NATIONAL AERONAUTICS
AND SPACE ADMINISTRATION

Reuse Readiness Levels (RRLs) – A Work in Progress

Robert Wolfe (Robert.E.Wolfe@nasa.gov)
James Marshall (James.J.Marshall@nasa.gov)
NASA Earth Science Data Systems (ESDS)
Software Reuse Working Group (WG)

January 10, 2008
Winter 2008 ESIP Federation Meeting

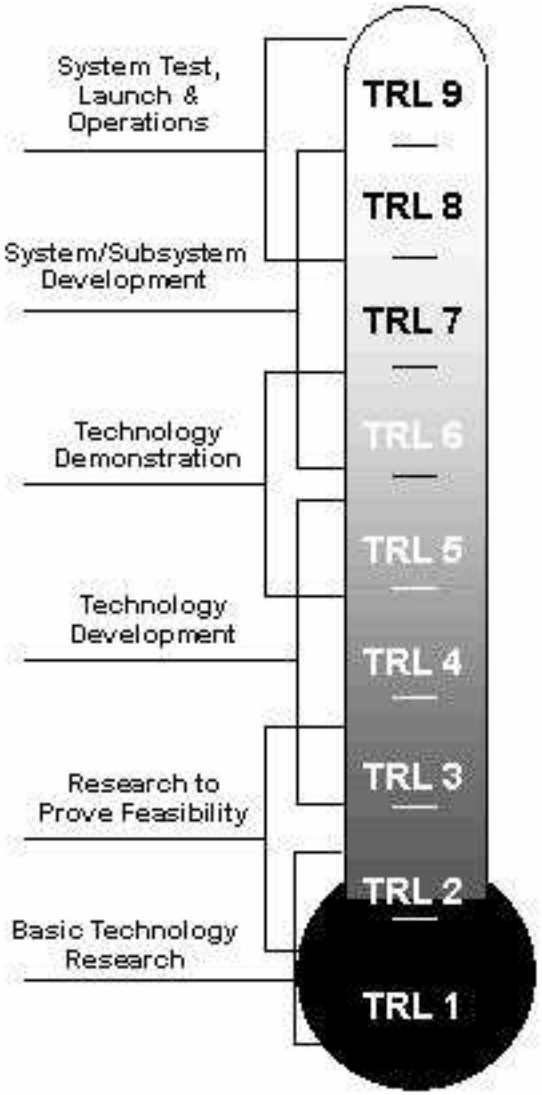
Background – TRLs

- Technology Readiness Levels (TRLs) and similar measures can be used to evaluate the maturity of a particular technology.
  - NASA TRLs range from 1 to 9, going from basic principles to mission proven.
  - While designed more for hardware, these have also been applied to software (see next slide).
- These measures typically do not consider reuse/reusability, or do so only in a limited manner.
  - The emphasis is the maturity of the technology as a whole.
  - The Open Process Framework's Technology Readiness Assessment is one of the few that includes reuse, but only in terms of reused critical technologies.

Technology Readiness Levels

Applied to Software

(v4 5/6/99)



TRL 9: Actual system "flight proven" through successful mission operations *Thoroughly debugged software. Fully integrated with operational hardware/software systems. All documentation completed. Successful operational experience. Sustaining software engineering support in place. Actual system fully demonstrated.*

TRL 8: Actual system completed and "flight qualified" through test and demonstration (Ground or Flight) *Thoroughly debugged software. Fully integrated with operational hardware and software systems. Most user documentation, training documentation, and maintenance documentation completed. All functionality tested in simulated and operational scenarios. V&V completed.*

TRL 7: System prototype demonstration in a space environment *Most functionality available for demonstration and test. Well integrated with operational hardware/software systems. Most software bugs removed. Limited documentation available.*

TRL 6: System/subsystem model or prototype demonstration in a relevant environment (Ground or Space) *Prototype implementations on full scale realistic problems. Partially integrated with existing hardware/software systems. Limited documentation available. Engineering feasibility fully demonstrated.*

TRL 5: Component and/or breadboard validation in relevant environment *Prototype implementations. Experiments with realistic problems. Simulated interfaces to existing systems.*

TRL 4: Component and/or breadboard validation in laboratory environment *Standalone prototype implementations. Experiments with full scale problems or data sets.*

