Sight Machine

Common Data Models for Manufacturing

SIGHT MACHINE

Background and Presentation Overview

- Sight Machine analyzes production by streaming plant floor data from all systems and sources
- Data is streamed into 4 Common Data Models. These models represent automated production in all industries
- This method is different. It's streaming, so it's real-time, which is necessary for plants – otherwise plants can't react
- Also, the foundational unit that is used to represent production Data is different
- To represent plant activity, most experts start by building models of machines and processes. Sight Machine structures OT Data differently into basic, universal units. These rows of Data describe each value-added step done by every machine
- This approach yields standardized Data Foundation: all plant data structured into standardized structures with a high degree of flexibility as to data types and parameters incorporated
- Data Foundation is then graphed into representations of machines, lines and plants, which are further visualized and analyzed. Data Foundation enables expansive KPIs, analytics and AI/ML
- Sight Machine's architecture is **modular, transparent, and configurable at each level**. Clients and partners can access and modify: (a) raw data, (b) configuration, and (c) Transformed Data via API and SDK layers
- This presentation reviews Common Data Models and graphing methods used to build up higher-level models. It then highlights a few from hundreds of analytics currently offered with web services. After several slides, it is mostly pictures ☺
- Connectivity, pre-processing, streaming and transformation are addressed elsewhere
- A summary architecture is provided last

Things we want to understand in manufacturing

- Machine
- Machine Type
- Facility
- Cycle
- Downtime
- Part
- Batch
- Defect
- Line
- KPI

Problem to be solved: when analyzing plant floor production, customary modeling approaches don't work

- Most manufacturing activities besides production itself are successfully analyzed with batch processing, existing software systems, and traditional modeling techniques. Areas where traditional methods work include Product lifecycle (PLM), business processes (ERP), physics-based simulation, Supply Chain (SCM)
- Analysis of actual plant floor production (OT Data) is different. Despite substantial investment in the last decade by both technology and industrial companies, traditional modeling techniques have not worked for plant floors
- Among the many challenges, one of the more fundamental is around representational units
- Because machines are often the representational unit we think of first, when analyzing production we usually model machines. Other representational units we might use include single sensors, larger data sources (*e.g.*, historians, quality systems) lines, plants, and parts
- But manufacturing has many thousands of kinds of machines and hundreds of systems, and relationships among data sources are complex. This complexity is one reason traditional approaches fail
- **For plant floors, we need a simpler approach with only a few elemental units that can be continuously generated and "built up" to represent all machines, lines, and plants**

What’s needed are common building blocks

- We need to use building blocks that are common across every machine and process
- These building blocks should be constructed from the OT data as it is created, independent of the machine types and software systems that generated the data
- We can then stream and transform Data into standardized units, and assemble these units into higher-level conceptual templates. This way, we can relate every machine, process, and part with the same underlying, standardized elements
- These building blocks are Common Data Models for Production. With these models, mapped through stream processing, we can represent almost every activity in manufacturing with the same Data Foundation
- **This approach is essential for scale. It enables analysis not just from the asset to enterprise level, but also across industries and Value Chains**

Manufacturing Common Data Models

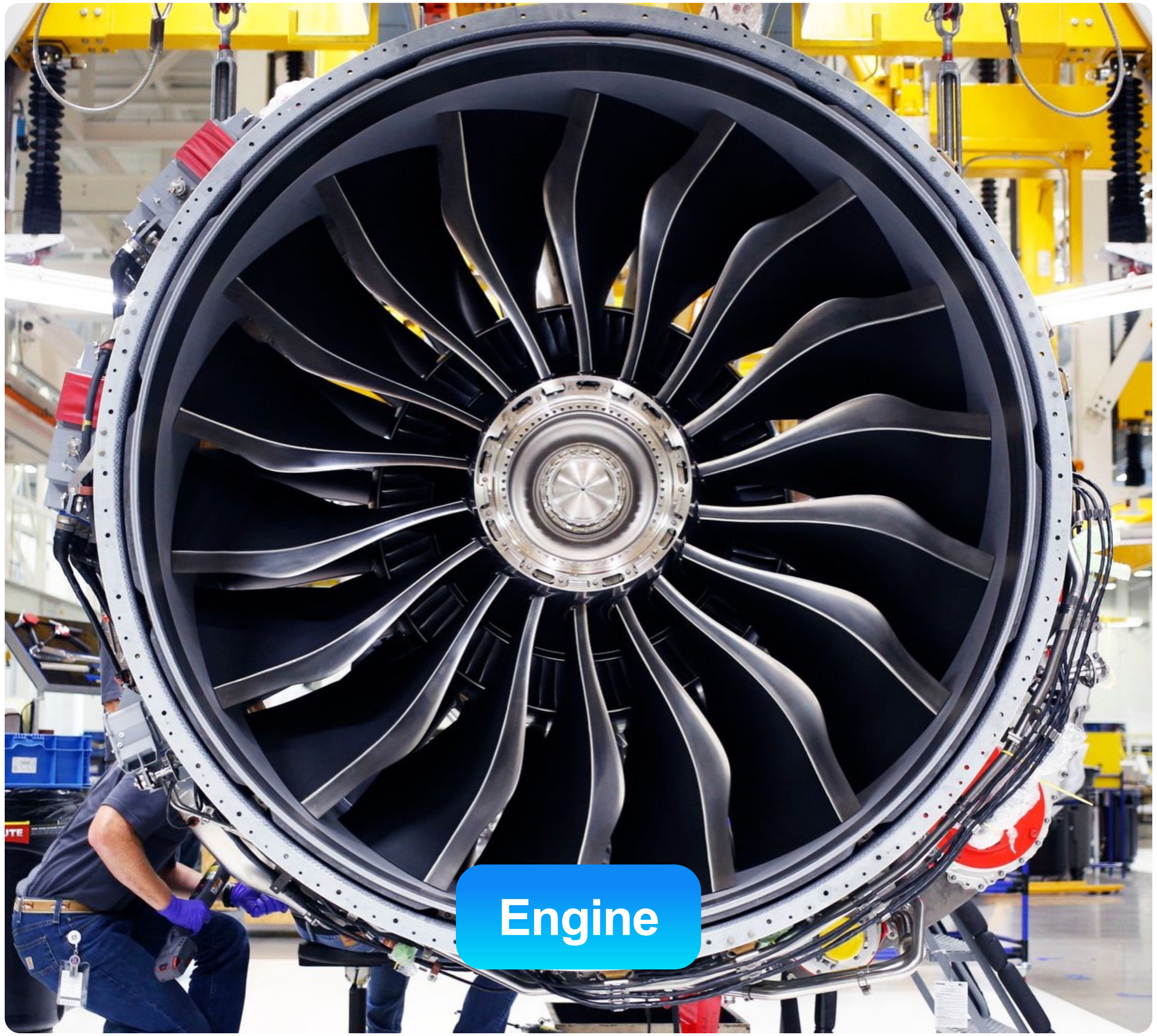
To represent infinite machines, parts, processes, and plants, only a few Common Data Models are needed

- Building blocks are represented as rows in data tables
- Columnar values are parameters of interest, generated through real-time data transformations
- Columnar data combines and transforms data from any source:
  - Machines (hundreds of sensors each, thousands of different machines per enterprise)
  - Operator data (Excel, entries, digitized reports)
  - Adjacent systems (historians, ERP, MES, quality, etc.)
  - Environment (temperature, humidity)
  - Value chain (raw material, upstream supplier)
- The resulting information is now **Data Foundation**. It is continuously generated and standardized, and can be associated, combined, compared, and analyzed with all other standardized units of information
- Data Foundation enables broad application of analytical techniques from visualization and KPIs to data science and AI/ML. Data Foundation can also be joined with other information (finance, energy, logistics) in manufacturing enterprises
- **Data Foundation supports both real-time operational analysis and firm, or industry-level analytics**

