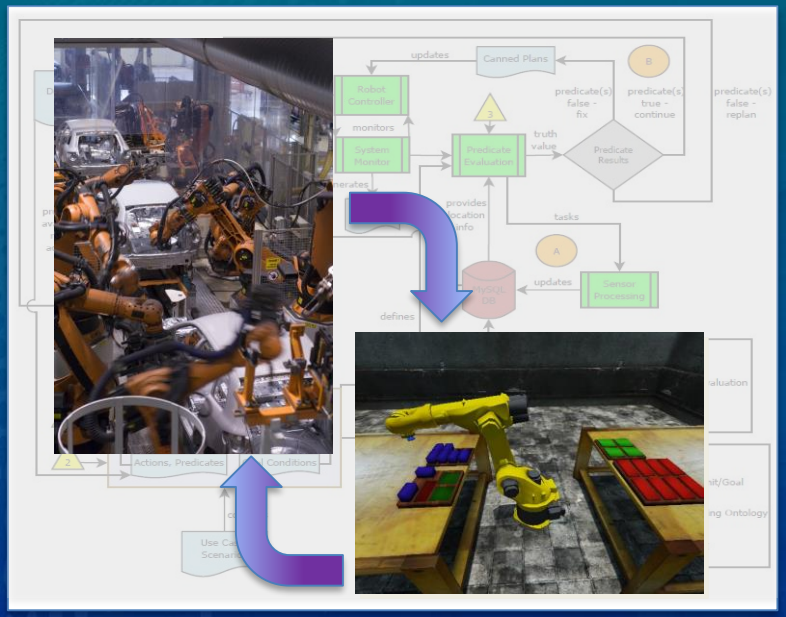
The Canonical Robot Command Language (CRCL)

Presented by: Craig Schlenoff (NIST)

Agility Performance Of Robotic Systems



Measurement Science Challenge

Develop performance metrics, **information models**, test methods, and **protocols** to enable manufacturers to assess and assure the agility performance of their robot systems

Potential Impact

- Lot size 1 assembly in automated lines
- Reduced line down time due to programming
- Less human intervention required due to assembly errors