TRL 3: Analytical and experimental critical function and/or characteristic proof-of-concept *Limited functionality implementations. Experiments with small representative data sets. Scientific feasibility fully demonstrated.*

TRL 2: Technology concept and/or application formulated *Basic principles coded. Experiments with synthetic data. Mostly applied research.*

TRL 1: Basic principles observed and reported *Basic properties of algorithms, representations & concepts. Mathematical formulations. Mix of basic and applied research.*

http://isd.gsfc.nasa.gov/Technology/TRL/TRL.ppt

Comments to kswanson@mail.arc.nasa.gov

Introduction to RRLs

- The issue of how to measure the maturity of software—in a reusability sense—was discussed at the 2006 ESDSWG Meeting.
- Having a measure of the reusability of an asset:
  - Provides potential reusers with additional information about the reuse maturity of the asset.
    - It lets them know what they're getting, and
    - Gives them a basic feel for what modifications may be needed.
  - Helps potential reusers make better informed choices about
    - What to reuse, and
    - What best meets their needs.
- *This measure can be used as a piece of metadata for assets placed in a Reuse Enablement System (RES), or anywhere else.*
- The Software Reuse WG is developing a set of Reuse Readiness Levels (RRLs) to measure the maturity of a technology with respect to reusability.

Developing the RRLs

- Through discussions on weekly and monthly telecons, the Software Reuse WG made the following decisions:
  - To use nine levels, to align with the familiar TRL scale.
  - To look at nine topic areas that we felt were important for measuring the reuse maturity of software.
- Volunteers from our WG:
  - Wrote an initial set of levels for each topic (2+ people per topic), and
  - Drafted summaries of each RRL, looking at the levels for all topic areas.
- During the 2007 ESDSWG Meeting, we discussed:
  - The level summaries, to determine what to use as the RRL scale, and
  - The longer descriptions for each level in the RRL scale.
- Some of our planned future work includes:
  - Completing the RRL discussion and revisions, and writing a document on the RRLs,
  - Publicizing the RRLs (post on our web site, present at meetings, etc.),
  - Developing one or more RRL calculator(s), possibly allowing users to apply their own weights to different topic areas, and
  - Submitting the work to NASA HQ (and maybe the Standards WG) as a recommendation.

RRL Topic Areas and Levels

Topic areas included:

- Documentation
- Extensibility
- Intellectual Property Issues
- Modularity
- Packaging
- Portability
- Standards compliance
- Support
- Verification and Testing

Example from Verification and Testing

RRL 4 – Software application tested and validated in laboratory environment

Following successful testing of inputs and outputs, the testing would include integrating an application to establish that the “pieces” will work together to achieve concept-enabling levels. This validation must be devised to support the concept that was formulated earlier and should also be consistent with the requirements of potential system applications. The validation is relatively “low-fidelity” compared to the eventual system: it could be composed of ad hoc discrete components in a laboratory; for example, an application tested with simulated inputs.