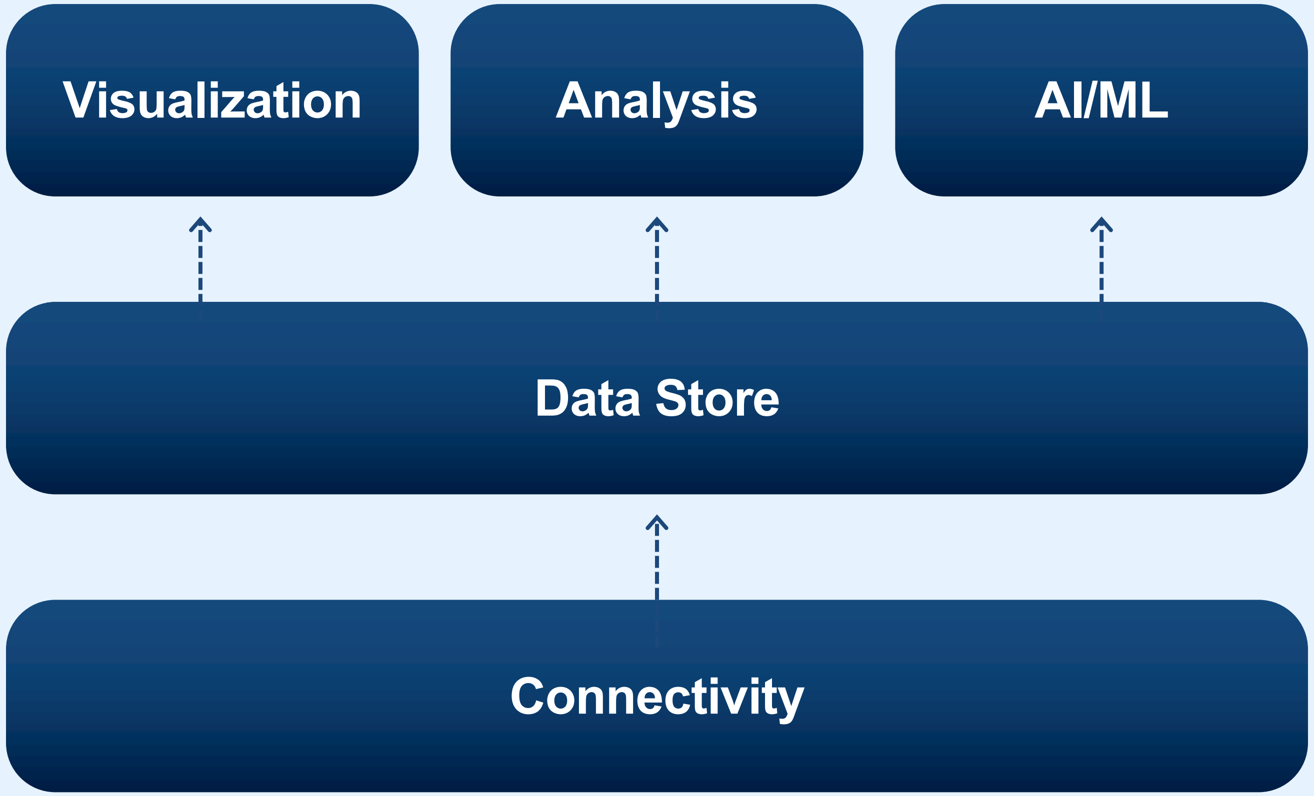
This Background In Pictures

SIGHT MACHINE

Free-standing, single assets can be modeled by representing “the Machine.” Manufacturing is different. Even simple products involve hundreds of machines



The ideal software stack is logical, streamlined.
Manufacturing software is accreted over decades, tailored, complex



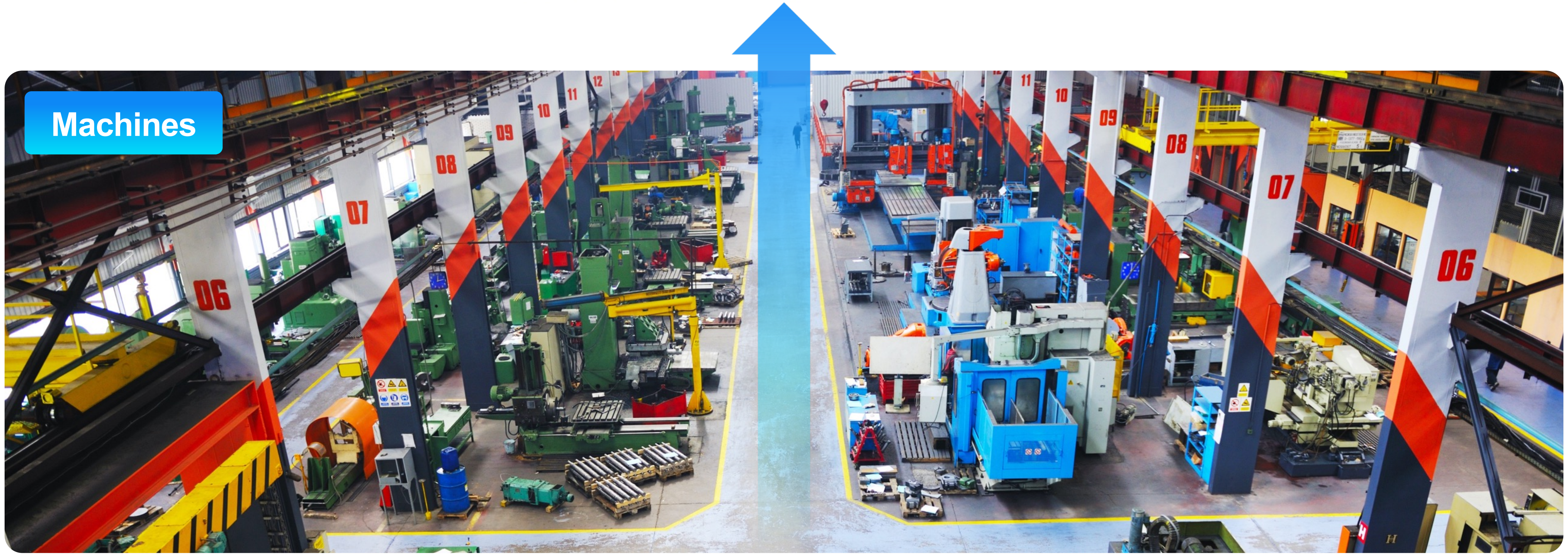
Ideal Software Architecture

Typical Manufacturing Architecture

Source: CESMII, 2021

The next challenge:
Production data flows through manufacturing software

Production*	New Mfg Systems	Quality*	Logistics*	Maintenance*	Management*
Production Monitoring	Digital Twin Apps	Campaigns	Environmental Systems	Tool Monitoring	Information Portal
Downtime Analysis	Device/Machine IoT Apps	Shipping – Error Proofing	Metrics (DTD, BTS)	Predictive	Energy/Building Mgmt
Production Counts	Predictive Maintenance 1	External Metrics	Shipping	Preventive	Cost Management
Constraint ID & Analysis	Predictive Maintenance 2	Internal Metrics	Container Tracking	Crisis	Policy Deployment
Schedule Execution/	Predictive Maintenance 3	Traceability	Lot Tracking	Reliability & Maintenance	Best Practice
Hit to Hit	Remote Monitoring	Error Proofing	Rack Tagging	Downtime Analysis	Timekeeping
Process Control Boards	Connected Worker 1	Defect Tracking	Raw Mat Ordering & Inv	Fluid Tracking	Metrics for Workforce
Dock Inventory	Connected Worker 2	SPC	Dock Inventory	Programmable Dev Supp	Issues Management
Process Sheets	Augmented Reality 1	Repair Bay OI	Schedule Optimization	Plant Design & Layout	Production Log
Manual Production Count	Augmented Reality 2	Quarantine	Inventory Control	Die Re-Chroming	Knowledge Management
Cycle Time Analysis	Supply Chain Analytics/Ops	Scrap & Reject Reporting	Line Side Replenishment		Training
Sequencing	Hot, Warm, Cold Data Lake	Birth History	Part Consumption		

