Fraunhofer IPA

Fraunhofer Institute for Manufacturing Engineering and Automation IPA

Desktop and Web based textual and graphical toolchain for kinematics modeling

—

ROSCon 2022 Kyoto, Japan

Composing kinematic models (URDF + Xacro)

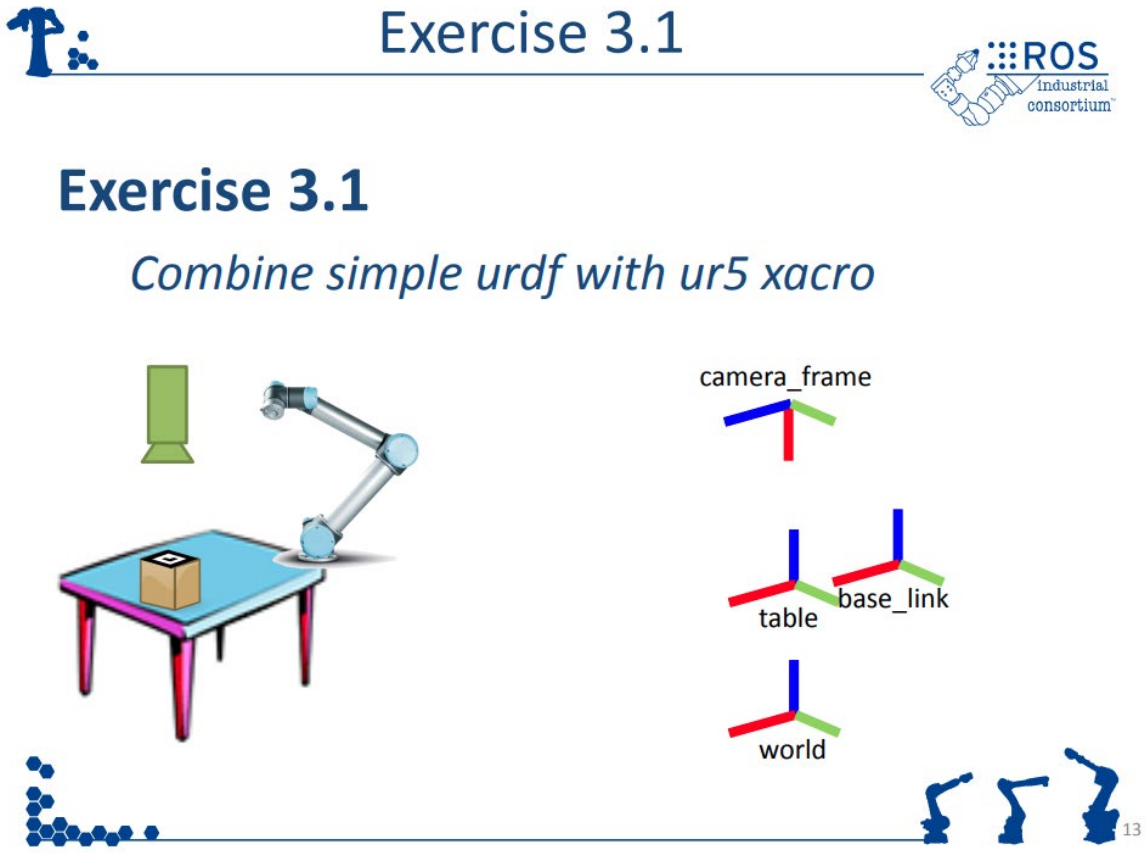
The pain of creating kinematics models manually

Typical steps to compose Xacro for your application

- Install dependent description packages
- In the composition Xacro file, include dependent xacros
- Instantiate macros with respective parameters
- Create joints between instantiated components
- Launch your application with the Xacro set as `robot_description`

Some tools to help developers

- `check_urdf`
- `urdf_to_graphviz`
- `roslaunch urdf_tutorial display.launch` (or its ROS2 counterpart)
  - And more recently URDF Preview VS Code Extension



Difficulties in composition with XACRO

Empirical study from ROS Answers

- Lacking Xacro Pipeline Awareness
  - Xacro files need to be converted to urdf before being consumed
- Adding and Configuring Hardware
  - Adding a gripper to a manipulator or (re)configuring simulated sensors
- Xacro Language Usage
  - How to pass arguments to nested Xacros
- Joint, Link, and Transform
- Syntax and Parser Errors
- Complex or Multi-Robot Systems
  - Xacros are organized hierarchically
- Dependencies Missing or Conflicting
  - Package that contains Xacro file with `<xacro:include>` may not be installed

Proposed solution:

"Nearly all Xacro-related errors manifest as text-based compiler messages, which have been shown to be difficult for developers to interpret. Therefore, we suggest that robot software tool builders focus on an Integrated Development Environment (IDE) extension or improvements to error messages generated by the xacro program. For the IDE extension, we suggest a targeted plug-in specifically for the Xacro format, linked to the xacro program on the backend. Such an extension could statically check Xacro files and provided targeted, Xacro aware feedback while the Xacro file is being edited."

