

Table of contents

1	Welcome to PigStat	3
2	What Is PigStat?	3
3	How PigStat Uses Your Data	4
3.1	How PigStat Builds Intelligence	5
3.2	A Simple Example	5
4	PigStat Standard Data Model	6
5	Reporting Period Philosophy	6
5.1	PigStat Is Period-Aware	6
5.2	Supported Reporting Structures	7
5.3	Periods or Weeks	7
6	Supported Reporting Systems	7
7	Minimum Data Requirements	8
8	Historical Data Requirements	9
9	Core Datasets	9
9.1	Farrowing & Weaning Dataset	9
9.2	Mortality Dataset	10
9.3	Sow Dataset	10
9.4	Post-Weaning Dataset	11
9.5	Future Datasets	11
9.5.1	Dry Sow Intelligence (Coming Soon)	11
9.5.2	Finisher Intelligence (Coming Soon)	12
10	Data Quality Guidelines	12
10.1	Recommended Practices	12
10.1.1	Use Consistent Identifiers	12
10.1.2	Keep Reporting Periods Consistent	13
10.1.3	Use Consistent Date Formats	13
10.1.4	Record Missing Values Consistently	13
10.1.5	Maintain Historical Records	13
11	Common Data Issues	14
11.1	Duplicate Identifiers	14
11.2	Multiple Names for the Same Animal	14
11.3	Inconsistent Room Names	14

11.4	Ambiguous Variable Names	15
11.5	Mixed Levels of Aggregation	15
12	Data Readiness Assessment	16
13	Farm-Specific Adaptation	16
14	Available Modules	17
14.1	Available Today	17
14.1.1	Overview & KPIs	17
14.1.2	Reproductive Performance	17
14.1.3	Maternal Performance	18
14.1.4	Teat Capacity Analysis	18
14.1.5	Easy Grow Intelligence	18
14.1.6	Mortality Analysis	19
14.1.7	Mortality Intelligence	19
14.1.8	Supported Data Levels	19
14.1.9	Recommended Data Source	19
14.1.10	Example Mortality Event Records	20
14.1.11	How PigStat Builds Mortality Intelligence	20
14.1.12	Why This Matters	20
14.1.13	Post-Weaning Intelligence	21
14.1.14	Historical Trends	21
14.1.15	Data Quality & Coverage	21
14.2	In Development	22
14.2.1	Sow Intelligence	22
14.3	Coming Soon	22
14.3.1	Dry Sow Intelligence	22
14.3.2	Finisher Intelligence	22
15	Module Availability	23
16	Next Steps	23
16.1	Step 1 – Share Sample Data	23
16.2	Step 2 – Data Structure Review	24
16.3	Step 3 – Mapping & Alignment	24
16.4	Step 4 – Data Readiness Assessment	24
16.5	Step 5 – Initial Report Generation	25
16.6	Step 6 – Ongoing Reporting	25
16.7	Start Simple	25
17	Appendix A – CSV Header Examples	25
17.1	Example: Farrowing & Weaning Dataset	26
17.2	Example: Mortality Dataset	26

17.3 Example: Sow Dataset	26
17.4 Example: Post-Weaning Dataset	26
18 Appendix B – CSV Templates	27
19 Appendix C – Frequently Asked Questions	27
19.1 Do I need to rename my columns?	27
19.2 Do I need to use weekly reporting?	27
19.3 Can PigStat work with Excel files?	28
19.4 What if some variables are missing?	28
19.5 Do I need historical data?	28
19.6 What if my farm software uses different terminology?	28
19.7 Does PigStat support custom datasets?	29
19.8 What happens if a variable cannot be mapped automatically?	29
19.9 Does PigStat require perfect data?	29

1 Welcome to PigStat

Welcome to PigStat.

PigStat is a production intelligence platform designed for pig farms, consultants and genetics companies.

The platform transforms routine production records into actionable insights covering reproductive performance, growth, mortality, sow performance and long-term production trends.

This document explains how PigStat uses farm data, which datasets are recommended, and how different farm structures can be integrated into the PigStat analytics engine.