NATIONAL AERONAUTICS
AND SPACE ADMINISTRATION
ESDS Reuse Working Group

Proposed RRL Topic Levels

Level	Documentation	Extensibility	Intellectual Property Issues	Modularity	Packaging	Portability	Standards Compliance	Support	Verification and Testing
1	Limited internal documentation available	No ability to extend or modify program behavior	Potential owners and stakeholders of product have been identified.	No designs for modularity or reuse	Source code available	The software is not portable at any cost	Follows no particular standard	No support available	No testing performed
2	Fully commented source code available	Prohibitive costs and efforts need to modify or extend the system	Relevant intellectual policies of potential owners and stakeholders have been reviewed.			Some parts of the software may be portable	Follows some parts of common standards and best practices	Known contact available	Software application formulated and unit testing performed
3	Basic external documentation available	Can be extended with the input of considerable time and effort on par with recreating system separately	Intellectual property agreements have been proposed to potential stakeholders.	Modularity at major system or subsystem level only	Detailed installation instructions available	The software is only portable with significant costs	Follows a company-wide standard for development and testing	Original developers provide proactive support	Testing includes testing for error conditions and proof of handling input errors
4	Reference manual available	Can be modified and extended through configuration changes, minimal modification of source	Potential stakeholders have negotiated on intellectual property agreements and authorship issues.			The software may be portable at a reasonable cost	Most components follow a complete, universal standard, but not validated	Latest updates or patches are available but not very frequently	Software application demonstrated in a laboratory environment
5	User manual available	Consideration for future extensibility designed into system, extensibility approach somewhat defined	Agreement and approval on authorship, attribution, and intellectual property issues has been obtained from stakeholders.	Partial segregation of generic and specific functionality	Software is easily configurable for different environments	The software is moderately portable	All components follow a universal standard, but only partially validated	Informal user community available	Software application tested and validated in a laboratory environment
6	Tutorials available	Designed from the start to allow easy extensibility, provides many points of extensibility and a thorough and detailed extensibility plan	Authorship, attribution, and intellectual property statements have been drafted to reflect agreement among stakeholders on intellectual property and authorship.			The software is portable	Validated to follow a specific proprietary standard	Centralized support available	Software application demonstrated in a relevant environment (Earth science related)
7	Interface guide available	Proven to be extensible internally, code structured to provide loose coupling and high cohesion	Authorship and intellectual property statements included in product prototype.	Clear delineations of specific and reusable components	OS detect and auto-build for supported platforms	The software is highly portable	Validated to comply to a specific open standard	Organized/defined support by the original developer available	Software application tested and validated in a relevant environment (Earth science related)
8	Extension guide and/or Design/Development guide available	Proven extensibility on a major external program, provides a clear plan for modifying and extending features	Manifestation of authorship, attribution, and intellectual property statements reviewed in product prototype before product release.				Proven by validation to comply with a “gold” standard	Support by organization available	Software application "qualified" through test and demonstration (meets requirements) and successfully delivered to the Earth science environment
9	Full software lifecycle engineering design documentation available	Proven extensibility in multiple scenarios, provides specific documentation and features to build extensions	Reviewed authorship, attribution, and intellectual property statements packaged with product for release.	All functions and data encapsulated into objects or accessible through web service interfaces	GUI installation environment provided	The software is completely portable	“Gold” standard compliance of entire system and development, independently validated	Large user community with well-defined support available	Actual software application tested and validated through successful use of application output

Example 1: Documentation Topic Levels

Level	Documentation Level Summary
1	Limited internal documentation available
2	Fully commented source code available
3	Basic external documentation available
4	Reference manual available
5	User manual available
6	Tutorials available
7	Interface guide available
8	Extension guide and/or Design/Development guide available
9	Full software lifecycle engineering design documentation available

Example 2: Support Levels

Level	Support Level Summary
1	No support available
2	Known contact available
3	Original developers provide proactive support
4	Latest updates or patches are available but not very frequently
5	Informal user community available
6	Centralized support available
7	Organized/defined support by the original developer available
8	Support by organization available
9	Large user community with well-defined support available

Example 3: Intellectual Property Levels

Level	Intellectual Property Level Summary
1	Potential owners and stakeholders of product have been identified.
2	Relevant intellectual policies of potential owners and stakeholders have been reviewed.
3	Intellectual property agreements have been proposed to potential stakeholders.
4	Potential stakeholders have negotiated on intellectual property agreements and authorship issues.
5	Agreement and approval on authorship, attribution, and intellectual property issues has been obtained from stakeholders.
6	Authorship, attribution, and intellectual property statements have been drafted to reflect agreement among stakeholders on intellectual property and authorship.
7	Authorship and intellectual property statements included in product prototype.
8	Manifestation of authorship, attribution, and intellectual property statements reviewed in product prototype before product release.
9	Reviewed authorship, attribution, and intellectual property statements packaged with product for release.

Draft RRL Summaries

RRL	RRL Summary
1	No reusability – software is not reusable (or is not recommended for reuse)
2	Initial reusability – software reuse is not practical
3	Basic reusability – software might be reusable by skilled users at substantial effort, cost, and risk
4	Reuse is possible – software might be reused by most users with some effort, cost, and risk
5	Reuse is practical – software could be reused by most users with reasonable cost and risk
6	Software is reusable – software can be reused by most users although there may be some cost and risk
7	Software is highly reusable – software can be reused by most users with minimum cost and risk
8	Demonstrated reusability – software has been reused by multiple users
9	Proven reusability – software is being reused by many classes of users over a wide range of systems

