

yields

# Best practices for Model Governance & Documentation

Jos Gheerardyn  
September 28, 2023



# Agenda

- Introduction
- Why documentation matters
- Challenges
- Architecture of documentation
- Technology
- Use cases



# Yields for Governance

## Model Catalog

- Capture all model attributes and rich metadata, such as model limitations and assumptions, using the flexible model catalog.
- Keep track of all model events in a structured way, ensuring periodic evaluation of model performance and accuracy.

## Workflow management

- Streamline the execution of end-to-end MRM processes and ensure a smooth, efficient workflow with the help of the workflow engine.
- Enhance collaboration among the lines of defense by sending in-app notifications and templated emails to users during process executions.

## Automated Documentation

- Automatically retrieve qualitative and quantitative content during the documentation generation steps using your own documentation templates.

## Dashboards

- Query and report intuitively using a data model purpose-built for model risk management.
- Extend internal dashboards by leveraging BI tools, either deployed by us or by utilizing your existing tools.



# Introduction



Model inventories expand  
at ~20%/y

SS1/23 – Model risk  
management principles for  
banks

Supervisory statement 1/23



Regulatory  
pressure

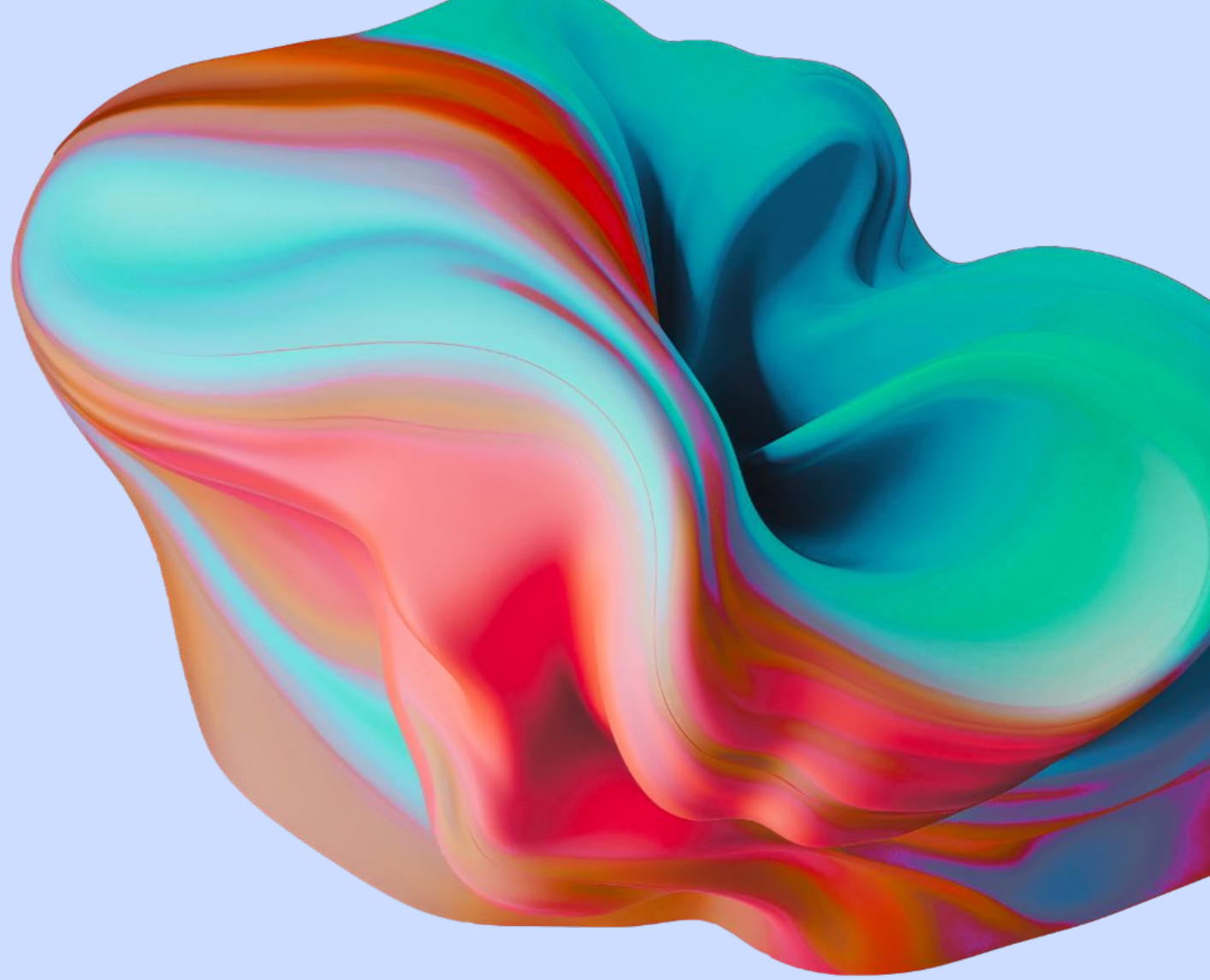


Documentation  
practices



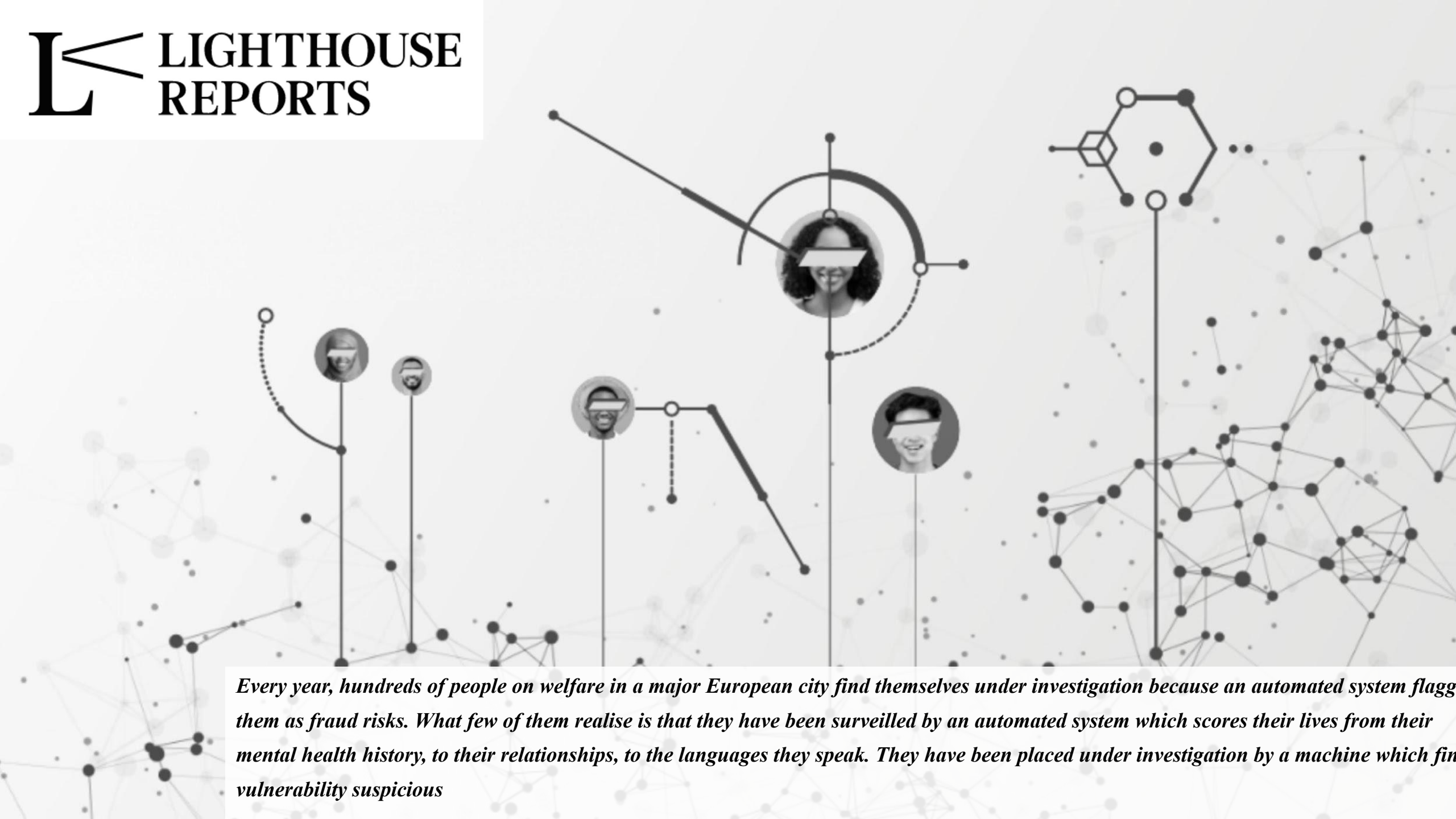
Flexibility

# Why documentation matters





# L LIGHTHOUSE REPORTS



*Every year, hundreds of people on welfare in a major European city find themselves under investigation because an automated system flags them as fraud risks. What few of them realise is that they have been surveilled by an automated system which scores their lives from their mental health history, to their relationships, to the languages they speak. They have been placed under investigation by a machine which finds vulnerability suspicious*

# Documentation fuels model usage

*“Without documentation, we lack the means to understand the model’s systematic impact prior to deployment.”\**

Stakeholder	Documentation need
<b>Model users</b>	Understand intended use cases & track performance
<b>Model developers</b>	Model selection & maintenance
<b>Software developers</b>	Inform design & implementation
<b>Policy makers</b>	Understand how models may fail & how they impact people
<b>Organizations</b>	Evaluate model risk appetite
<b>Quants &amp; data scientists</b>	Finetune the model
<b>Impacted people</b>	Understand how the model work & pursue remedies

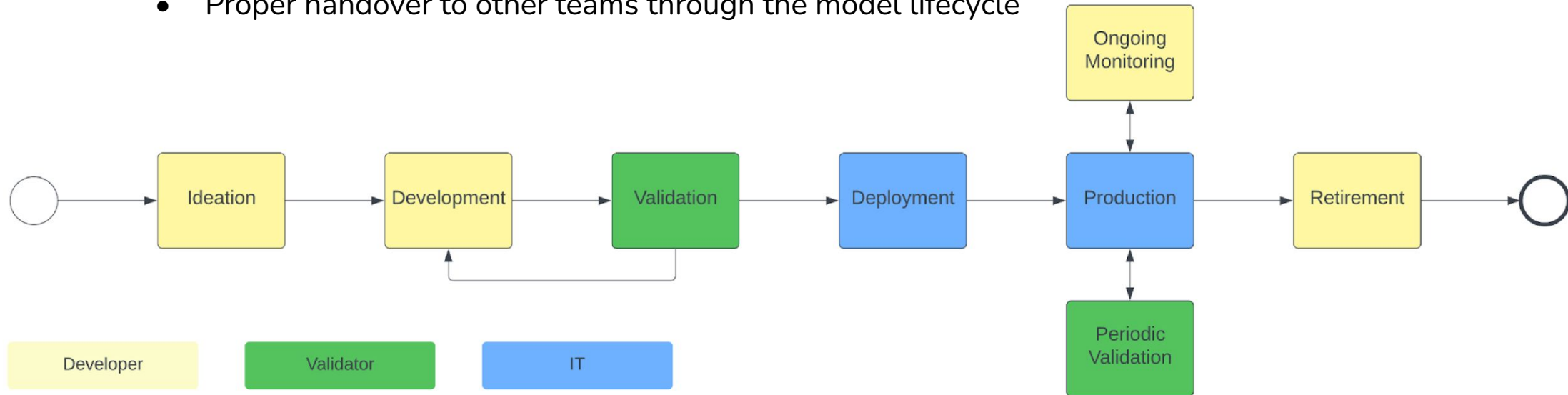
\* see <https://arxiv.org/abs/1810.03993>



