ROSIN QUALITY ASSURANCE INITIATIVES FOR ROS

Adam Alami & Zhoulai Fu

IT University of Copenhagen

rosin-project.eu

This project is funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 732287

ROS Quality Assurance Working Group

ROSIN Quality Assurance Initiatives

Iterations	ROSIN Initiatives	No. Of vote points
Iteration 1	Make ROS packages quality visible.	42
	Appoint ownership	32
	Energize the code review process	32
	Implement a code scanning method and tool	31
	Maintenance issues	31
Iteration 2	Energize Continuous Integration	31
	Quality Hub website	30
	Formalize the code ownership process	29
	Onboarding process for core and non-core community members	29
Iteration 3	Model-in-the-Loop testing	28
	Implement a continuous improvement process	26
	Automated unit test generation	26
	Quality Discourse	23
Iteration 4	QA promotion events	19
	Model Driven Development	18
	#ROSQA	15

ROSIN Quality Assurance Initiatives

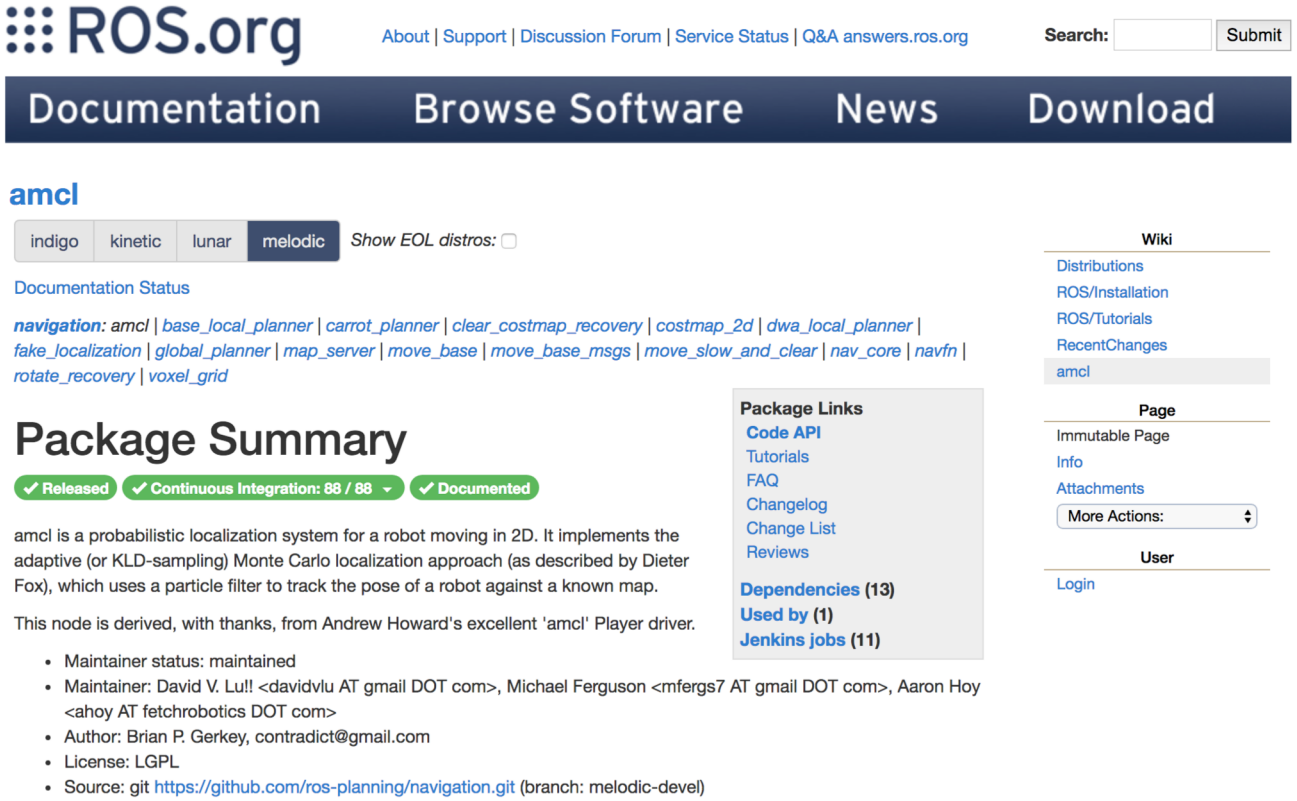
Problem/Opportunity Statement	Initiatives				
	Name	Description	The change impact magnitude	Expected Community Impact	
				QA Feature/Capability	Community Impact
Lack of centralized source for community quality assurance practices, knowledge and collaboration.	Quality Hub website	A single source of truth for ROS QA knowledge, practices, tools and methods.	Minor	QA knowledge sharing.	• Inspire people to share knowledge and experience. • Foster knowledge sharing behavior.
	Quality Discourse	QA Forum	Minor	QA Collaboration platform.	Inspire people to collaborate on QA themes and issues.
Absence of ownership for QA practices.	QA ownership	Appoint ownerships for QA practices, tools and infrastructure.	Medium	Ownership	Motivate and encourage ownership behavior.
Inconsistent practice of code review	Energize the code review process	Code review is an existing process; unfortunately, it is loosely implemented and practiced. The aim of this initiative is to bring this practice back to ROS QA core quality practices. Review the current process. Update the current process to reflect the SE practices. Implement it in ROS and ROS-I.	High	Code review process	• Higher software quality • Knowledge sharing • Early detection of defects
Deviation from software engineering and industry practices.	Code scanning	Implement a code scanning method and tool.	High	Code scanning tool and process	New software engineering practice in the community
	Energize Continuous Integration	This initiative is to review and enhance the current implementation of the current Continuous Integration (CI) services.	High	Continuous Integration Service.	New software engineering practice in the community

