Level	Description
9	- Actual application of the technology in its final form and under mission/operational conditions, such as those encountered in operational test and evaluation. Technology is ready for commercial deployment. - Software: readily repeatable and reusable. The software based on the technology is fully integrated with operational hardware/software systems. All software documentation verified. Successful operational experience. Sustaining software engineering support in place.
8	- Technology has been proven to work in its final form and under expected conditions. In almost all cases, this TRL represents the end of true system development - Software fully integrated with operational hardware and software systems, development documentation is complete. All functionality tested in simulated and operational scenarios.
7	- Prototype near or at planned operational system. Requiring demonstration of an actual system prototype in an operational environment (e.g., in an aircraft, in a vehicle, or in space). Normally only performed when the technology and/or subsystem is mission critical and relatively high risk. - Critical technological properties are measured against requirements in an operational environment. - Readiness in an operational environment requires evidence of the acceptable performance under operational factors, including, for example for a software system loading, user interaction, security etc.
6	- Representative model or prototype system, tested in a relevant environment. Represents a major step up and requires evidence of performance on full-scale, realistic problems. - For software: level at which the engineering feasibility of a software is demonstrated. This level extends to laboratory prototype implementations on full-scale realistic problems in which the software technology is partially integrated with existing hardware/software systems. - Examples: testing a prototype in a high-fidelity lab environment or simulated operational environment.
5	- Basic technological components integrated with reasonably realistic supporting elements so they can be tested in a simulated environment. Fidelity of breadboard technology increases significantly. - Integrated components provide a representation of a system/subsystem for to determining concept feasibility and to develop technical data. Lab use to validate the technical principles of interest. - Software: Module and/or subsystem validation in relevant environment. Ready to start integration with existing system, conforms to target environment/interfaces. System software architecture established and all components and elements affecting the operation of the critical software element. - Examples: a new type of solar photovoltaic material promising higher efficiencies would at this level be used in an actual fabricated solar array that would be integrated with power supplies, supporting structure, etc., and tested in a thermal vacuum chamber with solar simulation capability.
4	- Basic technological components are integrated to establish that they will work together. This is relatively “low fidelity” compared with the eventual system. System concepts considered and results from testing laboratory scale breadboard(s). Only limited and initial information about the end product function. - Software: module and/or subsystem validation in a laboratory environment (i.e. software prototype development environment). Basic software components are integrated to establish that they will work together. Architecture development initiated (e.g. interoperability, reliability). - Example: demo of a ‘fuzzy logic’ approach to avionics by testing algorithms in a partially computer-based, partially bench-top components to demo in a controls lab using simulated vehicle inputs.
3	- Active R&D is initiated. Analytical studies and laboratory-based studies to physically validate that analytical predictions are correct. Laboratory tests are performed to measure parameters of interest and comparison to analytical predictions for critical subsystems. - Software: limited functionality environments to validate critical properties/analytical predictions using non-integrated software components and partially representative data. - Example: super-cooled hydrogen as a propellant where the concept-enabling temperature/pressure for the fluid was achieved in a lab. Software algorithms run on a surrogate processor in lab environment.
2	- Invention begins. Once basic principles are observed, practical applications can be invented. Applications are speculative, no proof or detailed analysis to support the assumptions. - Software: analytic studies, studies on synthetic date, small code units - Example: observation of high critical temperature superconductivity, potential applications of the new material in instruments (e.g. telescope sensors) defined.
1	- Published research that identifies the principles that underlie this technology. Scientific research begins to be translated into applied research and development. - Software: development of basic use, basic properties of software architecture, mathematical formulations, and general algorithms. - Example: studies of basic properties e.g. tensile strength as a function of temperature for a new material