Source: "Understanding Xacro Misunderstandings"
https://doi.org/10.48550/arXiv.2109.09694

Need for a kinematics IDE

Model-driven engineering (MDE) approach

- URDF and Xacro are already MODELS
- MDE facilitates creation of nice developer tools to work with models

Advantages of MDE

- Less error prone, thanks to code generation
- More cost-efficient because remove a lot of copy-paste work, thanks to code generation
- Introspection at design time, thanks to the validation module

Why are we not using MDE tools?

Kinematics IDE

Concept

Compose and visualize kinematic models with formal meta-models at the core

Extraction

Extract elements from URDF or other formats and generate the kinematics DSL

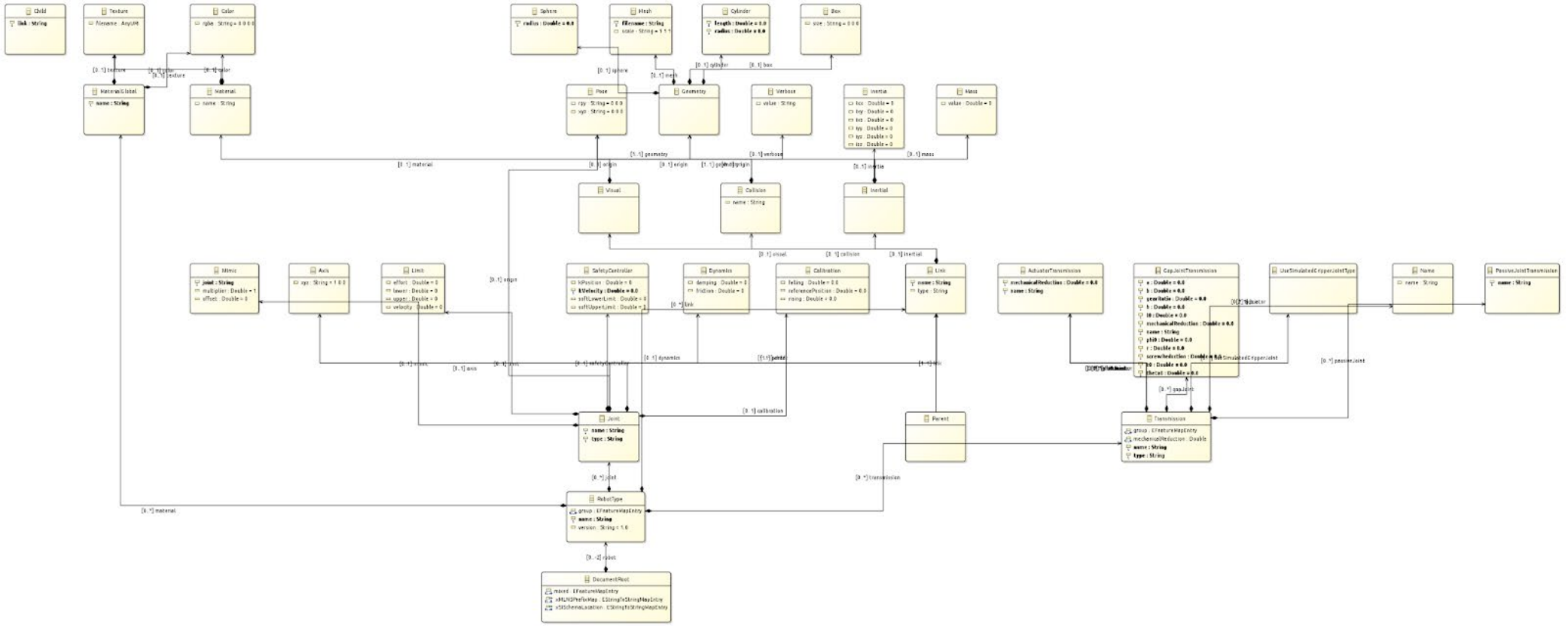
Validation

Validate the kinematic models (e.g. `<limit>` of `<joint>` can only have `"lower"` and `"upper"` attributes IF `"type"` of the joint is not `"continuous"`)

