Agricultural Intelligence
Trusted by leading agricultural companies and organizations since 1999
Ag-Tech Sectors

% of Total Investment Dollars

- Drones 49%
- Satellite 27%
- Software 15%
- Robotics 6%
- Irrigation 5%
- Hardware 4%
- Sensors 3%
- Weather 1%

NEXT GEN FARMS
- Freight Farms
- Bright Farms
- AeroFarms
- Fresh Box
- Green Sense Farms
- Gotham City Farms
- FarmLogics
- AgriWebb
- UNFarm

ANIMAL DATA
- AAD
- Farmnote
- Steffapos
- Mastilne
- TL Biolabs
- Connetora

SMART IRRIGATION
- HydroBio
- Hortau
- Sprinkl
- HydroPoint
- Smart Farm
- PowWoc
- cropx
- AquaSpy
- EDYN

FARM MANAGEMENT SOFTWARE
- scoutpro
- Granular
- Agworld
- picktrace

PRECISION AGRICULTURE AND PREDICTIVE ANALYTICS
- Strider
- CropLogic
- Adapt-N
- boxcontrol
- CropZilla
- prospera

MARKETPLACES
- AgroStar
- AgriTonic
- Agriagora
- EMA
- Yagro

ROBOTICS AND DRONES
- mavrx
- FoamBox
- AIRWIND
- BlueRiver
- SkySquirrel
- SkyCision

SENSORS
- Centaur
- SLANTRANGE
- Sensors
- flux
- Smart Yields

© Copyright 2016, aWhere. All Rights Reserved
Data: foundational agronomic weather

- Agricultural Earth Coverage
  - 1.5 million aWhere weather stations
- Daily Observed
  - Precipitation
  - Min/Max Temperature
  - Min/Max Relative Humidity
  - Max/Mean Wind speed
  - Solar Radiation
  - 10 years history
- Hourly Forecast
  - 8-15 days of hourly forecast (updated 4x daily)
- API Accessible

More than 7 billion data points processed each day

Global in coverage – yet localized & specific

Current, Correct, Consistent, & Complete
Blending Methodology

Satellite Radar: 35%
Ground Radar: 15%
Weather Station: 28%

Field:
- Temperature, Humidity etc.: 38%
- Precipitation: 5%
More Specific Data = Better Information...

Relying on inferior data results in incorrect decisions

More specific information gives more clarity at the micro and macro scale

India
Free and Premium Providers
78 Weather Stations
Approx. Every 120km

India
aWhere
41,000 Virtual Weather Stations
...one every 9km

© Copyright 2016, aWhere. All Rights Reserved
Better *Information also enables Field Insight*

- When to plant?
- When and how much to irrigate?
- What pests to scout and treat for?
- When and how much to fertilize?
- When will harvest be?
- How much yield is expected?
- When an insurance payout triggers?
- What is the risk of a financial loan?

Accumulated Water

<table>
<thead>
<tr>
<th>Month</th>
<th>Irrigated</th>
<th>Natural</th>
</tr>
</thead>
<tbody>
<tr>
<td>May</td>
<td></td>
<td></td>
</tr>
<tr>
<td>June</td>
<td></td>
<td></td>
</tr>
<tr>
<td>July</td>
<td></td>
<td></td>
</tr>
<tr>
<td>August</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Potential Evapotranspiration

<table>
<thead>
<tr>
<th>Season</th>
<th>Actual</th>
<th>Normal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Season</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Season + 7 Days</td>
<td>11.2&quot;</td>
<td>11&quot;</td>
</tr>
</tbody>
</table>

Rain Forecast

**Forecast**

Today should be Clear, with No Rain and Light Winds. Tomorrow will be Partly Cloudy with Trace Rain and Light Wind.
Access: information distribution

API & Developer Portal

Forecast App

Monitor App

Statistics Package

Code Samples & Widgets

Third Party Tools e.g. ESRI, Microsoft

Mobile Apps
Applications
**APIs**

- Also known as Application Programming Interfaces, at their most basic level, allows applications to talk to other applications.
- API’s enable communication of data but also provide functionality that can be requested over the web.

**REST**

- Representational state transfer (REST) or RESTful Web services are an the underlying architectural principle of the web.
- The browser doesn’t know in advance where to submit the information, and it doesn’t know in advance what information to submit.