# Documentation as scalable communication

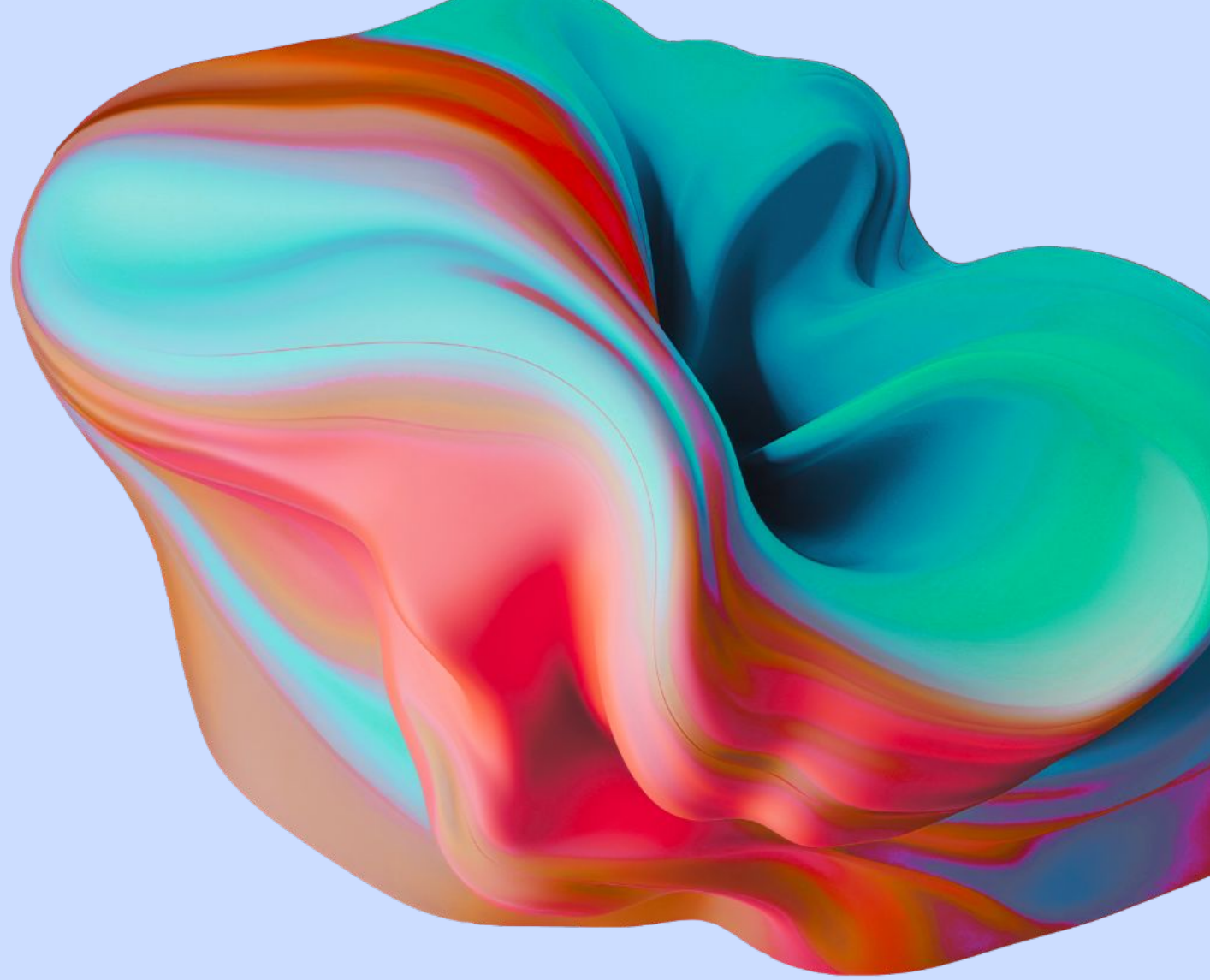
Knowledge sharing is critical during the model lifecycle

- Teams change over time
- Building up expertise - why does(n't) something work
- Proper handover to other teams through the model lifecycle





# Challenges with documentation

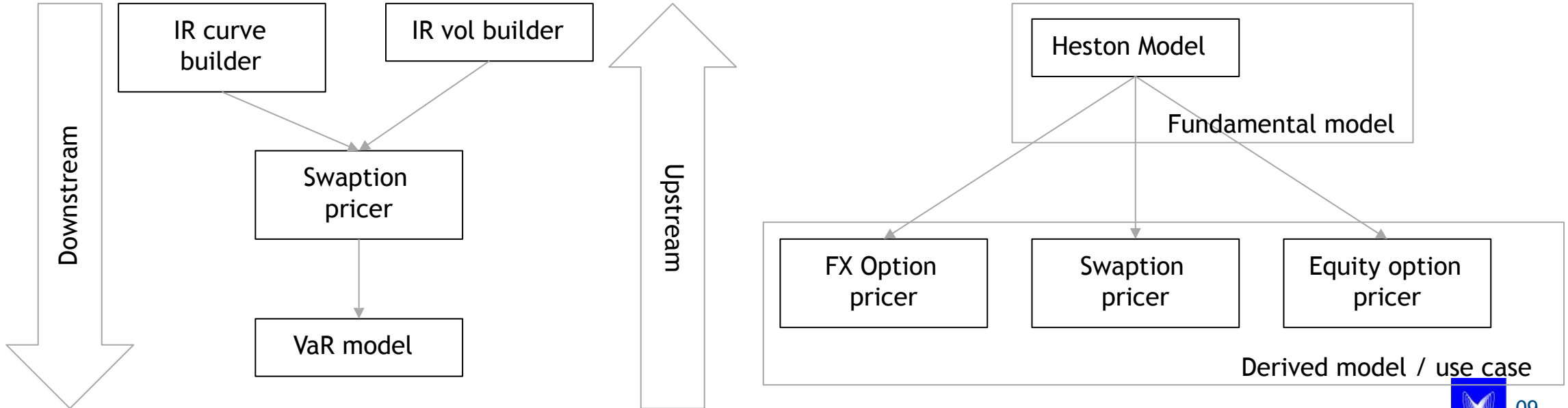


# Inconsistent documentation

We often have to generate multiple documents per model

- Per **geography** (e.g. SR 11/7 vs SS1/23)
- Per **stakeholder** type (E.g. credit officer vs internal model review team)
- Driven by **model dependencies**

