Johnson & Johnson

NEXUS: A ROS 2 framework for orchestrating industrial robotic lines and cells

Assistant Principal Engineer in Robotic Applications
Dejanira Araiza Illan
20 October 2023

Automation is changing into advanced robotics

Traditional automation (last 70 years)

Program once, repeat forever

- High speed (PPM) and no operator
- Predictability & reliability
- Simple logic code with low complexity (e.g. ladder code)
- Well established hardware-based solutions
- Certification & standards
- Ecosystem of key market players

Advanced robotics (last 5 years)

Adaptive perception & manipulation with ML/AI
Coordination/cooperation, task mgmt. and autonomy

- ML/AI + complex algorithms
- Learning behaviours
- Software-based solutions
- High-level programming languages, high performance computation with GPUs
- Open-source software
- Not many standards, metrics, etc.

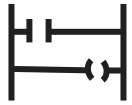
Challenges with traditional automation

Market demand for customization
Need for advanced automation for high-mix, low-volume use cases

Lack of agility, flexibility & reusability
Robotic applications cannot be easily modified or reconfigured

Long development time
Need to reduce effort and deployment costs

Hardware dependency & lack of interoperability
Technology transfer across geographies is challenging
Silo solutions, require custom integration for orchestration



Use of PLC logic
Control logic is hardcoded in PLCs
Supporting a “recipe” for a new product requires reprogramming the PLC

Availability of digital twins
No easy way to accurately simulate complex custom robotic solutions

We need an architecture for robotic platforms with...

Seamless orchestration & control

- At robotic workcell level
- At line level (multiple robotic workcells)

Modularity

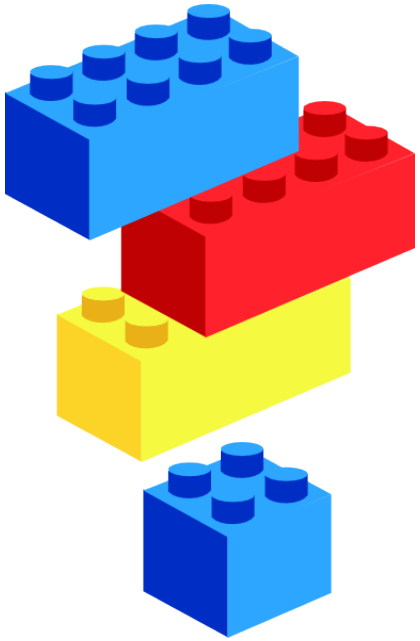
- Cells can be easily added, exchanged, modified

Flexibility & agility

- Easy reconfiguration of process flows

Scalability

- Lowers the cost for robotic cell adoption, reconfiguration, upgrade, etc.

