MessageCast and AgNet: Unleashing Collaboration, IoT and Machine Learning for Agriculture - Through Open-Source

Sumer Johal[1], Sandeep S. Joshi[2], Kelvin Degbotse[3], Mikela Jackson[4] Aanish Amir[5] [and the greater AgStack Open-Source Community]

Global agriculture is a massive multi-trillion dollar ecosystem encompassing several different stakeholders across a complex value chain. Digital technologies hold tremendous promise to advance inclusive agriculture transformation at a scale and speed needed to make a difference. There has been good progress in isolated point solutions, however, due to legacy issues, the broad transformation has been slow to materialize.

AgStack Foundation, a Linux Foundation project, is leading the charge to build and sustain the global data infrastructure for food and agriculture to help scale digital transformation, and address climate change, rural engagement and food and water security. AgStack's mission is to improve global agriculture efficiency through the creation, maintenance and enhancement of free, reusable, open and specialized digital infrastructure for data and applications. One of the first software frameworks being developed by AgStack is MessageCast, an open-source software message bus for sending and receiving messages between distributed systems.

Built specifically for the agriculture ecosystem, MessageCast is designed to meet the following criteria:

- Enable peer-to-peer asynchronous communication on "topics" as channels
- Access the data that is captured
- Support geo-spatial context to discovery
- Customize distinct data types (e.g., forms, ontologies)
- Connect to the edge sensors and cloud services

---

[1] Sumer Johal, Executive Director at AgStack Foundation (sjohal@linuxfoundation.com)
[2] Sandeep S. Joshi, CEO at AgPipe (sandeep@agpipe.com)
[3] Kelvin Degbotse Head of Product at AgPipe (kelvin@agpipe.com)
[4] Mikela Jackson, Technical Writer at Red Hat (mzjackson@redhat.com)
[5] Aanish Amir, Chief Maintainer at AgStack

- Enable data syndication (AgNet) that helps forster and facilitate Artificial Intelligence (AI) at scale for the agriculture domain

MessageCast, just like the rest of AgStack, is an open-source global community-led project that will be available for download on GitHub at MessageCast.

Django Transformed and Customized for Agriculture

MessageCast is a mobile websockets-based asynchronous message bus that supports users publishing and subscribing to messages on a *topic* to a chatroom. It is built using customizable Django channels. MessageCast can be leveraged not only for person-to-person communication, but will also support device-to-device communication. This software framework will not just leverage, but in fact enhance the combined value of several other AgStack hosted web service projects like User Registry and Asset Registry.

Let's look at some of the details of MessageCast, starting with the motivation for building this framework at a time when other seemingly comparable frameworks might exist.
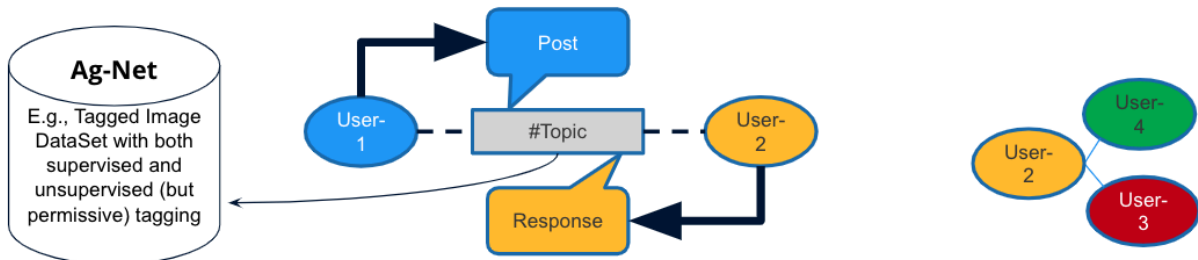
	User-1	User-2	Post	Response
WP #1	Breeder	Breeder	Phenotyping Observation	Phenotyping Affirmation
WP #2	Ag-Extension Expert	Ag-Extension Expert	Pest Trap /Disease Observations	Identifying and accounting
WP #3	Producer	Buyer and MIddlemen	Timing, quantity and quality of harvest	Purchase linkages

Motivation for building MessageCast

Why can't we use Whatsapp or something similar for agriculture? Why not use an off-the-shelf message bus like Kakka, RabbitMQ or Redis? We wanted to start by addressing some of these questions that we have heard from others, and questions that may also cross the minds of some of our readers.