**Note:** Model Dependency types



# Stale documentation

Documentation is often outdated when content is **not captured dynamically**.

Document refresh should be triggered by e.g.

- Periodic monitoring
- Recurring validation
- model version upgrades
- dependent models changes
- evolving regulatory requirements
- evolving templates

Typical time required to uplift a model document: **1 - 3 months**

Typical frequency per update: **4x / y**

Equity Vanilla Options - Binomial Tree Confidential – Version 1.1

**3. Model Dependencies**

These are the model dependencies:

- CVA/DVA model, xVA, Downstream
- **Volatility Surface Construction model, Pricing, Upstream**

*The upstream model dependency reported is a retired model that was replaced by the NEW volatility model*



# Varying quality

Quality of documentation as well as level of detail is **often varying all too much**.

- Only partially mitigated by using a template
- Especially challenging when outsourcing (e.g., vendor model, offshore validation)
- Will depend on the mother tongue, time available, ...

*Validator 1 from the Equity Pricing Model Validation team*

Equity Vanilla Options - Binomial Tree Confidential – Version 1.1

**1. Executive Summary**

**1.1 Fit-for-purpose Assessment**

Accountability	Overall Assessment
MRM	Approved

**1.2 Evidence of Effective Challenge**

Guidance: please **summarize key elements** to support your model validation assessment conclusions

**Test results provided by developers indicate that the model behaves as expected.**

*Validator 2 from the Equity Pricing Model Validation team*

Equity Vanilla Options - Binomial Tree Confidential – Version 1.1

**1. Executive Summary**

**1.1 Fit-for-purpose Assessment**

Accountability	Overall Assessment
MRM	Approved

**1.2 Evidence of Effective Challenge**

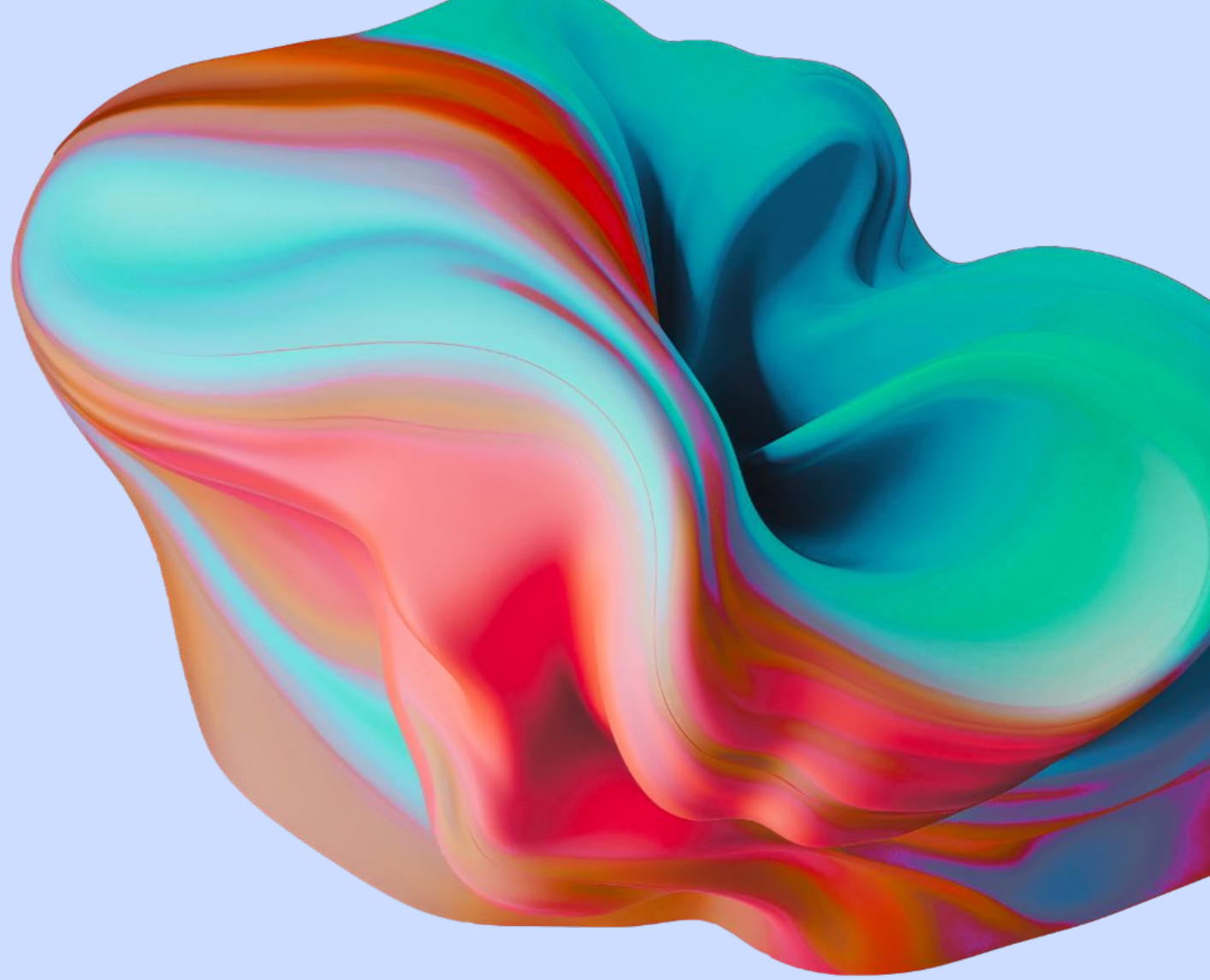
Guidance: please **summarize key elements** to support your model validation assessment conclusions