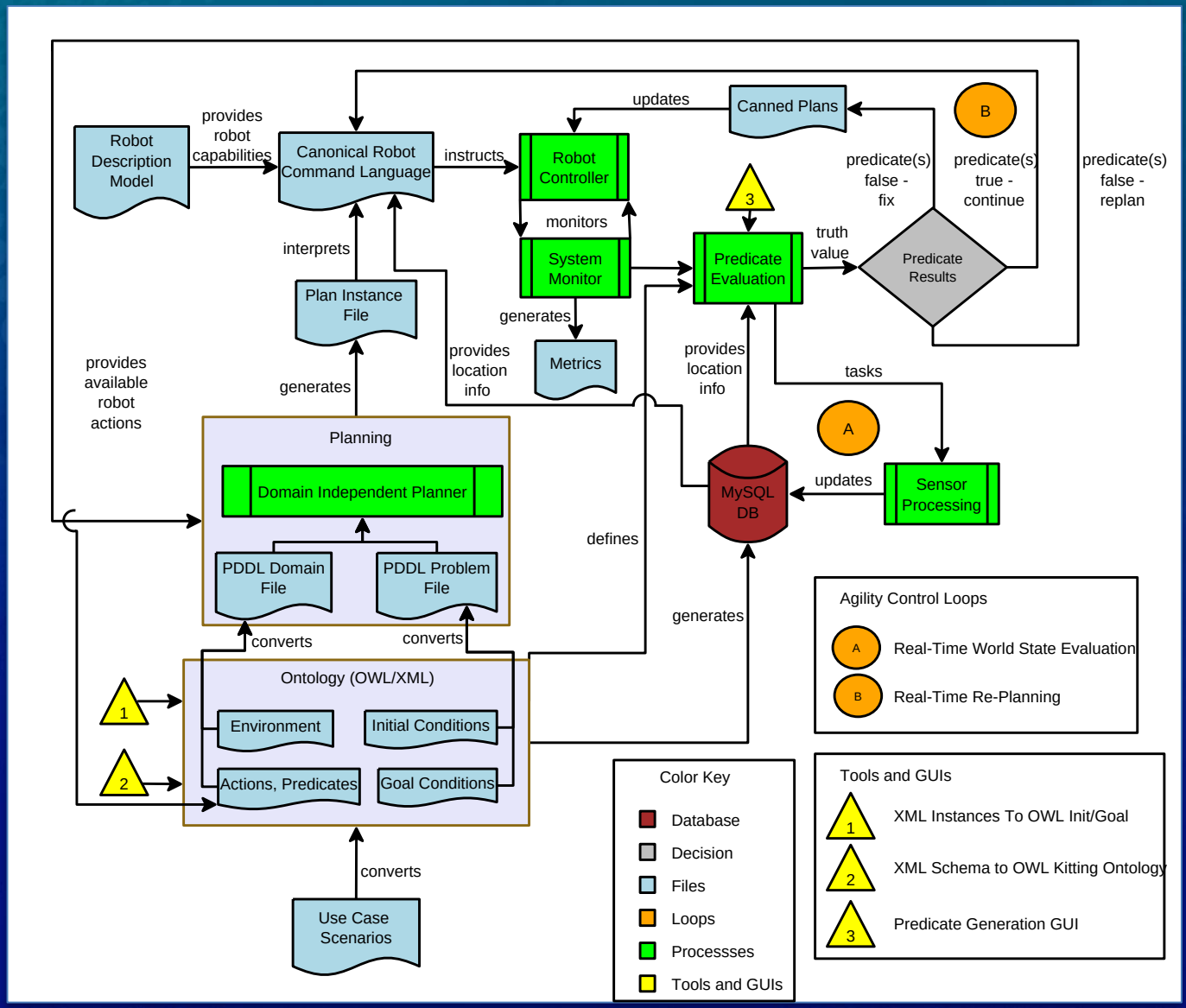
Major Outcomes

- Metrics and test methods to measure robot agility
- A robot description model allowing robot vendors and manufacturers to clearly and accurately characterize their robots
- **Methods, protocols, and information models to allow for dynamic tasking/re-tasking**
- An integrated agility framework enabling manufacturers to assess and assure the agility performance of a one or many robots

What is Robot Agility?

- Hardware agility
  - How can different hardware configurations affect a robot's ability to accomplish a variety of tasks?
- Software agility
  - How well can a robot adapt/respond to task failures?
  - How well can a robot re-plan when a new goal is provided to it?
  - How can we allow for interchangeability of robots without the need for reprogramming?

Big Picture


Robot Description Model
provides robot capabilities
Canonical Robot Command Language
instructs
Robot Controller
updates
Canned Plans
B
predicate(s) false - fix
predicate(s) true - continue
predicate(s) false - replan
3
monitors
System Monitor
Predicate Evaluation
truth value
Predicate Results
interprets
Plan Instance File
generates
Metrics
provides location info
provides location info
tasks
provides available robot actions
generates
Planning
Domain Independent Planner
PDDL Domain File
PDDL Problem File
converts
converts
A
MySQL DB
updates
Sensor Processing
defines
generates
Ontology (OWL/XML)
1
2
Environment
Initial Conditions
Actions, Predicates
Goal Conditions
converts
Use Case Scenarios
Agility Control Loops
A
Real-Time World State Evaluation
B
Real-Time Re-Planning
Color Key
Database
Decision
Files
Loops
Processses
Tools and GUIs
Tools and GUIs
1
XML Instances To OWL Init/Goal
2
XML Schema to OWL Kitting Ontology
3
Predicate Generation GUI

