Microsoft Project Bonsai

Project Moab Demo Talk Track

Do this	Land these points
1. Explain Project Bonsai	• Project Bonsai is a low-code AI platform to speed the creation of AI-powered industrial automation. • Industry challenges well suited for AI include: 1. Multiple, competing, or changing optimization goals 2. Uncertain and variable process environments 3. Human Operator Limitation • Key Use cases include: 1. Manufacturing Line Optimization 2. Chemical Process Optimization 3. Building Energy Management • Bonsai is powered by several technologies, including simulation, machine teaching, and deep reinforcement learning (RL), all scaled in the cloud.
2. Explain why machine teaching is different to traditional AI	• Today's standard approach for machine learning is to provide machines with a lot of data and expect them to make predictions on their own, such as learn associations or find patterns. • However, the desire to use AI for more scenarios has grown. • Unlike machine learning, RL makes decisions – what is the optimal thing to do once a prediction is known. • For these scenarios, we don't expect algorithms to learn on their own: we depend on subject matter experts to break a problem into easier tasks and give AI models important clues about how to find a solution faster—we call this Machine Teaching.
3. Explain the Moab device (and Demo Joystick mode)	• Project Moab is a simple toy problem we use to quickly demonstrate how to design, train, and deploy a high-level AI agent, called a Bonsai Brain. • (Activate Joystick mode) Using the joystick, you can try to balance the ball on the plate manually. Notice that this takes a bit of practice. • (Activate Brain Mode) • Now let's see how the AI agent, called a Bonsai Brain, balances the ball. • Moab uses an upward facing camera to detect the position and velocity of a ball on the plate in terms of x and y coordinates. • The brain then adjusts the angle of the plate to balance the ball
4. Explain the problem we are trying to solve	• Machine teaching allows us to break down the system to design an AI using human-friendly statements. • In this case, we want to drive the ball to the center of the plate, and we want to avoid it falling off the plate.
5. Show the Inkling in the Bonsai Web UI	• Inkling is a low code, single purpose language used to define what (and how) to teach a brain using reinforcement learning. • As a visual language, inkling represents the reinforcement learning problem as a loop • One training iteration, is one step through this loop: The brain observes the state and takes an action, which is then simulated to produce an updated state. • Moab is simulated using a custom python sim. • Bonsai can connect to a prebuilt simulator, or in some cases, a data driven sim can be built from preexisting data.

6. Show the training	• The training is represented in a Performance plot that shows the goal satisfaction against the training iterations. • The three colors in this plot represent the individual objectives as well as the total goal satisfaction. • In addition to the goal satisfaction, we can see how long the brain training for, and how many training iterations it completed. • The brain trains until it reaches 100% or it does not improve for a predefined number of iterations
7. Talk about brain export	• Once the brain is trained, it is exported as a container for use outside of the bonsai platform. • There are several ways to deploy a brain. For Moab, we deployed a docker container to the raspberry pi, but alternatives include Azure IoT and Rest API. • The brain is running directly on the Moab hardware to balance the ball. • For more complex problems Brains can be used in combination with more conventional controllers such as PID. • Brains can serve as supervisory controllers, adjust controller gains or setpoints, or provide high level decision support
8. Link to relevant industrial use case	• Talk through one of the several available industrial use cases depending on the customer's industry.