Generation

Generate composed XACRO / URDF or other formats

Ecore based formal kinematics model

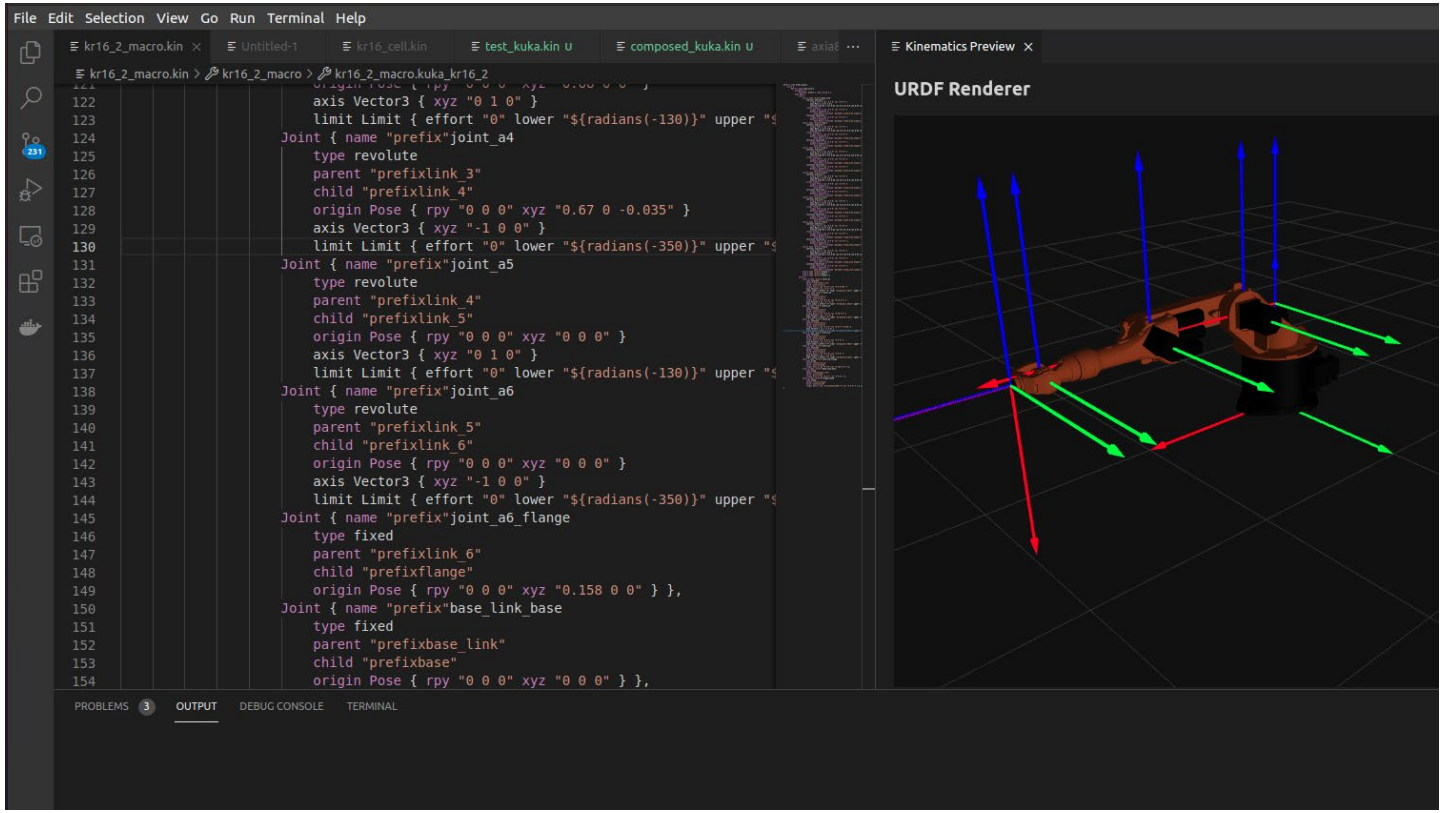


Reference: https://bit.ly/3MICBjA

Kinematics IDE

Text editor

- Based on Eclipse Xtext
- A generic DSL is developed to support multiple formats
  - Current version is similar to Xacro + URDF
- Provides typical IDE features
  - keywords highlighting, code completion, navigation to declaration of a symbol, rename refactoring
- Developed as a VS Code plugin which is seamlessly integrated as Theia extension
  - Communicates with Xtext server through language server protocol
- A Python-based model generator parses Xacro files and generates the DSL file

