**Use Cases**

*This draft document was provided to stakeholders for feedback on November 28, 2018.*

*After December 7, 2018, please see* [*https://solarforecastarbiter.org/usecases/*](https://solarforecastarbiter.org/usecases/) *for the latest version.*

**Purpose and Summary**

This document describes Solar Forecast Arbiter use cases and their associated requirements, and Solar Forecast Arbiter framework functional capabilities.

The Solar Forecast Arbiter is primarily designed to support the evaluation of solar forecasts that are useful for solar forecast users. It is not a general-purpose weather forecast analysis tool, though it may eventually be extended to incorporate wind power and load forecast analysis.

Use cases are grouped into two categories:

1. [Evaluate forecasts](https://solarforecastarbiter.org/usecases/#evaluatefx). These use cases anticipate a framework user whose primary interest is comparing one or more forecasts to data. Forecast quantities can include irradiance, power or net load, and can vary in lead time, interval and horizon. The framework can evaluate deterministic forecasts, event forecasts, and probabilistic forecasts.
2. [Analyze forecasts](https://solarforecastarbiter.org/usecases/#analyzefx). These use cases anticipate a framework user whose primary interest is investigating correlations between forecasts, forecast errors, and other quantities.

Framework functional capabilities are grouped into three categories:

1. [Perform forecast evaluation in a standard manner](https://solarforecastarbiter.org/usecases/#evaluation)
2. [Administer the framework](https://solarforecastarbiter.org/usecases/#administer)
3. [Archive data and forecasts](https://solarforecastarbiter.org/usecases/#archive)

Metrics, benchmark forecasts, data formats, data sharing policies, and legal considerations will be detailed in other documents.

## Contents

* [Definitions](https://solarforecastarbiter.org/usecases/#definitions)
  1. [Forecast evaluation timeseries, forecast runs, and forecast datapoints](https://solarforecastarbiter.org/usecases/#forecastdef)
  2. [Forecast evaluation timeseries attributes](https://solarforecastarbiter.org/usecases/#forecastattrs)
  3. [Framework user and framework administrator](https://solarforecastarbiter.org/usecases/#users)
* [Use Cases](https://solarforecastarbiter.org/usecases/#usecases)
  1. [Evaluate forecasts](https://solarforecastarbiter.org/usecases/#evaluatefx)
     1. [Compare a forecast to measurements](https://solarforecastarbiter.org/usecases/#uc1A)
     2. [Compare a probabilistic forecast to measurements](https://solarforecastarbiter.org/usecases/#uc1B)
     3. [Compare multiple forecasts to measurements](https://solarforecastarbiter.org/usecases/#uc1C)
     4. [Compare forecasts to measurements for sites and aggregates](https://solarforecastarbiter.org/usecases/#uc1D)
     5. [Evaluate an event forecast](https://solarforecastarbiter.org/usecases/#uc1E)
     6. [Conduct forecast trial](https://solarforecastarbiter.org/usecases/#uc1F)
     7. [Compare multiple forecast runs to measurements (stretch)](https://solarforecastarbiter.org/usecases/#uc1G)
  2. [Analyze forecasts](https://solarforecastarbiter.org/usecases/#analyzefx)
     1. [Select subsets of forecasts and data](https://solarforecastarbiter.org/usecases/#uc2A)
     2. [Identify events](https://solarforecastarbiter.org/usecases/#uc2A)
     3. [Find forecasts errors with large impacts (stretch)](https://solarforecastarbiter.org/usecases/#uc2A)
* [Framework functional capabilities](https://solarforecastarbiter.org/usecases/#capabilities)
  1. [Perform forecast evaluation in a standard manner](https://solarforecastarbiter.org/usecases/#evaluation)
     1. [Calculate forecast error metrics](https://solarforecastarbiter.org/usecases/#uc3A)
     2. [Calculate forecast impacts (stretch)](https://solarforecastarbiter.org/usecases/#uc3B)
     3. [Communicate probabilistic forecasts](https://solarforecastarbiter.org/usecases/#uc3C)
     4. [Facilitate communication of forecasts and forecast metrics](https://solarforecastarbiter.org/usecases/#uc3D)
     5. [Compute aggregate observations](https://solarforecastarbiter.org/usecases/#uc3E)
  2. [Administer the framework](https://solarforecastarbiter.org/usecases/#administer)
     1. [Manage users and organizations](https://solarforecastarbiter.org/usecases/#uc4A)
     2. [Manage data](https://solarforecastarbiter.org/usecases/#uc4B)
     3. [Provide the framework API](https://solarforecastarbiter.org/usecases/#uc4C)
     4. [Provide reference forecasts](https://solarforecastarbiter.org/usecases/#uc4D)
     5. [Provide reference data](https://solarforecastarbiter.org/usecases/#uc4E)
     6. [Protect forecasts and data](https://solarforecastarbiter.org/usecases/#uc4F)
     7. [Perform data quality checks](https://solarforecastarbiter.org/usecases/#uc4G)
  3. [Archive data and forecasts](https://solarforecastarbiter.org/usecases/#archive)
     1. [Provide historical data, forecasts and forecast errors](https://solarforecastarbiter.org/usecases/#uc5A)

## Definitions

The precise definition of a *forecast* is often a source of confusion in the forecasting industry. Consider these statements that use the term *forecast*:

1. "The forecast for the next 48 hours is 5 MW, 10 MW, 7 MW..."
2. "The hour ahead forecast this morning was 5 MW, 10 MW, 7 MW..."

