (UL);
- initiate a creative collaboration between technical and domain experts to iteratively refine a conceptual model that addresses particular domain problems.

PLUS (me)

- reify the conceptual model into a DSL that allows the domain experts to directly express subject matter in an executable and testable way.

More Wikipedia: Critics of DDD argue that developers must typically implement a great deal of isolation and encapsulation to maintain the model as a pure and helpful construct.

Working with DSLs is a bit like DDD++ and I am surprised not more DDDers care.

Critics?

I think this isolation is a massive benefit.

Why would you WANT to do that?

Subject Matter Experts are empowered – no longer 2nd class "behind" devs.

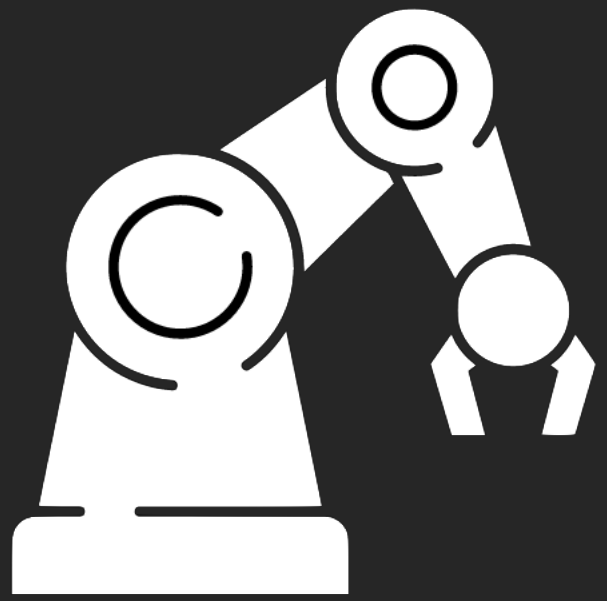
Devs can focus on technical concerns, don't have to unde

Subject matter is portable, the legacy problem is much re

Collaboration between SMEs and devs better because fo

Done right, the overall subject matter development proc

https://www.linkedin.com/pulse/relationship-between-domain-driven-design-languages-markus-voelter



Automation & Tools

Marek Ślusarczyk

You distinguish commonalities from variability and options.

You build platforms and automate the variability.

You tool up.

You hire toolsmiths.

You invest in tool-making machines.

You understand production logistics.

We do this in the space of technology as well as build, package, test, deploy.

But not for subject matter.

THAT‘S A MISTAKE!

Marek Ślusarczyk

One last thought on

Subject Matter

Subject Matter

Technical Stuff

Is this really a qualitative difference compared to well-structured “normal” code?

What do you do if you want to change the modeling technology?

Yes! You know the semantics of everything! And there is no technical stuff mixed into subject matter. You can always transform it into whatever other form you need.

Significantly reduces the legacy problem.

Subject Matter

Technical Stuff

Yes! You know the semantics of everything! And there is no technical stuff mixed into subject matter. You can always transform it into whatever other form you need.

Subject Matter

Technical Stuff

Yes! You know the semantics of everything! And there is no technical stuff mixed into subject matter. You can always transform it into whatever other form you need. Or from.

And no, we cannot replace the whole thing with AI!

https://www.infoq.com/articles/ai-based-prose-programming-for-subject-matter-experts/

Insurance Products
Telecom Pricing Policies
Tax Calculations
Salary Calculations
Tachograph Rules
Clinical Drug Trials
Digital Therapeutics

Large-Scale Printer Architecture
E/E architecture for Cars
Municipality Management Systems
Public Benefits Rules Spec

… many more …

All have been implemented this way!

“Domain-Specific Languages
MD(S){D|E}
(Meta)Data-driven Systems
Configurable Systems
Platforms”

TAKEAWAYS

TAKEAWAYS

Don’t be put off by “language” and “meta” and such.

Things have come a long way since your compiler course at university.

Be on the lookout for domains where SME “programming” could be a good fit. The approach can be a significant boost.

Familiarize yourself with some of the tools in the space of language development and code generation. These are a useful ingredient for a developer’s toolbox.

Read the HTUAA book ☺ it‘s useful even if you don‘t build a language and „just“ want to understand a domain for „regular“ development.

And: this stuff is really fun and satisfying.