**Test results provided by developers indicate that the model behaves as expected. This is supported by the following main aspects (please see dedicated sections for more detailed effective challenge analysis):**

- The binomial tree approximation price convergences to the analytical price for a wide set of parameters values which are observed in normal market conditions
- The default number of steps proposed by the developers delivers the required price accuracy within the required tolerance
- The binomial tree model output is robust to both small and large input parameters perturbation and it is also suitable to be used for stress testing purposes.



# Structural properties of documentation







# Software citation principles\*

**Importance:** Software should be considered a legitimate and citable product of research.

**Credit and Attribution:** Software citations should facilitate giving scholarly credit and normative, legal attribution.

**Unique Identification:** A software citation should include a method for identification that is machine actionable, globally unique, interoperable, and recognized by the community

**Persistence:** Unique identifiers and metadata describing the software and its disposition should persist

**Accessibility:** Software citations should facilitate access to the software itself and to its associated metadata, documentation, data, and other materials necessary for both humans and machines to make informed use of the referenced software.

**Specificity:** Software citations should facilitate identification of, and access to, the specific version of software that was used. Software identification should be as specific as necessary, such as using version numbers, revision numbers, or variants such as platforms.

\* see <https://force11.org/info/software-citation-principles-published-2016/>



# Model Card Example

A model card is a concise document used for description of AI models.\*

## Model Details

- stakeholders
- version
- references & methodology
- model tier

## Metrics

- Performance
- Thresholds
- Uncertainty & variability

## Ethical considerations

- Data
- Risks & Harms
- Mitigations
- Use cases of attention

## Intended use

- Primary intended use
- Primary intended users
- Out-of-scope use cases

## Evaluation (& training) data

- Which datasets
- Motivation
- Preprocessing

## Findings

- Caveats
- Recommendations

## Factors (relevant & evaluation)

- Groups: Unitary & complex
- Instrumentation
- Environment

## Quantitative analysis

- Unitary
- Intersectional

\* see <https://arxiv.org/abs/1810.03993>

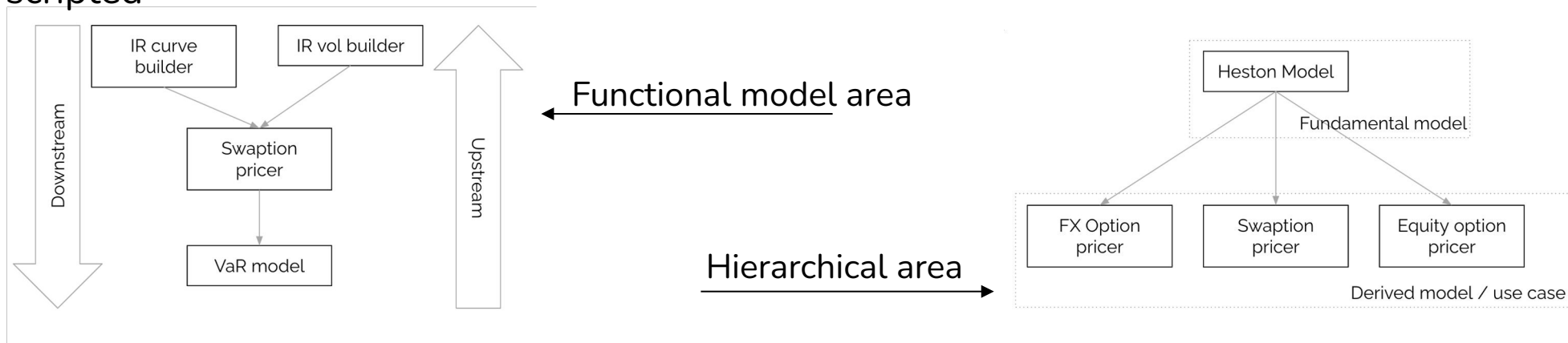


# Type of content

There are **four types of content** that are included in model documentation:

1. **Template specific**: defined at the level of the template
1. **Model area specific**: content that is shared amongst a group of models
1. **Model specific**: idiosyncratic content
1. **Test content**: Quantitative data (graphs, tables, ...)