In Statement A, the user refers to a series of expected values issued at a **single** point in time. In Statement B, the user refers to a series of expected values with the **same** lead time that are issued at **different** points in time.

An evaluation challenge arises if Statement A is extended to include a new 48 hour duration forecast every hour. In this scenario, many forecasts exist at each evaluation time. More information is needed to focus the analysis to a particular application.

Statement B suggests an alternative strategy: the forecast user or the forecast provider parses forecast data into a timeseries for evaluation according to her **application**. For example, from many 48 hour length forecasts, a timeseries of forecast values with lead times of 1 hour ahead could be compiled to evaluate forecasts that support participation in a particular market. Or, a timeseries of forecast values for each hour of the day ahead could be compiled from forecasts issued at the same time each day to evaluate forecasts input to a production cost model. In each case, the forecast under evaluation is a continuous, non-overlapping timeseries that can be compared to observations.

The Solar Forecast Arbiter is designed to analyze these user-supplied, continuous, non-overlapping timeseries. We refer to this as a *forecast evaluation timeseries*.

The Solar Forecast Arbiter provides a taxonomy to allow forecast users and forecast providers to clearly and completely describe each forecast evaluation timeseries. For most use cases, the user is required to consider the forecast application, parse the forecast data into a continuous, non-overlapping timeseries for evaluation, and provide the parsed data to the Solar Forecast Arbiter for analysis – not the individual forecast runs that went into the parsed forecast.

The sections below define and illustrate the key terms and concepts.

**Forecast evaluation timeseries, forecast runs, and forecast datapoints**

**Forecast datapoint**. A *forecast datapoint* is a single (*Time*, *Value*) data pair. Time can label a moment in time or an interval of time, as defined by the interval type label. For example, a Time of 17:00 can label the hour-long interval beginning from 17:00 to 18:00 (period beginning label), 16:00 to 17:00 (period ending label) or instantaneously at 17:00 (instantaneous label). Value has a *quantity* (e.g., mean, 95th percentile) and *type* (e.g. power, GHI). The type defines the unit.

**Forecast run**. A *forecast run* is a sequence of one or more *forecast datapoints* issued at the same time. An example forecast run for the average hourly power at Power Plant X, issued at 16:00 for the next three hours, could be {(17:00, 5 MW), (18:00, 10 MW), (19:00, 7 MW)}. A forecast run is described by the following attributes (with the values relevant to this example in parentheses):

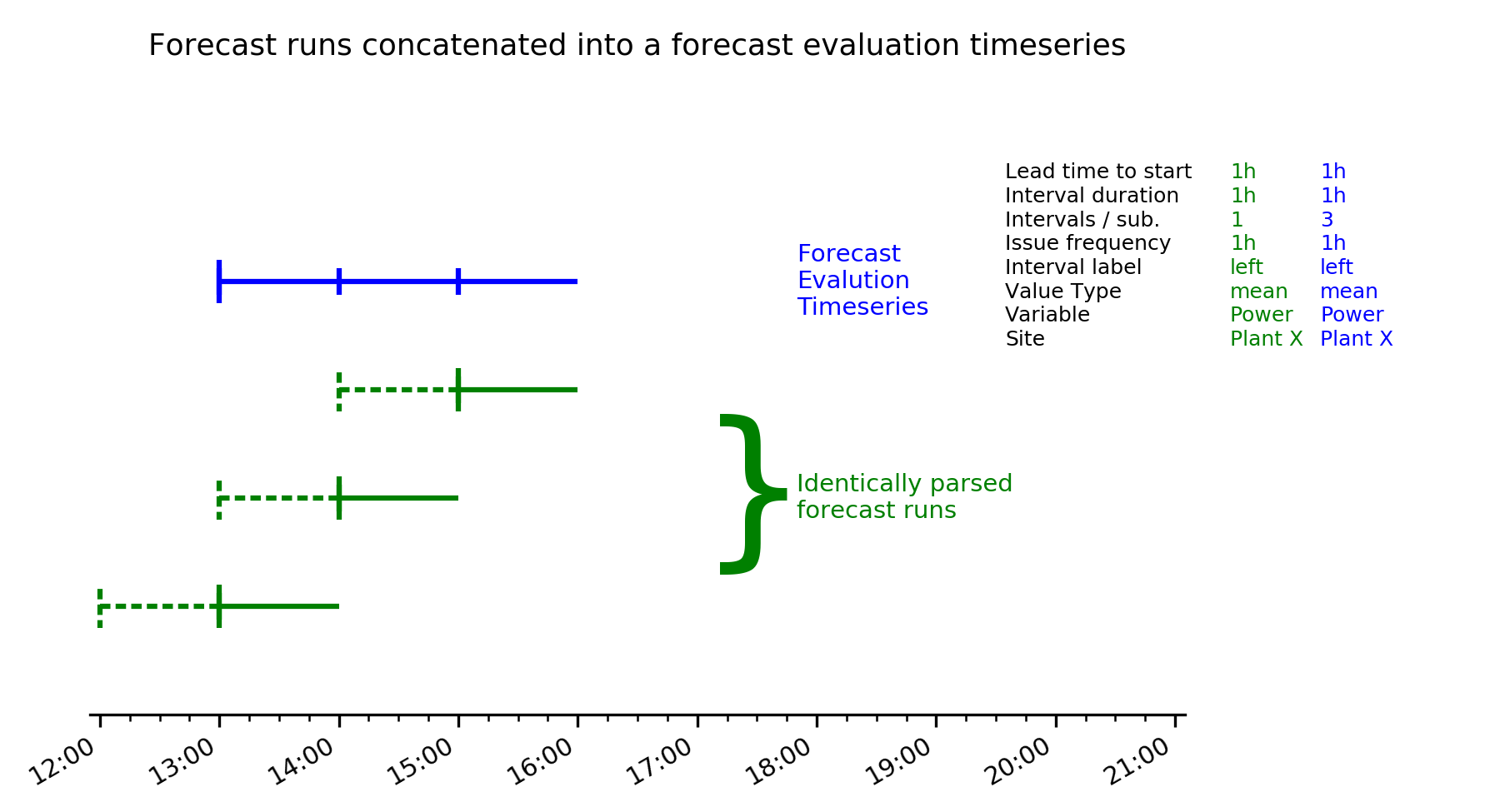
1. Issue time (16:00)
2. Lead time to start (1 hour)
3. Interval duration (1 hour)
4. Number of intervals per submission (3)
5. Forecast issue frequency (3 hr)
6. Interval label (start)
7. Value type (mean)
8. Variable (power)
9. Site (Power Plant X)