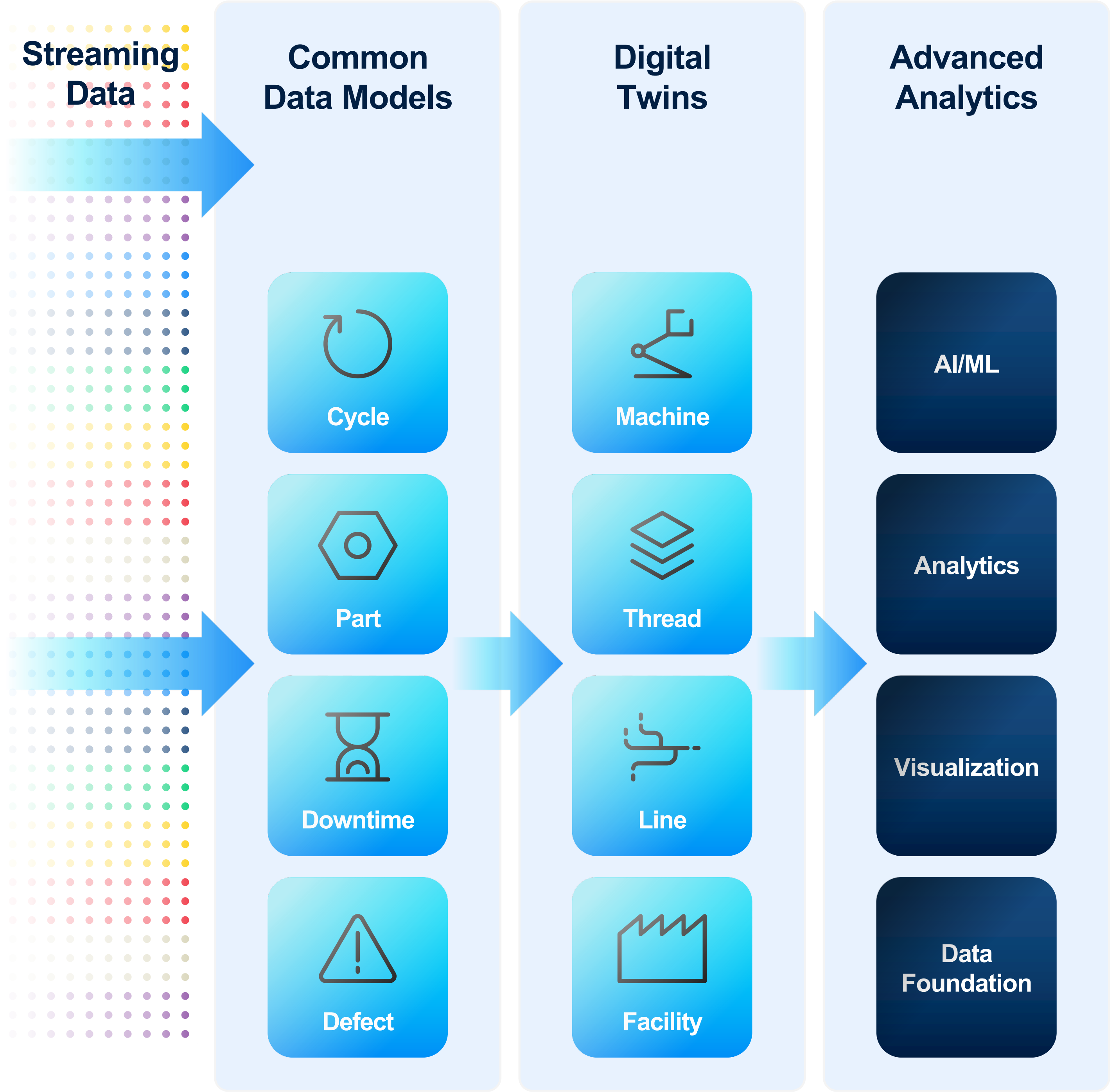


Source: CESMII, 2021

How to standardize analysis? It’s the data

Production*	New Mfg Systems	Quality*	Logistics*	Maintenance*	Management*
Production Monitoring	Digital Twin Apps	Campaigns	Environmental Systems	Tool Monitoring	Information Portal
Downtime Analysis	Device/Machine IoT Apps	Shipping – Error Proofing	Metrics (DTD, BTS)	Predictive	Energy/Building Mgmt
Production Counts	Predictive Maintenance 1	External Metrics	Shipping	Preventive	Cost Management
Constraint ID & Analysis	Predictive Maintenance 2	Internal Metrics	Container Tracking	Crisis	Policy Deployment
Schedule Execution/	Predictive Maintenance 3	Traceability	Lot Tracking	Reliability & Maintenance	Best Practice
Hit to Hit	Remote Monitoring	Error Proofing	Rack Tagging	Downtime Analysis	Timekegging
Process Control Boards	Connected Worker 1	Defect Tracking	Raw Mat Ordering & Inv	Fluid Tracking	Metrics for Workforce
Dock Inventory	Connected Worker 2	SPC	Dock Inventory	Programmable Dev Supp	Issues Management
Process Sheets	Augmented Reality 1	Repair Bay OI	Schedule Optimization	Plant Design & Layout	Production Log
Manual Production Count	Augmented Reality 2	Quarantine	Inventory Control	Die Re-Chroming	Knowledge Management
Cycle Time Analysis	Supply Chain Analytics/Ops	Scrap & Reject Reporting	Line Side Replenishment		Training
Sequencing	Hot, Warm, Cold Data Lake	Birth History	Part Consumption		

Source: CESMII, 2021



Common Data Models

SIGHT MACHINE

Cycle

Cycle characterizes discrete periods of machine activity. A cycle represents a unit of work by a machine.

Each row in a Sight Machine data table describes a single recurring unit of work by any machine. Examples of repetitive work done by machines are limitless and include pressing, cutting, filling, painting, and heating. **Cycle** creates a common thread across machines, lines, process areas, facilities, and industries

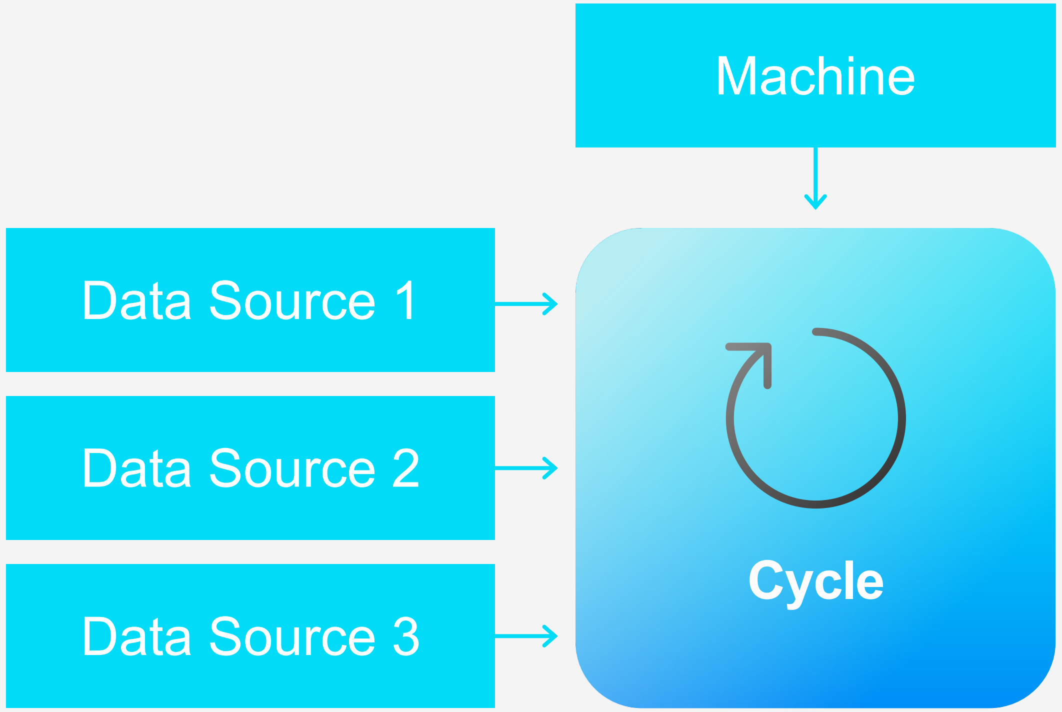
Cycle Data Model

Unit of work

- Primarily identified by Machine + Time (sometimes Machine + Counter/Unique Identifier)
- Contains a shift, production day, and output

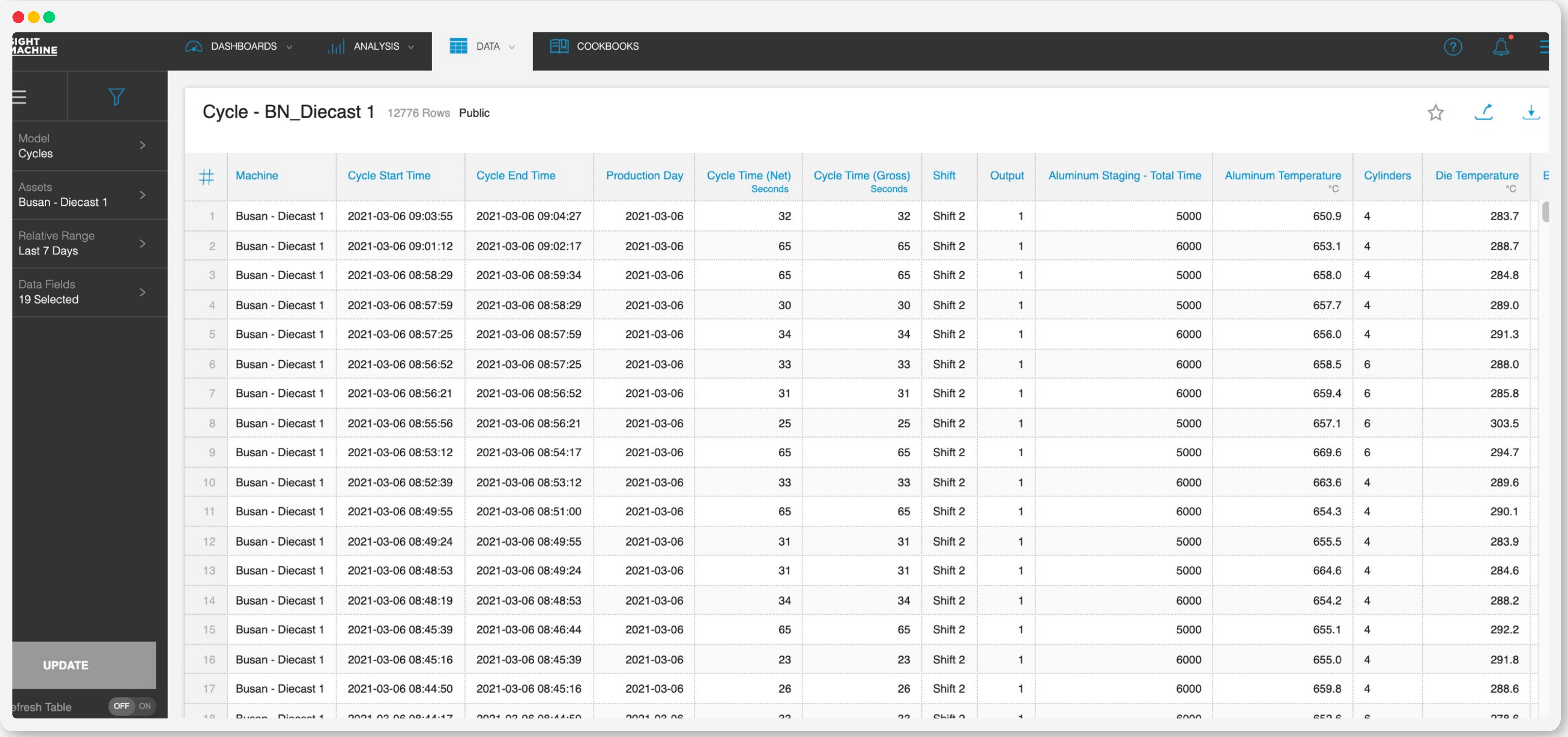
Commonly includes:

- Machine telemetry
- Running status
- Product/material information
- Energy use
- Quality/inspection data