Canonical Robot Command Language

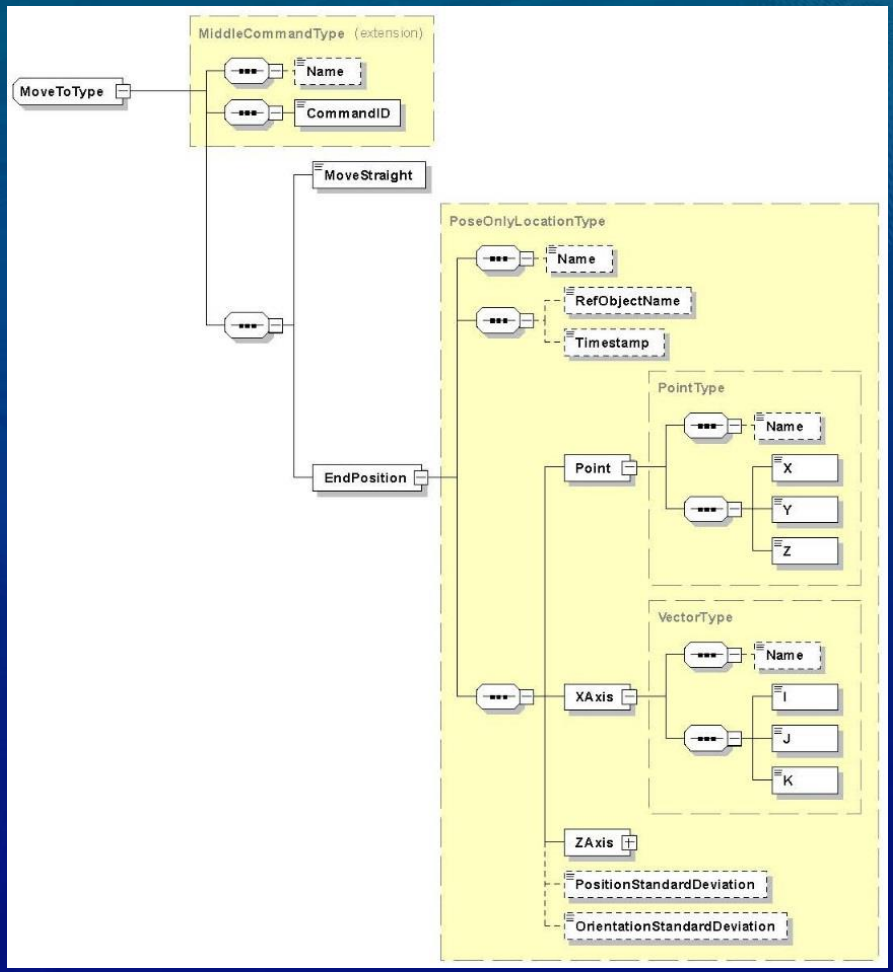
- A low-level messaging language for sending commands to, and receiving status from a robot.
- Provides basic commands that are independent of the kinematics of the robot that executes the commands.
- Formal definition that allows compliant code to run on multiple robots without change
- Ability to utilize set of commands on different vendor's robots with same results
- Implemented as XML Schema

Sample Commands

- Administrative Commands
  - Initialization
  - Run program
- Motion Commands
  - Dwell
  - Linear movement related
    - Move through
    - Move to
  - Joint related
    - Control mode (position, force, torque)
    - Actuate joint(s)
    - Configure joint(s) report
  - Open/Close tool changer
  - Screw motion
  - Stop motion
- Data Passing Commands
  - Get status
  - Message
- Set Parameter Commands
  - Set (acc, speed, units, tolerance)
  - Set end effector operation
  - Set parameters (robot, end effector)

Example Command

- MoveTo allows robot motion to single Cartesian point
- Composed of multiple schema elements
- Allows:
  - Requirement for straight-line motion
  - Specification of allowed deviations
  - Specification of 6-DOF point



CRCL In Action

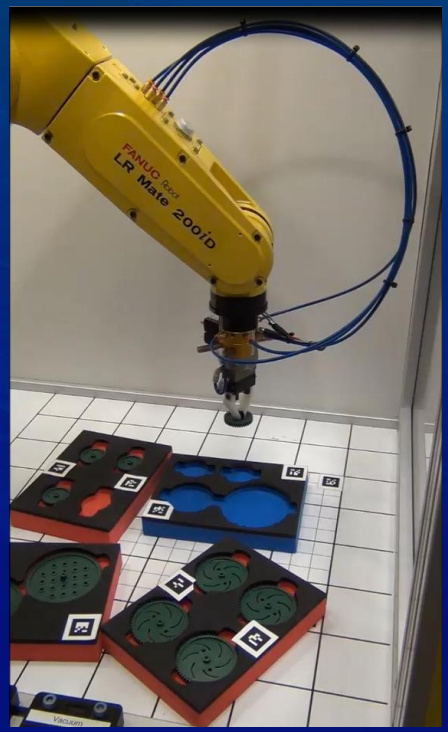
Applying CRCL at Georgia Tech

Applying CRCL at NIST

Canonical Vision Command Language (CVCL)