**Forecast evaluation timeseries**. A *forecast evaluation timeseries* is a complete series of (*forecast datapoints*) spanning from a *start time* to an *end time*. For example, a single *forecast run* is a *forecast evaluation timeseries*. Typically, a *forecast evaluation timeseries* is the concatenation of a series of identically specified forecast runs with sequential issue times, as shown in the figure below. For example, three forecast runs (green) could be:

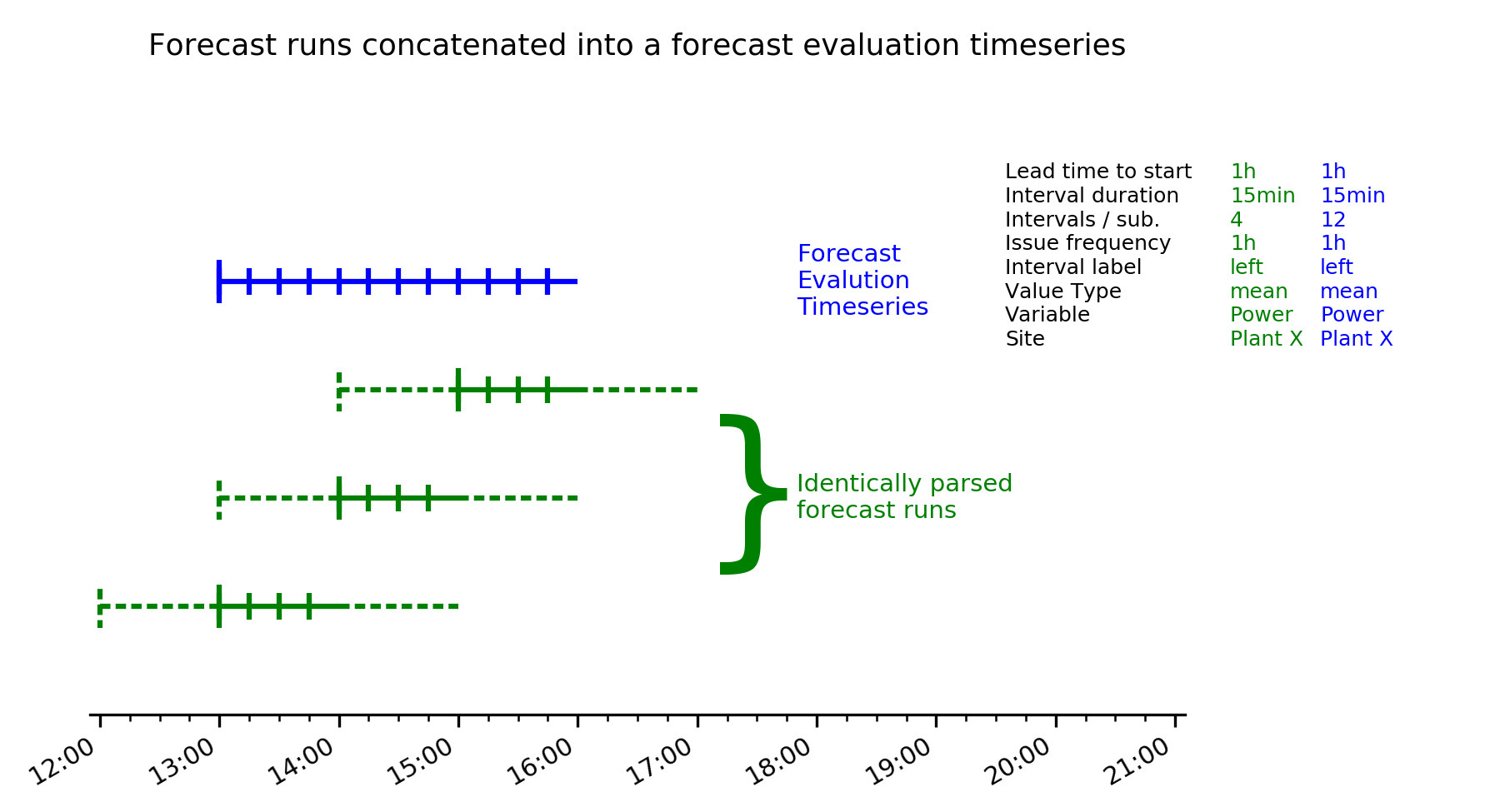
* {(13:00, 10MW)} issued at 12:00, and
* {(14:00, 7MW)} issued at 13:00, and
* {(15:00, 5MW)} issued at 14:00.

