IMEN891K / IDSC501 / AXGS-501 · DATA ENGINEERING · FALL 2026

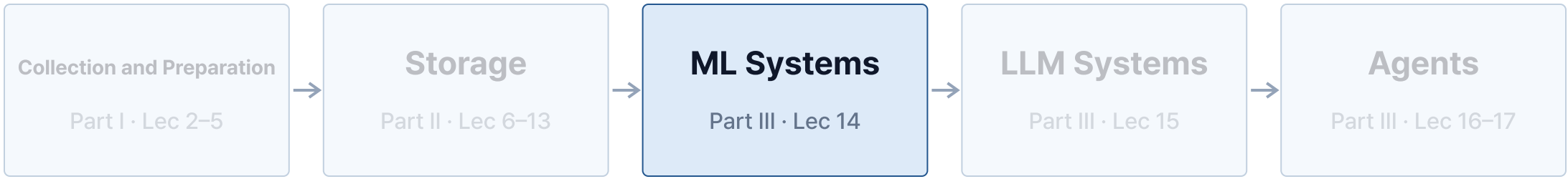
14.

ML Systems

Sangwoo Mo · POSTECH

Industrial & Management Engineering

Where We Are



Part III connects stored data to predictions, answers, and actions

Outline: ML Systems

01

Preparing Examples

use what was known at prediction time

02

Making Predictions

keep the input meaning the same

03

Checking Results

compare predictions with observed outcomes

01 · Preparing Examples

01

Preparing Examples

use what was known at prediction time

02

Making Predictions

keep the input meaning the same

03

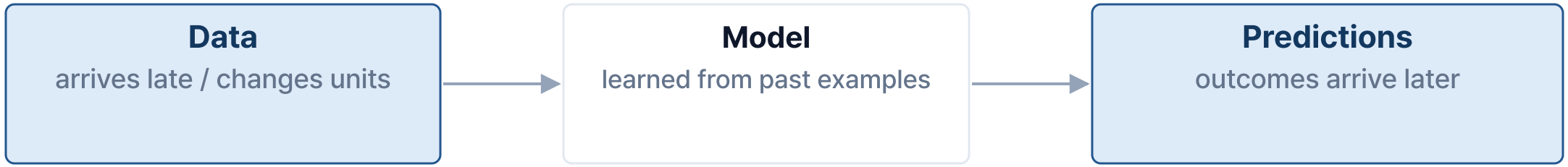
Checking Results

compare predictions with observed outcomes