ROSIN Quality Assurance Initiatives

- Make ROS packages quality visible.
  - Description of the initiative:
    - Create a process/tool where packages quality can be measured, assigned and displayed.
  - Intent:
    - Make ROS packages quality visible
  - Status:
    - Phase I: The CI badge is implemented.
    - Phase II: The quality dashboard is work in progress.
  - To do(s):
    - Integrate the GUI of the quality dashboard to Haros.
    - Implement user rating

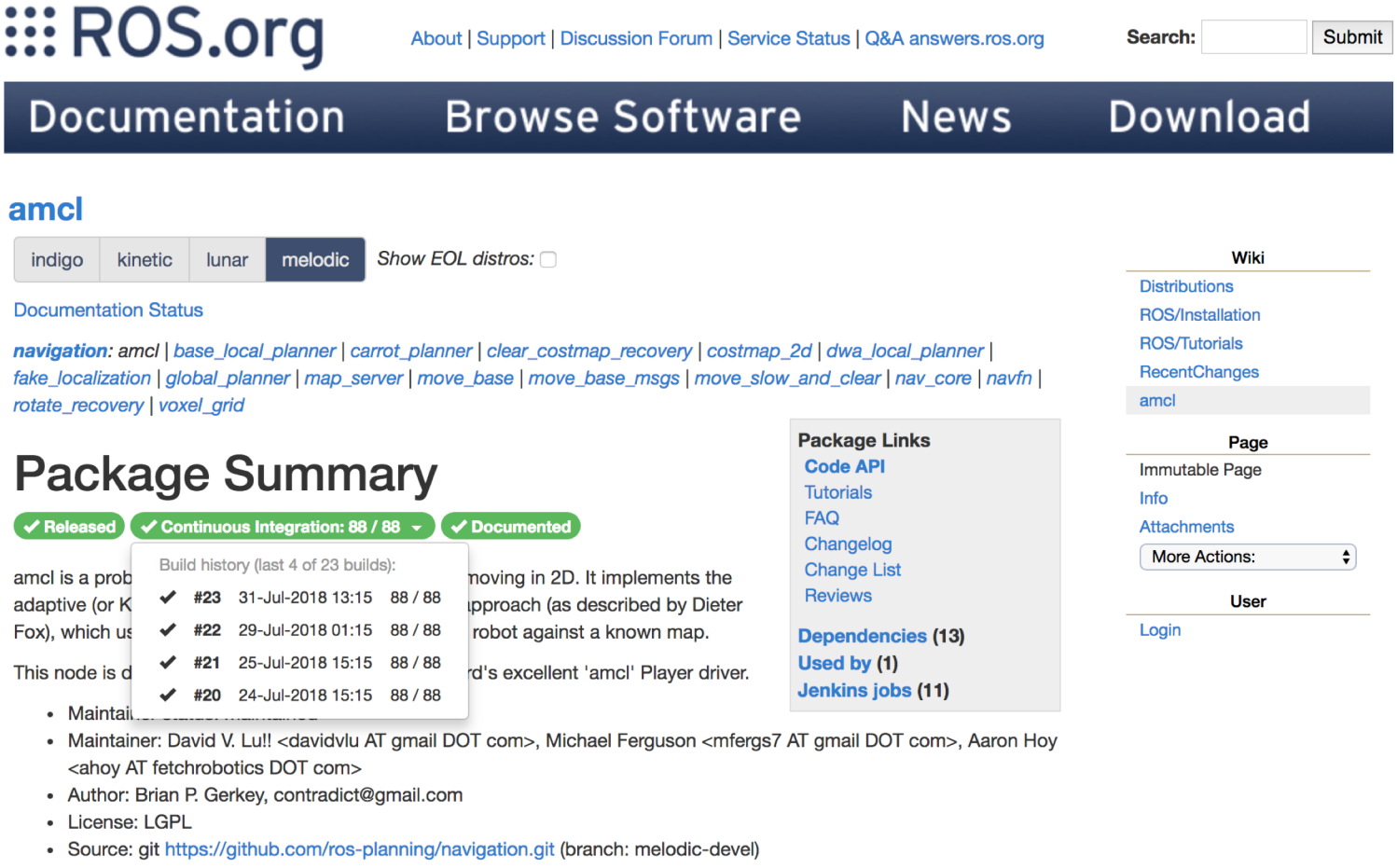
ROSIN Quality Assurance Initiatives

- Make ROS packages quality visible.