From these three runs, the user could upload an hour ahead, hour interval forecast of:

* {(13:00, 10MW), (14:00, 7MW), (15:00, 5MW)}



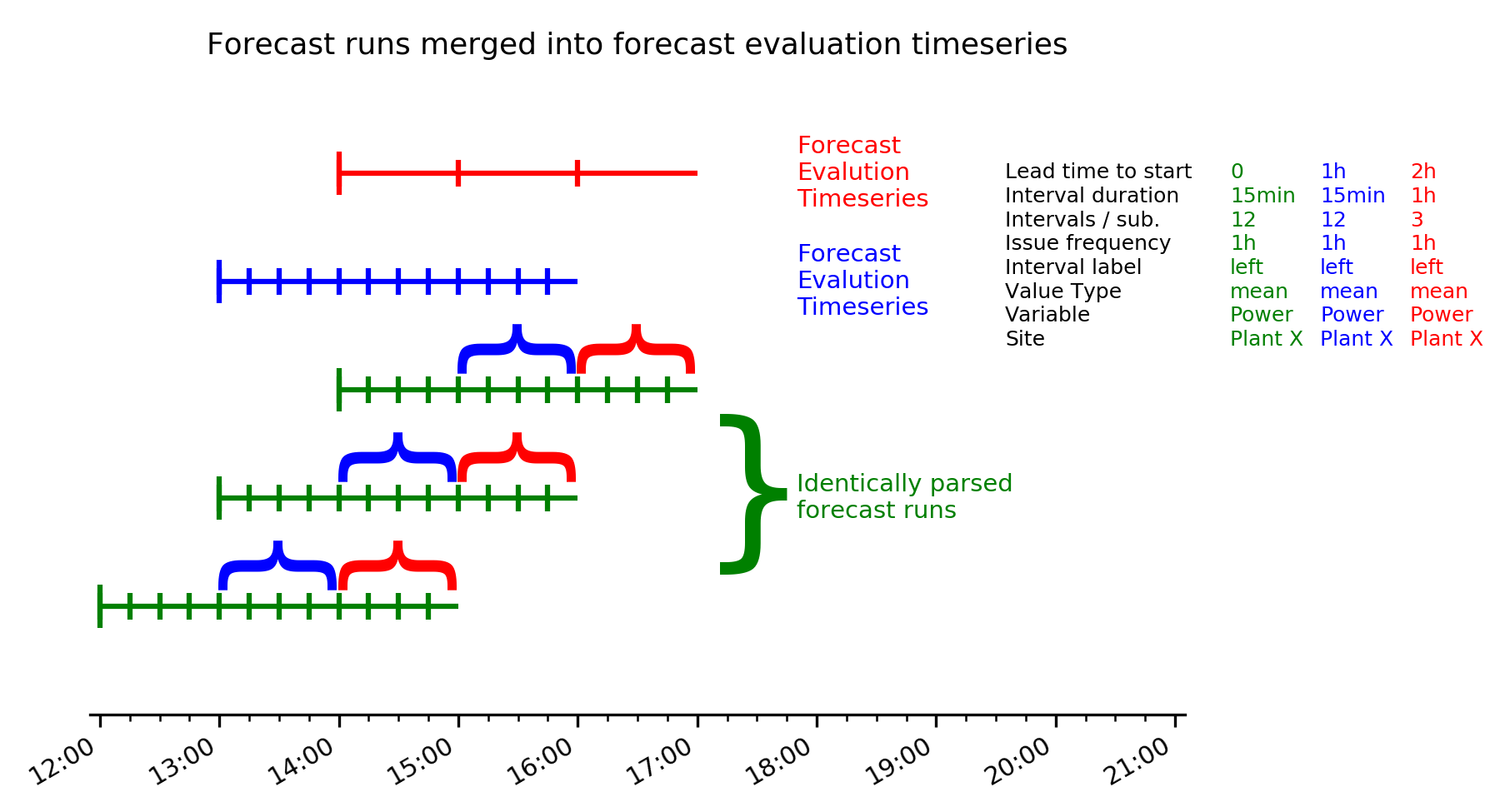
The framework also supports *forecast evaluation timeseries* that are the concatenation of *forecast runs* with more than one interval, so long as the intervals do not overlap. The example below shows how three different hour-ahead, 15-minute interval, 1-hour duration *forecast runs* may be parsed and concatenated into a single forecast evaluation timeseries.