**Note:** If models have a defined structure/API, generation of test content can be scripted



# Model card example

## Model Details

- stakeholders
- version
- methodology & references
- model tier

## Intended use

- Primary intended use
- Primary intended users
- Out-of-scope use cases

## Factors

- Groups
- Instrumentation
- Environment

## Metrics

- Performance
- Thresholds
- Uncertainty & variability

## Evaluation (& training) data

- Which datasets
- Motivation
- Preprocessing

## Quantitative analysis

- Unitary
- Intersectional

## Ethical considerations

- Data
- Human Life
- Risks & Harms
- Mitigations
- Use cases of attention

## Findings

- Caveats
- Recommendations

Template specific  
Model area specific  
Model specific  
Test content

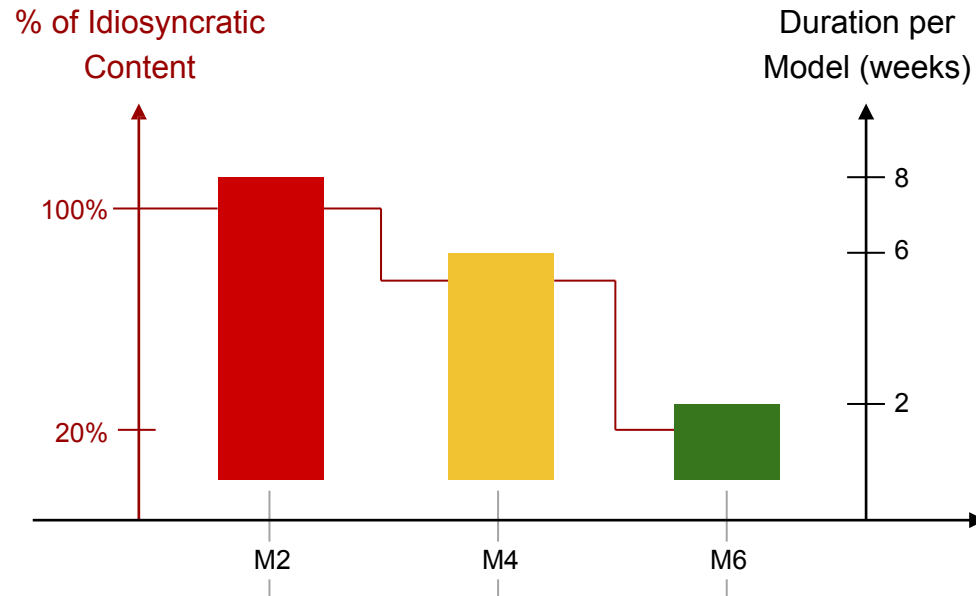
Sourced from inventory

80% shared!



# Onboarding process

Efficiency gains materialize gradually as more models are onboarded.



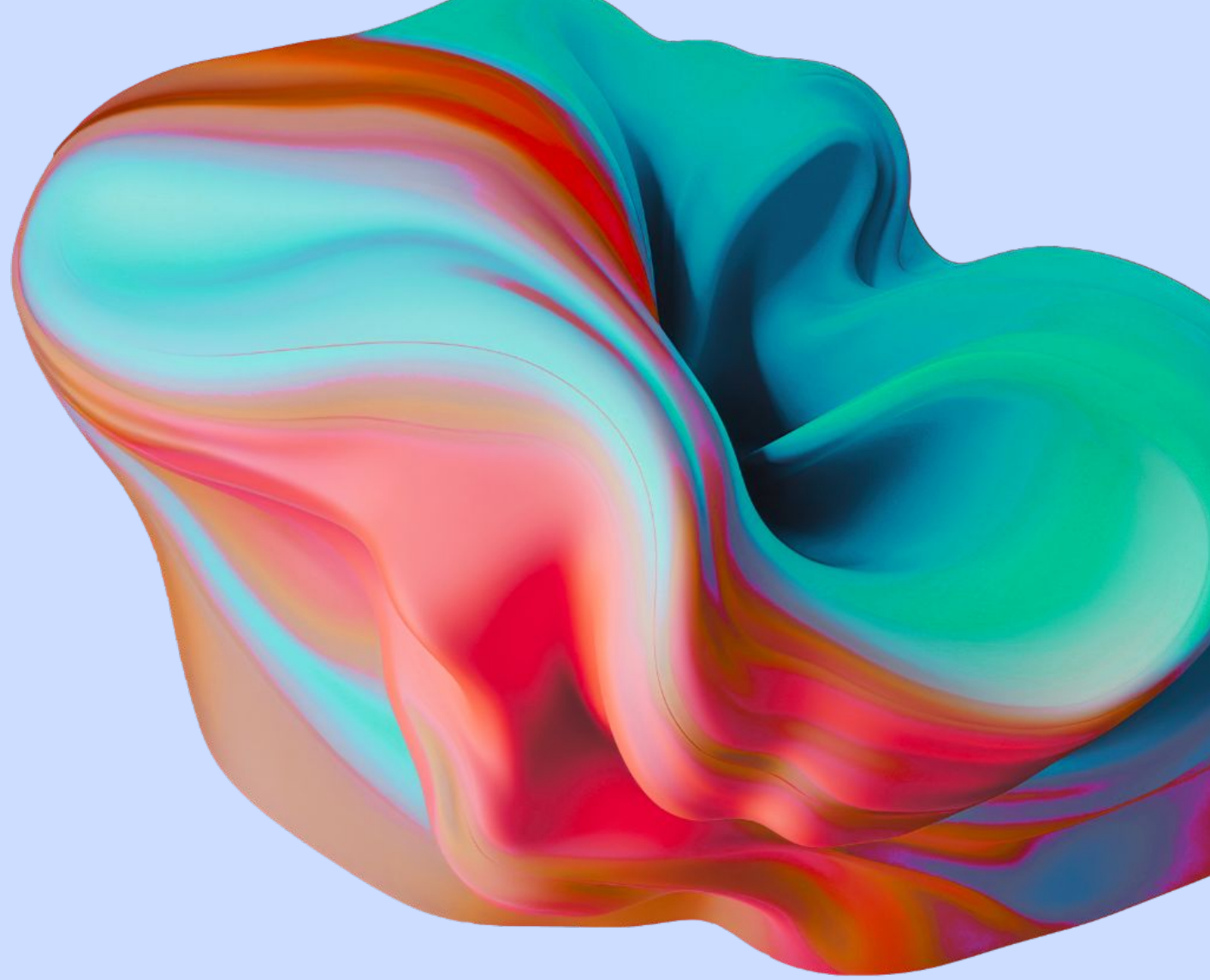
**Category (i):** No template and no shared content available; all content has to be created or sourced

**Category (ii):** Template available but incomplete shared content available.

**Category (iii):** Template and comprehensive shared content available.



# How technology can help





# Organizing content and testing

## Initiation

Transform the management of model documents into management of content elements.