The short answer is, we wanted a framework specifically built for the agriculture ecosystem to meet its unique needs, and one that anyone can download and run in their own environments. Existing frameworks have several shortcomings as it relates to our needs:

- The data that is shared on the framework is stored on the servers of the framework provider. We want to make sure that the framework can be run in private environments where the data exclusively resides on servers within that environment.
- Existing frameworks don't have the ability to support geospatial context for the data. This is a critical requirement for agriculture applications to support geolocation services, to receive data from a particular farm or general locality. Agriculture applications rely on geolocation to make sense of data, such as the soil moisture data, yield maps, remote sensing data, etc.
- Agriculture uses highly localized terms for different entities, be it crops (e.g. corn versus maize), diseases, pests, fertilizers, etc. The ability to tailor the framework for distinct data types (e.g., forms, ontologies) is not possible in most existing frameworks.
- Currently, each player in ML is designing and hosting their own proprietary repository of data for their machine learning algorithms. This is not ideas as the repository is not comprehensive. The distributed use of MessageCast will automatically create a tagged image repository - that we are calling AgNet (like imageNet) - that will be available under an open data license CDL v2.0. Additionally, this data repository will be built with APIs aligned with Tensorflow, Pytorch and OpenAI - three different open-source standards for tools for machine learning. This will unleash the true power of agriculture-domain-centric tagged and structured data towards ML applications such as pest and disease prediction. Carbon monitoring, nutrient response and optimization and water.
- Current messaging systems - out of the box - aren't designed to be integrated with IoT devices and geo-references (like a soil moisture sensor or the asset registry geoid). This prevents their automatic usage for sensor-to-human and sensor-to-cloud communication.

As a result, we are customizing Django Channels for building this asynchronous messaging bus for meeting the unique needs of agriculture.

Transformed and Customized for Agriculture

As we were designing this messaging bus, we wanted to make sure that it met the unique needs for asynchronous messaging for the agriculture industry. Here are some of the key features of MessageCast:

- User discovery: Users will be able to set their "discoverability preferences" in their profile preferences at AgStack's user-registry so any user of Messagecast can invite

"discoverable" users to their network. This will enable an quick and easy setup for any user to start collaboration.

- Geo-indexing: Agriculture applications require data to be georeferenced to help convert it into an actionable insight. As such, any data that is posted to MessageCast, whether by a person or a device, is geo-indexed and that information is made available as part of the message metadata.

- Multimedia support: As they say, a picture is worth a thousand words! This is no less true in agriculture applications than anywhere else. MessageCast supports the ability for a poster to attach a picture to any post. This feature allows for content-rich messages that are of significant value when one is messaging about something related to, say an ag machine, or a crop or fruit tree, pests, etc.

- Message topics and tagging: Messages on MessageCast are organized according to different *topics*, say, "*Operations in Field 123 of Any Name Farms, Australia*", "*Navel orangeworm issues in Fresno County, California*", or "*Cocoa commodity prices in Ghana*". In addition to the topics, MessageCast allows the message poster to add tags (e.g. `#navelorangeworm`, `#cocoa`) to their message. This helps with a quick understanding of the specific issue that the message is about, along with the ability for subsequent searching for a specific tag across various topics.

- Voting on messages: MessageCast supports the ability for users to upvote or downvote each message in a topic. In certain contexts, there is value in having other participants in the topic provide this type of reaction to a posted message. Say there is a chatroom that includes growers from a certain county or state and they have posted a few messages that include pictures of potential pest issues, along with certain recommendations for remedial actions from their trusted advisors and agronomists. Other users in that chatroom can use this feature to indicate their experiences as they relate to the infestation and the recommendation. There can be several other use cases for this feature.

- Timestamp: Perhaps not very unique to MessageCast, but nonetheless an important feature included is the detailed timestamp as part of every message.

- Integration with other AgStack services: The feature set above already offers significant value for agriculture data messaging. but the value increases many-fold when we connect this with other AgStack services like Asset Registry, User Registry, Weather Service, and many others that will be built in the future. Consider the following scenario where MessageCast is integrated with AgStack's Asset Registry. With this, say when a tractor in a field transmits data, say its fuel consumption, we not only know which field

the tractor is in (based on geolocation), but we can also determine the specific tractor (based on the asset ID from the Asset Registry). Other important AgStack service that MessageCast will be integrated with is the User Registry, which will allow for not only user authentication, but also provide a critical service of user-discovery (for those that have listed themselves as "discoverable") and visibility to other users and services.

- Integration/access to Agstack (and other AgStack registered 3rd-party) web services (connection to weather server, connection to form builder and library). By having a contect handle for a geospatial reference as the "geoid", Messagecast users will be able to subscribe to and post embedded information/data structures that is available through other services like the weather, soil, vigor, lab testing information, etc.

- Integration with other 3rd-party content, like the Ag-Rec Recommendation Server: AgStack is much more than the frameworks, tools, models, and services that are built. It is an entire innovation and collaboration platform for additional services that further extend its value. One great example of that is the Ag-Rec Recommendation Server. Ag-Rec is a framework where rural farmers anywhere in the world can have access to relevant, local agriculture information and advice on topics like crop production, agricultural land conservation, and pest management; similar to the cooperative extension networks currently available. The big difference is this wealth of information will be available through an online platform that can be accessed anywhere, anytime. You can find more about this integration from the Linux blog, Ag-Rec Improving Agriculture Around the World with Open Source Innovation.