This concept also applies to the evaluation of day-ahead forecasts issued at a particular time of day. For example, hour average forecasts for next day as of 16:00 previous day can be represented by a concatenation of forecasts with start time of midnight, 8-hour lead time to start, 1-hour interval, 24 intervals per submission, and 1 day frequency.

For most use cases ([1.A](https://solarforecastarbiter.org/usecases/#uc1A)-[1.E](https://solarforecastarbiter.org/usecases/#uc1E)), the framework expects forecast providers to upload a forecast evaluation timeseries. For the Forecast Trial use case ([1.F](https://solarforecastarbiter.org/usecases/#uc1F)), the Solar Forecast Arbiter expects forecast providers to regularly upload a forecast runs that the Solar Forecast Arbiter can concatenate into a forecast evaluation timeseries.

A stretch goal is to support the use case of uploading and analyzing multiple forecast runs with the same valid times ([1.G](https://solarforecastarbiter.org/usecases/#uc1G)). For example, the user could upload each of the forecast runs specified above, and then use the framework to merge the forecast runs into a 0, 1, or 2 hour ahead forecast for evaluation. The figure below shows three forecasts runs (green) merged into two different evaluation forecasts (blue, red).



**Forecast evaluation timeseries attributes**

The Solar Forecast Arbiter uses the set of attributes defined below to describe the characteristics of a forecast evaluation timeseries. This structure allows any type of forecast typically utilized in the solar energy field to be evaluated.

A *forecast evaluation timeseries* is a serially complete time series of values with the following attributes:

1. Issue time - The time of day that a forecast is issued, e.g. 13:00. Multiple issue times are uniquely determined by any one issue time and forecast issue frequency.
2. Lead time to start - The difference between each issue time and the start of the first forecast interval, e.g. 1 hour.
3. Interval duration - The length of time that each data point represents, e.g. 5 minutes, 1 hour.
4. Intervals per submission - The number of intervals in each forecast submission, e.g. 12.
5. Forecast issue frequency - The frequency at which new forecasts are issued, e.g. 1 hour, 6 hours, 1 day.
6. Interval label - Indicates if a time labels the start or the end of an interval, or indicates an instantaneous value.
7. Value type - The type of the data in the forecast, e.g. mean, max, 95th percentile.
8. Variable - The variable in the forecast, e.g. power, GHI, DNI. Each variable is associated with a standard unit.
9. Site - The predefined site that the forecast is for, e.g. Power Plant X or Aggregate Y.

A forecast must be associated with a pre-defined site. The site, not the forecast, defines geographic location and the variable determines the units.

**Framework user and framework administrator**

A *framework user* is an individual or organization that uses the capabilities of Solar Forecast Arbiter to evaluate and analyze forecasts. A framework user can be a *forecast provider* (one who creates a forecast), a *forecast user* (one using a forecast for a defined purpose), researchers, and others.

The *framework administrator* operates Solar Forecast Arbiter, ensures data security and anonymity, establishes standards for forecast evaluation, and provides benchmark forecasts.

## Use cases

A use case describes a sequence of actions to achieve a goal. Use cases are grouped into two categories: [Evaluate forecasts](https://solarforecastarbiter.org/usecases/#evaluatefx), and [Analyze forecasts](https://solarforecastarbiter.org/usecases/#analyzefx). From each use case a list of framework requirements is developed. To reduce repeating requirements, a use case may adopt, expand, or modify the requirements of another use case.

**In this section, for brevity, a forecast refers to the** [***forecast evaluation timeseries***](https://solarforecastarbiter.org/usecases/#forecastdef) **defined above.**

### 1. Evaluate forecasts

#### 1.A. Compare a forecast to measurements

**Use case narrative**: A framework user uploads a forecast or selects a benchmark forecast, uploads corresponding measured data or selects reference data, specifies filters, specifies temporal averaging, and selects metrics. The framework calculates metrics, provides a visual display of forecast performance and an evaluation report for download.

**Requirements**:

* User can upload measured data, select from previously uploaded measured data, or select a location/period with reference data ([4.E](https://solarforecastarbiter.org/usecases/#uc4E)).
  + If uploading data, user must define relevant metadata including location, data type, and units.
  + Data type can be irradiance, power, or net-load.
* User can upload a corresponding [forecast](https://solarforecastarbiter.org/usecases/#forecastdef) ([4.C](https://solarforecastarbiter.org/usecases/#uc4C)) or select corresponding benchmark forecast ([4.D](https://solarforecastarbiter.org/usecases/#uc4D)). The forecast value type must be deterministic.
* Framework applies data quality checks to uploaded data and forecasts ([4.G](https://solarforecastarbiter.org/usecases/#uc4G)).
* User can specify filters to exclude forecast or data points (a subset of (time, value) pairs), by time or by value ([2.A](https://solarforecastarbiter.org/usecases/#uc2A)).
* User can select among forecast performance metrics ([3.A](https://solarforecastarbiter.org/usecases/#uc3A)).
* Framework provides metric values and visuals of forecast performance ([3.A](https://solarforecastarbiter.org/usecases/#uc3A)).
* Framework provides a report for download ([3.D](https://solarforecastarbiter.org/usecases/#uc3D)).
* Framework protects forecasts and uploaded data as user-owned: user can specify other users and/or groups who can see/use the uploaded forecasts, data, or summary error statistics ([4.F](https://solarforecastarbiter.org/usecases/#uc4F)).