Why ML Systems

A useful model also needs reliable data, predictions and feedback

2015 · Hidden Technical Debt in Machine Learning Systems

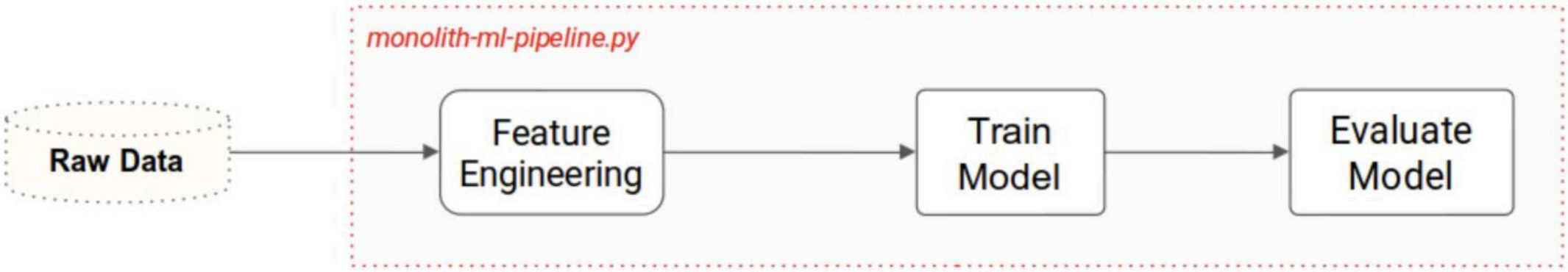


A good experiment still needs a dependable path to everyday predictions

The model is one component of a system that must keep working

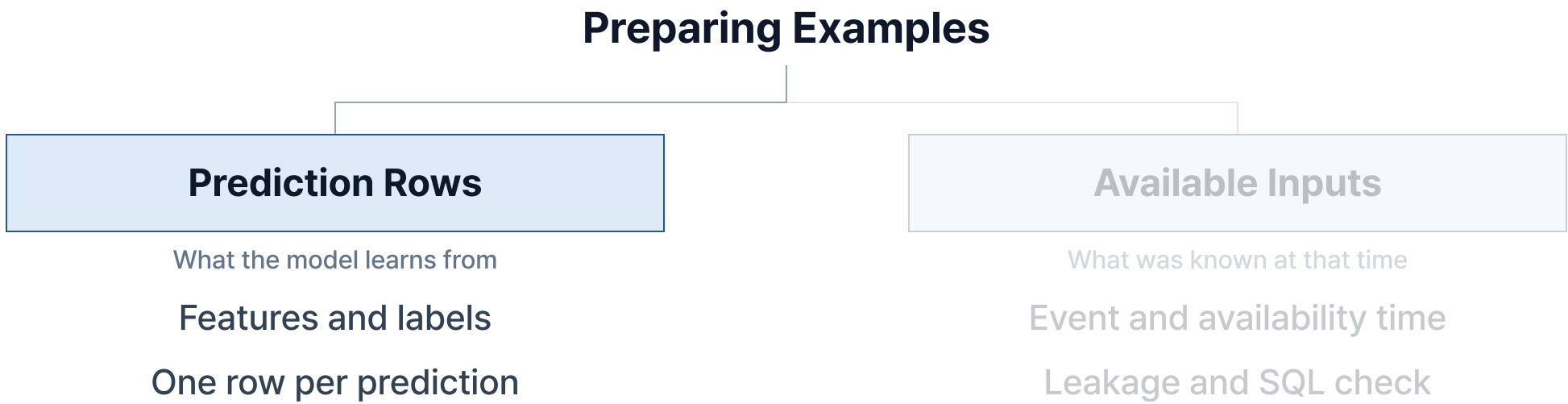
From Data to a Checked Model

Prepare model inputs, learn from past outcomes, then check predictions



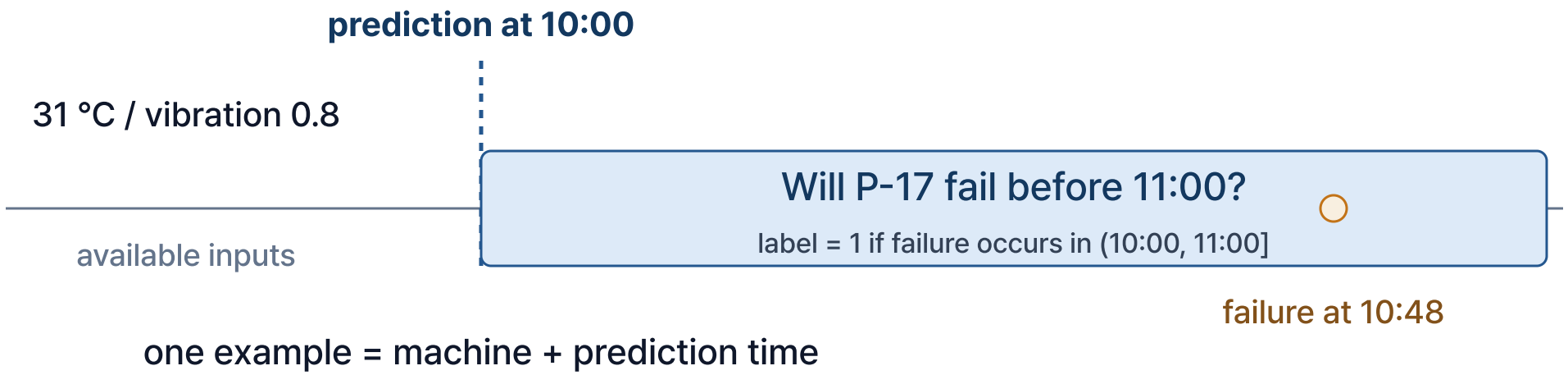
The pipeline connects data preparation to training and evaluation

§ Prediction Rows



Failure Prediction Example

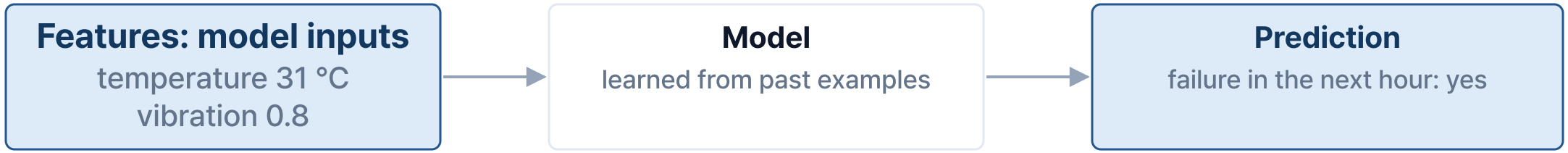
At 10:00, predict whether P-17 will fail during the next hour