NATIONAL AERONAUTICS
AND SPACE ADMINISTRATION
ESDS Reuse Working Group

Draft RRL Descriptions

RRL	RRL Description
1	Little is provided beyond limited source code or pre-compiled, executable binaries. There is no support, contact information, author attribution, or rights specified, the software is not extensible, and there is inadequate or no documentation.
2	Some source code, documentation, and contact information are provided, but these are still very limited. Initial testing has been done, but authorship and reuse rights are still unclear. Reuse would be challenging and cost-prohibitive.
3	Software has some modularity and standards compliance, intellectual property agreements have been proposed, some support is provided by developers, and detailed installation instructions are available, but rights are unspecified. An expert may be able to reuse the software, but general users would not.
4	Software and documentation are complete and understandable. Software has been demonstrated in a lab on one or more specific platforms, infrequent patches are available, and intellectual property issues have been negotiated. Reuse is possible, but may be difficult.
5	Software is moderately portable, modular, extendable, and configurable, has low-fidelity standards compliance, a user manual, and has been tested in a lab. A user community exists, but may be a small community of experts. Authorship and rights are not specified.
6	Software has been designed for extensibility, modularity, and portability, but software and documentation may still have limited applicability. Tutorials are available, and the software has been demonstrated in a relevant environment. Intellectual property statements have been drafted, but authorship and rights have not been formalized.
7	Software is highly portable and modular, has high-fidelity standards compliance, provides auto-build installation, and has been tested in a relevant environment. Support is developer-organized, and an interface guide is available. Software and documentation are applicable for most systems.
8	Software has been shown to be extensible, and has been qualified through test and demonstration. An extension guide and organization-provided support are available. Intellectual property is reviewed in the product before release, and authorship and rights are specified.
9	Software is fully portable and modular, with all appropriate documentation and standards compliance, encapsulated packaging, a GUI installer, and a large support community that provides patches. Software has been tested and validated through successful use of application output. Complete and clear attribution and permissions for implementation by various user levels are available.

Factors Under Consideration

- We have received feedback on the draft RRLs at the 2007 ESDSWG Meeting (October) and the 2007 Fall AGU Meeting (December).
- Suggested areas for improvement include:
  - *Security*: Could this be incorporated into verification/testing, should it be its own topic area, or is it not a factor of reusability?
  - *Use vs. reuse*: When is a factor more about how good it is for your application (use) than is it ready for you to use (reuse)? How much should use be considered? Higher reusability may sacrifice out-of-the-box usability.
  - *Quantitative measures*: to make the ratings easier to determine, with less ambiguity, more objective level criteria are needed; also, how to maintain consistency, should (self-)assessments be audited and how?
  - *Cost and Risk*: how to factor in these concerns?
  - *Topic level ratings*: these are viewed as useful information for reusers, so how should the information be offered?
  - *Target audience*: should be clarified since engineers may prefer the nine topic levels, but project managers the overall summaries.
  - *Various types of reuse*: how to consider things like black box reuse, white box reuse, reuse over long periods of time, reuse in virtual machines, etc.?

Possible Discussion Points

- Do we have appropriate topics?
- Should we delete or add any topics?
- Do we have an appropriate number of levels?
- Should any levels be split up or combined?
- Any feedback about our initial combination into a single RRL scale?
- Are there quantitative measures for each level?

Backup

References on TRLs

Here are links to a number of documents on TRLs and other measures:

- http://www.hq.nasa.gov/office/codeq/trl/trl.pdf
- http://esto.nasa.gov/files/TRL_definitions.pdf
- http://isd.gsfc.nasa.gov/Technology/TRL/TRL.ppt
- http://www.stsc.hill.af.mil/crosstalk/2005/05/0505Gold.pdf
- http://www.opfro.org/index.html?Components/WorkProducts/ArchitectureSet/TechnologyReadinessAssessment/TechnologyReadinessAssessment.html~Contents
- http://www.sei.cmu.edu/pub/documents/02.reports/pdf/02sr027.pdf
- http://csdl2.computer.org/comp/proceedings/hicss/2005/2268/09/22680315a.pdf and http://www.sei.cmu.edu/pub/documents/04.reports/pdf/04tr013.pdf
- http://www.iccbss.org/2004/proceedings/ImpACT.pdf
- http://www.openbrr.org/docs/BRR_whitepaper_2005RFC1.pdf
- http://www.hq.nasa.gov/office/codeq/trl/r&d3.pdf
- http://www.dtic.mil/ndia/2003systems/nolte.ppt
- https://acc.dau.mil/CommunityBrowser.aspx?id=25811