#### 1.B. Compare a probabilistic forecast to measurements

**Use case narrative**: A framework user uploads a probabilistic forecast, uploads corresponding data or selects reference data, specifies filters, and selects metrics. The framework calculates probabilistic metrics and provides a visual display of probabilistic forecast performance.

**Requirements**:

* All requirements as listed for Use Case [1.A](https://solarforecastarbiter.org/usecases/#uc1A) with the following modifications.
* A forecast value must be an empirical probability distribution at each time point (a sequence of {quantile, value} or {Prob(X), X} pairs).
* User can select among probabilistic forecast performance metrics ([3.A](https://solarforecastarbiter.org/usecases/#uc3A)).
* Framework provides probabilistic metric values for comparison among forecasts ([3.A](https://solarforecastarbiter.org/usecases/#uc3A)), visual displays of probabilistic forecast performance, and a report for download ([3.D](https://solarforecastarbiter.org/usecases/#uc3D)).

#### 1.C. Compare multiple forecasts to measurements

**Use case narrative**: Framework users upload forecasts, select benchmark forecasts, or invite other users to share forecasts. Framework users upload corresponding data or select reference data, specify data filters, specify temporal averaging, and select metrics. The framework calculates metrics and provides a visual display of forecast performance and forecast errors for download, and a visual display comparing the forecasts. The framework user can share results with other users.

**Requirements**:

* All requirements as listed for Use Case [1.A](https://solarforecastarbiter.org/usecases/#uc1A) or [1.B](https://solarforecastarbiter.org/usecases/#uc1B) for each forecast in the comparison.
* User can share forecast specifications (e.g., quantity, interval, horizon, averaging, lead times) with other framework users and invite other framework users to share forecasts for the comparison ([4.F](https://solarforecastarbiter.org/usecases/#uc4F)).
* Framework supports anonymous forecast comparison ([4.F](https://solarforecastarbiter.org/usecases/#uc4F)).
* Framework protects forecasts as specified by each forecast owner ([4.F](https://solarforecastarbiter.org/usecases/#uc4F)).
* Framework provides metric values for the comparison among forecasts ([3.A](https://solarforecastarbiter.org/usecases/#uc3A)), a visual display of comparison results, and a report for download ([3.D](https://solarforecastarbiter.org/usecases/#uc3D)).

#### 1.D. Compare forecasts to measurements for sites and aggregates

**Use case narrative**: A framework administrator defines an aggregate of locations. A framework user uploads forecasts for locations in the aggregate and a forecast for the aggregation of locations. A framework user uploads corresponding data or selects reference data, specifies filters, specifies temporal averaging, and selects metrics. The forecasts and data may represent individual point sensors, power plants, or aggregated quantities. The framework calculates metrics and provides a visual display comparing the forecasts across scales.

**Requirements**:

* All requirements as listed for Use Case [1.A](https://solarforecastarbiter.org/usecases/#uc1A), [1.B](https://solarforecastarbiter.org/usecases/#uc1B), or [1.C](https://solarforecastarbiter.org/usecases/#uc1C), as appropriate.
* User uploads an aggregation of forecasts with weights specified by the framework ([4.C](https://solarforecastarbiter.org/usecases/#uc4C)).
* Framework computes observations, as specified ([3.E](https://solarforecastarbiter.org/usecases/#uc3E)).
* Framework provides metric values for the comparison among forecasts ([3.A](https://solarforecastarbiter.org/usecases/#uc3A)), a visual display of comparison results, and a report for download ([3.D](https://solarforecastarbiter.org/usecases/#uc3D)).

#### 1.E. Evaluate an event forecast

**Use case narrative**: A framework user can analyze an event forecast.

**Requirements**:

* All requirements as listed by Use Case [1.A](https://solarforecastarbiter.org/usecases/#uc1A), [1.B](https://solarforecastarbiter.org/usecases/#uc1B), or [1.C](https://solarforecastarbiter.org/usecases/#uc1C), as appropriate.
* User can upload an event forecast, e.g., a time series of true/false values, or a time series of event occurrence likelihood ([4.C](https://solarforecastarbiter.org/usecases/#uc4C)).
* User can upload corresponding event data or connect to a saved event time series ([2.B](https://solarforecastarbiter.org/usecases/#uc2B)).
* User can select among event forecast performance metrics ([3.A](https://solarforecastarbiter.org/usecases/#uc3A)).
* Framework provides metric values and visuals of event forecast performance ([3.A](https://solarforecastarbiter.org/usecases/#uc3A)).
* Framework provides a report for user download ([3.D](https://solarforecastarbiter.org/usecases/#uc3D)).