2 What Is PigStat?

PigStat is designed to help producers move beyond simple record keeping and toward data-driven decision making.

The platform combines production records with statistical analysis to identify patterns, trends and opportunities that may not be visible through routine reporting alone.

PigStat is intended to support practical decisions such as:

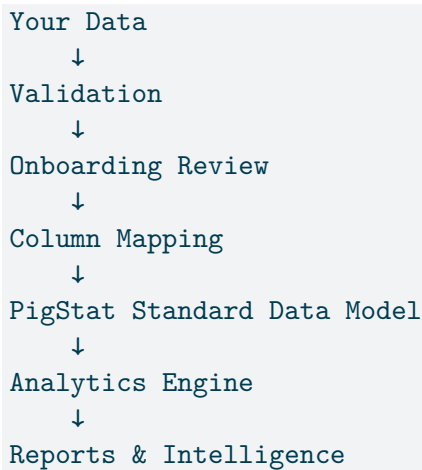
- Monitoring reproductive performance
- Understanding mortality patterns

- Evaluating sow performance
- Assessing growth and weaning outcomes
- Comparing production periods
- Identifying emerging risks and opportunities

The goal is not simply to generate reports, but to transform farm data into production intelligence.

3 How PigStat Uses Your Data

PigStat follows a structured process to transform farm records into analytics and reporting outputs.



PigStat adapts to different farm software systems and spreadsheet structures through an onboarding review and mapping process.

Farms do not need to rename their columns before starting, but PigStat needs to understand the structure of the files being provided.

In many cases, the mapping can be prepared from the file headers alone. For more complex datasets, a short alignment process may be required to confirm definitions, units, reporting periods and farm-specific conventions.

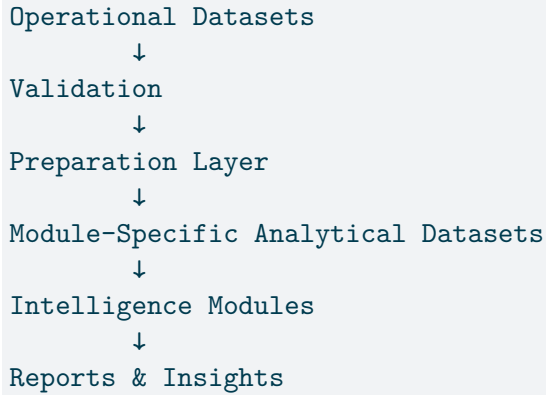
PigStat may suggest mappings automatically, but final alignment may require confirmation to ensure that farm-specific definitions are interpreted correctly.

No changes to the farm's data collection process are required.

3.1 How PigStat Builds Intelligence

PigStat does not require all variables to be stored in a single file.

Instead, the platform combines information from one or more operational datasets and creates module-specific analytical datasets used by the intelligence modules.



3.2 A Simple Example

A farm may record the same information using different column names.

For example:

Farm Column	Meaning
sow_tag	Sow identifier
par_no	Parity
liveborn	Live-born piglets
wean_count	Piglets weaned

PigStat simply maps these fields into its standard internal model:

Farm Column	PigStat Field
sow tag	Sow_ID
par_no	Parity
liveborn	Born_Alive
wean_count	Weaned

Once the mapping has been defined, the farm can continue using its existing files and naming conventions.

No changes to the farm's data collection process are required.

For many farms, the onboarding process is simply a matter of reviewing the file headers and confirming the meaning of a small number of columns.

4 PigStat Standard Data Model

PigStat uses a standardized internal data model.

This model allows data originating from different farms, software systems and reporting structures to be analyzed consistently.

The internal model provides:

- Consistent calculations
- Comparable reporting periods
- Standardized performance metrics
- Reliable historical trend analysis
- Scalable intelligence modules

The PigStat Data Dictionary provides a detailed description of all standard fields currently supported by the platform.

5 Reporting Period Philosophy

5.1 PigStat Is Period-Aware

PigStat is designed around production events and reporting periods rather than calendar assumptions.

The analytics engine does not depend on weeks, months or quarters. Instead, it operates on comparable production periods.