The 10:48 failure becomes the label for the 10:00 prediction

Features and Labels

Features are model inputs and the label is the observed outcome

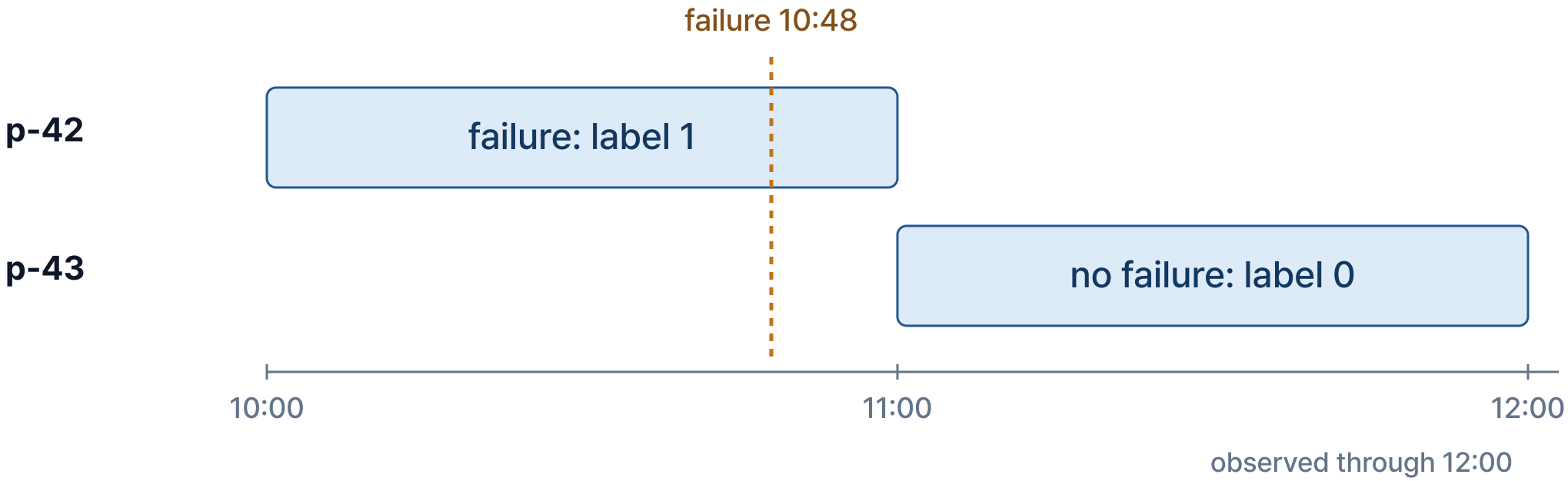


Label: the observed outcome arrives later

The model predicts a label that has not arrived yet

The Next-Hour Label

A label answers a question about one future time interval



The 10:48 failure belongs to the earlier prediction window

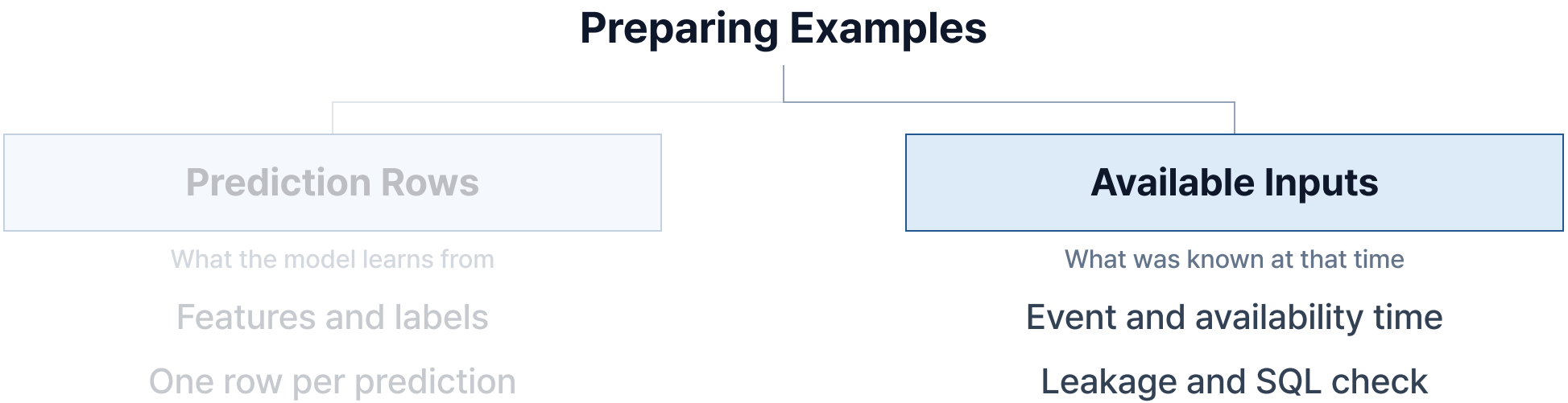
Training Examples

One row combines past inputs with the outcome that followed

Machine	Prediction time	Temperature	Vibration	Label
P-17	Sep 1 10:00	31 °C	0.8	1
P-17	Sep 1 11:00	28 °C	0.2	0
P-17	Sep 2 10:00	34 °C	0.9	1