#### 1.F. Conduct forecast trial

**Use case narrative**: The framework administrator, in consultation with trial participants, defines trial period, locations, forecast quantities, time horizons, time resolutions, etc. of a forecast trial. The trial may be either retrospective or live, and may involve a test period followed by an evaluation period. Separate trials may be administered for multiple forecast parameters (e.g. hour ahead and day ahead forecast evaluations for the same set of locations).

**Requirements**:

* All requirements as listed for Use Case [1.C](https://solarforecastarbiter.org/usecases/#uc1C).
* Administrator, in consultation with trial participants, defines data sources, forecast run intervals/horizons, trial period, metrics, filters ([4.C](https://solarforecastarbiter.org/usecases/#uc4C)).
* Framework concatenates regularly posted forecast runs into a continuous forecast ([4.C](https://solarforecastarbiter.org/usecases/#uc4C)).
* Framework accepts and allows for download of streaming data ([4.C](https://solarforecastarbiter.org/usecases/#uc4C)).
* Framework generates intermediate reports during the trial ([3.D](https://solarforecastarbiter.org/usecases/#uc3D)).

#### 1.G. Compare multiple forecast runs to measurements (stretch)

**Use case narrative**: A framework user uploads a sequence of forecast runs at a regular interval, uploads corresponding data or selects reference data, specifies filters, specifies temporal averaging, and selects metrics. The framework calculates metrics, provides a visual display of forecast performance and an evaluation report for download.

**Requirements**:

* All requirements as listed for Use Case [1.A](https://solarforecastarbiter.org/usecases/#uc1A), for each forecast run.
* Each [forecast run](https://solarforecastarbiter.org/usecases/#forecastdef) must be identically specified.
* Forecast runs are issued at a regular interval (e.g., every 6 hours).
* User can upload multiple forecast runs. Each forecast run must be:
  + A time series of uniform forecast intervals, duration, and value type (e.g. 5 minute interval, interval average forecasts through 24 hours from the start time)
* User can define forecast evaluation intervals, horizons, and temporal averaging.

### 2. Analyze forecasts

#### 2.A. Analyze subsets of forecasts and data

**Use case narrative**: A framework user selects a subset of an uploaded forecast, and the corresponding data, based on one or more conditions, for detailed analysis. The framework calculates metrics and provides a visual display of forecast performance.

**Requirements**:

* User can select a subset of points in a forecast or its corresponding data based on one or more conditions: forecast error (bias), forecast lead time, time of day, weather (irradiance, temperature, wind). Framework returns selected subsets from the forecast and corresponding data.
* User can select among forecast performance metrics ([3.A](https://solarforecastarbiter.org/usecases/#uc3A)).
* Framework applies data quality checks to uploaded data ([4.E](https://solarforecastarbiter.org/usecases/#uc4E)).
* Framework provides metric values and visuals of forecast performance ([3.A](https://solarforecastarbiter.org/usecases/#uc3A)).
* Framework provides a report for user download ([3.D](https://solarforecastarbiter.org/usecases/#uc3D)).

#### 2.B. Identify events

**Use case narrative**: The framework assists a user to identify events in a forecast or in data.

**Requirements**:

* User can upload a forecast ([4.C](https://solarforecastarbiter.org/usecases/#uc4C)).
* User can upload data ([4.C](https://solarforecastarbiter.org/usecases/#uc4C)).
* Framework applies data quality checks to uploaded data ([4.E](https://solarforecastarbiter.org/usecases/#uc4E)).
* User can specify conditions (e.g., ramp rates) used to identify events occurring in a forecast or in data.
* Framework provides visual display of identified events ([3.D](https://solarforecastarbiter.org/usecases/#uc3D)).
* Framework provides a report for user download ([3.D](https://solarforecastarbiter.org/usecases/#uc3D)).
* User can save event data, i.e., a time series of true/false values.

#### 2.C. Find forecasts errors with large impacts (stretch)

**Use case narrative**: A framework user can associate forecast errors with system impact and can filter forecasts based on system impact.

**Requirements**:

* All requirements associated with Use Case [1.A](https://solarforecastarbiter.org/usecases/#uc1A).
* User can select conditions to associate forecast error with system impact ([3.B](https://solarforecastarbiter.org/usecases/#uc3B)).
* Framework uses user-selected conditions to calculate system impact ([3.B](https://solarforecastarbiter.org/usecases/#uc3B)).
* Framework finds subset of forecast times corresponding to user-defined “large” system impacts.
* Framework provides a report for user download ([3.D](https://solarforecastarbiter.org/usecases/#uc3D)).

## Framework functional capabilities

Functional requirements are capabilities not specific to a use case but which the framework requires in order to satisfy the use cases. Functional requirements are grouped into three categories:

* [Perform forecast evaluation in a standard manner](https://solarforecastarbiter.org/usecases/#evaluation)
* [Administer the framework](https://solarforecastarbiter.org/usecases/#administer)
* [Archive data and forecasts](https://solarforecastarbiter.org/usecases/#archive)

### 3. Perform forecast evaluation in a standard manner

#### 3.A. Calculate forecast error metrics

**Use case narrative**: The framework provides a set of well-documented metrics which can be selected by a framework user for evaluating or analyzing forecasts. Metrics include, for example, statistics for forecast error, and cost estimated by a user-supplied $/MW-error.