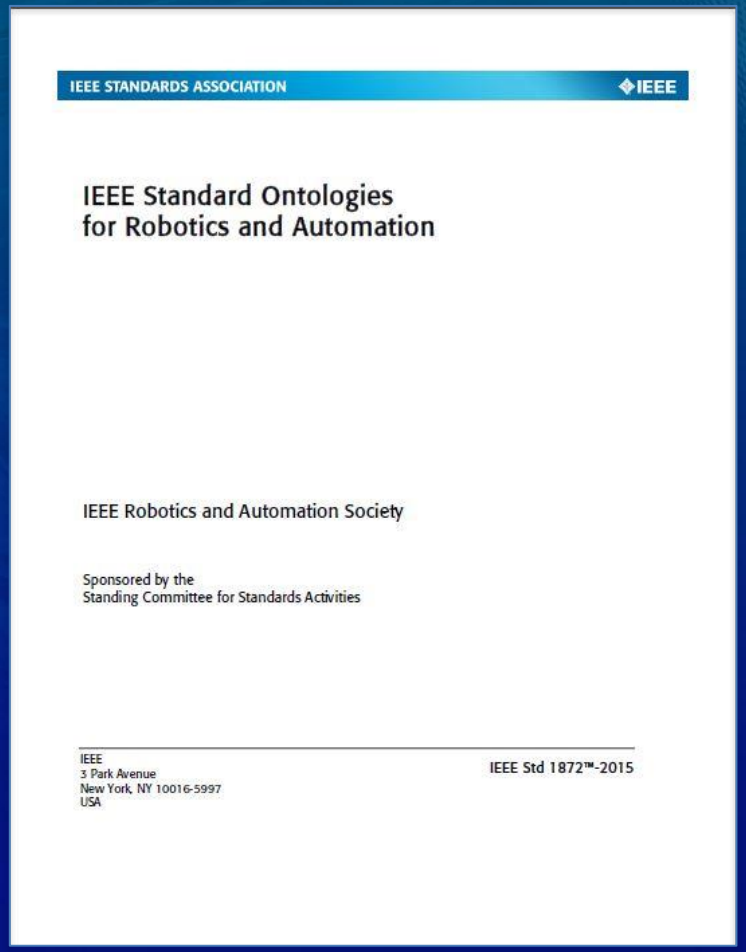
"Extensions to the Canonical Robot Command Language to be able to command a sensory processing system to locate objects for kit construction."

- CVCLInitCanonType – initialize vision system
- CVCLLookForObjectType
  - Input: SKU, part tray
  - Output: Named location of object
- CVCLLookForGraspType
  - Input: SKU, end effector type
  - Output: selection of a particular grasp offset to match the SKU
- CVCLAddPoseType
  - Performs pose math to add two different poses together



IEEE Ontologies for Robotics and Automation (ORA) Working Group

- Goal: To develop a standard ontology and associated methodology for knowledge representation and reasoning in robotics and automation, together with the representation of concepts in an initial set of application domains.
- 166 members across 23 countries
- First IEEE Standard: Standard for Ontologies for Robotics and Automation (P1872-2015)
- Future focus on industrial robotics efforts:
  - CRCL
  - Industrial robot ontology


IEEE STANDARDS ASSOCIATION IEEE

IEEE Standard Ontologies for Robotics and Automation

IEEE Robotics and Automation Society

Sponsored by the Standing Committee for Standards Activities

IEEE
3 Park Avenue
New York, NY 10016-5997
USA

IEEE Std 1872™-2015

Future Work

- Work with the community to validate the set of CRCL commands and add more as necessary
  - CRCL is in the process of being made available through a ROS-I repo
  - Work with members of the IEEE ORA Working Group
- Continued development of sample implementations involving more sophisticated assembly operations

Contact Info

Craig Schlenoff
Group Leader
NIST
Gaithersburg, MD
craig.schlenoff@nist.gov

Stephen Balakirsky
Senior Research Scientist
GTRI
Atlanta, GA
stephen.balakirsky@gtri.gatech.edu