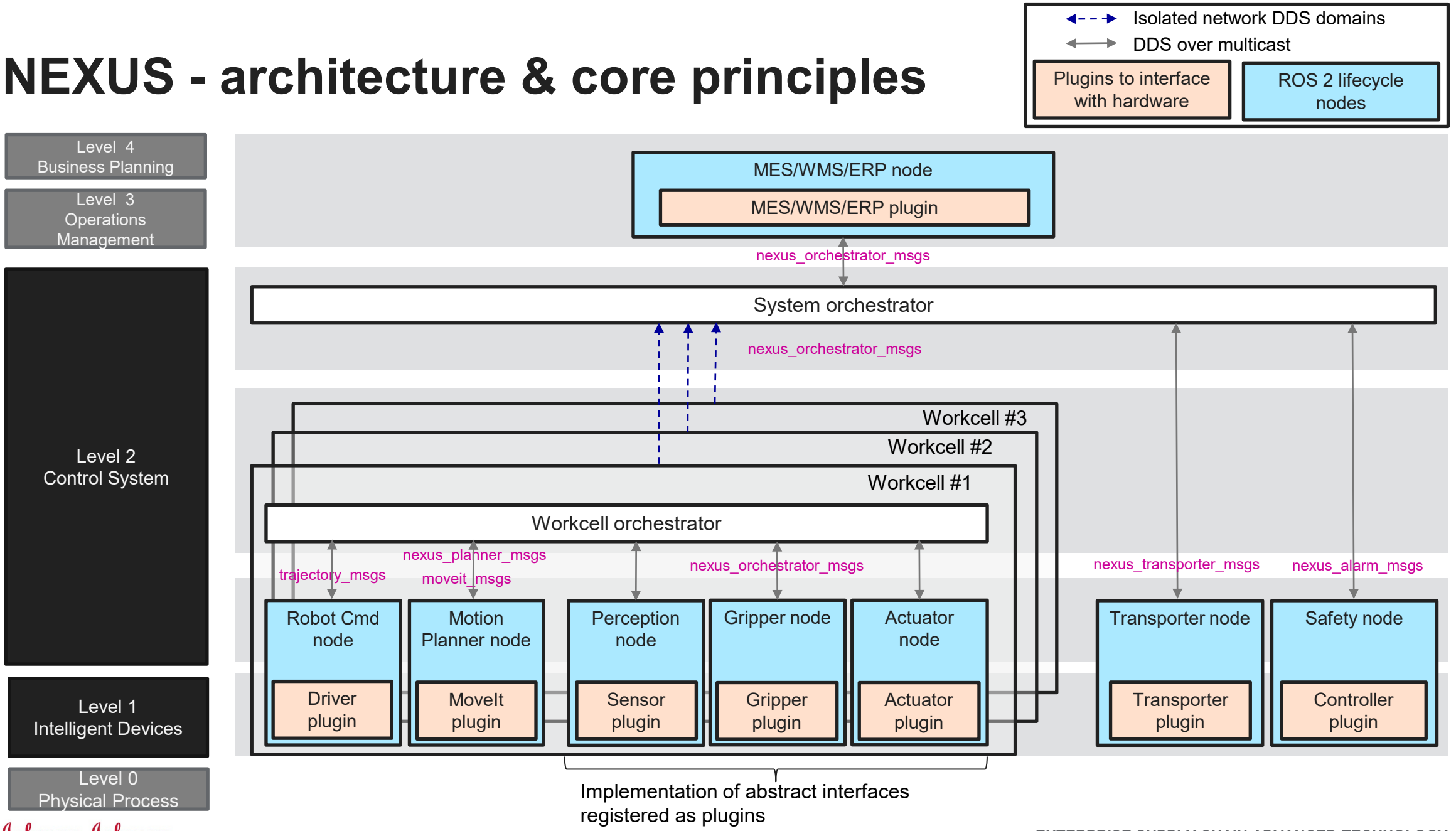


Line

Robotic workcells

Hardware components

NEXUS - architecture & core principles

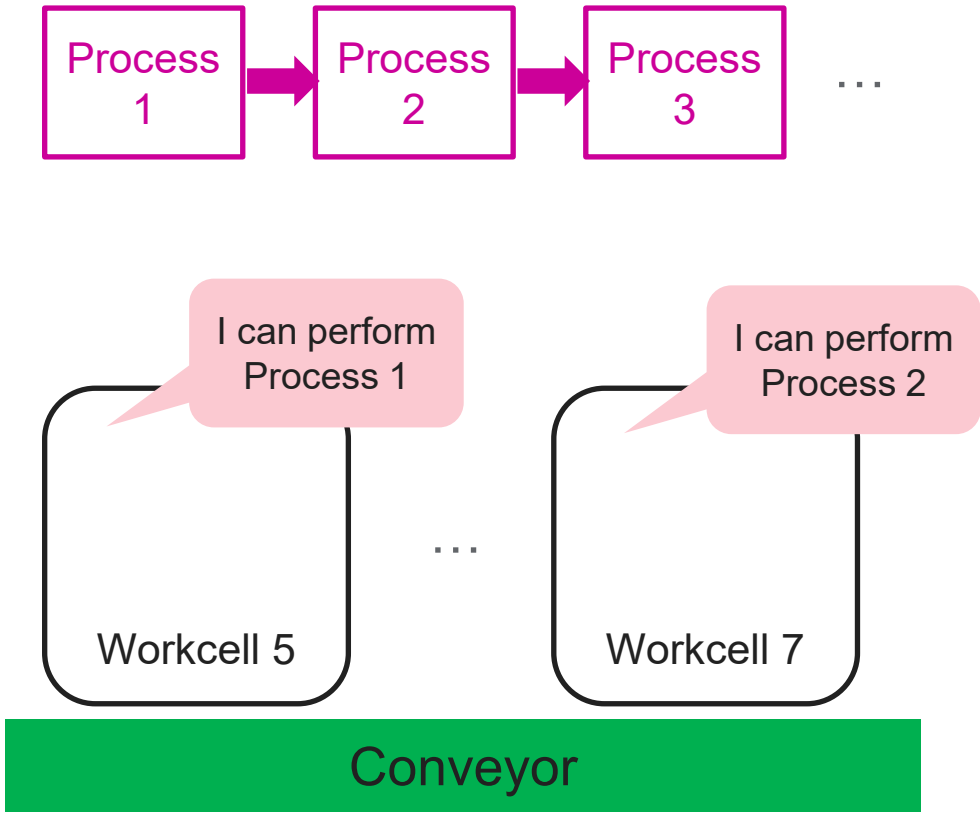


NEXUS - architecture & core principles

- Hardware - service providers
  - Orchestrator - coordinates activities among hardware.
- **Modularity** - Behavior trees to specify workflows and trigger hardware
  - BT nodes are capabilities/skills, available at the line or workcell level.
- **Flexibility** - Hardware registration, transmits capability to the orchestrator (e.g. transport, detect, move). Task capability is inferred based on registered hardware.
- **Agility** - Hardware agnostic logic
  - ROS 2 lifecycle (stateful) nodes
  - Runtime loadable plugins for hardware nodes
- **Scalability** - Hardware interfaces are standardized