A reporting period may represent:

- Weekly production
- Two-week batches
- Three-week batches
- Four-week batches

- Monthly production cycles
- Quarterly production cycles
- Other consistently defined production periods

What matters is not how a period is named.

What matters is that periods are comparable over time.

5.2 Supported Reporting Structures

Examples of valid reporting periods include:

- Week 23
- Week 24
- P2026-B04
- P2026-T07
- P2026-M03
- Q2-2026

PigStat compares equivalent production periods regardless of the naming convention used by the farm.

PigStat is designed around biological and production events, not calendar assumptions.

5.3 Periods or Weeks

What matters is not how a period is named. What matters is that periods are comparable over time.

6 Supported Reporting Systems

PigStat supports a wide range of production systems and reporting frequencies.

The platform is designed around reporting periods rather than fixed calendar assumptions.

Examples of supported systems include:

Reporting System	Supported
Weekly	Yes
2-Week Batch	Yes
3-Week Batch	Yes
4-Week Batch	Yes

Reporting System	Supported
Monthly	Yes
Quarterly	Yes
Custom Production Cycles	Yes

As long as production periods are defined consistently, PigStat can compare performance across time and generate meaningful analytics.

7 Minimum Data Requirements

PigStat can generate useful reports with a relatively small number of variables.

The minimum recommended dataset includes:

Field
Sow_ID
Period
Parity
Born_Alive
Weaned

With these fields alone, PigStat can already generate:

- Core production KPIs
- Reproductive performance summaries
- Historical trend analysis
- Basic benchmarking across reporting periods

Additional variables unlock additional intelligence modules and more advanced analyses.

8 Historical Data Requirements

PigStat performs best when historical data is available.

Recommended minimum history:

Reporting System	Recommended History
Weekly	12–24 periods
2-Week Batch	12 periods
3-Week Batch	12 periods
4-Week Batch	12 periods
Monthly	12 periods
Quarterly	8–12 periods

PigStat can still operate with less history, but trend analysis and benchmarking become more reliable as additional periods are collected.

9 Core Datasets

PigStat is designed to work with multiple datasets. Not all datasets are required.

Additional datasets unlock additional modules and intelligence layers.

9.1 Farrowing & Weaning Dataset

Status: Core Dataset

Recommended structure:

One row per litter.

Typical variables include:

- Sow_ID
- Period
- Parity
- Born_Alive
- Born_Dead
- Mummified

- Weaned
- Average_Weaning_Weight
- Days_Lactation

This dataset powers most core PigStat reporting modules.

9.2 Mortality Dataset

Status: Recommended

Recommended structure:

One row per mortality event.

Typical variables include:

- Piglet_ID
- Sow_ID
- Death_Date
- Age_Day
- Cause_Code

This dataset enables Mortality Analysis and Mortality Intelligence.

9.3 Sow Dataset

Status: Recommended

Recommended structure:

One row per sow.

Typical variables include:

- Sow_ID
- Parity
- Functional_Teats
- Health Flags
- Performance Indicators

This dataset supports advanced sow-level analyses and future Sow Intelligence modules.

9.4 Post-Weaning Dataset

Status: Optional

Recommended structure:

One row per batch, room or production group.

Typical variables include:

- Period
- Group_ID
- Number_Weaned
- Post_Weaning_Deaths
- Average_Weight

This dataset enables Post-Weaning Intelligence and nursery performance analysis.

9.5 Future Datasets

The following modules are currently under development.

9.5.1 Dry Sow Intelligence (Coming Soon)

Future datasets may include:

- Body Condition Score
- Service Date
- Pregnancy Status
- Feed Allocation
- Housing Group

9.5.2 Finisher Intelligence (Coming Soon)

Future datasets may include:

- Entry Date
- Exit Date
- Sale Weight
- Daily Gain
- Feed Intake
- Feed Conversion Ratio
- Mortality
- Carcass Quality

10 Data Quality Guidelines

PigStat is designed to work with real-world farm data.

Perfect datasets are not required. However, a few simple practices can significantly improve data quality and the value of the resulting analyses.

10.1 Recommended Practices

10.1.1 Use Consistent Identifiers

Whenever possible, use consistent identifiers for sows, litters and piglets.