**Authentication**

- A way to authenticate who you are

**Tokens**

- A unique code assigned to authenticate you

**Call**

- A request from an API
- GET requests information from an API
- POST pushes information to an API
aWhere Developer Account

**Developer Account**
- Your account on developer.awhere.com to manage your apps

**App**
- A specific app that you want to develop

**Key and Secret**
- A unique encrypted identifier assigned to you to make requests to aWhere’s API
Field
- A single location assumed to be the latitude and longitude of the center of a field

Planting
- A crop planted on a Field with a planting date, estimated harvest date and acreage

Model
- A model run for a planting to estimate growth stage for that planting
aWhere API Functional Groups

**Daily Observations**
- Historical data going back 10 years

**Forecasts**
- Forecast going out 7 days

**Norms**
- Comparisons to normal or specific years

**Current Conditions**
- Conditions at the current time (US only)

**Models**
- Growth stage models

**Maps**
- Maps of radar, temperature etc. (US only)

API Endpoints:
- `/v2/weather/fields/{fieldId}/observations`
- `/v2/weather/fields/{fieldId}/observations/{singleDate}`
- `/v2/weather/fields/{fieldId}/observations/{startDate},{endDate}`
Developer Portal

- Comprehensive Documentation
- Tutorials & Best Practices
- Sample Code
- Support Community
- Self-Service API credentials

Key: ICT4D_KEY
Secret: ICT4D_SECRET

Get started with a 30-day trial at developer.awhere.com
Postman Collection

Get started at https://developer.awhere.com/integration/postman
Code Samples and Github

Visit https://github.com/aWhereAPI
Excel Users - Download Tool

The GDA Download Tool

- Requires API credentials
- Download select data to Excel

Key: ICT4D_KEY
Secret: ICT4D_SECRET

Get started by visiting https://awhere-weather-data.herokuapp.com
Agricultural Intelligence Platform
apps.awhere.com

• Monitor
• Forecast
• Predict

Username: DataJam
Password: ICT4D
R-Studio – Statistical Package

The aWhere API R Package

- Comprehensive Documentation
- Pre-set functions
- Requires API credentials

Key: ICT4D_KEY
Secret: ICT4D_SECRET

Get started by downloading at https://github.com/aWhereAPI/aWhere-R-Library
Further Details & Demos
Data Overview

Location-based, point specific information
Daily Historical Observed Data
Forecast Data
Long Term Norms
Agronomic Data
Daily Historical Observed Data

- Min & Max Temperature
- Precipitation
- Solar Energy
- Min & Max Relative Humidity
- Average, Day’s Max, and Morning’s Max Wind Speed
Forecast Data

Min & Max Temperature
Chance of Precipitation
Precipitation Amount
Cloud Cover
Solar Energy

Average & Max Wind Speed
Average Relative Humidity
Dew Point
Min & Max Relative Humidity
Summary conditions
Long Term Norms

- Mean, Min, & Max Temperature
  - Precipitation
  - Solar Energy
- Min & Max Humidity
- Mean & Max Wind
- Agronomic Norms
Agronomic Data

Growing Degree Days/Units (GDD / GDU)
Potential Evapotranspiration (PET)
Ratio Precipitation to PET (P/PET)
Accumulated GDD
Accumulated Precipitation
Accumulated PET
Accumulated P/PET
Using aWhere’s Data

E.g. Temperature, Precipitation
  E.g. GDD
  E.g. PET
  E.g. P/PET
Standard Maize GDD Models

GDDs and Growth Stage Models Example

Using GDDs and/or the Growth Stage Models, your app can chart the history of a plant’s development, or estimate when the next growth stage is likely to be seen.
Potential Evapotranspiration (PET)

\[
evapotranspiration = \text{transpiration} + \text{evaporation}
\]

- Transpiration
  - Trees
  - Grass
- Evaporation
- Runoff
- Groundwater recharge
Precipitation / PET

- Proxy for whether soil moisture amount changing over time
- Doesn’t represent actual available water
Agricultural Intelligence Platform
apps.awhere.com