  - Minimal network traffic by selecting endpoints between workcells and line orchestrator

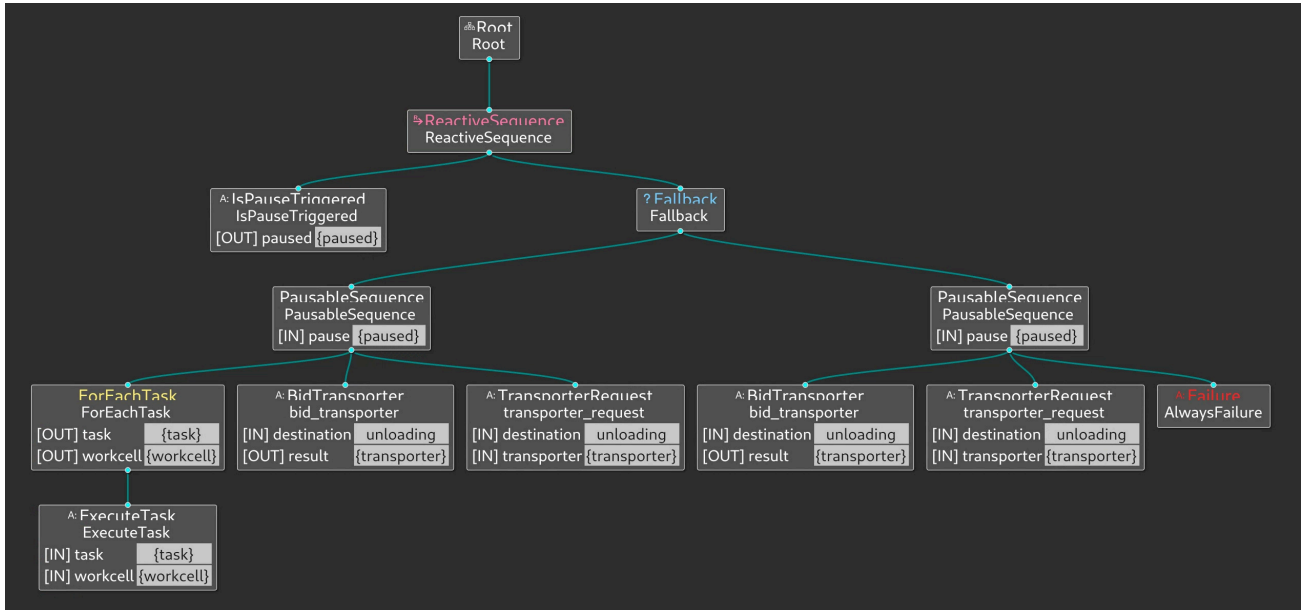
Intelligent recipe execution

- Recipe = one or more **process steps** in a line (different workcells)
- Recipes and process data stored in WMS/MES/ERP
  - Recipe execution dispatched (job request) to the line orchestrator
  - Enables **intuitive processes to onboard new recipes** (e.g. GUI) vs PLC programming
- Line orchestrator coordinates workcells and transporters to execute recipe (job)
  - Available workcells and transporters bid to execute a process – **self-organization**
  - Automatic queuing and buffering
- If a line and/or workcells reconfigure, the response to a recipe will be **adapted** by the line orchestrator