Examples:

S1001

S1002

S1003

Avoid changing identifiers between reporting periods.

10.1.2 Keep Reporting Periods Consistent

Reporting periods should follow a consistent structure over time.

Examples:

Weekly reporting

3-week batch reporting

Monthly reporting

Avoid mixing different reporting systems within the same historical dataset whenever possible.

10.1.3 Use Consistent Date Formats

Recommended formats:

2026-05-14

14-May-2026

Avoid mixing multiple date formats within the same file.

10.1.4 Record Missing Values Consistently

Missing information is normal.

It is generally better to leave a value blank than to enter placeholder values that may be interpreted incorrectly.

10.1.5 Maintain Historical Records

Historical data provides context.

The more comparable reporting periods available, the more reliable trend analysis and benchmarking become.

11 Common Data Issues

The following issues are frequently encountered during onboarding.

11.1 Duplicate Identifiers

Example:

```
S1001  
S1001  
S1001
```

Duplicates may be valid in litter-level datasets but should be reviewed when uniqueness is expected.

11.2 Multiple Names for the Same Animal

Example:

```
S1001  
1001  
001001
```

These may represent the same sow.

PigStat can often resolve such cases during onboarding, but consistency is preferred.

11.3 Inconsistent Room Names

Example:

```
Room A  
Room_A  
A  
North Wing
```

These may all refer to the same location.

Consistent naming improves reporting accuracy.

11.4 Ambiguous Variable Names

Examples:

```
Deaths  
Losses  
Weight
```

Such names may require clarification during onboarding.

More descriptive names improve automatic mapping and reduce interpretation errors.

11.5 Mixed Levels of Aggregation

Example:

One file contains:

- One row per sow

Another file contains:

- One row per piglet

Both structures can be useful, but PigStat must understand the level of aggregation represented by each dataset.

12 Data Readiness Assessment

Before analytics begin, PigStat performs a Data Readiness Assessment.

The objective is to understand:

- Which datasets are available
- Which variables are available
- Which modules can be activated
- Whether any clarification or mapping is required

Typical outputs include:

Recognized variables

Suggested mappings

Missing recommended variables

Data quality observations

Available intelligence modules

The assessment is intended to simplify onboarding and ensure that analyses are based on a correct understanding of the farm's data structure.

13 Farm-Specific Adaptation

Every farm collects data differently.

Different software systems, management styles and reporting practices often result in different file structures.

This is expected.

PigStat adapts through an onboarding review and mapping process.

The objective is not to make farms adapt to PigStat.

The objective is to make PigStat understand the farm's data.

In many cases, this process can be completed using only sample files and column headers.

More complex datasets may require a short alignment process to confirm definitions, units and farm-specific conventions.

14 Available Modules

PigStat is organized into modular intelligence components.

Not all modules require the same data. Additional datasets unlock additional analytical capabilities.

14.1 Available Today

14.1.1 Overview & KPIs

Provides a high-level summary of production performance.

Typical outputs include:

- Weaned per litter
 - Average weaning weight
 - Mortality indicators
 - Capacity utilization
 - Historical comparisons
-

14.1.2 Reproductive Performance

Analyzes farrowing performance and reproductive outcomes.

Typical outputs include:

- Born alive
 - Stillbirths
 - Mummified fetuses
 - Reproductive trends
 - Historical benchmarking
-

14.1.3 Maternal Performance

Evaluates sow and litter performance during lactation.

Typical outputs include:

- Lactation outcomes
 - Fostering activity
 - Litter management indicators
 - Maternal performance trends
-

14.1.4 Teat Capacity Analysis

Assesses the relationship between litter size and nursing capacity.

Typical outputs include:

- Functional teat utilization
 - Weaned per teat
 - Capacity efficiency indicators
-

14.1.5 Easy Grow Intelligence

Evaluates litter growth performance using PigStat growth indicators.

Typical outputs include:

- Easy Grow Score
 - Growth consistency
 - Growth benchmarking
 - Identification of high and low performing litters
-

14.1.6 Mortality Analysis

Provides descriptive mortality reporting.