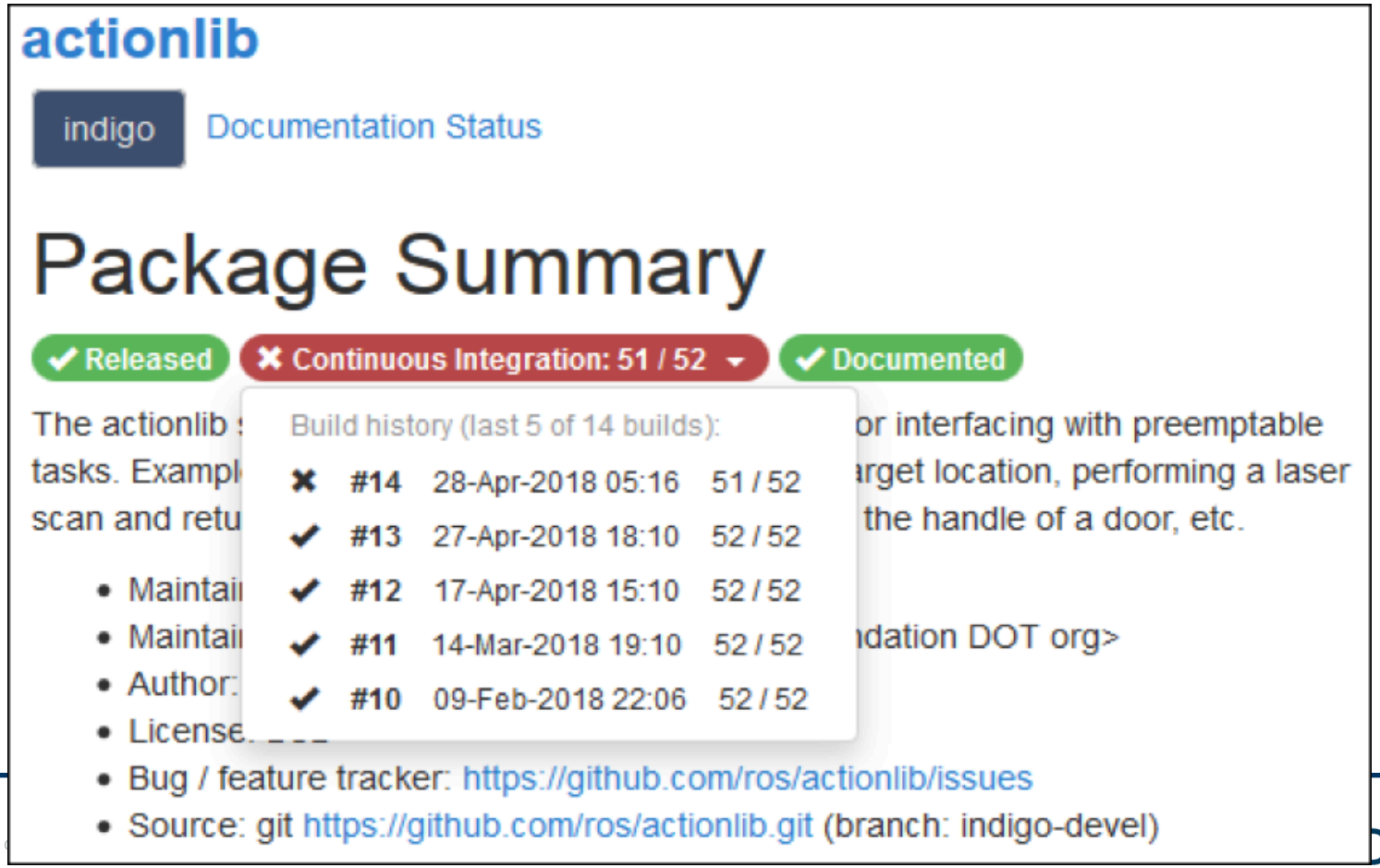
ROSIN Quality Assurance Initiatives

- Make ROS packages quality visible.