## BAU

Executing tests and selecting content to generate artifacts & documentation

### Yields for Governance



#### Populate & source metadata

Model validation templates  
Four types of model content  
Workflows

#### Execute process to generate documentation



Metadata management

Documentation generation

Document Review

### Yields for Performance



#### Create specifications

Define test cases by model area

#### Create instances and sessions

Execute test cases against the model code

Quantitative testing



# Using rules and properties

If content is managed as individual elements, it can be leveraged to create additional automations.

## Quality gates for deployment

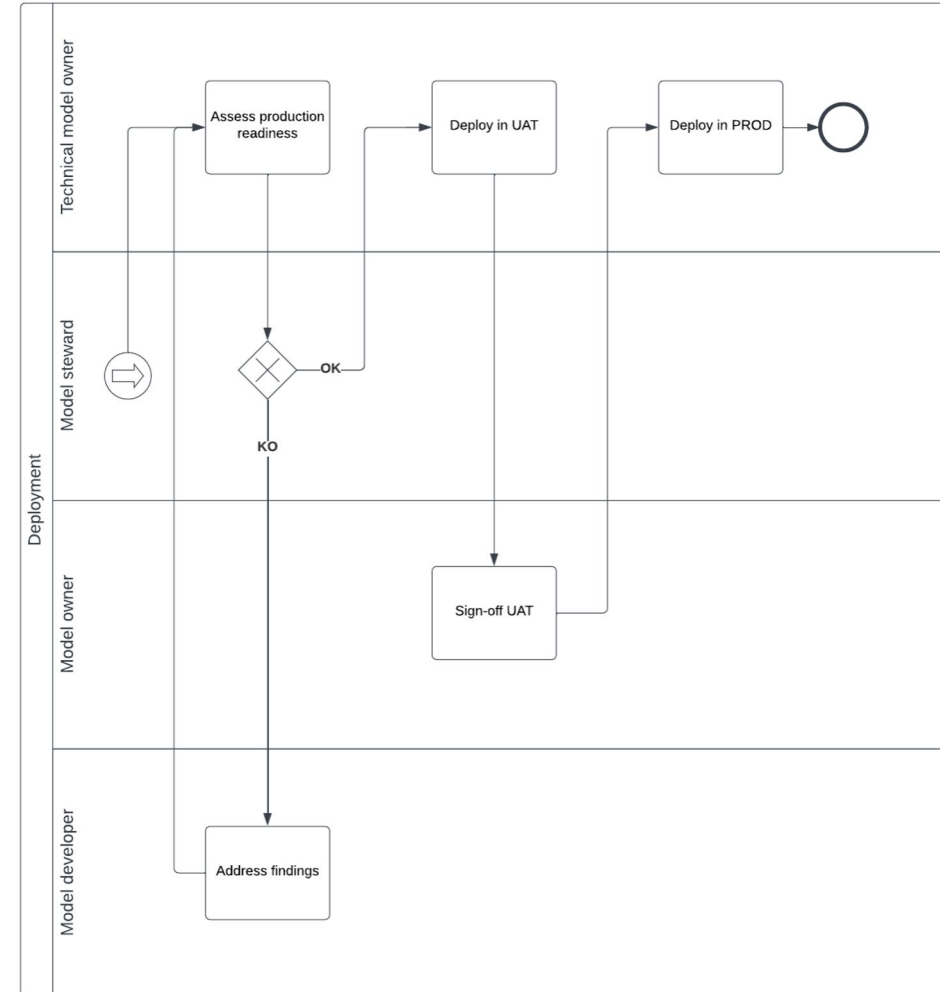
- Model can only reach PROD status if prerequisites are satisfied
- Verify the model status as part of a model deployment pipeline

## Trigger recalibration

- periodically run test templates to verify model performance
- If performance drops below threshold, then recalibrate
- This triggers a process for updating the documentation

## Generating multiple documents

- Generate by replacing placeholders with content
- Multiple templates reference different subsets of content



# Interactive model cards\*

**Design**  
Standard Model Card Details  
Visually Separate from interactive components

**Stakeholders**  
Use plain language

**Design**  
Overview summaries with details on demand

**Design**  
Visual emphasis on important details

**Guidance**  
Warnings and nudges

**Skepticism & Sensemaking**  
Define subpopulations, add examples or own data

**Guidance**  
Definitions and explanations with external links

**Guidance**  
Additional guidance documentation on metrics, the model, and next steps

**Skepticism & Sensemaking**  
Drill down into examples

**Design**  
Collapsible pointers to interaction options

**Skepticism & Sensemaking**  
Interactive legend for subpopulation companions

**Guidance**  
Collapsible explainer of chart interpretation

**Trust and Safety**  
Pre-defined (and customizable) analysis on at risk groups

**Design**  
Visually emphasize subpopulations with small sizes

**Landing Page**

**Data Comparison Panel**

\* See <https://arxiv.org/abs/2205.02894>



# Leveraging AI

## Measuring documentation quality

- Completeness
- Readability
- Adherence to a genre (requires building a corpus)