Each row describes one P-17 prediction, rather than the whole machine

§ Available Inputs



Feature Availability

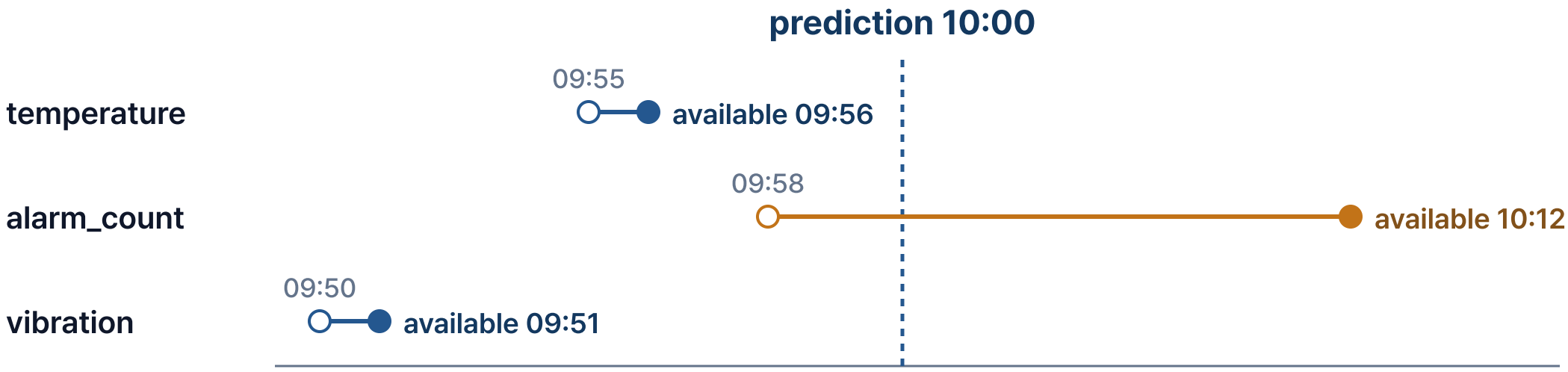
Event: when it happened. Available: when the system could read it

Feature	Event	Available	Value
temperature	09:55	09:56	31
alarm_count	09:58	10:12	3
vibration	09:50	09:51	0.8

The alarm happened before 10:00 but arrived at 10:12

Late Feature Arrival

An earlier event can still be unknown when the model predicts



Only temperature and vibration are available before the prediction line

Caveat: Future Information

Using a late input during training gives the model information it would lack

Training input at 10:00	Values	Available then?
temperature / vibration	31 / 0.8	yes
alarm_count	3	no, arrived 10:12

The late alarm would make the training example unrealistically informative

Demo: Available Inputs

Run the join, then exclude inputs that arrived after 10:00

1. Join Predictions and Features using machine_id; filter event_at ≤ predicted_at.
2. Also filter available_at ≤ predicted_at. Which feature disappears?

```
-- write your query here, then Run
```

QUERY RESULT

Write a query on the left, then Run.

Answer: Available Inputs

Both time conditions must hold for the input to be usable

```
SELECT p.id, f.feature, f.value
FROM Predictions p JOIN Features f ON
f.machine_id=p.machine_id
WHERE f.event_at <= p.predicted_at AND
f.available_at <= p.predicted_at
ORDER BY p.id, f.feature;
```

id	feature	value
prediction_42	temperature	31.0
prediction_42	vibration	0.8

The late alarm report is excluded

Available Model Inputs

The filtered result reconstructs what the predictor could actually read

Prediction	Usable input	Value
P-17 at 10:00	temperature	31
P-17 at 10:00	vibration	0.8

