

Node.js Bindings for ROOT6

November 18, 2015

Contents

1	Purpose	2
1.1	Must-have criteria	2
1.2	Nice-to-have criteria	2
1.3	Criteria for demarcation	2
2	Product usage	3
2.1	Applications	3
2.2	Audience	3
2.3	Operating conditions	3
3	Product environment	4
3.1	Software	4
3.2	Hardware	4
4	Product functions	5
5	Product data	6
6	Product deliverables	7
7	Product interface	8
8	Global testcases	9
9	Quality assurance	10
10	Appendix	11

1. Purpose

Project Goal The goal of this project is to create Node.js bindings for ROOT, thanks to which it will become possible to e.g. integrate ROOT into Node-based Web applications. We aim specifically at ROOT 6 because its LLVM-based C++ interpreter Cling offers many advantages over the one available in older ROOT versions.

1.1 Must-have criteria

The bindings should:

- work on Linux
- allow the user to interact with any ROOT class from the Node.js JavaScript interpreter
- accept C++ code for just-in-time compilation
- update dynamically following changes to C++ internals
- provide asynchronous wrappers for common I/O operations (i.e. file and tree access)

1.2 Nice-to-have criteria

The bindings should:

- support the streaming of data in JSON format compatible with JavaScript ROOT
- work OS independent (i.e. support MAC and Windows operating systems)

1.3 Criteria for demarcation

The bindings should not:

- add any extending functionality to the existing ROOT framework.
-

2. Product usage

2.1 Applications

2.2 Audience

2.3 Operating conditions

3. Product environment

3.1 Software

3.2 Hardware

4. Product functions

/BID/	Description
-------	-------------

5. Product data

6. Product deliverables

/FID/	Description
-------	-------------

7. Product interface

/LID/	Description
-------	-------------

8. Global testcases

/TID/	Description
-------	-------------

9. Quality assurance

10. Appendix