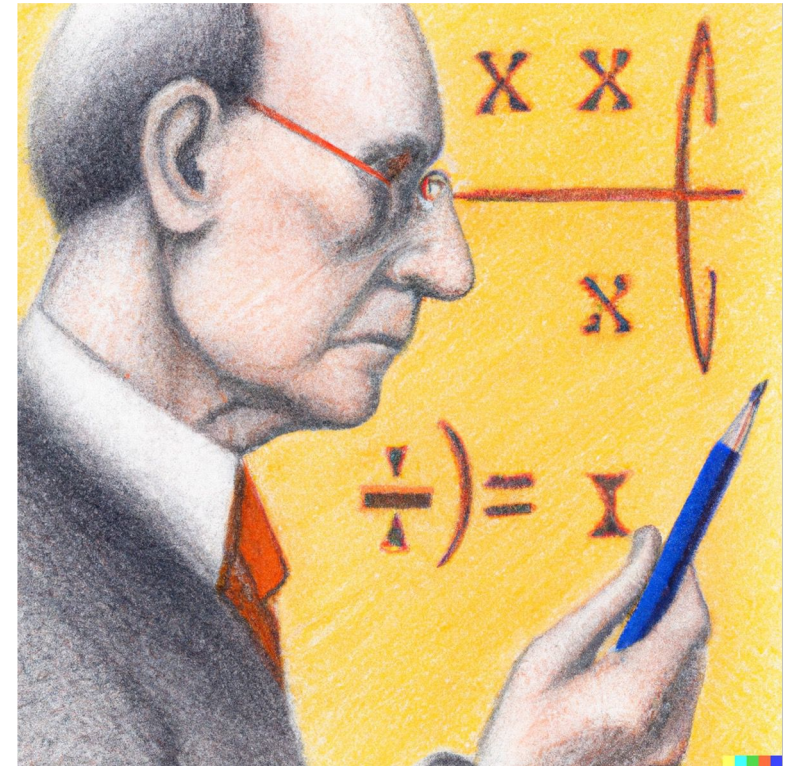
## Extracting content from files

- Populating shared content from existing documentation

## Translating Word to LaTeX and vice versa

**Note:** Generation of the documentation itself is not (yet) in scope

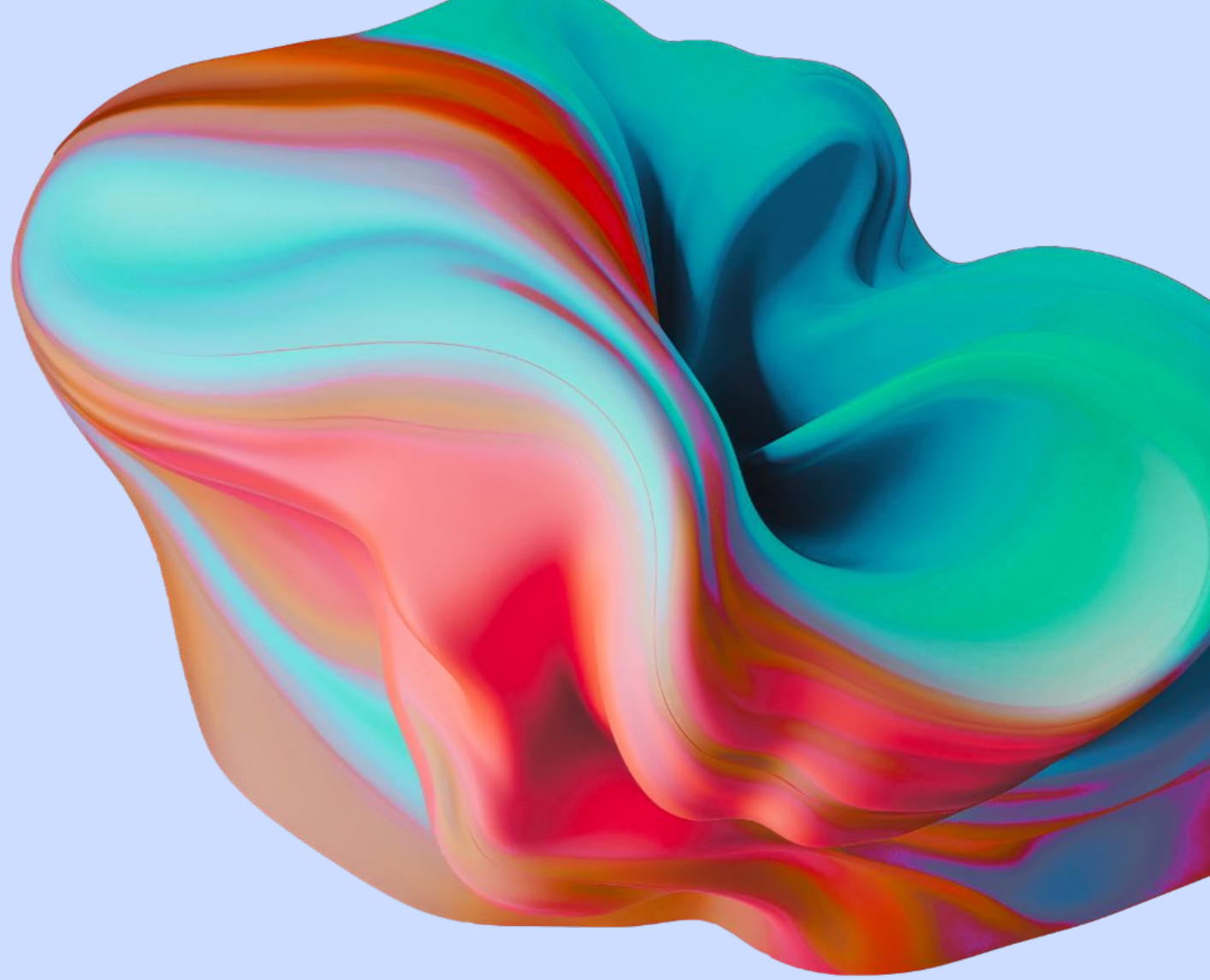
- Shorter documentation is better
- Issues with bias, hallucinations, ...



Generated by Dall-E, with prompt:  
“a color pencil drawing of a mathematician working as a model validator”



# Use cases from the field



# Examples from the field



Automated generation of consistent reporting across all credit risk models

- granular / technical reports
- high-level summary tables
- Interactive dashboards



Time spent on quantitative testing has been **reduced by a factor of 8**

**Automated reporting** of IRB monitoring





# Conclusion

Documentation is critical for model risk management.

By managing content elements, one can realize many benefits

- Consistency
- Increased efficiency
- Automation





# Thank you.

## Our Locations

-  HQ - Belgium
-  London, United Kingdom
-  New York, United States

 [www.yields.io](http://www.yields.io)

 [info@yields.io](mailto:info@yields.io)