The two remaining values are the inputs for the 10:00 prediction

02 · Making Predictions

01

Preparing Examples

use what was known at prediction time

02

Making Predictions

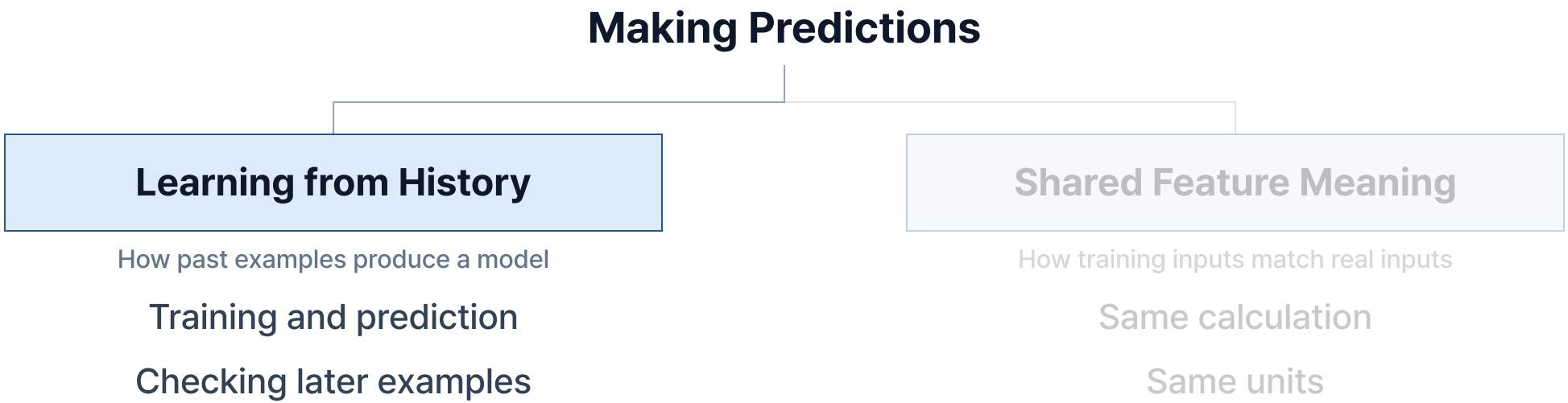
keep the input meaning the same

03

Checking Results

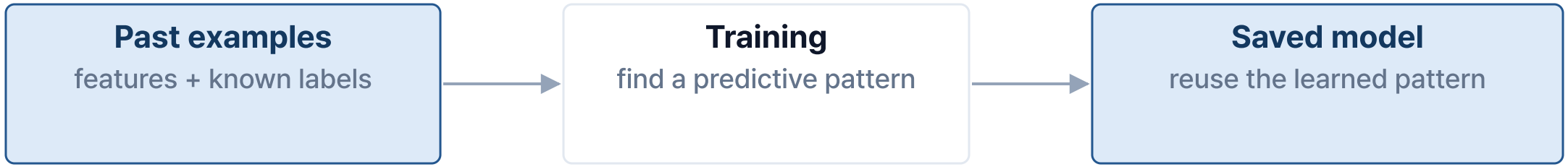
compare predictions with observed outcomes

§ Learning from History



Training and Prediction

Training learns from known outcomes and prediction uses the saved model

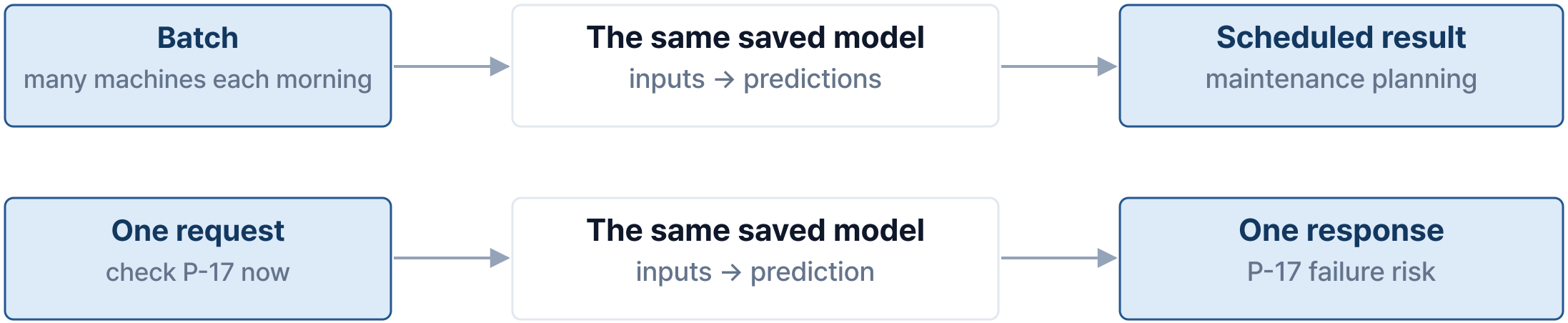


Prediction uses features while the future label is unknown

The model learns from past labels and predicts a future label

Batch and Request-Time Predictions

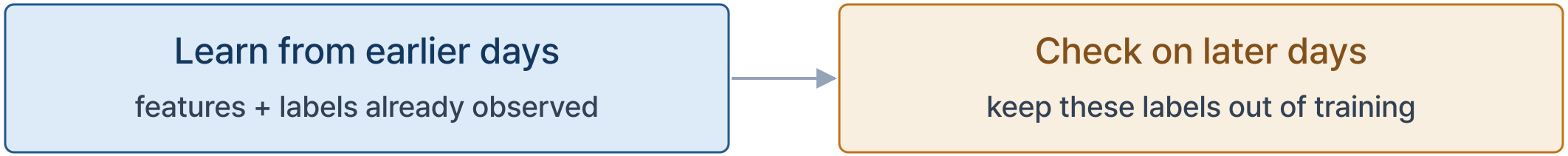
The same model can score many rows together or answer one request