**Requirements**:

* Metrics are documented on the framework website.
* Metrics are coded in the framework.
* Metrics can be selected by a framework user.
* Metrics are available for probabilistic forecasts.

#### 3.B. Calculate forecast error impacts (stretch)

**Use case narrative**: The framework provides a set of well-documented functions to associate forecast error with system impact. System impact may include, for example, increase or decrease in reserve requirements.

**Requirements**:

* Functions for estimating system impact are documented on the framework website.
* Functions are coded in the framework.
* Function parameters can be selected by a framework user.

#### 3.C. Communicate probabilistic forecasts

**Use case narrative**: The framework communicates probabilistic forecast information in a meaningful, well-documented manner.

**Requirements**:

* Quantities and terms used to describe and present probabilistic forecasts are documented on the framework website.
* Framework presents visuals for probabilistic forecasts.

#### 3.D. Facilitate communication of forecasts and forecast metrics

**Use case narrative**: The framework aids a forecast provider to communicate forecast performance.

**Requirements**:

* The framework produces reports of forecast evaluation and analysis for user download.
* The framework website includes pages that aid in explaining forecasts and their use.
* Framework users can share forecasts, forecast evaluation reports, and data with other framework users.
* Framework protects forecasts and uploaded data as specified by the user.

#### 3.E. Compute aggregate observations

**Use case narrative**: The framework computes aggregates of observational data, accounting for missing data and flagged data, in a well-documented manner.

**Requirements**:

* Aggregation functions are coded in the framework.
* Aggregation functions are documented on the framework website.

### 4. Administer the framework

#### 4.A. Manage users and organizations

**Use case narrative**: The framework administrator can manage users and organizations.

**Requirements**:

* Administrator can associate or disassociate users with organizations.
* Administrator can enable data upload privileges based on email communication.
* Administrator can reset user passwords.

#### 4.B. Manage data

**Use case narrative**: The framework administrator can manage data storage, and can delete data upon request or at the end of the DOE funding period.

**Requirements**:

* Framework administrator can assign data storage limits.
* Framework administrator can delete stored data.
* Users can delete user-owned data.

#### 4.C. Provide the framework API

**Use case narrative**: The framework provides a documented API and appropriate services for client users.

**Requirements**:

* The framework’s API is documented on the framework website.
* Users can interact with the framework via url requests.
* Users can request forecast evaluations via script (stretch).

#### 4.D. Provide reference forecasts

**Use case narrative**: The framework provides a set of documented reference forecasts for irradiance and power, for various locations, at various intervals and horizons.

**Requirements**:

* The framework provides a reference irradiance forecast at hourly intervals out to 60 hours for CONUS.
* The framework provides a reference irradiance forecast at user-selected intervals (5 to 60 minutes) out to 24 hours, using smart persistence of user-uploaded data.
* The framework provides a reference forecast for air temperature and wind speed to accompany the reference irradiance forecasts.
* The framework provides a reference model for translating irradiance to power.
* The framework provides a reference forecast for power, using user-uploaded power and a persistence method.
* The framework can provide load and net load forecasts from user-uploaded data using persistence (stretch).
* Reference forecasts and forecast methods are documented on the framework website.

#### 4.E. Provide reference data

**Use case narrative**: The framework provides reference weather and power data at selected locations.

**Requirements**:

* The framework provides reference irradiance data from a selection of stations e.g. SURFRAD stations.
* The framework provides reference power data from a selection of PV systems.
* The framework provides metadata for reference PV systems.

#### 4.F. Protect forecasts and data

**Use case narrative**: The framework protects data and forecasts according to user-selected conditions.

**Requirements**:

* The data sharing agreement terms are posted on the framework website.
* The framework provides user identification services (account management, password services).
* The framework records an owner, optionally under an anonymous pseudonym, for uploaded forecasts and data.
* Data owners can set permissions on data for viewing and downloading.
* Forecast owners can set permission on forecasts for viewing, downloading and attribution.

#### 4.G. Perform data quality checks

**Use case narrative**: The framework applies data quality checks when a user uploads data, provides visuals and text reports of data quality check results, and offers download of user data with data quality flags.

**Requirements**:

* Time stamps are checked for timezone consistency, increasing order, gaps.
* Irradiance data are checked for missing values, non-physical values, periods of constant values, consistency of irdiance components.
* Power data are checked for missing values, non-physical values, periods of curtailment, periods of inverter clipping, miligned trackers, incorrect system metadata.
* Data quality algorithms are coded and documented on the framework.

### 5. Archive data and forecasts

#### 5.A. Provide historical data, forecasts and forecast errors

**Use case narrative**: The framework provides archive service for data and forecasts.

**Requirements**:

* The framework provides an archive of reference irradiance and power data.
* The framework provides an archive of reference forecasts and forecast errors.
* Framework users can store forecasts and associated metadata on the framework up to account data limits.
* Framework users can store measurements and associated metadata on the framework up to account data limits.
* Framework users can share stored forecasts and data.