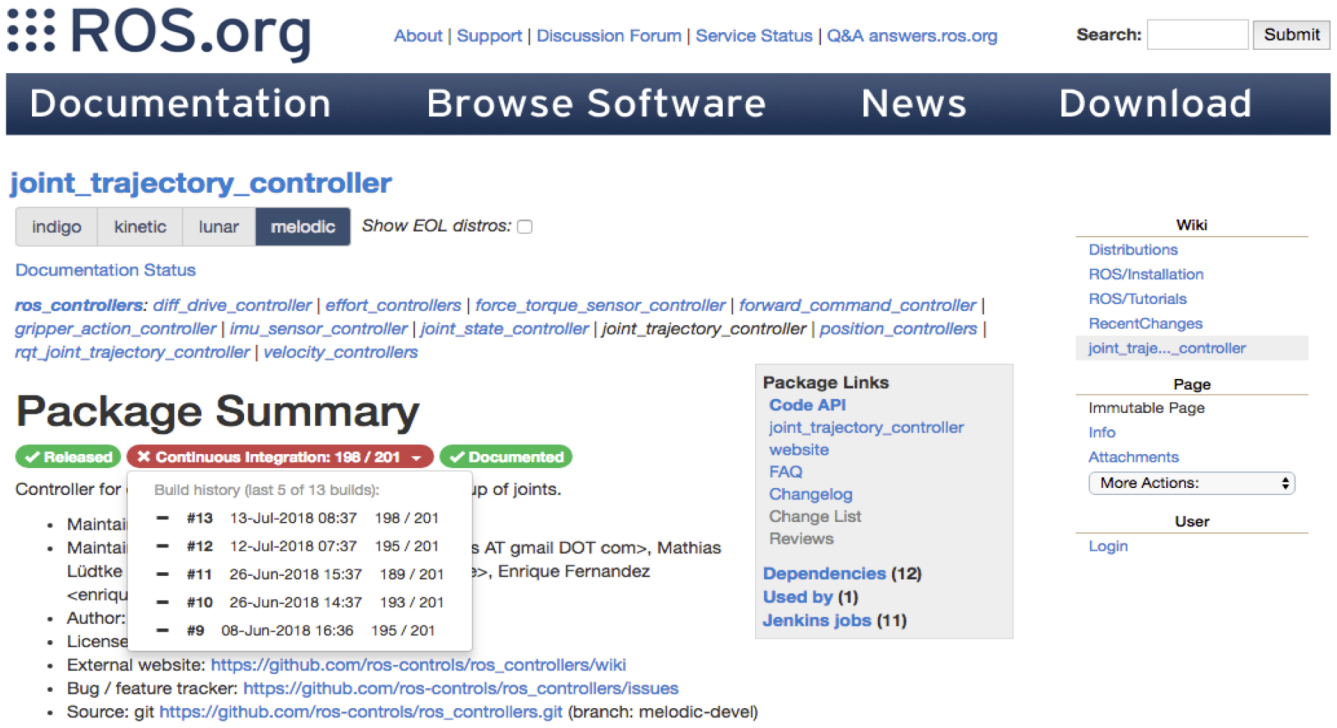
ROSIN Quality Assurance Initiatives

- Make ROS packages quality visible.



ROSIN Quality Assurance Initiatives

- Make ROS packages quality visible.