Choose when predictions are needed before choosing the execution mode

Checking Later Examples

Check the model on later examples that it did not learn from



P-17: train before Sep 8, check predictions from Sep 8 onward

Past training data and later checking data serve different purposes

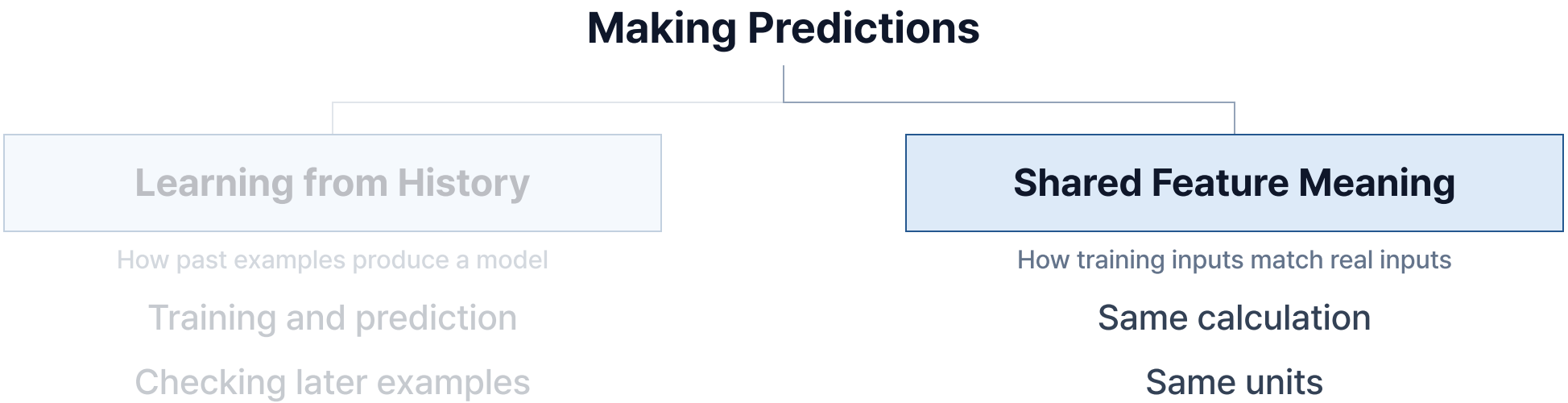
Predictions and Outcomes

A prediction is correct only when it matches the observed label

P-17 prediction	Predicted failure	Observed label	Correct?
Sep 8 10:00	yes	1	yes
Sep 8 11:00	no	0	yes
Sep 9 10:00	no	1	no
Sep 9 11:00	no	0	yes

Three of the four predictions match their observed outcomes

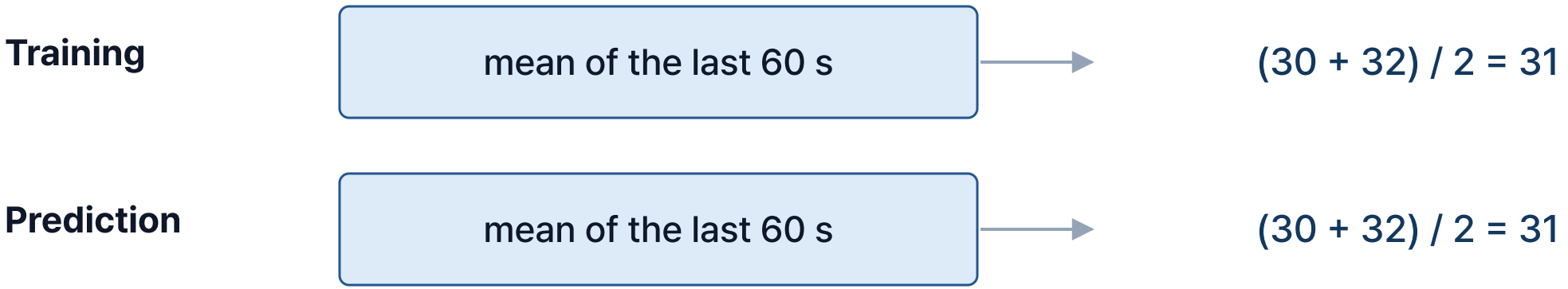
§ Shared Feature Meaning



A Temperature Feature

A feature definition tells us how source readings become one input

P-17 temperature readings: 30 °C at 09:59:10, 32 °C at 09:59:50



Both paths turn the same readings into the value 31

Caveat: Different Calculations

A last reading and a mean can share a name while carrying different meanings

P-17 temperature readings: 30 °C at 09:59:10, 32 °C at 09:59:50

Training	mean of the last 60 s	(30 + 32) / 2 = 31
Prediction	last reading	32

Training receives 31 while prediction receives 32 from the same readings

Shared Feature Meaning

Keep the calculation and the unit the same in both paths

Path	Calculation	Unit	Input
Training	mean of last 60 s	°C	31
Prediction	mean of last 60 s	°C	31