Records from grouping/ aggregation by time, counter, status codes

Example Cycle Data Table



#	Machine	Cycle Start Time	Cycle End Time	Production Day	Cycle Time (Net) Seconds	Cycle Time (Gross) Seconds	Shift	Output	Aluminum Staging - Total Time	Aluminum Temperature °C	Cylinders	Die Temperature °C
1	Busan - Diecast 1	2021-03-06 09:03:55	2021-03-06 09:04:27	2021-03-06	32	32	Shift 2	1	5000	650.9	4	283.7
2	Busan - Diecast 1	2021-03-06 09:01:12	2021-03-06 09:02:17	2021-03-06	65	65	Shift 2	1	6000	653.1	4	288.7
3	Busan - Diecast 1	2021-03-06 08:58:29	2021-03-06 08:59:34	2021-03-06	65	65	Shift 2	1	5000	658.0	4	284.8
4	Busan - Diecast 1	2021-03-06 08:57:59	2021-03-06 08:58:29	2021-03-06	30	30	Shift 2	1	5000	657.7	4	289.0
5	Busan - Diecast 1	2021-03-06 08:57:25	2021-03-06 08:57:59	2021-03-06	34	34	Shift 2	1	6000	656.0	4	291.3
6	Busan - Diecast 1	2021-03-06 08:56:52	2021-03-06 08:57:25	2021-03-06	33	33	Shift 2	1	6000	658.5	6	288.0
7	Busan - Diecast 1	2021-03-06 08:56:21	2021-03-06 08:56:52	2021-03-06	31	31	Shift 2	1	6000	659.4	6	285.8
8	Busan - Diecast 1	2021-03-06 08:55:56	2021-03-06 08:56:21	2021-03-06	25	25	Shift 2	1	5000	657.1	6	303.5
9	Busan - Diecast 1	2021-03-06 08:53:12	2021-03-06 08:54:17	2021-03-06	65	65	Shift 2	1	5000	669.6	6	294.7
10	Busan - Diecast 1	2021-03-06 08:52:39	2021-03-06 08:53:12	2021-03-06	33	33	Shift 2	1	6000	663.6	4	289.6
11	Busan - Diecast 1	2021-03-06 08:49:55	2021-03-06 08:51:00	2021-03-06	65	65	Shift 2	1	6000	654.3	4	290.1
12	Busan - Diecast 1	2021-03-06 08:49:24	2021-03-06 08:49:55	2021-03-06	31	31	Shift 2	1	5000	655.5	4	283.9
13	Busan - Diecast 1	2021-03-06 08:48:53	2021-03-06 08:49:24	2021-03-06	31	31	Shift 2	1	5000	664.6	4	284.6
14	Busan - Diecast 1	2021-03-06 08:48:19	2021-03-06 08:48:53	2021-03-06	34	34	Shift 2	1	6000	654.2	4	288.2
15	Busan - Diecast 1	2021-03-06 08:45:39	2021-03-06 08:46:44	2021-03-06	65	65	Shift 2	1	5000	655.1	4	292.2
16	Busan - Diecast 1	2021-03-06 08:45:16	2021-03-06 08:45:39	2021-03-06	23	23	Shift 2	1	6000	655.0	4	291.8
17	Busan - Diecast 1	2021-03-06 08:44:50	2021-03-06 08:45:16	2021-03-06	26	26	Shift 2	1	6000	659.8	4	288.6

Downtime

Downtime describes instances and durations of non-productive, idle, or stop time for a machine

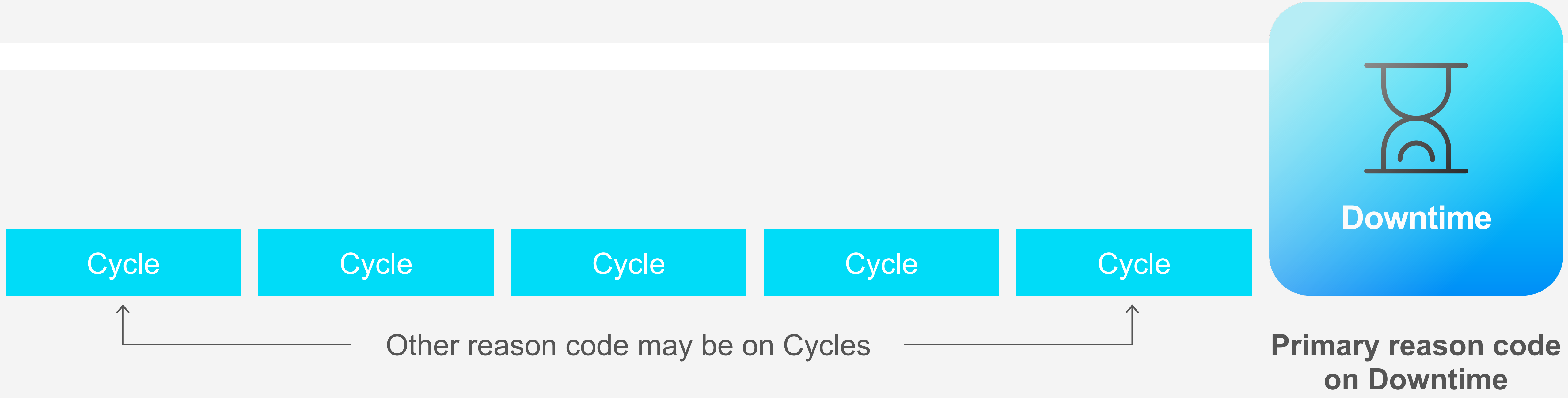
Downtime Data Model

Track when a machine is down

- Start and end of downtime
- Reason code for downtime

Notes

- Split on shift, day boundaries, and reason code changes
- Commonly contains type, category, and reason code



Part and Batch

Part

Part provides traceability across production processes, facilities, and supply chains, and associates the process data with quality outcomes

Batch

Batch represent raw material, output grouping, and summary data, all of which are associated with multiple parts

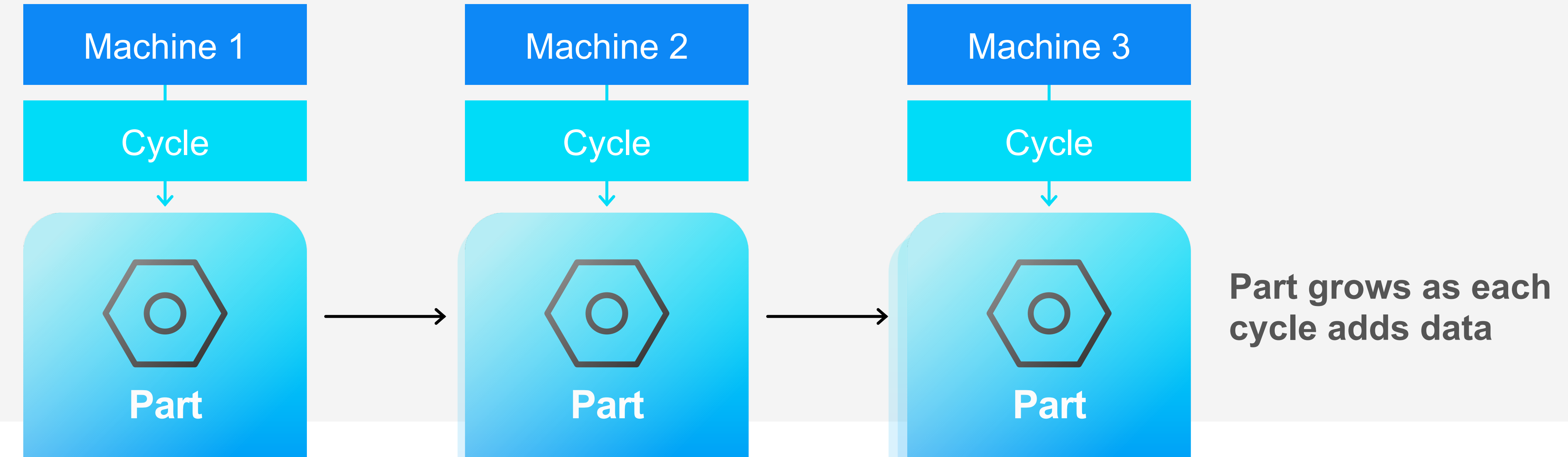
Part and Batch Data Model

Represents a Part/Batch as it moves down the line

- Used for product traceability, identified by serial, or sometimes synthetic timestamp “serial”
- Will have multiple machines’ data

Contains

- Cycles for all machines a part will/could touch
- Start time is from the first machine, end time is the most recent machine — across many facilities as needed