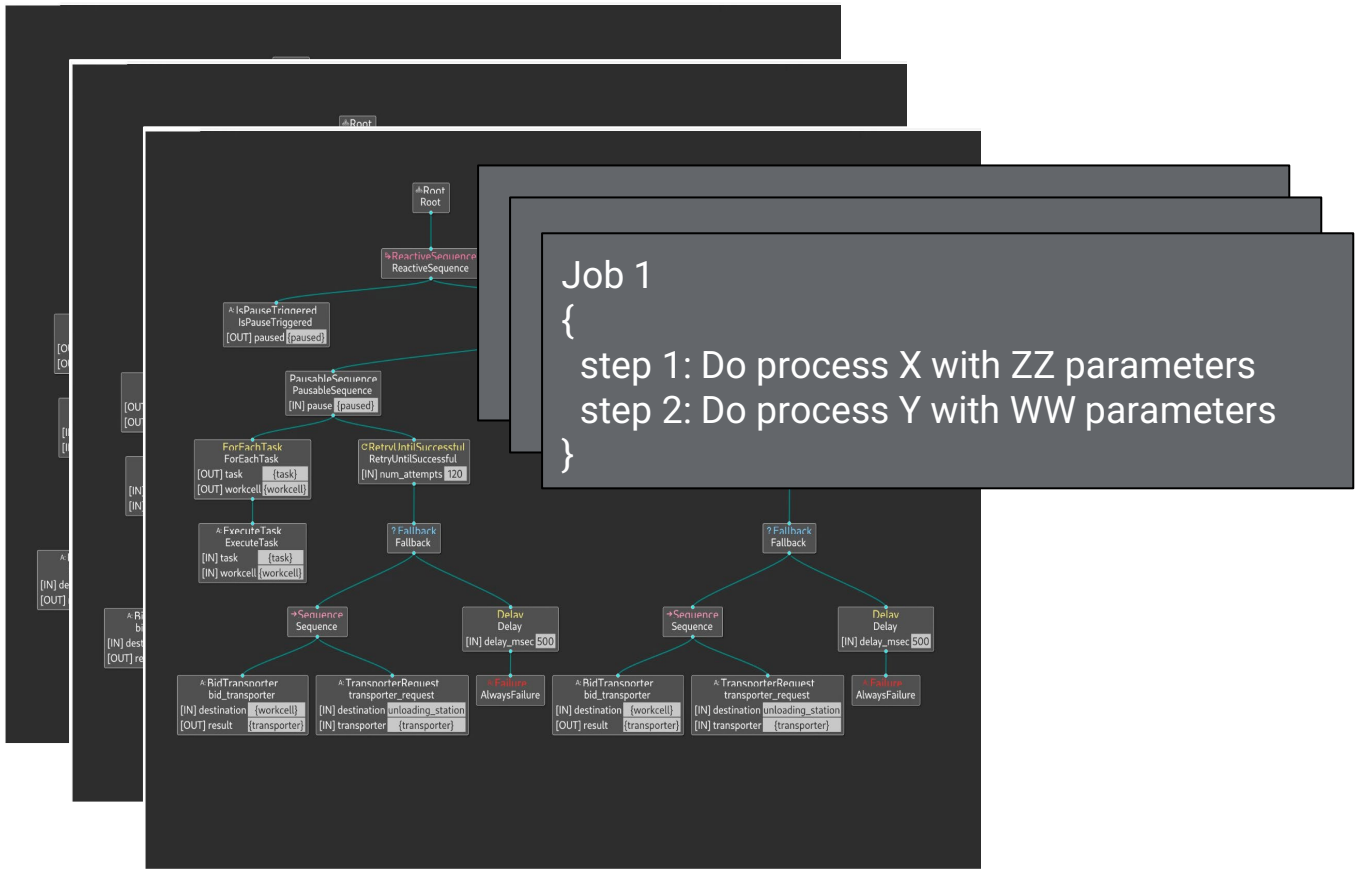


Behaviour Trees to specify process flows

- **Intuitive** representation of processes
- Enables **sequential and parallel** process execution
- Easy to **reconfigure**
- Applicable at line and workcell levels
- **Composable**
- Can be edited via GUI