• Monitor
• Forecast
• Predict

Username: DataJam
Password: ICT4D
R-Studio – Statistical Package

The aWhere API R Package

- Comprehensive Documentation
- Pre-set functions
- Requires API credentials

Key: ICT4D_KEY
Secret: ICT4D_SECRET

Get started by downloading at https://github.com/aWhereAPI/aWhere-R-Library
Agricultural Intelligence Platform - Monitor Tool
Agricultural Intelligence Platform - Forecast Tool
Agricultural Intelligence Platform - Predict Tool
RStudio IDE
Package Installation

```
11 install.packages('devtools')
12 install.packages(c('chron', 'magrittr', 'btcops', 'DBI', 'assertthat', 'Rcpp', 'tibble'))
14 devtools::install_github("awhereAPI/awhere-R-Library")
```
Authentication and Fields

```r
### Basic Commands in the awhere R Package
# You may run these as many times as you would like, and update as needed

library(awareAPI)

### Get Token - key and secret
# Update the fields "key" and "secret" with your personal key and secret accessed
# via the awhere Developer Portal - http://developer.awhere.com/user/login
get_token("ICT4D_KEY", "ICT4D_SECRET")

### Create Field
# This will create a field in your awhere account at the following location.
# Update location and ID to create additional fields.
create_field(field_id = "test_india1", latitude = "18.344480", longitude = "79.881494",
             farm_id = "Test", field_name = "India Field 1", acres = "25")
create_field(field_id = "test_india2", latitude = "20.809051", longitude = "75.224085",
             farm_id = "Test", field_name = "India Field 2")

### Get Fields List
# This will provide a list of all the fields currently stored in your awhere account.
```
Observed & Forecast

```R
# Weather data - input name/id of field or latitude/longitude points
# This pulls the data and creates a dataset in R titled "obs" that can be viewed later.
# The two lines below will perform an identical function - one calls location by field/ID/name and
# the two lines below will perform an identical function - one calls location by field/ID/name and
obs1 <- daily_observed_fields("test_india", day_start = "2016-06-01", day_end = "2016-07-01")
obs2 <- daily_observed_latlng(20.890051, 75.224885, day_start = "2016-06-01", day_end = "2016-07-01")
view(obs1)

# Forecast data - customize call as needed
# This pulls the data and creates a dataset in R titled "fcst" that can be viewed later.
# The two lines below will perform an identical function - one calls location by field/ID/name and
# note: update day_start to be a day in the near future or today.
fct1 <- forecasts_fields("test_india", day_start = "2017-05-19")
fct2 <- forecasts_latlng(20.890051, 75.224885, day_start = "2017-05-19")
view(fct1)
```
Long-term Norms & Agronomic Data

```r
# Long-term norm data - norms determined based on month-day (MM-DD) spans, with default
# as 10-year norms. Can customize years and exclude years.
# This pulls the data and creates a dataset in R titled "ln" that can be viewed later.
#
# 10-year norms
ln1 <- weather_n norms_fields("test_india l", monthday_start = "06-01", monthday_end = "09-01",
  year_start = 2006, year_end = 2015)

# Custom-year norms
ln2 <- weather_n norms_fields("test_india l", monthday_start = "01-01", monthday_end = "03-29",
  year_start = 2008, year_end = 2012)

# Agronomic data
# This pulls the data and creates a dataset in R titled "ag" or "ag_ln" that can be viewed later.
ag1 <- agronomic_values_fields("test_indial", day_start = "2016-03-15", day_end = "2016-03-15")
ag2 <- agronomic_values_fields("test_indial", day_start = "2016-03-15", day_end = "2016-04-01")
```
Agronomic Norms

```r
### Agronomic data
# This pulls the data and creates a dataset in R titled "ag" or "ag_ltn" that can be viewed later.
ag1 <- agronomic_values_fields("test_india", day_start = "2016-03-15", day_end = "2016-03-15")
ag2 <- agronomic_values_fields("test_india", day_start = "2016-03-15", day_end = "2016-04-01")

### Agronomic Norms
#### 10-year norms
ag_ltn1 <- agronomic_norms_fields("test_india", month_day_start = "03-15",
                                 month_day_end = "04-02", year_start = 2006, year_end = 2015)

#### Custom-year norms
ag_ltn2 <- agronomic_norms_fields("test_india", month_day_start = "03-15",
                                 month_day_end = "04-01", year_start = 2010, year_end = 2015)

View(ag2)
View(ag_ltn2)

### Set working directory
# Update this to use the file path on your computer where you would like to save your data.
setwd("C:/Users/Hannacamp/Documents/ahere 2017/Scripts/output")

### Save & export data into .csv file
# You can change which dataset you want to export - here we're exporting "ag_ltn2" and "obs1" into
write.csv(ag_ltn2, file = "Agronomic_Long_Term_Norms.csv")
write.csv(obs1, file = "observations.csv")
```
Help documentation

daily_observed_lating

Description

daily_observed_lating pulls historical weather data from aWhere's API based on latitude & longitude.

Usage

daily_observed_lating(latitude, longitude, day_start, day_end)

Arguments

- latitude: the latitude of the requested location (double)
- longitude: the longitude of the requested locations (double)
- day_start: character string of the first day for which you want to retrieve data, in the form: YYYY-MM-DD
- day_end: character string of the last day for which you want to retrieve data, in the form: YYYY-MM-DD
Quick Excel Use

![Daily Average Temperatures - Sample Farmer Locations](chart.png)
Quick Excel Use