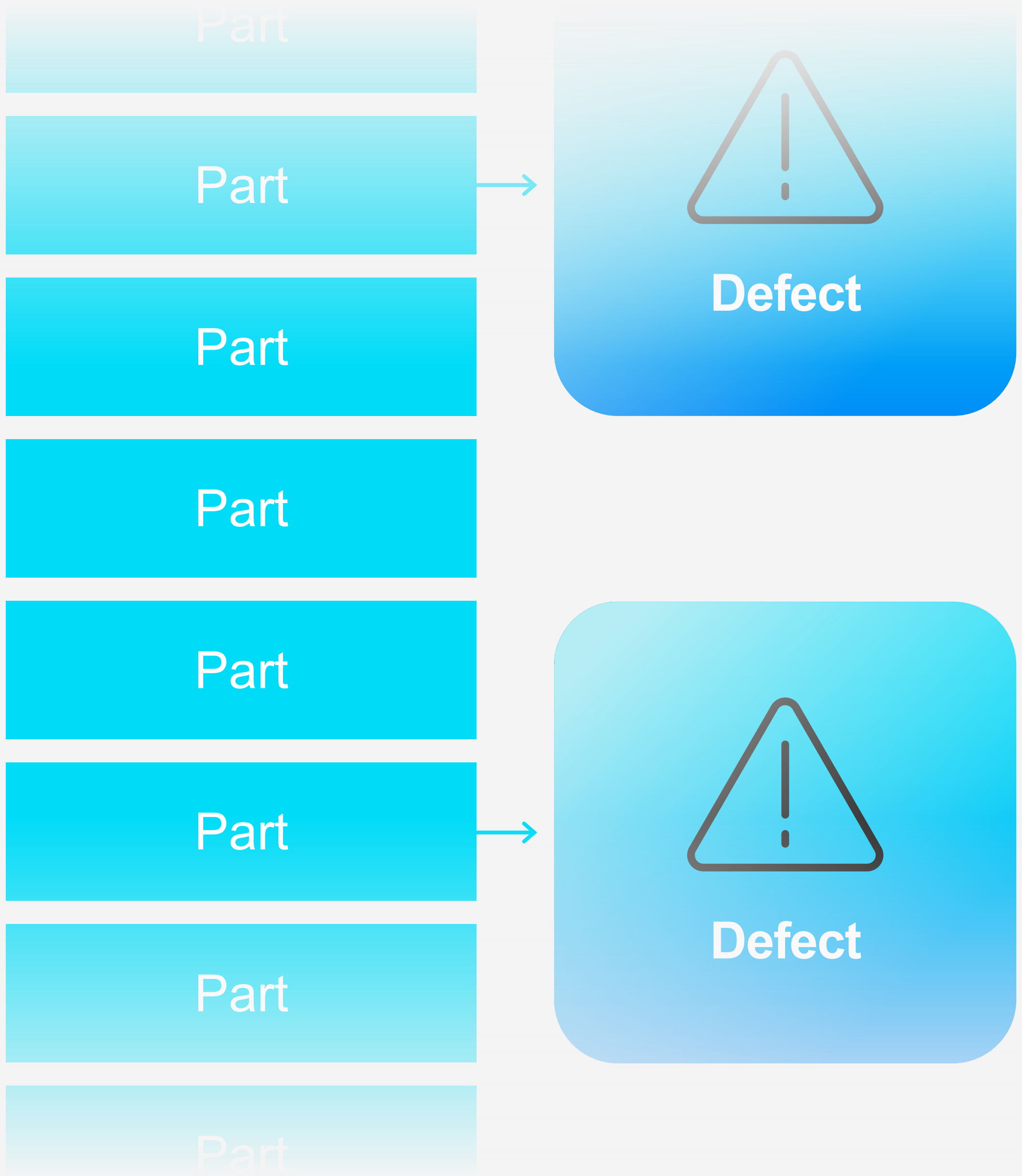
Defect

Defect represents non-conformant production output, in both single parts and batches

Defect Data Model

Track defect-specific information, associated with parts or batches

- Defect test time
- Category, type, and quantity



Configuring Common Data Models Into Plants, Lines, Machines, KPIs

SIGHT MACHINE

Facility

Facility defines the location, time zone, and shift schedule unique to each facility to determine when machines should be running, overall facility-wide KPIs, and shift performance analysis

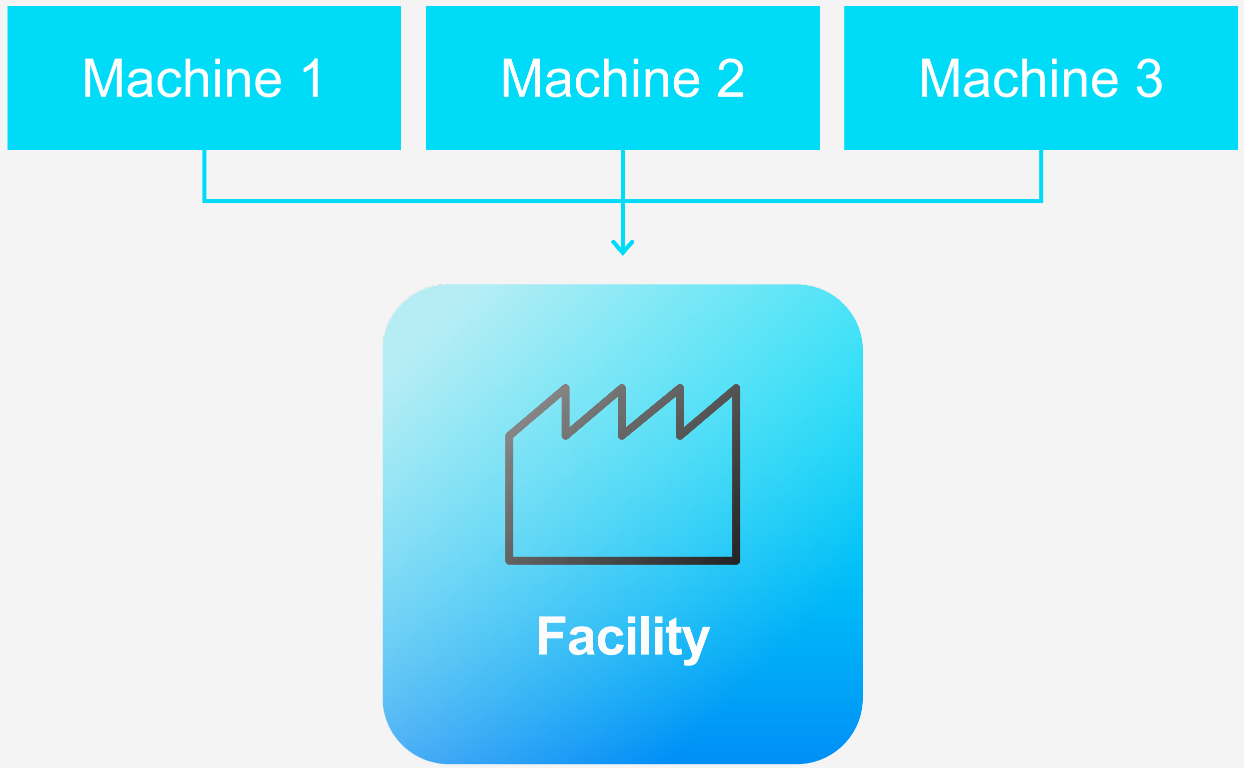
Example Facility

Facility

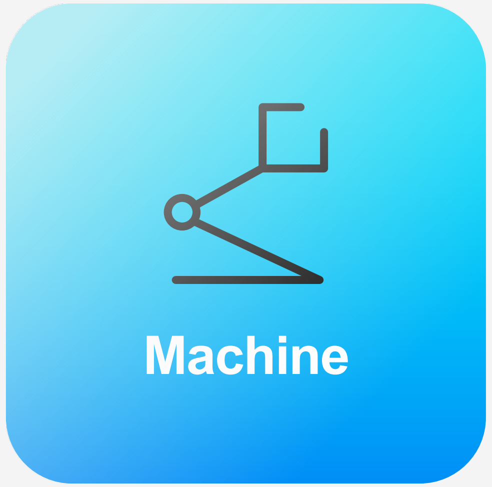
- Facility location (TZ-aware)
- Shift schedule
- Machine assignment

Configuration

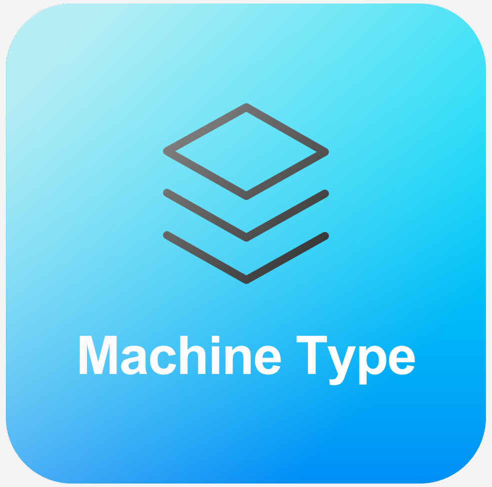

```
"factory_id": "Discrete_P2_F2",
"factory_partner": "Discrete_P2",
"factory_location": "Discrete_F2",
"factory_location_clean": "Ann Arbor Facility",
"geo_location": {
  "lat": 37.799271,
  "lng": -122.4019282
},
"place_name": "221 N Main St, Ann Arbor, MI 48104, USA",
"shift_events": [
  {
    "starttime": "2000-01-01 07:00:00.000000",
    "endtime": "2000-01-01 19:00:00.000000",
    "rrule": {
      "freq": "WEEKLY",
      "byday": [
        "SU",
        "MO",
        "TU",
        "WE",
        "TH",
        "FR",
        "SA"
      ]
    },
    "exdates": [],
    "shiftid": "shift_1558055951922",
```

Machine and Machine Type



Machine is an instantiation of a machine type mapped to a facility location and shift schedule



Machine Type represents several of the exact same machine that can be modeled with the same schema (same sensors, etc.)