- Integration/access to edge/IOT devices registered at AgStack (and 3rd-party registries): By registering the 3rd-party sensor APIs with MessageCast, users will be able to discover sensor devices and automatically "pair" with their cloud using messageCast. As such messageCast will act as an asynchronous guarantor of data for IoT devices - provided vendors pre-register their API into MessageCast.
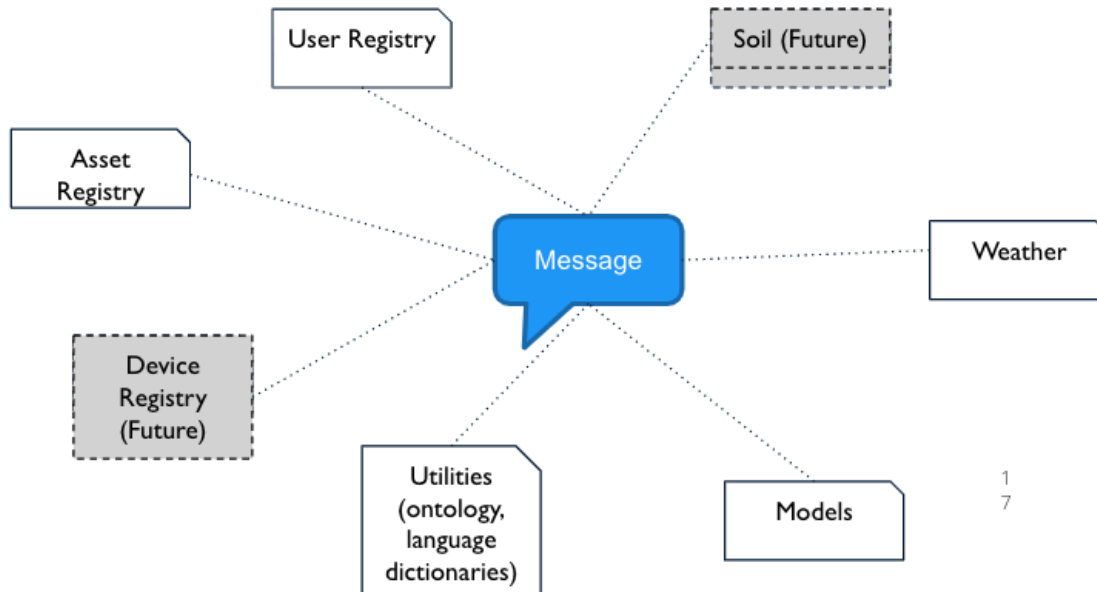
Auto-creates "AgNet" to turbo-charge AI/ML for Agriculture globally

Everywhere we look around us, we find the increased use of and value creation through Artificial Intelligence (AI) and Machine Learning (ML) technologies. This is no different for agriculture. However, one of the current challenges is the availability of a tagged-image dataset that is open and permissively-licensed. With the active approval from the poster, MessageCast can be leveraged to auto-create such a dataset of images. The images will be user-anonymized and structured based on the topics and tags. The dataset will be open and permissively-licensed per the Community Data License Agreement – Permissive, Version 2.0. As MessageCast usage scales across geographies, it accelerates the creation of this ever-growing dataset that users can leverage for their AI/ML solutions.

"Linux-like" downloadable for your own cloud/server

By packaging the server with pre-built component and enabling no-code customizations through a web interface, messageCast will be able to be easily downloaded as a package from AgStack's website onto your Linux server - in your own personal space on any of the major Public cloud providers. Users will be able to download, configure, connect with others and start

6

using this - without any deep knowledge of tech. Simply with click-and-go. AgStack will also provide a much more in-depth developer-centric configuration file that will be customizable through a much wider scope.

Some of the features of this ease / convenience will be:

- Python-first (requirements/dependencies included)
- Docker image to download and run in your own server / Public cloud. This is a VERY important point because this is a BIG plus for AWS/GCP/IBM/MSFT to get behind this.
- Documentation for how-to, including Use Cases
- Training materials (like videos)

Use Cases: The Goose that lays a lot of eggs

Given the vastness and complexity of the agriculture ecosystem, there are innumerable use cases for a software framework like MessageCast. We list a few here showing different value points across the agriculture ecosystem:

- Pest and disease collaboration between farmer, their trusted advisor or agronomist, and ag-extension service
- Supply-chain linkages between farmers and aggregators/buyers
- Researchers collaborating to characterize/phenotype a crop for breeding research
- Sensors connecting to other sensors or humans (IoT/Edge)

We look to the global community to contribute to MessageCast and leverage it to further extend these use cases.

Call to action

Innovation comes from everywhere! Linux Foundation helps companies and developers identify and contribute to the projects that matter. No industry matters to each one of us more than agriculture. Even if we are not active contributors to agriculture, we consume their products multiple times every day.

We invite you to bring your unique talent and contribute to AgStack in ways which only you can. Please find below a link to a Google Form. Once done, we will invite you to our Slack channel where you will not only learn about our projects, but also start contributing to them.