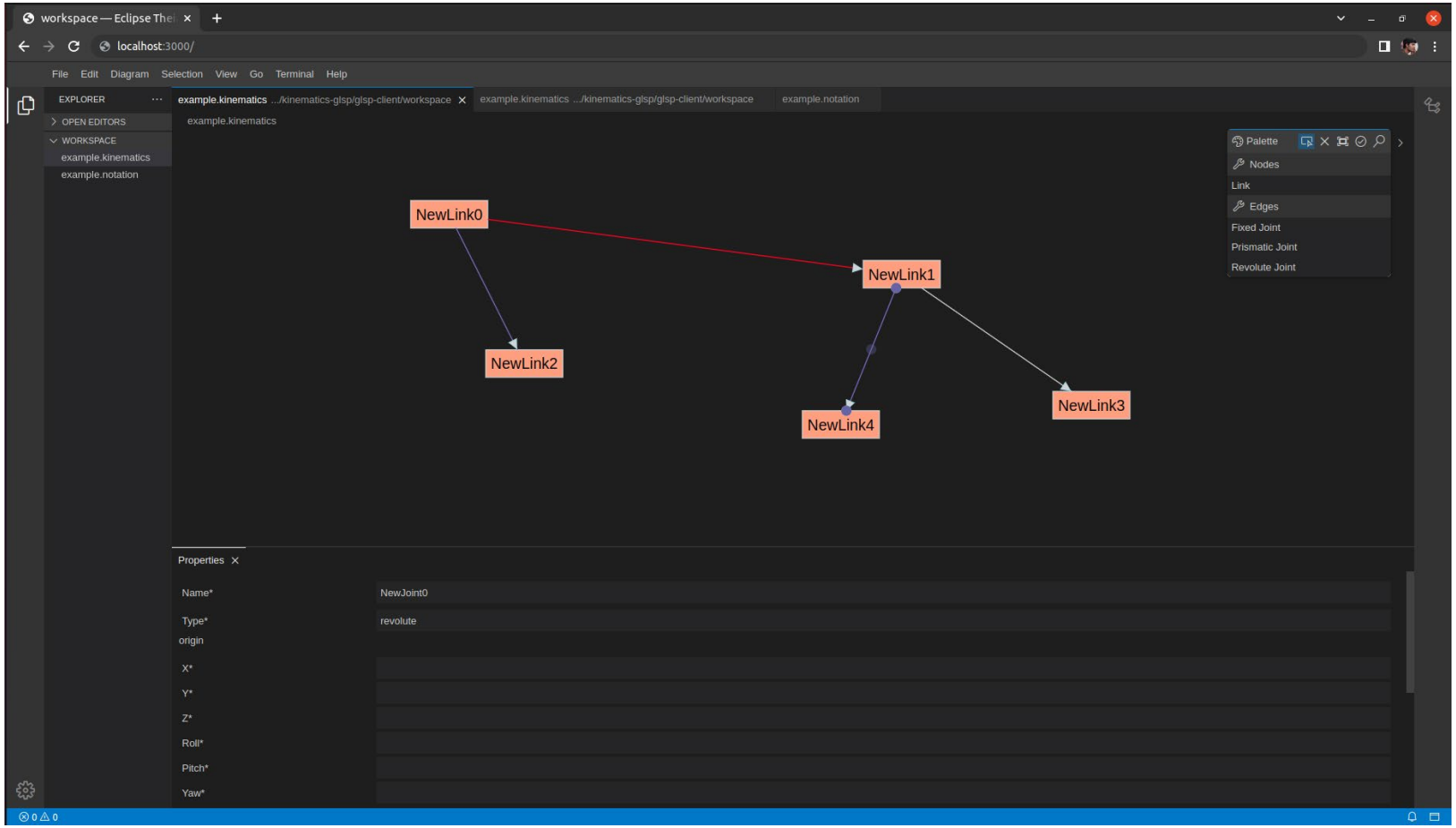


https://bit.ly/3Tip2ty
https://bit.ly/3VGxp3H
https://bit.ly/3SnZmuA

Kinematics IDE

Graphical editor

- Based on Eclipse GLSP
- GLSP Server provides the language-specific smarts and editing capabilities
- Client provides rendering capabilities
- The same Ecore model can be used at the backend as in the text editor
- Properties of kinematic elements can be edited through a property window
- Currently only web (Theia) version is available
- [WIP] compose existing models and generate URDF file