Example Machine Type

Machine Type

- Data schema and tag metadata:
  - Internal name
  - UI naming
  - Function used to create
  - Output format

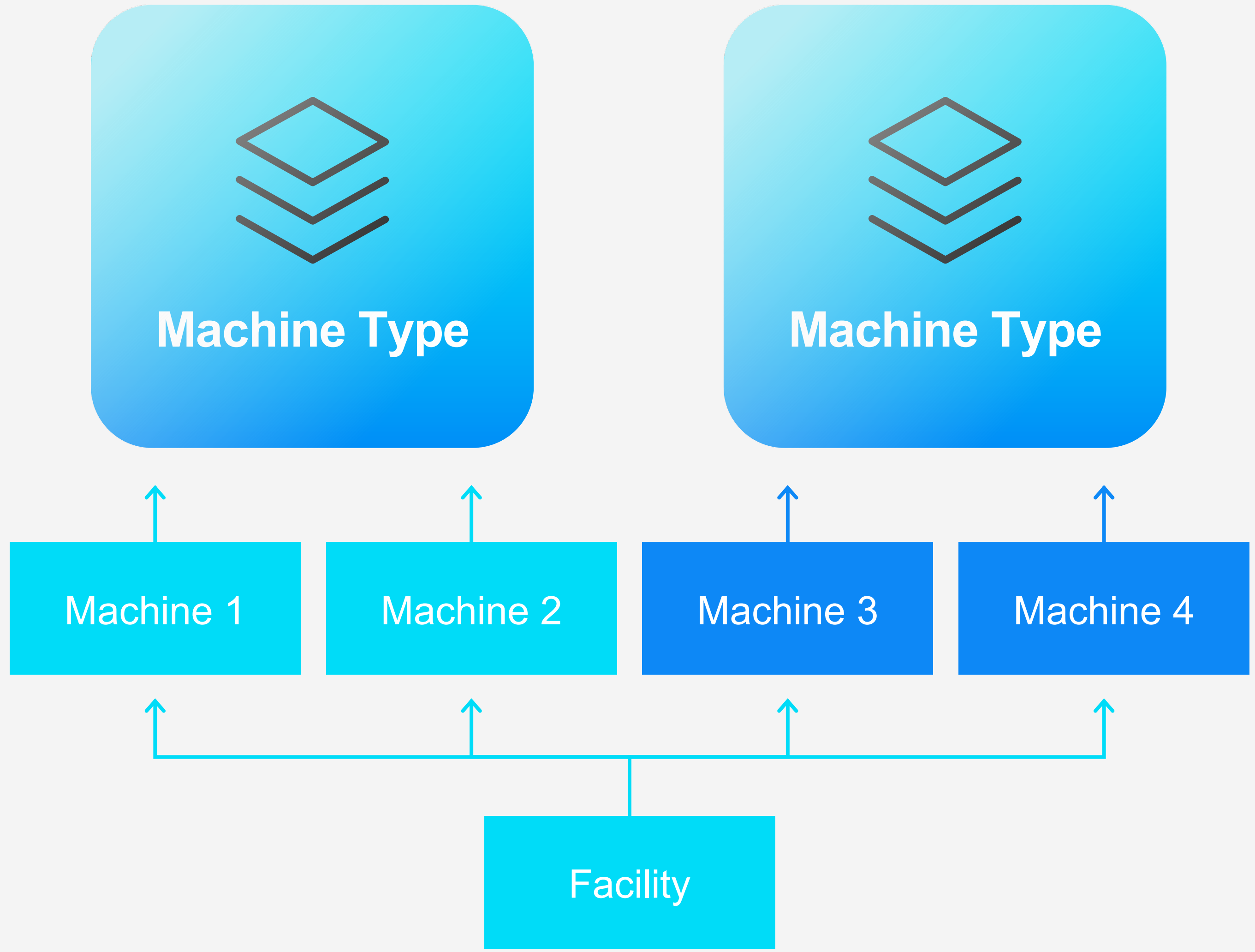
Configuration


```
{
  "analytics": {
    "columns": [
      {
        "name": "stats__BM_Axial_Load__val",
        "source_field": "val",
        "stat_type": "Continuous",
        "type": "float"
      }
    ]
  },
  "display": {
    "title_prefix": "BM: Axial Load",
    "ui_hidden": false
  },
  "func": "last",
  "in_timeline": false,
  "round": 2,
  "sslog_types": [
    "qs"
  ],
  "title": "BM_Axial_Load",
  "unit": "",
  "var": "Axial_Load_rl",
  "variance": 0
},
{
```

Example: Linking Machines to Facilities and shifts

Link individual asset on the plant floor to the Machine Type

- Assign Machine to Facility and shift schedule



Line

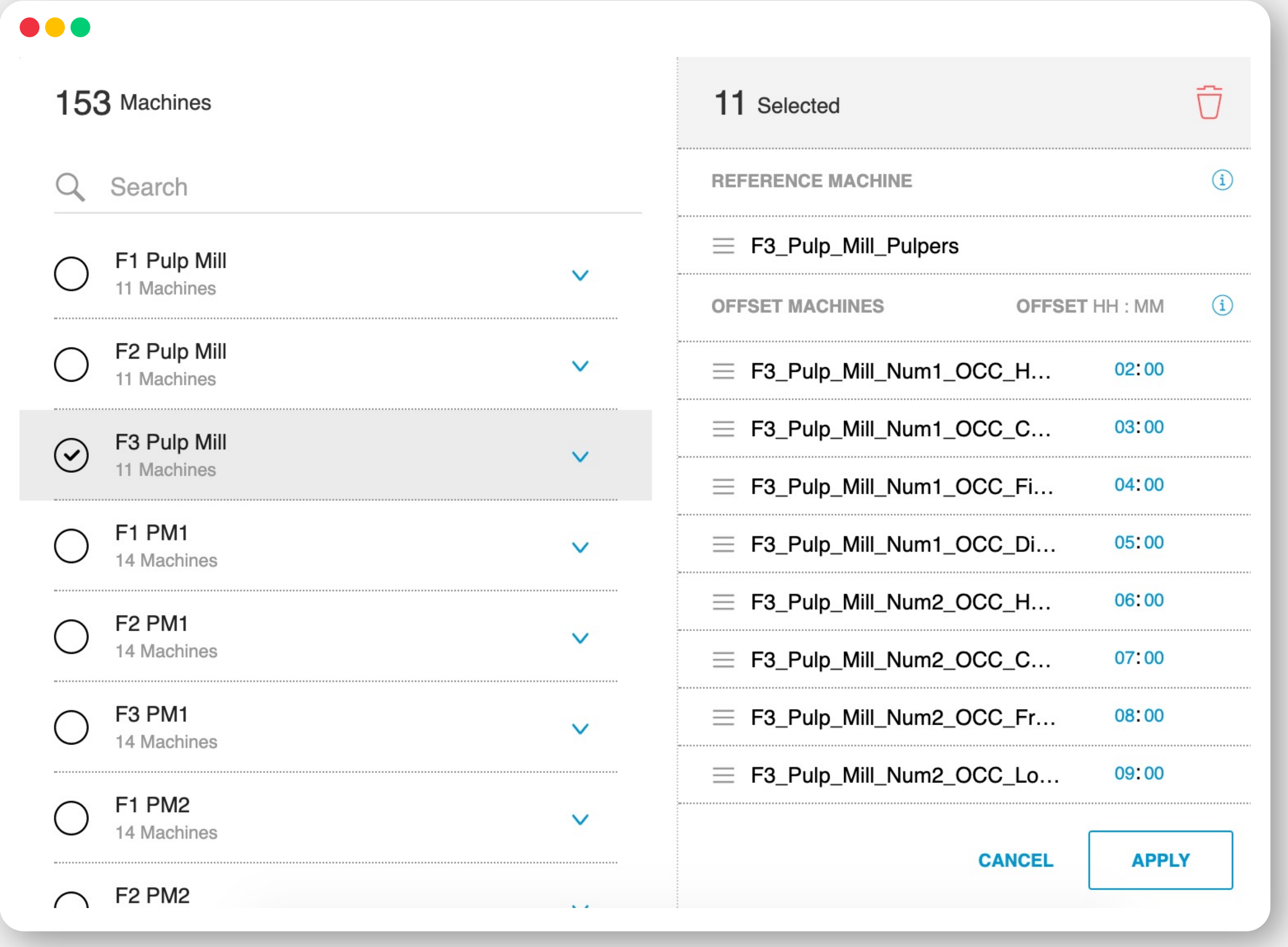
Line defines the layout and sequence of a series of machines involved in the production process and allows for functionality like bottleneck detection, overall process OEE, traceability, and cross-asset analysis

Example Line

Configured line topology


```
{
  "obj": {
    "factory_location": "Atlanta Facility",
    "factory_partner": "Continuous_P1",
    "id": "9741d96e2d68d3ddd6e1d98e",
    "line_topology": {
      "F1_Paper_Mill_Line_PM1": {},
      "F1_Paper_Mill_PM1_Broke": {
        "F1_Paper_Mill_PM1_White_Water": {}
      },
      "F1_Paper_Mill_PM1_Calender_Stacks": {
        "F1_Paper_Mill_PM1_Reel": {}
      },
      "F1_Paper_Mill_PM1_Dryer_Section": {
        "F1_Paper_Mill_PM1_Calender_Stacks": {}
      },
      "F1_Paper_Mill_PM1_Forming_Section": {
        "F1_Paper_Mill_PM1_Press_Section": {}
      },
      "F1_Paper_Mill_PM1_Headbox": {
        "F1_Paper_Mill_PM1_Forming_Section": {}
      },
      "F1_Paper_Mill_PM1_Lab_Tests": {
        "F1_Paper_Mill_PM1_Production_Status": {}
      },
      "F1_Paper_Mill_PM1_Press_Section": {
        "F1_Paper_Mill_PM1_Broke": {}
      },
```


Time-based offsets to analyze machine interactions



KPI

KPI represents key performance indicators unique to each facility’s process and goals. KPIs are formulas calculated dynamically at runtime

Example KPI

Formula configuration

- Dynamic run-time calculation


```
[
  {
    "obj": {
      "dependencies": [
        {
          "aggregate": "sum",
          "name": "gross_cycle_time"
        },
        {
          "aggregate": "sum",
          "name": "net_cycle_time"
        }
      ],
      "display_name": "Availability",
      "formula": "((net_cycle_time)/(gross_cycle_time)*100 if gross_cycle_time>0 else None)",
      "id": "f86e11dabd1a00112444ce2e",
      "model": "line",
      "name": "availability"
    },
    "type": "KPI"
  },
```