The model receives the same kind of input during training and prediction

03 · Checking Results

01

Preparing Examples

use what was known at prediction time

02

Making Predictions

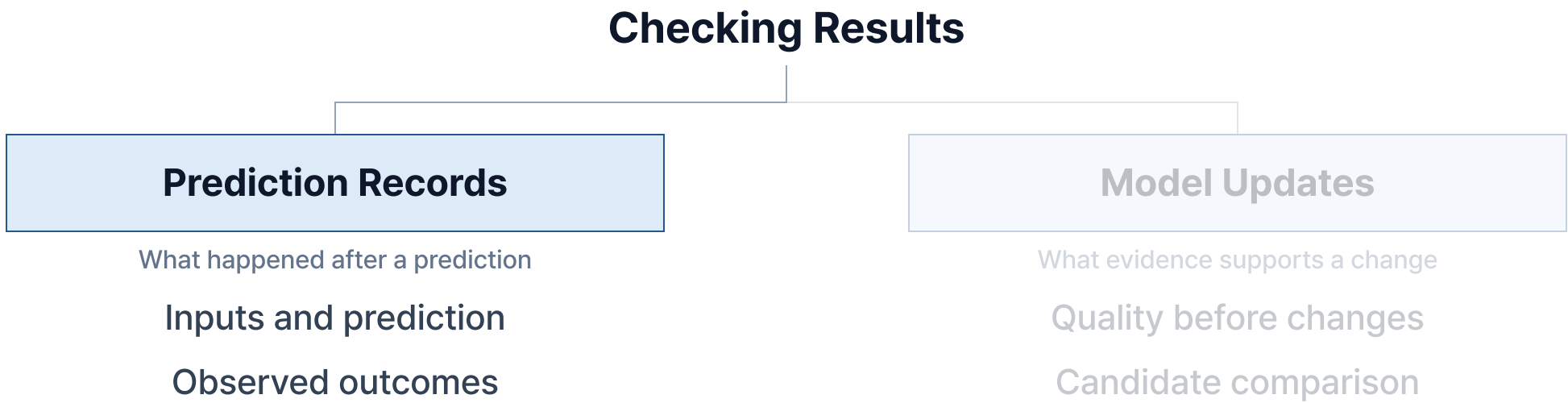
keep the input meaning the same

03

Checking Results

compare predictions with observed outcomes

§ Prediction Records



Prediction Record

Keep the input and prediction so the later outcome can be matched

Prediction ID	Machine / Time	Model inputs	Prediction	Observed label
p-42	P-17 / 10:00	31 °C / 0.8	failure: yes	?

The 10:00 prediction is recorded before its outcome is known

Observed Outcome

Attach the observed failure to the prediction it answers

Prediction ID	Machine / Time	Model inputs	Prediction	Observed label
p-42	P-17 / 10:00	31 °C / 0.8	failure: yes	1: failure