Typical outputs include:

- Mortality rates
 - Mortality trends
 - Cause distributions
 - Mortality benchmarking
-

14.1.7 Mortality Intelligence

Applies statistical methods to investigate factors associated with mortality and to identify operational areas where management changes may have the greatest impact.

Typical outputs include:

- Mortality hotspots
- Birth weight signals
- Parity signals
- Management signals
- Adjusted mortality models
- Operational priorities

14.1.8 Supported Data Levels

Level	Description	Supported
Basic	Weekly mortality summaries	
Enhanced	Mortality grouped by room, parity or stage	
Advanced	Individual piglet mortality records	

14.1.9 Recommended Data Source

Many farms already record each mortality event individually.

14.1.10 Example Mortality Event Records

```
Date,Piglet_ID,Room,Pen,Reason,Note
2026-06-01,F12345,R2,14,Crushed,
2026-06-01,F12346,R2,14,Starvation,
2026-06-02,F12347,R5,8,Scours,
```

PigStat can use these operational records directly.

14.1.11 How PigStat Builds Mortality Intelligence

PigStat does not require all mortality information to be stored in a single file.

Instead, it combines information from multiple operational datasets.

Example:

Farrowing Dataset ↓ Born Alive Parity Birth Weight Room

Mortality Events Dataset ↓ Date Piglet_ID Reason Location

PigStat Preparation Layer ↓ Mortality Intelligence Dataset

This approach allows farms to keep their existing workflows while still benefiting from advanced analytics.

PigStat does not require all variables to be stored in a single CSV file. The platform can combine information from multiple operational datasets to create module-specific analytical datasets.

14.1.12 Why This Matters

Using individual mortality records allows PigStat to identify:

- mortality hotspots
- room-level patterns
- parity-related mortality
- birth weight-related mortality patterns
- seasonal signals
- management-related mortality risk factors
- emerging mortality trends

while still supporting farms that only have aggregated mortality summaries.

Farms with more detailed operational records can unlock deeper intelligence, while farms with simpler datasets can still benefit from mortality monitoring and trend analysis.

14.1.13 Post-Weaning Intelligence

Analyzes performance after weaning.

Typical outputs include:

- Batch mortality
 - Batch stability
 - Growth signals
 - Historical comparisons
-

14.1.14 Historical Trends

Tracks long-term production performance across reporting periods.

Typical outputs include:

- Trend analysis
 - Benchmarking
 - Performance trajectories
-

14.1.15 Data Quality & Coverage

Evaluates dataset completeness and consistency.

Typical outputs include:

- Missing variable assessment
 - Coverage metrics
 - Data quality observations
-

14.2 In Development

14.2.1 Sow Intelligence

A dedicated sow-level intelligence module.

Planned capabilities include:

- Sow performance scoring
 - Nurse sow identification
 - Cull candidate identification
 - Performance alerts
 - Longitudinal sow tracking
-

14.3 Coming Soon

14.3.1 Dry Sow Intelligence

Planned capabilities include:

- Dry sow monitoring
 - Reproductive status tracking
 - Body condition assessment
 - Management signals
-

14.3.2 Finisher Intelligence

Planned capabilities include:

- Growth performance
 - Mortality analysis
 - Feed efficiency
 - Batch comparisons
 - Carcass performance indicators
-

15 Module Availability

The modules available in a report depend on the data provided.

PigStat automatically identifies which modules can be supported by the available datasets and variables.

Additional data unlocks additional intelligence.

Farms can start with a small dataset and expand analytical capabilities over time.

16 Next Steps

Getting started with PigStat is typically a straightforward process.

The exact onboarding workflow may vary depending on the complexity of the available datasets, but most projects follow the steps below.

16.1 Step 1 – Share Sample Data

Provide one or more sample files representing your current data collection system.

Examples may include:

- Farrowing and weaning records
- Mortality records
- Sow records
- Post-weaning records

Sample files are usually sufficient for an initial assessment.

16.2 Step 2 – Data Structure Review

PigStat reviews the structure of the provided files.

This review focuses on:

- Available datasets
- Column headers
- Reporting periods
- Levels of aggregation
- Data completeness

The objective is to understand how the farm currently records and organizes production data.