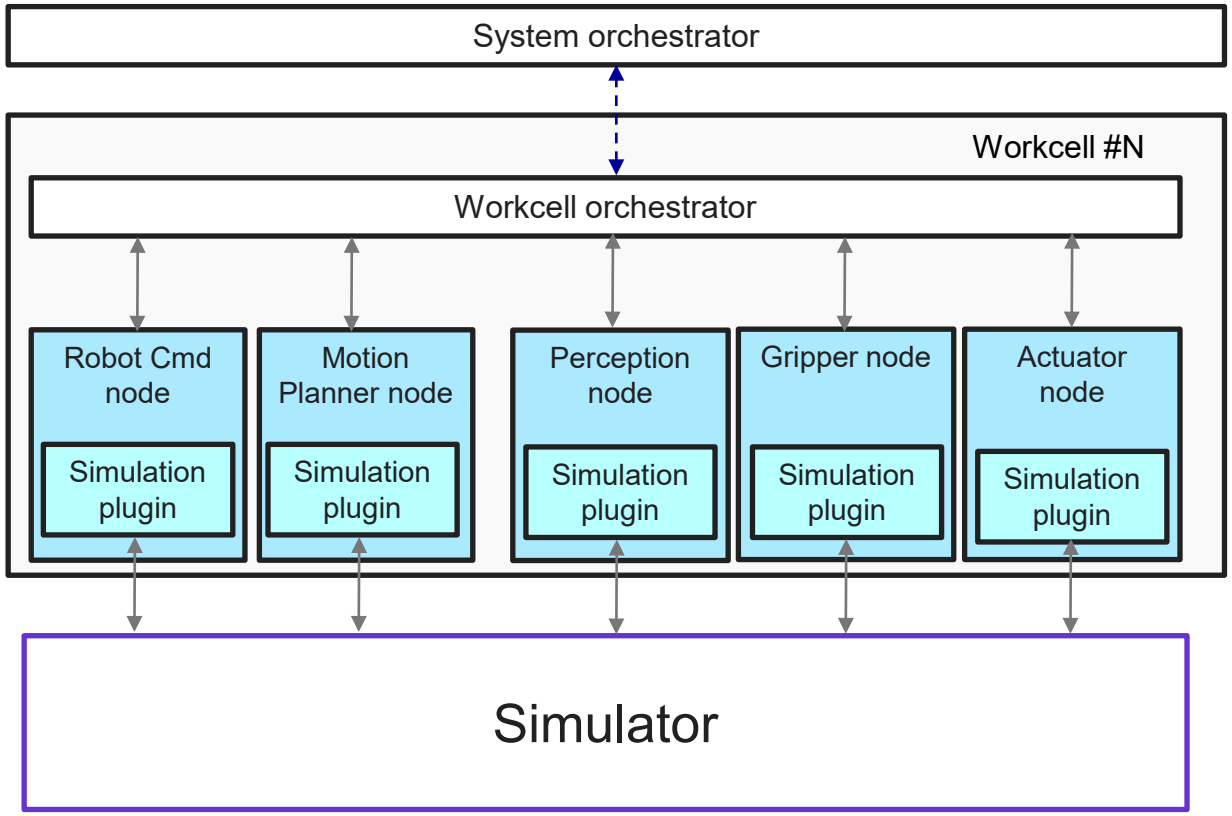


Coordination of processes and control in workcells



- Multiple jobs can be executed concurrently – **enables high-mix, low volume**
- Steps between workcells are synchronized using **ROS services**
- **Data propagates** from one workcell to other workcells automatically
- Workcell orchestrators coordinate hardware from different vendors through **ROS as a middleware**

Simulations to test exhaustively



- Running the orchestrators and nodes (code) with simulated hardware
- Simulation plugins can interface with Gazebo, RViz, etc. or customized hardware models
- Mixes of hardware and simulation are possible for individual component testing

Adopting NEXUS

Workcell setup

- Plugins for hardware components
- Algorithms with parametrization for adaptation to recipes
- Build behaviour trees for workcells - codes for each process step

Line setup

- Building behaviour trees for line orchestrators

Recipe preparation

- DB of recipes to create jobs in WMS/EMS/ERP – lists of processes, steps and parameters

Open challenges

ROS

- Lack of support for open-source **drivers** in ROS 2 (robots, sensors, industrial equipment)
- **ROS 2** capabilities and optimization are work in progress

Other

- Lack of an ecosystem of **system integrators who work with ROS.**
- **Certification** – how to?
- Adoption of new tools & algorithms – frequent **upgrades** might be required
- Cybersecurity – **authentication and data encryption**

Thank you

For more details and information please contact us

daraizai@its.jnj.com

Johnson & Johnson