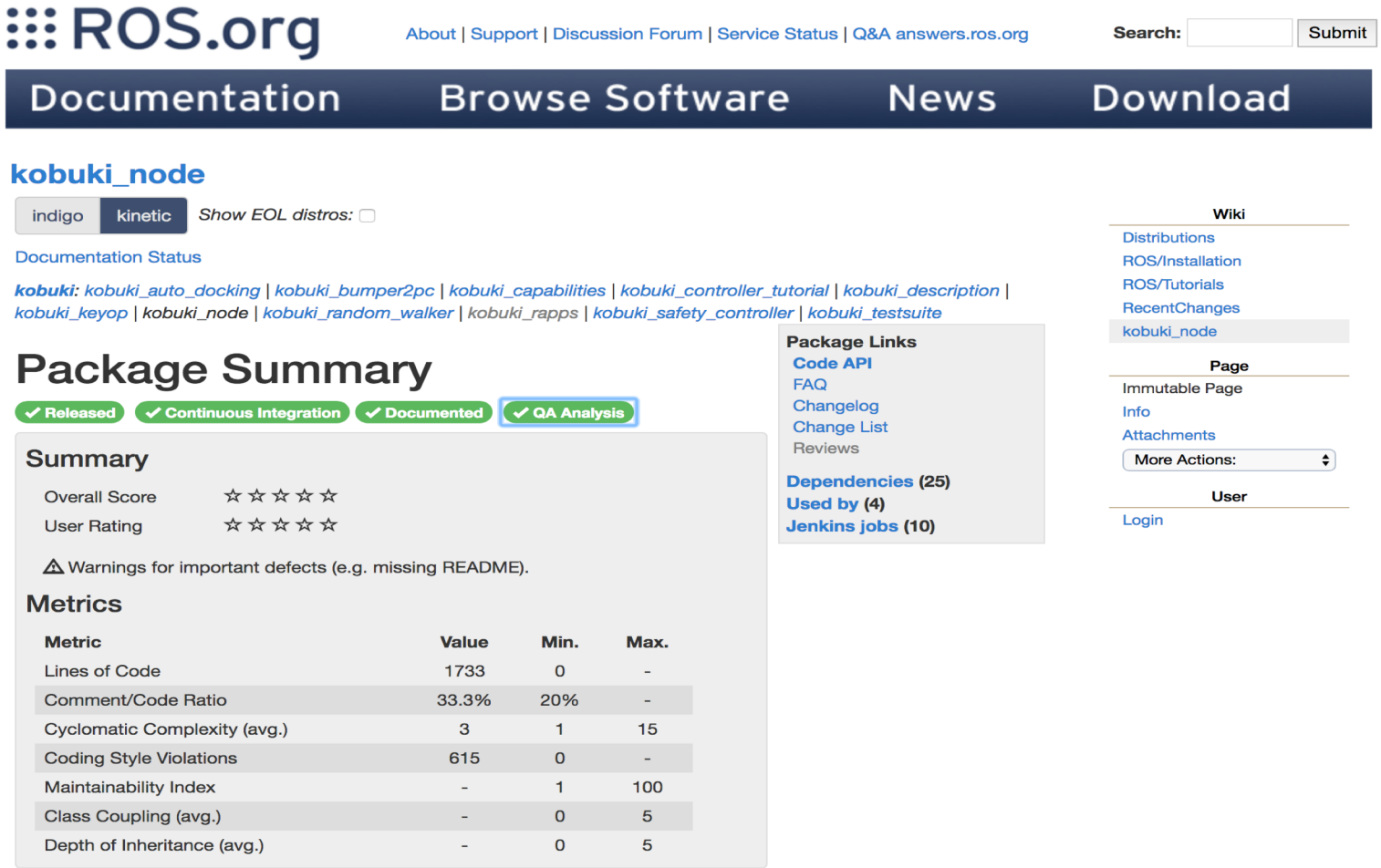


ROSIN Quality Assurance Initiatives

- Make ROS packages quality visible.

ROSIN Quality Assurance Initiatives

- Make ROS packages quality visible.



ROSIN Quality Assurance Initiatives

- Appoint Ownership.
  - Description of the initiative:
    - Appoint ownerships for QA practices, tools and infrastructure.
  - Intent:
    - Establishing ownership of QA practices to ensure continuity.
  - Status:
    - Implemented. A decision has been made by the ROS Quality Working Group to ask the individual(s) who work on the implementation of the initiative to be the default owners.

ROSIN Quality Assurance Initiatives

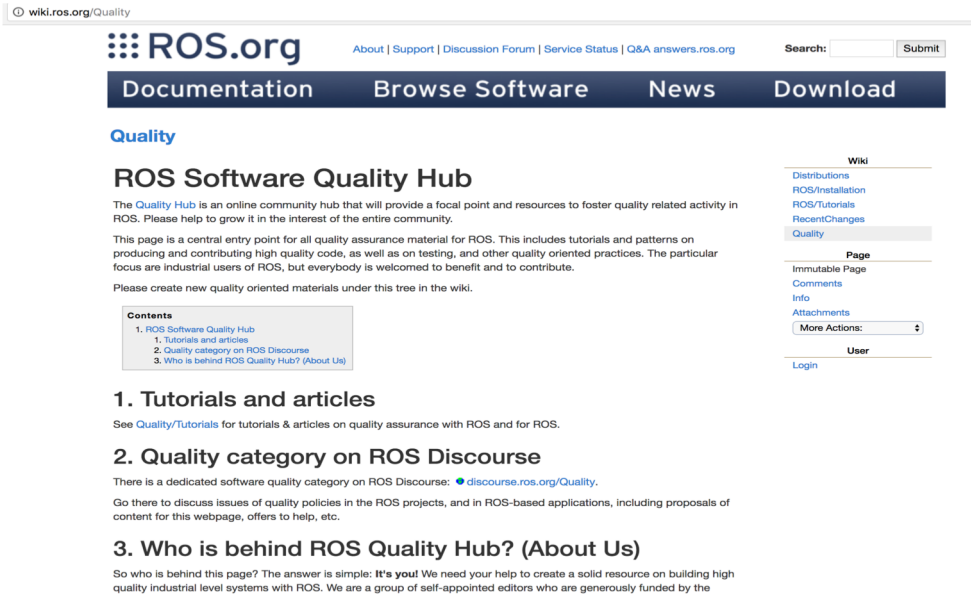
- Energize the code review process.
  - Description of the initiative:
    - Reinstitute the code review practice.
  - Intent:
    - Code Review is a prominent QA practice in open source
  - Status:
    - In progress. A decision has been made to use within same repository/organization review. This to ensure the right expertise of reviewers.
  - To do(s):
    - Code Review Guideline
    - Commence the pilots in two repositories (MoveIt and rviz)

ROSIN Quality Assurance Initiatives

- Implement a code scanning method and tool.
  - Description of the initiative:
    - Haros will be able to build a representation of ROS software code that include Python and not only C++.
    - Improve this representation with reliable name resolution.
  - Status:
    - The build work has commenced.