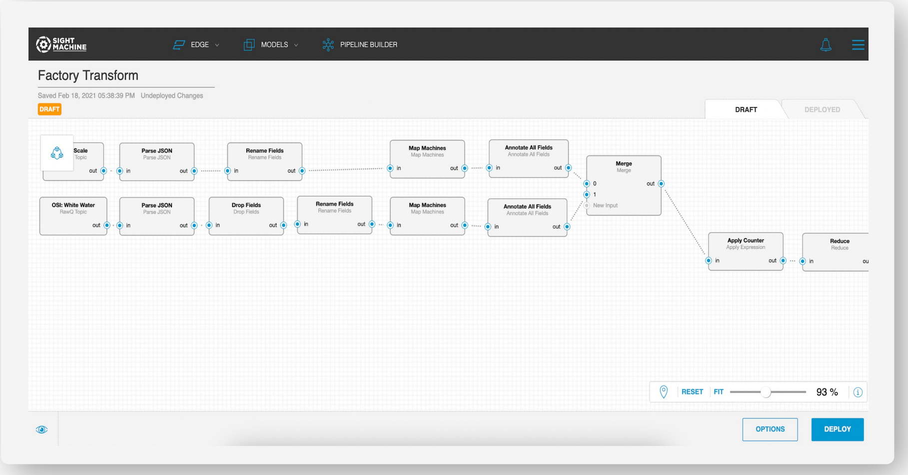
Real-time Applications

SIGHT MACHINE

To drive outcomes, manufacturers require real-time insight

For real-time analysis, stream-processing data into Common Data Models is the ideal approach

- Sight Machine's **Pipeline as a Service** is a configurable stream-processing product that continuously transforms operational data into Data Foundation. The screen at the right is one of our browser-based tools for configuring pipelines

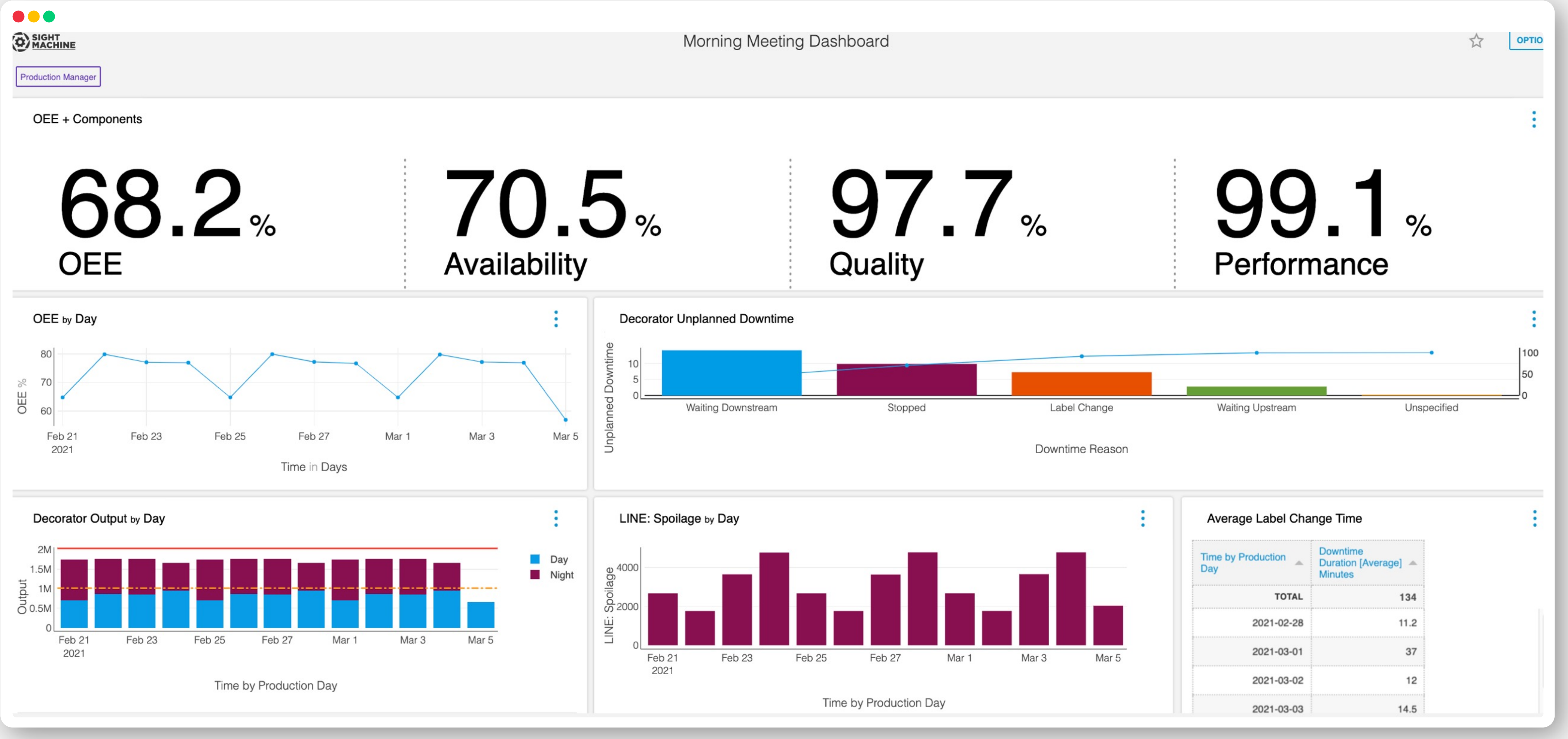


- Factories frequently add and change data. In this environment, robust pipeline management tools are essential. Sight Machine has management tools for modifying data sources, systematically changing configurations across pipelines, dynamically provisioning cloud services, and generating new analytics
- With a standardized Data Foundation, a wide array of analytics can be applied from visualization and KPIs to data science and AI/ML
- Data Foundation can be supplied to other systems (through API and SDK layers) such as Supply Chain Control Towers, CMMS, S&OP, etc.

Visibility and KPIs

Data Foundation supports continuous visibility into operations. Data can be visualized and analyzed through Sight Machine and other leading applications such as Power BI, Tableau, Looker

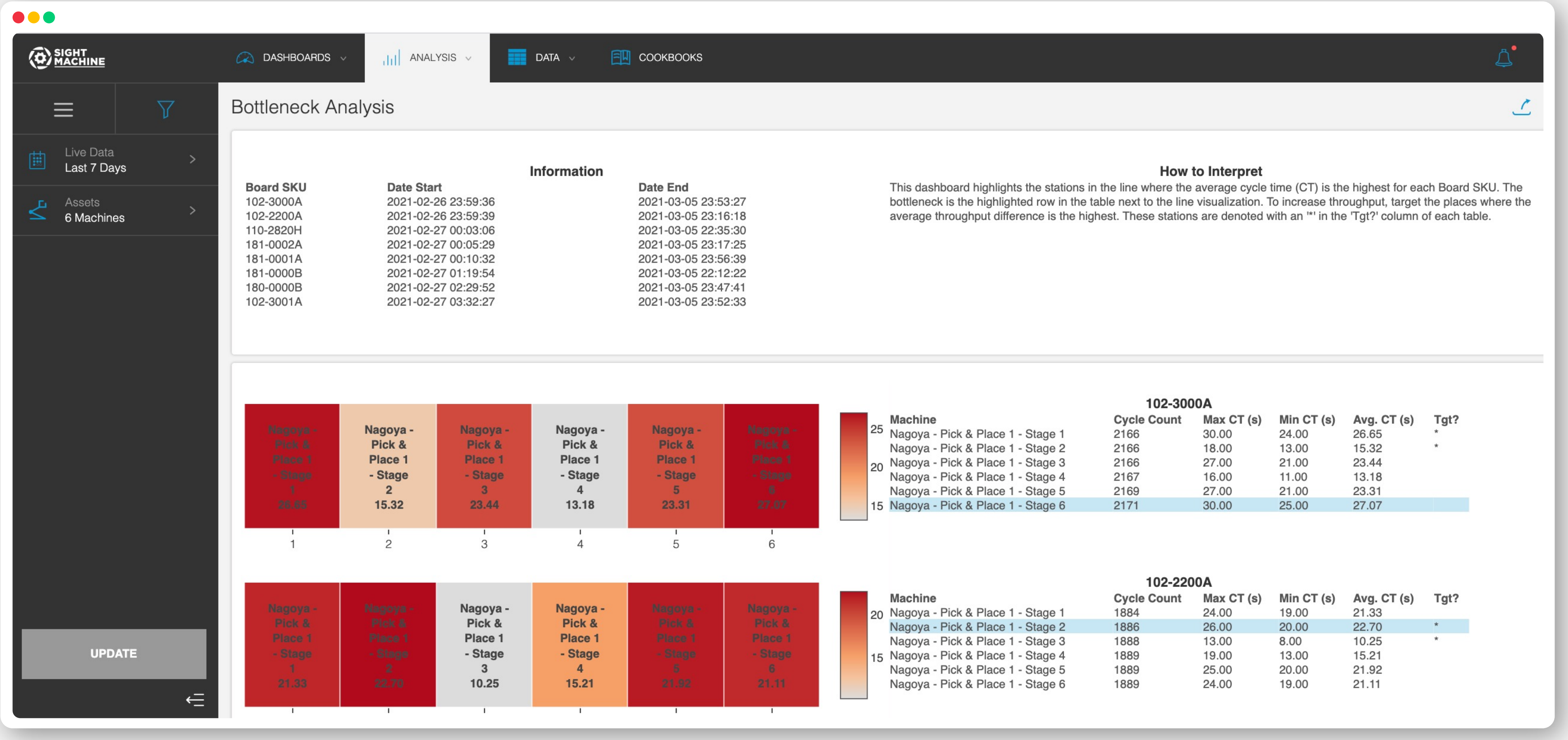
An example of a series of visualizations and analytics used to run a plant