16.3 Step 3 – Mapping & Alignment

PigStat maps farm-specific variables into the standard internal data model.

In many cases, this process can be completed using only the file headers.

More complex datasets may require a short alignment process to clarify:

- Variable definitions
 - Units of measurement
 - Reporting conventions
 - Farm-specific terminology
-

16.4 Step 4 – Data Readiness Assessment

A Data Readiness Assessment is performed to identify:

- Recognized variables
- Suggested mappings
- Missing recommended variables
- Available intelligence modules

The assessment provides a clear picture of the analytical capabilities supported by the current dataset.

16.5 Step 5 – Initial Report Generation

Once the mapping has been confirmed, PigStat generates an initial report using the available data.

This report serves as a validation step and provides an opportunity to review:

- Calculations
 - Definitions
 - Reporting logic
 - Available modules
-

16.6 Step 6 – Ongoing Reporting

After onboarding, the same mapping configuration can be reused for future reporting cycles.

The farm continues using its existing data collection process while PigStat applies the established mapping and analytics workflow.

16.7 Start Simple

A perfect dataset is not required to begin.

Many farms start with a small number of variables and gradually expand the scope of their reporting over time.

PigStat is designed to grow alongside the farm's data capabilities.

The objective is not to collect more data.

The objective is to extract more value from the data that already exists.

17 Appendix A – CSV Header Examples

The examples below illustrate typical dataset structures supported by PigStat.

Actual farm files may differ. PigStat can adapt to different naming conventions through the onboarding review and mapping process.

17.1 Example: Farrowing & Weaning Dataset

```
Period,Sow_ID,Parity,Born_Alive,Born_Dead,Mummified,Weaned,Average_Weaning_Weight  
P2026-W23,S1001,3,14,1,0,12,7.3
```

One row per litter.

17.2 Example: Mortality Dataset

```
Piglet_ID,Sow_ID,Death_Date,Age_Day,Cause_Code  
P100001,S1001,2026-05-14,3,Starvation
```

One row per mortality event.

17.3 Example: Sow Dataset

```
Sow_ID,Parity,Teats_Functional,Room  
S1001,3,14,Room_A
```

One row per sow.

17.4 Example: Post-Weaning Dataset

```
Period,Group_ID,Number_Weaned,Post_Weaning_Deaths,Average_Weight  
P2026-W23,G01,420,8,13.4
```

One row per production group or batch.

18 Appendix B – CSV Templates

PigStat provides example templates to help farms understand the expected structure of common datasets.

Available templates include:

- weekly__template.csv
- 2week__template.csv
- 3week__template.csv
- 4week__template.csv

These templates are intended as examples only.

Farms do not need to adopt PigStat column names or reporting structures.

The templates illustrate how information is represented within the PigStat standard data model.

19 Appendix C – Frequently Asked Questions

19.1 Do I need to rename my columns?

No.

PigStat can adapt to different column naming conventions through the onboarding review and mapping process.

19.2 Do I need to use weekly reporting?

No.

PigStat is period-aware and supports weekly, biweekly, three-week, four-week, monthly and custom reporting structures.

19.3 Can PigStat work with Excel files?

Yes.

Excel files can be reviewed and converted into compatible datasets during onboarding.

19.4 What if some variables are missing?

This is common.

PigStat can generate useful reports with a relatively small number of variables.

Additional variables unlock additional intelligence modules.

19.5 Do I need historical data?

Historical data is recommended but not required.

PigStat can begin with a small number of reporting periods and expand analytical capabilities as additional data becomes available.

19.6 What if my farm software uses different terminology?

This is expected.

PigStat is designed to work with different software systems, naming conventions and reporting structures.

19.7 Does PigStat support custom datasets?

Yes.

Custom datasets can often be integrated through the onboarding review and mapping process.

19.8 What happens if a variable cannot be mapped automatically?

A short alignment process may be required to clarify the meaning of the variable before reporting begins.

19.9 Does PigStat require perfect data?

No.

PigStat is designed to work with real-world farm data.

The objective is not to create perfect datasets.

The objective is to extract meaningful intelligence from the data already available.