Source: https://bit.ly/3TBFIME

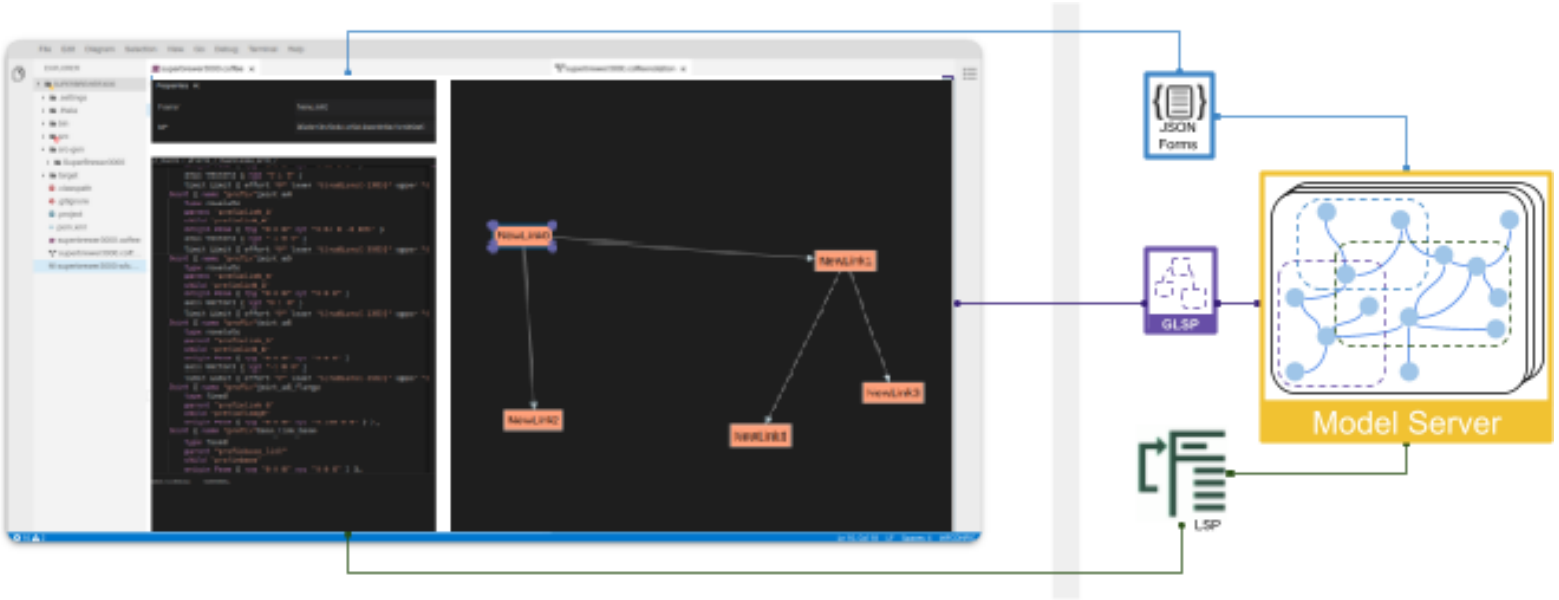
https://bit.ly/3eNpxwT

Kinematics IDE

[WIP] Synchronized text and graphical editors

Model server:

- Keeps a central version of your data instance on the server
- Clients (Xtext server, GLSP server, etc) access this data
  - Retrieving parts of the data model, triggering changes, subscribing to changes



https://bit.ly/3MLwvyP

Future plans

Tackling limitations of URDF components with MDE?

- URDF “specification” refers to outdated ROS Wiki pages
  - XSD-based prototype available since 2013, but not been used
- Specification is hardcoded in the parser
  - Parser needs to be updated as per changes in schema
  - Ideally, parsers should validate URDF syntax against XML schema
- Extensions to the schema are not possible (easily)
  - `<sensor>` proposal is an PR since 2016
  - `<capsule>` proposal closed after being open since 2016

Can MDE solve these problems?

- Generate URDF class headers from schema

  (Note: EMF generates Java classes)
- Generate documentation and parser along with new schema versions
  - Xtext internally uses Antlr4 which *can* be used to create standalone parser

  (Note: the DSL should match Xacro / URDF syntax)

Thank you for your attention!

Contact

Harsh Deshpande
Fraunhofer IPA – Robots and assistive systems
Phone +49 711 970-3737
harshavardhan.deshpande@ipa.fraunhofer.de

Fraunhofer IPA
Nobelstraße 12
70569 Stuttgart
Germany
www.ipa.fraunhofer.de

Fraunhofer
IPA

Fraunhofer Institute for Manufacturing Engineering and Automation IPA

Future is our product

Sustainable. Personalized. Smart.