Data Foundation supports advanced analysis

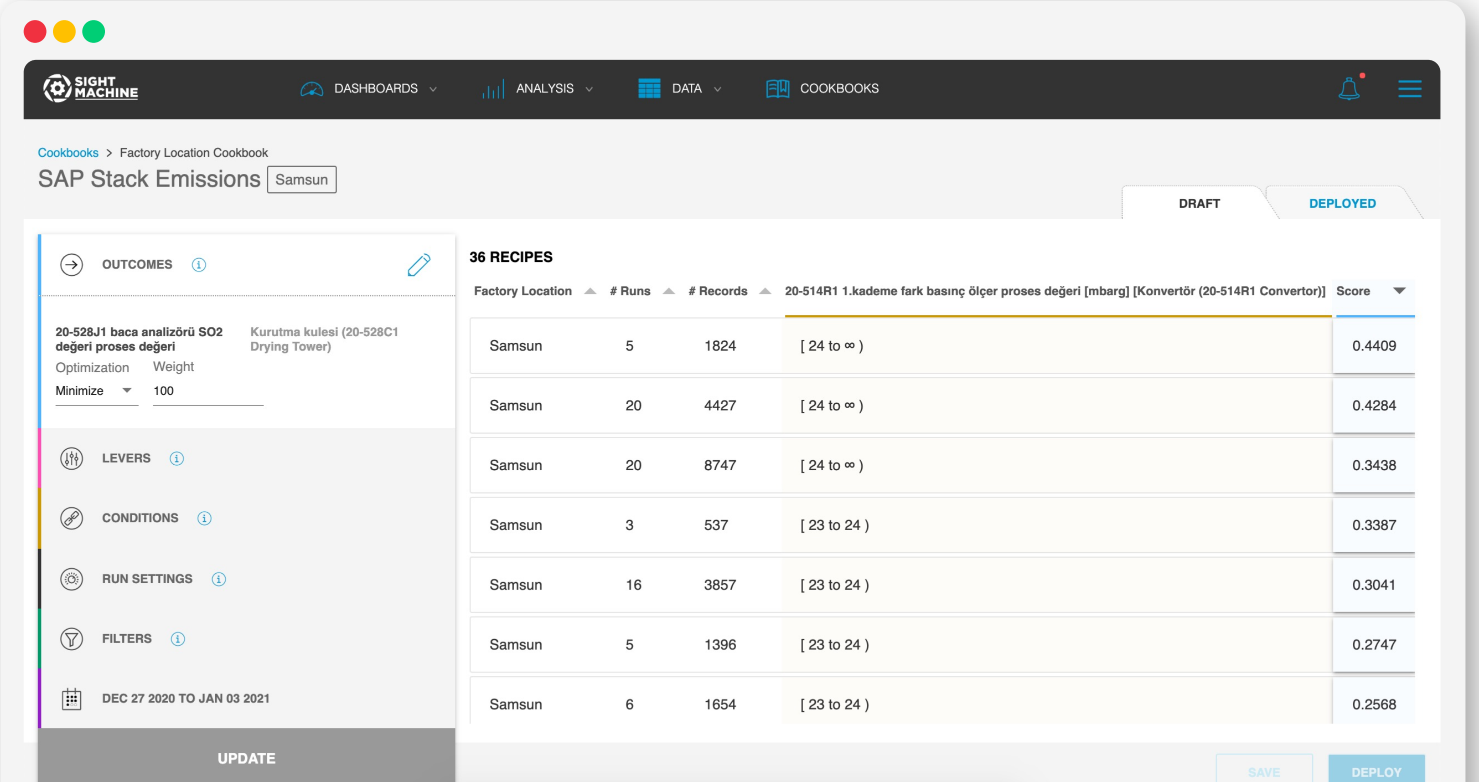
Sight Machine has built dozens of analytics into its platform to query Data Foundation

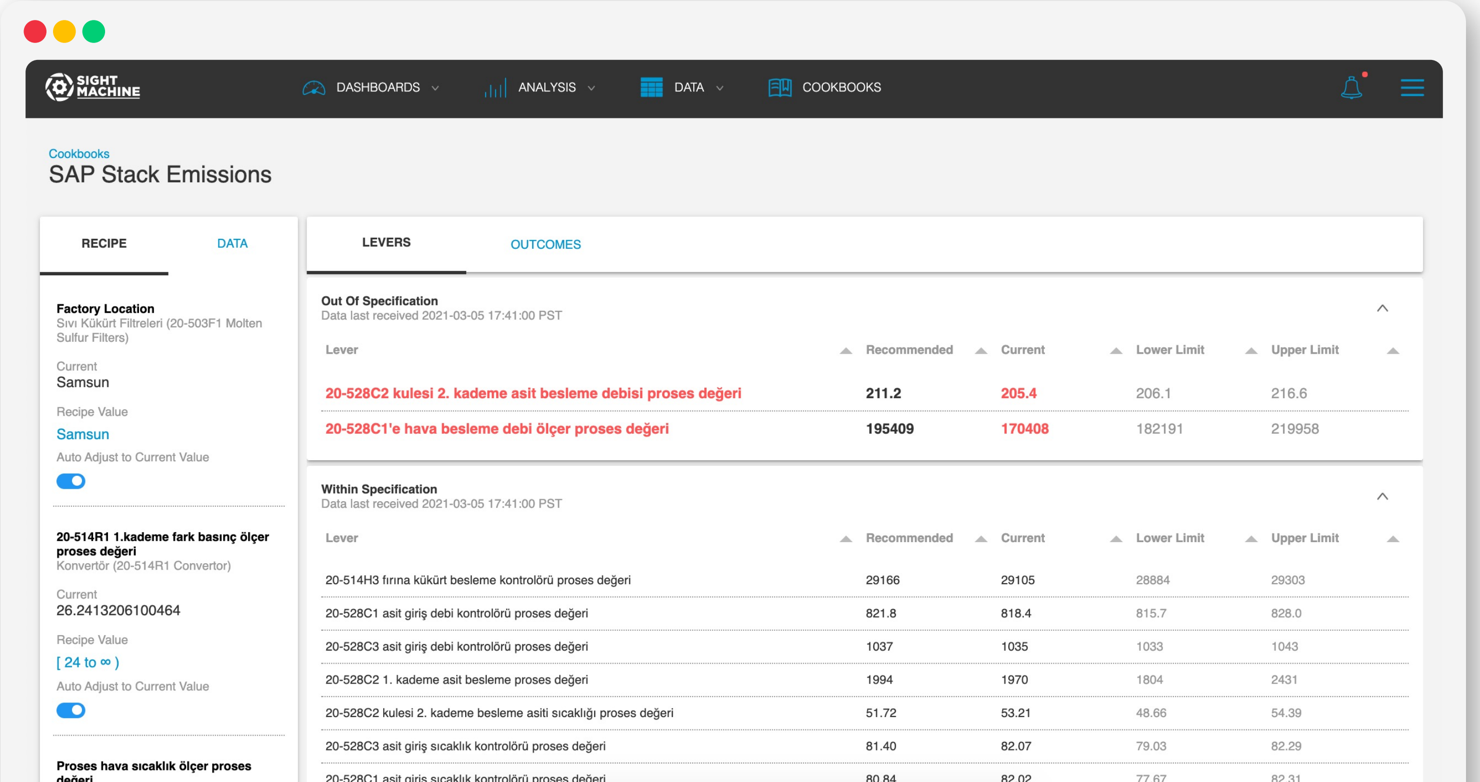
Example: Line Productivity This “bottleneck detector” shows by SKU where problems arise in multiple lines. Takt time for each step is updated and shown in a heatmap of blocked and starved steps. The analytic uses machine and line models to continuously optimize production. The client achieved a 15% efficiency lift at one of its best plants



Dynamic Recipes and Operator Co-Pilot: Continuous guidance for optimization

- **Dynamic Recipes** are another example of advanced analysis. This analytic sets optimal settings for complex arrays of assets by analyzing all previous production runs and optimizing based on goals set by the process engineer. The **Operator Co-Pilot** advises the engineer of any variations from ideal settings
- **Example: Energy and Emissions** The screens below show a real-time Dynamic Recipe for minimizing SO_2 emissions in a chemicals process at a given temperature, and a Co-Pilot advising of variations that need to be corrected





AI/ML

Sight Machine has applied many AI/ML techniques to Data Foundation

Example: Dynamic Optimization of Assets Milk supplies and product mix are ever-changing in dairies. How best to use critical assets, like pasteurizers? This model uses genetic algorithms to schedule optimal asset use. The model adjusts every 30 minutes to reflect dynamic changes in supply and demand

Manufacturing Data Foundation provides abundant opportunities for AI/ML models
Representative AI/ML use cases are listed here

Architecture

SIGHT MACHINE

Sight Machine Architecture

Each layer is open and modular: transformed Data supports 3rd party products and services

Connect

Securely ingest all plant data

Transform

Automate data transformation

Analyze

Analyze transformed data with 1st or 3rd party apps

Build

Share transformed data with other applications

Results

Make data useful for all stakeholders

Manufacturing-specific data transformations (~50 per stream)

OT Data is now transformed and IT-ready

- ✓ Operations Leadership
- ✓ Process Engineers
- ✓ Operators
- ✓ IT Managers
- ✓ Data Scientists
- ✓ Supply Chain
- ✓ Sustainability
- ✓ Connected Machines Sales Teams
- ✓ Planning

Thank you

SIGHT MACHINE