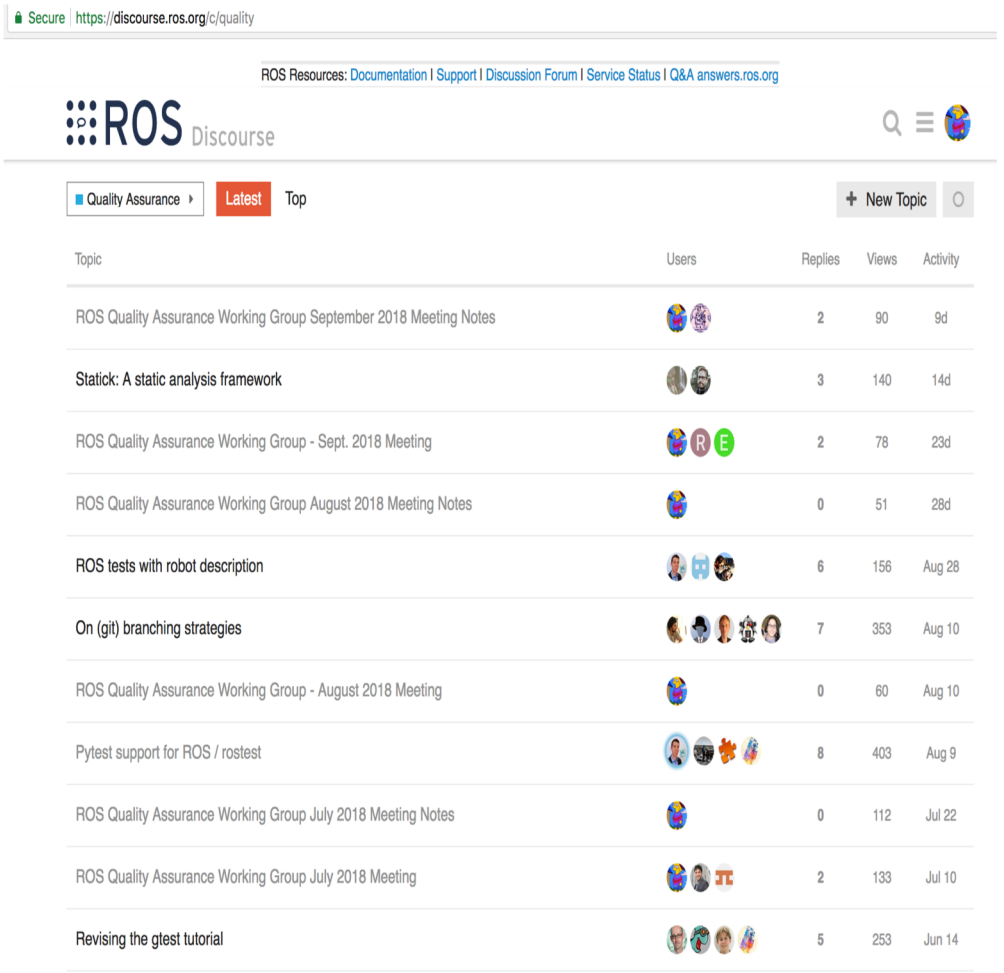
ROSIN Quality Assurance Initiatives

- Quality Hub website.
  - Description of the initiative:
    - A central “go-to” place for QA (like Mozilla) knowledge sharing (documentation of QA practices)
  - Intent:
    - Create a source of knowledge for quality assurance
  - Status:
    - Implemented
    - http://wiki.ros.org/Quality
  - To do(s):
    - Contributions to the hub.



ROSIN Quality Assurance Initiatives

- Quality Forum.
  - Description of the initiative:
    - Quality Assurance Forum (follows the mozilla model).
  - Intent:
    - A dedicated forum to discuss quality.
  - Status:
    - Implemented
    - https://discourse.ros.org/c/quality



ROSIN Quality Assurance Initiatives

Iterations	ROSIN Initiatives	No. Of vote points
Iteration 1	Make ROS packages quality visible.	42
	Appoint ownership	32
	Energize the code review process	32
	Implement a code scanning method and tool	31
	Maintenance issues	31
Iteration 2	Energize Continuous Integration	31
	Quality Hub website	30
	Formalize the code ownership process	29
	Onboarding process for core and non-core community members	29
Iteration 3	Model-in-the-Loop testing	28
	Implement a continuous improvement process	26
	Automated unit test generation	26
	Quality Discourse	23
Iteration 4	QA promotion events	19
	Model Driven Development	18
	#ROSQA	15

Questions