The 10:48 failure confirms the recorded prediction p-42

Delayed Outcomes

Recent predictions may still have missing outcomes

Ten P-17 predictions from the same day



Day 1: 4 labels, 4 correct

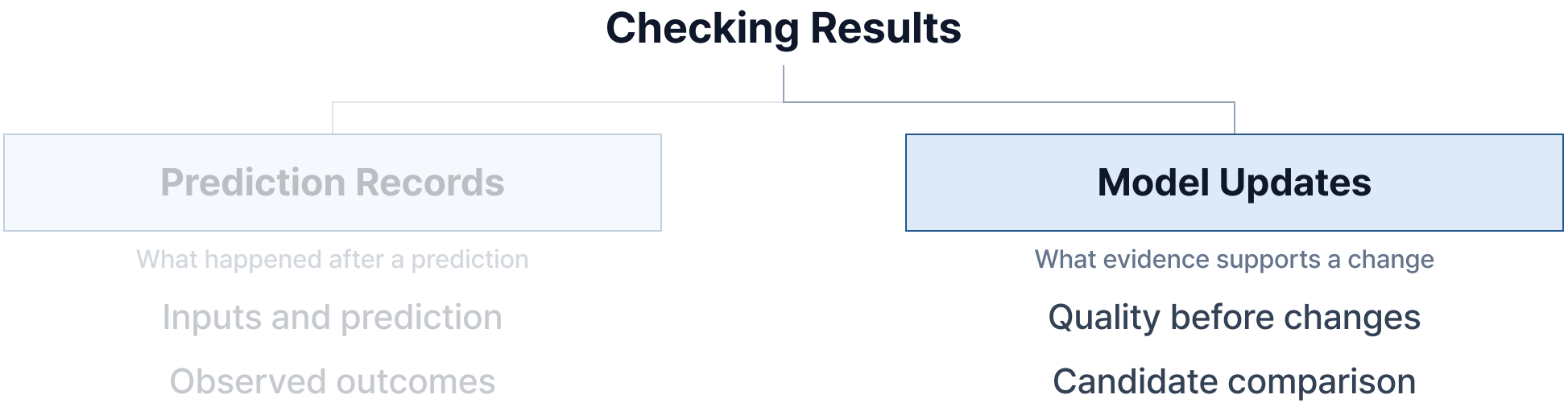
Only 4 outcomes are known

Day 3: 10 labels, 6 correct

All 10 outcomes are known

The same ten predictions score differently as more labels arrive

§ Model Updates



Input Changes and Quality

Check what changed before deciding to train a new model

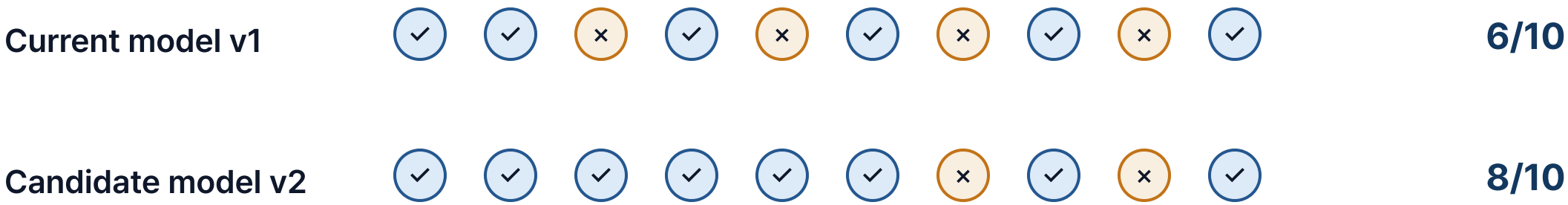
Check	Earlier	Now	Meaning
Temperature unit	°C	°F	fix input conversion
Correct predictions	6 / 10	6 / 10	quality unchanged

A unit change needs an input fix before any model update

Candidate Comparison

Compare the current model and a candidate on the same later examples

The same ten later predictions, with all labels observed



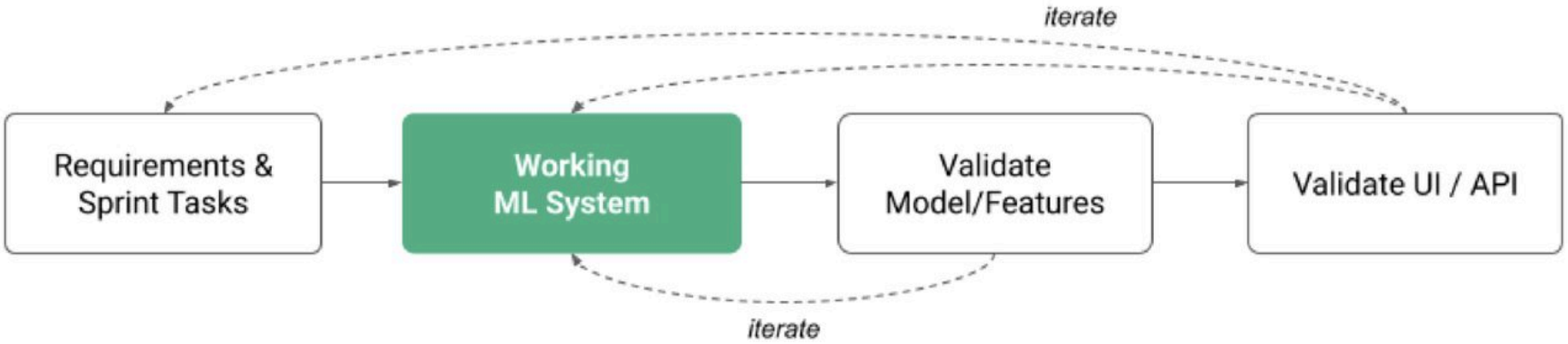
Each mark compares one prediction with its observed label

The candidate matches eight outcomes while the current model matches six

MLOps: A Repeating Cycle

Develop a model, use it, observe outcomes and improve the next candidate

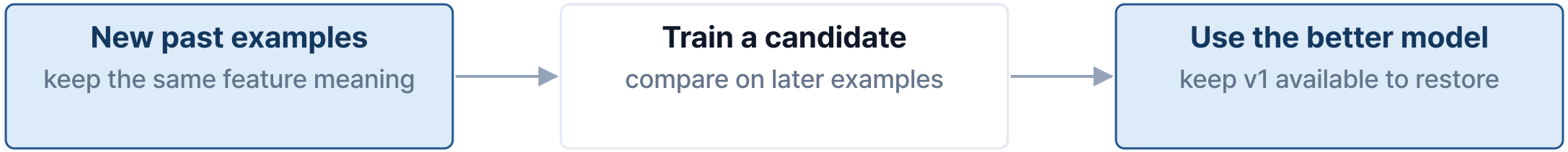
- A goal of MLOps is to improve both iteration speed and quality when developing ML



Operating the model continues after the first deployment

Model Update

Use observed examples to train and check a candidate before replacing the model



A checked candidate can replace v1 while v1 remains available

Takeaway: Operate with available data

1. Use only inputs available when the prediction was made
2. Compute the same feature meaning during training and prediction
3. Record predictions and observed outcomes before deciding to update